



Kernel Based High Order “Explicit” Unconditionally Stable Scheme for Nonlinear Degenerate Advection-Diffusion Equations

Andrew Christlieb^{1,2} · Wei Guo³ · Yan Jiang⁴ · Hyoseon Yang¹

Received: 22 May 2019 / Accepted: 1 February 2020 / Published online: 12 February 2020
© Springer Science+Business Media, LLC, part of Springer Nature 2020

Abstract

In this paper, we present a novel numerical scheme for solving a class of nonlinear degenerate parabolic equations with non-smooth solutions. The proposed method relies on a special kernel based formulation of the solutions found in our early work on the method of lines transpose and successive convolution. In such a framework, a high order weighted essentially non-oscillatory methodology and a nonlinear filter are further employed to avoid spurious oscillations. High order accuracy in time is realized by using the high order explicit strong-stability-preserving (SSP) Runge-Kutta method. Moreover, theoretical investigations of the kernel based formulation combined with an explicit SSP method indicate that the combined scheme is unconditionally stable and up to third order accuracy. Evaluation of the kernel based approach is done with a fast $\mathcal{O}(N)$ summation algorithm. The new method allows for much larger time step evolution compared with other explicit schemes with the same order accuracy, leading to remarkable computational savings.

A. Christlieb: Research is supported in part by AFOSR grants FA9550-12-1-0343, FA9550-12-1-0455, and FA9550-15-1-0282, and NSF Grant DMS-1418804. W. Guo: Research is supported in part by NSF Grants DMS-1620047, DMS-1830838. Y. Jiang: Research is supported in part by NSFC Grant 11901555.

✉ Yan Jiang
jiangy@ustc.edu.cn

Andrew Christlieb
christli@msu.edu

Wei Guo
weimath.guo@ttu.edu

Hyoseon Yang
hyoseon@msu.edu

¹ Department of Computational Mathematics, Scienc and Engineering, Michigan State University, East Lansing, MI 48824, USA

² Department of Mathematics and Department of Electrical Engineering, Michigan State University, East Lansing, MI 48824, USA

³ Department of Mathematics and Statistics, Texas Tech University, Lubbock, TX 79409, USA

⁴ School of Mathematical Sciences, University of Science and Technology of China, Hefei 230026, Anhui, People’s Republic of China

Keywords Integral solution · Unconditionally stable · Weighted essentially non-oscillatory methodology · High order accuracy · Nonlinear degenerate advection-diffusion equation

1 Introduction

In this paper, we are interested in numerically solving the following nonlinear, possibly degenerate, parabolic equation

$$u_t + f(u)_x = g(u)_{xx}, \quad x \in [a, b], \quad (1.1)$$

where $g'(u) \geq 0$ and $g'(u)$ can vanish for some values of u . Such an equation arises in a wide range of applications, e.g., collisional transport models in plasmas, radiative transport, porous medium flow, etc. The Eq. (1.1) has similar properties to hyperbolic conservation laws, including possible existence of discontinuous solutions and sharp fronts, and a finite speed of propagation of wave fronts. When considering models of the form of Eq. (1.1), it is necessary to design a numerical scheme capable of capturing these features.

A variety of schemes have been developed in the literature, e.g., finite volume schemes [8], finite difference schemes [36], local discontinuous Galerkin methods [46], kinetic schemes [3], and relaxation schemes [16], among others. Most of these methods are in the method of lines (MOL) framework, meaning that the spatial variable is first discretized, then the numerical solution is updated in time by coupling a suitable time integrator. The most commonly used time evolution methods are the strong-stability-preserving Runge-Kutta (SSP RK) schemes and SSP multi-step schemes [24,25,42]. SSP methods preserve the strong stability in some desired norm of an appropriate spatial discretization in conjunction with the forward Euler time stepping, thus preventing spurious oscillations near spatial discontinuities. However, it is well known that an explicit time discretization does have a restriction on the time step in order to maintain stability. For example, for advection problems, maintaining stability usually requires the time step $\Delta t \propto \Delta x$, where Δx is the spatial mesh size. Solving diffusion problems with explicit time stepping methods introduces a more stringent restriction $\Delta t \propto \Delta x^2$ for stability. Hence, using an explicit SSP method for Eq. (1.1) means that satisfying the stability condition demands $\Delta t \propto \Delta x^2$. There are two other approaches one can utilize in these situations including Implicit-Explicit (IMEX) methods and fully implicit methods [5,28]. Both of these approaches are effective in the sense that they permit larger time steps without stability issue. Except for the backwards Euler method and the second order fully implicit method proposed by Ketcheson [32], this class of methods often requires that $\Delta t \propto \Delta x$ to maintain non-oscillatory numerical solutions near discontinuities or steep gradients that arise from the Eq. (1.1). Meanwhile, in this approach, one needs to invert matrices or nonlinear operators resulting from the spatial discretization at each time step. This typically involves the use of some form of iterative solvers, Krylov or multi-grid methods. In practice, full matrix inversions for problems in the form of Eq. (1.1) may become prohibitively complicated and costly, especially when memory is extremely limited.

An alternative approach to solving (1.1) is the method of lines transpose (MOL^T), also known as Rothe's method or transverse method of lines [13,39,40] in the literature. In the MOL^T framework, the discretization is first carried out for the temporal variable, resulting in a boundary value problem (BVP) at discrete time levels. Then a preferred BVP solver is applied to advance the numerical solution. A notable advantage of the MOL^T approach is that an implicit method, e.g. the backward Euler method, can be used in the first step, and then in the second step, the operator of the BVP, e.g. the modified Helmholtz operator for solving wave

equations, is inverted analytically using an integral formulation based on a Green's/kernel function. Because the method utilizes direct inversion of the operator, the MOL^T approach eliminates the need to solve linear systems at each time step. Moreover, many well-established fast convolution algorithms can be readily used in reducing the computational complexity of the scheme from $\mathcal{O}(N^2)$ to $\mathcal{O}(N)$ with N being the number of discrete mesh points [7, 15, 26]. In [12, 14], a novel technique known as successive convolution (or resolvent expansions) is developed and analyzed for the wave equation and the Allen-Cahn equation. The resulting scheme is unconditionally stable, but would be rarely applied on general nonlinear problems possibly due to the complex formulation. Additionally, the MOL^T framework has been studied to solve the linear and non-linear heat equation [11, 12, 30, 33], Maxwell's equations [17] as well as others. Recently, an implicit high order SSP RK method and a robust WENO integral formulation have been incorporated into the MOL^T framework so that the method can be applied to the advection equation and the Vlasov equation [20]. With the help of the SSP property and the WENO based quadrature, the method is able to take large time steps, and at the same time, generate a solution being free of oscillations. Unfortunately, this scheme as designed does not directly extend to problems in the form of Eq. (1.1).

In this paper, we will propose a novel numerical scheme following the MOL^T philosophy of employing a kernel based approach for degenerate parabolic equations of the form (1.1). The major distinction between the newly proposed scheme and our previous work [20], is that we employ explicit time stepping methods that are traditionally used in the MOL formulation. In particular, we start with explicit SSP RK methods for the time discretization, which can be easily applied to nonlinear problems. Following the idea of solving the BVP as in the MOL^T framework, we transform the spatial derivatives into a kernel based representation. This approach makes the method effectively “implicit” at each stage of the explicit SSP RK method, but without the need to invert any matrices. The robust WENO methodology in conjunction with a new nonlinear filter is adopted with the aim to effectively capture sharp gradients of the solution without producing spurious oscillations. In addition, a special parameter β is introduced in the scheme formulation and we are able to make this scheme A-stable through a careful choice of β . The unconditional stability property of the scheme can be established accordingly. In summary, the proposed scheme we have designed for solving Eq. (1.1) is robust, high order accurate (up to third order), matrix free, unconditionally stable, and efficient.

The paper is organized as follows. In Sect. 2, we represent the spatial derivatives as infinite series, in which each term relies on a special kernel based formulation of the solution. In Sect. 3, the approximation accuracy of the associated partial sum is studied. We introduce the WENO-based quadrature as well as the nonlinear filter for evaluation of the partial sum in Sect. 4. The fully-discrete scheme for solving (1.1) is designed by coupling the partial sum formulation with the high order explicit SSP RK method, and a stability analysis for linear problems is established in Sect. 5. This method can be extended directly to high dimensional problems, and we discuss the details of the two-dimensional formulation in Sect. 6. We present several numerical tests in Sect. 7 to verify the performance of the proposed scheme. Finally, we conclude with a brief discussion in Sect. 8.

2 Representation of Differential Operators

In this section, we will review a class of representations of the first spatial derivative ∂_x and the second spatial derivative ∂_{xx} from the nonlinear convection-diffusion Eq. (1.1). Such

representations are based on a successive convolution of the underlying kernel functions and will serve as the key building block of the proposed scheme. Below, we first introduce the operator \mathcal{L} and the associated operator \mathcal{D} . Then, both differential operators ∂_x and ∂_{xx} are represented by infinite series of \mathcal{D} . Such an idea was first used in [14] for designing an MOL^T A-stable scheme for solving linear wave equations. If the underlying equation is written in the form of $u_t + f(u)_x = (v(u)u_x)_x$, we then first convert it into the original form (1.1), i.e. $u_t + f(u)_x = g(u)_{xx}$, where the function $g(u)$ satisfies $g'(u) = v(u)$.

2.1 The Second Order Derivative ∂_{xx}

We consider the following differential operator:

$$\mathcal{L}_0 = \mathcal{I} - \frac{1}{\alpha^2} \partial_{xx}, \quad x \in [a, b], \quad (2.1)$$

where \mathcal{I} is the identity operator and $\alpha > 0$ is a constant. Suppose $w(x)$ satisfies the differential equation

$$\mathcal{L}_0[w, \alpha](x) = w(x) - \frac{1}{\alpha^2} w_{xx}(x) = v(x), \quad (2.2)$$

where $v(x)$ is a given function. Then, by analytically inverting operator \mathcal{L}_0 , we can obtain the explicit expression of $w(x)$ as

$$w(x) = \mathcal{L}_0^{-1}[v, \alpha](x) = I^0[v, \alpha](x) + A_0 e^{-\alpha(x-a)} + B_0 e^{-\alpha(b-x)}, \quad (2.3)$$

where

$$I^0[v, \alpha](x) := \frac{\alpha}{2} \int_a^b e^{-\alpha|x-y|} v(y) dy, \quad (2.4)$$

and A_0 and B_0 are constants determined by the boundary conditions, see [13]. For example, for periodic boundary conditions, i.e., $w(a) = w(b)$ and $w_x(a) = w_x(b)$, we have

$$A_0 = \frac{I^0[v, \alpha](b)}{1 - \mu} \quad \text{and} \quad B_0 = \frac{I^0[v, \alpha](a)}{1 - \mu}, \quad (2.5)$$

with $\mu = e^{-\alpha(b-a)}$.

We then define the operator \mathcal{D}_0 as

$$\mathcal{D}_0 = \mathcal{I} - \mathcal{L}_0^{-1}. \quad (2.6)$$

Clearly, $\mathcal{L}_0 = (\mathcal{I} - \mathcal{D}_0)^{-1}$. Moreover, by the definition (2.1), the second derivative can be rewritten as

$$\frac{1}{\alpha^2} \partial_{xx} = \mathcal{I} - \mathcal{L}_0 = \mathcal{L}_0(\mathcal{L}_0^{-1} - \mathcal{I}) = -\mathcal{D}_0(\mathcal{I} - \mathcal{D}_0)^{-1} = -\sum_{p=1}^{\infty} \mathcal{D}_0^p, \quad (2.7)$$

where \mathcal{D}_0^p is successively defined as $\mathcal{D}_0^p = \mathcal{D}_0[\mathcal{D}_0^{p-1}]$. Hence, $g(u)_{xx}$ from (1.1) can be represented as

$$g(u)_{xx} = -\alpha^2 \sum_{p=1}^{\infty} \mathcal{D}_0^p[g(u), \alpha](x). \quad (2.8)$$

2.2 The First Derivative ∂_x

Following a similar idea, we are able to deal with the first derivative ∂_x as well. Note that, when designing a numerical method for solving a hyperbolic conservation law

$$u_t + f(u)_x = 0, \quad (2.9)$$

we have to consider the propagation direction of the wave solution to ensure stability of the scheme. To this end, we introduce operators \mathcal{L}_L and \mathcal{L}_R and the associated operators \mathcal{D}_L and \mathcal{D}_R to account for waves traveling in opposite directions:

$$\mathcal{L}_L = \mathcal{I} + \frac{1}{\alpha} \partial_x, \quad \mathcal{D}_L = \mathcal{I} - \mathcal{L}_L^{-1}; \quad (2.10)$$

$$\mathcal{L}_R = \mathcal{I} - \frac{1}{\alpha} \partial_x, \quad \mathcal{D}_R = \mathcal{I} - \mathcal{L}_R^{-1}, \quad (2.11)$$

where $x \in [a, b]$ and $\alpha > 0$ is a constant.

Let us consider \mathcal{L}_L and \mathcal{D}_L first. Assume $w(x)$ satisfies the differential equation

$$\mathcal{L}_L[w, \alpha](x) = w(x) + \frac{1}{\alpha} \partial_x w(x) = v(x),$$

where $v(x)$ is a given function. As with \mathcal{L}_0 , we can analytically invert \mathcal{L}_L as follows:

$$\mathcal{L}_L^{-1}[v, \alpha](x) = I^L[v, \alpha](x) + A_L e^{-\alpha(x-a)}, \quad (2.12)$$

where

$$I^L[v, \alpha](x) = \alpha \int_a^x e^{-\alpha(x-y)} v(y) dy, \quad (2.13)$$

and the constant A_L is determined by the boundary condition. For example, for periodic boundary conditions, A_L is taken as

$$A_L = \frac{I^L[v, \alpha](b)}{1 - \mu}. \quad (2.14)$$

Similar to $\frac{1}{\alpha^2} \partial_{xx}$, we are able to represent $\frac{1}{\alpha} \partial_x$ using an infinite series of the operator \mathcal{D}_L

$$\frac{1}{\alpha} \partial_x = \mathcal{L}_L - \mathcal{I} = \sum_{p=1}^{\infty} \mathcal{D}_L^p, \quad (2.15)$$

where \mathcal{D}_L is given in (2.10),

$$\mathcal{D}_L[v, \alpha](x) = v(x) - \alpha \int_a^x e^{-\alpha(x-y)} v(y) dy - A_L e^{-\alpha(x-a)}.$$

Note that, the integral $I_L[v, \alpha](x)$ only depends on the function values of v from the left end point a to x . On the other hand, it is well-known that, for the hyperbolic conservation law (2.9), the information of the solution propagates from left to right over time if the flux function $f(u)$ has a positive derivative, i.e., $f'(u) \geq 0$. Hence, it is reasonable to represent $f(u)_x$ as

$$\partial_x f(u) = \alpha \sum_{p=1}^{\infty} \mathcal{D}_L^p[f(u), \alpha](x). \quad (2.16)$$

Similarly, when $f'(u) \leq 0$, we can represent $f(u)_x$ using the following infinite series of \mathcal{D}_R :

$$\partial_x f(u) = -\alpha \sum_{p=1}^{\infty} \mathcal{D}_R^p[f(u), \alpha](x), \quad (2.17)$$

where \mathcal{D}_R is given in (2.11) with

$$\mathcal{L}_R^{-1}[v, \alpha](x) = I^R[v, \alpha](x) + B_R e^{-\alpha(b-x)}, \quad (2.18)$$

$$I^R[v, \alpha](x) = \alpha \int_x^b e^{-\alpha(y-x)} v(y) dy, \quad (2.19)$$

and B_R is a constant determined by the boundary conditions. For example, for periodic boundary conditions,

$$B_R = \frac{I^R[v, \alpha](a)}{1 - \mu}. \quad (2.20)$$

In the case of $f(u)$ not being a monotone function of u , we employ the following global “flux splitting” strategy:

$$f(u) = f^+(u) + f^-(u), \quad (2.21)$$

with $df^+(u)/du \geq 0$ and $df^-(u)/du \leq 0$. We then use (2.16) to represent $f^+(u)_x$, and (2.17) to represent $f^-(u)_x$. The most commonly used splitting strategy is the Lax-Friedrichs splitting

$$f^{\pm}(u) = \frac{1}{2}(f(u) \pm cu), \quad \text{with } c = \max_u |f'(u)|,$$

which has been widely used in the design of high order finite difference schemes for conservation laws [44].

In summary, $-f(u)_x + g(u)_{xx}$ can be represented as a linear combination of three infinite series:

$$-\alpha_L \sum_{p=1}^{\infty} \mathcal{D}_L^p[f^+(u), \alpha_L] + \alpha_R \sum_{p=1}^{\infty} \mathcal{D}_R^p[f^-(u), \alpha_R] - \alpha_0^2 \sum_{p=1}^{\infty} \mathcal{D}_0^p[g(u), \alpha_0]. \quad (2.22)$$

3 Approximation of Partial Sums

So far, we have shown that the derivatives ∂_x and ∂_{xx} can be represented as infinite series. In numerical simulations, we have to truncate the series and only compute the corresponding partial sum. In particular, (2.22) is approximated by the k^{th} partial sum

$$\begin{aligned} -f(u)_x + g(u)_{xx} \approx & -\alpha_L \sum_{p=1}^k \mathcal{D}_L^p[f^+(u), \alpha_L] + \alpha_R \sum_{p=1}^k \mathcal{D}_R^p[f^-(u), \alpha_R] \\ & - \alpha_0^2 \sum_{p=1}^k \mathcal{D}_0^p[g(u), \alpha_0]. \end{aligned} \quad (3.1)$$

In this section, we will theoretically investigate the truncation error incurred. Below, we restrict our attention to two special cases

1. u is a smooth periodic function

$$u(a, t) = u(b, t), \quad u_x^p(a, t) = u_x^p(b, t), \quad p \geq 1, \quad t \geq 0, \quad (3.2)$$

2. u satisfies following special homogeneous condition

$$\partial_x^p u(a, t) = 0, \quad \partial_x^p u(b, t) = 0, \quad p \geq 1, \quad t \geq 0. \quad (3.3)$$

Note that a function that is constant near the boundary satisfies such a condition.

We remark that (3.2) is a necessary smoothness assumption to attain high order accuracy for the partial sum approximation. The extension to other boundary conditions is considered in our future work.

3.1 Periodic Boundary Conditions

In the case of periodic boundary conditions (3.2), it is straightforward to require

$$\begin{aligned} \mathcal{D}_0^p[g(u), \alpha_0](a) &= \mathcal{D}_0^p[g(u), \alpha_0](b), \\ \mathcal{D}_L^p[f^+(u), \alpha_L](a) &= \mathcal{D}_L^p[f^+(u), \alpha_L](b), \\ \mathcal{D}_R^p[f^-(u), \alpha_R](a) &= \mathcal{D}_R^p[f^-(u), \alpha_R](b), \end{aligned} \quad (3.4)$$

for $p = 1, 2, 3, \dots, k$. Consequently, we can obtain the coefficients A_0, B_0, A_L , and B_R by (2.5), (2.14) and (2.20).

With the boundary treatments (3.4), we are able to establish the following theorem, which provides error estimates when the infinite series (2.8), (2.16) and (2.17) are truncated by the corresponding k^{th} partial sum.

Theorem 3.1 *Suppose $v(x)$ is a periodic smooth function.*

1. *Consider the operator \mathcal{D}_0 with the boundary treatment $\mathcal{D}_0(a) = \mathcal{D}_0(b)$. If $v(x) \in C^{2k+2}[a, b]$, then we have*

$$\|\partial_{xx} v(x) + \alpha^2 \sum_{p=1}^k \mathcal{D}_0^p[v, \alpha](x)\|_\infty \leq C \left(\frac{1}{\alpha}\right)^{2k} \|\partial_x^{2k+2} v(x)\|_\infty \quad (3.5)$$

where C is a constant only depending on k .

2. *Consider the operator \mathcal{D}_L and \mathcal{D}_R with the boundary treatment $\mathcal{D}_L(a) = \mathcal{D}_L(b)$ and $\mathcal{D}_R(a) = \mathcal{D}_R(b)$, respectively. If $v(x) \in C^{k+1}[a, b]$, then we have*

$$\|\partial_x v(x) - \alpha \sum_{p=1}^k \mathcal{D}_L^p[v, \alpha](x)\|_\infty \leq C \left(\frac{1}{\alpha}\right)^k \|\partial_x^{k+1} v(x)\|_\infty, \quad (3.6)$$

and

$$\|\partial_x v(x) + \alpha \sum_{p=1}^k \mathcal{D}_R^p[v, \alpha](x)\|_\infty \leq C \left(\frac{1}{\alpha}\right)^k \|\partial_x^{k+1} v(x)\|_\infty, \quad (3.7)$$

where C is a constant depending only on k .

To prove this theorem, we first introduce the following lemma regarding operator \mathcal{D}_* , where $*$ can be $0, L$ and R .

Lemma 3.2 *Suppose $v(x)$ is a periodic smooth function.*

1. For the operator \mathcal{D}_0 with the boundary treatment $\mathcal{D}_0(a) = \mathcal{D}_0(b)$, we have

$$\mathcal{D}_0[v, \alpha](x) = - \sum_{p=1}^k \left(\frac{1}{\alpha} \right)^{2p} \partial_x^{2p} v(x) - \left(\frac{1}{\alpha} \right)^{2k+2} \mathcal{L}_0^{-1}[\partial_x^{2k+2} v, \alpha](x). \quad (3.8)$$

if $v(x) \in C^{2k+2}[a, b]$.

2. For the operators \mathcal{D}_L and \mathcal{D}_R with the boundary treatment $\mathcal{D}_L(a) = \mathcal{D}_L(b)$ and $\mathcal{D}_R(a) = \mathcal{D}_R(b)$, respectively, we have

$$\mathcal{D}_L[v, \alpha](x) = - \sum_{p=1}^k \left(-\frac{1}{\alpha} \right)^p \partial_x^p v(x) + \left(-\frac{1}{\alpha} \right)^{k+1} \mathcal{L}_L^{-1}[\partial_x^{k+1} v, \alpha](x), \quad (3.9a)$$

$$\mathcal{D}_R[v, \alpha](x) = - \sum_{p=1}^k \left(\frac{1}{\alpha} \right)^p \partial_x^p v(x) - \left(\frac{1}{\alpha} \right)^{k+1} \mathcal{L}_R^{-1}[\partial_x^{k+1} v, \alpha](x), \quad (3.9b)$$

if $v(x) \in C^{k+1}[a, b]$.

The proof of the lemma heavily relies on integration by parts and is provided in the appendix. Below, we prove Theorem 3.1.

Proof For brevity, we only show the details of the proof for case 1. Following a similar argument, one can easily prove the case 2.

First, by repeating the proof of Lemma 3.2 with $\mathcal{D}_0(a) = \mathcal{D}_0(b)$, we have the following equality for any m with $0 \leq m < k$:

$$\mathcal{D}_0[\partial_x^{2m} v, \alpha](x) = - \sum_{p=m+1}^k \left(\frac{1}{\alpha} \right)^{2(p-m)} \partial_x^{2p} v(x) - \left(\frac{1}{\alpha} \right)^{2(k+1-m)} \mathcal{L}_0^{-1}[\partial_x^{2k+2} v, \alpha](x).$$

Meanwhile, it is easy to verify that $\partial_{xx} \mathcal{L}_0^{-1}[v, \alpha](x) = \alpha^2 \mathcal{L}_0^{-1}[v, \alpha](x) - \alpha^2 v(x) = -\alpha^2 \mathcal{D}_0[v, \alpha](x)$. Therefore, we have $\partial_{xx} \mathcal{D}_0[v, \alpha](x) = \partial_{xx} v(x) + \alpha^2 \mathcal{D}_0[v, \alpha](x)$. Furthermore, for $0 \leq p \leq k$, there is a general form

$$\begin{aligned} \partial_x^{2p} \mathcal{D}_0[v, \alpha](x) &= \sum_{m=1}^p \alpha^{2(p-m)} \partial_x^{2m} v(x) + \alpha^{2p} \mathcal{D}_0[v, \alpha](x) \\ &= - \sum_{m=p+1}^k \left(\frac{1}{\alpha} \right)^{2(m-p)} \partial_x^{2m} v(x) - \left(\frac{1}{\alpha} \right)^{2(k+1-p)} \mathcal{L}_0^{-1}[\partial_x^{2k+2} v, \alpha](x). \end{aligned}$$

Next, let us consider the operator $\mathcal{D}_0^2 = \mathcal{D}_0[\mathcal{D}_0]$. Since the periodic boundary treatment (3.4) is imposed for \mathcal{D}_0 , we deduce that

$$\begin{aligned} \mathcal{D}_0^2[v, \alpha](x) &= - \sum_{p=1}^{k-1} \left(\frac{1}{\alpha} \right)^{2p} \partial_x^{2p} \mathcal{D}_0[v, \alpha](x) - \left(\frac{1}{\alpha} \right)^{2k} \mathcal{L}_0^{-1}[\partial_x^{2k} \mathcal{D}_0[v, \alpha], \alpha](x) \\ &= - \sum_{p=1}^{k-1} \left(\frac{1}{\alpha} \right)^{2p} \left(- \sum_{m=p+1}^k \left(\frac{1}{\alpha} \right)^{2(m-p)} \partial_x^{2m} v(x) - \left(\frac{1}{\alpha} \right)^{2(k+1-p)} \mathcal{L}_0^{-1}[\partial_x^{2k+2} v, \alpha](x) \right) \\ &\quad - \left(\frac{1}{\alpha} \right)^{2k} \mathcal{L}_0^{-1}[-\frac{1}{\alpha^2} \mathcal{L}_0^{-1}[\partial_x^{2k+2} v, \alpha], \alpha](x) \end{aligned}$$

$$\begin{aligned}
&= \sum_{p=2}^k (p-1) \left(\frac{1}{\alpha}\right)^{2p} \partial_x^{2p} v(x) + (k-1) \left(\frac{1}{\alpha}\right)^{2k+2} \mathcal{L}_0^{-1}[\partial_x^{2k+2} v, \alpha](x) \\
&\quad + \left(\frac{1}{\alpha}\right)^{2k+2} \left(\mathcal{L}_0^{-1}\right)^2 [\partial_x^{2k+2} v, \alpha](x).
\end{aligned}$$

Therefore,

$$\begin{aligned}
\mathcal{D}_0[v, \alpha](x) + \mathcal{D}_0^2[v, \alpha](x) &= -\frac{1}{\alpha^2} \partial_{xx} v(x) + \sum_{p=3}^k (p-2) \left(\frac{1}{\alpha}\right)^{2p} \partial_x^{2p} v(x) \\
&\quad + \left(\frac{1}{\alpha}\right)^{2k+2} Q_2(x),
\end{aligned}$$

where $Q_2(x) = (k-2) \mathcal{L}_0^{-1}[\partial_x^{2k+2} v, \alpha](x) + \left(\mathcal{L}_0^{-1}\right)^2 [\partial_x^{2k+2} v, \alpha](x)$. Repeating the process, and finally, we arrive at

$$\sum_{p=1}^k \mathcal{D}_0^p[v, \alpha](x) = -\frac{1}{\alpha^2} \partial_{xx} v(x) + \left(\frac{1}{\alpha}\right)^{2k+2} Q_k(x),$$

where $Q_k(x)$ is a linear combination of functions $\left(\mathcal{L}_0^{-1}\right)^p [\partial_x^{2k+2} v, \alpha](x)$, $p = 1, 2, \dots, k$. Note that, for any $w(x) \in C[a, b]$, we have

$$\|\mathcal{L}_0^{-1}[w, \alpha](x)\|_\infty \leq C_0 \|w\|_\infty,$$

where C_0 is a constant independent of w and α . Then, there is a constant C only depending on k , such that

$$\|\partial_{xx} v(x) + \alpha^2 \sum_{p=1}^k \mathcal{D}_0^p[v, \alpha](x)\|_\infty = \left\| \left(\frac{1}{\alpha}\right)^{2k} Q_k(x) \right\|_\infty \leq C \left(\frac{1}{\alpha}\right)^{2k} \|\partial_x^{2k+2} v(x)\|_\infty,$$

which completes the proof. \square

Remark 3.3 For the numerical schemes formulated below, we will choose

$$\alpha_0 = \sqrt{\frac{\beta}{q \Delta t}}, \quad q = \max_u |g'(u)|, \quad (3.10a)$$

$$\alpha_L = \alpha_R = \frac{\beta}{c \Delta t}, \quad c = \max_u |f'(u)|, \quad (3.10b)$$

in (3.1). Here, Δt denotes the time step and β is a prescribed constant independent of Δt . Define

$$\begin{aligned}
\mathcal{H}[u](x) &= -\frac{\beta}{c \Delta t} \sum_{p=1}^k \mathcal{D}_L^p \left[f^+(u), \frac{\beta}{c \Delta t} \right](x) + \frac{\beta}{c \Delta t} \sum_{p=1}^k \mathcal{D}_R^p \left[f^-(u), \frac{\beta}{c \Delta t} \right](x) \\
&\quad - \frac{\beta}{q \Delta t} \sum_{p=1}^k \mathcal{D}_0^p \left[g(u), \sqrt{\frac{\beta}{q \Delta t}} \right](x),
\end{aligned} \quad (3.11)$$

which approximates $-f(u)_x + g(u)_{xx}$ with accuracy $\mathcal{O}(\Delta t^k)$.

3.2 Homogeneous Boundary Condition

With the boundary condition $\partial_x^p u(a) = 0$ and $\partial_x^p u(b) = 0$, $p \geq 1$, we require

$$\begin{aligned}\mathcal{D}_0^p[g(u), \alpha_0](a) &= 0, \quad \mathcal{D}_0^p[g(u), \alpha_0](b) = 0, \\ \alpha_L \mathcal{D}_L^p[f^+(u), \alpha_L](a) - \alpha_R \mathcal{D}_R^p[f^-(u), \alpha_R](a) &= 0, \\ \alpha_L \mathcal{D}_L^p[f^+(u), \alpha_L](b) - \alpha_R \mathcal{D}_R^p[f^-(u), \alpha_R](b) &= 0,\end{aligned}\quad (3.12)$$

for $p = 1, 2, 3, \dots, k$. The coefficients are obtained from the following formula:

- The operator \mathcal{D}_0 is required to satisfy

$$\mathcal{D}_0[v, \alpha](a) = \mathcal{D}_0[v, \alpha](b) = 0 \quad (3.13)$$

with a given function $v(x)$. Then, we have

$$A_0 = \frac{\mu (I^0[v, \alpha](b) - v(b)) - (I^0[v, \alpha](a) - v(a))}{1 - \mu^2}, \quad (3.14)$$

$$B_0 = \frac{\mu (I^0[v, \alpha](a) - v(a)) - (I^0[v, \alpha](b) - v(b))}{1 - \mu^2}. \quad (3.15)$$

- The operators \mathcal{D}_L and \mathcal{D}_R are required to satisfy

$$\mathcal{D}_L[v_1, \alpha](a) - \mathcal{D}_R[v_2, \alpha](a) = 0, \quad \text{and} \quad \mathcal{D}_L[v_1, \alpha](b) - \mathcal{D}_R[v_2, \alpha](b) = 0 \quad (3.16)$$

with given functions $v_1(x)$ and $v_2(x)$. Then,

$$A_L = \frac{\mu (v_2(b) - v_1(b) + I^L[v_1, \alpha](b)) - (v_2(a) - v_1(a) - I^R[v_2, \alpha](a))}{1 - \mu^2}, \quad (3.17)$$

$$B_R = \frac{(v_2(b) - v_1(b) + I^L[v_1, \alpha](b)) - \mu (v_2(a) - v_1(a) - I^R[v_2, \alpha](a))}{1 - \mu^2}. \quad (3.18)$$

By analogy with Theorem 3.1, we can establish the error estimate for the partial sum (3.1). We omit the proof, since it is quite similar to that of Theorem 3.1.

Theorem 3.4 Suppose $v(x)$ is a function defined on $[a, b]$ with the homogeneous boundary condition that $\partial_x^p v(a) = \partial_x^p v(b) = 0$, for $p \geq 1$.

1. Consider the operator \mathcal{D}_0 with the boundary treatment (3.13). If $v(x) \in C^{2k+2}[a, b]$, then we have

$$\|\partial_{xx} v(x) + \alpha^2 \sum_{p=1}^k \mathcal{D}_0^p[v, \alpha](x)\|_\infty \leq C \left(\frac{1}{\alpha}\right)^{2k} \|\partial_x^{2k+2} v(x)\|_\infty, \quad (3.19)$$

where C is a constant only depending on k .

2. Assume $v(x) = v_1(x) + v_2(x)$ and both $v_1(x)$ and $v_2(x)$ satisfy the homogeneous boundary condition. Consider the operator \mathcal{D}_L and \mathcal{D}_R with the boundary treatment (3.16). If $v_1(x), v_2(x) \in C^{k+1}[a, b]$, then we have

$$\|\partial_x v(x) - \left(\alpha \sum_{p=1}^k \mathcal{D}_L^p[v_1, \alpha](x) - \alpha \sum_{p=1}^k \mathcal{D}_R^p[v_2, \alpha](x)\right)\|_\infty \leq C \left(\frac{1}{\alpha}\right)^k \|\partial_x^{k+1} v(x)\|_\infty, \quad (3.20)$$

where C is a constant depending only on k .

Remark 3.5 For the homogeneous boundary condition case, we can still use (3.11) to approximate $-f(u)_x + g(u)_{xx}$ with accuracy $\mathcal{O}(\Delta t^k)$.

4 Spatial Discretization

In this section, we present the details about the spatial discretization of $\mathcal{H}[u]$ in (3.11). The proposed algorithm is based on our early work on a high order WENO MOL^T schemes for transport problems. Suppose the domain $[a, b]$ is divided by $N + 1$ uniformly distributed grid points

$$a = x_0 < x_1 < \cdots < x_{N-1} < x_N = b,$$

with the mesh size $\Delta x = \frac{b-a}{N}$. Denote u_i^n as the numerical solution at the spatial location x_i and time level t^n . At each grid point x_i , we further denote $I^*[v, \alpha](x_i)$ as I_i^* , where $*$ can be 0, L and R . Note that the convolution integrals I_i^L and I_i^R satisfy a recursive relation

$$I_i^L = I_{i-1}^L e^{-\alpha_L \Delta x} + J_i^L, \quad i = 1, \dots, N, \quad I_0^L = 0, \quad (4.1a)$$

$$I_i^R = I_{i+1}^R e^{-\alpha_R \Delta x} + J_i^R, \quad i = 0, \dots, N-1, \quad I_N^R = 0, \quad (4.1b)$$

respectively, where

$$J_i^L = \alpha_L \int_{x_{i-1}}^{x_i} v(y) e^{-\alpha_L (x_i - y)} dy, \quad J_i^R = \alpha_R \int_{x_i}^{x_{i+1}} v(y) e^{-\alpha_R (y - x_i)} dy. \quad (4.2)$$

Therefore, once we have computed J_i^L and J_i^R for all i , we then can obtain I_i^L and I_i^R via the recursive relation. In addition, the convolution integral $I^0[v, \alpha_0](x)$ can be split into $I^L[v, \alpha_0](x)$ and $I^R[v, \alpha_0](x)$, i.e.,

$$I^0[v, \alpha_0](x) = \frac{1}{2}(I^L[v, \alpha_0](x) + I^R[v, \alpha_0](x)).$$

Thus, I_i^0 is evaluated in the same way as I_i^L and I_i^R , see [15].

A distinct feature of the Eq. (1.1) is that discontinuous solution structures and sharp fronts may develop. The WENO methodology has long been a standard tool to solve hyperbolic problems with discontinuous solutions, which can achieve sharp and non-oscillatory shock transitions and high order accuracy in smooth regions [31,41,43]. Recently, in [36], the authors developed a finite difference WENO scheme to solve degenerate parabolic equations. Such an approach directly approximates the second derivative term using a conservative flux difference formulation. In [20], a WENO-based high order quadrature was developed to evaluate J_i^L and J_i^R . Some related works in the literature about the WENO-based quadrature include [18,19,35]. In this work, we still employ the WENO-based quadrature from [20] with the aim to avoid spurious oscillations when solving discontinuous problems. For the reader's convenience, we will briefly review the main procedure. Note that, as suggested in our numerical results, the WENO methodology itself may not be adequate to suppress solution overshoots. To enhance robustness of the method, we propose to further incorporate a nonlinear filter. Such a filter is constructed via the information from the WENO procedure and hence will not increase the cost significantly. Moreover, we apply the WENO quadrature only for approximating operators with $p = 1$ in (3.11), and use cheap high order linear quadrature for those with $p > 1$. The numerical evidence indicates that by doing so we can reduce the cost and the scheme is still high order accurate and free of oscillations.

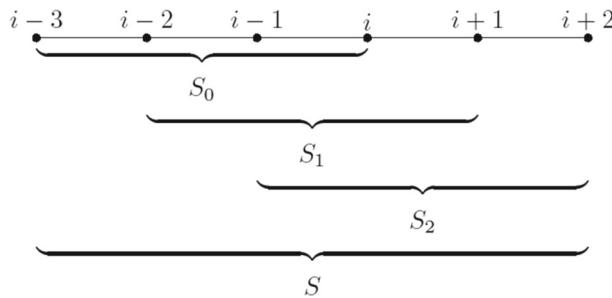


Fig. 1 The structure of the stencils in WENO integration

4.1 WENO-based Quadrature

Below, the fifth order WENO-based quadrature for approximating $J_i^L = J^L[v, \alpha](x_i)$ is provided as an example. The corresponding stencil used is shown in Fig. 1, and all coefficients can be found in [20]. The process to obtain J_i^R is mirror symmetric to that of J_i^L with respect to point x_i .

1. As with the standard WENO methodology, we first choose the three small stencils as $S_r(i) = \{x_{i-3+r}, x_{i-2+r}, x_{i-1+r}, x_{i+r}\}$, $r = 0, 1, 2$. On each small stencil, there is a unique polynomial $p_r(x)$ of degree at most three which interpolates $v(x)$ at the nodes in $S_r(i)$. Then we are able to compute three candidates for J_i^L denoted by $J_{i,r}^L$, $r = 0, 1, 2$

$$J_{i,r}^L = \alpha \int_{x_{i-1}}^{x_i} e^{-\alpha(x_i-y)} p_r(y) dx = \sum_{j=0}^3 c_{-3+r+j}^{(r)} v_{i-3+r+j},$$

where the coefficients $c_{-3+r+j}^{(r)}$ depend on α and the cell size Δx , but not on v .

2. On the entire big stencil $S(i) = \{x_{i-3}, \dots, x_{i+2}\}$, there is a unique polynomial $p(x)$ of degree at most five interpolating $v(x)$ at the nodes in $S(i)$. Then we have

$$J_{i,S}^L = \alpha \int_{x_{i-1}}^{x_i} e^{-\alpha(x_i-y)} p(y) dx = \sum_{j=0}^5 c_{-3+j} v_{i-3+j} = \sum_{r=0}^3 d_r J_{i,r}^L \quad (4.3)$$

that approximates J_i^L with the linear weights d_r .

3. Replace the linear weights d_r with the nonlinear weights ω_r that are defined as

$$\omega_r = \frac{\tilde{\omega}_r}{\sum_{s=0}^2 \tilde{\omega}_s}, \quad \text{with } \tilde{\omega}_r = \frac{d_r}{(\epsilon + SI_r)^2}, \quad r = 0, 1, 2. \quad (4.4)$$

Here, $\epsilon > 0$ is a small number to avoid a zero denominator, and we take $\epsilon = 10^{-6}$ in our numerical tests. The smoothness indicators SI_r that measure the relative smoothness of the function $v(x)$ in the stencil $S_r(i)$ are defined as

$$\begin{aligned} SI_0 = & \frac{781}{720} (-v_{i-3} + 3v_{i-2} - 3v_{i-1} + v_i)^2 + \frac{13}{48} (v_{i-3} - 5v_{i-2} + 7v_{i-1} - 3v_i)^2 \\ & + (v_{i-1} - v_i)^2, \end{aligned}$$

$$SI_1 = \frac{781}{720}(-v_{i-2} + 3v_{i-1} - 3v_i + v_{i+1})^2 + \frac{13}{48}(v_{i-2} - v_{i-1} - v_i + v_{i+1})^2 + (v_{i-1} - v_i)^2,$$

$$SI_2 = \frac{781}{720}(-v_{i-1} + 3v_i - 3v_{i+1} + v_{i+2})^2 + \frac{13}{48}(-3v_{i-1} + 7v_i - 5v_{i+1} - v_{i+2})^2 + (v_{i-1} - v_i)^2.$$

4. Lastly, J_i^L is approximated by $J_{i,W}^L$, where $J_{i,W}^L = \sum_{r=0}^2 \omega_r J_{i,r}^L$.

4.2 Nonlinear Filter

As mentioned above, the k^{th} order accuracy in time is attained via the k^{th} partial sum. However, it is observed from the numerical results that, when $k \geq 2$, spurious oscillations will appear for some non-smooth problems, even though the WENO quadrature is applied to compute the partial sum \mathcal{H} . Therefore, to further enhance robustness and to avoid spurious oscillations, we introduce a nonlinear “filter” denoted by σ when approximating ∂_x . Note that such a filter is only needed for the convection part. Below, we only consider periodic boundary conditions to illustrate the idea and the proposed methodology can be extended straightforwardly to handle the special homogeneous boundary condition.

The key idea of the proposed nonlinear filter is that, unlike (3.1), we use the following modified formulation to approximate ∂_x :

$$\partial_x \approx \frac{\beta}{c\Delta t} \mathcal{D}_L + \frac{\beta}{c\Delta t} \sum_{p=2}^k \sigma_{L,i}^{p-1} \mathcal{D}_L^p, \quad \text{and} \quad \partial_x \approx -\frac{\beta}{c\Delta t} \mathcal{D}_R - \frac{\beta}{c\Delta t} \sum_{p=2}^k \sigma_{R,i}^{p-1} \mathcal{D}_R^p, \quad (4.5)$$

where the filters $\sigma_{L,i}$ and $\sigma_{R,i}$ are incorporated. In this work, the filters are designed to fulfill several requirements: (a) $\sigma_{L,i}$ and $\sigma_{R,i}$ are $1 + \mathcal{O}(\Delta x^k)$ when the solution is continuous, thus maintaining the original high order accuracy; (b) they are close to 0 when the grid point x_i is in the vicinity of a discontinuity, thus decreasing the influence of the high order terms which may lead to oscillations; (c) the associated computational overhead is relatively low. To achieve this goal, the design of the filter is based on the smoothness indicators from the WENO methodology and the underlying idea is similar to that proposed in [9].

Below, we provide the details of construction of the filter. Assume that we have obtained the approximation to the integral J_i^L based on the WENO-based quadrature using the six-point stencil $S(i) = \{x_{i-3}, \dots, x_{i+2}\}$ and the associated three small stencils $S_0(i)$, $S_1(i)$ and $S_2(i)$ given in Sect. 4.1. Following the idea in [9], we introduce a new parameter τ_i , which is simply defined as the absolute difference between the smoothness indicators SI_0 and SI_2 , namely,

$$\tau_i = |SI_0 - SI_2|.$$

Note that τ_i can be obtained with little cost since SI_0 and SI_2 are already available. If $v(x)$ is smooth on the entire stencil $S(i)$, applying the Taylor expansion to SI_0 and SI_2 gives

$$SI_0 = (v'_{i-1/2})^2 \Delta x^2 + \frac{1}{12} (13(v''_{i-1/2})^2 + v'_{i+1/2} v^{(3)}_{i-1/2}) \Delta x^4 + \frac{1}{2880} (3129(v^{(3)}_{i-1/2})^2 - 1820v''_{i-1/2} v^{(4)}_{i-1/2} + 3v'_{i-1/2} v^{(5)}_{i-1/2}) \Delta x^6 + \mathcal{O}(\Delta x^7),$$

$$SI_2 = (v'_{i-1/2})^2 \Delta x^2 + \frac{1}{12} (13(v''_{i-1/2})^2 + v'_{i+1/2} v^{(3)}_{i-1/2}) \Delta x^4 + \frac{1}{2880} (3129(v^{(3)}_{i-1/2})^2 - 1820v''_{i-1/2} v^{(4)}_{i-1/2} + 3v'_{i-1/2} v^{(5)}_{i-1/2}) \Delta x^6 + \mathcal{O}(\Delta x^7).$$

Thus, we deduce that $\tau_i = \mathcal{O}(\Delta x^7)$. We further define

$$\xi_i = \frac{1 + \tau_i^2 / (SI_{max} + \epsilon)^2}{1 + \tau_i^2 / (SI_{min} + \epsilon)^2},$$

where

$$SI_{max} = \max(SI_0, SI_2), \quad \text{and} \quad SI_{min} = \min(SI_0, SI_2).$$

Note that SI_{max} and SI_{min} are both $\mathcal{O}(\Delta x^2)$ in a monotone region, and $\mathcal{O}(\Delta x^4)$ near a critical point. A simple Taylor expansion applying to ξ_i yields

$$\xi_i = 1 + \mathcal{O}(\Delta x^6).$$

Here, we take $\epsilon = 10^{-6}$ to avoid a zero denominator. On the other hand, if the solution $v(x)$ contains a discontinuity within the interval $[x_{i-3}, x_{i-1}]$ or $[x_i, x_{i+2}]$, we can show that $\xi_i = \mathcal{O}(\Delta x^4)$ via a Taylor expansion. Meanwhile, if $v(x)$ is discontinuous within $[x_{i-1}, x_i]$, then ξ_i may be $\mathcal{O}(1)$, while we have $\xi_{i+1} = \mathcal{O}(\Delta x^4)$ which is defined at the neighboring grid point x_{i+1} . The nonlinear filter is defined as

$$\sigma_{L,i} = \min(\xi_i, \xi_{i+1}).$$

$\sigma_{R,i}$ is mirror symmetric to $\sigma_{L,i}$ with respect to x_i , and it reads

$$\sigma_{R,i} = \min(\xi_{i-1}, \xi_i),$$

where ξ_i is obtained through the smoothness indicators that are used for computing $J_{i,W}^R$.

5 Time Discretization and Stability

In this section, we introduce the time discretization methods for evolving (1.1) based on the partial sum approximation (3.11), and then analyze the linear stability property. Denote u^n as the semi-discrete solution at time t^n . In this work, we propose to use the classic explicit SSP RK methods [25] to advance u^n to u^{n+1} . For example, the first order scheme is the forward Euler scheme

$$u^{n+1} = u^n + \Delta t \mathcal{H}[u^n]. \quad (5.1)$$

The second order SSP RK scheme is given as

$$\begin{aligned} u^{(1)} &= u^n + \Delta t \mathcal{H}[u^n], \\ u^{n+1} &= \frac{1}{2} u^n + \frac{1}{2} \left(u^{(1)} + \Delta t \mathcal{H}[u^{(1)}] \right). \end{aligned} \quad (5.2)$$

And the third order SSP RK scheme is given as

$$\begin{aligned} u^{(1)} &= u^n + \Delta t \mathcal{H}[u^n], \\ u^{(2)} &= \frac{3}{4} u^n + \frac{1}{4} \left(u^{(1)} + \Delta t \mathcal{H}[u^{(1)}] \right), \end{aligned}$$

Table 1 β_{\max} in Theorem 5.1, Remark 5.2 and Theorem 5.3 for $k = 1, 2, 3$

k	$\beta_{1,k,\max}$	$\beta_{2,k,\max}$	$\beta_{k,\max}$
1	2	2	1
2	1	1	0.5
3	1.243	0.8375	0.4167

$$u^{n+1} = \frac{1}{3}u^n + \frac{2}{3} \left(u^{(2)} + \Delta t \mathcal{H}[u^{(2)}] \right). \quad (5.3)$$

Note that, to achieve k^{th} order accuracy in time, we should employ the k^{th} order SSP RK method as well as the k^{th} partial sum $\mathcal{H}[u]$. A remarkable advantage of the proposed scheme is that, even though the explicit SSP RK method is used for time integration, unlike the MOL approach [1, 8, 29, 36, 46], the scheme can be A-stable and hence allowing for large time step evolution if β in (3.11) is appropriately chosen. In particular, we establish linear stability of the scheme in the following theorem.

Theorem 5.1 (a) For the linear advection equation $u_t + c u_x = 0$ with periodic boundary conditions, there exists constant $\beta_{1,k,\max} > 0$ for $k = 1, 2$, such that the scheme is A-stable provided $0 < \beta \leq \beta_{1,k,\max}$;

(b) For the linear diffusion equation $u_t = q u_{xx}$ with $q > 0$ and periodic boundary conditions, there exists constant $\beta_{2,k,\max} > 0$ for $k = 1, 2, 3$, such that the scheme is A-stable provided $0 < \beta \leq \beta_{2,k,\max}$.

The constants $\beta_{1,k,\max}$ and $\beta_{2,k,\max}$ are summarized in Table 1.

Proof Here, we only show the proof for $k = 1$ for brevity. Given the ansatz $u^n = \hat{u}^n e^{i\kappa x}$, we can obtain the amplification factor λ via a Von Neumann analysis. The scheme is unconditionally stable if $|\lambda| \leq 1$ for any κ and time step Δt .

(a) We present the proof for the case $c > 0$. For $c < 0$, the proof can be established in the same way. Upon the definitions of \mathcal{D}_L and \mathcal{L}_L , by taking the Fourier transform in space, we obtain that $\widehat{\mathcal{L}}_L = 1 + (i\kappa)/\alpha_L$, and then

$$\widehat{\mathcal{D}}_L = 1 - 1/\widehat{\mathcal{L}}_L = \frac{i\kappa/\alpha_L}{1 + i\kappa/\alpha_L}.$$

For the forward Euler scheme $u^{n+1} = u^n - \Delta t \alpha_L \mathcal{D}_L[cu^n, \alpha_L]$ with the parameter $\alpha_L = \beta/(c\Delta t)$, we could compute the amplification factor λ

$$\lambda = 1 - \beta \frac{i\kappa c \Delta t / \beta}{1 + i\kappa c \Delta t / \beta}.$$

Then, we have $|\lambda| \leq 1$ when $\beta \leq 2$, which implies the scheme is A-stable. Hence, for the first order scheme $k = 1$, we can choose $\beta_{1,1,\max} = 2$.

(b) Similarly, for the forward Euler scheme $u^{n+1} = u^n - \Delta t \alpha_0^2 \mathcal{D}_0[qu^n, \alpha_0]$ and $\alpha_0 = \sqrt{\beta}/q\Delta t$, the amplification factor λ is

$$\lambda = 1 - \beta \widehat{\mathcal{D}}_0 \text{ with } \widehat{\mathcal{D}}_0 = 1 - 1/\widehat{\mathcal{L}}_0 = \frac{(\kappa/\alpha_0)^2}{1 + (\kappa/\alpha_0)^2} \in [0, 1].$$

Then, we still have $\beta \leq 2$ to ensure $|\lambda| \leq 1$. Thus, we let $\beta_{2,1,\max} = 2$.

□

Remark 5.2 Note that, for the linear advection equation $u_t + cu_x = 0$ with periodic boundary conditions, the third order scheme (Eq. (3.11) with $k = 3$ coupled with third order SSP RK) can only be $A(\alpha)$ -stable. But, fortunately, we find that if the scheme is modified as

$$\begin{aligned} \mathcal{H}[u](x) = & -\frac{\beta}{c\Delta t} \sum_{p=1}^3 \mathcal{D}_L^p \left[f^+(u), \frac{\beta}{c\Delta t} \right] (x) + \frac{\beta}{c\Delta t} \sum_{p=1}^3 \mathcal{D}_R^p \left[f^-(u), \frac{\beta}{c\Delta t} \right] (x) \\ & -\frac{\beta}{q\Delta t} \sum_{p=1}^3 \mathcal{D}_0^p \left[g(u), \sqrt{\frac{\beta}{q\Delta t}} \right] (x) \\ & + \frac{\beta}{c\Delta t} \mathcal{D}_0 \left[\mathcal{D}_L^2 \left[f^+(u), \frac{\beta}{c\Delta t} \right] - \mathcal{D}_L^2 \left[f^-(u), \frac{\beta}{c\Delta t} \right], \frac{\beta}{c\Delta t} \right] (x), \end{aligned} \quad (5.4)$$

still with the periodic boundary treatment for the last term, then the scheme coupled with the third order SSP RK integrator is also A -stable provided $0 < \beta \leq \beta_{1,3,\max}$. In light of Lemma 3.2, the extra term in (5.4) is in fact an approximation to f_{xxx} . It will enhance the stability of the scheme and make $\mathcal{H}[u](x)$ fourth order accurate for the case $q \ll c$, i.e., convection dominates. For the homogeneous boundary condition case, we can similarly add the extra term with the treatment (3.13) and make the scheme A -stable. The parameter $\beta_{1,3,\max}$ is given in Table 1.

Below, we provide a Fourier analysis for the fully discrete scheme with the sixth order linear quadrature rule

$$J_j^L = \sum_{r=-3}^2 c_r v_{j+r}, \quad \text{and} \quad J_j^R = \sum_{r=-3}^2 c_r v_{j-r}. \quad (5.5)$$

Without loss of generality, we take $c = 1$ and $q = 1$ for simplicity. Under the assumption that $u_j^n = \hat{u}^n e^{i\kappa x_j}$, we can obtain

$$\hat{D}_L = 1 - \frac{\sum_{r=-3}^2 c_r e^{ir\kappa\Delta x}}{1 - e^{-\alpha\Delta x - i\kappa\Delta x}} \quad (5.6)$$

and

$$\hat{D}_0 = 1 - \frac{1}{2} \left(\frac{\sum_{r=-3}^2 c_r e^{ir\kappa\Delta x}}{1 - e^{-\alpha\Delta x - i\kappa\Delta x}} + \frac{\sum_{r=-3}^2 c_r e^{-ir\kappa\Delta x}}{1 - e^{-\alpha\Delta x + i\kappa\Delta x}} \right) \quad (5.7)$$

Moreover, it is straightforward to check that the amplification factor λ for the linear advection equation $u_t + u_x = 0$ depends on $\beta, \kappa\Delta x$ and $\Delta t/\Delta x$, while for the linear diffusion equation $u_t = u_{xx}$, λ depends on $\beta, \kappa\Delta x$ and $\Delta t/\Delta x^2$. Even though it is very tedious and difficult to derive analytically the condition of $\lambda \leq 1$, as a common practice, we can still numerically verify that, if $0 < \beta < \beta_{\cdot,k,\max}$, for $k = 1, 2, 3$, then $|\lambda| \leq 1$ for any $\kappa\Delta x \in [0, 2\pi]$, Δt and Δx . In Fig. 2, we plot the contours of $|\lambda|$ with $\beta = \beta_{\cdot,k,\max}$ to justify this property. In other words, the scheme is unconditionally stable if β is appropriately chosen according to Table 1.

Combining the both cases in Theorem 5.1, Remark 5.2 and above analysis, we can easily establish a similar unconditional stability property of the scheme for solving linear convection-diffusion problems.

Theorem 5.3 Consider the linear convection-diffusion problem

$$u_t + c u_x = q u_{xx}, \quad (5.8)$$

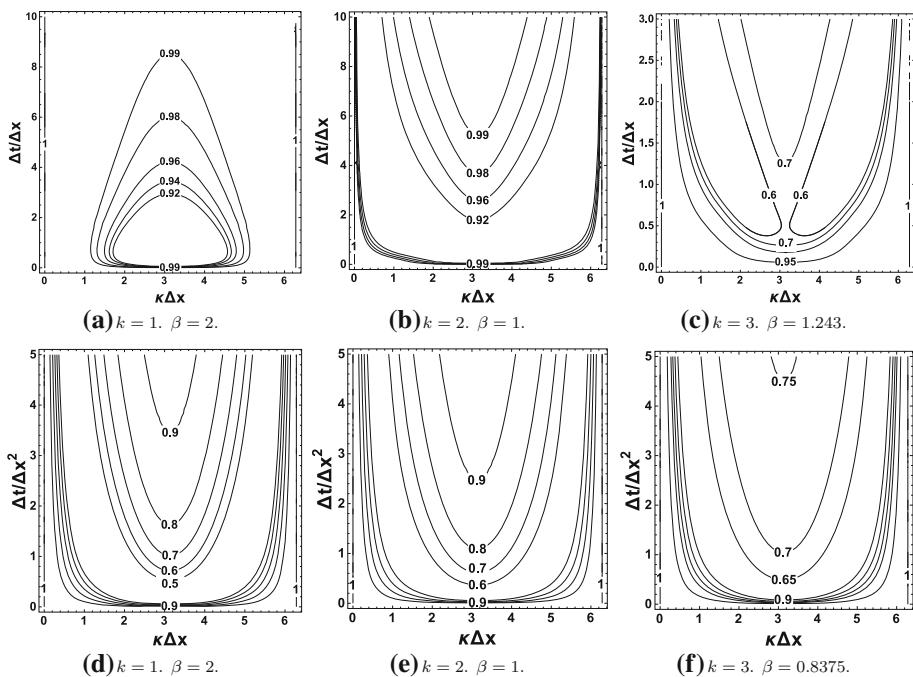


Fig. 2 Contours of $|\lambda|$. Upper: linear advection equation $u_t + u_x = 0$; below: linear diffusion equation $u_t = u_{xx}$

with periodic boundary conditions. c and q are both constants and $q \geq 0$. Suppose the scheme employs the k^{th} order SSP RK method, the k^{th} partial sum in (3.11) or (5.4) for $k = 1, 2, 3$, and the linear quadrature rule (5.5). Then, the scheme is unconditionally stable if $0 < \beta \leq \beta_{k,\max}$, where $\beta_{k,\max} := \frac{1}{2} \min(\beta_{1,k,\max}, \beta_{2,k,\max})$, and the constants $\beta_{1,k,\max}$ and $\beta_{2,k,\max}$ are summarized in Table 1.

Remark 5.4 The unconditional stability is only established for the scheme for solving (5.8) with periodic boundary conditions. Such an analysis cannot be extended to the case of a Dirichlet boundary condition, as the ansatz used does not hold. The G-K-S theory [27] may be applicable, and we leave the investigation to the future study.

Remark 5.5 The WENO methodology together with the nonlinear filter are used to suppress unphysical oscillations. As with the standard finite volume/difference approach, it is numerically confirmed that the use of WENO and the filter is able to enhance the robustness and nonlinear stability, while preserving the linear stability of the original scheme. In other words, the scheme with WENO and the filter is free of unphysical oscillations and unconditional stable at the same time. We cannot provide a theoretical analysis now in this regard now.

To close this section, we summarize the proposed method for approximating one-dimensional problem (1.1) in the following algorithm flowchart.

Given the function u^n , the approximation order $k \leq 3$, the mesh size Δx and the time step Δt .

1. Choose β from the Table 1. Compute q and c at time t^n .
2. On each inner stage of the k^{th} order SSP RK scheme:
 - (a) Compute q and c , and then obtain parameters $\alpha_{0,L,R}$ by (3.10).
 - (b) Convection part:
 - i. Split $f(u)$ into $f^\pm(u)$.
 - ii. Apply \mathcal{D}_L and \mathcal{D}_R on f^+ and f^- , respectively. Use the WENO quadrature to calculate $J^{L,R}$ and at the same time obtain the nonlinear filter $\sigma_{L,R}$. Compute $I^{L,R}$ via (4.1) and then calculate the parameter A_L and B_R based on the boundary condition. Combine $I^{L,R}$, A_L and B_R to construct $\mathcal{D}_L[f^+]$ and $\mathcal{D}_R[f^-]$.
 - iii. For $k > 1$, further construct $\mathcal{D}_L^p[f^+] = \mathcal{D}_L^{p-1}[\mathcal{D}_L[f^+]]$ and $\mathcal{D}_R^p[f^-] = \mathcal{D}_R^{p-1}[\mathcal{D}_R[f^-]]$ by a similar procedure for $1 < p \leq k$. WENO quadrature is not needed for construction of these high order terms.
 - (c) Diffusion part: follow a similar procedure of (b) to construct the partial sum approximation to $g(u)_{xx}$ using operator \mathcal{D}_0 .
 - (d) Substitute the partial sum approximations from (b) and (c) into (3.11) and update the solution accordingly.

6 Two-dimensional Implementation

Consider the following two-dimensional problem

$$u_t + f_1(u)_x + f_2(u)_y = g_1(u)_{xx} + g_2(u)_{yy}. \quad (6.1)$$

The proposed one-dimensional formulation can be directly extended to solving (6.1) based on a dimension-by-dimension approach, namely, approximating ∂_x and ∂_{xx} for fixed y_j and approximating ∂_y and ∂_{yy} for fixed x_i . More specifically, for periodic boundary conditions

$$\begin{aligned} g_1(u)_{xx}|_{(x_i, y_j)} &\approx -\alpha_{0,x}^2 \sum_{p=1}^k \mathcal{D}_0^p[g_1(\cdot, y_j), \alpha_{0,x}](x_i), \\ g_2(u)_{yy}|_{(x_i, y_j)} &\approx -\alpha_{0,y}^2 \sum_{p=1}^k \mathcal{D}_0^p[g_2(x_i, \cdot), \alpha_{0,y}](y_j), \end{aligned}$$

where $\alpha_{0,x} = \sqrt{q_x/(\beta \Delta t)}$ and $\alpha_{0,y} = \sqrt{q_y/(\beta \Delta t)}$ with $q_x = \max_u |g_1'(u)|$, $q_y = \max_u |g_2'(u)|$. To approximate $f_1(u)_x$ and $f_2(u)_y$, the flux splitting strategy is still needed:

$$f_1^\pm(u) = \frac{1}{2}(f_1(u) \pm c_x u), \quad f_2^\pm(u) = \frac{1}{2}(f_2(u) \pm c_y u),$$

where $c_x = \max_u |f'_1(u)|$ and $c_y = \max_u |f'_2(u)|$. Then, the dimension-by-dimension approach can be similarly applied. Again, for periodic boundary conditions,

$$\begin{aligned} -f_1(u)_x|_{(x_i, y_j)} &\approx -\alpha_{L,x} \sum_{p=1}^k \mathcal{D}_L^p [f_1^+(\cdot, y_j), \alpha_{L,x}](x_i) + \alpha_{R,x} \sum_{p=1}^k \mathcal{D}_R^p [f_1^-(\cdot, y_j), \alpha_{R,x}](x_i), \\ -f_2(u)_y|_{(x_i, y_j)} &\approx -\alpha_{L,y} \sum_{p=1}^k \mathcal{D}_L^p [f_2^+(x_i, \cdot), \alpha_{L,y}](y_j) + \alpha_{R,y} \sum_{p=1}^k \mathcal{D}_R^p [f_2^-(x_i, \cdot), \alpha_{R,y}](y_j), \end{aligned}$$

or with a modified term for $k = 3$. In addition, in the x -direction, we choose $\alpha_{L,x} = \alpha_{R,x} = \beta/(c_x \Delta t)$, and in the y -direction, we choose $\alpha_{L,y} = \alpha_{R,y} = \beta/(c_y \Delta t)$.

In the two-dimensional case, β_{\max} needs to be chosen as half of that for one-dimensional problems to attain the unconditional stability for the scheme.

Remark 6.1 We remark that the method is based on a dimension-by-dimension approach for solving high-dimensional problems, and hence suitable for parallel implementation specially in GPU architecture or distributed shared-memory machines. Furthermore, the scheme can be implemented in complex geometry with non-uniform meshes. In [21], we provided a technique together with numerical examples to handle complex geometry in the context of the Hamilton-Jacobi equations. The methodology proposed in [21] can be directly adapted to nonlinear degenerate advection-diffusion equations considered in this paper.

7 Numerical Results

In this section, we present the numerical results to demonstrate efficiency and efficacy of the proposed scheme. The code implements our algorithm in Python is available on the Web. See [22] for details of this implementation and some sample results.

For one-dimensional problems, we choose the time step as

$$\Delta t = \text{CFL} \frac{\Delta x}{q + c}, \quad (7.1)$$

while for two-dimensional problems, the time step is set as

$$\Delta t = \frac{\text{CFL}}{\max((q_x + c_x)/\Delta x, (q_y + c_y)/\Delta y)}.$$

Note that time step Δt is chosen in a form similar to a standard MOL type method. It will enable us to conveniently test accuracy and compare the scheme with other methods. We remark that the CFL number can be chosen arbitrarily large due to the unconditional stability. Meanwhile, the larger the CFL number, the larger the numerical error. If the problem does not have an analytical solution, we will use the numerical solutions by the following first order numerical scheme

$$u_i^n = u_i^n - \frac{\Delta t}{\Delta x} (f_i^+ - f_{i-1}^+) - \frac{\Delta t}{\Delta x} (f_{i+1}^- - f_i^-) + \frac{\Delta t}{\Delta x^2} (g_{i+1} - 2g_i + g_{i-1})$$

with $N = 3000$ grid points and $\Delta t = 0.1 \Delta x^2 / (c \Delta x + 2q)$ as a reference solution.

Example 1 We test the accuracy of the scheme for the one-dimensional linear advection-diffusion problem

$$\begin{cases} u_t + c u_x = q u_{xx}, & -\pi \leq x \leq \pi, \\ u(x, 0) = \sin(x), \end{cases} \quad (7.2)$$

with the 2π -periodic boundary condition. Here, c and $q \geq 0$ are given constants. This problem has the exact solution $u^e(x, t) = e^{-qt} \sin(x - ct)$.

In Tables 2 and 3, we summarize the convergence study for the case of $c = 1, q = 0.01$ and $c = 1, q = 1$ at final time $T = 2$, and the L_∞ errors and the associated orders of accuracy are provided. It is observed that the use of the k^{th} partial sum yields k^{th} order accuracy, thereby verifying the analysis presented in Sect. 3. Moreover, the scheme allows for large CFL numbers due to its unconditionally stability.

To demonstrate the efficiency, we compare the proposed method with the famous Kurganov-Tadmor (KT) method [34] for solving the linear advection-diffusion problem. Note that KT method is second order accurate and explicit, hence subject to CFL time step restriction

$$\Delta t = \frac{\text{CFL}}{q/\Delta x^2 + c/\Delta x},$$

where CFL is chosen to be 0.125 as suggested in [34]. For a fair comparison, we choose $k = 2$ (second order accurate) and CFL = 0.125 for the proposed scheme as well, but pointing out that the scheme is up to third order accurate and unconditionally stable. We summarize the comparison result in Fig. 3, where we plot the CPU time versus L_∞ error for both schemes. It is observed that, for the convection-dominated case ($c = 1, q = 0.01$), the KT method performs slightly better than the proposed scheme, but the slopes are comparable; when considering the diffusion-dominated case, the proposed scheme becomes more efficient, since the KT method suffers the stringent CFL condition ($\Delta t = \mathcal{O}(\Delta x^2)$). We remark that the choice of the step Δt in the method is only an accuracy concern instead of stability. One has the flexibility to choose the “optimal” Δt to balance the CPU cost and the accuracy.

Example 2 We test the porous medium equation (PME) [4,37]

$$u_t = (u^m)_{xx}, \quad (7.3)$$

for some $m > 1$. This equation describes a gas flowing isentropically in a porous medium, where the quantity u represents density of the gas considered. However, for the PME, the classical solutions may not exist in general, even if the initial solution is smooth. Therefore, weak solutions must be considered, and their existence and uniqueness are studied in [2,23, 38].

One famous weak solution of PME is the Barenblatt solution [6,45], which is defined as

$$B_m(x, t) = t^{-p} \left[\left(1 - \frac{p(m-1)}{2m} \frac{|x|^2}{t^{2p}} \right)_+ \right]^{1/(m-1)}, \quad m > 1,$$

where $u_+ = \max(u, 0)$ and $p = (m+1)^{-1}$. For any time $t > 0$, the solution has a compact support $[-a_m(t), a_m(t)]$ with

$$a_m(t) = t^p \sqrt{\frac{2m}{p(m-1)}}.$$

Table 2 Example 1: L_∞ errors and orders of accuracy at $T = 2$, $c = 1$ and $q = 0.01$

CFL	N_x	$k = 1, \beta = 1$		$k = 2, \beta = 0.5$		$k = 3, \beta = 0.4$	
		Error	Order	Error	Order	Error	Order
0.5	40	7.260E-02	–	4.729E-02	–	2.559E-03	–
	80	3.715E-02	0.967	1.218E-02	1.957	1.712E-04	3.902
	160	1.885E-02	0.979	3.077E-03	1.985	1.091E-05	3.972
	320	9.473E-03	0.992	7.703E-04	1.998	6.865E-07	3.990
	640	4.750E-03	0.996	1.928E-04	1.999	4.357E-08	3.978
1	40	1.388E-01	–	1.697E-01	–	3.263E-02	–
	80	7.260E-02	0.935	4.729E-02	1.843	2.559E-03	3.672
	160	3.717E-02	0.966	1.218E-02	1.956	1.712E-04	3.902
	320	1.885E-02	0.980	3.077E-03	1.986	1.091E-05	3.973
	640	9.473E-03	0.992	7.703E-04	1.998	6.864E-07	3.990
2	40	2.474E-01	–	4.375E-01	–	2.313E-01	–
	80	1.388E-01	0.834	1.697E-01	1.366	3.271E-02	2.822
	160	7.260E-02	0.935	4.733E-02	1.842	2.561E-03	3.675
	320	3.717E-02	0.966	1.218E-02	1.958	1.713E-04	3.902
	640	1.885E-02	0.980	3.077E-03	1.986	1.091E-05	3.973

Table 3 Example 1: L_∞ errors and orders of accuracy at $T = 2$, $c = 1$ and $q = 1$

CFL	N_x	$k = 1, \beta = 1$		$k = 2, \beta = 0.5$		$k = 3, \beta = 0.4$	
		Error	Order	Error	Order	Error	Order
0.5	40	1.047E-02	–	1.821E-03	–	1.912E-04	–
	80	5.272E-03	0.990	4.953E-04	1.879	2.787E-05	2.779
	160	2.646E-03	0.995	1.293E-04	1.937	3.751E-06	2.893
	320	1.326E-03	0.997	3.307E-05	1.968	4.870E-07	2.946
	640	6.637E-04	0.998	8.361E-06	1.984	6.206E-08	2.972
1	40	2.043E-02	–	6.088E-03	–	1.117E-03	–
	80	1.047E-02	0.964	1.822E-03	1.741	1.924E-04	2.537
	160	5.272E-03	0.990	4.955E-04	1.878	2.788E-05	2.787
	320	2.646E-03	0.995	1.293E-04	1.938	3.752E-06	2.893
	640	1.326E-03	0.997	3.307E-05	1.968	4.869E-07	2.946
2	40	3.941E-02	–	1.747E-02	–	4.522E-03	–
	80	2.045E-02	0.946	6.098E-03	1.518	1.118E-03	2.016
	160	1.047E-02	0.966	1.822E-03	1.743	1.924E-04	2.539
	320	5.273E-03	0.990	4.955E-04	1.878	2.788E-05	2.787
	640	2.646E-03	0.995	1.293E-04	1.938	3.752E-06	2.894

Here, we choose $t = 1$ as the initial time and the computation domain $[-6, 6]$ with a zero boundary condition $u(\pm 6, t) = 0$. We plot the numerical solutions and exact solutions at $T = 2$ with 200 grid points, respectively, with $m = 2, 5$ and 8 (Fig. 4). Here, we only plot the results for the third order scheme, i.e. $k = 3$, and β is taken as 0.8. It is observed that our scheme is able to approximate the Barenblatt solution accurately without noticeable

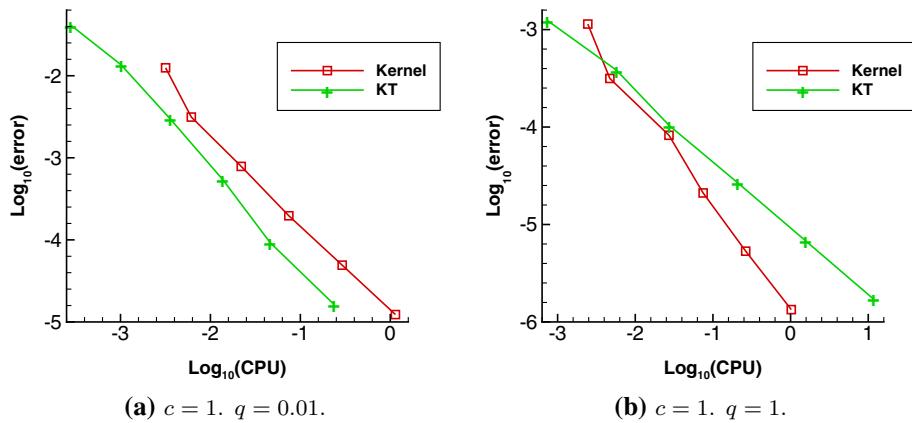


Fig. 3 Example 1: Comparison of CPU time against L_∞ error between the KT scheme and our scheme with $k = 2$. $\beta = 0.5$. CFL = 0.125

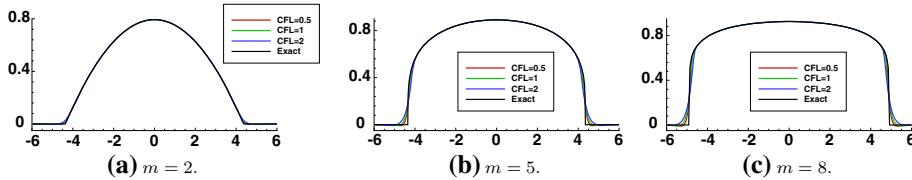


Fig. 4 Example 2: Barenblatt solution for PME. $N = 200$ grid points. $k = 3$. $\beta = 0.8$

oscillations even with a large CFL number. On the other hand, the method with a smaller CFL generates sharper interface transition around $|x| = a_m$.

Example 3 Now, we consider the interaction of two boxes for PME (7.3). Such a model describes how temperature changes when two hot spots are suddenly placed in the domain. Here, we choose the initial condition as

$$u(x, 0) = \begin{cases} 1, & x \in (-4, -1), \\ 2, & x \in (0, 3), \\ 0, & \text{otherwise,} \end{cases} \quad (7.4)$$

in which the two boxes have different heights. We let $m = 6$ in (7.3). In Fig. 5, we show the numerical solution at several instances of time for $k = 3$. The computational domain is chosen as $[-6, 6]$ and a zero boundary condition $u(\pm 6, t) = 0$ is imposed. Here, we use $N = 400$ grid points. Note that the exact solution is unknown and we benchmark the scheme against the reference. It is observed that the scheme is able to capture the sharp interface even though a large CFL number is used and numerical solutions agree with the reference solution very well. We also would like to remark that the scheme can choose large CFLs due to its unconditionally stable property. Meanwhile, the approximation quality deteriorates due to the corresponding large temporal error. The CFL is chosen as a trade-off between CPU efficiency and accuracy.

Example 4 Next, let us consider the Buckley-Leverett equation [10]

$$u_t + f(u)_x = \epsilon(v(u)u_x)_x. \quad (7.5)$$

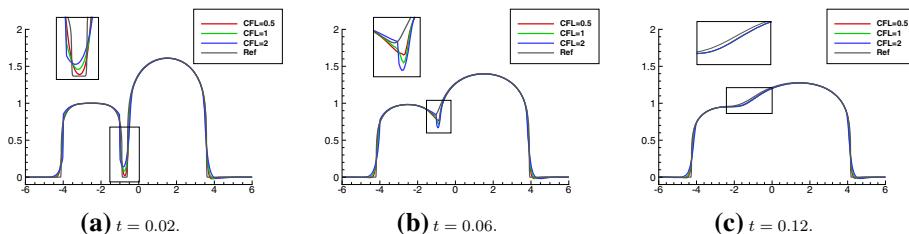


Fig. 5 Example 3: Interaction of the two-Box solution with different heights. $N = 400$ grid points. $k = 3$. $\beta = 0.8$. The small figures are the enlarged view inside the rectangles

In fluid dynamics, this equation is used to model two-phase flow in porous media, such as displacing oil by water in a one-dimensional or quasi-one-dimensional reservoir. We choose

$$v(u) = \begin{cases} 4u(1-u), & 0 \leq u \leq 1, \\ 0, & \text{otherwise,} \end{cases} \quad (7.6)$$

and consider the flux without gravitational effects

$$f(u) = \frac{u^2}{u^2 + (1-u)^2}, \quad (7.7)$$

as well as with gravitational effects

$$f(u) = \frac{u^2}{u^2 + (1-u)^2} (1 - 5(1-u)^2). \quad (7.8)$$

In the simulation, we let $\epsilon = 0.01$. Here, the initial condition is

$$u(x, 0) = \begin{cases} 0, & 0 \leq x < 1 - \frac{1}{\sqrt{2}}, \\ 1, & 1 - \frac{1}{\sqrt{2}} \leq x \leq 1. \end{cases}$$

Up to the final time $T = 0.2$, the solutions keep constant on both endpoints, meaning $u(0, t) = 0$ and $u(1, t) = 1$.

Numerical solutions for $k = 3$ are presented in Fig. 6. It is observed that the scheme with a small CFL, e.g. 0.5 outperforms the one with a large CFL, e.g. 2. In fact, even though the scheme is unconditionally stable, the performance may not be satisfactory when a exceedingly large CFL number is used, which will introduce too much numerical diffusion and thus smear the interface. Nevertheless, when a small CFL number is used, the scheme is able to solve both Riemann problems accurately without generating noticeable spurious oscillations, and the solution is benchmarked against the results reported in [34,36].

Example 5 In this example, we consider a strongly degenerate parabolic convection-diffusion equation

$$u_t + f(u)_x = \epsilon(v(u)u_x)_x. \quad (7.9)$$

We take $\epsilon = 0.1$, $f(u) = u^2$, and

$$v(u) = \begin{cases} 0, & |u| \leq 0.25, \\ 1, & |u| > 0.25. \end{cases} \quad (7.10)$$

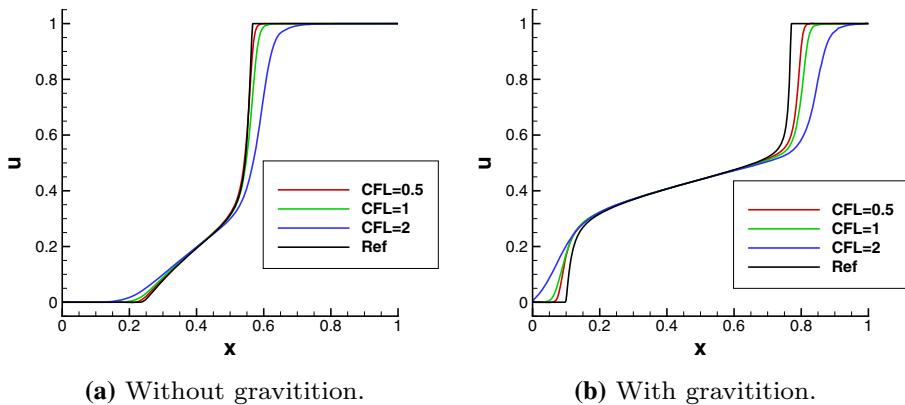


Fig. 6 Example 4: Buckley-Leverett equation. $T = 0.2$. $N = 200$ grid points. $k = 3$. $\beta = 0.4$

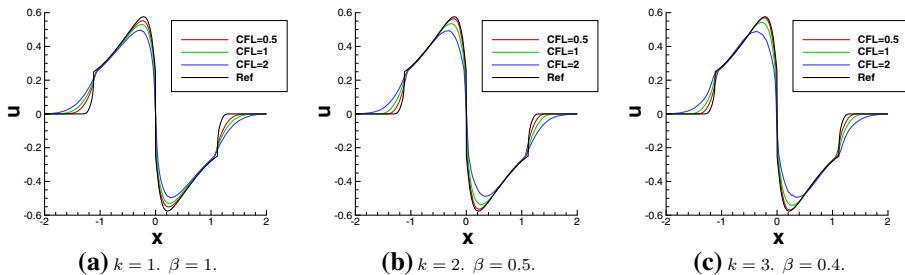


Fig. 7 Example 5: 1D strong degenerate parabolic equation. $T = 0.7$. $N = 200$ grid points

The choice of ν will lead to an interesting fact that the equation is hyperbolic when $u \in [-0.25, 0.25]$ and parabolic elsewhere. We solve the problem with the initial function

$$u(x, 0) = \begin{cases} 1, & -\frac{1}{\sqrt{2}} - 0.4 < x < -\frac{1}{\sqrt{2}} + 0.4, \\ -1, & \frac{1}{\sqrt{2}} - 0.4 < x < \frac{1}{\sqrt{2}} + 0.4, \\ 0, & \text{otherwise,} \end{cases} \quad (7.11)$$

and a zero boundary condition $u(\pm 2, t) = 0$ is imposed before final time $T = 0.7$. Numerical results are presented in Fig. 7. In particular, we compare the performance of the schemes with different orders of accuracy, i.e. $k = 1, 2, 3$. It is observed that the high order scheme performs better in capturing the sharp interface as well as the kinks where the equation changes its type. As expected, the performance deteriorates when a large CFL number is used.

Example 6 We test the accuracy of the proposed scheme for the two-dimensional linear advection-diffusion problem

$$u_t + c u_x + c u_y = q u_{xx} + q u_{yy}, \quad (7.12)$$

on $[-2, 2] \times [-2, 2]$ with the initial function

$$u(x, y, 0) = \sin\left(\frac{x+y}{2}\pi\right).$$

Table 4 Example 6: L_∞ errors and orders of accuracy at $T = 1$, $c = 1$ and $q = 0.01$

CFL	$N_x \times N_y$	$k = 1, \beta = 0.5$		$k = 2, \beta = 0.25$		$k = 3, \beta = 0.2$	
		Error	Order	Error	Order	Error	Order
0.5	20×20	2.03e-01	–	2.53e-01	–	1.45e-01	–
	40×40	1.09e-01	0.904	7.23e-02	1.806	2.07e-02	2.806
	60×60	5.62e-02	0.948	1.86e-02	1.955	2.69e-03	2.944
	160×160	2.86e-02	0.977	4.69e-03	1.991	3.40e-04	2.983
	320×320	1.44e-02	0.987	1.18e-03	1.996	4.27e-05	2.993
1	20×20	3.54e-01	–	6.03e-01	–	5.81e-01	–
	40×40	2.03e-01	0.805	2.53e-01	1.255	1.45e-01	2.003
	60×60	1.09e-01	0.902	7.23e-02	1.806	2.07e-02	2.806
	160×160	5.62e-02	0.948	1.86e-02	1.955	2.69e-03	2.944
	320×320	2.86e-02	0.976	4.69e-03	1.991	3.40e-04	2.984
2	20×20	5.61e-01	–	8.50e-01	–	8.03e-01	–
	40×40	3.57e-01	0.652	6.03e-01	0.496	5.84e-01	0.459
	60×60	2.03e-01	0.814	2.53e-01	1.252	1.45e-01	2.006
	160×160	1.09e-01	0.905	7.23e-02	1.809	2.07e-02	2.810
	320×320	5.62e-02	0.948	1.86e-02	1.955	2.69e-03	2.945

Here, we test with constants $c = 1$ and $q = 0.01$ and the periodic boundary condition. The L_∞ errors and the associated orders of accuracy are provided in Table 4. The k^{th} order accuracy is observed with the use of the k^{th} partial sum, $k = 1, 2$ and 3 allowing for several CFL numbers. We note that β_{\max} needs to be chosen as half of that for one-dimensional problems to attain the unconditional stability for the scheme.

Example 7 We consider the two-dimensional strongly degenerate parabolic convection-diffusion equation

$$u_t + f(u)_x + f(u)_y = \epsilon(v(u)u_x)_x + \epsilon(v(u)u_y)_y, \quad (7.13)$$

in which, ϵ , $f(u)$, $v(u)$ are the same as the one-dimensional case. The initial function is given as

$$u(x, y, 0) = \begin{cases} 1, & (x + 0.5)^2 + (y + 0.5)^2 < 0.16, \\ -1, & (x - 0.5)^2 + (y - 0.5)^2 < 0.16, \\ 0, & \text{otherwise.} \end{cases} \quad (7.14)$$

The solutions at $T = 0.5$ computed by the third order scheme with $\beta = 0.2$, $CFL = 0.5$ and 200×200 grid points are shown in Fig. 8, which agree well with results provided in [36].

Example 8 As the last example, we solve the two-dimensional Buckley-Leverett Equation

$$u_t + f_1(u)_x + f_2(u)_y = \epsilon(u_{xx} + u_{yy}), \quad (7.15)$$

where $\epsilon = 0.01$ and the flux functions are given as

$$f_1(u) = \frac{u^2}{u^2 + (1-u)^2}, \quad f_2(u) = (1 - 5(1-u)^2)f_1(u).$$

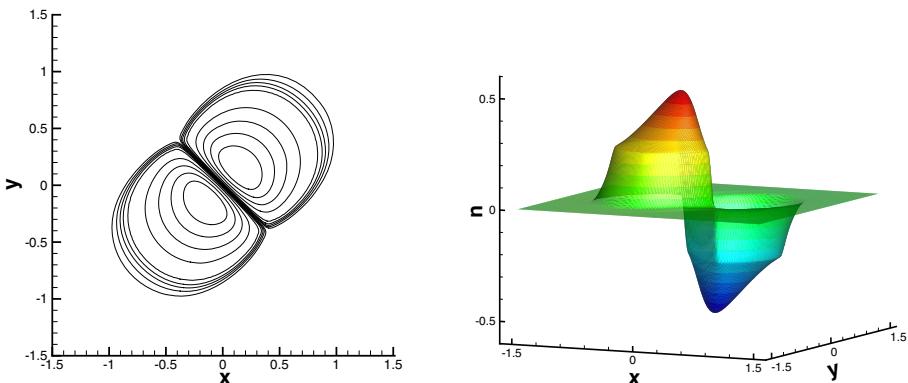


Fig. 8 Example 7: 2D strong degenerate parabolic equation. $k = 3$. $T = 0.5$. $CFL = 0.5$. 200×200 grid points

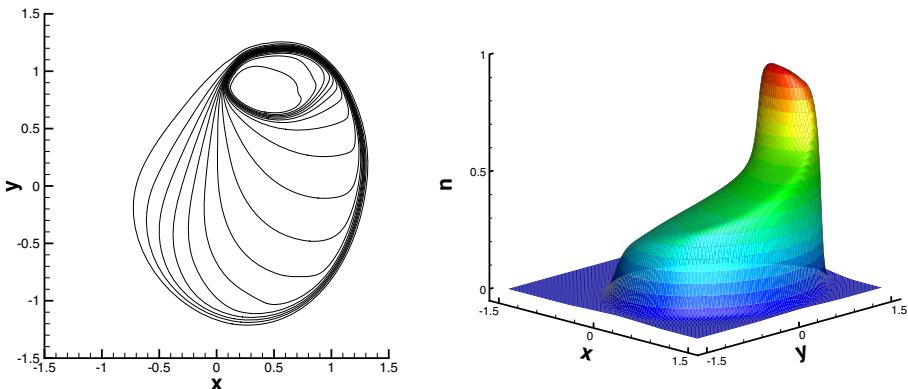


Fig. 9 Example 8: 2D Buckley-Leverett Equation. $k = 3$. $T = 0.5$. $CFL = 0.5$. 200×200 grid points

We compute the problem on $[-1.5, 1.5] \times [-1.5, 1.5]$, with the initial condition

$$u(x, y, 0) = \begin{cases} 1, & x^2 + y^2 < 0.5, \\ 0, & \text{otherwise.} \end{cases}$$

Here, we only show the results computed by the third order scheme with $\beta = 0.2$ and 200×200 grid points in Fig. 9. The result agrees with that reported in [34], demonstrating the effectiveness of the scheme for solving this challenging two-dimensional problem.

8 Conclusion

In this paper, we proposed a novel numerical scheme to solve the nonlinear degenerate parabolic equations with non-smooth solutions. In such a framework, the spatial derivatives were represented as a special kernel based formulation of the solutions found in the method of lines transpose framework, and a fast summation algorithm was used to reduce the computational complexity of the kernel based approach to $O(N)$. The kernel based formulation

used in this work is as fast as explicit time stepping methods. In time, we coupled the scheme with the high order explicit SSP RK method. Theoretical investigations indicated that the proposed scheme is unconditionally stable up to third order accuracy. Therefore, the new method allowed for much larger time step evolution compared with other explicit schemes with the same order accuracy. Moreover, to avoid spurious oscillations, a high order WENO methodology and a nonlinear filter are further employed. A collection of numerical tests verified the performance of the proposed scheme, demonstrating both its designed high order accuracy and the ability to produce non-oscillatory shock transitions for discontinuous solutions. Future work consists of extending the scheme to other equations and deal with general boundary conditions.

Appendix A. Proof of Lemma 3.2

Here, we only give the proof for the case of \mathcal{D}_0 and \mathcal{D}_L . For \mathcal{D}_R , the proof can be established by a similar idea.

Using the definition of I^0 and integration by parts twice, we have

$$\begin{aligned} I^0[v, \alpha](x) &= v(x) + \frac{1}{\alpha^2} I^0[v_{xx}, \alpha](x) - \left(\frac{1}{2}v(a) - \frac{1}{2\alpha}v_x(a) \right) e^{-\alpha(x-a)} \\ &\quad - \left(\frac{1}{2}v(b) + \frac{1}{2\alpha}v_x(b) \right) e^{-\alpha(b-x)}. \end{aligned} \quad (\text{A.1})$$

Thus,

$$\begin{aligned} \mathcal{D}_0[v, \alpha](x) &= -\frac{1}{\alpha^2} I^0[v_{xx}, \alpha](x) - \left(A_0[v, \alpha] - \frac{1}{2}v(a) + \frac{1}{2\alpha}v_x(a) \right) e^{-\alpha(x-a)} \\ &\quad - \left(B_0[v, \alpha] - \frac{1}{2}v(b) - \frac{1}{2\alpha}v_x(b) \right) e^{-\alpha(b-x)}. \end{aligned}$$

Here, $A_0[v, \alpha]$ and $B_0[v, \alpha]$ are obtained from the boundary treatment of $\mathcal{D}_0[v, \alpha]$ (2.5). Moreover, based on (A.1), $A_0[v, \alpha]$ and $B_0[v, \alpha]$ can be rewritten as

$$\begin{aligned} A_0[v, \alpha] &= \frac{1}{1-\mu} \left(\frac{1}{\alpha^2} I^0[v_{xx}, \alpha](b) - \frac{1}{2} \left(v(a) - \frac{1}{\alpha}v_x(a) \right) \mu + \frac{1}{2} \left(v(b) - \frac{1}{\alpha}v_x(b) \right) \right), \\ B_0[v, \alpha] &= \frac{1}{1-\mu} \left(\frac{1}{\alpha^2} I^0[v_{xx}, \alpha](a) + \frac{1}{2} \left(v(a) + \frac{1}{\alpha}v_x(a) \right) - \frac{1}{2} \left(v(b) + \frac{1}{\alpha}v_x(b) \right) \mu \right). \end{aligned}$$

Therefore, we have

$$\begin{aligned} \mathcal{D}_0[v, \alpha](x) &= -\frac{1}{\alpha^2} I^0[v_{xx}, \alpha](x) - \frac{1}{\alpha^2} \frac{I^0[v_{xx}, \alpha](b)}{1-\mu} e^{-\alpha(x-a)} - \frac{1}{\alpha^2} \frac{I^0[v_{xx}, \alpha](a)}{1-\mu} e^{\alpha(b-x)} \\ &= -\frac{1}{\alpha^2} \mathcal{L}_0^{-1}[v_{xx}, \alpha](x) = -\frac{1}{\alpha^2} v_{xx}(x) + \frac{1}{\alpha^2} \mathcal{D}_0[v_{xx}, \alpha](x) \end{aligned}$$

Upon iterating this process k times, we obtain that

$$\mathcal{D}_0[v, \alpha](x) = -\sum_{p=1}^k \frac{1}{\alpha^{2p}} \partial_x^{2p} v(x) - \frac{1}{\alpha^{2k+2}} \mathcal{L}_0^{-1}[\partial_x^{2k+2} v, \alpha](x).$$

Similarly, we can do integration by parts on I^L , and have

$$I^L[v, \alpha] = -\frac{1}{\alpha} I^L[v_x, \alpha](x) + v(x) - v(a) e^{-\alpha(x-a)},$$

$$A_L[v, \alpha] = -\frac{1}{\alpha} A_L[v_x, \alpha] + \frac{1}{1-\mu} \left(v(b) - v(a) e^{-\alpha(b-a)} \right) = -\frac{1}{\alpha} A_L[v_x, \alpha] + v(a).$$

Therefore, we have

$$\begin{aligned} \mathcal{D}_L[v, \alpha] &= \frac{1}{\alpha} I^L[v_x, \alpha](x) + \frac{1}{\alpha} A_L[v_x, \alpha] e^{-\alpha(x-a)} = \frac{1}{\alpha} \mathcal{L}_L^{-1}[v_x, \alpha](x) \\ &= \frac{1}{\alpha} v_x(x) - \frac{1}{\alpha} \mathcal{D}_L[v_x, \alpha](x) \\ &= - \sum_{p=1}^k \left(-\frac{1}{\alpha} \right)^p \partial_x^p v(x) + \left(-\frac{1}{\alpha} \right) \mathcal{L}_L^{-1}[\partial_x^{k+1} v, \alpha](x), \end{aligned}$$

and the lemma is proved.

References

1. Abedian, R., Adibi, H., Dehghan, M.: A high-order weighted essentially non-oscillatory (WENO) finite difference scheme for nonlinear degenerate parabolic equations. *Comput. Phys. Commun.* **184**(8), 1874–1888 (2013)
2. Alt, H.W., Luckhaus, S.: Quasilinear elliptic-parabolic differential equations. *Mathematische Zeitschrift* **183**(3), 311–341 (1983)
3. Aregba-Driollet, D., Natalini, R., Tang, S.: Explicit diffusive kinetic schemes for nonlinear degenerate parabolic systems. *Math. Comput.* **73**(245), 63–94 (2004)
4. Aronson, D.G.: The Porous Medium Equation. Nonlinear Diffusion Problems, pp. 1–46. Springer, Berlin (1986)
5. Ascher, U., Ruuth, S., Spiteri, R.: Implicit-explicit Runge-Kutta methods for time-dependent partial differential equations. *Appl. Numer. Math.* **25**(2–3), 151–167 (1997)
6. Barenblatt, G.I.: On self-similar motions of a compressible fluid in a porous medium. *Akad. Nauk SSSR. Prikl. Mat. Meh* **16**(6), 79–6 (1952)
7. Barnes, J., Hut, P.: A hierarchical $O(N \log N)$ force-calculation algorithm. *Nature* **324**, 446–449 (1986)
8. Bessemoulin-Chatard, M., Filbet, F.: A finite volume scheme for nonlinear degenerate parabolic equations. *SIAM J. Sci. Comput.* **34**(5), B559–B583 (2012)
9. Borges, R., Carmona, M., Costa, B., Don, W.S.: An improved weighted essentially non-oscillatory scheme for hyperbolic conservation laws. *J. Comput. Phys.* **227**(6), 3191–3211 (2008)
10. Buckley, S.E., Leverett, M., et al.: Mechanism of fluid displacement in sands. *Trans. AIME* **146**(01), 107–116 (1942)
11. Causley, M., Cho, H., Christlieb, A.: Method of lines transpose: energy gradient flows using direct operator inversion for phase field models (2016). *arXiv preprint arXiv:1611.04214*
12. Causley, M., Cho, H., Christlieb, A., Seal, D.: Method of lines transpose: High order L-stable $\mathcal{O}(N)$ schemes for parabolic equations using successive convolution. *SIAM J. Numer. Anal.* **54**(3), 1635–1652 (2016)
13. Causley, M., Christlieb, A., Ong, B., Van Groningen, L.: Method of lines transpose: an implicit solution to the wave equation. *Math. Comput.* **83**(290), 2763–2786 (2014)
14. Causley, M.F., Christlieb, A.J.: Higher order A-stable schemes for the wave equation using a successive convolution approach. *SIAM J. Numer. Anal.* **52**(1), 220–235 (2014)
15. Causley, M.F., Christlieb, A.J., Guclu, Y., Wolf, E.: Method of lines transpose: a fast implicit wave propagator (2013). *arXiv preprint arXiv:1306.6902*
16. Cavalli, F., Naldi, G., Puppo, G., Semplice, M.: High-order relaxation schemes for nonlinear degenerate diffusion problems. *SIAM J. Numer. Anal.* **45**(5), 2098–2119 (2007)
17. Cheng, Y., Christlieb, A.J., Guo, W., Ong, B.: An asymptotic preserving Maxwell solver resulting in the Darwin limit of electrodynamics. *J. Sci. Comput.* **71**, 1–35 (2015)
18. Chou, C.-S., Shu, C.-W.: High order residual distribution conservative finite difference WENO schemes for steady state problems on non-smooth meshes. *J. Comput. Phys.* **214**(2), 698–724 (2006)
19. Chou, C.-S., Shu, C.-W.: High order residual distribution conservative finite difference WENO schemes for convection-diffusion steady state problems on non-smooth meshes. *J. Comput. Phys.* **224**(2), 992–1020 (2007)

20. Christlieb, A., Guo, W., Jiang, Y.: A WENO-based Method of Lines Transpose approach for Vlasov simulations. *J. Comput. Phys.* **327**, 337–367 (2016)
21. Christlieb, A., Guo, W., Jiang, Y.: A kernel based high order “explicit” unconditionally stable scheme for time dependent Hamilton-Jacobi equations. *J. Comput. Phys.* **379**, 214–236 (2019)
22. Christlieb, A., Guo, W., Jiang, Y., Yang, H.: <https://github.com/hyoseonyang/molt-tutorial.git> (2019)
23. Duy, Cv, Peletier, L.: Nonstationary filtration in partially saturated porous media. *Arch. Ration. Mech. Anal.* **78**(2), 173–198 (1982)
24. Gottlieb, S.: On high order strong stability preserving Runge-Kutta and multi step time discretizations. *J. Sci. Comput.* **25**(1), 105–128 (2005)
25. Gottlieb, S., Shu, C.-W., Tadmor, E.: Strong stability-preserving high-order time discretization methods. *SIAM Rev.* **43**(1), 89–112 (2001)
26. Greengard, L., Rokhlin, V.: A fast algorithm for particle simulations. *J. Comput. Phys.* **73**(2), 325–348 (1987)
27. Gustafsson, B., Kreiss, H.-O., Sundström, A.: Stability theory of difference approximations for mixed initial boundary value problems. II. *Mathematics of Computation*, pp. 649–686, (1972)
28. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II: Stiff and Differential Algebraic Problems, 2nd edn. Springer, Berlin (1996)
29. Hajipour, M., Malek, A.: High accurate NRK and MWENO scheme for nonlinear degenerate parabolic PDEs. *Appl. Math. Modell.* **36**(9), 4439–4451 (2012)
30. Jia, J., Huang, J.: Krylov deferred correction accelerated method of lines transpose for parabolic problems. *J. Comput. Phys.* **227**(3), 1739–1753 (2008)
31. Jiang, G.-S., Shu, C.-W.: Efficient implementation of weighted eno schemes. *J. Comput. Phys.* **126**(1), 202–228 (1996)
32. Ketcheson, D.I.: Step sizes for strong stability preservation with downwind-biased operators. *SIAM J. Numer. Anal.* **49**(4), 1649–1660 (2011)
33. Kropinski, M.C.A., Quaife, B.D.: Fast integral equation methods for Rothe’s method applied to the isotropic heat equation. *Comput. Math. Appl.* **61**(9), 2436–2446 (2011)
34. Kurganov, A., Tadmor, E.: New high-resolution central schemes for nonlinear conservation laws and convection-diffusion equations. *J. Comput. Phys.* **160**(1), 241–282 (2000)
35. Liu, Y., Shu, C.-W., Zhang, M.: On the positivity of linear weights in WENO approximations. *Acta Math. Appl. Sin. Engl. Ser.* **25**(3), 503–538 (2009)
36. Liu, Y., Shu, C.-W., Zhang, M.: High order finite difference WENO schemes for nonlinear degenerate parabolic equations. *SIAM J. Sci. Comput.* **33**(2), 939–965 (2011)
37. Muskat, M.: The flow of homogeneous fluids through porous media. *Soil Sci.* **46**(2), 169 (1938)
38. Otto, F.: L1-contraction and uniqueness for quasilinear elliptic-parabolic equations. *J. Diff. Equ.* **131**(1), 20–38 (1996)
39. Salazar, A., Raydan, M., Campo, A.: Theoretical analysis of the exponential transversal method of lines for the diffusion equation. *Numer. Methods Partial Diff. Equ.* **16**(1), 30–41 (2000)
40. Schemann, M., Bornemann, F.A.: An adaptive rothe method for the wave equation. *Comput. Vis. Sci.* **1**(3), 137–144 (1998)
41. Shu, C.-W.: Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. In: *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations*, pp. 325–432. Springer (1998)
42. Shu, C.-W.: A survey of strong stability preserving high order time discretizations. *Collected Lectures Preserv. Stab. Discret.* **109**, 51–65 (2002)
43. Shu, C.-W.: High order weighted essentially nonoscillatory schemes for convection dominated problems. *SIAM Rev.* **51**(1), 82–126 (2009)
44. Shu, C.-W., Osher, S.: Efficient implementation of essentially non-oscillatory shock-capturing schemes. *J. Comput. Phys.* **77**(2), 439–471 (1988)
45. Zel’dovich, Y.B., Kompaneets, A.: Towards a theory of heat conduction with thermal conductivity depending on the temperature. In: *Collection of Papers Dedicated to 70th Birthday of Academician AF Ioffe*, Izd. Akad. Nauk SSSR, Moscow, pp. 61–71 (1950)
46. Zhang, Q., Wu, Z.-L.: Numerical simulation for porous medium equation by local discontinuous galerkin finite element method. *J. Sci. Comput.* **38**(2), 127–148 (2009)