Journal of Computational Physics 529 (2025) 113838

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

High-order exponential time differencing multi-resolution alternative finite difference WENO methods for nonlinear degenerate parabolic equations

Ziyao Xu, Yong-Tao Zhang [*,1]

Department of Applied and Computational Mathematics and Statistics, University of Notre Dame, Notre Dame, IN 46556, USA

ARTICLE INFO

Keywords:
Exponential time differencing
Multi-resolution WENO
Alternative finite difference WENO methods
Nonlinear degenerate parabolic equations

ABSTRACT

In this paper, we focus on the finite difference approximation of nonlinear degenerate parabolic equations, a special class of parabolic equations where the viscous term vanishes in certain regions. This vanishing gives rise to additional challenges in capturing sharp fronts, beyond the restrictive CFL conditions commonly encountered with explicit time discretization in parabolic equations. To resolve the sharp front, we adopt the high-order multi-resolution alternative finite difference WENO (A-WENO) methods for the spatial discretization, which is designed to effectively suppress oscillations in the presence of large gradients and achieve nonlinear stability. To alleviate the time step restriction from the nonlinear stiff diffusion terms, we employ the exponential time differencing Runge-Kutta (ETD-RK) methods, a class of efficient and accurate exponential integrators, for the time discretization. However, for highly nonlinear spatial discretizations such as high-order WENO schemes, it is a challenging problem how to efficiently form the linear stiff part in applying the exponential integrators, since direct computation of a Jacobian matrix for high-order WENO discretizations of the nonlinear diffusion terms is very complicated and expensive. Here we propose a novel and effective approach of replacing the exact Jacobian of high-order multi-resolution A-WENO scheme with that of the corresponding high-order linear scheme in the ETD-RK time marching, based on the fact that in smooth regions the nonlinear weights closely approximate the corresponding linear weights, while in non-smooth regions the stiff diffusion degenerates. The algorithm is described in detail, and numerous numerical experiments are conducted to demonstrate the effectiveness of such a treatment and the good performance of our method. The stiffness of the nonlinear parabolic partial differential equations (PDEs) is resolved well, and large time-step size computations of $\Delta t \sim O(\Delta x)$ are achieved.

1. Introduction

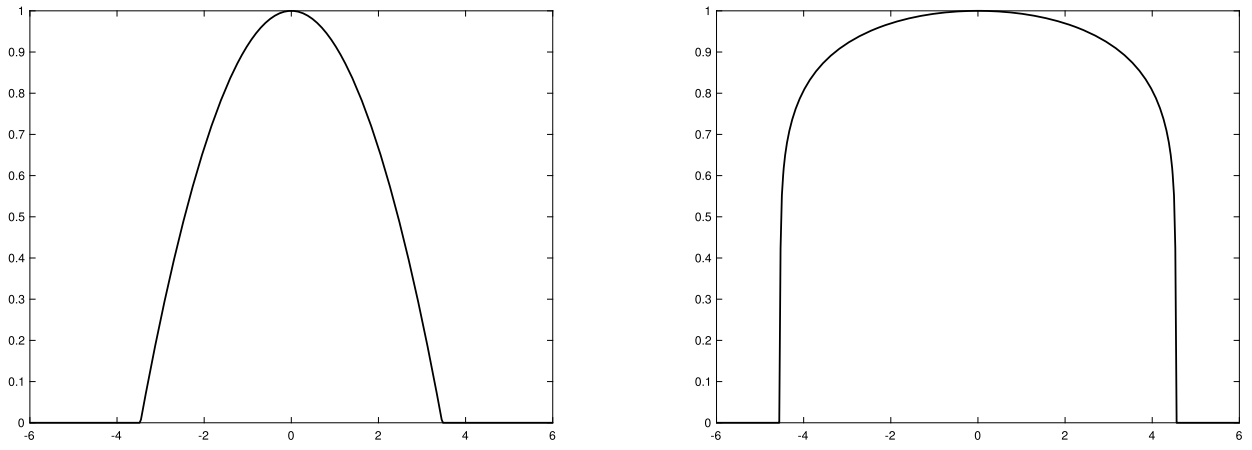
In this paper, we are concerned with the finite difference methods for nonlinear degenerate parabolic equations, which arise in many science and engineering applications. The equations are formulated as follows:

* Corresponding author.
E-mail addresses: zxu25@nd.edu (Z. Xu), yzhang10@nd.edu (Y.-T. Zhang).

[1] Research is partially supported by Simons Foundation MPS-TSM-00007854.

https://doi.org/10.1016/j.jcp.2025.113838
Received 7 June 2024; Received in revised form 23 December 2024; Accepted 8 February 2025

Available online 12 February 2025
0021-9991/© 2025 The Authors. Published by Elsevier Inc. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).



(a) $m = 2$ (b) $m = 8$

Fig. 1.1. The Barenblatt solutions (1.4) of PME (1.3) at $t = 1$.

$$u_t + f(u)_x = g(u)_{xx} := (a(u)u_x)_x, \tag{1.1}$$

in one space dimension, where $f(u)$ is the flux of the hyperbolic term, and $g(u)$ is the function of the degenerate parabolic term. The diffusion coefficient $a(u) = g'(u)$ vanishes for certain regions of u. Similarly, in two space dimensions, the nonlinear degenerate parabolic equations are formulated as:

$$u_t + f_1(u)_x + f_2(u)_y = \nabla \cdot (\nabla g(u)), \tag{1.2}$$

where the function of the parabolic term $g(u)$ may degenerate. Due to the vanishing of parabolic terms, the equations (1.1) and (1.2) exhibit some features of hyperbolic equations, e.g., the existence of non-smooth weak solutions and the propagation of sharp wave fronts with finite speed.

A prototypical example of nonlinear degenerate parabolic equations is the porous media equation (PME) [47,6]:

$$u_t = (u^m)_{xx}, \tag{1.3}$$

used to model gas flow in porous media, where $m > 1$ is a constant, and $u \geq 0$ denotes the fluid density. The PME admits the well-known Barenblatt weak solution:

$$B_m(x,t) = t^{-p}\left(\left(1 - \frac{p(m-1)}{2m}\frac{|x|^2}{t^{2p}}\right)^+\right)^{\frac{1}{m-1}}, \tag{1.4}$$

where $z^+ := \max(z, 0)$ and $p = \frac{1}{m+1}$. The wave fronts $x = \pm t^p\sqrt{\frac{2m}{p(m-1)}}$ in the Barenblatt solution propagate at finite speed, as illustrated by the solutions at $t = 1$ in Fig. 1.1.

Due to the existence of sharp fronts caused by the hyperbolic features of the equations (1.1) and (1.2), high-order linear schemes suffer from spurious oscillations in the presence of large gradients, i.e. the Gibbs phenomenon, even though they work well in smooth regions. Various numerical methods with nonlinear components have been developed for solving nonlinear degenerate parabolic PDEs, for example, linear approximation scheme based on the nonlinear Chernoff formula with a relaxation parameter [45,51], local discontinuous Galerkin method [66], finite volume method [9], kinetic schemes [5], high-order relaxation schemes [14], kernel based integral method [16], moving mesh finite element method [49], etc.

In the literature (e.g. [41,1,4,28,2,65,3]), it has been shown that the weighted essentially non-oscillatory (WENO) methods, a popular class of high-order accuracy schemes for solving hyperbolic PDEs, are very effective to be adopted in the spatial discretization of degenerate parabolic equations, to achieve high-order accuracy in smooth regions while suppressing oscillations near wave fronts. The WENO methods are based on the successful essentially non-oscillatory (ENO) methods with additional advantages. The ENO methods were first proposed by Harten et al. [23] in the finite volume framework for hyperbolic equations, adopting the smoothest stencil among several candidates to reconstruct the solutions. To enhance efficiency, Shu and Osher proposed the finite difference framework in the subsequent work [54,55], which allow for computation in a dimension-splitting fashion. The WENO methods use weighted stencils rather than choosing only one stencil, to reconstruct solutions. The weighting strategy is a crucial component of WENO methods, designed based on the principle that in smooth regions of the solution, the nonlinear weights are close to the linear weights in linear schemes to yield improvement in accuracy, while being close to zero to minimize the contribution of stencils with large gradient of the solution. Various types of WENO methods have been proposed since the celebrated WENO methods developed in [39,30], e.g., very high-order finite difference WENO schemes [8], the mapped WENO schemes [24] and WENO-Z schemes [11,13]

for optimal order near critical points, central WENO schemes [35], energy stable WENO schemes [64], robust WENO schemes [42], among others.

Recently in [67], the multi-resolution WENO schemes were developed to solve hyperbolic conservation laws. This class of WENO schemes construct WENO approximations on unequal-sized substencils and exhibit interesting properties. For example, it is flexible to construct linear weights for the multi-resolution WENO schemes, which in general can be taken as arbitrary positive numbers with the only requirement that their sum equals 1. Such flexibility simplifies the procedure in applications of WENO schemes, e.g., sparse-grid simulations for multidimensional problems [59], treatment of negative linear weights for solving degenerate parabolic equations [28], etc. In this paper, similar to [28], we adopt the multi-resolution WENO method [67] in the finite difference framework developed in [54] for the spatial discretization of nonlinear degenerate parabolic equations. The WENO method based on the finite difference framework in [54] is referred to as the alternative finite difference WENO (A-WENO) method in [29] to distinguish it from the more commonly practiced one [30,55]. Significant advantages of the A-WENO method include that arbitrary monotone fluxes can be used in this framework [29], and it allows for a flexible choice of interpolated variables [62,63].

Efficient and high-order temporal numerical schemes are crucial for the performance of high accuracy numerical simulations of time-dependent PDEs. For purely hyperbolic equations, explicit methods, e.g., the strong stability preserving Runge-Kutta (SSP-RK) methods and multi-step methods [53,22,19,20], and Lax-Wendroff type methods [52,36,61] are widely adopted, as these explicit schemes require less computational effort per time step than implicit schemes and simulations are generally efficient with time-step size Δt proportional to spatial grid size Δx (i.e., $\Delta t \sim O(\Delta x)$) under the Courant-Friedrichs-Lewy (CFL) condition. However, in the context of parabolic equations, time-step sizes of explicit schemes under the CFL condition are much more restrictive which requires $\Delta t \sim O(\Delta x^2)$, due to the stiffness introduced by the diffusion terms. In such scenarios, implicit schemes are more desirable to allow for considerably larger time-step sizes, albeit with a significant rise in computational cost per time step. Computational efficiency of fully implicit schemes can be improved by semi-implicit or implicit-explicit (IMEX) methods, e.g., [7,33,12,27]. For solving nonlinear degenerate parabolic equations by high-order WENO schemes, most of work in the literature uses an explicit approach and small time-step sizes are required, which lead to relatively high computational costs. Recently in [4,65], implicit and semi-implicit approaches for WENO schemes were developed to show much better efficiency than explicit schemes. Another class of efficient temporal schemes for stiff problems are exponential integrators [25]. The exponential integrators perform an "exact" integration of stiff linear part of the problem to remove the severe time-step size restriction. Among explicit exponential integrators, a popular class of methods are the exponential time differencing (ETD) approaches, e.g., ETD multi-step methods and ETD Runge-Kutta (ETD-RK) methods [10,17,18,32,38]. ETD schemes have the advantages such as relatively small numerical errors, good steady-state preservation property, etc. [10,32]. To deal with stiff nonlinear reaction terms in reaction-diffusion equations and advection-diffusion-reaction equations, a class of implicit exponential integrators, called "implicit integration factor" (IIF) methods [48,15,31,43,40], were developed such that the implicit terms are free of the exponential operation of the stiff linear terms for achieving efficient computations.

For the fully nonlinear stiff problems such as the degenerate parabolic equations considered in this paper, the exponential integrators such as the ETD schemes are still very efficient methods to tackle the stiffness explicitly and accurately. Here we adopt the ETD-RK methods in the time stepping due to their larger stability regions than the ETD multi-step methods [17]. To handle the full nonlinearity, a popular approach is to use the exponential Rosenborg-type method [26] or the exponential propagation iterative method [57], which form the linear stiff part for applying the exponential integrators by computing the Jacobian of the nonlinear stiff system in each time step around the numerical solution. However, for highly nonlinear high-order spatial schemes such as the multi-resolution A-WENO schemes used here, it is a challenging problem how to efficiently form the linear stiff part in applying the exponential integrators, since direct computation of a Jacobian matrix based on high-order A-WENO discretizations of the nonlinear diffusion terms to obtain the linear stiff part is very complicated and expensive. In this paper, we propose a novel and effective approach to solve this difficulty by replacing the exact Jacobian of a high-order multi-resolution A-WENO scheme with that of the corresponding high-order linear scheme in the ETD-RK time marching, based on the fact that in smooth regions the nonlinear weights of WENO schemes closely approximate the corresponding linear weights, while in non-smooth regions the stiff diffusion degenerates.

The remaining of the paper is organized as follows. In Section 2, we formulate the numerical methods in details for the nonlinear degenerate parabolic equations, incorporating the multi-resolution A-WENO spatial discretization and ETD-RK time integration by the proposed novel approach. A comprehensive set of numerical experiments and comparisons with traditional methods are performed in Section 3 to demonstrate the effectiveness of the proposed approach, and verify high-order accuracy, nonlinear stability and high efficiency of the new methods. The stiffness of the nonlinear degenerate parabolic PDEs is resolved well, and large time-step size computations of $\Delta t \sim O(\Delta x)$ are achieved. Conclusions and discussions of the paper are provided in Section 4.

2. The numerical methods

In this section, we present the details of the numerical methods that will be employed in the subsequent numerical tests. The spatial discretization is based on the multi-resolution WENO interpolations in the alternative formulation of the finite difference method. The ETD-RK methods are adopted to evolve the nonlinear ODE system resulted from the spatial discretization, with the linear stiff component of the exponential integrators derived from the corresponding high-order linear schemes for the diffusion terms.

2.1. Spatial discretization

2.1.1. Discretization for the diffusion terms

We first consider the parabolic equations with only diffusion terms. In one space dimension, the general form of the equation is

$$u_t = g(u)_{xx}. \tag{2.1}$$

A uniform grid $\cdots < x_0 < x_1 < x_2 < \cdots$ is adopted for computation, where $x_i, i = 0, \pm 1, \dots$ are the grid points, and Δx is the grid size. The finite difference method for the diffusion equation (2.1) is formulated as follows:

$$\frac{du_i}{dt} = \frac{\hat{b}_{i+\frac{1}{2}} - \hat{b}_{i-\frac{1}{2}}}{\Delta x}, \quad i = 0, \pm 1, \dots, \tag{2.2}$$

where the grid function $\{u_i\}_{\forall i}$ approximates the function values $u(x_i, t)$ at grid points, and $\hat{b}_{i+\frac{1}{2}}$ is the discretization for the flux $b := g(u)_x$ at the interface $x_{i+\frac{1}{2}} := (x_i + x_{i+1})/2$. Here and henceforth, we denote $g_i = g(u_i)$ for brevity.

To achieve the $2r$-th order accurate approximation

$$\frac{\hat{b}_{i+\frac{1}{2}} - \hat{b}_{i-\frac{1}{2}}}{\Delta x} = g(u)_{xx}|_{x=x_i} + O(\Delta x^{2r}),$$

the numerical flux $\hat{b}_{i+\frac{1}{2}}$, which depends on $g_{i-r+1}, \dots, g_{i+r}$ in the alternative finite difference formulation [54,28], is taken as

$$\hat{b}_{i+\frac{1}{2}} = \sum_{m=0}^{r-1} c_m \Delta x^{2m} \left(\widehat{\partial_x^{2m+1} g}\right)_{i+\frac{1}{2}}, \tag{2.3}$$

where the constant coefficients $c_0 = 1, c_1 = -\frac{1}{24}, c_2 = \frac{7}{5760}, c_3 = -\frac{31}{967680}, \dots$ are determined from Taylor expansion to attain the designed accuracy, and $(\widehat{\partial_x^{2m+1} g})_{i+\frac{1}{2}}$ are approximations to $\partial_x^{2m+1} g$ at the interface $x_{i+\frac{1}{2}}$, for $m = 0, 1, \dots, r-1$.

When the parabolic equation is degenerate, it is desired to control spurious oscillations around sharp fronts and achieve nonlinear stability of simulation. Following [28], we adopt the multi-resolution WENO interpolations to obtain the lowest order term $\widehat{(\partial_x g)}_{i+\frac{1}{2}}$, and use central numerical differentiation (linear scheme) to calculate the high order ones $(\widehat{\partial_x^{2m+1} g})_{i+\frac{1}{2}}, m \geq 1$. The $2r$-th order multi-resolution WENO procedure for obtaining $\widehat{(\partial_x g)}_{i+\frac{1}{2}}$ is outlined as follows.

- Step 1: find Lagrange interpolation on nested stencils.
  In the multi-resolution WENO, we choose the r nested stencils $S^{(1)} = \{x_i, x_{i+1}\}$, $S^{(2)} = \{x_{i-1}, x_i, x_{i+1}, x_{i+2}\}$,..., $S^r = \{x_{i-r+1}, \dots, x_{i+r}\}$, and perform the Lagrange interpolation on the stencils, i.e., find polynomials $p_k(x)$ of degree $2k-1$, for $k = 1, 2, \dots, r$, such that

  $$p_k(x_j) = g_j, \quad x_j \in S^{(k)}. \tag{2.4}$$

  Consequently, we approximate $g'(x_{i+\frac{1}{2}})$ by $p_k'(x_{i+\frac{1}{2}})$ with accuracy $O(\Delta x^{2k})$:

  $$\begin{aligned}
  p_1'(x_{i+\frac{1}{2}}) &= \frac{1}{\Delta x}(-g_i + g_{i+1}),\\
  p_2'(x_{i+\frac{1}{2}}) &= \frac{1}{24\Delta x}(g_{i-1} - 27g_i + 27g_{i+1} - g_{i+2}),\\
  p_3'(x_{i+\frac{1}{2}}) &= \frac{1}{1920\Delta x}(-9g_{i-2} + 125g_{i-1} - 2250g_i + 2250g_{i+1} - 125g_{i+2} + 9g_{i+3}),\\
  p_4'(x_{i+\frac{1}{2}}) &= \frac{1}{107520\Delta x}(75g_{i-3} - 1029g_{i-2} + 8575g_{i-1} - 128625g_i + 128625g_{i+1}\\
  &\quad - 8575g_{i+2} + 1029g_{i+3} - 75g_{i+4}),\\
  &\dots
  \end{aligned} \tag{2.5}$$

- Step 2: form linear weights for interpolation.
  We write the highest order interpolation polynomial $p_r(x)$ as a convex combination of polynomials $q_1(x), q_2(x), \dots, q_r(x)$ with linear coefficients $\{d_k\}_{k=1}^r$ as follows:

  $$p_r(x) = \sum_{k=1}^{r} d_k q_k(x), \tag{2.6}$$

  where

$$d_k = \theta_{k,r}, \quad \theta_{n,m} := \frac{\tilde{\theta}_{n,m}}{\sum_{\ell=1}^{m} \tilde{\theta}_{\ell,m}}, \quad 1 \le n \le m \le r, \tag{2.7}$$

and

$$\begin{aligned} q_1(x) &= p_1(x), \\ q_k(x) &= \frac{1}{\theta_{k,k}} p_k(x) - \sum_{\ell=1}^{k-1} \frac{\theta_{\ell,k}}{\theta_{k,k}} q_\ell(x), \quad k = 2, 3, \dots, r. \end{aligned} \tag{2.8}$$

Following the practice in [67,28], we take $\tilde{\theta}_{n,m} = 10^{n-1}$. Consequently, we have the linear weights

$$\begin{aligned} d_1 &= 1, \quad \text{for } r = 1, \\ d_1 &= \frac{1}{11}, d_2 = \frac{10}{11}, \quad \text{for } r = 2, \\ d_1 &= \frac{1}{111}, d_2 = \frac{10}{111}, d_3 = \frac{100}{111}, \quad \text{for } r = 3, \\ d_1 &= \frac{1}{1111}, d_2 = \frac{10}{1111}, d_3 = \frac{100}{1111}, d_4 = \frac{1000}{1111}, \quad \text{for } r = 4, \\ &\dots \end{aligned} \tag{2.9}$$

and

$$\begin{aligned} q_1(x) &= p_1(x), \\ q_2(x) &= \frac{11}{10} p_2(x) - \frac{1}{10} q_1(x), \\ q_3(x) &= \frac{111}{100} p_3(x) - \frac{1}{10} q_2(x) - \frac{1}{100} q_1(x), \\ q_4(x) &= \frac{1111}{1000} p_4(x) - \frac{1}{10} q_3(x) - \frac{1}{100} q_2(x) - \frac{1}{1000} q_1(x), \\ &\dots \end{aligned} \tag{2.10}$$

- Step 3: compute nonlinear weights for interpolation.
  The nonlinear weights ω_k are obtained based on the smoothness indicators β_k and the linear weights d_k for the polynomials $q_k(x)$, $k = 1, 2, \dots, r$. Following the practice in [28], we use the smoothness indicator measuring the smoothness of $p_k(x)$ instead of $q_k(x)$ on the interval $[x_i, x_{i+1}]$, as $p_k(x)$ is the major component of $q_k(x)$. It is defined as

$$\beta_k = \sum_{m=1}^{2k-1} \Delta x^{2m-1} \int_{x_i}^{x_{i+1}} \left(\frac{d^m p_k(x)}{dx^m} \right)^2 dx. \tag{2.11}$$

  The detailed expressions of $\beta_1, \beta_2, \beta_3$ and β_4 are given in the Appendix A. Consequently, we calculate the nonlinear weights ω_k as

$$\omega_k = \frac{\tilde{\omega}_k}{\sum_{m=1}^{r} \tilde{\omega}_m}, \quad \tilde{\omega}_k = d_k \left(1 + \frac{\tau_r}{\beta_k + \epsilon} \right), \quad k = 1, 2, \dots, r, \tag{2.12}$$

  where $\tau_r = \left(\sum_{m=1}^{r-1} |\beta_m - \beta_r| \right)^{\frac{r+1}{2}}$, and ϵ is a small positive number to avoid dividing by zero, e.g., $\epsilon = 10^{-10}$.
- Step 4: compute multi-resolution WENO interpolation.
  Finally, we obtain the multi-resolution WENO interpolation polynomial by replacing the linear weights with the nonlinear weights in (2.6):

$$Q(x) = \sum_{k=1}^{r} \omega_k q_k(x),$$

  and consequently,

$$\widehat{(\partial_x g)}_{i+\frac{1}{2}} = \sum_{k=1}^{r} \omega_k q_k'(x_{i+\frac{1}{2}}). \tag{2.13}$$

It remains to calculate the high-order derivative terms in (2.3), which is based on linear central numerical differentiation with much less computational costs than the first-order derivative term. For different orders of approximation, one can calculate:

- $r = 2$:

$$c_1 \Delta x^2 \widehat{\left(\partial_x^3 g\right)}_{i+\frac{1}{2}} = \frac{1}{24\Delta x} (g_{i-1} - 3g_i + 3g_{i+1} - g_{i+2}), \tag{2.14}$$

- $r = 3$:

$$\sum_{m=1}^{r-1} c_m \Delta x^{2m} \left(\widehat{\partial_x^{2m+1} g}\right)_{i+\frac{1}{2}} = \frac{1}{5760\Delta x}(-37g_{i-2} + 425g_{i-1} - 1090g_i + 1090g_{i+1} - 425g_{i+2} + 37g_{i+3}), \tag{2.15}$$

- $r = 4$:

$$\sum_{m=1}^{r-1} c_m \Delta x^{2m} \left(\widehat{\partial_x^{2m+1} g}\right)_{i+\frac{1}{2}} = \frac{1}{322560\Delta x}(351g_{i-3} - 4529g_{i-2} + 31171g_{i-1} - 73325g_i + 73325g_{i+1} - 31171g_{i+2} + 4529g_{i+3} - 351g_{i+4}). \tag{2.16}$$

The $2r$-th order alternative formulation of finite difference multi-resolution WENO scheme for the one-dimensional diffusion equation (2.1) is then obtained by integrating (2.2), (2.3), (2.13), (2.14) (or (2.15), (2.16)).

Likewise, for the two-dimensional diffusion equation

$$u_t = g(u)_{xx} + g(u)_{yy}, \tag{2.17}$$

we adopt the uniform grid $\cdots < x_0 < x_1 < x_2 < \cdots$ and $\cdots < y_0 < y_1 < y_2 < \cdots$ for the x and y directions, respectively, where $(x_i, y_j), i, j = 0, \pm 1, \ldots$ are grid points, and $\Delta x, \Delta y$ are the grid sizes.

The alternative formulation of finite difference method for the two-dimensional diffusion equation (2.17) is formulated as follows:

$$\frac{du_{i,j}}{dt} = \frac{\hat{b}^1_{i+\frac{1}{2},j} - \hat{b}^1_{i-\frac{1}{2},j}}{\Delta x} + \frac{\hat{b}^2_{i,j+\frac{1}{2}} - \hat{b}^2_{i,j-\frac{1}{2}}}{\Delta y}, \quad i, j = 0, \pm 1, \ldots, \tag{2.18}$$

where the grid function $\{u_{i,j}\}_{\forall i,j}$ approximates the function values $u(x_i, y_j, t)$ at grid points, and $\hat{b}^1_{i+\frac{1}{2},j}$ and $\hat{b}^2_{i,j+\frac{1}{2}}$ are the discretizations for the fluxes $b^1 := g(u)_x$ and $b^2 := g(u)_y$ at $(x_{i+\frac{1}{2}}, y_j)$ and $(x_i, y_{j+\frac{1}{2}})$, respectively. Since we are using finite difference method, the discretization in (2.18) is performed in a dimension by dimension fashion, and the computation for each dimension follows exactly the same procedure as we have established for the one-dimensional case. The equation in the 3D case is handled in exactly the same manner and is therefore omitted.

The resulting ODE system from the scheme (2.2) for the one-dimensional problems or the scheme (2.18) for the two-dimensional problems is generally denoted by

$$\mathbf{u}_t = G(\mathbf{u}), \tag{2.19}$$

where $\mathbf{u}$ is the vector $\{u_i\}_{\forall i}$ in the one-dimensional case, or $\{u_{i,j}\}_{\forall i,j}$ in the two-dimensional case, at grid points.

To prepare for the ETD time evolution method in the later subsections, we describe the corresponding linear scheme of (2.19) as follows. This scheme is obtained by replacing the nonlinear weights ω_k's with the linear weights d_k's in the WENO interpolation (2.13):

$$\mathbf{u}_t = G_L(\mathbf{u}). \tag{2.20}$$

In one space dimension, the $2r$-th order linear scheme (2.20) is given as follows:

- $r = 1$:

$$\frac{du_i}{dt} = \frac{1}{\Delta x^2}(g_{i-1} - 2g_i + g_{i+1}), \tag{2.21}$$

- $r = 2$:

$$\frac{du_i}{dt} = \frac{1}{12\Delta x^2}(-g_{i-2} + 16g_{i-1} - 30g_i + 16g_{i+1} - g_{i+2}), \tag{2.22}$$

- $r = 3$:

$$\frac{du_i}{dt} = \frac{1}{180\Delta x^2}(2g_{i-3} - 27g_{i-2} + 270g_{i-1} - 490g_i + 270g_{i+1} - 27g_{i+2} + 2g_{i+3}), \tag{2.23}$$

- $r = 4$:

$$\frac{du_i}{dt} = \frac{1}{5040\Delta x^2}(-9g_{i-4} + 128g_{i-3} - 1008g_{i-2} + 8064g_{i-1} - 14350g_i + 8064g_{i+1} - 1008g_{i+2} + 128g_{i+3} - 9g_{i+4}). \tag{2.24}$$

The formulation of the $2r$-th order linear scheme (2.20) in two and three space dimensions is obtained in a dimension by dimension manner.

2.1.2. Discretization for the convection terms

In this section, we describe the multi-resolution WENO discretization for the convection terms in the degenerate parabolic equations, using the alternative formulation of finite difference method. Consider the one-dimensional convection equation

$$u_t + f(u)_x = 0. \tag{2.25}$$

The semi-discrete conservative finite difference scheme is formulated as

$$\frac{du_i}{dt} = -\frac{\hat{f}_{i+\frac{1}{2}} - \hat{f}_{i-\frac{1}{2}}}{\Delta x}, \quad i = 0, \pm 1, \dots, \tag{2.26}$$

where $\{u_i\}_{\forall i}$ is the grid function to approximate $u(x_i, t)$ at grid points, and $\hat{f}_{i+\frac{1}{2}}$ is the numerical flux to approximate $f(u)$ at the interface $x_{i+\frac{1}{2}}$.

To achieve the $2r$-th order accurate approximation, the numerical flux $\hat{f}_{i+\frac{1}{2}}$, which depends on $u_{i-r}, \dots, u_{i+r+1}$, is defined in a similar fashion as for the diffusion equations:

$$\hat{f}_{i+\frac{1}{2}} = h(u^-_{i+\frac{1}{2}}, u^+_{i+\frac{1}{2}}) + \sum_{m=1}^{r-1} c_m \Delta x^{2m} \widehat{\left(\partial_x^{2m} f\right)}_{i+\frac{1}{2}}. \tag{2.27}$$

Here the coefficients $c_1, c_2, c_3, \dots$ are the same as in (2.3). $h(\cdot,\cdot)$ is the numerical flux function of $f(u)$ obtained from an exact or approximate Riemann solver, e.g., the Lax-Friedrichs flux. $u^{\pm}_{i+\frac{1}{2}}$ is computed using the multi-resolution WENO interpolation for the grid function $\{u_j\}_{\forall j}$ at the interface $x_{i+\frac{1}{2}}$ with left/right bias, and $\widehat{(\partial_x^{2m} f)}_{i+\frac{1}{2}}$ are approximations to $\partial_x^{2m} f$ at $x_{i+\frac{1}{2}}$, for $m = 1, \dots, r-1$. In the following, we first outline the $(2r+1)$-th order multi-resolution WENO procedure for computing $u^-_{i+\frac{1}{2}}$. The procedure of the multi-resolution WENO interpolation for $u^+_{i+\frac{1}{2}}$ follows similarly in that it is mirror-symmetric with respect to $x_{i+\frac{1}{2}}$.

- Step 1: find Lagrange interpolation on nested stencils.
  In the multi-resolution WENO interpolation, we choose the nested stencils $S^{(0)} = \{x_i\}$, $S^{(1)} = \{x_{i-1}, x_i, x_{i+1}\}, \dots$, $S^{(r)} = \{x_{i-r}, \dots, x_{i+r}\}$, and perform the Lagrange interpolation on these stencils, i.e., find polynomials $p_k(x)$ of degree $2k$ for $k = 0, 1, \dots, r$, such that,

  $$p_k(x_j) = u_j, \quad x_j \in S^{(k)}. \tag{2.28}$$

  Consequently, we use $p_k(x_{i+\frac{1}{2}})$ to approximate $u(x_{i+\frac{1}{2}})$ with accuracy Δx^{2k+1} as follows:

  $$\begin{aligned}
  p_0(x_{i+\frac{1}{2}}) &= u_i, \\
  p_1(x_{i+\frac{1}{2}}) &= \frac{1}{8}(-u_{i-1} + 6u_i + 3u_{i+1}), \\
  p_2(x_{i+\frac{1}{2}}) &= \frac{1}{128}(3u_{i-2} - 20u_{i-1} + 90u_i + 60u_{i+1} - 5u_{i+2}), \\
  p_3(x_{i+\frac{1}{2}}) &= \frac{1}{1024}(-5u_{i-3} + 42u_{i-2} - 175u_{i-1} + 700u_i + 525u_{i+1} - 70u_{i+2} + 7u_{i+3}), \\
  p_4(x_{i+\frac{1}{2}}) &= \frac{1}{32768}(35u_{i-4} - 360u_{i-3} + 1764u_{i-2} - 5880u_{i-1} + 22050u_i + 17640u_{i+1} \\
  &\quad - 2940u_{i+2} + 504u_{i+3} - 45u_{i+4}), \\
  &\dots
  \end{aligned} \tag{2.29}$$

- Step 2: form linear weights for interpolation.
  We write the highest order interpolation polynomial $p_r(x)$ as a convex combination of polynomials $q_0(x), q_1(x), \dots, q_r(x)$ with linear coefficients $\{d_k\}_{k=0}^r$ as follows:

  $$p_r(x) = \sum_{k=0}^{r} d_k q_k(x), \tag{2.30}$$

  where

  $$d_k = \theta_{k,r}, \quad \theta_{n,m} := \frac{\tilde{\theta}_{n,m}}{\sum_{\ell=0}^{m} \tilde{\theta}_{\ell,m}}, \quad 0 \le n \le m \le r, \tag{2.31}$$

and

$$\begin{aligned} q_0(x) &= p_0(x), \\ q_k(x) &= \frac{1}{\theta_{k,k}} p_k(x) - \sum_{\ell=0}^{k-1} \frac{\theta_{\ell,k}}{\theta_{k,k}} q_\ell(x), \quad k = 1, 2, \dots, r. \end{aligned} \tag{2.32}$$

In practice, we take $\tilde{\theta}_{n,m} = 10^n$. Consequently, we have the linear weights

$$\begin{aligned} &d_0 = 1, \quad \text{for } r = 0, \\ &d_0 = \frac{1}{11}, d_1 = \frac{10}{11}, \quad \text{for } r = 1, \\ &d_0 = \frac{1}{111}, d_1 = \frac{10}{111}, d_2 = \frac{100}{111}, \quad \text{for } r = 2, \\ &d_0 = \frac{1}{1111}, d_1 = \frac{10}{1111}, d_2 = \frac{100}{1111}, d_3 = \frac{1000}{1111}, \quad \text{for } r = 3, \\ &d_0 = \frac{1}{11111}, d_1 = \frac{10}{11111}, d_2 = \frac{100}{11111}, d_3 = \frac{1000}{11111}, d_4 = \frac{10000}{11111} \quad \text{for } r = 4, \\ &\dots \end{aligned} \tag{2.33}$$

and

$$\begin{aligned} q_0(x) &= p_0(x), \\ q_1(x) &= \frac{11}{10} p_1(x) - \frac{1}{10} q_0(x), \\ q_2(x) &= \frac{111}{100} p_2(x) - \frac{1}{10} q_1(x) - \frac{1}{100} q_0(x), \\ q_3(x) &= \frac{1111}{1000} p_3(x) - \frac{1}{10} q_2(x) - \frac{1}{100} q_1(x) - \frac{1}{1000} q_0(x), \\ q_4(x) &= \frac{11111}{10000} p_4(x) - \frac{1}{10} q_3(x) - \frac{1}{100} q_2(x) - \frac{1}{1000} q_1(x) - \frac{1}{10000} q_0(x), \\ &\dots \end{aligned} \tag{2.34}$$

- Step 3: compute nonlinear weights for interpolation.
  For $k = 1, \dots, r$, we compute the nonlinear weight ω_k for the multi-resolution WENO interpolation, based on the smoothness indicator β_k which measures the smoothness of $q_k(x)$ on the interval $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$:

$$\beta_k = \sum_{m=1}^{2k} \Delta x^{2m-1} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \left(\frac{d^m q_k(x)}{dx^m} \right)^2 dx, \quad k \geq 1. \tag{2.35}$$

  The detailed expressions of $\beta_1, \beta_2, \beta_3$ and β_4 are given in the Appendix B. The only exception for (2.35) is β_0, which is magnified from zero to a tiny value. See [67] for details. Consequently, we calculate the nonlinear weights ω_k as

$$\omega_k = \frac{\tilde{\omega}_k}{\sum_{m=0}^{r} \tilde{\omega}_m}, \quad \tilde{\omega}_k = d_k \left(1 + \frac{\tau_r}{\beta_k + \epsilon} \right), \quad k = 0, 1, 2, \dots, r, \tag{2.36}$$

  where $\tau_r = \left(\frac{1}{r} \sum_{m=0}^{r-1} |\beta_m - \beta_r| \right)^r$, and ϵ is a small positive number to avoid dividing by zero, e.g. $\epsilon = 10^{-10}$.
- Step 4: compute multi-resolution WENO interpolation.
  Finally, the multi-resolution WENO interpolation $u^-_{i+\frac{1}{2}}$ is obtained by replacing the linear weights d_k's with the nonlinear weights ω_k's in (2.30) and evaluating at $x_{i+\frac{1}{2}}$:

$$u^-_{i+\frac{1}{2}} = \sum_{k=0}^{r} \omega_k q_k(x_{i+\frac{1}{2}}).$$

It remains to calculate the high-order derivative terms in (2.27), which is based on linear central numerical differentiation with much less computational costs than the first term. Here and henceforth, we denote $f(u_i)$ by f_i for brevity. The high-order derivative terms are computed as follows:

- $r = 2$:

$$c_1 \Delta x^2 (\widehat{\partial_x^2 f})_{i+\frac{1}{2}} = \frac{1}{48}(-f_{i-1} + f_i + f_{i+1} - f_{i+2}), \tag{2.37}$$

- $r = 3$:

$$\sum_{m=1}^{r-1} c_m \Delta x^{2m} \left(\widehat{\partial_x^{2m} f}\right)_{i+\frac{1}{2}} = \frac{1}{3840}(19f_{i-2} - 137f_{i-1} + 118f_i + 118f_{i+1} - 137f_{i+2} + 19f_{i+3}), \tag{2.38}$$

- $r = 4$:

$$\sum_{m=1}^{r-1} c_m \Delta x^{2m} \left(\widehat{\partial_x^{2m} f}\right)_{i+\frac{1}{2}} = \frac{1}{215040}(-243f_{i-3} + 2279f_{i-2} - 9859f_{i-1} + 7823f_i + 7823f_{i+1} - 9859f_{i+2} + 2279f_{i+3} - 243f_{i+4}). \tag{2.39}$$

The discretizations for the two-dimensional convection equation

$$u_t + f_1(u)_x + f_2(u)_y = 0 \tag{2.40}$$

are performed in a dimension by dimension manner, with each spatial dimension being exactly the same as in the one-dimensional case. Also, the same procedure is applied to the three-dimensional case.

To this end, we denote the resulting ODE system from these finite difference spatial discretizations of the convection equations by

$$\mathbf{u}_t = F(\mathbf{u}), \tag{2.41}$$

for all the one-, two-, and three-dimensional cases.

Remark 2.1. The major difference between the scheme (2.2) to approximate the diffusion term and the scheme (2.26) to approximate the convection term is that a monotone numerical flux function $h(\cdot,\cdot)$ must be used to compute the first term (the major term) in the numerical flux $\hat{f}_{i+\frac{1}{2}}$ as in (2.27). This monotone numerical flux is an exact or approximate Riemann solver, which is essential in numerical approximations of hyperbolic terms to maintain the upwinding and linear stability of the numerical scheme. In this paper, we use the Lax-Friedrichs monotone flux for $h(\cdot,\cdot)$ in (2.27), and it takes the following form

$$h(u^-_{i+\frac{1}{2}}, u^+_{i+\frac{1}{2}}) = \frac{1}{2}[f(u^-_{i+\frac{1}{2}}) + f(u^+_{i+\frac{1}{2}}) - \alpha(u^+_{i+\frac{1}{2}} - u^-_{i+\frac{1}{2}})]. \tag{2.42}$$

Here $u^-_{i+\frac{1}{2}}$ and $u^+_{i+\frac{1}{2}}$ are numerical approximations for the value $u(x_{i+\frac{1}{2}})$, based on left and right biased stencils respectively. $\alpha = \max_u |f'(u)|$ is a constant and the maximum is taken over the relevant range of u on all grid points. However, for the diffusion term, stencils centered around the target point (e.g. the point $x_{i+\frac{1}{2}}$) are directly used for computing the numerical flux (e.g. $\hat{b}_{i+\frac{1}{2}}$ in (2.3)), based on the property of parabolic terms, which is different from the upwinding and biased-stencil requirement for approximating hyperbolic terms. Note that both the numerical flux (2.3) for the diffusion term and the numerical flux (2.27) for the convection term are based on the Taylor expansion in the alternative finite difference formulation [54,28], and these high-order derivative terms in (2.3) and (2.27) contribute to the desired accuracy in high-order spatial approximations to $g(u)_{xx}|_{x=x_i}$ and $f(u)_x|_{x=x_i}$ that the right-hand-side terms in (2.2) and (2.26) carry out. Since the numerical flux $\hat{b}_{i+\frac{1}{2}}$ approximates $g(u)_x|_{x=x_{i+\frac{1}{2}}}$ and the numerical flux $\hat{f}_{i+\frac{1}{2}}$ approximates $f(u)|_{x=x_{i+\frac{1}{2}}}$, the high-order derivative terms have odd orders in $\hat{b}_{i+\frac{1}{2}}$ and even orders in $\hat{f}_{i+\frac{1}{2}}$ respectively. However, all of these high-order derivative terms are directly approximated by the corresponding central difference schemes with low computational costs as suggested in [54,28].

Remark 2.2. Both the standard finite difference WENO schemes (e.g. [30,41]) and the finite difference A-WENO schemes (e.g. [29, 28]) were developed for efficiently solving multidimensional hyperbolic or convection-diffusion problems on uniform or structured Cartesian grids. The standard finite difference WENO schemes use the reconstruction to compute a numerical flux. The idea of WENO reconstruction for finite difference schemes is similar to that in finite volume schemes. Namely, numerical values of flux function on grid points are "identified" as cell averages of an approximation polynomial, and these cell averages are used to reconstruct the needed polynomial function and obtain high-order WENO approximations to numerical fluxes at the cell interfaces. Since the reconstruction-based finite difference WENO schemes directly approximate the fluxes in the hyperbolic PDEs rather than the conservative variables, a robust flux-splitting approach is needed to maintain the upwinding and linear stability of the schemes. The flux-splitting approach has restrictions on the monotone numerical flux used in the scheme. For example, it requires the monotone numerical flux to be smooth enough for achieving the desired high-order accuracy, and not every monotone numerical flux can be split as needed. However, the finite difference A-WENO methods use the WENO interpolations, rather than the WENO reconstructions, to compute numerical fluxes. For hyperbolic terms, interpolation polynomials are constructed based on numerical values of the conservative variables on grid points and the corresponding WENO interpolations are computed at the interface points to supply the values $u^-_{i+\frac{1}{2}}$ and $u^+_{i+\frac{1}{2}}$ for a monotone numerical flux $h(\cdot,\cdot)$. For example, see (2.27) in the discretization of the convection term for the nonlinear degenerate parabolic equations. Any monotone numerical flux $h(\cdot,\cdot)$ can be used in the finite difference A-WENO schemes, which is more flexible

than the standard reconstruction-based finite difference WENO schemes. For the discretization of the diffusion term in the nonlinear degenerate parabolic equations, there is no need to use a monotone numerical flux. The reconstruction-based finite difference WENO schemes in [41] use the standard equal-sized substencils to compute the WENO reconstructions of the numerical fluxes for the diffusion term. However, for this kind of equations, the standard WENO procedure of combining the low-order reconstructions on the smaller substencils to provide the high-order reconstruction on the big stencil generates negative linear weights, which require a special splitting technique to maintain the stability of the schemes. Moreover, the error analysis in [41] shows that these WENO reconstructions lose accuracy and a mapped function has to be applied to the nonlinear weights for achieving the designed high-order accuracy of these WENO schemes. These issues make the algorithm of the reconstruction-based finite difference WENO schemes in [41] relatively complicated. The multi-resolution A-WENO schemes [28] adopted for the spatial discretization in this paper, use a series of unequal-sized hierarchical central substencils to perform high-order WENO interpolations for computing numerical fluxes. In general the linear weights can be taken as any positive numbers with the only condition that their sum equals one. The analysis in [28] shows that the nonlinear weights satisfy the required condition to ensure the designed high-order accuracy for the A-WENO schemes. Hence the multi-resolution A-WENO schemes resolve those issues aforementioned in the standard WENO schemes for solving the nonlinear degenerate parabolic equations. The ETD-RK A-WENO methods developed in this paper, can efficiently resolve the stiffness of the degenerate parabolic equations and are more efficient than the SSP-RK A-WENO methods in [28] as shown in the next numerical experiments, while the SSP-RK A-WENO methods [28] have similar performance to the standard WENO methods [41] in the accuracy and efficiency for solving problems with a smooth solution. In our numerical experiments, it is interesting to find that for stiff problem with a non-smooth solution, e.g. the PME problem with $m = 8$ in Example 3 of Section 3, the scheme with the multi-resolution A-WENO spatial discretization permits much larger time-step sizes than the scheme with the standard WENO spatial discretization [41], to achieve nonlinear stability and non-oscillatory numerical solution. This verifies that the multi-resolution A-WENO spatial discretization is more robust in solving a problem with non-smooth solution, and it is consistent with the advantage of the multi-resolution WENO schemes [67,37] over the standard WENO schemes [30] in solving hyperbolic conservation laws and their steady-state problems.

2.2. ETD temporal discretization

2.2.1. A new semilinearization approach

The overall nonlinear ODE system resulting from the multi-resolution A-WENO spatial discretization for the nonlinear degenerate parabolic equation (1.1) (or (1.2)) is denoted by

$$\mathbf{u}_t = G(\mathbf{u}) + F(\mathbf{u}) := L(\mathbf{u}), \tag{2.43}$$

where $G(\mathbf{u})$ and $F(\mathbf{u})$ correspond to the discretizations for the diffusion and convection terms of the equations, respectively. Suitable time discretization approaches for the ODE system (2.43), which is stiff due to $G(\mathbf{u})$ from the diffusion terms, are desired to obtain efficient and accurate approximation to the solution. Implicit schemes are often used to solve such kind of stiff systems and achieve large time-step size computations. Challenges in designing implicit methods include developing efficient iterative solvers for nonlinear algebraic systems, computing the exact Jacobian matrices of nonlinear terms which are often highly complex for nonlinear schemes (e.g., WENO methods [21]), etc.

As an alternative approach for solving stiff systems efficiently, the exponential integrator methods are originally designed to solve the semilinear ODEs with stiff linear part and non-stiff nonlinear part:

$$\mathbf{u}_t = C\mathbf{u} + N(\mathbf{u}), \tag{2.44}$$

where C is a constant matrix and $N(\mathbf{u})$ is a nonlinear vector function of $\mathbf{u}$. The idea behind the exponential integrator methods is to multiply (2.44) by the integrating factor e^{-Ct} to absorb the linear stiff term:

$$\frac{d}{dt}(e^{-Ct}\mathbf{u}) = e^{-Ct}N(\mathbf{u}). \tag{2.45}$$

Consequently, the system can be transformed into the exact integral formula,

$$\mathbf{u}^{n+1} = e^{C\Delta t}\mathbf{u}^n + \int_0^{\Delta t} e^{(\Delta t-\tau)C}N(\mathbf{u}(t^n+\tau))d\tau, \tag{2.46}$$

after being integrated over one time interval $[t^n, t^{n+1}]$ where t^n and t^{n+1} are the time steps, and $\Delta t = t^{n+1} - t^n$. $\mathbf{u}^{n+1}$ and $\mathbf{u}^n$ are the numerical solutions of $\mathbf{u}$ at t^{n+1} and t^n respectively. Different numerical discretization strategies for the integral in (2.46) give rise to various methods in the family of the exponential integrators, e.g., the implicit integration factor methods, ETD multi-step methods, ETD-RK methods, etc. The severe time-step size restriction imposed by the stiff linear part is removed, as this stiff component of the system is integrated exactly here.

For a fully nonlinear stiff ODE system with the general form $\mathbf{u}_t = L(\mathbf{u})$, the exponential Rosenborg-type method takes a linearization at every time step to yield the equivalent semilinear reformulation:

$$\mathbf{u}_t = C^n\mathbf{u} + N^n(\mathbf{u}), \quad t \in [t^n, t^{n+1}], \tag{2.47}$$

where $C^n := L'(\mathbf{u}^n)$ is the Jacobian matrix of $L(\mathbf{u})$ and $N^n(u) := L(\mathbf{u}) - C^n\mathbf{u}$ is the nonlinear remainder. Then the exponential integrators are ready to be applied in solving (2.47) and remove the stiffness from the linear part of the system.

Note that in the stiff ODE system (2.43), since the stiffness arises from the parabolic term, it is reasonable to compute the Jacobian matrix C^n only based on $G(\mathbf{u})$ in the semilinear reformulation (2.47). However, even without the complex computation of the Jacobian matrix from the hyperbolic term $F(\mathbf{u})$, due to the high nonlinearity of high-order WENO discretizations for the diffusion term, it is still very difficult and expensive to compute the exact Jacobian matrix of $G(\mathbf{u})$ to obtain C^n in (2.47). The feasible approach we propose here to resolve this difficulty is based on the key observation that the stiffness of the system indeed comes from the non-degenerate region of the diffusion terms, where the solution is smooth and approximations by the nonlinear WENO discretizations are very close to these by the corresponding linear schemes. Therefore, we adopt the Jacobian matrix of the spatial discretization of the linear scheme (2.20) to formulate the semilinear system (2.47), i.e., $C^n = G_L'(\mathbf{u}^n)$ and $N^n(\mathbf{u}) = G(\mathbf{u}) + F(\mathbf{u}) - C^n\mathbf{u}$, where $G_L(\mathbf{u})$ is the spatial discretization for the diffusion terms in the linear scheme (2.20). More specifically, for the problems with one space dimension, we have

- $r = 1$,

$$C_{i\cdot}^n = \frac{1}{\Delta x^2}(0,\dots,0,g'_{i-1},-2g'_i,g'_{i+1},0,\dots,0), \tag{2.48}$$

- $r = 2$,

$$C_{i\cdot}^n = \frac{1}{12\Delta x^2}(0,\dots,0,-g'_{i-2},16g'_{i-1},-30g'_i,16g'_{i+1},-g'_{i+2},0,\dots,0), \tag{2.49}$$

- $r = 3$,

$$\begin{aligned} C_{i\cdot}^n = &\frac{1}{180\Delta x^2}(0,\dots,0,2g'_{i-3},-27g'_{i-2},270g'_{i-1},-490g'_i,\\ &270g'_{i+1},-27g'_{i+2},2g'_{i+3},0,\dots,0), \end{aligned} \tag{2.50}$$

- $r = 4$,

$$\begin{aligned} C_{i\cdot}^n = &\frac{1}{5040\Delta x^2}(0,\dots,0,-9g'_{i-4},128g'_{i-3},-1008g'_{i-2},8064g'_{i-1},-14350g'_i,\\ &8064g'_{i+1},-1008g'_{i+2},128g'_{i+3},-9g'_{i+4},0,\dots,0), \end{aligned} \tag{2.51}$$

where $C_{i\cdot}^n$ denotes the i-th row of the matrix C^n, $g'_k = g'(u_k)$, and the lower ellipsis stands for zero components of the vectors. Likewise, the Jacobian matrix of the spatial discretization of the linear scheme for the problems with two space dimensions is computed in a similar manner.

In the consequent subsections, we present an efficient and accurate class of explicit exponential integrators, the ETD Runge-Kutta schemes [17], for solving (2.47) using the new semilinearization approach proposed above. For conciseness, we drop the superscript n.

2.2.2. ETD-RK methods

It is important to note that the matrices involved in the algorithms are highly sparse. Therefore, the data structure used for storing and computing these matrices is specialized for sparse matrices in our implementation.

The first-order ETD scheme is derived through approximation to the nonlinear part $N(\mathbf{u})$ in the integrand (2.46) by the constant quantity $N(\mathbf{u}^n)$, and a direct evaluation of the integral gives

$$\begin{aligned} \mathbf{u}^{n+1} =& e^{\Delta t C}\mathbf{u}^n + \Delta t\varphi_1(\Delta t C)N(\mathbf{u}^n)\\ =& \mathbf{u}^n + \Delta t\varphi_1(\Delta t C)(C\mathbf{u}^n + N(\mathbf{u}^n)), \end{aligned} \tag{2.52}$$

where $\varphi_1(z) := \frac{e^z-1}{z}$ is a member of the family of φ-functions (see e.g. [25]). Analogous to the classic RK methods, the high-order ETD-RK schemes were derived in [17]. Also see e.g. [38] for their formulations in terms of the φ-functions. In the following, we show the third-order and the fourth-order ETD-RK methods used for solving the system (2.47) in this paper:

- ETD-RK3

$$\begin{aligned} \mathbf{a}^n =& \mathbf{u}^n + \frac{\Delta t}{2}\varphi_1(\frac{\Delta t}{2}C)(C\mathbf{u}^n + N(\mathbf{u}^n)),\\ \mathbf{b}^n =& \mathbf{u}^n + \Delta t\varphi_1(\Delta t C)(C\mathbf{u}^n - N(\mathbf{u}^n) + 2N(\mathbf{a}^n)),\\ \mathbf{u}^{n+1} =& \mathbf{u}^n + \Delta t\varphi_1(\Delta t C)(C\mathbf{u}^n + N(\mathbf{u}^n))\\ &+ \Delta t\varphi_2(\Delta t C)(-3N(\mathbf{u}^n) + 4N(\mathbf{a}^n) - N(\mathbf{b}^n))\\ &+ \Delta t\varphi_3(\Delta t C)(4N(\mathbf{u}^n) - 8N(\mathbf{a}^n) + 4N(\mathbf{b}^n)), \end{aligned} \tag{2.53}$$

- ETD-RK4

$$\begin{aligned}
\mathbf{a}^n &= \mathbf{u}^n + \frac{\Delta t}{2}\varphi_1(\frac{\Delta t}{2}C)(C\mathbf{u}^n + N(\mathbf{u}^n)),\\
\mathbf{b}^n &= \mathbf{u}^n + \frac{\Delta t}{2}\varphi_1(\frac{\Delta t}{2}C)(C\mathbf{u}^n + N(\mathbf{a}^n)),\\
\mathbf{c}^n &= \mathbf{a}^n + \frac{\Delta t}{2}\varphi_1(\frac{\Delta t}{2}C)(C\mathbf{a}^n - N(\mathbf{u}^n) + 2N(\mathbf{b}^n)),\\
\mathbf{u}^{n+1} &= \mathbf{u}^n + \Delta t\varphi_1(\Delta tC)(C\mathbf{u}^n + N(\mathbf{u}^n))\\
&\quad + \Delta t\varphi_2(\Delta tC)(-3N(\mathbf{u}^n) + 2N(\mathbf{a}^n) + 2N(\mathbf{b}^n) - N(\mathbf{c}^n))\\
&\quad + \Delta t\varphi_3(\Delta tC)(4N(\mathbf{u}^n) - 4N(\mathbf{a}^n) - 4N(\mathbf{b}^n) + 4N(\mathbf{c}^n)),
\end{aligned} \tag{2.54}$$

where the φ-functions in (2.53)-(2.54) are defined by the recurrence relation $\varphi_\ell(z) = z\varphi_{\ell+1}(z) + \frac{1}{\ell!}, \ell = 0, 1, \ldots$ [25]. For example, for small values of ℓ,

$$\varphi_0 = e^z, \quad \varphi_1(z) = \frac{e^z - 1}{z}, \quad \varphi_2(z) = \frac{e^z - 1 - z}{z^2}, \quad \varphi_3(z) = \frac{e^z - 1 - z - \frac{1}{2}z^2}{z^3}. \tag{2.55}$$

It is clear that the semidiscrete scheme (2.43) is conservative, as the convection and diffusion terms in the problem (1.1) (or (1.2)) are both discretized in conservation form. More importantly, the fully discretized ETD-RK schemes (2.52) - (2.54) are also conservative, which is stated in the following theorem.

Theorem 2.1. *The ETD-RK schemes* (2.52) - (2.54) *are conservative in the sense that* $\sum_i u_i^n = \sum_i u_i^{n+1}$ *under periodic or compactly supported boundary conditions.*

Proof. Let $\mathbf{1}$ denote the vector of all ones with the same size as $\mathbf{u}^n$. Since the semidiscrete WENO discretization (2.19) for the diffusion term and (2.41) for the convection term, as well as the linear discretization (2.20) for the diffusion term, are all conservative, we have:

$$\mathbf{1}^T G(\mathbf{u}) = \mathbf{1}^T F(\mathbf{u}) = \mathbf{1}^T G_L(\mathbf{u}) = 0.$$

Consequently, $\mathbf{1}^T C = \mathbf{0}^T$ and $\mathbf{1}^T N(\mathbf{u}) = 0$ because:

$$\mathbf{1}^T C = \mathbf{1}^T G_L'(\mathbf{u}^n) = \frac{\partial(\mathbf{1}^T G_L(\mathbf{u}^n))}{\partial \mathbf{u}} = \frac{\partial 0}{\partial \mathbf{u}} = \mathbf{0}^T,$$

and

$$\mathbf{1}^T N(\mathbf{u}) = \mathbf{1}^T G(\mathbf{u}) + \mathbf{1}^T F(\mathbf{u}) - \mathbf{1}^T C\mathbf{u} = 0.$$

Moreover, we have $\mathbf{1}^T \varphi_k(C\Delta t) = \frac{1}{k!}\mathbf{1}^T$, which follows from the recurrence relation,

$$\mathbf{1}^T \varphi_k(C\Delta t) = \mathbf{1}^T C\Delta t\varphi_{k+1}(C\Delta t) + \frac{1}{k!}\mathbf{1}^T = \frac{1}{k!}\mathbf{1}^T.$$

Combining all the results above, it is straightforward to verify that $\mathbf{1}^T\mathbf{u}^{n+1} = \mathbf{1}^T\mathbf{u}^n$ for the schemes (2.52) - (2.54). □

Remark 2.3. The current scheme does not guarantee the preservation of positivity in the solution. In [18], first- and second-order exponential time-differencing approaches were developed to satisfy the maximum principle for semilinear parabolic equations of the form $u_t = \Delta u + f(u)$. However, it remains an open question how to enforce the maximum principle for degenerate parabolic equations in high-order ETD-RK methods. In numerical tests, we observed negative values at certain time steps, on specific grids, or for certain index values of m in the porous media equation. That said, the ETD-RK A-WENO algorithm is robust enough to avoid blow-up even when negative values (anti-diffusion) appear.

Remark 2.4. The design of the linear operator term here for the ETD-RK schemes is based on balancing efficiency (in terms of derivation, coding, and computation) and effectiveness (in terms of achieving large time-step sizes). Note that accuracy is not a concern for any choice of splitting, as the integration of the nonlinear residual is high-order accurate. Obtaining the Jacobian of the corresponding linear operator term is significantly simpler and cheaper in computational cost than that of the WENO operator itself. The ETD schemes are considered effective if they resolve the stiffness of the ODE system, thus allowing for a large time-step size. The motivation for using the linear operator lies in the fact that the WENO discretization closely approximates the linear operator in non-degenerate regions, which are believed to be the source of stiffness in the degenerate parabolic equations. Indeed, as validated by extensive numerical tests in Section 3, our choice of the linear operator yields satisfactory results, achieving the expected large time-step sizes and computational efficiency.

2.2.3. Fast computation of φ-functions in the implementation

Notice that every stage in the ETD-RK schemes (2.53)-(2.54) is a linear combination of φ-functions of matrices acting on a set of vectors,

$$\varphi_0(A)\mathbf{v}_0 + \varphi_1(A)\mathbf{v}_1 + \cdots + \varphi_p(A)\mathbf{v}_p, \tag{2.56}$$

where A is an $\mathcal{N}$-by-$\mathcal{N}$ sparse matrix, $\mathbf{v}_k \in \mathbb{R}^{\mathcal{N}}, k = 0, \ldots, p$, and $\mathcal{N}$ is the size of the system (2.47). Hence fast computation of φ-functions is crucial for the efficient implementation of these ETD methods. In this paper, the algorithm *phipm* developed in [50], which is based on the Krylov subspace method for matrix exponentials (see e.g. [46]), is adopted for the computation of (2.56). One may also conduct the computation based on its modified versions, e.g. *phipm-simul-iom2* in [44]. For completeness, a brief description of the algorithm is given below.

The Krylov subspace with a dimension m ($m \ll \mathcal{N}$) for the matrix-vector pair $A \in \mathbb{R}^{\mathcal{N}\times\mathcal{N}}, \mathbf{v} \in \mathbb{R}^{\mathcal{N}}$ is defined as

$$K_m = \text{span}\{\mathbf{v}, A\mathbf{v}, \ldots, A^{m-1}\mathbf{v}\}.$$

The Arnoldi iteration [58], a stabilized Gram-Schmidt process, is employed to obtain an orthonormal basis of K_m. We denote the basis by $\{\tilde{\mathbf{v}}_1, \tilde{\mathbf{v}}_2, \ldots, \tilde{\mathbf{v}}_m\}$ and let $V_m = [\tilde{\mathbf{v}}_1, \tilde{\mathbf{v}}_2, \ldots, \tilde{\mathbf{v}}_m] \in \mathbb{R}^{\mathcal{N}\times m}$, then the Hessenberg matrix $H_m = V_m^T A V_m \in \mathbb{R}^{m\times m}$ obtained as one of the products of the Arnoldi iteration is the matrix representation of A in the Krylov subspace K_m with the basis $\{\tilde{\mathbf{v}}_1, \ldots, \tilde{\mathbf{v}}_m\}$. Note that $V_m H_m V_m^T$ is the matrix representation of A in the Krylov subspace K_m with the standard basis of $\mathbb{R}^{\mathcal{N}}$, so one can approximate $\varphi_k(A)\mathbf{v}$ in the Krylov subspace K_m as

$$\varphi_k(A)\mathbf{v} \approx \varphi_k(V_m H_m V_m^T)\mathbf{v} = V_m \varphi_k(H_m) V_m^T \mathbf{v} = ||\mathbf{v}|| V_m \varphi_k(H_m)\mathbf{e}_1,$$

where $||\cdot||$ is the standard Euclidean norm and $\mathbf{e}_1$ is the first standard basis in $\mathbb{R}^m$. The Krylov subspace method reduces the computation of $\varphi_k(A)\mathbf{v}$ down to that of $\varphi_k(H_m)\mathbf{e}_1$, whose computation is discussed in [50] and the references therein.

As that pointed out in [50], the linear combination (2.56) is actually the solution of the initial value problem of the ordinary differential equation

$$\mathbf{y}_t = A\mathbf{y} + \mathbf{v}_1 + t\mathbf{v}_2 + \cdots + \frac{t^{p-1}}{(p-1)!}\mathbf{v}_p, \quad \mathbf{y}(0) = \mathbf{v}_0, \tag{2.57}$$

at $t = 1$. A time-stepping method built upon efficient computation of φ-function of matrices by the Krylov subspace method is used to compute $\mathbf{y}(1)$. As the two critical parameters affecting the computational cost, the dimension m of the Krylov subspace K_m and the time-step size τ in the time-stepping method for (2.57) are determined adaptively during computations to optimize the performance. See [50] for details of the method.

2.2.4. Other time marching approaches for comparison

To show the efficiency of the ETD schemes with the proposed techniques for solving the complex ODE systems resulting from the multi-resolution A-WENO spatial discretization, we compare their computational costs and accuracy with some commonly used explicit and implicit strong stability preserving Runge-Kutta (SSP-RK) methods. Specifically, the following three-stage third order explicit Runge-Kutta (SSP-ERK3) and three-stage third order diagonally implicit Runge-Kutta (SSP-IRK3) schemes are employed for the comparison. Their forms for solving the ODE system $\mathbf{u}_t = L(\mathbf{u})$ are

- SSP-ERK3:

$$\begin{aligned}
\mathbf{u}^{(1)} &= \mathbf{u}^n + \Delta t L(\mathbf{u}^n),\\
\mathbf{u}^{(2)} &= \frac{3}{4}\mathbf{u}^n + \frac{1}{4}(\mathbf{u}^{(1)} + \Delta t L(\mathbf{u}^{(1)})),\\
\mathbf{u}^{n+1} &= \frac{1}{3}\mathbf{u}^n + \frac{2}{3}(\mathbf{u}^{(2)} + \Delta t L(\mathbf{u}^{(2)})),
\end{aligned} \tag{2.58}$$

- SSP-IRK3:

$$\begin{aligned}
\mathbf{u}^{(1)} &= \mathbf{u}^n + \Delta t(a_{11} L(\mathbf{u}^{(1)})),\\
\mathbf{u}^{(2)} &= \mathbf{u}^n + \Delta t(a_{21} L(\mathbf{u}^{(1)}) + a_{22} L(\mathbf{u}^{(2)})),\\
\mathbf{u}^{(3)} &= \mathbf{u}^n + \Delta t(a_{31} L(\mathbf{u}^{(1)}) + a_{32} L(\mathbf{u}^{(2)}) + a_{33} L(\mathbf{u}^{(3)})),\\
\mathbf{u}^{n+1} &= \mathbf{u}^n + \Delta t(a_{41} L(\mathbf{u}^{(1)}) + a_{42} L(\mathbf{u}^{(2)}) + a_{43} L(\mathbf{u}^{(3)})),
\end{aligned} \tag{2.59}$$

where the coefficients are given as

$$\begin{aligned}
&a_{11} = 0.1464466094067262,\\
&a_{21} = 0.3535533905932738, a_{22} = 0.1464466094067262,\\
&a_{31} = 0.3535533905932738, a_{32} = 0.3535533905932738, a_{33} = 0.1464466094067262,\\
&a_{41} = 0.33333333333333333, a_{42} = 0.33333333333333333, a_{43} = 0.33333333333333333.
\end{aligned}$$

As discussed in the previous sections, the high complexity and costs in the calculation of Jacobian matrices for high-order WENO spatial discretizations lead to significant challenges in applying a fully implicit time-stepping method, which is also a motivation for us to design a new semilinearization technique for the exponential integrators in this paper. To simplify the implementation in applying the SSP-IRK3 method for the numerical comparison studies here, we couple the SSP-IRK3 method with the linear spatial discretization scheme (2.20). The Newton iteration is used to solve the nonlinear algebraic systems at every time step of the SSP-IRK3 scheme. Note that appropriate preconditioning techniques could speed up the computation of an implicit method such as the SSP-IRK3 method. Here, we do not further explore more sophisticated nonlinear system solvers for the implicit methods, but leave this interesting topic in our future work. For more detailed introduction of these SSP-RK schemes, we refer the readers to the monograph [20]. In the following section, the numerical results obtained from these different time-stepping approaches are presented to compare their efficiency and accuracy.

3. Numerical experiments

This section is devoted to the numerical tests of the ETD-RK multi-resolution A-WENO schemes developed in the previous sections. All numerical examples, except for Example 2, are adopted from the published benchmarks that are widely used in the literature, e.g. [34,41,4,28,65,60].

We present the examples in order of increasing complexity and compare the results of the ETD-RK methods with those obtained using the explicit SSP-RK (SSP-ERK3) and the diagonally implicit SSP-RK (SSP-IRK3) methods. For the ETD-RK and the implicit SSP-RK methods, we adopt the time-step size $\Delta t = \text{CFL} \times \Delta x$ for one-dimensional problems and $\Delta t = \text{CFL} \times \min\{\Delta x, \Delta y\}$ for two-dimensional problems, respectively, where CFL is a constant that may take different values in different examples. For the SSP-ERK3 method, we follow the CFL stability condition of explicit methods and take $\Delta t = \frac{0.4}{c/\Delta x + b/\Delta x^2}$ for one-dimensional problems, where $c = \max_u |f'(u)|$ and $b = \max_u |g'(u)|$; and $\Delta t = \frac{0.4}{c_x/\Delta x + c_y/\Delta y + b/\Delta x^2 + b/\Delta y^2}$ for two-dimensional problems, where $c_x = \max_u |f_1'(u)|$, $c_y = \max_u |f_2'(u)|$ and $b = \max_u |g'(u)|$. The CFL conditions for three-dimensional problems follow the same pattern. All computations in this paper are conducted using Matlab R2023b on an Apple M2 Pro chip 12-core CPU with 16 GB of RAM. In the following discussions of numerical results, N, M and L denote the number of grid points in the $x-$, $y-$ and $z-$ directions of the computational grid respectively.

Example 1. Heat equations

In this test, we solve the heat equation

$$u_t = \Delta u, \tag{3.1}$$

on domains $\Omega = [-\pi, \pi]^d$ with periodic boundary conditions, where $d = 1, 2, 3$ are the dimensions of space. The exact solutions of the problem are given as $u(x,t) = e^{-t}\sin(x)$, $u(x,y,t) = e^{-2t}\sin(x+y)$ and $u(x,y,z,t) = e^{-3t}\sin(x+y+z)$ for $d = 1, 2, 3$, respectively.

Using the ETD-RK multi-resolution A-WENO methods of different temporal and spatial orders, we compute the solutions up to $T = 1$ with the time-step size $\Delta t = \Delta x$. The numerical errors and orders of convergence are presented in Table 1, 2, 3 for $d = 1, 2, 3$, respectively. The designed accuracy orders of the spatial discretizations, instead of the temporal discretizations, are observed in the tables. The results verify that for pure linear diffusion problems with a smooth solution, the numerical errors of the temporal discretizations using these exponential integrators for the diffusion term are very small.

In addition, we compare the computational efficiency of the ETD-RK3 scheme with the SSP-ERK3 scheme and the SSP-IRK3 scheme, coupled with the fourth-order spatial discretizations (i.e., the $r = 2$ case in the section 2.1.1). Specifically, the fourth-order linear spatial discretization is used for the SSP-IRK3 scheme as that discussed in the section 2.2.4. For $d = 1$ and 2, the computation is conducted on the grids with $N = 20, 40, 60, \ldots, 140$ cells in each dimension. On all grids, we take $\Delta t = \Delta x$ for the SSP-IRK3 scheme as that for the ETD-RK3 scheme. In the ETD-RK3 scheme, both the A-WENO and the linear spatial discretizations are applied for the comparison. The L^1 errors versus the CPU times for different methods are shown in Fig. 3.1. In the figure, we observe the superiority of the efficiency of the ETD-RK method compared to both the explicit and the implicit SSP-RK methods. It takes less CPU time costs for the ETD-RK3 scheme than the other schemes to achieve a similar level of small numerical errors on refined meshes. For $d = 3$, the computation is conducted on the grids with $N = 20, 40, 60, 80$ cells in each direction. The L^1 accuracy and CPU times for SSP-ERK3 and ETD-RK3 are presented in Table 4. From the table, we observe superiority of the efficiency of the ETD-RK3 method compared to SSP-ERK3. The SSP-IRK3 time-stepping method is not included in the 3D test because it requires significantly more computational time than the other methods, primarily due to the costly Newton iterations. To make a fair comparison of implicit methods, appropriate preconditioning or other specifically designed acceleration techniques would be necessary. Since implicit methods are not the focus of this paper, we have chosen not to present such results. We also performed computations using the WENO spatial discretization from [41]. The results indicate that both the WENO scheme from [41] and the multi-resolution A-WENO scheme adopted in this paper demonstrate very close accuracy and efficiency for smooth problems when used with the SSP-ERK3 method. Therefore, we do not present these results here to save space.

Table 1
Example 1. Numerical errors of the ETD-RK multi-resolution A-WENO methods for the heat equation with the time-step size $\Delta t = \Delta x$ in one-dimensional space. MRWENO2r stands for the 2r-th order multi-resolution A-WENO discretization in space.

ETD-RK3						
	MRWENO4		MRWENO6		MRWENO8	
N	L^1 Error	Order	L^1 Error	Order	L^1 Error	Order
20	1.58×10^{-4}	-	2.48×10^{-6}	-	4.31×10^{-8}	-
40	9.92×10^{-6}	3.99	3.93×10^{-8}	5.98	1.72×10^{-10}	7.97
60	1.96×10^{-6}	4.00	3.46×10^{-9}	5.99	6.74×10^{-12}	7.99
80	6.22×10^{-7}	4.00	6.16×10^{-10}	6.00	6.60×10^{-13}	8.07
100	2.55×10^{-7}	4.00	1.62×10^{-10}	6.00	1.16×10^{-13}	7.80
ETD-RK4						
	MRWENO4		MRWENO6		MRWENO8	
N	L^1 Error	Order	L^1 Error	Order	L^1 Error	Order
20	1.58×10^{-4}	-	2.48×10^{-6}	-	4.31×10^{-8}	-
40	9.92×10^{-6}	3.99	3.93×10^{-8}	5.98	1.72×10^{-10}	7.97
60	1.96×10^{-6}	4.00	3.46×10^{-9}	5.99	6.74×10^{-12}	7.99
80	6.22×10^{-7}	4.00	6.16×10^{-10}	6.00	6.60×10^{-13}	8.08
100	2.55×10^{-7}	4.00	1.62×10^{-10}	6.00	1.15×10^{-13}	7.84

Table 2
Example 1. Numerical errors of the ETD-RK multi-resolution A-WENO methods for the heat equation with the time-step size $\Delta t = \Delta x$ in two-dimensional space. MRWENO2r stands for the 2r-th order multi-resolution A-WENO discretization in space.

ETD-RK3						
	MRWENO4		MRWENO6		MRWENO8	
$N\times M$	L^1 Error	Order	L^1 Error	Order	L^1 Error	Order
20×20	7.28×10^{-4}	-	1.14×10^{-5}	-	1.99×10^{-7}	-
40×40	4.59×10^{-5}	3.99	1.81×10^{-7}	5.98	7.94×10^{-10}	7.97
60×60	9.08×10^{-6}	4.00	1.60×10^{-8}	5.99	3.11×10^{-11}	7.99
80×80	2.87×10^{-6}	4.00	2.85×10^{-9}	6.00	3.05×10^{-12}	8.08
100×100	1.18×10^{-6}	4.00	7.47×10^{-10}	6.00	5.27×10^{-13}	7.86
ETD-RK4						
	MRWENO4		MRWENO6		MRWENO8	
$N\times M$	L^1 Error	Order	L^1 Error	Order	L^1 Error	Order
20×20	7.28×10^{-4}	-	1.14×10^{-5}	-	1.99×10^{-7}	-
40×40	4.59×10^{-5}	3.99	1.81×10^{-7}	5.98	7.94×10^{-10}	7.97
60×60	9.08×10^{-6}	4.00	1.60×10^{-8}	5.99	3.11×10^{-11}	7.99
80×80	2.87×10^{-6}	4.00	2.85×10^{-9}	6.00	3.05×10^{-12}	8.08
100×100	1.18×10^{-6}	4.00	7.47×10^{-10}	6.00	5.28×10^{-13}	7.86

Example 2. Nonlinear stiff reaction-diffusion equations

In this test, we solve the nonlinear stiff reaction-diffusion equation

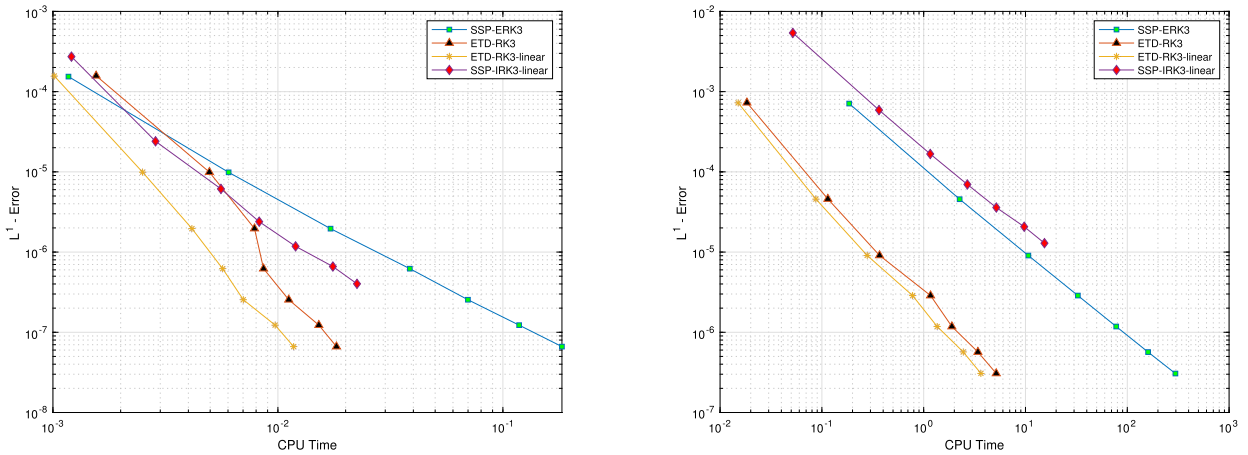
$$u_t = 128\Delta u^8 + R_d(u), \tag{3.2}$$

on domains $\Omega = [-\pi,\pi]^d$ with periodic boundary conditions, where $d = 1,2,3$ are the dimensions of space. The source terms and exact solutions of the problem are given as $R_1(u) = \frac{1}{1024u^7} - \frac{u}{8} + 128u^8 - 1$, $u(x,t) = \frac{1}{2}(e^{-t}\sin(x)+2)^{\frac{1}{8}}$, $R_2(u) = -\frac{u}{4} + 256u^8 + \frac{1}{512u^7} - 2$, $u(x,y,t) = \frac{1}{2}(e^{-2t}\sin(x+y)+2)^{\frac{1}{8}}$ and $R_3(u) = -\frac{u}{4} + \frac{1}{512u^7} + 384u^8 - 3$, $u(x,y,z,t) = \frac{1}{2}(e^{-2t}\sin(x+y+z)+2)^{\frac{1}{8}}$, for $d = 1,2$ and 3, respectively.

Using the ETD-RK multi-resolution A-WENO methods of different temporal and spatial orders, we compute the solutions with the time-step size $\Delta t = 0.01\Delta x$. Note that due to these complex nonlinear stiff diffusion and reaction terms, a smaller CFL number is required in this example than Example 1. However, time-step size $\Delta t \sim O(\Delta x)$ can still be preserved in the mesh refinement study. The solutions are computed up to $T = 1, 0.2$ and 0.05 for $d = 1,2$ and 3, respectively. The numerical errors and orders of convergence are presented in Tables 5 - 7. We observe that for the fourth-order multi-resolution A-WENO scheme coupled with either the ETD-RK3 or the ETD-RK4 temporal discretizations, the numerical errors of the spatial discretization show strong influence and a fourth-order / close to fourth-order convergence rate is obtained. However, for the sixth-order and the eighth-order multi-resolution A-WENO schemes coupled with either the ETD-RK3 or the ETD-RK4 temporal discretizations, the numerical errors of the

Table 3
Example 1. Numerical errors of the ETD-RK multi-resolution A-WENO methods for the heat equation with the time-step size $\Delta t = \Delta x$ in three-dimensional space. MRWENO2r stands for the 2r-th order multi-resolution A-WENO discretization in space.

ETD-RK3						
	MRWENO4		MRWENO6		MRWENO8	
$N \times M \times L$	L^1 Error	Order	L^1 Error	Order	L^1 Error	Order
$20 \times 20 \times 20$	2.52×10^{-3}	-	3.96×10^{-5}	-	6.89×10^{-7}	-
$40 \times 40 \times 40$	1.59×10^{-4}	3.99	6.29×10^{-7}	5.98	2.75×10^{-9}	7.97
$60 \times 60 \times 60$	3.15×10^{-5}	4.00	5.54×10^{-8}	5.99	1.08×10^{-10}	7.99
$80 \times 80 \times 80$	9.96×10^{-6}	4.00	9.87×10^{-9}	6.00	1.06×10^{-11}	8.08
$100 \times 100 \times 100$	4.08×10^{-6}	4.00	2.59×10^{-9}	6.00	1.80×10^{-12}	7.92
ETD-RK4						
	MRWENO4		MRWENO6		MRWENO8	
$N \times M \times L$	L^1 Error	Order	L^1 Error	Order	L^1 Error	Order
$20 \times 20 \times 20$	2.52×10^{-3}	-	3.96×10^{-5}	-	6.89×10^{-7}	-
$40 \times 40 \times 40$	1.59×10^{-4}	3.99	6.29×10^{-7}	5.98	2.75×10^{-9}	7.97
$60 \times 60 \times 60$	3.15×10^{-5}	4.00	5.54×10^{-8}	5.99	1.08×10^{-10}	7.99
$80 \times 80 \times 80$	9.96×10^{-6}	4.00	9.87×10^{-9}	6.00	1.06×10^{-11}	8.08
$100 \times 100 \times 100$	4.08×10^{-6}	4.00	2.59×10^{-9}	6.00	1.80×10^{-12}	7.92



(a) $d = 1$. (b) $d = 2$.

Fig. 3.1. Example 1. Comparison of efficiency for different time-stepping methods for the heat equation. SSP-IRK3-linear and ETD-RK3-linear indicate that the linear spatial discretization is used for them. CPU time unit: second.

Table 4
Example 1. Comparison of efficiency for SSP-ERK3 and ETD-RK3 methods for the three-dimensional heat equation. The fourth-order multi-resolution A-WENO spatial discretization is adopted. The time-step size $\Delta t = \Delta x$ is used in the ETD-RK3 method.

	SSP-ERK3		ETD-RK3	
$N \times M \times L$	L^1 Error	CPU (s)	L^1 Error	CPU (s)
$20 \times 20 \times 20$	2.50×10^{-3}	5.93×10^{0}	2.52×10^{-3}	5.51×10^{-1}
$40 \times 40 \times 40$	1.59×10^{-4}	1.61×10^{2}	1.59×10^{-4}	6.14×10^{0}
$60 \times 60 \times 60$	3.14×10^{-5}	1.17×10^{3}	3.15×10^{-5}	2.64×10^{1}
$80 \times 80 \times 80$	9.96×10^{-6}	4.77×10^{3}	9.96×10^{-6}	7.70×10^{1}

temporal discretizations dominate along with the refinement of the meshes, hence the designed accuracy orders of the temporal discretizations, instead of the spatial discretizations, are observed in the tables. Comparing with Example 1 which is a linear problem with a pure diffusion term, in this example the equation has a highly nonlinear and complex reaction term in addition to a stiff nonlinear diffusion term, and numerical errors of the spatial discretizations and the temporal discretizations from the different terms have a richer structure.

Table 5
Example 2. Numerical errors of the ETD-RK multi-resolution A-WENO methods for the nonlinear stiff reaction-diffusion equation with the time-step size $\Delta t = 0.01\Delta x$ in one-dimensional space. MRWENO2r stands for the 2r-th order multi-resolution A-WENO discretization in space.

ETD-RK3						
	MRWENO4		MRWENO6		MRWENO8	
N	L^1 Error	Order	L^1 Error	Order	L^1 Error	Order
50	1.19×10^{-1}	-	2.78×10^{-3}	-	2.52×10^{-3}	-
100	6.74×10^{-3}	4.15	3.22×10^{-4}	3.11	3.18×10^{-4}	2.99
150	1.36×10^{-3}	3.96	9.47×10^{-5}	3.01	9.44×10^{-5}	2.99
200	4.38×10^{-4}	3.92	3.99×10^{-5}	3.00	3.99×10^{-5}	2.99
250	1.84×10^{-4}	3.90	2.05×10^{-5}	3.00	2.04×10^{-5}	3.00
ETD-RK4						
	MRWENO4		MRWENO6		MRWENO8	
N	L^1 Error	Order	L^1 Error	Order	L^1 Error	Order
50	1.16×10^{-1}	-	2.70×10^{-4}	-	1.25×10^{-5}	-
100	6.42×10^{-3}	4.18	4.78×10^{-6}	5.82	7.43×10^{-7}	4.07
150	1.26×10^{-3}	4.01	5.03×10^{-7}	5.56	1.44×10^{-7}	4.05
200	3.98×10^{-4}	4.00	1.01×10^{-7}	5.56	4.48×10^{-8}	4.05
250	1.63×10^{-4}	4.00	4.19×10^{-8}	3.96	1.91×10^{-8}	3.83

In addition, we compare the computational efficiency of the ETD-RK3 method with the SSP-ERK3 and the SSP-IRK3 methods, coupled with the fourth-order spatial discretization (i.e., the $r = 2$ case in the section 2.1.1) on grids of different levels of refinement. Similar to Example 1, the fourth-order linear spatial discretization is used for the SSP-IRK3 scheme as that discussed in the section 2.2.4, and both the A-WENO and the linear spatial discretizations are applied for the ETD-RK3 scheme in one- and two-dimensional tests. For $d = 1$, the time-step size $\Delta t = 0.02\Delta x$ is taken for the simulations using the ETD-RK3 method on all grids, while for the SSP-IRK3 method, we take $\Delta t = 0.02\Delta x$ for the simulations on the grids with $N = 50, 100$, $\Delta t = 0.01\Delta x$ for the simulations on the grids with $N = 150, 200$, and $\Delta t = 0.005\Delta x$ for the simulations on the other finer grids to ensure the stability in solving this stiff nonlinear problem. The L^1 errors versus the CPU times for different methods are shown in Fig. 3.2 (a), which demonstrates that the ETD-RK3 method is more efficient than both the explicit and the implicit SSP-RK methods here. It takes less CPU time costs for the ETD-RK3 method than the other two methods to achieve a similar level of small numerical errors. For $d = 2$, we take the time-step size $\Delta t = 0.01\Delta x$ in the computations using the ETD-RK3 method on all grids, while for the SSP-IRK3 method, we take $\Delta t = 0.01\Delta x$ in the simulations on the grids with $N = M = 50, 100$, $\Delta t = 0.005\Delta x$ in the simulations on the grids with $N = M = 150, 200$, and $\Delta t = 0.002\Delta x$ in the simulation on the grid with $N = M = 250$ to ensure the stability in solving this 2D stiff nonlinear problem. The L^1 errors versus the CPU times for different methods are shown in Fig. 3.2 (b), which verifies that the ETD-RK3 method is much more efficient than both the explicit and the implicit SSP-RK methods here. It takes much less CPU time costs for the ETD-RK3 method than the other two methods to reach a similar level of numerical errors. For $d = 3$, we compare the efficiency of the ETD-RK3 method with the SSP-ERK3 method. The accuracy and CPU times on different grids are presented in Table 8. The SSP-IRK method requires significantly longer computational times without appropriate preconditioning or fast solvers and is therefore not included in the comparison. Based on the data in the table, a consistent conclusion regarding efficiency, similar to that observed in the one- and two-dimensional problems, is drawn for this three-dimensional case. We also performed computations using the SSP-ERK3 method coupled with the WENO spatial discretization developed in [41]. The results indicate that both the WENO scheme from [41] and the A-WENO scheme adopted in this paper demonstrate very close accuracy and efficiency for the smooth nonlinear problems when used with the SSP-ERK3 method. Therefore, we do not present these results here to save space.

Example 3. PME with the Barenblatt solutions

We solve the PME

$$u_t = \Delta u^m, \quad \mathbf{x} \in \mathbb{R}^d, \tag{3.3}$$

with the Barenblatt solutions

$$B_m(\mathbf{x}, t) = t^{-p}\left(\left(1 - \frac{p(m-1)}{2dm}\frac{|\mathbf{x}|^2}{t^{\frac{2p}{d}}}\right)^+\right)^{\frac{1}{m-1}}, \quad p = \frac{1}{m-1+2/d}. \tag{3.4}$$

Different values of m ($m = 2, 3, 5, 8$) and d ($d = 1, 2, 3$) are taken in this test. The solutions at $t_0 = 1$ are used as the initial conditions and the homogeneous Dirichlet boundary conditions are applied if not otherwise stated.

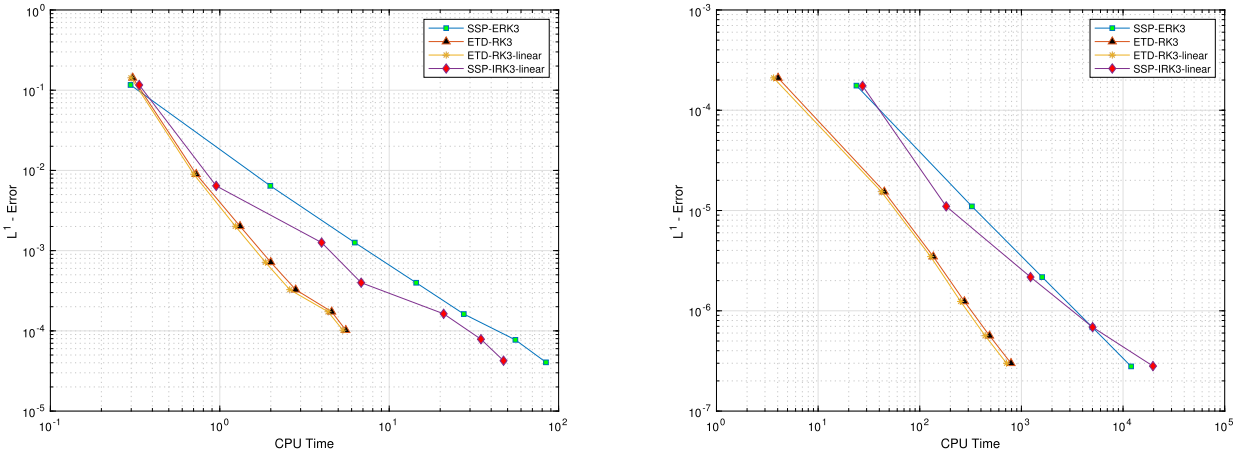
We first compute the solutions up to $T = 2$ using the ETD-RK4 scheme coupled with the sixth-order multi-resolution A-WENO (A-WENO6) spatial discretization. Numerical results with other spatial and temporal order accuracy exhibit a similar pattern and are

Table 6
Example 2. Numerical errors of the ETD-RK multi-resolution A-WENO methods for the nonlinear stiff reaction-diffusion equation with the time-step size $\Delta t = 0.01\Delta x$ in two-dimensional space. MRWENO2r stands for the $2r$-th order multi-resolution A-WENO discretization in space.

ETD-RK3						
	MRWENO4		MRWENO6		MRWENO8	
$N \times M$	L^1 Error	Order	L^1 Error	Order	L^1 Error	Order
50×50	2.10×10^{-4}	-	3.48×10^{-5}	-	3.43×10^{-5}	-
100×100	1.54×10^{-5}	3.77	4.38×10^{-6}	2.99	4.38×10^{-6}	2.97
150×150	3.47×10^{-6}	3.67	1.31×10^{-6}	2.99	1.30×10^{-6}	2.98
200×200	1.24×10^{-6}	3.59	5.52×10^{-7}	2.99	5.52×10^{-7}	2.99
250×250	5.64×10^{-7}	3.52	2.83×10^{-7}	2.99	2.83×10^{-7}	2.99
ETD-RK4						
	MRWENO4		MRWENO6		MRWENO8	
$N \times M$	L^1 Error	Order	L^1 Error	Order	L^1 Error	Order
50×50	1.76×10^{-4}	-	7.68×10^{-7}	-	3.25×10^{-7}	-
100×100	1.10×10^{-5}	4.00	2.75×10^{-8}	4.80	2.06×10^{-8}	3.98
150×150	2.17×10^{-6}	4.00	4.70×10^{-9}	4.36	4.09×10^{-9}	3.98
200×200	6.88×10^{-7}	4.00	1.41×10^{-9}	4.19	1.30×10^{-9}	4.00
250×250	2.82×10^{-7}	4.00	5.62×10^{-10}	4.12	5.35×10^{-10}	3.96

Table 7
Example 2. Numerical errors of the ETD-RK multi-resolution A-WENO methods for the nonlinear stiff reaction-diffusion equation with the time-step size $\Delta t = 0.01\Delta x$ in three-dimensional space. MRWENO2r stands for the $2r$-th order multi-resolution A-WENO discretization in space.

ETD-RK3						
	MRWENO4		MRWENO6		MRWENO8	
$N \times M \times L$	L^1 Error	Order	L^1 Error	Order	L^1 Error	Order
$50 \times 50 \times 50$	4.74×10^{-5}	-	1.47×10^{-5}	-	1.46×10^{-5}	-
$100 \times 100 \times 100$	3.73×10^{-6}	3.67	1.88×10^{-6}	2.96	1.88×10^{-6}	2.96
$150 \times 150 \times 150$	9.01×10^{-7}	3.50	5.63×10^{-7}	2.98	5.63×10^{-7}	2.97
ETD-RK4						
	MRWENO4		MRWENO6		MRWENO8	
$N \times M \times L$	L^1 Error	Order	L^1 Error	Order	L^1 Error	Order
$50 \times 50 \times 50$	3.74×10^{-5}	-	2.87×10^{-7}	-	2.19×10^{-7}	-
$100 \times 100 \times 100$	2.34×10^{-6}	4.00	1.51×10^{-8}	4.24	1.41×10^{-8}	3.96
$150 \times 150 \times 150$	4.61×10^{-7}	4.00	2.90×10^{-9}	4.08	2.81×10^{-9}	3.98



(a) $d = 1$. (b) $d = 2$.

Fig. 3.2. Example 2. Comparison of efficiency for different time-stepping methods. SSP-IRK3-linear and ETD-RK3-linear indicate that the linear spatial discretization is used for them. CPU time unit: second.

Table 8
Example 2. Comparison of efficiency for SSP-ERK3 and ETD-RK3 methods for the three-dimensional nonlinear stiff reaction-diffusion equation. The fourth-order multi-resolution A-WENO spatial discretization is adopted. The time-step size $\Delta t = 0.01\Delta x$ is used in the ETD-RK3 method.

	SSP-ERK3		ETD-RK3	
$N \times M \times L$	L^1 Error	CPU (s)	L^1 Error	CPU (s)
$50 \times 50 \times 50$	3.72×10^{-5}	3.74×10^{2}	4.74×10^{-5}	6.26×10^{1}
$100 \times 100 \times 100$	2.33×10^{-6}	1.12×10^{4}	3.73×10^{-6}	1.10×10^{3}

therefore not shown here to save space. For $d = 1$, we take the computational domain $\Omega = [-6, 6]$ with $N = 200$ grid points and time-step size $\Delta t = \Delta x$. The numerical solutions are shown in Fig. 3.3. Furthermore, we use the same mesh size and time-step size in a larger computational domain to track the solution fronts at different times and the results are shown in Fig. 3.4. The excellent agreement between the numerical solutions and the exact ones at different times is observed in Fig. 3.4, which indicates that the scheme can track the movement of interface well. For $d = 2$, we take the computational domains $\Omega = [-6, 6]^2$ with $N \times M = 200 \times 200$ grid points for the $m = 2, 3, 5$ cases and $\Omega = [-7, 7]^2$ with $N \times M = 233 \times 233$ grid points for the $m = 8$ case, to capture the entire non-zero profile. The time-step size is taken as $\Delta t = \frac{1}{2}\Delta x$. The numerical results are shown in Fig. 3.5. For $d = 3$, we take the computational domains $\Omega = [0, 6]^3, [0, 7]^3, [0, 8]^3, [0, 9]^3$ for $m = 2, 3, 5, 8$ cases, respectively. Here to reduce the computational cost, we have exploited the symmetry of the problem by restricting the domain to the first octant and imposing the homogeneous Neumann boundary conditions. The mesh size is uniformly set to $\Delta x = \Delta y = \Delta z = \frac{1}{15}$, and the time-step size is taken as $\Delta t = \frac{1}{3}\Delta x$. The numerical results are shown in Fig. 3.6. From the numerical results presented in the figures, we observe that the sharp wave fronts are captured with non-oscillatory performance and the numerical solutions match the exact solutions very well in all space dimensions.

Then, we compare the computational efficiency of different third-order time-marching approaches for solving the equations. For $d = 1$, the solutions are computed up to $T = 10$ on the computational domain $\Omega = [-9, 9]$ with $N = 300$ grid points. For $d = 2$, the solutions are computed up to $T = 5$ on the computational domain $\Omega = [-8, 8]^2$ with $N \times M = 200 \times 200$ grid points. The time-step sizes used in different methods are chosen to be at their maximum values to achieve stable computations and numerical solutions that approximate well the exact solutions. The comparison of the ratios of the time-step sizes to the spatial grid size, $\frac{\Delta t}{\Delta x}$, and the corresponding CPU times of different methods are shown in Table 9 and 10 for $d = 1$ and 2, respectively. From the numerical results in the tables, we observe that the ETD-RK3 method, which is coupled with either the multi-resolution A-WENO spatial discretizations or the corresponding linear spatial discretizations, allows for much larger time-step sizes and takes much less CPU time costs than both the explicit and the implicit SSP-RK methods here. Moreover, the permitted maximum time-step sizes have very small changes as the complexity of the equation and the accuracy order of spatial discretization increase, which shows the robustness of the ETD-RK method. For $d = 3$, the computational time of SSP-ERK3 is significantly longer, and that of SSP-IRK3 appears prohibitively large. It is not feasible to provide a comprehensive comparison as we did for $d = 1$ and $d = 2$, since doing so would be excessively time-consuming. Instead, we conducted a specific test for the most challenging case, $m = 8$, with $\Delta x = \Delta y = \Delta z = \frac{1}{10}$. We found that, when coupled with the sixth-order spatial discretization, the CPU times for ETD-RK3 and SSP-ERK3 were 4.44×10^2 seconds and 1.84×10^4 seconds, respectively. Again, similar to the previous examples, the high computational efficiency of the ETD-RK method is verified here.

Finally, it is interesting to compare the performance of the WENO discretization from [41] with the multi-resolution A-WENO method adopted here. When coupled with the SSP-ERK3 time-stepping method, both WENO and A-WENO exhibit similar robustness in the one-dimensional tests. However, for the very stiff problem ($m = 8$), unlike the smooth problems in Examples 1 and 2, the WENO method [41] shows weaker nonlinear stability in higher dimensions for non-smooth solutions. Consequently, it requires much smaller time-step sizes and thus longer computational time. Fig. 3.7 shows the numerical results of the 2D case which is solved by the sixth-order A-WENO and WENO schemes. A much smaller CFL number (about five times smaller) has to be used for the WENO method in [41] to achieve a comparable numerical solution to the multi-resolution A-WENO method, and a slight larger CFL number for the WENO method [41] leads to oscillation / nonlinear instability around the sharp wave front. As a result, the multi-resolution A-WENO method is about five times faster than the WENO method [41] in solving this problem.

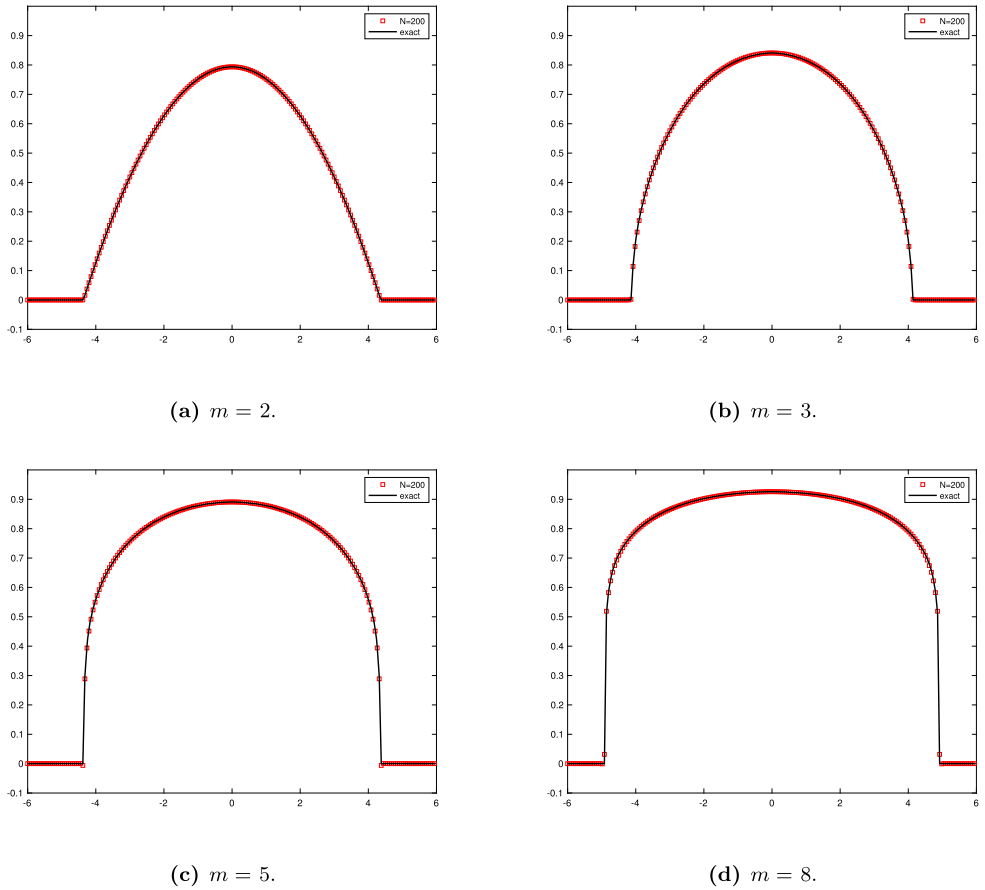
Example 4. Interacting and merging regions

We solve the PME (3.3) to investigate the interaction and merging of two compactly supported solutions. The ETD-RK4 scheme coupled with the sixth-order multi-resolution A-WENO spatial discretization is used in the simulations.

For $d = 1$, we solve the equation with $m = 6$. The interaction of two boxes is depicted by the initial condition

$$u(x, 0) = \begin{cases} 1, & -4 < x < -1, \\ 2, & 0 < x < 3, \\ 0, & \text{otherwise}, \end{cases} \tag{3.5}$$

on the domain $\Omega = [-6, 6]$ with the homogeneous Dirichlet boundary conditions. This setup models how temperature changes when two hot spots are suddenly placed in the domain. The solution is computed up to $T = 0.8$ on a grid with $N = 160$, and the time-step



(a) $m = 2$. (b) $m = 3$.

(c) $m = 5$. (d) $m = 8$.

Fig. 3.3. Example 3. Numerical solutions of the ETD-RK4 multi-resolution A-WENO6 method on the grid with $N = 200$, for the one-dimensional PME with different values of m for the Barenblatt solution at $T = 2$. The time-step size is $\Delta t = \Delta x$.

size is taken as $\Delta t = 0.01\Delta x$. The numerical results at different times are shown in Fig. 3.8, from which we observe merging of the two boxes over time. Similar to Example 3, the sharp wave fronts of the solution are captured stably with high resolution.

For $d = 2$, we solve the equation with $m = 2$ and the initial condition

$$u(x,y,0) = \begin{cases} \exp(\frac{-1}{6-(x-2)^2-(y+2)^2}), & (x-2)^2+(y+2)^2 < 6, \\ \exp(\frac{-1}{6-(x+2)^2-(y-2)^2}), & (x+2)^2+(y-2)^2 < 6, \\ 0, & \text{otherwise}, \end{cases} \tag{3.6}$$

which represents two cones on the domain $\Omega = [-10, 10]^2$. The homogeneous Dirichlet boundary conditions are applied. The solution is computed up to $T = 4$ on a grid with $N \times M = 100 \times 100$. The time-step size is taken as $\Delta t = 0.3\Delta x$. The numerical solution profiles at the times $t = 0, 0.5, 1$, and 4 are shown in Fig. 3.9, which illustrates the merging process of two cones. Similar to the 1D case, the sharp wave fronts of the solution are captured stably with high resolution in the simulation.

Example 5. Buckley-Leverett equations

In this test, we consider the viscous Buckley-Leverett equations, which are used to model the two-phase flow in porous media. The proposed ETD-RK4 multi-resolution A-WENO6 scheme is used to solve these problems.

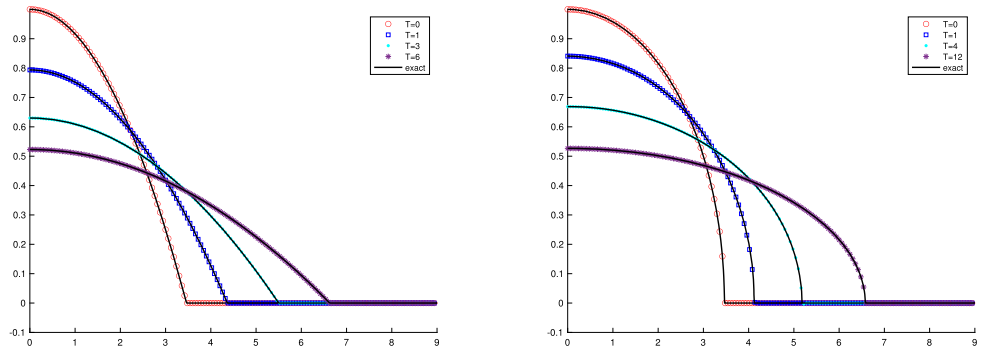
In one-dimensional space, the equation is given by

$$u_t + f(u)_x = \epsilon(v(u)u_x)_x. \tag{3.7}$$

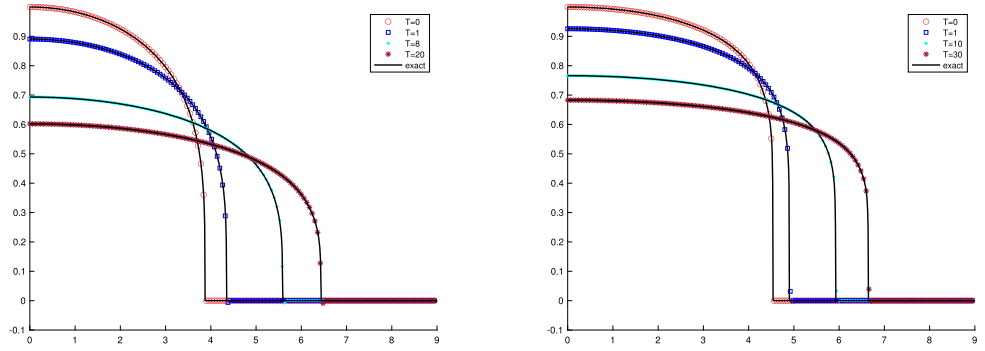
The nonlinear diffusion coefficient is taken as

$$v(u) = \begin{cases} 4u(1-u), & 0 \le u \le 1, \\ 0, & \text{otherwise}, \end{cases} \tag{3.8}$$

so that the function of the parabolic term $g(u)_{xx} = \epsilon(v(u)u_x)_x$ is given by



(a) $m = 2$. (b) $m = 3$.



(c) $m = 5$. (d) $m = 8$.

Fig. 3.4. Example 3. Numerical solutions of the ETD-RK4 multi-resolution A-WENO6 method for the one-dimensional PME with different values of m for the Barenblatt solution at different times $T + t_0 = T + 1$. T is the time points indicated in the pictures. The mesh size and time-step size are the same as those used in Fig. 3.3.

$$g(u) = \begin{cases} 0, & u < 0, \\ \epsilon(2u^2 - \frac{4}{3}u^3), & 0 \le u \le 1, \\ \frac{2}{3}\epsilon, & u > 1. \end{cases} \tag{3.9}$$

Two different convection fluxes are considered, namely, the flux without gravitational effects given by

$$f(u) = \frac{u^2}{u^2 + (1-u)^2}, \tag{3.10}$$

and the one with gravitational effects given by

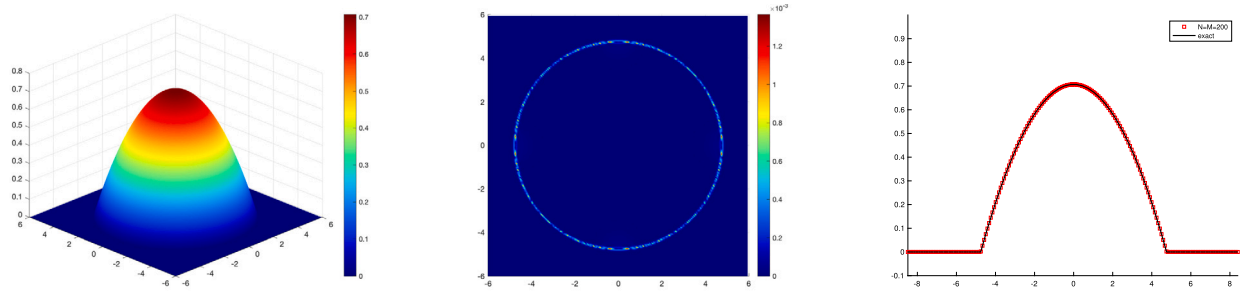
$$f(u) = \frac{u^2}{u^2 + (1-u)^2}(1 - 5(1-u)^2). \tag{3.11}$$

We take $\epsilon = 0.01$, and solve two initial-boundary value problems. The first initial-boundary value problem defined on $\Omega = [0, 1]$ has the initial condition

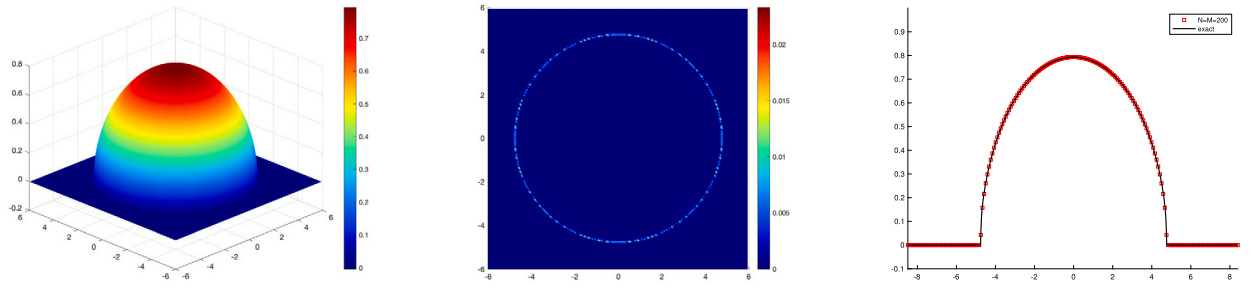
$$u(x, 0) = \begin{cases} 1 - 3x, & 0 \le x \le \frac{1}{3}, \\ 0, & \frac{1}{3} < x \le 1, \end{cases} \tag{3.12}$$

and the boundary condition $u(0, t) = 1, u(1, t) = 0$. The second problem is a Riemann problem defined on $\Omega = [0, 1]$, which has the initial condition

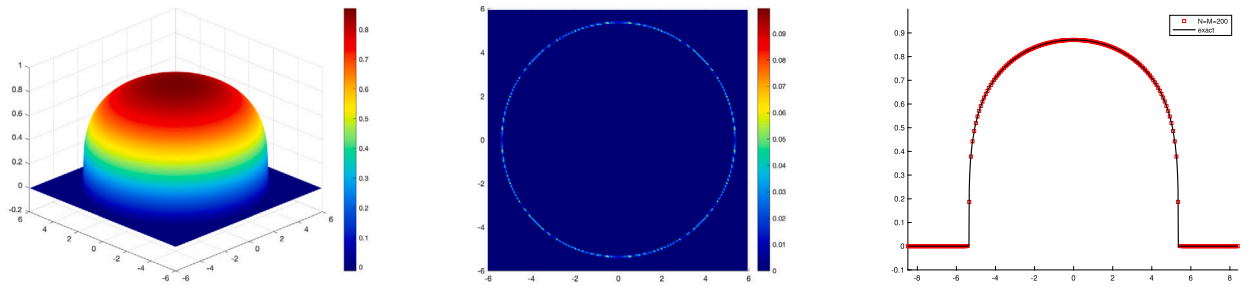
$$u(x, 0) = \begin{cases} 0, & 0 \le x < 1 - \frac{1}{\sqrt{2}}, \\ 1, & 1 - \frac{1}{\sqrt{2}} \le x \le 1, \end{cases} \tag{3.13}$$



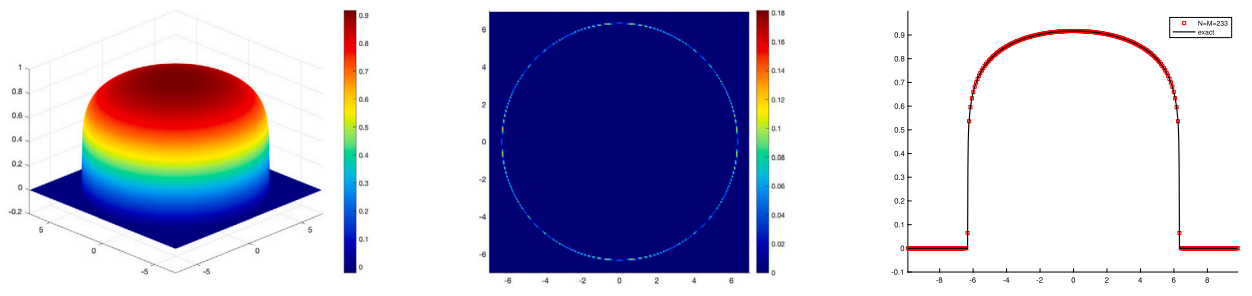
(a) Surface, $m = 2$ (b) Error, $m = 2$ (c) Slice, $m = 2$



(d) Surface, $m = 3$ (e) Error, $m = 3$ (f) Slice, $m = 3$

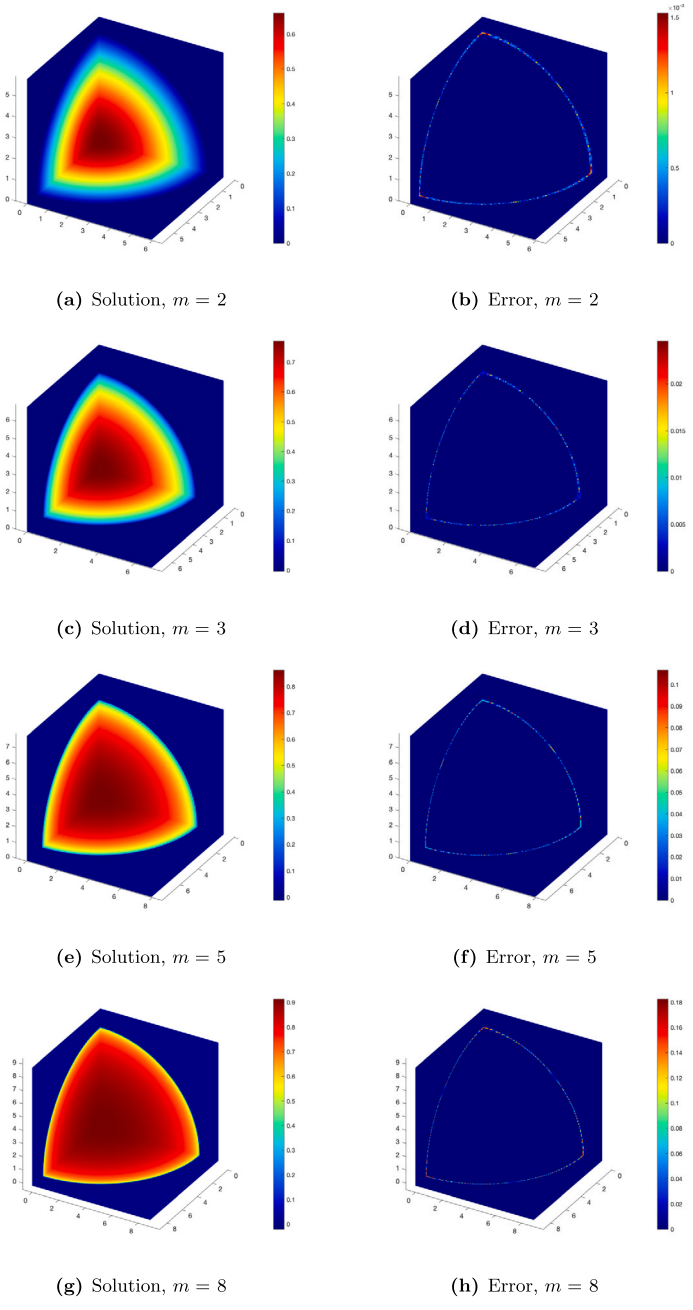


(g) Surface, $m = 5$ (h) Error, $m = 5$ (i) Slice, $m = 5$



(j) Surface, $m = 8$ (k) Error, $m = 8$ (l) Slice, $m = 8$

Fig. 3.5. Example 3. Numerical approximations at $T = 2$ to the Barenblatt solutions of the two-dimensional PME with different values of m. (a),(d),(g),(j): surface plots of the numerical solutions; (b), (e), (h), (k): contour plots of the numerical errors; (c), (f), (i), (l): 1D slice-plots of the numerical solutions along $x = y$. The ETD-RK4 multi-resolution A-WENO6 method is used on the grid with $N \times M = 200 \times 200$ for the $m = 2, 3, 5$ cases and on the grid with $N \times M = 233 \times 233$ for the $m = 8$ case. The time-step size is $\Delta t = \frac{1}{2}\Delta x$. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)



(a) Solution, $m = 2$ (b) Error, $m = 2$

(c) Solution, $m = 3$ (d) Error, $m = 3$

(e) Solution, $m = 5$ (f) Error, $m = 5$

(g) Solution, $m = 8$ (h) Error, $m = 8$

Fig. 3.6. Example 3. Numerical approximations at $T = 2$ to the Barenblatt solutions of the three-dimensional PME with different values of m. (a),(c),(e),(g): the numerical solutions; (b), (d), (f), (h): the numerical errors. The ETD-RK4 multi-resolution A-WENO6 method is used to solve the problems on grids with uniform mesh size $h = \frac{1}{15}$. The time-step size is $\Delta t = \frac{1}{3}\Delta x$.

and the boundary condition $u(0,t) = 0$ and $u(1,t) = 1$. The simulations are performed on the computational grids with $N = 100$ and $N = 800$. The time-step sizes are chosen to be the maximum permitted values for achieving a stable computation in each problem. The obtained numerical solutions at $T = 0.2$ are shown in Fig. 3.10. We observe that the numerical solutions on different grids match very well, which verifies the convergence of the numerical solutions. Similar to the previous examples, the large gradients of the solutions are captured stably with high resolution in the simulations, which verifies the nonlinear stability of the method. Moreover, the desired time-step size $\Delta t \sim O(\Delta x)$ for the ETD-RK method is still preserved on different grids in this example.

In two-dimensional space, the equation is given by

$$u_t + f_1(u)_x + f_2(u)_y = \epsilon(u_{xx} + u_{yy}), \tag{3.14}$$

Table 9
Example 3. Maximum ratios of the time-step sizes to the spatial grid size and the corresponding CPU times of different methods in the computation of the Barenblatt solutions of the one-dimensional PME. MRWENO2r stands for the 2r-th order multi-resolution A-WENO discretization in space. SSP-IRK3-linear and ETD-RK3-linear indicate that the corresponding linear spatial discretizations of the MRWENO schemes are used.

$m=2$						
Time	MRWENO4		MRWENO6		MRWENO8	
marching	$\Delta t/\Delta x$	CPU (s)	$\Delta t/\Delta x$	CPU (s)	$\Delta t/\Delta x$	CPU (s)
ETD-RK3	1.6	0.27	1.6	0.31	1.5	0.35
SSP-ERK3	0.0198	2.13	0.0198	1.93	0.0198	2.82
ETD-RK3-linear	1.6	0.26	1.5	0.30	1.4	0.33
SSP-IRK3-linear	0.3	1.28	0.3	1.41	0.2	2.31
$m=3$						
Time	MRWENO4		MRWENO6		MRWENO8	
marching	$\Delta t/\Delta x$	CPU (s)	$\Delta t/\Delta x$	CPU (s)	$\Delta t/\Delta x$	CPU (s)
ETD-RK3	1.5	0.34	1.4	0.38	1.5	0.42
SSP-ERK3	0.0166	2.72	0.0166	2.54	0.0166	3.53
ETD-RK3-linear	1.5	0.32	1.4	0.38	1.5	0.41
SSP-IRK3-linear	0.2	2.00	0.2	2.18	0.2	2.38
$m=5$						
Time	MRWENO4		MRWENO6		MRWENO8	
marching	$\Delta t/\Delta x$	CPU (s)	$\Delta t/\Delta x$	CPU (s)	$\Delta t/\Delta x$	CPU (s)
ETD-RK3	1.5	0.42	1.5	0.47	1.5	0.51
SSP-ERK3	0.0125	3.73	0.0125	3.47	0.0125	4.74
ETD-RK3-linear	1.4	0.40	1.4	0.47	1.4	0.51
SSP-IRK3-linear	0.1	3.83	0.1	4.14	0.1	4.60
$m=8$						
Time	MRWENO4		MRWENO6		MRWENO8	
marching	$\Delta t/\Delta x$	CPU (s)	$\Delta t/\Delta x$	CPU (s)	$\Delta t/\Delta x$	CPU (s)
ETD-RK3	1.4	0.51	1.4	0.57	1.4	0.62
SSP-ERK3	0.009	5.25	0.009	4.90	0.009	6.65
ETD-RK3-linear	1.4	0.50	1.3	0.58	1.3	0.62
SSP-IRK3-linear	0.09	4.52	0.07	5.94	0.06	7.59

where $f_1(u)$ and $f_2(u)$ are the fluxes without and with gravitational effects as in the one-dimensional case (3.10) and (3.11) respectively, and $\epsilon = 0.01$. We solve the problem on the domain $\Omega = [-\frac{3}{2}, \frac{3}{2}]^2$ with the initial condition

$$u(x, y, 0) = \begin{cases} 1, & x^2 + y^2 < \frac{1}{2}, \\ 0, & \text{otherwise}, \end{cases} \tag{3.15}$$

and the homogeneous Dirichlet boundary condition. The computation is performed on the grid with $N \times M = 120 \times 120$. The time-step size is taken as $\Delta t = 0.3\Delta x$. The obtained numerical solution at $T = 0.5$ is shown in Fig. 3.11. Similar to the 1D case, the large gradients of the solution are captured stably with high resolution and without numerical oscillation in the simulation of this 2D problem, which verifies the nonlinear stability of the proposed method.

Example 6. Strongly degenerate convection-diffusion equations

We consider the convection-diffusion equations

$$u_t + (u^2)_x = \epsilon(v(u)u_x)_x, \tag{3.16}$$

and

$$u_t + (u^2)_x + (u^2)_y = \epsilon(v(u)u_x)_x + \epsilon(v(u)u_y)_y, \tag{3.17}$$

in one- and two-dimensional spaces, respectively, with the strongly degenerate viscosity coefficient

$$v(u) = \begin{cases} 0, & |u| \le \frac{1}{4}, \\ 1, & |u| > \frac{1}{4}, \end{cases} \tag{3.18}$$

Table 10
Example 3. Maximum ratios of the time-step sizes to the spatial grid size and the corresponding CPU times of different methods in the computation of the Barenblatt solutions of the two-dimensional PME. MRWENO2r stands for the $2r$-th order multi-resolution A-WENO discretization in space. SSP-IRK3-linear and ETD-RK3-linear indicate that the corresponding linear spatial discretizations of the MRWENO schemes are used.

$m=2$						
Time	MRWENO4		MRWENO6		MRWENO8	
marching	$\Delta t/\Delta x$	CPU (s)	$\Delta t/\Delta x$	CPU (s)	$\Delta t/\Delta x$	CPU (s)
ETD-RK3	1.0	2.6×10^{1}	1.0	3.0×10^{1}	0.9	3.5×10^{1}
SSP-ERK3	0.0129	8.8×10^{2}	0.0129	8.9×10^{2}	0.0129	1.1×10^{3}
ETD-RK3-linear	1.0	2.5×10^{1}	0.9	3.0×10^{1}	0.9	3.4×10^{1}
SSP-IRK3-linear	0.2	4.4×10^{2}	0.2	6.7×10^{2}	0.2	1.1×10^{3}
$m=3$						
Time	MRWENO4		MRWENO6		MRWENO8	
marching	$\Delta t/\Delta x$	CPU (s)	$\Delta t/\Delta x$	CPU (s)	$\Delta t/\Delta x$	CPU (s)
ETD-RK3	1.0	3.2×10^{1}	1.0	3.5×10^{1}	1.0	4.0×10^{1}
SSP-ERK3	0.01	1.1×10^{3}	0.01	1.2×10^{3}	0.01	1.4×10^{3}
ETD-RK3-linear	1.0	3.0×10^{1}	1.0	3.4×10^{1}	1.0	3.8×10^{1}
SSP-IRK3-linear	0.2	4.5×10^{2}	0.1	1.3×10^{3}	0.1	1.8×10^{3}
$m=5$						
Time	MRWENO4		MRWENO6		MRWENO8	
marching	$\Delta t/\Delta x$	CPU (s)	$\Delta t/\Delta x$	CPU (s)	$\Delta t/\Delta x$	CPU (s)
ETD-RK3	1.0	3.9×10^{1}	1.0	4.4×10^{1}	1.0	5.0×10^{1}
SSP-ERK3	0.0067	1.7×10^{3}	0.0067	1.8×10^{3}	0.0067	2.1×10^{3}
ETD-RK3-linear	1.0	3.8×10^{1}	1.0	4.3×10^{1}	1.0	4.8×10^{1}
SSP-IRK3-linear	0.1	9.7×10^{2}	0.09	1.7×10^{3}	0.08	2.3×10^{3}
$m=8$						
Time	MRWENO4		MRWENO6		MRWENO8	
marching	$\Delta t/\Delta x$	CPU (s)	$\Delta t/\Delta x$	CPU (s)	$\Delta t/\Delta x$	CPU (s)
ETD-RK3	1.0	4.9×10^{1}	1.0	5.6×10^{1}	1.0	6.4×10^{1}
SSP-ERK3	0.0045	2.6×10^{3}	0.0045	2.7×10^{3}	0.0045	3.1×10^{3}
ETD-RK3-linear	1.0	4.8×10^{1}	1.0	5.5×10^{1}	0.9	6.3×10^{1}
SSP-IRK3-linear	0.06	1.9×10^{3}	0.05	3.2×10^{3}	0.04	6.1×10^{3}

and the corresponding function of the parabolic term

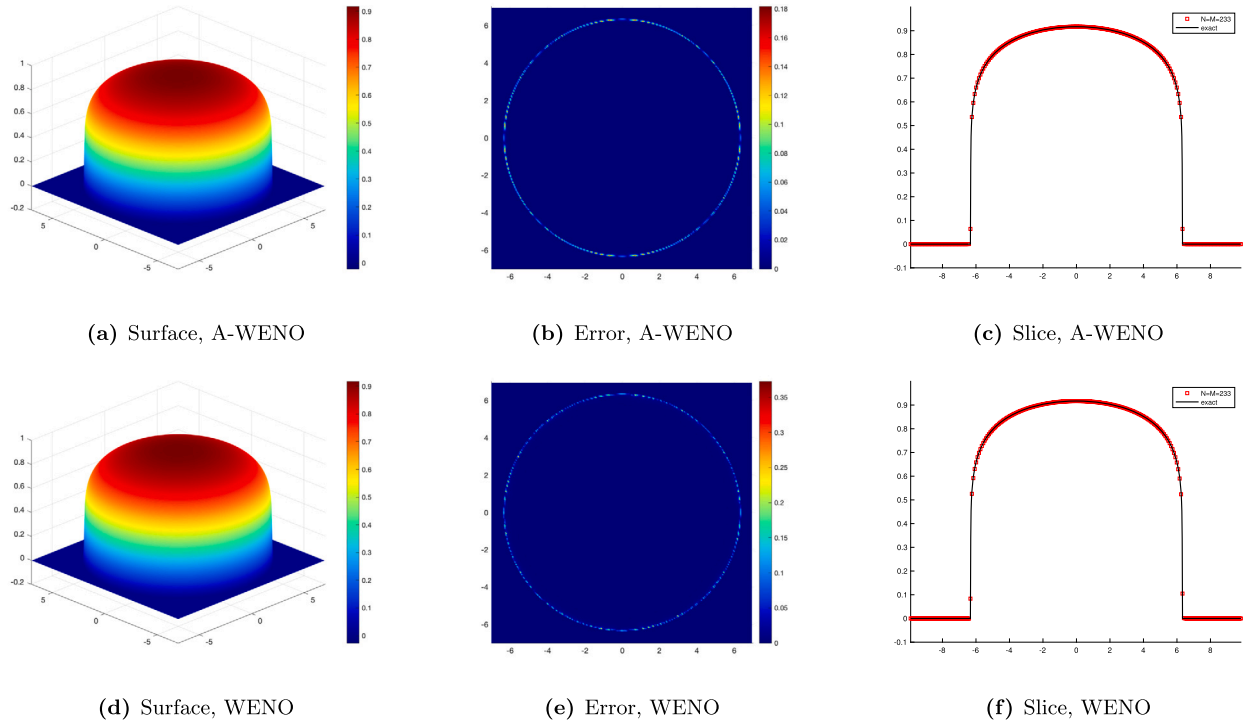
$$g(u)=\begin{cases}\epsilon(u+\frac{1}{4}), & u<-\frac{1}{4},\\ \epsilon(u-\frac{1}{4}), & u>\frac{1}{4},\\ 0, & |u|\le\frac{1}{4}.\end{cases} \tag{3.19}$$

So the equation is hyperbolic when $u\in[-\frac{1}{4},\frac{1}{4}]$ and parabolic otherwise. We take $\epsilon=0.1$ in the tests. The ETD-RK4 multi-resolution A-WENO6 scheme is used to perform the simulations.

In one-dimensional space, we solve the problem on the domain $\Omega=[-2,2]$ with the initial condition

$$u(x,0)=\begin{cases}1, & -\frac{1}{\sqrt{2}}-\frac{2}{5}<x<-\frac{1}{\sqrt{2}}+\frac{2}{5},\\ -1, & \frac{1}{\sqrt{2}}-\frac{2}{5}<x<\frac{1}{\sqrt{2}}+\frac{2}{5},\\ 0, & \text{otherwise},\end{cases} \tag{3.20}$$

and the homogeneous Dirichlet boundary condition. The computational grids with $N=100$ and $N=800$ are used in computation. The time-step size is taken as $\Delta t=0.08\Delta x$ such that the desired time-step size condition, $\Delta t\sim O(\Delta x)$, for the ETD-RK method is satisfied on different grids. The obtained numerical results at $T=0.7$ are reported in Fig. 3.12. It is observed that the numerical solutions on different grids match very well, which verifies the convergence of the numerical solutions. The sharp interfaces of the solutions and the kinks where the type of the equation changes are captured stably with high resolution in the simulations, which indicates good nonlinear stability of the method.



(a) Surface, A-WENO (b) Error, A-WENO (c) Slice, A-WENO

(d) Surface, WENO (e) Error, WENO (f) Slice, WENO

Fig. 3.7. Example 3. Numerical approximations at $T = 2$ to the Barenblatt solutions of the two-dimensional PME with $m = 8$ on the grid $N \times M = 233 \times 233$. Row 1: sixth-order multi-resolution A-WENO method with SSP-ERK3 time stepping and CFL number 0.4; Row 2: sixth-order WENO method from [41] with SSP-ERK3 time stepping and CFL number 0.084. (a),(d): surface plots of the numerical solutions; (b), (e): contour plots of the numerical errors; (c), (f): 1D slice-plots of the numerical solutions along $x = y$. The WENO method established in [41] exhibits weaker nonlinear stability, requiring much smaller time-step sizes and thereby significantly longer (roughly 5 times) computational time.

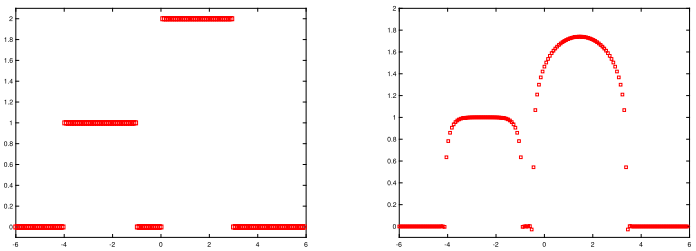
In two-dimensional space, we solve the problem on the domain $\Omega = [-\frac{3}{2}, \frac{3}{2}]^2$ with the initial condition

$$u(x, y, 0) = \begin{cases} 1, & (x + \frac{1}{2})^2 + (y + \frac{1}{2})^2 < \frac{4}{25}, \\ -1, & (x - \frac{1}{2})^2 + (y - \frac{1}{2})^2 < \frac{4}{25}, \\ 0, & \text{otherwise}, \end{cases} \tag{3.21}$$

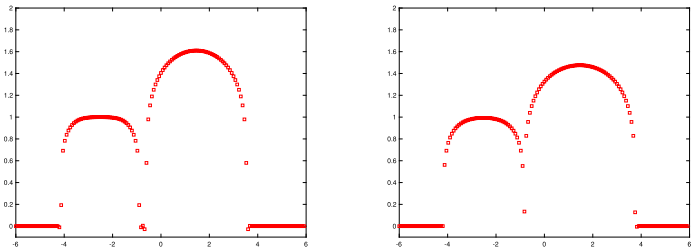
and the homogeneous Dirichlet boundary condition. The computational grid with $N \times M = 120 \times 120$ is adopted. The time-step size is taken as $\Delta t = 0.1\Delta x$. The obtained numerical solution at $T = 0.5$ is shown in Fig. 3.13. Again, similar to the 1D example, the sharp interfaces of the solution are captured stably with high resolution and without numerical oscillation in the simulation of this 2D degenerate convection-diffusion problem, which verifies the nonlinear stability of the proposed numerical method.

4. Conclusions and discussions

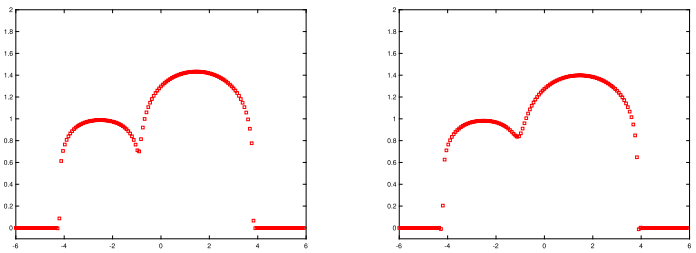
High-order WENO methods have been well developed in the literature to effectively solve nonlinear degenerate parabolic equations with high resolution. However, most of them are coupled with explicit temporal numerical schemes for the evolution of simulation. Since the sophisticated nonlinear properties and high-order accuracy of WENO methods require more operations than many other schemes, their computational costs increase significantly when they are applied to stiff degenerate parabolic PDEs and the time-step sizes are small under the CFL condition, especially for multidimensional problems. How to achieve fast computations of high-order WENO methods is a very important and challenging question. To tackle this challenge, implicit temporal schemes have been applied to the high-order WENO methods for the nonlinear degenerate parabolic equations, which are more efficient than explicit schemes. Due to the highly nonlinear property of the high-order WENO spatial discretizations, sophisticated techniques [4,65] need to be designed for efficiently solving the complex nonlinear systems from implicit schemes. Instead of implicit schemes, in this paper we apply the ETD-RK methods, a class of accurate exponential integrators, to the high-order multi-resolution alternative finite difference WENO methods for solving degenerate parabolic equations efficiently and dealing with the challenge of stiffness. The motivation of studying exponential integrators here is that although this class of stiff-problem solvers are also global schemes as implicit schemes, their implementations are essentially explicit. In other words, exponential integrators have some advantages of both explicit schemes and implicit schemes, which lead to the relatively simple implementation of the methods and excellent computational efficiency. However, it is difficult to directly use the popular semilinearization approach in the exponential Rosenborg-type method and apply exponential integrators to highly nonlinear schemes such as the WENO schemes for solving a fully nonlinear PDE, since the Jacobian



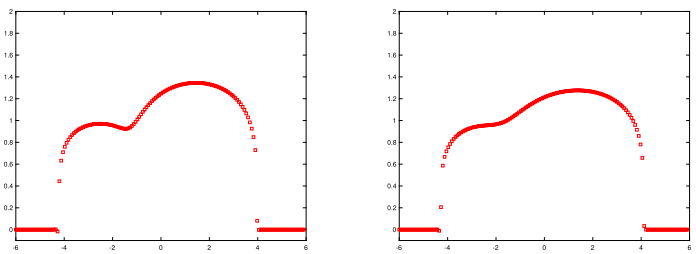
(a) $t = 0$ (b) $t = 0.01$



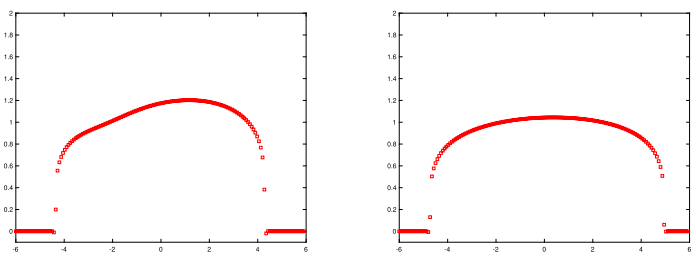
(c) $t = 0.02$ (d) $t = 0.04$



(e) $t = 0.05$ (f) $t = 0.06$

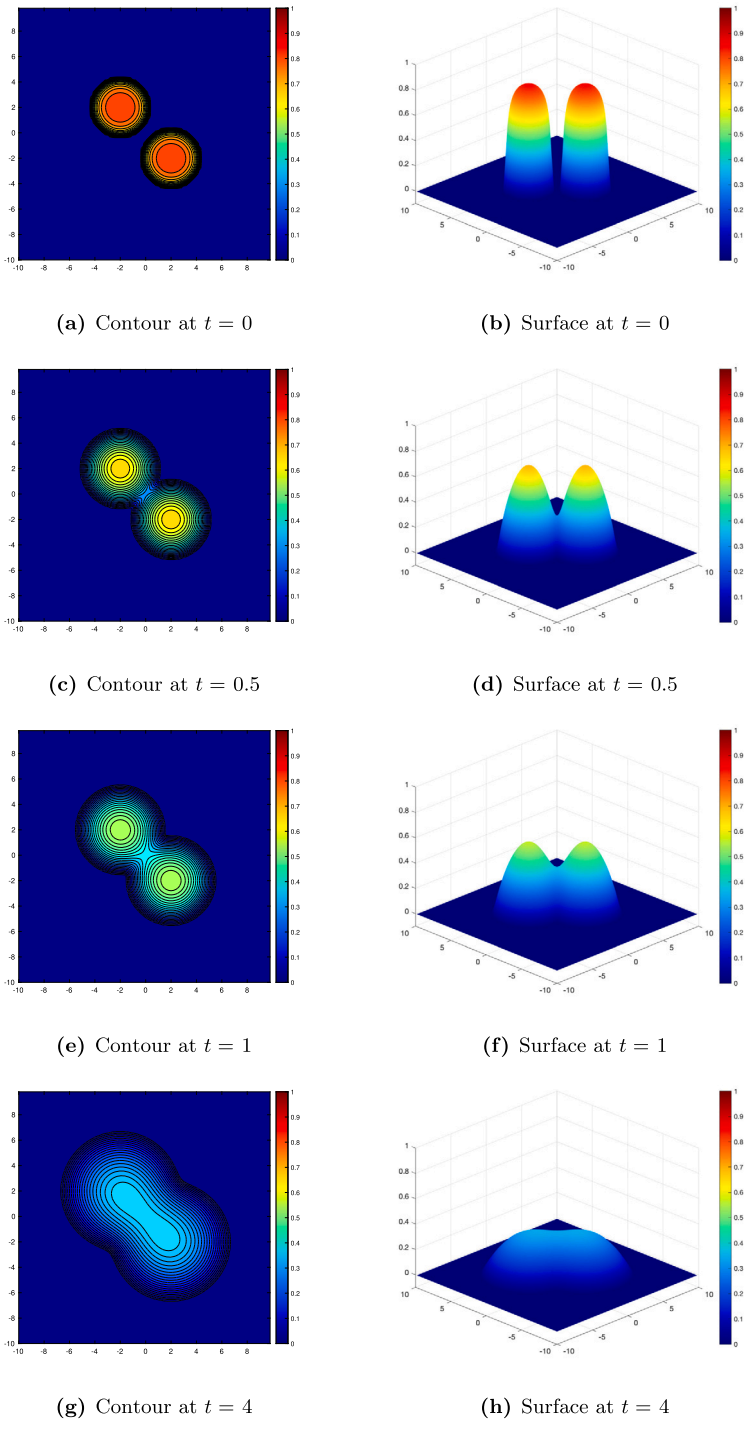


(g) $t = 0.08$ (h) $t = 0.12$



(i) $t = 0.2$ (j) $t = 0.8$

Fig. 3.8. Example 4. Numerical solutions of the ETD-RK4 multi-resolution A-WENO6 method on the grid with $N = 160$, for the one-dimensional problem of interaction of two boxes. The time-step size is $\Delta t = 0.01\Delta x$.



(a) Contour at $t = 0$ (b) Surface at $t = 0$

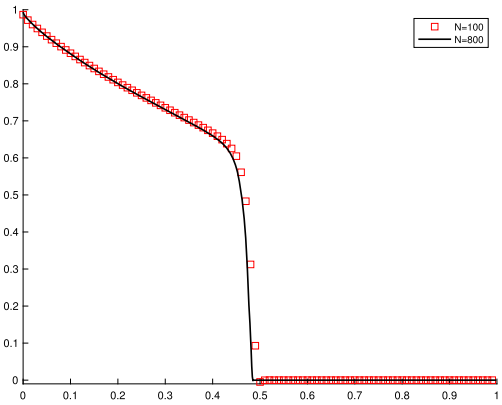
(c) Contour at $t = 0.5$ (d) Surface at $t = 0.5$

(e) Contour at $t = 1$ (f) Surface at $t = 1$

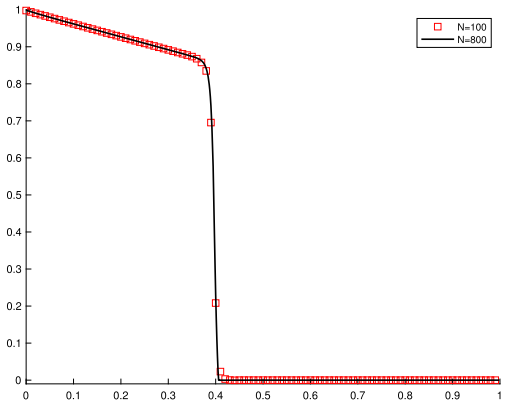
(g) Contour at $t = 4$ (h) Surface at $t = 4$

Fig. 3.9. Example 4. Numerical solutions for the problem of the merging cones, modeled by the two-dimensional PME. The ETD-RK4 multi-resolution A-WENO6 scheme is use on a grid with $N \times M = 100 \times 100$. The time-step size is $\Delta t = 0.3\Delta x$.

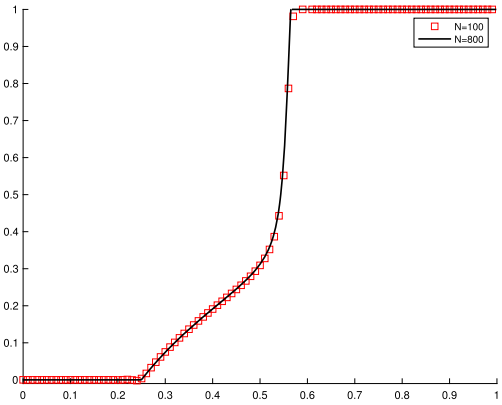
matrix from the highly nonlinear schemes is too complicated and expensive to compute at every time step. In this paper, a novel and effective semilinearization approach, namely replacing the exact Jacobian of a high-order nonlinear scheme such as a WENO scheme with that of the corresponding high-order linear scheme, is proposed to efficiently form the linear stiff part in applying the exponential integrators. This novel method is designed based on the property of the degenerate parabolic PDEs. Namely, the diffusion coefficient vanishes in non-smooth regions of the solution and the stiffness of the parabolic PDEs occurs in smooth regions. Since the nonlinear scheme is well approximated by its corresponding linear scheme in the smooth regions of the PDEs (i.e., the stiff regions),



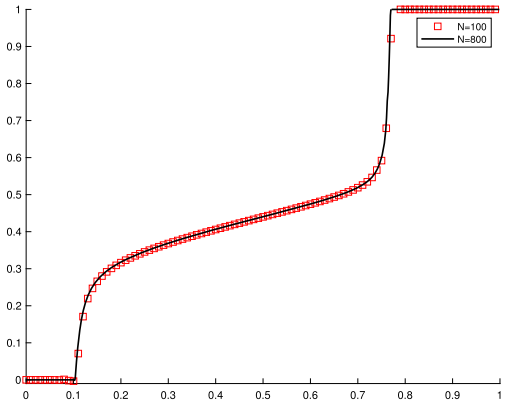
(a) Initial-boundary value problem (3.12) without gravity. Time-step size satisfies $\frac{\Delta t}{\Delta x} = 0.9$.



(b) Initial-boundary value problem (3.12) with gravity. Time-step size satisfies $\frac{\Delta t}{\Delta x} = 0.5$.

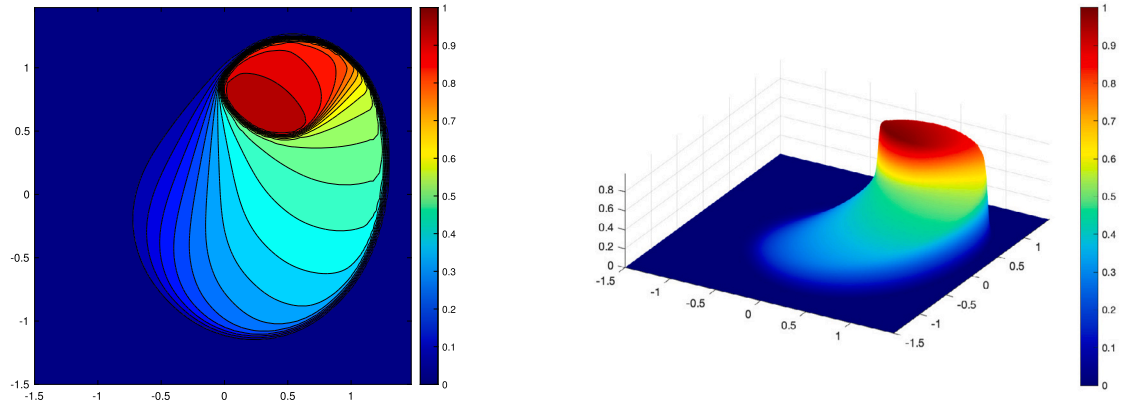


(c) Riemann problem (3.13) without gravity. Time-step size satisfies $\frac{\Delta t}{\Delta x} = 0.4$.



(d) Riemann problem (3.13) with gravity. Time-step size satisfies $\frac{\Delta t}{\Delta x} = 0.1$.

Fig. 3.10. Example 5. Numerical solutions of the one-dimensional Buckley-Leverett equation at $T = 0.2$. The ETD-RK4 multi-resolution A-WENO6 scheme is used. The time-step sizes are chosen to be the maximum permitted values for achieving a stable computation.



(a) Contour

(b) Surface

Fig. 3.11. Example 5. Numerical solution of the two-dimensional Buckley–Leverett equation at $T = 0.5$. The ETD-RK4 multi-resolution A-WENO6 scheme is used to solve the problem on a grid with $N \times M = 120 \times 120$. The time-step size is $\Delta t = 0.3\Delta x$.

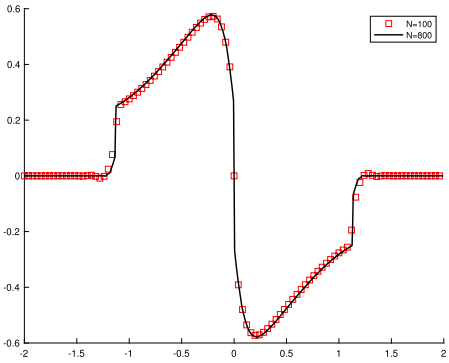
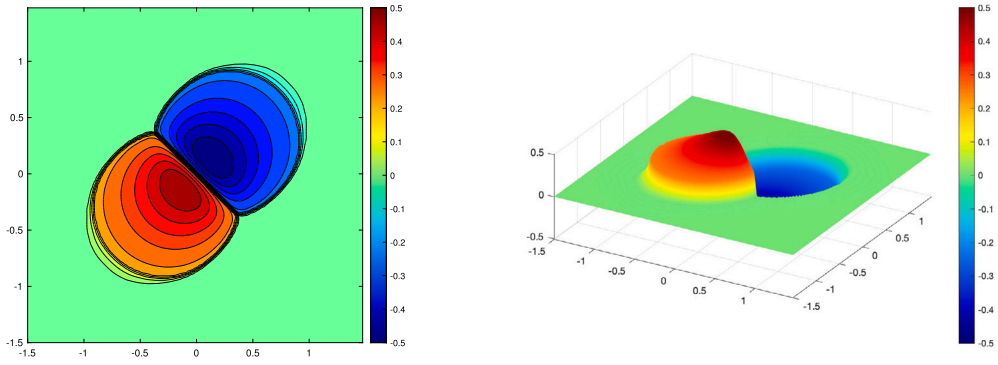


Fig. 3.12. Example 6. Numerical solutions of the one-dimensional strongly degenerate convection-diffusion equation at $T = 0.7$. The ETD-RK4 multi-resolution A-WENO6 scheme is used. The time-step size is $\Delta t = 0.08\Delta x$.



(a) Contour (b) Surface

Fig. 3.13. Example 6. Numerical solution of the two-dimensional strongly degenerate convection-diffusion equation at $T = 0.5$. The ETD-RK4 multi-resolution A-WENO6 scheme is used on a grid with $N \times M = 120 \times 120$. The time-step size is $\Delta t = 0.1\Delta x$.

the proposed construction of the linear stiff term based on the corresponding linear scheme is effective here for using the exponential integrators to solve the degenerate parabolic PDEs. Extensive numerical experiments are performed to demonstrate the effectiveness of this new approach, and verify high-order accuracy, nonlinear stability and high efficiency of the developed ETD-RK multi-resolution alternative WENO methods. The ETD-RK methods resolve the stiffness of the nonlinear degenerate parabolic PDEs very well, and the desired large time-step size simulations of $\Delta t \sim O(\Delta x)$ are achieved. The comparisons with some commonly used explicit and implicit SSP-RK methods show that the proposed methods are more efficient in solving the nonlinear degenerate parabolic PDEs, especially the multidimensional problems.

In this paper, we focus on applying the ETD-RK schemes with the new semilinearization approach to high-order multi-resolution A-WENO spatial discretizations for solving the nonlinear degenerate parabolic PDEs. However, the proposed novel semilinearization approach in applying the exponential integrators to a high-order nonlinear spatial discretization is a general method and is expected to have a broad impact. For example, the method can be applied to other high-order nonlinear schemes besides the WENO schemes, such as the discontinuous Galerkin (DG) finite element methods. The DG methods have been developed to solve the nonlinear degenerate parabolic PDEs (e.g. [66,56]), which have advantages to deal with complex domain geometries in application problems over the finite difference methods. The DG methods often use a nonlinear limiter or a nonlinear damping term to control numerical oscillations and ensure nonlinear stability in non-smooth regions of the solution. They are also nonlinear schemes with rich structures. Again, most of them are still coupled with explicit temporal schemes to solve the stiff PDEs such as the nonlinear degenerate parabolic PDEs, which are often computationally expensive due to very small time-step sizes under restrictive CFL condition. The exponential integrators with the new semilinearization method developed in this paper are ready to be applied to the high-order DG spatial discretizations and obtain significantly more efficient DG finite element methods for solving multidimensional degenerate parabolic PDEs on complex domain geometries, which have broad applications in engineering problems. According to the method developed in this paper, a DG formulation without the nonlinear limiter or the nonlinear damping term will serve as the linear scheme in the semilinearization procedure for the exponential integrators. In smooth regions (the stiff regions) of the solution, the full nonlinear DG scheme agrees well with its corresponding linear scheme, hence the simple and computational cheap Jacobian matrix based on the linear scheme captures the stiffness of the PDEs and efficient computations will be obtained for the high-order nonlinear DG methods coupled with the exponential integrators. This interesting project will be carried out in our next research.

CRediT authorship contribution statement

Ziyao Xu: Writing – review & editing, Writing – original draft, Validation, Software, Methodology, Investigation, Formal analysis, Conceptualization. **Yong-Tao Zhang:** Writing – review & editing, Writing – original draft, Validation, Supervision, Methodology, Investigation, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. The smoothness indicators of the multi-resolution WENO in Section 2.1.1

The expressions of the smoothness indicators $\beta_1, \beta_2, \beta_3$ and β_4 in the definition (2.11) are given as follows:

$$
\begin{aligned}
\beta_1 =& (-g_i + g_{i+1})^2,\\
\beta_2 =& \frac{781}{720}(g_{i-1} - 3g_i + 3g_{i+1} - g_{i+2})^2 + \frac{13}{48}(g_{i-1} - g_i - g_{i+1} + g_{i+2})^2 + (g_{i-1} - g_i)^2,\\
\beta_3 =& \frac{21520059541}{19838649600}(g_{i-2} - 5g_{i-1} + 10g_i - 10g_{i+1} + 5g_{i+2} - g_{i+3})^2\\
&+ \frac{1}{440858880}(1851g_{i-2} - 31123g_{i-1} + 84114g_i - 84114g_{i+1} + 31123g_{i+2} - 1851g_{i+3})^2\\
&+ \frac{1}{2246400}(131g_{i-2} - 1173g_{i-1} + 1042g_i + 1042g_{i+1} - 1173g_{i+2} + 131g_{i+3})^2\\
&+ \frac{1421461}{5241600}(g_{i-2} - 3g_{i-1} + 2g_i + 2g_{i+1} - 3g_{i+2} + g_{i+3})^2 + (-g_i + g_{i+1})^2,\\
\beta_4 =& \frac{1}{326918592000}(459034864256g_{i-3}^2 + 21743036840504g_{i-2}^2 + 193082473956456g_{i-1}^2\\
&- 633842107028865g_{i-1}g_i + 533907688202000g_i^2 + 610844549719320g_{i-1}g_{i+1}\\
&- 1054387388310025g_ig_{i+1} + 533907688202000g_{i+1}^2 - 346465395978597g_{i-1}g_{i+2}\\
&+ 610844549719320g_ig_{i+2} - 633842107028865g_{i+1}g_{i+2} + 193082473956456g_{i+2}^2\\
&+ 107421495993504g_{i-1}g_{i+3} - 192940877965535g_ig_{i+3} + 204591754773560g_{i+1}g_{i+3}\\
&- 127805625375939g_{i+2}g_{i+3} + 21743036840504g_{i+3}^2 - 7g_{i-2}(18257946482277g_{i-1}\\
&- 29227393539080g_i + 27562982566505g_{i+1} - 15345927999072g_{i+2}\\
&+ 4682081208019g_{i+3} - 605170517992g_{i+4}) + g_{i-3}(-6214446276409g_{i-2}\\
&+ 17764726801752g_{i-1} - 27790531210295g_i + 25709223617840g_{i+1}\\
&- 14082592044087g_{i+2} + 4236193625944g_{i+3} - 540644243257g_{i+4})\\
&- 14082592044087g_{i-1}g_{i+4} + 25709223617840g_ig_{i+4} - 27790531210295g_{i+1}g_{i+4}\\
&+ 17764726801752g_{i+2}g_{i+4} - 6214446276409g_{i+3}g_{i+4} + 459034864256g_{i+4}^2).
\end{aligned}
$$

Appendix B. The smoothness indicators of the multi-resolution WENO in Section 2.1.2

The expressions of the smoothness indicators $\beta_1, \beta_2, \beta_3$ and β_4 in the definition (2.35) are given as follows:

$$
\begin{aligned}
\beta_1 =& \frac{121}{300}(4u_{i-1}^2 - 13u_{i-1}u_i + 13u_i^2 + 5u_{i-1}u_{i+1} - 13u_iu_{i+1} + 4u_{i+1}^2),\\
\beta_2 =& \frac{1}{67200000}(112756316u_{i-2}^2 + 1657473113u_{i-1}^2 + 3613771547u_i^2 - 4707412996u_iu_{i+1}\\
&+ 1657473113u_{i+1}^2 + u_{i-1}(-4707412996u_i + 2846027902u_{i+1} - 631012985u_{i+2})\\
&- 37u_{i-2}(22231031u_{i-1} - 29557877u_i + 17054405u_{i+1} - 3632623u_{i+2})\\
&+ 1093641449u_iu_{i+2} - 822548147u_{i+1}u_{i+2} + 112756316u_{i+2}^2),\\
\beta_3 =& \frac{1}{696729600000000}(1191368301143900u_{i-3}^2 + 40597776375544695u_{i-2}^2 + 247305166240620450u_{i-1}^2\\
&- 645469279961828850u_{i-1}u_i + 435991844146445900u_i^2 + 462392655194742375u_{i-1}u_{i+1}\\
&- 645469279961828850u_iu_{i+1} + 247305166240620450u_{i+1}^2 - 173309501486101245u_{i-1}u_{i+2}
\end{aligned}
$$

$$
\begin{aligned}
&+ 248679888594363540u_iu_{i+2} - 196807231970740065u_{i+1}u_{i+2} + 40597776375544695u_{i+2}^2\\
&- 1111u_{i-3}(12262548688269u_{i-2} - 28726145190345u_{i-1} + 35285736074690u_i\\
&- 24003850977690u_{i+1} + 8594065910145u_{i+2} - 1267677095269u_{i+3})\\
&+ 26668278436213590u_{i-1}u_{i+3} - 39202452778980590u_iu_{i+3} + 31914747306473295u_{i+1}u_{i+3}\\
&- 13623691592666859u_{i+2}u_{i+3} + 11913683011439000u_{i+3}^2 - 3u_{i-2}(6560241065691355u_{i-1}\\
&- 82893296198121180u_i + 57769833828700415u_{i+1} - 21137663643408778u_{i+2} + 3182669075390365u_{i+3})),
\end{aligned}
$$

$$
\begin{aligned}
\beta_4 = &\frac{1}{669529276416000000000}(1180230306475238657612 68u_{i-4}^2 + 731054774922664067575039 1u_{i-3}^2\\
&+ 879327994465025255384481 31u_{i-2}^2 - 346770742441690854589241870u_{i-2}u_{i-1}\\
&+ 348437796897631753060606543u_{i-1}^2 + 420998749597335849842664020u_{i-2}u_i\\
&- 861675614228526685698528040u_{i-1}u_i + 542828069875679609578330025u_i^2\\
&- 322534893789595815857580562u_{i-2}u_{i+1} + 671504849402026017193551778u_{i-1}u_{i+1}\\
&- 861675614228526685698528040u_iu_{i+1} + 348437796897631753060606543 * u_{i+1}^2\\
&+ 152497445486154126541234954u_{i-2}u_{i+2} - 322534893789595815857580562u_{i-1}u_{i+2}\\
&+ 420998749597335849842664020u_iu_{i+2} - 346770742441690854589241870u_{i+1}u_{i+2}\\
&+ 879327994465025255384481 31u_{i+2}^2 - 407489491577688276536913 22u_{i-2}u_{i+3}\\
&+ 874507730896306211178472 10u_{i-1}u_{i+3} - 115964286531853100210189720u_iu_{i+3}\\
&+ 971772905559671388466163 42u_{i+1}u_{i+3} - 502116860594574792984604 54u_{i+2}u_{i+3}\\
&+ 731054774922664067575039 1u_{i+3}^2 + u_{i-3}(-502116860594574792984604 54u_{i-2}\\
&+ 971772905559671388466163 42u_{i-1} - 115964286531853100210189720u_i\\
&+ 874507730896306211178472 10u_{i+1} - 407489491577688276536913 22u_{i+2}\\
&+ 107406125183364697543711 54u_{i+3} - 122758196183122420016952 7u_{i+4})\\
&- 11111u_{i-4}(165355769190611079815u_{i-3} - 5567222922425786917777u_{i-2}\\
&+ 105853436307630587240 3u_{i-1} - 124318974776026698656 5u_i\\
&+ 923938536129339626101u_{i+1} - 424690494183694671475u_{i+2}\\
&+ 110483481399624174257u_{i+3} - 124652574659982795 83u_{i+4})\\
&+ 471873608087503149475872 5u_{i-2}u_{i+4} - 102658810749330925856082 11u_{i-1}u_{i+4}\\
&+ 138130812873643264877237 15u_iu_{i+4} - 117613753081408345482697 33u_{i+1}u_{i+4}\\
&+ 618574139114291844342024 7u_{i+2}u_{i+4} - 183726795147687970782446 5u_{i+3}u_{i+4}\\
&+ 1180230306475238657612 68u_{i+4}^2).
\end{aligned}
$$

Data availability

Data will be made available on request.

References

[1] R. Abedian, H. Adibi, M. Dehghan, A high-order weighted essentially non-oscillatory (WENO) finite difference scheme for nonlinear degenerate parabolic equations, Comput. Phys. Commun. 184 (2013) 1874–1888.

[2] R. Abedian, M. Dehghan, A RBF-WENO finite difference scheme for non-linear degenerate parabolic equations, J. Sci. Comput. 93 (2022) 60.

[3] M. Ahmat, J. Qiu, Hybrid HWENO method for nonlinear degenerate parabolic equations, J. Sci. Comput. 96 (2023) 83.

[4] T. Arbogast, C.S. Huang, X. Zhao, Finite volume WENO schemes for nonlinear parabolic problems with degenerate diffusion on non-uniform meshes, J. Comput. Phys. 399 (2019) 108921.

[5] D. Aregba-Driollet, R. Natalini, S. Tang, Explicit diffusive kinetic schemes for nonlinear degenerate parabolic systems, Math. Comput. 73 (2004) 63–94.

[6] D.G. Aronson, The porous medium equation, in: Nonlinear Diffusion Problems, in: Lecture Notes in Math., vol. 1224, Springer, Berlin, 1986, pp. 1–46.

[7] U. Ascher, S. Ruuth, B. Wetton, Implicit-explicit methods for time-dependent PDE's, SIAM J. Numer. Anal. 32 (1995) 797–823.

[8] D.S. Balsara, C.-W. Shu, Monotonicity preserving weighted essentially nonoscillatory schemes with increasingly high order of accuracy, J. Comput. Phys. 160 (2000) 405–452.

[9] M. Bessemoulin-Chatard, F. Filbet, A finite volume scheme for nonlinear degenerate parabolic equations, SIAM J. Sci. Comput. 34 (2012) B559–B583.

[10] G. Beylkin, J.M. Keiser, L. Vozovoi, A new class of time discretization schemes for the solution of nonlinear PDEs, J. Comput. Phys. 147 (1998) 362–387.
[11] R. Borges, M. Carmona, B. Costa, W.S. Don, An improved weighted essentially non-oscillatory scheme for hyperbolic conservation laws, J. Comput. Phys. 227 (2008) 3191–3211.
[12] S. Boscarino, J. Qiu, G. Russo, T. Xiong, High order semi-implicit WENO schemes for all-Mach full Euler system of gas dynamics, SIAM J. Sci. Comput. 44 (2022) B368–B394.
[13] M. Castro, B. Costa, W.S. Don, High order weighted essentially non-oscillatory WENO-Z schemes for hyperbolic conservation laws, J. Comput. Phys. 230 (2011) 1766–1792.
[14] F. Cavalli, G. Naldi, G. Puppo, M. Semplice, High-order relaxation schemes for nonlinear degenerate diffusion problems, SIAM J. Numer. Anal. 45 (2007) 2098–2119.
[15] S. Chen, Y.-T. Zhang, Krylov implicit integration factor methods for spatial discretization on high dimensional unstructured meshes: application to discontinuous Galerkin methods, J. Comput. Phys. 230 (2011) 4336–4352.
[16] A. Christlieb, W. Guo, Y. Jiang, H. Yang, Kernel based high order "explicit" unconditionally stable scheme for nonlinear degenerate advection-diffusion equations, J. Sci. Comput. 82 (2020) 52.
[17] S.M. Cox, P.C. Matthews, Exponential time differencing for stiff systems, J. Comput. Phys. 176 (2002) 430–455.
[18] Q. Du, L. Ju, X. Li, Z. Qiao, Maximum bound principles for a class of semilinear parabolic equations and exponential time-differencing schemes, SIAM Rev. 63 (2021) 317–359.
[19] S. Gottlieb, D.I. Ketcheson, C.-W. Shu, High order strong stability preserving time discretizations, J. Sci. Comput. 38 (2009) 251–289.
[20] S. Gottlieb, D.I. Ketcheson, C.-W. Shu, Strong Stability Preserving Runge-Kutta and Multistep Time Discretizations, World Scientific, 2011.
[21] S. Gottlieb, J.S. Mullen, S.J. Ruuth, A fifth order flux implicit WENO method, J. Sci. Comput. 27 (2006) 271–287.
[22] S. Gottlieb, C.-W. Shu, E. Tadmor, Strong stability-preserving high-order time discretization methods, SIAM Rev. 43 (2001) 89–112.
[23] A. Harten, B. Engquist, S. Osher, S.R. Chakravarthy, Uniformly high order essentially non-oscillatory schemes, III, J. Comput. Phys. 71 (1987) 231–303.
[24] A.K. Henrick, T.D. Aslam, J.M. Powers, Mapped weighted essentially non-oscillatory schemes: achieving optimal order near critical points, J. Comput. Phys. 207 (2005) 542–567.
[25] M. Hochbruck, A. Ostermann, Exponential integrators, Acta Numer. 19 (2010) 209–286.
[26] M. Hochbruck, A. Ostermann, J. Schweitzer, Exponential Rosenbrock-type methods, SIAM J. Numer. Anal. 47 (2009) 786–803.
[27] G. Huang, Y. Xing, T. Xiong, High order well-balanced asymptotic preserving finite difference WENO schemes for the shallow water equations in all Froude numbers, J. Comput. Phys. 463 (2022) 111255.
[28] Y. Jiang, High order finite difference multi-resolution WENO method for nonlinear degenerate parabolic equations, J. Sci. Comput. 86 (16) (2021) 1–20.
[29] Y. Jiang, C.-W. Shu, M. Zhang, An alternative formulation of finite difference weighted ENO schemes with Lax-Wendroff time discretization for conservation laws, SIAM J. Sci. Comput. 35 (2013) A1137–A1160.
[30] G. Jiang, C.-W. Shu, Efficient implementation of weighted ENO schemes, J. Comput. Phys. 126 (1996) 202–228.
[31] T. Jiang, Y.-T. Zhang, Krylov single-step implicit integration factor WENO methods for advection–diffusion–reaction equations, J. Comput. Phys. 311 (2016) 22–44.
[32] A.K. Kassam, L.N. Trefethen, Fourth-order time-stepping for stiff PDEs, SIAM J. Sci. Comput. 26 (2005) 1214–1233.
[33] C.A. Kennedy, M.H. Carpenter, Additive Runge–Kutta schemes for convection-diffusion-reaction equations, Appl. Numer. Math. 44 (2003) 139–181.
[34] A. Kurganov, E. Tadmor, New high-resolution central schemes for nonlinear conservation laws and convection-diffusion equations, J. Comput. Phys. 160 (2000) 241–282.
[35] D. Levy, G. Puppo, G. Russo, Central WENO schemes for hyperbolic systems of conservation laws, Math. Model. Numer. Anal. 33 (1999) 547–571.
[36] J. Li, Z. Du, A two-stage fourth order time-accurate discretization for Lax–Wendroff type flow solvers I. Hyperbolic conservation laws, SIAM J. Sci. Comput. 38 (2016) A3046–A3069.
[37] L. Li, J. Zhu, Y.-T. Zhang, Absolutely convergent fixed-point fast sweeping WENO methods for steady state of hyperbolic conservation laws, J. Comput. Phys. 443 (2021) 110516.
[38] S. Liu, X. Liu, Exponential time differencing method for a reaction-diffusion system with free boundary, Commun. Appl. Math. Comput. 6 (2024) 354–371.
[39] X.-D. Liu, S. Osher, T. Chan, Weighted essentially non-oscillatory schemes, J. Comput. Phys. 115 (1994) 200–212.
[40] Y. Liu, Y. Cheng, S. Chen, Y.-T. Zhang, Krylov implicit integration factor discontinuous Galerkin methods on sparse grids for high dimensional reaction-diffusion equations, J. Comput. Phys. 388 (2019) 90–102.
[41] Y. Liu, C.-W. Shu, M. Zhang, High order finite difference WENO schemes for nonlinear degenerate parabolic equations, SIAM J. Sci. Comput. 33 (2011) 939–965.
[42] Y. Liu, Y.-T. Zhang, A robust reconstruction for unstructured WENO schemes, J. Sci. Comput. 54 (2013) 603–621.
[43] D. Lu, Y.-T. Zhang, Krylov integration factor method on sparse grids for high spatial dimension convection-diffusion equations, J. Sci. Comput. 69 (2016) 736–763.
[44] V.T. Luan, J.A. Pudykiewicz, D.R. Reynolds, Further development of efficient and accurate time integration schemes for meteorological models, J. Comput. Phys. 376 (2019) 817–837.
[45] E. Magenes, R.H. Nochetto, C. Verdi, Energy error estimates for a linear scheme to approximate nonlinear parabolic problems, ESAIM: Math. Model. Numer. Anal. 21 (1987) 655–678.
[46] C. Moler, C. Van Loan, Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later, SIAM Rev. 45 (2003) 3–49.
[47] M. Muskat, The Flow of Homogeneous Fluids Through Porous Media, McGraw-Hill, New York, 1937.
[48] Q. Nie, Y.-T. Zhang, R. Zhao, Efficient semi-implicit schemes for stiff systems, J. Comput. Phys. 214 (2006) 521–537.
[49] C. Ngo, W. Huang, A study on moving mesh finite element solution of the porous medium equation, J. Comput. Phys. 331 (2017) 357–380.
[50] J. Niesen, W.M. Wright, Algorithm 919: a Krylov subspace algorithm for evaluating the φ-functions appearing in exponential integrators, ACM Trans. Math. Softw. 38 (2012) 1–19.
[51] R.H. Nochetto, A. Schmidt, C. Verdi, A posteriori error estimation and adaptivity for degenerate parabolic problems, Math. Comput. 69 (2000) 1–24.
[52] J. Qiu, C.-W. Shu, Finite difference WENO schemes with Lax–Wendroff-type time discretizations, SIAM J. Sci. Comput. 24 (2003) 2185–2198.
[53] C.-W. Shu, Total-variation-diminishing time discretizations, SIAM J. Sci. Stat. Comput. 9 (1988) 1073–1084.
[54] C.-W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes, J. Comput. Phys. 77 (1988) 439–471.
[55] C.-W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes, II, J. Comput. Phys. 83 (1989) 32–78.
[56] Q. Tao, Y. Liu, Y. Jiang, J. Lu, An oscillation free local discontinuous Galerkin method for nonlinear degenerate parabolic equations, Numer. Methods Partial Differ. Equ. 39 (2023) 3145–3169.
[57] M. Tokman, Efficient integration of large stiff systems of ODEs with exponential propagation iterative (EPI) methods, J. Comput. Phys. 213 (2006) 748–776.
[58] L.N. Trefethen, D. Bau III, Numerical Linear Algebra, SIAM, 1997.
[59] E. Tsybulnik, X. Zhu, Y.-T. Zhang, Efficient sparse-grid implementation of a fifth-order multi-resolution WENO scheme for hyperbolic equations, Commun. Appl. Math. Comput. 5 (2023) 1339–1364.
[60] A. Vijaywargiya, G. Fu, Two finite element approaches for the porous medium equation that are positivity preserving and energy stable, arXiv:2303.14216, 2023.
[61] Z. Xu, C.-W. Shu, Third order maximum-principle-satisfying and positivity-preserving Lax-Wendroff discontinuous Galerkin methods for hyperbolic conservation laws, J. Comput. Phys. 470 (2022) 111591.
[62] Z. Xu, C.-W. Shu, Local characteristic decomposition-free high-order finite difference WENO schemes for hyperbolic systems endowed with a coordinate system of Riemann invariants, SIAM J. Sci. Comput. 46 (2024) A1352–A1372.

[63] Z. Xu, C.-W. Shu, A high-order well-balanced alternative finite difference WENO (A-WENO) method with the exact conservation property for the shallow water equations, Adv. Water Resour. 196 (2025) 104898.
[64] N. Yamaleev, M. Carpenter, A systematic methodology for constructing high-order energy stable WENO schemes, J. Comput. Phys. 228 (2009) 4248–4272.
[65] P. Zhang, T. Xiong, High order implicit finite difference schemes with a semi-implicit WENO reconstruction for nonlinear degenerate parabolic equations, J. Comput. Phys. 467 (2022) 111442.
[66] Q. Zhang, Z.-L. Wu, Numerical simulation for porous medium equation by local discontinuous Galerkin finite element method, J. Sci. Comput. 38 (2009) 127–148.
[67] J. Zhu, C.-W. Shu, A new type of multi-resolution WENO schemes with increasingly higher order of accuracy, J. Comput. Phys. 375 (2018) 659–683.