

A Numerical Study of Riemann Problem Solutions for the Homogeneous One-Dimensional Shallow Water Equations

Pavlos Stampolidis, Maria Ch. Gousidou-Koutita

Department of Mathematics, Faculty of Sciences, Aristotle University, Thessaloniki, Greece
Email: pavloss@math.auth.gr, gousidou@math.auth.gr

How to cite this paper: Stampolidis, P. and Gousidou-Koutita, M.Ch. (2024) A Numerical Study of Riemann Problem Solutions for the Homogeneous One-Dimensional Shallow Water Equations. *Applied Mathematics*, 15, 765-817.
<https://doi.org/10.4236/am.2024.1511044>

Received: September 30, 2024

Accepted: November 18, 2024

Published: November 21, 2024

Copyright © 2024 by author(s) and Scientific Research Publishing Inc.
This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).
<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

The solution of the Riemann Problem (RP) for the one-dimensional (1D) non-linear Shallow Water Equations (SWEs) is known to produce four potential wave patterns for the scenario where the water depth is always positive. In this paper, we choose four test problems with exact solutions for the 1D SWEs. Each test problem is a RP with one of the four possible wave patterns as its solution. These problems are numerically solved using schemes from the family of Weighted Essentially Non-Oscillatory (WENO) methods. For comparison purposes, we also include results obtained from the Random Choice Method (RCM). This study has three main objectives. Firstly, we outline the procedures for the implementation of the methods employed in this paper. Secondly, we assess the performance of the schemes in conjunction with a second-order Total Variation Diminishing (TVD) flux on a variety of RPs for the 1D SWEs (for both short- and long-time simulations). Thirdly, we investigate if a single method yields optimal outcomes for all test problems. Optimal outcomes refer to numerical solutions devoid of spurious oscillations, exhibiting high resolution of discontinuities, and attaining high-order accuracy in the smooth parts of the solution.

Keywords

1D Shallow Water Equations, Finite Volume WENO Schemes, Multi-Resolution WENO Schemes, Random Choice Method, Riemann Problem

1. Introduction

The SWEs, commonly referred to as the Saint-Venant system, are extensively employed by scientists for the purpose of modeling and simulating a diverse range of

physical phenomena involving water flows with a free surface under the influence of gravity. The breaking of waves on shallow beaches, roll waves in open channels, tides in oceans, flood waves in rivers, surges and dam-break wave modelling are examples of such phenomena (see [1]). The equations may also be reinterpreted and applied to the modelling of atmospheric flows. According to [2], the SWEs are an approximation to the full free surface problems and can be derived from the Navier-Stokes equations. Additional details regarding the SWEs can be found in the textbooks [1]-[3] and the references mentioned therein.

The SWEs that arise are a time-dependent system of non-linear Partial Differential Equations (**PDEs**). The governing equations are predominantly hyperbolic and belong to the broader class of hyperbolic systems commonly referred to as Hyperbolic Conservation Laws (**HCLs**), *i.e.*, a (hyperbolic) system of PDEs that (in one space dimension) may be expressed in the form

$$\partial_t \mathbf{U} + \partial_x \mathbf{F}(\mathbf{U}) = \mathbf{0}, \quad (1)$$

where $\mathbf{U} = [u_1, u_2, \dots, u_n]^\top$ is called the vector of conserved variables and $\mathbf{F}(\mathbf{U}) = [f_1, f_2, \dots, f_n]^\top$ is the vector of fluxes with $f_i = f_i(u_1, u_2, \dots, u_n)$, $i = 1, \dots, n$. Hyperbolicity and non-linearity mean that even for smooth Initial Conditions (**ICs**), the HCLs admit solutions that include discontinuities [4] [5]. The most challenging aspects for numerical methods are these discontinuous features. Over the past few decades, one of the most significant obstacles in numerical analysis has been the creation of numerical schemes that can approximate discontinuous solutions accurately.

There are essentially two main ways to computing solutions that involve discontinuities: the shock-fitting approach and the shock-capturing method approach. One category of effective shock-capturing numerical methods is referred to as high-resolution methods. The numerical solutions generated by these methods are oscillation-free in the vicinity of discontinuities and maintain second or higher order accuracy in smooth parts of the solution (see [1] [2]). The methods tested in this study are high-resolution methods. What follows is their concise overview.

The computational scheme known as the RCM was initially proposed by Chorin [6]. Its foundation lies in a Glimm-developed [7] approximation method utilised to construct solutions for non-linear hyperbolic systems of conservation laws. The primary benefit of the RCM is that it captures discontinuities with an infinite resolution; conversely, its accuracy in the smooth parts of the flow is poor. There have been numerous contributions to the development of the RCM since its inception. This article applies a specific version of RCM, known as the non-staggered grid version (see [2]).

The utilisation of WENO schemes to approximate solutions to HCLs is widespread. After extensive research since the 1990s, they have generally been found to be extremely stable and robust. WENO schemes are, in essence, an approximation method designed to resolve discontinuities precisely and essentially in a non-oscillatory manner while attaining arbitrarily high-order accuracy in smooth

regions. Initialization of the classical WENO approximation method is described in [8], and it is generalised in [9]. Following the publication of Jiang and Shu [9], there have been a variety of improvements and applications of WENO methods in simulating various physical and engineering problems. One can find a comprehensive collection of applications of WENO schemes in [10].

Using the WENO procedure, researchers have developed both Finite Volume (**FV**) and Finite Difference (**FD**) schemes. The WENO schemes that are considered in this work are: 1) the classical FV WENO scheme of Shu [11], which is an extension of the original FD WENO scheme of Jiang and Shu [9], abbreviated as WENO-JS in this paper, 2) the WENO-M scheme of Henrick *et al.* [12], 3) the WENO-Z scheme of Borges *et al.* [13], 4) the WENO-NS scheme of Ha *et al.* [14], 5) the WENO-P scheme of Kim *et al.* [15], 6) the WENO-ZQ scheme of Zhu and Qiu [16], 7) the MWENO-P scheme of Rathan and Raju [17], 8) the WENO scheme of Zhu and Shu [18], abbreviated as WENO-ZS in this paper, and 9) the WENO-ZR scheme of Gu *et al.* [19]. The original publications introduced the WENO-M, WENO-Z, WENO-NS, WENO-P, MWENO-P and WENO-ZR schemes within the FD framework. In this paper, these schemes are effortlessly extended to the FV framework by emulating the approach taken in [11].

Within the framework of FV WENO methods, Titarev and Toro suggest in [20] replacing first-order monotone fluxes with second-order TVD fluxes, thereby constructing a TVD modification of FV WENO methods. One of their work's proposals was to employ the upwind TVD WAF flux [1] [21] [22] as the building block, which they applied to the FV WENO-JS. The WENO schemes utilised in this research operate within the framework of FV methods applying the TVD WAF flux. In order to simplify the discussion in this paper, we will refer to FV WENO schemes that are equipped with the TVD WAF flux as simply WENO schemes.

The manuscript is organized as follows. In Section 2, we present the mathematical elements required to describe the SWEs, as well as the terminology and basic notation that will be used throughout the paper. In Section 3, we describe the FV discretization and provide the procedures for applying the numerical methods discussed in this research. In Section 4, we compare the numerical and computational performance of the methods for solving RPs of SWEs. Concluding remarks are given in Section 5. To ensure completeness, we have included additional supplementary details of this study in the **Appendices**, some of which, to our knowledge, are not available in literature.

Throughout this paper, vectors and matrices are represented using bold face letters, while scalars are represented using plain letters. All vector-related operations are carried out component-wise.

2. General Background and Notation

2.1. Governing Equations

The 1D SWEs are obtained by assuming an incompressible fluid in a channel of

unit width with negligible vertical velocity and almost constant horizontal velocity across any cross section of the channel. This assertion remains valid by considering small-amplitude waves in a fluid that is relatively shallow in comparison to its wavelength (see [3]). In differential form, the conservative formulation of the homogeneous 1D SWEs is

$$\partial_t \mathbf{U} + \partial_x \mathbf{F}(\mathbf{U}) = 0, \tag{2}$$

where \mathbf{U} and $\mathbf{F}(\mathbf{U})$ are the vectors of conserved variables and fluxes, given respectively by

$$\mathbf{U} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} h \\ hu \end{bmatrix}, \mathbf{F}(\mathbf{U}) = \begin{bmatrix} f_1(u_1, u_2) \\ f_2(u_1, u_2) \end{bmatrix} = \begin{bmatrix} u_2 \\ \frac{u_2^2}{u_1} + \frac{1}{2} g u_1^2 \end{bmatrix} = \begin{bmatrix} hu \\ hu^2 + \frac{1}{2} g h^2 \end{bmatrix}, \tag{3}$$

where $u_i = u_i(x, t)$, $i = 1, 2$, are conserved variables, while $h = h(x, t)$ and $u = u(x, t)$ are primitive variables. These variables are functions of space x and time t . Equations (2)-(3) express the physical laws of conservation of mass and momentum, where h denotes the water depth, u is the horizontal velocity of the fluid, and $g \approx 9.81 \text{ m/s}^2$ is the gravitational acceleration. In shallow water theory, the quantity hu is frequently referred to as the discharge because it quantifies the water flow rate past a point.

2.2. Eigenstructure of the System

For smooth solution, the SWEs (2)-(3) can alternatively be expressed in quasi-linear form

$$\partial_t \mathbf{U} + \mathbf{A}(\mathbf{U}) \partial_x \mathbf{U} = \mathbf{0},$$

where the coefficient matrix $\mathbf{A}(\mathbf{U})$ is the Jacobian matrix

$$\begin{aligned} \mathbf{A}(\mathbf{U}) &= \frac{\partial \mathbf{F}}{\partial \mathbf{U}} = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \frac{\partial f_1}{\partial u_2} \\ \frac{\partial f_2}{\partial u_1} & \frac{\partial f_2}{\partial u_2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\left(\frac{u_2}{u_1}\right)^2 + g u_1 & \frac{2u_2}{u_1} \end{bmatrix} \\ &= \begin{bmatrix} 0 & 1 \\ -u^2 + gh & 2u \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -u^2 + c^2 & 2u \end{bmatrix}, \end{aligned} \tag{4}$$

where $c = \sqrt{gh}$ represents the speed of a shallow water wave, which is often referred to as celerity. The eigenvalues of $\mathbf{A}(\mathbf{U})$ are

$$\lambda_1 = \frac{u_2}{u_1} - \sqrt{g u_1} = u - c, \lambda_2 = \frac{u_2}{u_1} + \sqrt{g u_1} = u + c, \tag{5}$$

with the corresponding right eigenvectors

$$\mathbf{R}^{(1)} = \begin{bmatrix} 1 \\ \lambda_1 \end{bmatrix} = \begin{bmatrix} 1 \\ u - c \end{bmatrix}, \mathbf{R}^{(2)} = \begin{bmatrix} 1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} 1 \\ u + c \end{bmatrix}. \tag{6}$$

The matrix having columns that are eigenvectors (6) is denoted as

$$\mathbf{R} = \mathbf{R}(u, c) = \begin{bmatrix} 1 & 1 \\ u - c & u + c \end{bmatrix}. \tag{7}$$

Then, the rows of the subsequent matrix

$$L = R^{-1}(u, c) = \begin{bmatrix} \frac{u+c}{2c} & \frac{-1}{2c} \\ -\frac{u+c}{2c} & \frac{1}{2c} \end{bmatrix}, \tag{8}$$

are left eigenvectors of $A(U)$.

The SWEs are hyperbolic, *i.e.*, the System (2)-(3) of 2 equations has 2 real eigenvalues. If the celerity c remains positive, the equations are strictly hyperbolic, meaning that the eigenvalues are real and distinct.

2.3. The Riemann Problem for the SWEs

The subsequent notations are in accordance with those in [2]. The RP for (2)-(3) is the initial value problem for (2)-(3) with Initial Conditions (ICs):

$$U(x, 0) = \begin{cases} U_L = [u_{1L}, u_{2L}]^T & \text{if } x < x_0, \\ U_R = [u_{1R}, u_{2R}]^T & \text{if } x > x_0, \end{cases} \tag{9}$$

and is represented by the notation RP (U_L, U_R) . Subscripts L and R indicate the left and right states, respectively, of the piecewise constant data with a single jump discontinuity at some point, say $x = x_0$. It is possible to solve the initial value problem (2), (3), (9) exactly. **Figure A1** in **Appendix A** depicts the structure of the solution to the RP (2), (3), (9) in the $x-t$ plane, in the manner indicated in [23].

The solution is a similarity solution, *i.e.*, a function only dependent on the ratio x/t . Two waves, one corresponding to each eigenvalue $\lambda_i, i = 1, 2$, partition it into three constant states. These waves propagate away from the point of the initial discontinuity at constant speeds. The solution to the left of the λ_1 -wave is equal to the initial data U_L , whereas the solution to the right of the λ_2 -wave is equal to U_R . The region bounded by the λ_1 and λ_2 waves is commonly referred to as the Star Region, and the denotation U^* represents the solution in that area.

There are two types of waves: shock waves and rarefaction waves. Within the context of shallow water theory, shock and rarefaction waves are alternatively referred to as bore and depression waves, respectively. In the following discussion, shock waves and rarefaction waves shall be referred to as shocks and rarefactions, respectively. **Figure A2** of **Appendix A**, also available in [23], illustrates each of the four potential wave patterns. At any given time moment, the solution varies continuously with x via rarefactions, whereas it jumps discontinuously via shocks.

When solving the RP (2), (3), (9), the speeds of the head and tail of a left or right rarefaction will be designated as S_{HL}, S_{TL}, S_{HR} , and S_{TR} , respectively. Whereas the speeds of a left or right shock will be represented as S_{SL} and S_{SR} , correspondingly. The formulas needed to compute these speeds are available in [1]. If rarefactions are present, the solution within the left or right rarefaction, denoted as U_{Lfan} and U_{Rfan} , respectively, is given by

$$U_{Lfan}(x, t) = \left[\frac{(2c_L + u_L - S)^2}{9g}, \frac{(u_L + 2S + 2c_L)(2c_L + u_L - S)^2}{27g} \right]^T, \tag{10}$$

$$\mathbf{U}_{Rfan}(x, t) = \left[\frac{(2c_R - u_R + S)^2}{9g}, \frac{(u_R + 2S - 2c_R)(2c_R - u_R + S)^2}{27g} \right]^T, \quad (11)$$

where $S = (x - x_0)/t$, $u_K = u_{2K}/u_{1K}$, $c_K = \sqrt{gu_{1K}}$, $K = L, R$. See references [1]-[4] for specifics regarding the RP solutions. For SWEs, the exact solution to the RP is provided in [1] [3].

3. Numerical Methods

Prior to presenting the procedures of the schemes, two essential topics need to be addressed: the construction of Finite Volume (FV) methods for Hyperbolic Conservation Laws (HCLs) and the selection of the time step Δt .

3.1. The Finite Volume Framework

Consider the 1D HCLs (1). The computational domain is discretized into grid cells, or FVs $I_i = \left[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}} \right]$. In the $x-t$ plane, assume a control volume $V = I_i \times [t^n, t^{n+1}]$ with dimensions $\Delta x = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}$ and $\Delta t = t^{n+1} - t^n$. One approach is to formulate entirely discrete one-step schemes, while the alternative is to develop semi-discrete schemes. The latter approach is employed to execute the WENO schemes.

By maintaining the time variable t continuous while integrating (1) over the volume V , the subsequent system of Ordinary Differential Equations (ODEs) is derived:

$$\frac{d}{dt} \bar{U}_i(t) = -\frac{1}{\Delta x} \left(F_{i+\frac{1}{2}} - F_{i-\frac{1}{2}} \right),$$

where $\bar{U}_i(t)$ represents the spatial-integral average of the solution U in the cell I_i at time t , commonly known as the cell average, while $F_{i+\frac{1}{2}}$ represents the physical flux at the cell boundary, or cell interface $x = x_{i+\frac{1}{2}}$ and time t , that

is

$$\bar{U}_i(t) = \frac{1}{\Delta x} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} U(x, t) dx, \quad F_{i+\frac{1}{2}} = F \left(U \left(x_{i+\frac{1}{2}}, t \right) \right), \quad (12)$$

where $U \left(x_{i+\frac{1}{2}}, t \right)$ is the solution of (1) at the cell interface $x_{i+\frac{1}{2}}$. In consideration of the subsequent approximations, semi-discrete FV schemes may be formulated.

Initially, $F_{i+\frac{1}{2}}$ must be estimated due to the fact that $U \left(x_{i+\frac{1}{2}}, t \right)$ varies with time along each cell interface and the exact solution is unavailable. Hence, the FV numerical scheme is expressed as

$$\frac{d}{dt} \bar{U}_i(t) = -\frac{1}{\Delta x} \left(\hat{f}_{i+\frac{1}{2}} - \hat{f}_{i-\frac{1}{2}} \right) = L_i(U), \tag{13}$$

where $\hat{f}_{i+\frac{1}{2}}$ is an approximation to $F_{i+\frac{1}{2}}$, known as the numerical flux or the building block. The numerical flux at the cell interface $x_{i+\frac{1}{2}}$ is defined as a monotone function of two values,

$$\hat{f}_{i+\frac{1}{2}} = H \left(U_{i+\frac{1}{2}}^-, U_{i+\frac{1}{2}}^+ \right) \tag{14}$$

which are referred to as the left and right boundary extrapolated values denoted as $U_{i+\frac{1}{2}}^-$ and $U_{i+\frac{1}{2}}^+$, respectively. These extrapolated values $U_{i+\frac{1}{2}}^-$ and $U_{i+\frac{1}{2}}^+$ are approximations to $U \left(x_{i+\frac{1}{2}}, t \right)$ from the left and right, respectively. They are derived from cell averages via a high-order polynomial reconstruction that is essentially non-oscillatory.

Additionally, in order to advance the solution in time, a numerical method is employed to solve the system of ODEs (13). Typically, TVD Runge-Kutta methods are applied to maintain a consistently high-order of time accuracy. This work employs the subsequent third-order TVD Runge-Kutta method [24]:

$$U_i^{(1)} = U_i^n + \Delta t L_i(U_i^n), \tag{15}$$

$$U_i^{(2)} = \frac{3}{4} U_i^n + \frac{1}{4} U_i^{(1)} + \frac{1}{4} \Delta t L_i(U_i^{(1)}), \tag{16}$$

$$U_i^{n+1} = \frac{1}{3} U_i^n + \frac{2}{3} U_i^{(2)} + \frac{2}{3} \Delta t L_i(U_i^{(2)}), \tag{17}$$

where U_i^n is an approximation to the cell average $\bar{U}_i(t^n)$ in (12) at time t^n . The evaluation of the cell averages U_i^0 should be conducted using the exact integration of the suitable ICs, that is

$$U_i^0 = \bar{U}_i(0) = \frac{1}{\Delta x} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} U(x, 0) dx, \tag{18}$$

where $U(x, 0)$ are the ICs.

3.2. Time Step

The subsequent condition is applied to the time step Δt for every method examined in this study

$$\Delta t = C_{\text{CFL}} \frac{\Delta x}{S_{\text{max}}^n}, \tag{19}$$

where $C_{\text{CFL}} \in (0, 1]$ denotes the Courant-Friedrichs-Lewy (CFL) coefficient, Δx signifies the spatial step and S_{max}^n represents the maximal wave speed detected across the domain at time level $t = t^n$. The computation of S_{max}^n in this study is

performed using the equation

$$S_{\max}^n = \max_i \left\{ \left| S_{i+\frac{1}{2}}^L \right|, \left| S_{i+\frac{1}{2}}^R \right| \right\}, \tag{20}$$

where $S_{i+\frac{1}{2}}^L$ and $S_{i+\frac{1}{2}}^R$ denote the wave speeds of the left and right non-linear waves, respectively, in the solution of the RP (U_i^n, U_{i+1}^n) , which is computed using an exact Riemann solver. The speed of the head is chosen for rarefactions, while the shock speed is chosen for shocks. Refer to [2] for further information.

3.3. Procedures of the Methods

The spatial domain $[0, L]$ is discretized into M computing cells

$$I_i = \left[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}} \right] \text{ with a dimension of } x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}} = \Delta x = \frac{L}{M}, \text{ with } i = 1, \dots, M.$$

The procedures developed for the methods examined in this paper (except the RCM) are structured as follows:

1) Initial conditions. ICs are established in accordance with (18). Using the cell averages $\{U_i^n\}$ as input, our objective is to advance the solution from its starting values $\{U_i^n\}$ at time t^n to new values $\{U_i^{n+1}\}$ at time $t^{n+1} = t^n + \Delta t$, with a time step of size Δt . To achieve this, execute the subsequent steps.

2) Boundary Conditions (BCs). BCs were implemented through the application of the periodicity condition. Whenever necessary, we positioned r fictitious cells adjacent to each boundary, with the value of r varying based on the method. For the left boundary, the fictitious cells are denoted by $i = 0, -1, -2, \dots$ and for the right boundary they are denoted by $i = M + 1, M + 2, \dots$. Then for $j = 0, 1, \dots, r$ apply

$$U_{0-j}^n = U_{1+j}^n \text{ (Left BCs)} \tag{21}$$

$$U_{M+1+j}^n = U_{M-j}^n \text{ (Right BCs)} \tag{22}$$

3) Computation of time step. The time step in each method was determined in accordance with subsection 3.2. At the outset of the computations, a choice of C_{CFL} must be made, which is contingent upon the method. We employ $C_{\text{CFL}} = 0.45$ for the RCM, as recommended in [1]. For the WENO TVD schemes, the Courant-Friedrichs-Lewy (CFL) coefficient is usually within the range of 0.4 to 0.6. When necessary, the C_{CFL} was adjusted to a lower value for the first few steps.

4) Computation of the numerical flux. This step varies based on the method used. We will cover the process of calculating the numerical flux for each method in the subsequent sections.

5) Updating of solution and advancing to the next time level. The solution is updated to the new time level $n + 1$ by employing (15)-(17). After that, proceed to step 2 using $\{U_i^{n+1}\}$ as starting values.

Details regarding the value of r , as well as the numerical flux computation, are provided in the subsequent sections for each method.

3.3.1. WENO TVD Schemes

To have a thorough understanding of the following topics, consult [10] [11]. The fundamental concept of most FV WENO methods is as follows: based on the cell averages $\{\bar{u}_j\}$ of a piecewise smooth function $u(x)$, determine k ($k=1,2,\dots$) reconstruction polynomials $p_i^{(k,s)}(x)$ over the stencils $T_s(i) = \{I_{i-s}, \dots, I_{i-s+k-1}\}$, $s = 0, 1, \dots, k-1$, of degree at most $k-1$ at each cell I_i that satisfy

$$\frac{1}{\Delta x} \int_{I_j} p_i^{(k,s)}(\xi) d\xi = \bar{u}_j, \quad j = i-s, \dots, i-s+k-1; \quad s = 0, 1, \dots, k-1.$$

Following this, a convex combination of the k reconstruction polynomials $p_i^{(k,s)}(x)$ is employed to derive $(2k-1)$ -th order approximations to the extrapolated values $u_{i+\frac{1}{2}}^\pm$ according to

$$u_{i+\frac{1}{2}}^- = \sum_{s=0}^{k-1} \omega_s p_i^{(k,s)}\left(x_{i+\frac{1}{2}}\right), u_{i-\frac{1}{2}}^+ = \sum_{s=0}^{k-1} \tilde{\omega}_s p_i^{(k,s)}\left(x_{i-\frac{1}{2}}\right) \Rightarrow u_{i+\frac{1}{2}}^+ = \sum_{s=0}^{k-1} \tilde{\omega}_s p_{i+1}^{(k,s)}\left(x_{i+\frac{1}{2}}\right),$$

where the coefficients ω_s and $\tilde{\omega}_s$ are referred to as nonlinear weights, and their calculation is specified for each WENO scheme in the subsequent paragraphs.

Subsequently, insert them into a numerical flux $\hat{f}\left(u_{i+\frac{1}{2}}^-, u_{i+\frac{1}{2}}^+\right)$ to generate a scheme that is analogous to (13) in the scalar case, resulting in $(2k-1)$ -th order accuracy in smooth regions. To simplify notation, we denote the $(2k-1)$ -th order WENO scheme as WENO $(2k-1)$.

For the purpose of demonstration, when $k=3$, each cell I_i provides three reconstruction quadratic polynomials, $p_i^{(3,0)}(x)$, $p_i^{(3,1)}(x)$, and $p_i^{(3,2)}(x)$, over the stencils $T_0(i) = \{I_i, I_{i+1}, I_{i+2}\}$, $T_1(i) = \{I_{i-1}, I_i, I_{i+1}\}$, and $T_2(i) = \{I_{i-2}, I_{i-1}, I_i\}$, respectively. These polynomials must adhere to the following conditions

$$\frac{1}{\Delta x} \int_{I_j} p_i^{(3,s)}(\xi) d\xi = \bar{u}_j, \quad j = i-s, \dots, i-s+2; \quad s = 0, 1, 2.$$

After that, the fifth-order reconstruction is obtained by

$$u_{i+\frac{1}{2}}^- = \sum_{s=0}^2 \omega_s p_i^{(3,s)}\left(x_{i+\frac{1}{2}}\right), \quad u_{i+\frac{1}{2}}^+ = \sum_{s=0}^2 \tilde{\omega}_s p_{i+1}^{(3,s)}\left(x_{i+\frac{1}{2}}\right),$$

An essential part of the WENO procedure is the computation of what is commonly known as Smooth or Smoothness Indicators (**SI**s). Denoted as β_s^{JS} , these indicators are defined as [9]

$$\beta_s^{JS} = \sum_{l=1}^m \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \Delta x^{2l-1} \left(\frac{d^l p_i^{(k,s)}(x)}{dx^l} \right)^2 dx, \quad s = 0, 1, \dots, k-1; \quad m = k-1. \quad (23)$$

They can be found explicitly in [11] for $k=2,3$, and in [25] for $k=4,5$. Apart from that, they are included in **Appendix C**.

The reconstruction for systems can be handled by applying the WENO reconstruction procedure to each individual component of the vector U . Nevertheless, spurious oscillations may result from the component-wise WENO reconstruction. To prevent this, the reconstruction is conducted using local characteristic

variables. The characteristic reconstruction process involves the initial transformation of the conservative variables to characteristic variables, followed by the application of the WENO reconstruction procedure to each component of these variables. The final values are acquired through the process of transforming back to conservative variables. In this study, we consistently implement reconstruction in local characteristic variables.

1) WENO (2k - 1)-JS TVD Scheme (k = 2, 3, 4, 5)

The WENO3-JS, WENO5-JS, WENO7-JS, and WENO9-JS schemes are obtained for $k = 2, 3, 4, 5$, respectively. The WENO schemes have linear stability for $C_{CFL} \leq 1$ [9]. For solutions exhibiting discontinuities, reduced CFL coefficients are generally employed, typically within the range of $C_{CFL} = 0.4 - 0.6$. Following [20], we use $C_{CFL} = 0.4$.

Procedure 1 WENO (2k - 1)-JS TVD Scheme, k = 2, 3, 4, 5

Step 1.1. For $i = 1, \dots, M$ set the ICs using (18).

Step 1.2. Use the cell averages $\{U_i^n\}_{i=1}^{i=M}$, $n \geq 0$, as input.

Step 2. Apply the BCs as specified in (21)-(22) with $r = k$.

Step 3.1. For $i = 0, \dots, M$ transform the vector of conserved variables $U_i^n = [u_{1i}^n, u_{2i}^n]^T$ to the vector of primitive variables $UP_i^n = [h_i^n, u_i^n]^T = [u_{1i}^n, u_{2i}^n / u_{1i}^n]^T$.

Step 3.2. For $i = 0, \dots, M$ solve RP (UP_i^n, UP_{i+1}^n) , utilising an exact Riemann solver [1], to compute S_{max}^n using (20) and then calculate Δt according to (19).

Step 4.1. For $i = -1, \dots, M + 1$

Step 4.1.1. Utilise (7) and (8) to calculate $R = R(\tilde{u}, \tilde{c})$ and $L = L(\tilde{u}, \tilde{c})$, where \tilde{u} and \tilde{c} denote the Roe average [26] between U_i and U_{i+1} which are defined as

$$\tilde{u} = \frac{u_i \sqrt{h_i} + u_{i+1} \sqrt{h_{i+1}}}{\sqrt{h_i} + \sqrt{h_{i+1}}}, \quad \tilde{c} = \sqrt{\frac{1}{2}(c_i^2 + c_{i+1}^2)}, \quad c_i = \sqrt{gh_i}.$$

Step 4.1.2. Transform into characteristic variables using

$$V_j = LU_j, \quad j = i - k + 1, \dots, i + k.$$

Step 4.1.3. To estimate the left extrapolated value $V_{i+\frac{1}{2}}^-$ calculate

Step 4.1.3.1. $p_i^{(k,s)} \left(x_{i+\frac{1}{2}} \right) = \sum_{j=0}^{k-1} c_{k,s,j} V_{i-s+j}, \quad s = 0, 1, \dots, k-1,$

Step 4.1.3.2. $\beta_s^{JS} = \beta_s^{JS}(V, i), \quad s = 0, 1, \dots, k-1,$ employing (30)-(33), contingent upon k .

Step 4.1.3.3. $\alpha_s^{JS} = \frac{d_{k,s}}{(\epsilon + \beta_s^{JS})^2}, \quad s = 0, 1, \dots, k-1,$

Step 4.1.3.4. $\omega_s^{JS} = \frac{\alpha_s^{JS}}{\sum_{j=0}^{k-1} \alpha_j^{JS}}, \quad s = 0, 1, \dots, k-1,$ (nonlinear weights).

Step 4.1.3.5. Finally, $V_{i+\frac{1}{2}}^- = \sum_{s=0}^{k-1} \omega_s^{JS} p_i^{(k,s)} \left(x_{i+\frac{1}{2}} \right).$

The constant ϵ is introduced to prevent the denominator from becoming

zero. The coefficients $c_{k,s,j}$ are provided in [11], whereas the values of $d_{k,s}$ can be located in [11] for $k = 2, 3$, and in [25] for $k = 4, 5$. **Table A1** and **Table A2** of **Appendix B** contain them, as well. In the literature, the coefficients $d_{k,s}$ are commonly referred to as ideal, optimal, or linear weights. For WENO-JS TVD scheme we take $\epsilon = 10^{-20}$ [20].

Step 4.1.4. To estimate the right extrapolated value $V_{i+\frac{1}{2}}^+$ calculate

Step 4.1.4.1. $p_{i+1}^{(k,s)} \left(x_{i+\frac{1}{2}} \right) = \sum_{j=0}^{k-1} c_{k,s-1,j} V_{i+1-s+j}, \quad s = 0, 1, \dots, k-1,$

Step 4.1.4.2. $\beta_s^{JS} = \beta_s^{JS}(V, i+1), \quad s = 0, 1, \dots, k-1,$ employing (30),

Step 4.1.4.3. $\alpha_s^{JS} = \frac{d_{k-1-s}}{(\epsilon + \beta_s^{JS})^2}, \quad s = 0, 1, \dots, k-1,$

Step 4.1.4.4. $\omega_s^{JS} = \frac{\alpha_s^{JS}}{\sum_{j=0}^{k-1} \alpha_j^{JS}}, \quad s = 0, 1, \dots, k-1.$

Step 4.1.4.5. Thus, $V_{i+\frac{1}{2}}^+ = \sum_{s=0}^{k-1} \omega_s^{JS} p_{i+1}^{(k,s)} \left(x_{i+\frac{1}{2}} \right).$

Step 4.1.5. Transform back into conservative variables

$$U_{i+\frac{1}{2}}^\pm = R V_{i+\frac{1}{2}}^\pm = \begin{bmatrix} u_{1i+\frac{1}{2}}^\pm \\ u_{2i+\frac{1}{2}}^\pm \end{bmatrix}^\top.$$

Step 4.1.6. Convert to primitive variables

$$UP_{i+\frac{1}{2}}^\pm = \begin{bmatrix} u_{1i+\frac{1}{2}}^\pm \\ u_{2i+\frac{1}{2}}^\pm / u_{1i+\frac{1}{2}}^\pm \end{bmatrix}^\top = \begin{bmatrix} h_{i+\frac{1}{2}}^\pm \\ u_{i+\frac{1}{2}}^\pm \end{bmatrix}^\top.$$

Step 4.1.7. Solve RP $(UP_{i+\frac{1}{2}}^-, UP_{i+\frac{1}{2}}^+)$ utilising an HLL Riemann [1] [27] solver and store the following values

$$c_{1,i+\frac{1}{2}} = S_{i+\frac{1}{2}}^L \frac{\Delta t}{\Delta x}, \quad c_{2,i+\frac{1}{2}} = S_{i+\frac{1}{2}}^R \frac{\Delta t}{\Delta x}, \quad \Delta h_{1,i+\frac{1}{2}} = \hat{h}^* - h_{i+\frac{1}{2}}^-, \tag{24}$$

$$\Delta h_{2,i+\frac{1}{2}} = h_{i+\frac{1}{2}}^+ - \hat{h}^*, \quad SF_{i+\frac{1}{2}} = F_{i+\frac{1}{2}}^- + F_{i+\frac{1}{2}}^+, \tag{25}$$

$$\Delta F_{2,i+\frac{1}{2}} = F_{i+\frac{1}{2}}^+ - \hat{F}^*, \quad \Delta F_{1,i+\frac{1}{2}} = \hat{F}^* - F_{i+\frac{1}{2}}^-, \tag{26}$$

where $F_{i+\frac{1}{2}}^- = F \left(U_{i+\frac{1}{2}}^- \right), F_{i+\frac{1}{2}}^+ = F \left(U_{i+\frac{1}{2}}^+ \right).$ The values \hat{h}^* and \hat{F}^* are estimates of the exact solutions for h^* and $F(U^*)$ in the star region, respectively. The two-rarefaction Riemann solver [1] is employed to calculate $\hat{h}^*.$

Step 4.2. For $i = 0, \dots, M$ calculate

$$4.2.1. \quad r_{j,i+\frac{1}{2}} = \begin{cases} \frac{\Delta h_{j,i-\frac{1}{2}}}{\Delta h_{j,i+\frac{1}{2}}} & \text{if } c_{j,i+\frac{1}{2}} > 0 \\ \frac{\Delta h_{j,i+\frac{3}{2}}}{\Delta h_{j,i+\frac{1}{2}}} & \text{if } c_{j,i+\frac{1}{2}} < 0 \end{cases}, \quad j = 1, 2 \tag{27}$$

and store the subsequent values

$$4.2.2. \hat{f}_{i+\frac{1}{2}} = \frac{1}{2} SF_{i+\frac{1}{2}} - \frac{1}{2} \sum_{j=1}^2 \text{sign} \left(c_{j,i+\frac{1}{2}} \right) \phi_{i+\frac{1}{2}} \left(r_{j,i+\frac{1}{2}}, c_{j,i+\frac{1}{2}} \right) \Delta F_{j,i+\frac{1}{2}}, \quad (28)$$

where $\phi = \phi(r, c)$ is a WAF limiter function that corresponds to the well-known flux limiter SUPERBEE of Roe [28] and is defined as

$$\phi(r, c) = \begin{cases} 1 & \text{if } r \leq 0, \\ 1 - 2(1 - |c|)r & \text{if } 0 < r < \frac{1}{2}, \\ |c| & \text{if } \frac{1}{2} \leq r \leq 1, \\ 1 - (1 - |c|)r & \text{if } 1 < r < 2, \\ 2|c| - 1 & \text{if } r \geq 2. \end{cases} \quad (29)$$

Step 4.3. For $i = 1, \dots, M$ store the values $L_i(U_i^n) = -\frac{1}{\Delta x} \left(\hat{f}_{i+\frac{1}{2}} - \hat{f}_{i-\frac{1}{2}} \right)$.

Step 5.1. For $i = 1, \dots, M$ apply (15) to compute $U_i^{(1)}$.

Step 5.2. Retain the values of $\{U_i^n\}$ and $\{U_i^{(1)}\}$, then execute **Steps 1.2-4.3** by setting $\{U_i^{(1)}\}$ as initial values. Once $L_i(U_i^{(1)})$ have been calculated, utilise (16), for $i = 1, \dots, M$, to produce $U_i^{(2)}$.

Step 5.3. Preserve the values of $\{U_i^{(2)}\}$, and imitate **Steps 1.2-4.3** by assigning $\{U_i^{(2)}\}$ as the starting values. After $L_i(U_i^{(2)})$ have been computed, apply (17), for $i = 1, \dots, M$, to obtain U_i^{n+1} .

Step 5.4. Utilising $\{U_i^{n+1}\}$ as initial values, proceed to **Step 1.2** to advance to the next time level.

2) WENO5-M TVD Scheme

At certain smooth extrema or near critical points, the smoothness indicators-based nonlinear weights ω_s^{JS} , $k = 3$, are unable to meet the sufficient condition for fifth-order accuracy. Henrick *et al.* [12] introduced mapping functions to address this issue by applying them to the nonlinear weights ω_s^{JS} . In addition to meeting the fifth-order accuracy requirement with the resulting nonlinear weights ω_s^M , the mapped WENO scheme yields more accurate numerical solutions around discontinuities. Higher-order WENO-M schemes were also devised in [29]. For the WENO5-M TVD scheme, we use $\epsilon = 10^{-40}$ [12] and $C_{CFL} = 0.4$ [20].

Procedure 2 WENO5-M TVD Scheme

Steps 1.1-4.1.3.4. Execute **Steps 1.1-4.1.3.4** of **Procedure 1** for $k = 3$.

$$4.1.3.5. \alpha_s^M = g_s(\omega_s^{JS}) = \frac{\omega_s^{JS} \left(d_{3,s} + d_{3,s}^2 - 3d_{3,s} \omega_s^{JS} + (\omega_s^{JS})^2 \right)}{d_{3,s}^2 + \omega_s^{JS} (1 - 2d_{3,s})}, \quad s = 0, 1, 2.$$

$$4.1.3.6. \omega_s^M = \frac{\alpha_s^M}{\sum_{j=0}^2 \alpha_j^M}, \quad s = 0, 1, 2, \text{ (mapped weights).}$$

$$4.1.3.7. V_{i+\frac{1}{2}}^- = \sum_{s=0}^2 \omega_s^M p_i^{(3,s)} \left(x_{i+\frac{1}{2}} \right).$$

Steps 4.1.4-4.1.4.4. Execute **Steps 4.1.4-4.1.4.4** of **Procedure 1** for $k = 3$.

Steps 4.1.4.5-6. Compute α_s^M and ω_s^M in the same manner as in **Steps**

4.1.3.5-6.

Step 4.1.4.7.
$$V_{i+\frac{1}{2}}^+ = \sum_{s=0}^2 \omega_s^M p_{i+1}^{(3,s)} \left(x_{i+\frac{1}{2}} \right).$$

Steps 4.1.5-5.4. Execute **Steps 4.1.5-5.4** of **Procedure 1**.

3) WENO5-Z TVD Scheme

Borges *et al.* developed a different technique in [13] that enabled them to achieve fifth-order accuracy at the critical points of smooth solutions without the need of a map. Their introduction of the global SI and the development of Z-type nonlinear weights, ω_s^Z , led to the WENO-Z scheme. Compared to WENO5-JS and WENO5-M, the WENO5-Z scheme that results is less dissipative. The development of higher-order WENO-Z schemes was presented in [30]. We employ $\epsilon = 10^{-40}$ and $C_{CFL} = 0.4$ [13] for the WENO5-Z TVD scheme.

Procedure 3 WENO5-Z TVD Scheme

Steps 1.1-4.1.3.2. Implement **Steps 1.1-4.1.3.2** of **Procedure 1** for $k = 3$.

Step 4.1.3.3. $\tau_s = |\beta_0^{JS} - \beta_2^{JS}|$ (global SI).

Step 4.1.3.4.
$$\beta_s^Z = \frac{\beta_s^{JS} + \epsilon}{\beta_s^{JS} + \tau_s + \epsilon}, \quad s = 0, 1, 2.$$

Step 4.1.3.5.
$$\alpha_s^Z = \frac{d_{3,s}}{\beta_s^Z} = d_{3,s} \left(1 + \frac{\tau_s}{\beta_s^{JS} + \epsilon} \right), \quad s = 0, 1, 2.$$

Step 4.1.3.6.
$$\omega_s^Z = \frac{\alpha_s^Z}{\sum_{j=0}^2 \alpha_j^Z}, \quad s = 0, 1, 2.$$

Step 4.1.3.7.
$$V_{i+\frac{1}{2}}^- = \sum_{s=0}^2 \omega_s^Z p_i^{(3,s)} \left(x_{i+\frac{1}{2}} \right).$$

Steps 4.1.4-4.1.4.2. Implement **Steps 4.1.4-4.1.4.2** of **Procedure 1** for $k = 3$.

Steps 4.1.4.3-6. Calculate τ_s , β_s^Z , α_s^Z and ω_s^Z , $s = 0, 1, 2$, in the same manner as described in **Steps 4.1.3.3-6** above.

Step 4.1.4.7.
$$V_{i+\frac{1}{2}}^+ = \sum_{s=0}^2 \omega_s^Z p_{i+1}^{(3,s)} \left(x_{i+\frac{1}{2}} \right).$$

Steps 4.1.5-5.4 Implement **Steps 4.1.5-5.4** of **Procedure 1**.

4) WENO5-NS TVD Scheme

The WENO5-M and WENO5-Z schemes are developed by utilising the well-known SIs of WENO5-JS, β_s^{JS} , in an innovative manner. In contrast to the SIs β_s^{JS} , which are constructed via the L_2 -norm, Ha *et al.* introduce an alternative version of SIs that is based on the L_1 -norm in [14]. Their explicit expression is located in [14] and has been provided as well in **Appendix D**. The WENO5-NS TVD scheme is implemented with $\epsilon = 10^{-40}$ and $C_{CFL} = 0.5$ [14].

Procedure 4 WENO5-NS TVD Scheme

Steps 1.1-4.1.3.1. Perform **Steps 1.1-4.1.3.1** of **Procedure 1** for $k = 3$.

Step 4.1.3.2. $\beta_s^{NS} = \beta_s^{NS}(V, i)$, $s = 0, 1, 2$, employing (34).

Step 4.1.3.3.
$$\zeta = \zeta(i) = \frac{1}{2} \left(|\beta_0^{NS} - \beta_2^{NS}|^2 + g(|V_{i+1} - V_i|)^2 \right), \quad g(x) = \frac{x^3}{x^3 + 1}.$$

Step 4.1.3.4. $\alpha_s^{NS} = d_{3,s} \left(1 + \frac{\zeta}{\epsilon + \beta_s^{NS}} \right), \quad s = 0, 1, 2.$

Step 4.1.3.5. $\omega_s^{NS} = \frac{\alpha_s^{NS}}{\sum_{j=0}^2 \alpha_j^{NS}}, \quad s = 0, 1, 2.$

Step 4.1.3.6. $V_{i+\frac{1}{2}}^- = \sum_{s=0}^2 \omega_s^{NS} p_i^{(3,s)} \left(x_{i+\frac{1}{2}} \right).$

Steps 4.1.4-4.1.4.1. Perform **Steps 4.1.4-4.1.4.1** of **Procedure 1**.

Steps 4.1.4.2-5. Compute $\beta_s^{NS}(V, i+1)$, $\zeta(i+1)$, α_s^{NS} , and ω_s^{NS} , $s = 0, 1, 2$, using the formulas given in **Steps 4.1.3.2-5**.

Step 4.1.4.6. $V_{i+\frac{1}{2}}^+ = \sum_{s=0}^2 \omega_s^{NS} p_{i+1}^{(3,s)} \left(x_{i+\frac{1}{2}} \right).$

Steps 4.1.5-5.4 Perform **Steps 4.1.5-5.4** of **Procedure 1**.

5) WENO5-P TVD Scheme

Kim *et al.* introduce a novel fifth-order WENO in [15] that enhances the WENO5-NS in [14]. They achieve this by simplifying WENO5-NS's nonlinear weights and modifying its SIs. We utilise the values $\epsilon = 10^{-40}$ [15] and $C_{CFL} = 0.4$ [20] for the WENO5-P TVD scheme.

Procedure 5 WENO5-P TVD Scheme

Steps 1.1-4.1.3.2. Run **Steps 1.1-4.1.3.2** of **Procedure 4**.

Step 4.1.3.3. $\beta_0^P = \beta_0^{NS}$, $\beta_1^P = (1 + \delta)\beta_1^{NS}$, $\beta_2^P = (1 - \delta)\beta_2^{NS}$, $\delta = 0.05$ as per [15].

Step 4.1.3.4. $\zeta = (\beta_0^{NS} - \beta_2^{NS})^2.$

Step 4.1.3.5. $\alpha_s^P = d_{3,s} \left(1 + \frac{\zeta}{(\epsilon + \beta_s^P)^2} \right), \quad s = 0, 1, 2.$

Step 4.1.3.6. $\omega_s^P = \frac{\alpha_s^P}{\sum_{j=0}^2 \alpha_j^P}, \quad s = 0, 1, 2.$

Step 4.1.3.7. $V_{i+\frac{1}{2}}^- = \sum_{s=0}^2 \omega_s^P p_i^{(3,s)} \left(x_{i+\frac{1}{2}} \right).$

Steps 4.1.4-4.1.4.2. Run **Steps 4.1.4-4.1.4.2** of **Procedure 4**.

Steps 4.1.4.3-6. Evaluate, β_s^P , ζ , α_s^P , and ω_s^P , $s = 0, 1, 2$, in the same manner as described in **Steps 4.1.3.3-6** above.

Step 4.1.4.7. $V_{i+\frac{1}{2}}^+ = \sum_{s=0}^2 \omega_s^P p_{i+1}^{(3,s)} \left(x_{i+\frac{1}{2}} \right).$

Steps 4.1.5-5.4 Run **Steps 4.1.5-5.4** of **Procedure 1**.

6) WENO5-ZQ TVD Scheme

The WENO5-ZQ TVD scheme employs different reconstruction polynomials, designated $p_i^{ZQ(3,s)}(x)$, as opposed to the conventional ones, $p_i^{(3,s)}(x)$, $s = 0, 1, 2$. In fact, $p_i^{ZQ(3,0)}(x) = p_i^{(5,2)}(x)$, $p_i^{ZQ(3,1)}(x) = p_i^{(2,1)}(x)$, and $p_i^{ZQ(3,2)}(x) = p_i^{(2,0)}(x)$. To determine the SIs, the authors in [16] utilised Equation (23), with $p_i^{(k,s)}(x) = p_i^{ZQ(3,s)}(x)$ and the values of m being 4 for $s = 0$, and 1 for $s = 2, 3$. As a result, if we represent the SIs of WENO $(2k - 1)$ -JS as $\beta_{k,s}^{JS}$,

we can see that the SIs of WENO5-ZQ, β_s^{ZQ} , are: $\beta_0^{ZQ} = \beta_{5,2}^{JS}$, $\beta_1^{ZQ} = \beta_{2,1}^{JS}$ and $\beta_2^{ZQ} = \beta_{2,0}^{JS}$. For the WENO5-ZQ TVD scheme, we use $\epsilon = 10^{-6}$ and $C_{\text{CFL}} = 0.6$ [16].

Procedure 6 WENO5-ZQ TVD Scheme

Steps 1.1-4.1.3 Execute **Steps 1.1-4.1.3** of **Procedure 1** for $k = 3$.

Step 4.1.3.1.
$$p_i^{ZQ(3,0)}\left(x_{i+\frac{1}{2}}\right) = p_i^{(5,2)}\left(x_{i+\frac{1}{2}}\right), \quad p_i^{ZQ(3,1)}\left(x_{i+\frac{1}{2}}\right) = p_i^{(2,1)}\left(x_{i+\frac{1}{2}}\right),$$

$$p_i^{ZQ(3,2)}\left(x_{i+\frac{1}{2}}\right) = p_i^{(2,0)}\left(x_{i+\frac{1}{2}}\right).$$

The computation of $p_i^{(k,s)}\left(x_{i+\frac{1}{2}}\right)$ can be derived from **Step 4.1.3.1** of **Procedure 1**.

Step 4.1.3.2. $\beta_0^{ZQ} = \beta_{5,2}^{JS}(V, i)$, $\beta_1^{ZQ} = \beta_{2,1}^{JS}(V, i)$, $\beta_2^{ZQ} = \beta_{2,0}^{JS}(V, i)$, from (33), (30).

Step 4.1.3.3.
$$\tau = \left(\frac{|\beta_1^{ZQ} - \beta_2^{ZQ}| + |\beta_1^{ZQ} - \beta_3^{ZQ}|}{2} \right)^2.$$

Step 4.1.3.4.
$$\alpha_s^{ZQ} = \gamma_s \left(1 + \frac{\tau}{\epsilon + \beta_s^{ZQ}} \right), \quad s = 0, 1, 2, \text{ where } \gamma_0 = 0.98,$$

$\gamma_1 = \gamma_2 = 0.01$, as indicated by [16].

Step 4.1.3.5.
$$\omega_s^{ZQ} = \frac{\alpha_s^{ZQ}}{\sum_{j=0}^2 \alpha_j^{ZQ}}, \quad s = 0, 1, 2.$$

Step 4.1.3.6.
$$V_{i+\frac{1}{2}}^- = \omega_0^{ZQ} \left(\frac{1}{\gamma_0} p_i^{ZQ(3,0)}\left(x_{i+\frac{1}{2}}\right) - \frac{\gamma_1}{\gamma_0} p_i^{ZQ(3,1)}\left(x_{i+\frac{1}{2}}\right) - \frac{\gamma_2}{\gamma_0} p_i^{ZQ(3,2)}\left(x_{i+\frac{1}{2}}\right) \right) + \sum_{s=1}^2 \omega_s^{ZQ} p_i^{ZQ(3,s)}\left(x_{i+\frac{1}{2}}\right).$$

Step 4.1.4. To estimate the right extrapolated value $V_{i+\frac{1}{2}}^+$ calculate

Step 4.1.4.1.
$$p_{i+1}^{ZQ(3,0)}\left(x_{i+\frac{1}{2}}\right) = p_{i+1}^{(5,2)}\left(x_{i+\frac{1}{2}}\right), \quad p_{i+1}^{ZQ(3,1)}\left(x_{i+\frac{1}{2}}\right) = p_{i+1}^{(2,1)}\left(x_{i+\frac{1}{2}}\right),$$

$$p_{i+1}^{ZQ(3,2)}\left(x_{i+\frac{1}{2}}\right) = p_{i+1}^{(2,0)}\left(x_{i+\frac{1}{2}}\right).$$

The computation of $p_{i+1}^{(k,s)}\left(x_{i+\frac{1}{2}}\right)$ can be derived from **Step 4.1.4.1** of **Procedure 1**.

Steps 4.1.4.2-5. $\beta_s^{ZQ}(V, i+1)$, τ , α_s^{ZQ} , and ω_s^{ZQ} , $s = 0, 1, 2$, using the formulas provided above.

Step 4.1.4.6.
$$V_{i+\frac{1}{2}}^+ = \omega_0^{ZQ} \left(\frac{1}{\gamma_0} p_{i+1}^{ZQ(3,0)}\left(x_{i+\frac{1}{2}}\right) - \frac{\gamma_1}{\gamma_0} p_{i+1}^{ZQ(3,1)}\left(x_{i+\frac{1}{2}}\right) - \frac{\gamma_2}{\gamma_0} p_{i+1}^{ZQ(3,2)}\left(x_{i+\frac{1}{2}}\right) \right) + \sum_{s=1}^2 \omega_s^{ZQ} p_{i+1}^{ZQ(3,s)}\left(x_{i+\frac{1}{2}}\right).$$

Steps 4.1.5-5.4 Execute **Steps 4.1.5-5.4** of **Procedure 1**.

7) MWENO5-P TVD Scheme

The WENO5-NS and WENO5-P schemes achieve fifth-order accuracy, even in the presence of first-order critical points. However, they do not maintain this accuracy when approaching second-order critical points. In [17], Rathan *et al.* developed a modified version of the WENO-P scheme that was capable of attaining fifth-order accuracy even at the points where the second-order derivatives become zero. They accomplished this by creating a new global SI. The MWENO5-P TVD scheme is implemented with $\epsilon = 10^{-40}$, $\xi = 0.1$, $\delta = 0.05$ and $C_{CFL} = 0.5$ [17].

Procedure 7 MWENO5-P TVD Scheme

Steps 1.1-4.1.3.2. Run **Steps 1.1-4.1.3.2** of **Procedure 4**.

Step 4.1.3.3. $\beta_0^{MP} = \beta_0^{NS}$, $\beta_1^{MP} = (1 + \delta)\beta_1^{NS}$, $\beta_2^{MP} = (1 - \delta)\beta_2^{NS}$.

Step 4.1.3.4. $\eta = \eta(i) = (V_{i-2} - 4V_{i-1} + 6V_i - 4V_{i+1} + V_{i+2})^2$,

Step 4.1.3.5. $\alpha_s^{MP} = d_{3,s} \left(1 + \frac{\eta}{(\epsilon + \beta_s^{MP})^2} \right)$, $s = 0, 1, 2$.

Step 4.1.3.6. $\omega_s^{MP} = \frac{\alpha_s^{MP}}{\sum_{j=0}^2 \alpha_j^{MP}}$, $s = 0, 1, 2$.

Step 4.1.3.7. $V_{i+\frac{1}{2}}^- = \sum_{s=0}^2 \omega_s^{MP} p_i^{(3,s)} \left(x_{i+\frac{1}{2}} \right)$.

Steps 4.1.4-4.1.4.2. Run **Steps 4.1.4-4.1.4.2** of **Procedure 4**.

Steps 4.1.4.3-6. Evaluate, β_s^{MP} , $\eta(i+1)$, α_s^{MP} , and ω_s^{MP} , $s = 0, 1, 2$, in the same manner as in **Steps 4.1.3.3-6**.

Step 4.1.4.7. $V_{i+\frac{1}{2}}^+ = \sum_{s=0}^2 \omega_s^{MP} p_{i+1}^{(3,s)} \left(x_{i+\frac{1}{2}} \right)$.

Steps 4.1.5-5.4 Run **Steps 4.1.5-5.4** of **Procedure 1**.

8) WENO (2k - 1)-ZS TVD Scheme (k = 2, 3, 4, 5)

In [18], Zhu and Shu present a novel class of multi-resolution WENO schemes with $(2k - 1)$ -th order accuracy, $k = 2, 3, 4, 5$, in the FD and FV frameworks. Contrary to the WENO-JS scheme, there are two significant distinctions. Initially, the new WENO reconstruction is a convex combination of reconstruction polynomials $p_i^{(k,s)}(x)$, denoted as $p_i^{ZS(k,s)}(x)$, of varying degrees. Secondly, the positive linear weights may be arbitrarily selected provided that they add up to one.

The definition of $p_i^{ZS(k,s)}(x)$ can be found in [18] and are also provided in **Appendix E.1**. In order to calculate the SIs β_s^{ZS} , the authors in [18] employed Equation (23), where $p_i^{(k,s)}(x) = p_i^{ZS(k,s)}(x)$ and $m = 2s$ for $s = 1, \dots, k - 1$, respectively. For $s = 0$, the definition of β_0^{ZS} is different, and its specific expression is available in [18]. The calculation equations for β_1^{ZS} and β_2^{ZS} are offered in [31] for $k = 3$ (and as a result for $k = 2$). We utilised an algebraic manipulator (Mathematica) to determine the precise expressions for β_3^{ZS} ($k = 4$) and β_4^{ZS} ($k = 5$), which are presented in **Appendix E.2**. Additionally, we have

added the formulas for β_s^{ZS} for $s=0,1,2$ to ensure thoroughness.

The values $\epsilon = 10^{-10}$ and $C_{CFL} = 0.6$ [18] are employed in the WENO $(2k-1)$ -ZS TVD scheme.

Procedure 8 WENO $(2k-1)$ -ZS TVD Scheme, $k=2,3,4,5$

Steps 1.1-4.1.3 Execute **Steps 1.1-4.1.3** of **Procedure 1**.

Step 4.1.3.1. Calculate $p_i^{ZS(k,s)} \left(x_{i+\frac{1}{2}} \right)$, $s=0, \dots, k-1$, from (43), (45), (47),

and (49) for $k=2,3,4,5$, respectively.

Step 4.1.3.2. $\beta_s^{ZS} = \beta_s^{ZS}(V, i)$, $s=0, \dots, k-1$, employing (37)-(38), (39), (40), and (41) for $k=2,3,4,5$, respectively.

Step 4.1.3.3. $\tau_k = \left(\frac{\sum_{j=0}^{k-2} |\beta_{k-1}^{ZS} - \beta_j^{ZS}|}{k-1} \right)^{k-1}$.

Step 4.1.3.4. $\alpha_{s,k}^{ZS} = \gamma_{s,k-1} \left(1 + \frac{\tau_k}{\epsilon + \beta_s^{ZS}} \right)$, $s=0, \dots, k-1$.

Step 4.1.3.5. $\omega_{s,k}^{ZS} = \frac{\alpha_{s,k}^{ZS}}{\sum_{j=0}^{k-1} \alpha_{j,k}^{ZS}}$, $s=0, \dots, k-1$.

Step 4.1.3.6. $V_{i+\frac{1}{2}}^- = \sum_{s=0}^{k-1} \omega_{s,k}^{ZS} p_i^{ZS(k,s)} \left(x_{i+\frac{1}{2}} \right)$.

Step 4.1.4. To estimate the right extrapolated value $V_{i+\frac{1}{2}}^+$ calculate

Step 4.1.4.1. $p_{i+1}^{ZS(k,s)} \left(x_{i+\frac{1}{2}} \right)$, $s=0, \dots, k-1$ from (44), (46), (48), and (50) for

$k=2,3,4,5$, respectively.

Steps 4.1.4.2-5. $\beta_s^{ZS}(V, i+1)$, τ_k , $\alpha_{s,k}^{ZS}$, and $\omega_{s,k}^{ZS}$, $s=0, \dots, k-1$, applying the formulas provided above.

Step 4.1.4.6. $V_{i+\frac{1}{2}}^+ = \sum_{s=0}^{k-1} \omega_{s,k}^{ZS} p_{i+1}^{ZS(k,s)} \left(x_{i+\frac{1}{2}} \right)$.

Steps 4.1.5-5.4 Execute **Steps 4.1.5-5.4** of **Procedure 1**.

9) WENO5-ZR TVD Scheme

In [19], Gu *et al.* introduce a novel set of Z-type nonlinear weights, ω_s^{ZR} . The new nonlinear weights meet the criteria for fifth-order accuracy, even at critical points, and the dissipation is reduced around the discontinuities. For the WENO5-ZR TVD scheme, $\epsilon = 10^{-40}$ and $C_{CFL} = 0.4$ are used [19].

Procedure 9 WENO5-ZR TVD Scheme

Steps 1.1-4.1.3.2. Run **Steps 1.1-4.1.3.2** of **Procedure 1** for $k=3$.

Step 4.1.3.3. $\tau_s^{ZR} = \left| \sqrt[p]{\beta_0^{JS}} - \sqrt[p]{\beta_2^{JS}} \right|$, $p=3$ as per [19].

Step 4.1.3.4. $\alpha_s^{ZR} = d_{3,s} \left(1 + \left(\frac{\tau_s^{ZR}}{\sqrt[p]{\beta_s^{JS}} + \epsilon} \right)^p \right)$, $s=0,1,2$, $p=3$.

Step 4.1.3.5. $\omega_s^{ZR} = \frac{\alpha_s^{ZR}}{\sum_{j=0}^2 \alpha_j^{ZR}}$, $s=0,1,2$.

Step 4.1.3.6.
$$V_{i+\frac{1}{2}}^- = \sum_{s=0}^2 \omega_s^{ZR} p_i^{(3,s)} \left(x_{i+\frac{1}{2}} \right).$$

Steps 4.1.4-4.1.4.2. Run **Steps 4.1.4-4.1.4.2** of **Procedure 1** for $k = 3$.

Steps 4.1.4.3-5. Calculate τ_s^{ZR} , α_s^{ZR} and ω_s^{ZR} , $s = 0, 1, 2$, in the same manner as described in **Steps 4.1.3.3-5** above.

Step 4.1.4.6.
$$V_{i+\frac{1}{2}}^+ = \sum_{s=0}^2 \omega_s^{ZR} p_{i+1}^{(3,s)} \left(x_{i+\frac{1}{2}} \right).$$

Steps 4.1.5-5.4 Run **Steps 4.1.5-5.4** of **Procedure 1**.

3.3.2. Random Choice Method (RCM)

In order to perform the RCM, it is required to produce a random number, represented as θ^n , at each time level n . Consequently, it is necessary to implement a procedure that generates a sequence of random numbers. The current study employs the (5, 3) van der Corput sequence [2] [32].

This study's version of the RCM does not operate in the Finite Volume (FV) framework. This implies that the solution at the new time level is obtained without the use of the FV numerical scheme (13). The solution is derived by utilising the solutions to RPs, as demonstrated in the subsequent procedure. The solution to the RP (U_i^n, U_{i+1}^n) is denoted as $U_{i+\frac{1}{2}}^n(x/t)$ and it is computed utilising an exact Riemann solver.

Procedure 10 RCM

Step 1.1. To initiate the scheme, utilise the ICs $U(x, 0)$ to calculate the values $U(x_i, 0)$ at the cell centres x_i for each cell I_i . Set $U(x_i, 0) = U_i^0$, $i = 1, \dots, M$.

Step 1.2. Use the values $\{U_i^n\}_{i=1}^{i=M}$, $n \geq 0$, as input.

Step 2. Apply the BCs as specified in (21)-(22) with $r = 1$.

Steps 3.1-2. Execute **Steps 3.1-2** of **Procedure 1**.

Step 4. Generate the random number θ^n .

Step 5.1. For $i = 1, \dots, M$

- If $0 \leq \theta^n \leq 0.5$ then solve RP (UP_{i-1}^n, UP_i^n) and set $UP_i^{n+1} = UP_{i-\frac{1}{2}}^n(\theta^n \Delta x / \Delta t)$.
- If $0.5 < \theta^n \leq 1$ then solve RP (UP_i^n, UP_{i+1}^n) and set $UP_i^{n+1} = UP_{i+\frac{1}{2}}^n((\theta^n - 1)\Delta x / \Delta t)$.

Step 5.2. For $i = 1, \dots, M$, transform the vector of primitive variables $UP_i^{n+1} = [h_i^{n+1}, u_i^{n+1}]^T$ to the vector of conserved variables $U_i^{n+1} = [u_{1i}^{n+1}, u_{2i}^{n+1}]^T = [h_i^{n+1}, h_i^{n+1} u_i^{n+1}]^T$.

Step 5.3. Utilising $\{U_i^{n+1}\}$ as initial values, proceed to **Step 1.2** to advance to the next time level.

4. Numerical Results

This section presents a comparison of the numerical performance of the schemes outlined in the preceding section when applied to test problems. The schemes are also examined in terms of their computational efficiency. The test problems are

RPs for the 1D SWEs, and each one corresponds to a certain wave pattern, which is depicted in **Figure A2** of **Appendix A**. Both Tests 1 and 2 were sourced from [1], while Test 3 was acquired from [3]. In order to finalise the four wave patterns, we constructed Test 4.

To normalise the provided spatial domain to the interval $[0, 1]$, we adjusted the initial discontinuity's position and the (short) output time in Tests 1, 2, 3. We discretize the interval $[0, 1]$ for all test problems using $M = 100, 200, 400,$ and 800 cells, and compute the numerical solution at both short and long output times. The exact solutions to the test problems are acquired by the utilisation of an exact Riemann solver.

In order to quantify the magnitude of the error E , we employed the L_1 norm [3]:

$$\|E\|_1 = \Delta x \sum_{i=1}^M \|\bar{U}_i^n - U_i^n\|.$$

Here, \bar{U}_i^n is the cell average of $U(x, t)$ over the interval I_i at time t^n , U_i^n is an approximation to the \bar{U}_i^n provided by the numerical method, and $\|U\|_1 = \|[u_1, u_2]^T\|_1 = |u_1| + |u_2|$ is the L_1 vector norm on \mathbb{R}^2 . For the RCM, \bar{U}_i^n is substituted by $U(x_i, t^n)$, which is the value of the exact solution $U(x, t)$ at $(x, t) = (x_i, t^n)$.

To conserve space, we will only provide numerical data and graphs from three numerical methods for each test problem. For short time simulation, we selected the numerical methods based on their capacity to accurately resolve discontinuities such as jump discontinuities caused by shocks, as well as derivative discontinuities located in the head and tail of rarefactions. Secondary considerations include the L_1 error and CPU cost. In Tests 1, 3, and 4, the water depth and velocity plots exhibit comparable features. Thus, to save space, we only provide graphs for the water depth on both coarse and fine meshes.

In long time simulations, the solution structure is rather uncomplicated (free of shock discontinuities and rarefaction corners), which leads to similar numerical patterns. To assess the efficacy of the methods in this case, it is sufficient to compare the L_1 errors. Computing time is additionally a factor to evaluate. The numerical methods that yielded the smallest L_1 error were chosen for presentation. For each Test, we only plot the numerical solution of water depth obtained from three different schemes on a 100-cell mesh within the same image; the velocity plots are comparable and thus not presented. To enhance visualisation, the y scales are greatly enlarged.

Tables for each Test provide the L_1 errors of cell averages of the solution at both output times, with the exception of the RCM, which employs the exact solution to calculate the L_1 errors. For each Δx , the scheme with the minimum L_1 error is displayed in bold. This paper does not aim to conduct a rate of convergence study of the numerical methods. Subsequently, the convergence rates in the Tables and an accuracy analysis are ignored, here.

4.1. Computer-Related Issues

It is necessary to make a few comments regarding computer-related matters. The first is about the solution to the RP (2), (3), (9). Specifically, one must solve an algebraic nonlinear equation to obtain $h(x, t)$ in the star region. As far as we know, there is no closed-form solution to this equation. Consequently, we numerically solve it employing the Newton-Raphson iteration scheme [33] to a tolerance of $TOL = 10^{-6}$.

The second involves the termination of the methods' algorithms once the solution reaches the designated output time, represented as t_{out} . The algorithm stops whenever $\left| \sum_{n=1}^m \Delta t^{(n)} - t_{out} \right| \leq TOL$ for some m , where $\Delta t^{(n)}$ denotes the time step at time level t^n .

The third matter concerns the calculation of the ratios $r_{j,i+\frac{1}{2}}$ in (27), specifically when the denominator is of small magnitude. Following [20], we set

$$\Delta h_{j,i+l} = TOL \cdot \text{sign}(1, \Delta h_{j,i+l}) \quad \text{if } |\Delta h_{j,i+l}| \leq TOL, \quad l = \pm \frac{1}{2}, \frac{3}{2}, \quad j = 1, 2.$$

4.2. Test 1: Left Rarefaction and Right Shock

Consider the RP (2), (3), (9) with $U_L = [1, 2.5]^T$, $U_R = [0.1, 0]^T$, $(x, t) \in [0, 1] \times (0, +\infty)$ and $x_0 = 0.2$. **Figure 1**, left, depicts the structure of the solution to the RP in the $x-t$ plane. The exact solution is

$$U(x, t) = \begin{cases} U_L, & \text{if } S \leq S_{HL}, \\ U_{Lfan}(x, t), & \text{if } S_{HL} < S < S_{TL}, \\ U^*, & \text{if } S_{TL} \leq S < S_{SR}, \\ U_R, & \text{if } S_{SR} < S, \end{cases} \quad S = \frac{x-0.2}{t},$$

where $S_{HL} \approx -0.632092$, $S_{TL} \approx 1.415611$, $S_{SR} \approx 4.620578$, $U^* \approx [0.611638, 2.364063]^T$, and U_{Lfan} is determined by (10).

The initial data of this test results in the formation of a right shock and a left sonic or critical rarefaction. This means that when crossing the wave from left to right the eigenvalue λ_1 transitions from a negative value to a positive value, attaining the value $\lambda_1 = 0$, commonly referred to as the sonic point. The primary difficulties for this test are the numerical methods' inability to accurately manage the sonic point, the head and tail of the rarefaction, and the shock.

We calculate the numerical solution at the output times of $t_{out} = 0.14$ and $t_{out} = 100$. By analysing the exact solution, it is evident that when $t_{out} < 0.8/S_{SR} \approx 0.173$, the solution comprises four distinct states characterised by smooth regions, derivative discontinuities, and a jump discontinuity. However, when $t_{out} > 0.8/S_{TL} \approx 0.565$, the solution is within the left rarefaction and is hence smooth. In this case, the solution has a simple structure and therefore lacks significance for examination.

For short time simulation, the chosen methods for presentation are RCM, WENO5-ZR, and WENO3-ZS. **Figure 2** displays the outcomes on both coarse and

fine meshes, while **Table 1** provides the L_1 errors.

The sonic point was accurately treated by all of the methods. The RCM handles the discontinuities in the derivative and the jump discontinuity with great precision. Its vulnerability lies in its inadequate approximation inside the rarefaction; though, mesh refinement enhances the approximation (**Figure 2(a₁)** and **Figure 2(a₂)**). However, when $t_{out} = 100$ and the exact solution is within the rarefaction, the numerical solution produced by the RCM closely resembles the true solution (**Figure 1**, right). Furthermore, the position of the shock differs by Δx from its exact position.

Out of all the WENO methods, the WENO5-ZQ was found to be less satisfactory due to the presence of oscillations near the head and tail of the rarefaction, as well as in the neighbourhood of the shock. The methods WENO5-P, WENO5-NS, WENO5-Z, MWENO5-P, and WENO5-M produced comparable results to WENO5-ZR (see **Figure 2(b₁)** and **Figure 2(b₂)**). The rarefaction is well approximated, even in close proximity to the head and tail, which are smeared over 4 - 6 cells. However, beyond the tail, there are non-physical oscillations that cease after a few cells in the star region. The visibility of these oscillations increases as the mesh is refined. The shock is effectively resolved within a range of 3 to 4 cells.

The WENO $(2k - 1)$ -JS scheme exhibited identical behaviour, save for the fact that the oscillations persisted beyond the tail. The oscillations are scarcely discernible with the WENO3-JS scheme, but become clearer as the mesh is refined and the order of the scheme is increased. The shock is resolved across four cells.

The WENO $(2k - 1)$ -ZS scheme exhibited similar behaviour to the WENO $(2k - 1)$ -JS scheme, with the exception that it generated minor oscillations in the vicinity of the shock. The WENO3-ZS scheme is an exception, as it successfully approximates the solution in smooth regions and provides a satisfactory resolution of discontinuities. These features of the WENO3-ZS scheme are improved by mesh refinement (see **Figure 2(c₁)** and **Figure 2(c₂)**).

The method that resulted in the smallest L_1 error for long time simulation is RCM (except for $M = 400$), followed by WENO3-JS, and WENO3-ZS (refer to **Table 2** and **Figure 1**, right).

Table 1. Test 1. L_1 -norm errors of several schemes at $t = 0.14$.

M	Δx	RCM	WENO3-ZS	WENO5-ZR
100	0.01	2.2909E-03	5.5325E-03	3.8616E-03
200	0.005	1.6092E-02	3.2359E-03	2.0202E-03
400	0.0025	8.7300E-03	1.6110E-03	1.1562E-03
800	0.00125	5.3099E-04	6.8180E-04	3.9836E-04

4.3. Test 2: Left and Right Rarefaction

Consider the RP (2), (3), (9) with $U_L = [1, -5]^T$, $U_R = [1, 5]^T$, $(x, t) \in [0, 1] \times (0, +\infty)$ and $x_0 = 0.5$. **Figure 3**, left, illustrates the structure of the solution to the RP in the $x-t$ plane. The exact solution is

Table 2. Test 1. L_1 -norm errors of several schemes at $t = 100$.

M	Δx	RCM	WENO3-JS	WENO3-ZS
100	0.01	1.1676E-05	2.1040E-04	2.2357E-02
200	0.005	7.7622E-06	9.9828E-06	1.5257E-02
400	0.0025	4.3555E-06	2.6433E-06	7.9449E-03
800	0.00125	2.5609E-06	1.7722E-05	4.0256E-03

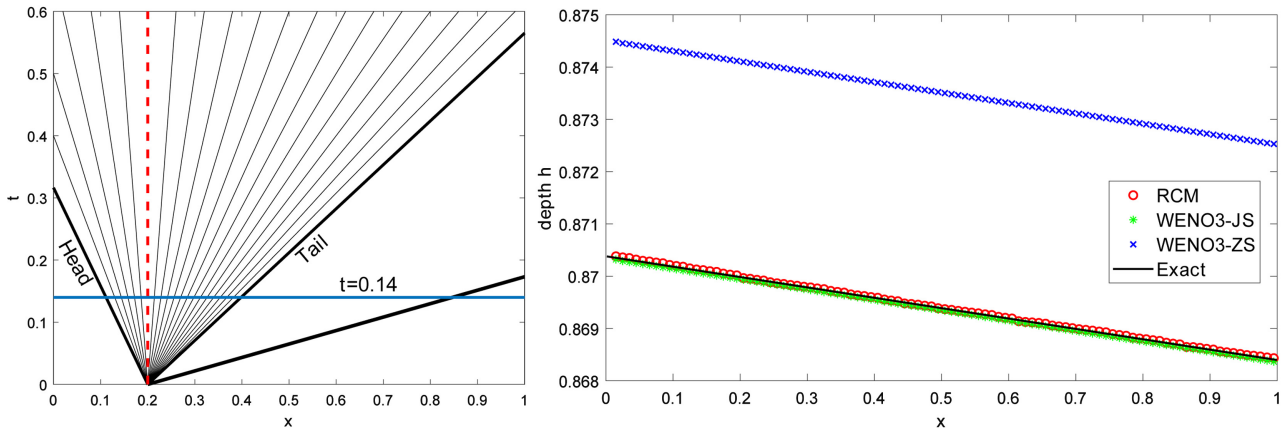


Figure 1. Test 1. Left: The structure of the solution of the RP in the $x-t$ plane. Right: Close-up view of water depth profiles at $t = 100$ with $M = 100$.

$$U(x, t) = \begin{cases} U_L, & \text{if } S \leq S_{HL}, \\ U_{Lfan}(x, t), & \text{if } S_{HL} < S < S_{TL}, \\ U^*, & \text{if } S_{TL} \leq S \leq S_{TR}, \\ U_{Rfan}(x, t), & \text{if } S_{TR} < S < S_{HR}, \\ U_R, & \text{if } S_{HR} \leq S, \end{cases} \quad S = \frac{x - 0.5}{t},$$

where $S_{HL} \approx -8.132092$, $S_{TL} \approx -0.632092$, $S_{TR} \approx 0.632092$, $S_{HR} \approx 8.132092$, $U^* \approx [0.040728, 0]^T$. The values of U_{Lfan} and U_{Rfan} are obtained by (10) and (11) accordingly.

For Test 2, the initial data generates two symmetric rarefactions. The two waves are symmetrical about the line $x = 0.5$. The main challenges of this test, as indicated by the numerical results, lie in properly managing the tails of the two rarefactions and the star region between these waves. The water in the star region is characterised by a shallow depth. Many methods will calculate a negative value for the depth h in that particular area, causing the algorithms to crash when trying to compute the celerity [1]. Thus, this problem is suitable for evaluating the efficiency of numerical methods for **low-depth water flows**.

The numerical solution is computed at the output times of $t_{out} = 0.05$ and $t_{out} = 100$. Upon studying the exact solution, it becomes clear that for $t_{out} < 0.5/S_{HR} \approx 0.061$, the solution consists of five distinct states with smooth parts and derivative discontinuities. On the other hand, for $t_{out} > 0.5/S_{TR} \approx 0.791$, the solution falls within the star region and is thus constant. In this instance, the solution is devoid of significance for examination due to its simple structure.

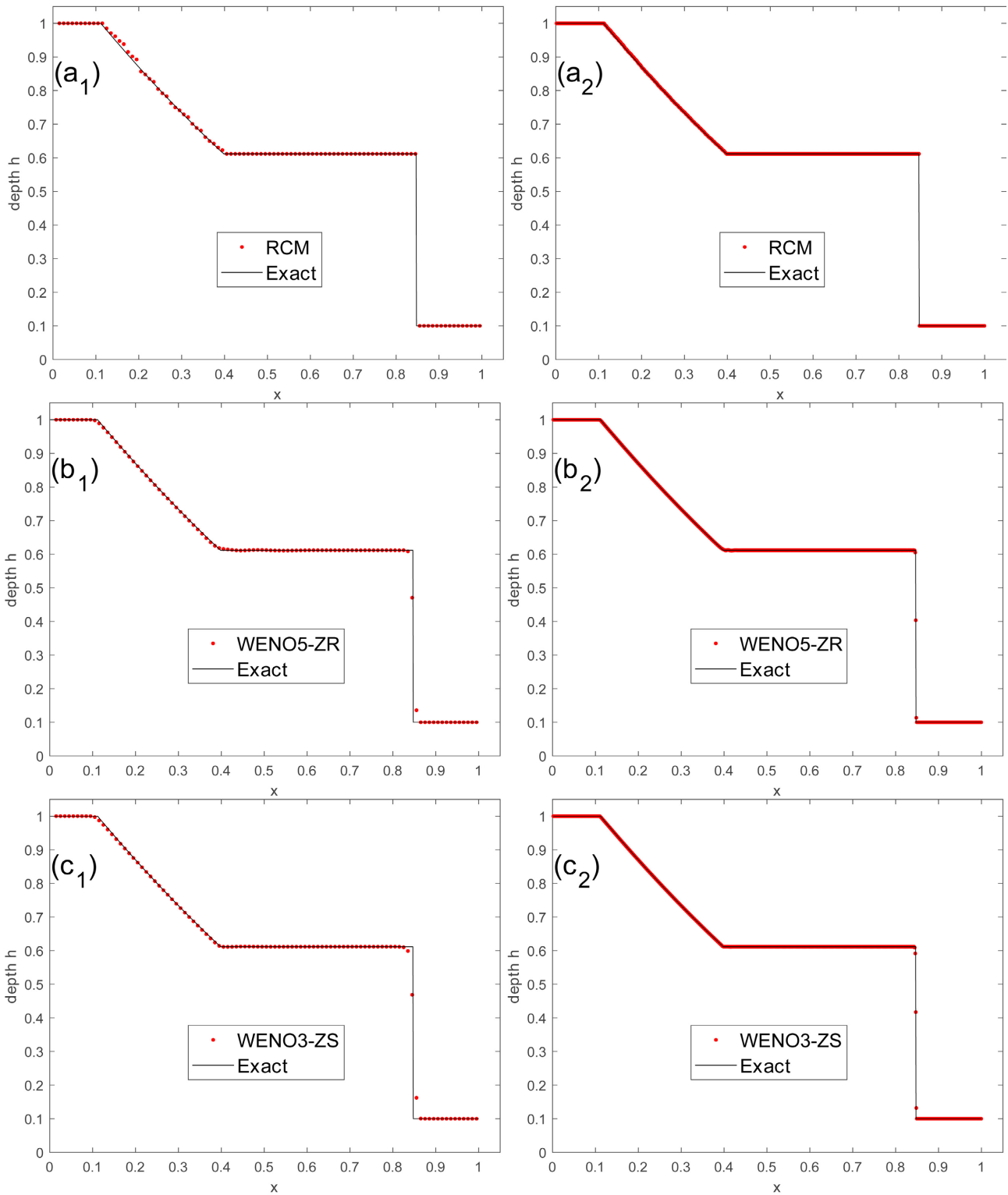


Figure 2. Test 1. Water depth profiles at $t = 0.14$ with $M = 100$ (left) and $M = 800$ (right).

To begin, it is necessary to make some comments regarding the algorithms as coded by the authors. We were required to decrease the Courant-Friedrichs-Lewy (CFL) coefficient beyond the value specified by the algorithms for the initial 5

time steps in order to run MWENO5-P, WENO5-M, and WENO5-P. The MWENO5-P and WENO5-M had their CFL coefficient reduced to 0.2, while the WENO5-P had its CFL coefficient reduced to 0.1. Furthermore, the value of parameter ξ for the MWENO5-P was adjusted from 0.1 to 0.4. To run the WENO5-NS, we reduced the CFL coefficient from 0.5 to 0.4. The WENO9-JS method proved to be ineffective in providing a solution for this test problem, even when the CFL coefficient was reduced to a significantly low value of 0.05.

All methods, except for the RCM, failed to fully realize the constant state between the left and right rarefactions (the star region) for the output time $t_{out} = 0.05$ on the 100-cell mesh. The mesh refinement enhances the realization of the constant space; however, it results in overshoots and undershoots in the vicinity of the tails of the left and right rarefactions. Further information will be provided subsequently. For short time simulation, the methods selected for demonstrating are RCM, WENO5-ZR, and WENO5-Z. The numerical solutions for water depth and velocity are depicted in **Figure 4** and **Figure 5**, respectively, and the L_1 errors are presented in **Table 3**.

Figure 4(a₁) and **Figure 5(a₁)** show the results of the RCM for the output time $t_{out} = 0.05$ on the mesh of 100 cells. It is apparent that it perfectly captures the constant state between the rarefactions. It appears that the rarefactions are not smooth, but they are sufficiently close to the exact solution. Furthermore, the corners at the endpoints of the rarefactions are slightly rounded. These issues can be very well resolved by refining the mesh, see **Figure 4(a₂)** and **Figure 5(a₂)**.

Figure 4(b₁) and **Figure 5(b₁)** display the outcomes obtained using the WENO5-ZR method for the output time $t_{out} = 0.05$ on the mesh of 100 cells. It is evident that it only partially reproduces the constant state between the rarefactions. The water depth profile provided by the WENO5-Z (**Figure 4(c₁)** and **Figure 5(c₁)**), MWENO5-P, and WENO5-JS, as well as the velocity profile obtained by the WENO5-ZS, likewise exhibit this feature. In the numerical solution produced by the remaining WENO methods, the same constant state is either scarcely present or entirely absent. These illustrations also demonstrate that the rarefactions are accurately approximated and that the corners at the heads of the left and right rarefactions are curved. This is also the case for the solutions obtained from the remaining WENO methods, with the exception of WENO5-ZQ and WENO $(2k - 1)$ -ZS, $k = 3, 4, 5$, where small amplitude oscillations were observed prior to the head of the left rarefaction and subsequent to the head of the right rarefaction.

On the finest mesh of 800 cells (**Figure 4(b₂)** and **Figure 5(b₂)**), we can visually confirm that the rarefactions, along with their heads, are indistinguishable from the exact solution. In addition, the constant state between the rarefactions is significantly more apparent; however, it is important to observe the overshoots and undershoots in the vicinity of the rarefactions' tails. To be more precise, the water depth profile suffers from undershoots following the tail of the left rarefaction and prior to the tail of the right rarefaction, while the velocity profile exhibits an

overshoot and an undershoot, respectively. All WENO methods yielded a solution that adhered to the identical pattern. The magnitude of these overshoots and undershoots (both in terms of height and width) vary depending on the WENO method. The solutions produced by WENO5-Z (Figure 4(c₂) and Figure 5(c₂)), WENO5-JS, and WENO5-ZR exhibited the lowest height measurements, over 8 cells, 9 cells, and 7 cells, respectively.

Furthermore, we noticed that spurious oscillations in the star region accompany the overshoots and undershoots in the solution generated by WENO5-ZQ and WENO (2k - 1)-ZS, k = 3, 4, 5. Additionally, regarding these methods, the small amplitude oscillations that were noticed near the heads of the rarefactions on the coarser mesh still persist and are more prominent in the water depth profile.

For long time simulation, consult Figure 3, right, and Table 4. RCM produced the smallest L₁ error, followed by WENO5-M and WENO5-ZQ.

Table 3. Test 2. L₁-norm errors of several schemes at t = 0.05.

M	Δx	RCM	WENO5-ZR	WENO5-Z
100	0.01	1.0180E-01	2.6473E-02	2.8579E-02
200	0.005	5.3293E-02	1.3311E-02	1.4331E-02
400	0.0025	1.9662E-02	6.6564E-03	7.1691E-03
800	0.00125	6.6116E-03	3.3275E-03	3.5858E-03

Table 4. Test 2. L₁-norm errors of several schemes at t = 100.

M	Δx	RCM	WENO5-M	WENO5-ZQ
100	0.01	2.8679E-09	1.4091E-03	1.6534E-03
200	0.005	2.8679E-09	4.6309E-04	6.1939E-04
400	0.0025	2.8679E-09	1.6672E-04	2.5940E-04
800	0.00125	2.8679E-09	6.9874E-05	1.1608E-04

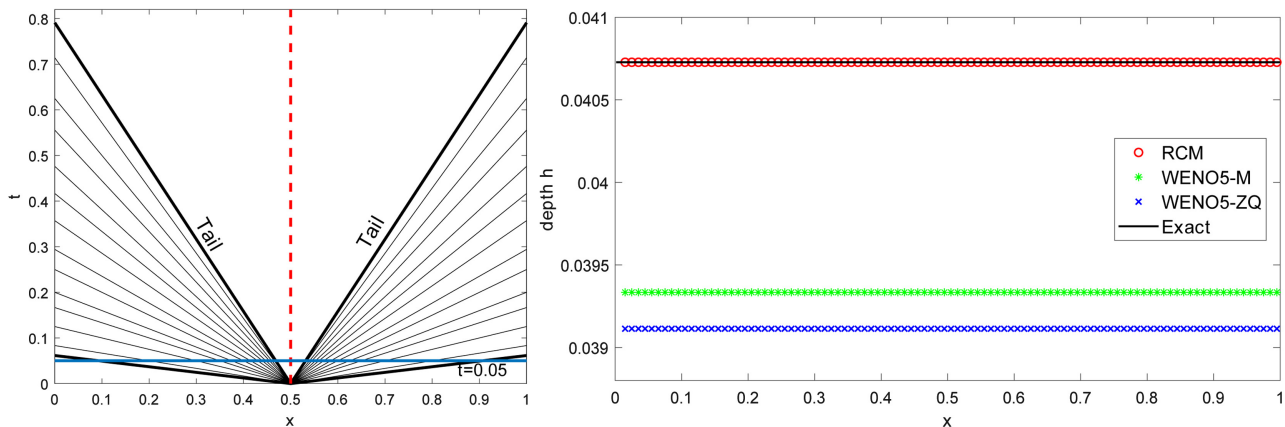


Figure 3. Test 2. Left: The structure of the solution of the RP in the x-t plane. Right: Close-up view of water depth profiles at t = 100 with M = 100.

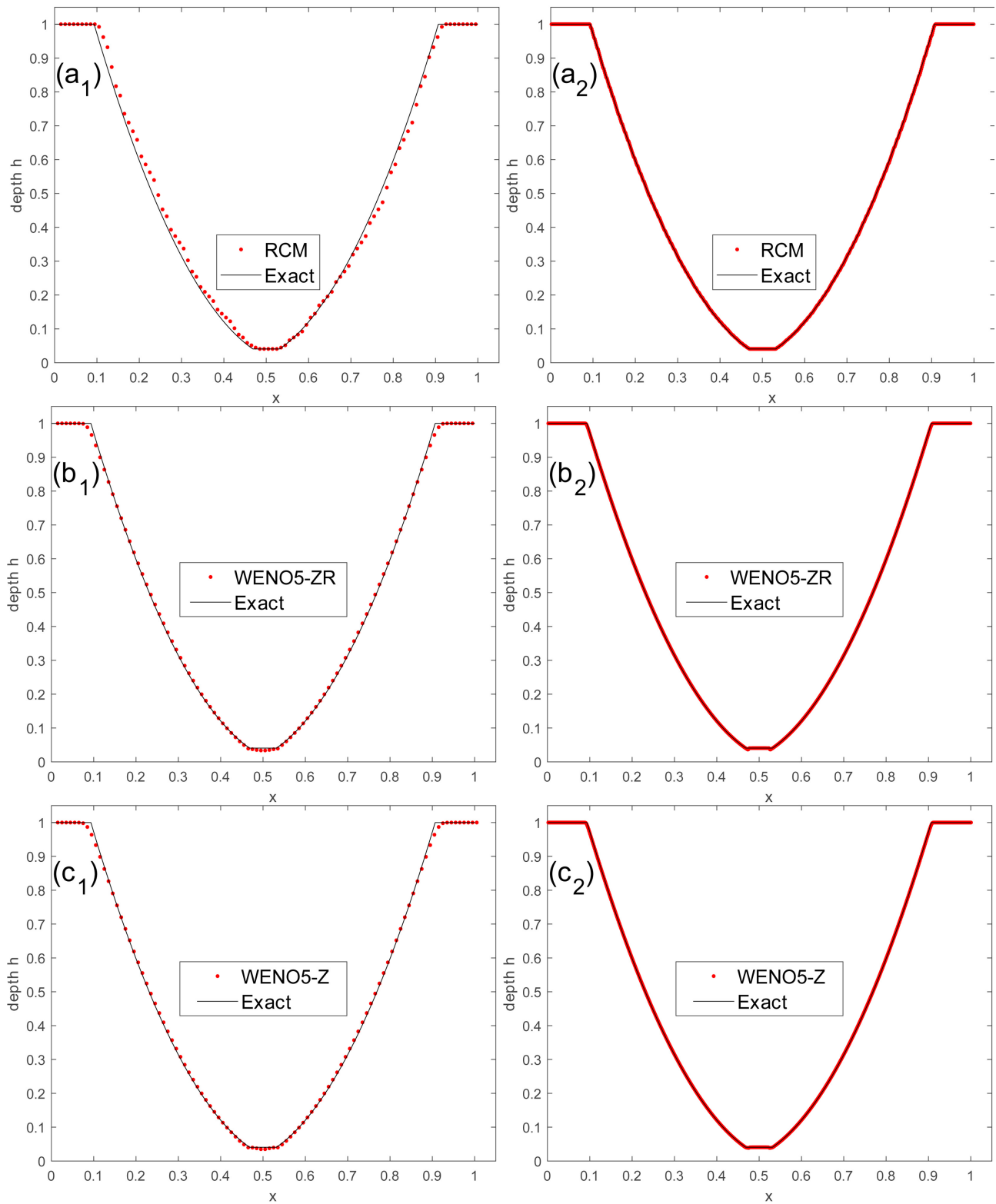


Figure 4. Test 2. Water depth profiles at $t = 0.05$ with $M = 100$ (left) and $M = 800$ (right).

4.4. Test 3: Left and Right Shock

We solve the RP (2), (3), (9) with $U_L = [1, 0.5]^T$, $U_R = [1, -0.5]^T$,

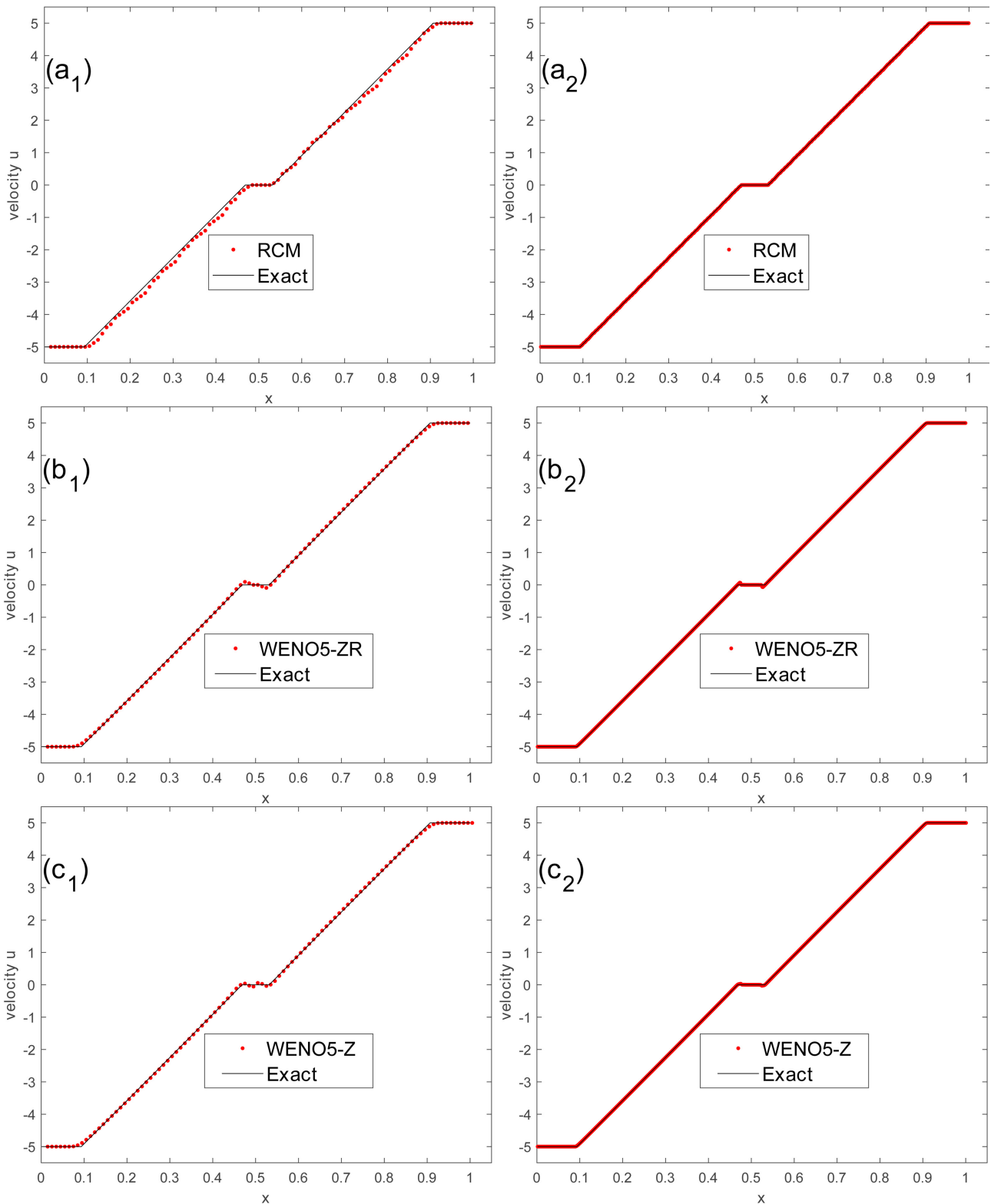


Figure 5. Test 2. Velocity profiles at $t = 0.05$ with $M = 100$ (left) and $M = 800$ (right).

$(x, t) \in [0, 1] \times (0, +\infty)$ and $x_0 = 0.5$. **Figure 6**, left, shows the structure of the solution to the RP in the $x-t$ plane. The exact solution is

$$U(x,t) = \begin{cases} U_L, & \text{if } S < S_{SL}, \\ U^*, & \text{if } S_{SL} < S < S_{SR}, \\ U_R, & \text{if } S_{SR} < S, \end{cases} \quad S = \frac{x - 0.5}{t},$$

where $S_{SL} \approx -3.018779$, $S_{SR} \approx 3.018779$, $U^* \approx [1.165630, 0]^T$.

The initial data for this test problem produce a left and a right shock that propagate in opposite directions. The two waves are symmetrical about the line $x = 0.5$. Based on the numerical results, the primary challenge faced by the methods was generating a solution that is free from spurious oscillations in the vicinity of the shocks, as well as in the constant space between them (star region).

Numerical solutions are calculated at the output times of $t_{out} = 0.1$ and $t_{out} = 100$. It is evident upon studying the exact solution that, for $t_{out} < 0.5/S_{SR} \approx 0.166$, the solution is composed of three distinct constant states that are distinguished by jump discontinuities. Conversely, the solution is constant when $t_{out} > 0.5/S_{SR} \approx 0.166$, as it is contained within the star region. In this situation, the solution is simple in structure and hence lacks significance for evaluation.

For short time simulation, the methods selected for their performance are RCM, WENO5-ZR, and WENO3-ZS. **Figure 7** illustrates the numerical solutions for water depth, whereas **Table 5** gives the L_1 errors.

Figure 7(a₁) and **Figure 7(a₂)** depict the numerical solution provided by the RCM. The shocks have been computed with infinite resolution. However, the positions of the shocks are not exact. More precisely, the left shock's location on a 100-cell mesh deviates by Δx from its exact location, whereas the locations of both shocks on an 800-cell mesh deviate by Δx . Moreover, the RCM accurately calculates values in constant states with a precision of at least eight decimal places, and the margin of error is approximately 10^{-9} . Nevertheless, the RCM has a substantially greater L_1 error (refer to **Table 5**). This is primarily attributable to inaccuracies in the shocks' position.

Among the various WENO methods, only WENO3-JS and WENO3-ZS successfully yielded a numerical solution that was free from spurious oscillations in the vicinity of the shocks on all meshes. **Figure 7(b₁)** and **Figure 7(b₂)** shows the numerical solution obtained using the WENO3-ZS scheme. In comparison with WENO3-JS, the WENO3-ZS resolved the shocks more sharply (5 - 7 cells) and resulted in a smaller L_1 error on all meshes.

The solution provided by the WENO5-ZR, WENO5-M, and WENO5-Z exhibited minor oscillations following the left shock and preceding the right shock. Refinement of the mesh increases the visibility of these oscillations, which diminish quickly. The numerical solution derived by applying the WENO5-ZR scheme is illustrated in **Figure 7(c₁)** and **Figure 7(c₂)**. The WENO5-ZR scheme resolves the shocks more accurately (3 - 5 cells), exhibits smaller amplitude oscillations, and yields smaller L_1 errors in comparison to WENO5-Z and WENO5-M. The WENO $(2k - 1)$ -JS schemes ($k = 3, 4, 5$) displayed larger amplitude oscillations in the previously stated locations, which get more intense as k increases.

The solution produced by the MWENO5-P, WENO5-P, and WENO5-NS schemes displays very slight overshoots and undershoots that become apparent upon mesh refinement, in addition to the oscillations in the aforementioned locations. To be more specific, the velocity (respectively, water depth) plot displays a slight overshoot (resp. undershoot) prior to the left shock, and both plots exhibit a slight undershoot following the right shock. Among the WENO methods investigated in this study, the WENO $(2k - 1)$ -ZS, $k = 3, 4, 5$, schemes, as well as the WENO5-ZQ scheme, had the least accurate performance in resolving shocks. The solution generated by these schemes exhibits overshoots and undershoots accompanied by spurious oscillations prior to and following both shocks.

For long time simulation, refer to **Figure 6**, right, and **Table 6**. In terms of L_1 error, RCM exhibited the lowest value, followed by WENO9-ZS (for $M = 100$), and WENO5-ZQ (for $M = 200, 400, 800$). Notice that the WENO9-ZS is more accurate on the coarser mesh than the WENO5-ZQ on the finest mesh.

Table 5. Test 3. L_1 -norm errors of several schemes at $t = 0.1$.

M	Δx	RCM	WENO3-ZS	WENO5-ZR
100	0.01	6.6563E-03	6.8179E-03	3.7364E-03
200	0.005	3.3282E-03	2.5327E-03	1.0317E-03
400	0.0025	3.3282E-03	1.5894E-03	8.4818E-04
800	0.00125	1.6641E-03	5.3050E-04	2.1727E-04

Table 6. Test 3. L_1 -norm errors of several schemes at $t = 100$.

M	Δx	RCM	WENO5-ZQ	WENO9-ZS
100	0.01	1.6932E-09	2.5483E-03	1.7626E-03
200	0.005	1.6932E-09	2.4380E-03	5.8478E-03
400	0.0025	1.6932E-09	3.0275E-03	6.5827E-03
800	0.00125	1.6932E-09	1.7795E-03	4.9842E-03

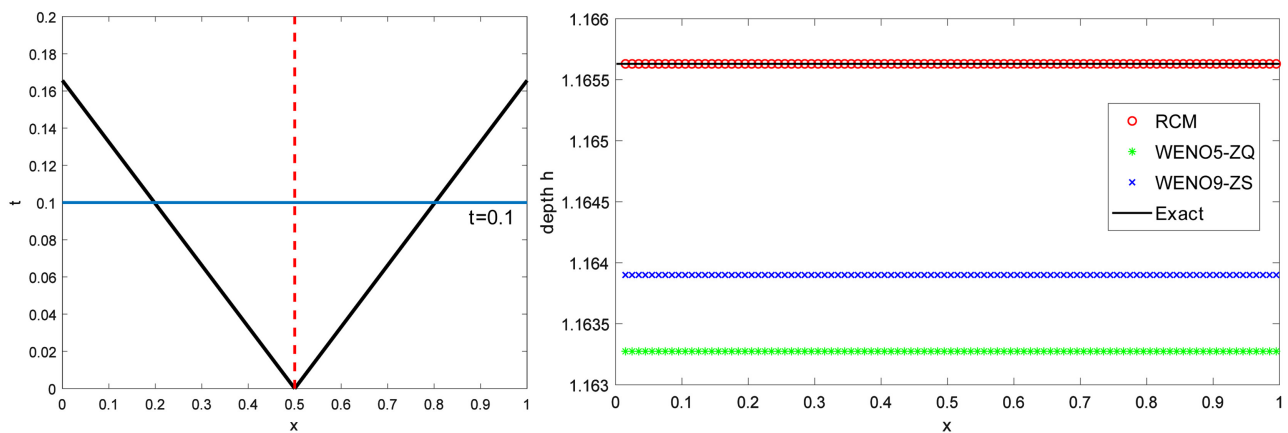


Figure 6. Test 3. Left: The structure of the solution of the RP in the $x-t$ plane. Right: Close-up view of water depth profiles at $t = 100$ with $M = 100$.

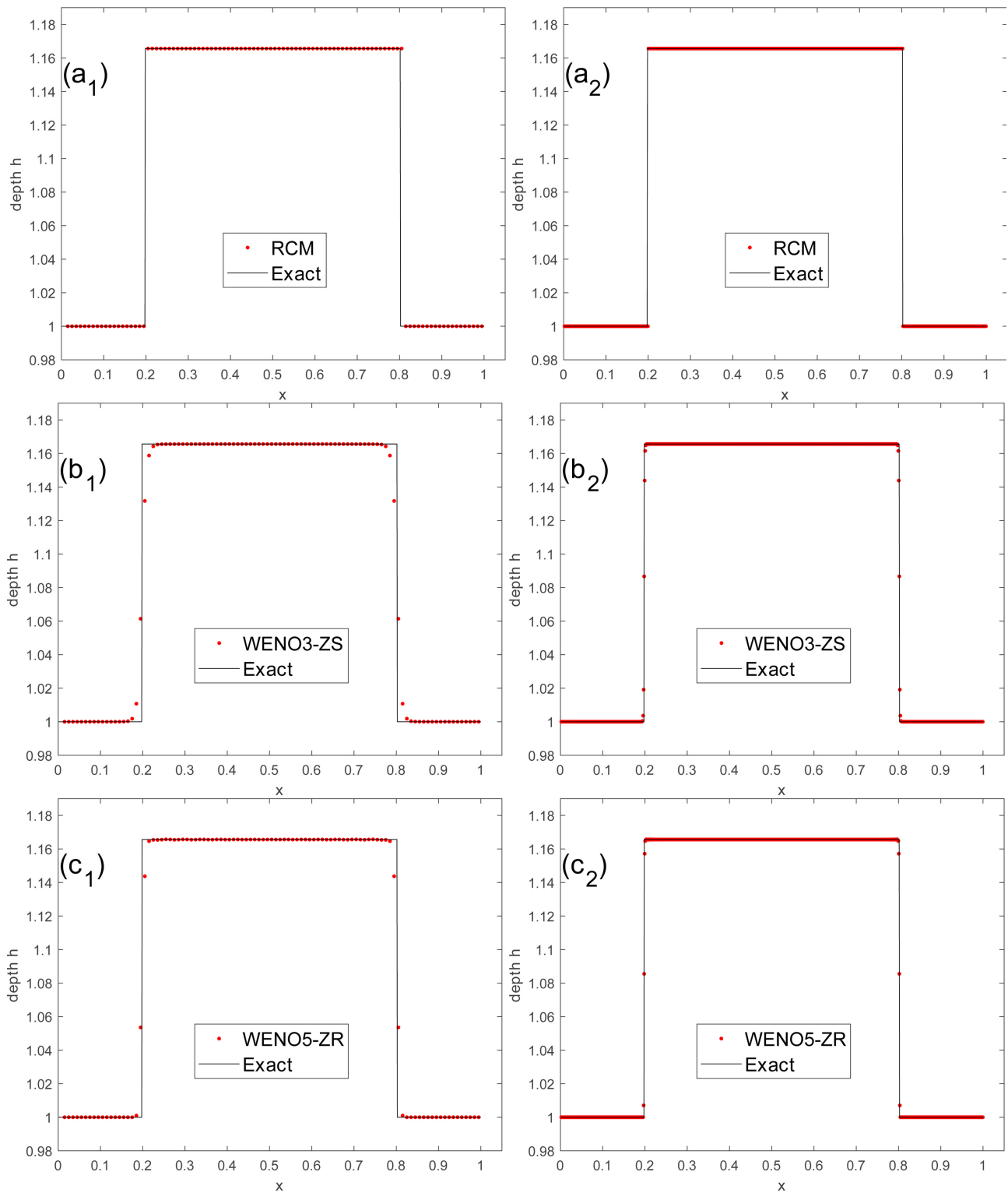


Figure 7. Test 3. Water depth profiles at $t = 0.1$ with $M = 100$ (left) and $M = 800$ (right).

4.5. Test 4: Left Shock and Right Rarefaction

We solve the RP (2), (3), (9) with $U_L = [2, 3.5]^T$, $U_R = [3, 3]^T$, $(x, t) \in [0, 1] \times (0, +\infty)$ and $x_0 = 0.5$. **Figure 8**, left, displays the structure of the

solution to the RP in the $x-t$ plane. The exact solution is

$$U(x,t) = \begin{cases} U_L, & \text{if } S < S_{SL}, \\ U^*, & \text{if } S_{SL} < S \leq S_{TR}, \\ U_{Rfan}(x/t), & \text{if } S_{TR} < S < S_{HR}, \\ U_R, & \text{if } S_{HR} \leq S, \end{cases} \quad S = \frac{x-0.5}{t},$$

where $S_{SL} \approx -3.770040$, $S_{TR} \approx 5.486301$, $S_{HR} \approx 6.424942$, $U^* \approx [2.663932, 0.996948]^T$, and U_{Rfan} is determined by (11).

We selected the initial data for Test 4 so that the solution consists of a left shock and a narrow right rarefaction. The main challenges encountered by the methods used were in producing a solution that effectively handled the shock and the rarefaction, including its tail and head. Due to the narrowness of the rarefaction, the solution contains a thin region with steep gradients that is around 0.047 units wide. This makes it difficult to adequately represent this region on a 100-cell mesh.

We compute the numerical solution at the output times of $t_{out} = 0.05$ and $t_{out} = 100$. Upon examination of the exact solution, it is obvious that, when $t_{out} < 0.5/S_{HR} \approx 0.077$, the solution has a rich structure consisting of four distinct states that can be defined by smooth regions, derivative discontinuities, and a jump discontinuity. For $t_{out} > -0.5/S_{SL} \approx 0.133$, however, the solution is located within the star region and thus constant. In this case, the solution has a simple structure and therefore lacks significance for examination.

A broad variety of numerical behaviour was observed in the results derived from the WENO methods. For short time simulation, we selected RCM, WENO5-ZR, and WENO3-ZS for presentation. The numerical solutions for water depth are plotted in **Figure 9**, while the L_1 errors are provided in **Table 7**.

Figure 9(a₁) and **Figure 9(a₂)** depict the numerical solution obtained from the RCM on both meshes. The numerical solution aligns with the exact solution in the constant states. On the 100-cell mesh, the shock has been calculated with infinite resolution, and its position is precise, but the resolution of the rarefaction is insufficient. Refining the mesh significantly enhances the accuracy of the rarefaction approximation while maintaining the sharpness of the corners. However, it also adds a minor error in the shock position.

Figure 9(b₁) and **Figure 9(b₂)** show the numerical solution acquired by the WENO5-ZR scheme on both meshes. On the 100-cell mesh, the shock is sharply resolved (2 - 3 cells), the star region is well defined, and the rarefaction is slightly curved at the edges. Nevertheless, a low-amplitude undershoot occurs prior to the rarefaction's tail. The 800-cell mesh exhibits significantly improved resolution of the shock and rarefaction, and the undershoot is practically nonexistent. The WENO5-ZR scheme also had the lowest L_1 error among all other WENO methods on all meshes.

Figure 9(c₁) and **Figure 9(c₂)** illustrate the numerical solution generated with the WENO3-ZS method on both meshes. It produced the only oscillation-free solution on all meshes among WENO methods. Compared to WENO5-ZR, the transition of the shock occupies four to five cells, and the resolution of the rarefaction

is less satisfactory, resulting in more smeared corners. Refining the mesh significantly enhances the resolution of both shock and rarefaction.

Compared to the WENO3-ZS, the WENO3-JS demonstrated a greater degree of discontinuity smearing. It also displayed minor oscillations on the 800-cell mesh following the shock. Similar to the WENO5-ZR, the WENO5-NS demonstrated comparable performance; however, there was one distinction. The water depth (resp. velocity) plot depicts a slight undershoot (resp. overshoot) preceding the shock. The WENO5-P exhibits a similar pattern to the WENO5-NS, but with the inclusion of very small amplitude oscillations following the shock.

In the WENO5-JS and WENO5-M results, spurious oscillations were observed after the shock, which persisted throughout the entire star region until the tail of the rarefaction on the 100-cell mesh. On the finest mesh, these oscillations were compressed in frequency and amplified in amplitude over a limited region following the shock before being damped. The MWENO5-P, WENO5-Z, and WENO $(2k - 1)$ -JS, $k = 4, 5$, exhibit a similar structure, but with the inclusion of an undershoot preceding the tail of the rarefaction.

The WENO5-ZS and WENO5-ZQ results demonstrated small amplitude oscillations following the shock, as well as an undershoot prior to the tail and an overshoot following the head of the rarefaction, with regard to the 800-cell mesh. The WENO $(2k - 1)$ -ZS, $k = 4, 5$, results, on the other hand, exhibited both overshoots and undershoots before and after the shock, as well as small amplitude oscillations after the shock. Additionally, there was an undershoot before the tail and an overshoot after the head of the rarefaction.

For long time simulation, the results are displayed in **Figure 8**, right, and the L_1 errors are tabulated in **Table 8**. RCM had the lowest L_1 errors, followed by WENO7-ZS (for $M = 100, 200, 400$), and WENO3-JS (for $M = 800$). The L_1 error of the WENO7-ZS scheme exhibits a noticeable rise when the mesh is refined from 400 to 800 cells.

Table 7. Test 4. L_1 -norm errors of several schemes at $t = 0.05$.

M	Δx	RCM	WENO3-ZS	WENO5-ZR
100	0.01	3.6502E-02	3.0322E-02	1.7967E-02
200	0.005	1.0190E-02	1.3883E-02	7.3292E-03
400	0.0025	4.8721E-03	6.5827E-03	3.1581E-03
800	0.00125	5.3079E-03	3.8281E-03	2.0017E-03

Table 8. Test 4. L_1 -norm errors of several schemes at $t = 100$.

M	Δx	RCM	WENO3-JS	WENO7-ZS
100	0.01	6.9576E-09	1.3338E-01	8.6709E-02
200	0.005	6.9576E-09	1.3490E-01	6.6159E-02
400	0.0025	6.9576E-09	1.3608E-01	5.9510E-02
800	0.00125	6.9576E-09	1.3675E-01	1.9880E-01

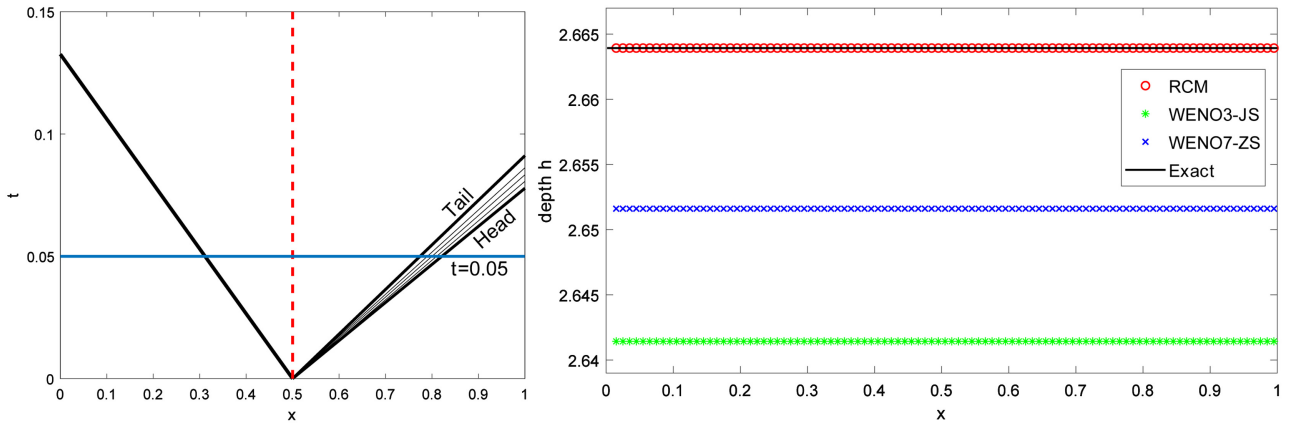


Figure 8. Test 4. Left: The structure of the solution of the RP in the $x-t$ plane. Right: Close-up view of water depth profiles at $t = 100$ with $M = 100$.

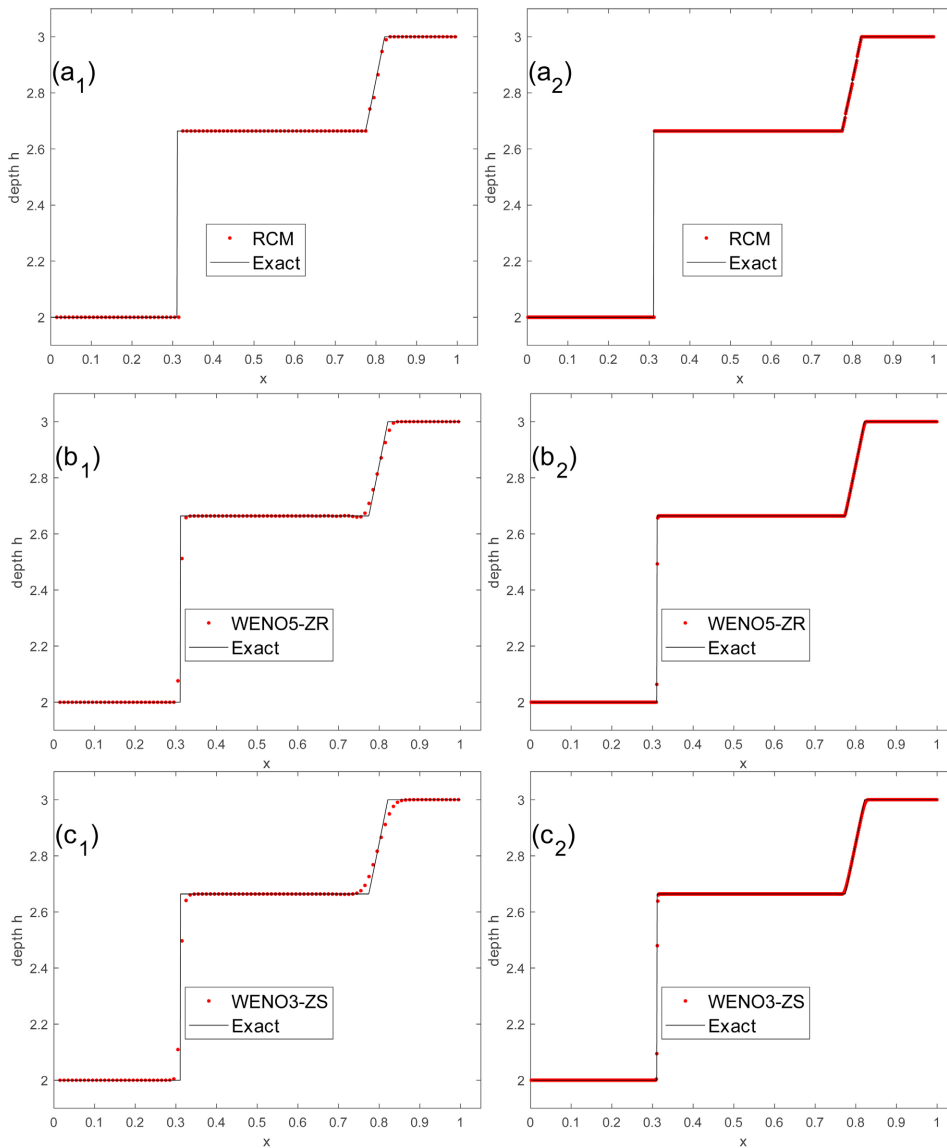


Figure 9. Test 4. Water depth profiles at $t = 0.05$ with $M = 100$ (left) and $M = 800$ (right).

4.6. Computational Cost Comparison

In this subsection, we examine the computational performance of the schemes. The computations were performed using the gfortran compiler with double precision on a system equipped with an Intel(R) Core(TM) i7-6700HQ x64-based CPU running at a clock speed of 2.60 GHz and with 16 GB of RAM. The code was not intended to optimize all procedures; therefore, it is feasible that computational costs may be decreased through improved implementations.

The same CFL coefficient, $C_{CFL} = 0.4$, was employed to compare the computational cost of the WENO schemes. **Table 9** displays the CPU times for the WENO schemes applied to Test 1 for the output time $t_{out} = 100$ on the 800-cell mesh. For these values, the numerical solution is evolved over approximately 1173×10^3 time steps, regardless of the scheme employed. The comparison of the computational cost of the schemes on the other Tests leads to similar conclusions. To reduce the impact of randomness, each Test is conducted five times under identical conditions, and the computation time provided in **Table 9** is the average of the five trials.

RCM is the most efficient scheme for these types of problems, as it is one-step scheme in time and do not require the expensive computations required in the WENO reconstruction process. It took about 2.04 minutes to complete 1,041,646 time steps. **Table 9** indicates that the WENO $(2k - 1)$ -ZS is approximately 14% - 21% slower (depending on k) than the WENO $(2k - 1)$ -JS for the same value of k . Additionally, we can observe that all of the WENO5 schemes exhibit comparable efficiency, with the exception of the WENO5-ZS and WENO5-ZR schemes. Compared to the other WENO5 schemes, the WENO5-ZS is approximately 10% more expensive, whereas the WENO5-ZR is over twice as expensive. Surprisingly, the cost of the WENO5-ZR exceeds even that of the WENO9-ZS. As anticipated, the methods with higher order accuracy (excluding WENO5-ZR) are more costly as a result of the large number of computations.

Table 9. CPU time (in minutes) analysis for various schemes as applied to Test 1.

scheme	time	scheme	time	scheme	time
WENO3-JS	55.11	WENO5-P	76.71	WENO5-ZR	159.54
WENO3-ZS	64.68	WENO5-ZQ	77.89	WENO7-JS	99.67
WENO5-M	79.22	MWENO5-P	76.69	WENO7-ZS	113.84
WENO5-Z	77.17	WENO5-JS	74.59	WENO9-JS	128.55
WENO5-NS	79.18	WENO5-ZS	85.04	WENO9-ZS	154.94

5. Concluding Remarks

In this paper, we focus on FV WENO schemes for solving the initial-value problem for homogeneous 1D SWEs. A proposal made by Titarev and Toro [20] in the context of FV WENO methods was to utilize the upwind TVD WAF flux as the numerical flux, which they implemented in the FV WENO-JS. Our motivation for

the study was to employ the upwind TVD WAF flux in conjunction with other WENO schemes that have developed since then and investigate their performance by numerically solving RPs for the homogeneous 1D SWEs. This study also included the RCM for comparative analysis with the WENO methodologies for two reasons. Firstly, for its capacity to resolve discontinuities with infinite resolution, and secondly, to illustrate its considerable similarity to the exact solution for prolonged simulations of the problems examined in this study.

To validate the use of the TVD WAF flux, we conducted a comparison with the first-order Godunov flux [1] [34] on the same test problems. Our numerical experiments, not shown here, demonstrated that the WENO schemes were generally more effective in terms of L_1 error and discontinuity resolution when used with the TVD WAF flux. In order to utilise the TVD WAF flux, we extended the existing FD WENO schemes (WENO-M, WENO-Z, WENO-NS, WENO-P, MWENO-P and WENO-ZR) into FV schemes, based on the concepts proposed in [11]. Our first objective was to outline the procedures for implementing the methods employed in this research and provide the necessary equations to facilitate their coding. As far as we know, certain ones of these have not been reported in the literature. In the case of the WENO methods, these are the explicit expressions for 1) the Smoothness Indicators (SIs) for the WENO $(2k-1)$ -ZS schemes, $k = 4, 5$, and 2) the values of the reconstruction polynomials $p_i^{ZS,k}(x)$ at the cell interfaces $x = x_{i+\frac{1}{2}}$ for the WENO $(2k-1)$ -ZS, $k = 2, 3, 4, 5$.

Our second objective was to evaluate the efficiency of the TVD WAF flux on four RPs for the SWEs, each with a different solution structure. Moreover, we sought to determine whether a single method produces the best results for all test problems in either short- or long-time simulations. When we refer to “best results,” we are primarily concerned with the methods’ capacity to accurately and sharply resolve discontinuities without oscillations, with the L_1 error and CPU cost being secondary factors to consider.

The findings indicate that, for short time simulation, both the RCM and the WENO5-ZR produce the most favourable outcomes for all test problems. The RCM did not demonstrate the lowest L_1 error; however, it resolved shock discontinuities with infinite resolution and extremely accurately approximated the corners at the endpoints of the rarefactions. Due to its randomness, the RCM’s primary drawbacks include the lack of smoothness in the rarefactions, which can be improved with mesh refinement, and the incorrect positioning of shocks. Multiple versions of the RCM may be found in the literature that utilise hybrid methods to remove undesired randomness; see, e.g., [35]-[38].

Apart from the RCM, the WENO5-ZR offered the least oscillatory solutions. Furthermore, it exhibited the lowest L_1 error compared to all other WENO methods on all meshes, albeit at a substantial increase in CPU cost. The WENO3-ZS’s performance is also worthy of discussion. Aside from Test 2, the WENO3-ZS produced oscillation-free solutions with sharp discontinuity resolution. It should

be noted, however, that it did not yield the second-best L_1 error.

For long time simulations, the RCM yielded the smallest L_1 errors for all test problems on all meshes, with the exception of Test 1 with $M = 400$. Aside from the RCM, there was no single method that consistently produced the smallest L_1 errors for all test problems.

We would like to provide a concise summary of the numerical behaviour of the methods in relation to the wave configurations that were encountered in the test problems. For Test 1 (left rarefaction, right shock), the challenging parts of the solution included the tail of the rarefaction, the constant state between the waves, and the region preceding the shock. All approaches, with the exception of RCM, WENO3-JS, and WENO3-ZS, faced difficulties in accurately approximating the tail of the rarefaction. The approximation of the tail by the schemes WENO $(2k - 1)$ -JS, WENO $(2k - 1)$ -ZS for $k = 3, 4, 5$ is inadequate. WENO $(2k - 1)$ -ZS encountered challenges with accurately approximating the constant state, while WENO $(2k - 1)$ -ZS and WENO5-ZQ were unable to resolve the shock without introducing oscillations, $k = 3, 4, 5$.

For Test 2 (left rarefaction, right rarefaction), the tails of the rarefactions presented the greatest difficulties in the solution. All approaches, with the exception of the RCM, approximated the tails with either overshoots or undershoots. Consequently, the constant state between the rarefactions is only partially realised. Certain schemes (WENO $(2k - 1)$ -ZS, $k = 3, 4, 5$, and WENO5-ZQ) approximated the heads of the rarefactions with oscillations as well.

For Test 3 (left shock, right shock), the most demanding aspects of the solution were noted to occur after the left shock and prior to the right shock. The sole approaches that did not generate oscillations in those regions were WENO3-JS, WENO3-ZS, and RCM. The remaining methods produced oscillations, rendering the outcomes of WENO $(2k - 1)$ -JS for $k = 4, 5$, WENO $(2k - 1)$ -ZS for $k = 3, 4, 5$, and WENO5-ZQ unsatisfactory. Challenges were also noted prior to the left shock and subsequent to the right shock using the methods MWENO5-P, WENO5-NS, WENO5-P, WENO $(2k - 1)$ -ZS for $k = 3, 4, 5$, and WENO5-ZQ. The outcomes generated by WENO $(2k - 1)$ -ZS, for $k = 3, 4, 5$, and WENO5-ZQ in those regions are unacceptable.

For Test 4 (left shock, right rarefaction), the most tough features of the solution were observed to happen after the shock and prior to the tail of the rarefaction. The only methods that did not produce oscillations post-shock were WENO3-ZS, WENO5-NS, WENO5-ZR, and RCM. The results obtained in that domain by the WENO3-JS, WENO5-P, and WENO5-ZQ methods are acceptable. The methods WENO7-JS, WENO9-JS, WENO $(2k - 1)$ -ZS for $k = 3, 4, 5$, MWENO5-P, WENO5-Z, WENO5-ZQ, and WENO5-ZR exhibited an undershoot preceding the tail of the rarefaction, with the outcomes of WENO9-JS, WENO $(2k - 1)$ -ZS for $k = 3, 4, 5$, and WENO5-ZQ in that region being inadequate. The less challenging parts included the parts of the solution prior to the shock, the constant state between the waves, and the head of the rarefaction. The WENO7-ZS,

WENO9-ZS, WENO5-NS, WENO5-P, and WENO5-ZQ exhibited an overshoot prior to the shock, whereas the WENO9-JS, WENO $(2k - 1)$ -ZS for $k = 3, 4, 5$, and MWENO5-P failed to accurately approximate the constant state. An overshoot following the head of the rarefaction is generated by the WENO $(2k - 1)$ -ZS procedures for $k = 3, 4, 5$, and WENO5-ZQ, which is deemed unsatisfactory.

Based on our prior discussion, for the problems studied in this research, we advocate employing the WENO5-ZR scheme or the RCM with a small spatial step size for short time simulations, particularly when the solution has a complex structure comprising both smooth areas and discontinuities. Our third selection would be the WENO3-ZS scheme. Our recommendation for long time simulation is the RCM.

Before closing, we wish to emphasize that comparing the methods' performance solely based on the L_1 error is not always adequate. It is also important to evaluate the positions of discontinuities and the oscillations in their vicinity. For instance, **the RCM and the WENO3-ZS (Tests 1, 3, and 4) did not produce the smallest L_1 error, but they did offer solutions without oscillations and with fine discontinuity resolution.**

As a final note, it has been observed that the schemes that were up to the fifth order of accuracy outperformed those of the seventh and ninth orders due to the costly calculations involved. We also tested the WENO $(2k - 1)$ -X, $k = 4, 5$, X = JS, ZS, schemes using the fourth-order non-TVD Runge-Kutta procedure [9] and the five-stage, fourth-order nonlinear version SSPRK (5,4) [39] method for time discretization with $C_{CFL} = 0.2$. We employed quadruple precision additionally to mitigate contamination of the solution by round-off errors. Overall, the numerical results obtained exhibit a consistent pattern with the previous oscillations study. However, the L_1 errors were decreased, and certain oscillations were slightly restricted. The results of these computational studies are not presented here.

This performance of the high-order methods may be attributable to the simple nature of the test problems. The procedures outlined in this paper can be implemented for any hyperbolic conservation laws with certain modifications to the code. In the future, our intention is to employ the current schemes to solve more complex models and carry out a comparative investigation.

Acknowledgements

Research of Pavlos Stampolidis has been financially supported by General Secretariat for Research and Technology (GSRT) and the Hellenic Foundation for Research and Innovation (HFRI). The authors would also like to thank the anonymous referees for their insightful comments and recommendations.

Conflicts of Interest

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

References

- [1] Toro, E.F. (2001) Shock-Capturing Methods for Free-Surface Shallow Flows. Wiley

and Sons Ltd.

- [2] Toro, E.F. (2009) Riemann Solvers and Numerical Methods for Fluid Dynamics—A Practical Introduction. 3rd Edition, Springer-Verlag. <https://doi.org/10.1007/b79761>
- [3] LeVeque, R.J. (2002) Finite Volume Methods for Hyperbolic Problems. Cambridge University Press. <https://doi.org/10.1017/cbo9780511791253>
- [4] Thomas, J.W. (1999) Numerical Partial Differential Equations: Conservation Laws and Elliptic Equations, Texts in Applied Mathematics 33. Springer.
- [5] Gousidou-Koutita, M. (2008) Numerical Methods with Applications to Ordinary and Partial Differential Equations. Lecture Notes for Postgraduate Studies, Aristotle University of Thessaloniki.
- [6] Chorin, A.J. (1976) Random Choice Solution of Hyperbolic Systems. *Journal of Computational Physics*, **22**, 517-533. [https://doi.org/10.1016/0021-9991\(76\)90047-4](https://doi.org/10.1016/0021-9991(76)90047-4)
- [7] Glimm, J. (1965) Solutions in the Large for Nonlinear Hyperbolic Systems of Equations. *Communications on Pure and Applied Mathematics*, **18**, 697-715. <https://doi.org/10.1002/cpa.3160180408>
- [8] Liu, X., Osher, S. and Chan, T. (1994) Weighted Essentially Non-Oscillatory Schemes. *Journal of Computational Physics*, **115**, 200-212. <https://doi.org/10.1006/jcph.1994.1187>
- [9] Jiang, G. and Shu, C. (1996) Efficient Implementation of Weighted ENO Schemes. *Journal of Computational Physics*, **126**, 202-228. <https://doi.org/10.1006/jcph.1996.0130>
- [10] Shu, C. (2020) Essentially Non-Oscillatory and Weighted Essentially Non-Oscillatory Schemes. *Acta Numerica*, **29**, 701-762. <https://doi.org/10.1017/s0962492920000057>
- [11] Shu, C. (1998) Essentially Non-Oscillatory and Weighted Essentially Non-Oscillatory Schemes for Hyperbolic Conservation Laws. In: Quarteroni, A., Ed., *Lecture Notes in Mathematics*, Springer Berlin Heidelberg, 325-432. <https://doi.org/10.1007/bfb0096355>
- [12] Henrick, A.K., Aslam, T.D. and Powers, J.M. (2005) Mapped Weighted Essentially Non-Oscillatory Schemes: Achieving Optimal Order Near Critical Points. *Journal of Computational Physics*, **207**, 542-567. <https://doi.org/10.1016/j.jcp.2005.01.023>
- [13] Borges, R., Carmona, M., Costa, B. and Don, W.S. (2008) An Improved Weighted Essentially Non-Oscillatory Scheme for Hyperbolic Conservation Laws. *Journal of Computational Physics*, **227**, 3191-3211. <https://doi.org/10.1016/j.jcp.2007.11.038>
- [14] Ha, Y., Ho Kim, C., Ju Lee, Y. and Yoon, J. (2013) An Improved Weighted Essentially Non-Oscillatory Scheme with a New Smoothness Indicator. *Journal of Computational Physics*, **232**, 68-86. <https://doi.org/10.1016/j.jcp.2012.06.016>
- [15] Kim, C.H., Ha, Y. and Yoon, J. (2016) Modified Non-Linear Weights for Fifth-Order Weighted Essentially Non-Oscillatory Schemes. *Journal of Scientific Computing*, **67**, 299-323. <https://doi.org/10.1007/s10915-015-0079-3>
- [16] Zhu, J. and Qiu, J. (2017) A New Type of Finite Volume WENO Schemes for Hyperbolic Conservation Laws. *Journal of Scientific Computing*, **73**, 1338-1359. <https://doi.org/10.1007/s10915-017-0486-8>
- [17] Rathan, S. and Naga Raju, G. (2018) A Modified Fifth-Order WENO Scheme for Hyperbolic Conservation Laws. *Computers & Mathematics with Applications*, **75**, 1531-1549. <https://doi.org/10.1016/j.camwa.2017.11.020>
- [18] Zhu, J. and Shu, C. (2018) A New Type of Multi-Resolution WENO Schemes with Increasingly Higher Order of Accuracy. *Journal of Computational Physics*, **375**, 659-683. <https://doi.org/10.1016/j.jcp.2018.09.003>

- [19] Gu, J., Chen, X. and Jung, J. (2023) Fifth-Order Weighted Essentially Non-Oscillatory Schemes with New Z-Type Nonlinear Weights for Hyperbolic Conservation Laws. *Computers & Mathematics with Applications*, **134**, 140-166. <https://doi.org/10.1016/j.camwa.2023.01.009>
- [20] Titarev, V.A. and Toro, E.F. (2005) WENO Schemes Based on Upwind and Centred TVD Fluxes. *Computers & Fluids*, **34**, 705-720. <https://doi.org/10.1016/j.compfluid.2004.05.009>
- [21] Toro, E.F. (1989) A Weighted Average Flux Method for Hyperbolic Conservation Laws. *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*, **423**, 401-418. <https://doi.org/10.1098/rspa.1989.0062>
- [22] Toro, E.F. (1992) The Weighted Average Flux Method Applied to the Euler Equations. *Philosophical Transactions: Physical Sciences and Engineering*, **341**, 499-530. <https://doi.org/10.1098/rsta.1992.0113>
- [23] Toro, E.F. (1992) Riemann Problems and the WAF Method for Solving the Two-Dimensional Shallow Water Equations. *Philosophical Transactions: Physical Sciences and Engineering*, **338**, 43-68. <http://doi.org/10.1098/rsta.1992.0002>
- [24] Shu, C. (1988) Total-Variation-Diminishing Time Discretizations. *SIAM Journal on Scientific and Statistical Computing*, **9**, 1073-1084. <https://doi.org/10.1137/0909073>
- [25] Balsara, D.S. and Shu, C. (2000) Monotonicity Preserving Weighted Essentially Non-Oscillatory Schemes with Increasingly High Order of Accuracy. *Journal of Computational Physics*, **160**, 405-452. <https://doi.org/10.1006/jcph.2000.6443>
- [26] Roe, P.-L. and Pike, J. (1985) Efficient Construction and Utilisation of Approximate Riemann Solutions. *Proceedings of the Sixth International Symposium on Computing Methods in Applied Sciences and Engineering, VI*, Versailles, 1983, 499-518.
- [27] Harten, A., Lax, P.D. and van Leer, B. (1983) On Upstream Differencing and Godunov-Type Schemes for Hyperbolic Conservation Laws. *SIAM Review*, **25**, 35-61. <https://doi.org/10.1137/1025002>
- [28] Roe, P.-L. (1985) Some Contributions to the Modelling of Discontinuous Flows. *Proceedings of the Fifteenth Summer Seminar on Applied Mathematics*, La Jolla, 27 June-8 July 1983, 163-193. <https://ui.adsabs.harvard.edu/abs/1985ams..conf..163R/abstract>
- [29] Gerolymos, G.A., Sénéchal, D. and Vallet, I. (2009) Very-High-Order WENO Schemes. *Journal of Computational Physics*, **228**, 8481-8524. <https://doi.org/10.1016/j.jcp.2009.07.039>
- [30] Castro, M., Costa, B. and Don, W.S. (2011) High Order Weighted Essentially Non-Oscillatory WENO-Z Schemes for Hyperbolic Conservation Laws. *Journal of Computational Physics*, **230**, 1766-1792. <https://doi.org/10.1016/j.jcp.2010.11.028>
- [31] Wang, Z., Zhu, J. and Zhao, N. (2020) A New Fifth-Order Finite Difference Well-Balanced Multi-Resolution WENO Scheme for Solving Shallow Water Equations. *Computers & Mathematics with Applications*, **80**, 1387-1404. <https://doi.org/10.1016/j.camwa.2020.07.003>
- [32] Hammersley, J.M. and Handscomb, D.C. (1964) Monte Carlo Methods. Springer Dordrecht. <https://doi.org/10.1007/978-94-009-5819-7>
- [33] Burden, R.L. and Faires, J.D. (2010) Numerical Analysis. 9th Edition, Brooks/Cole, Cengage Learning.
- [34] Godunov, S.K. (1959) Finite Difference Methods for the Computation of Discontinuous Solutions of the Equations of Fluid Dynamics. *Matematicheskii Sbornik*, **47**, 271-306.

- [35] Toro, E.F. and Roe, P.L. (1989) A Hybrid Scheme for the Euler Equations Using the Random Choice and Roe's Methods, In: Morton, K.W. and Baines, M.J., Eds., *Numerical Methods for Fluid Dynamics III*, Oxford University Press, 701-708.
- [36] Toro, E.F. (1989) Random-Choice Based Hybrid Methods for One and Two Dimensional Gas Dynamics. In: Ballmann, J. and Jeltsch, R., Eds., *Nonlinear Hyperbolic Equations—Theory, Computation Methods, and Applications*, Vieweg + Teubner Verlag, 630-639. https://doi.org/10.1007/978-3-322-87869-4_61
- [37] Zahran, Y.H. (2008) RCM-TVD Hybrid Scheme for Hyperbolic Conservation Laws. *International Journal for Numerical Methods in Fluids*, **57**, 745-760. <https://doi.org/10.1002/flid.1641>
- [38] Zahran, Y.H. and Abdalla, A.H. (2024) Central Random Choice Methods for Hyperbolic Conservation Laws. *Ricerche di Matematica*, **73**, 2091-2130. <https://doi.org/10.1007/s11587-022-00747-9>
- [39] Gottlieb, S. (2005) On High Order Strong Stability Preserving Runge-Kutta and Multi Step Time Discretizations. *Journal of Scientific Computing*, **25**, 105-128. <https://doi.org/10.1007/bf02728985>

Appendices

A. Structure of the Solution of the Riemann Problem (RP) for a 2×2 System

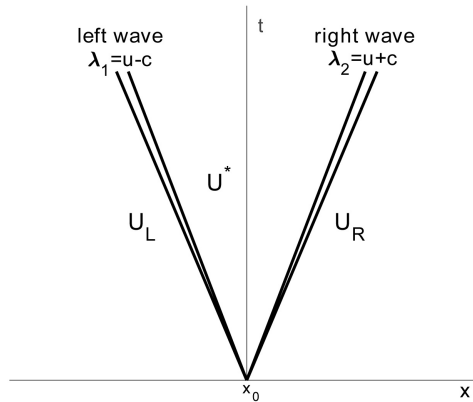


Figure A1. The structure of the RP solution in the $x-t$ plane for the 1D SWEs. The eigenvalues $\lambda_1 = u - c$ and $\lambda_2 = u + c$ correspond to two distinct wave families.

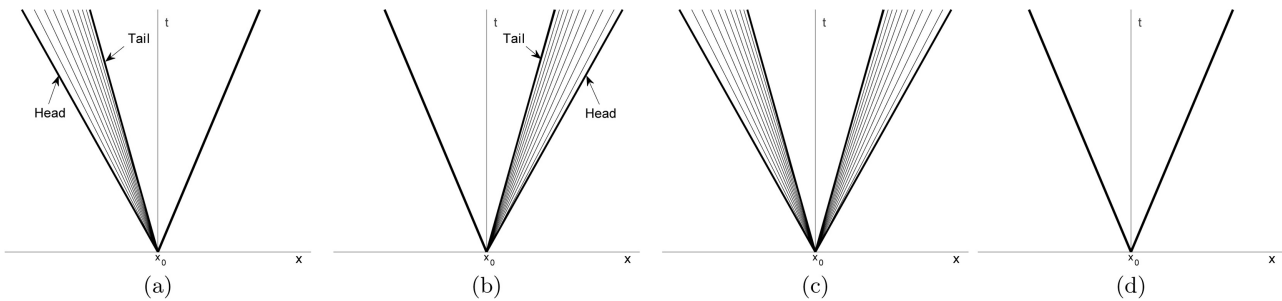


Figure A2. Potential wave patterns in the solution of the RP for the 1D SWEs: (a) left rarefaction, right shock (b) left shock, right rarefaction (c) left rarefaction, right rarefaction (d) left shock, right shock

B. Tables of Coefficients

Table A1. The coefficients $c_{k,s,j}$ of Procedure 1.

k	s	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$
*2	-1	3/2	-1/2			
	0	1/2	1/2			
	1	-1/2	3/2			
*3	-1	11/6	-7/6	1/3		
	0	1/3	5/6	-1/6		
	1	-1/6	5/6	1/3		
	2	1/3	-7/6	11/6		
*4	-1	25/12	-23/12	13/12	-1/4	
	0	1/4	13/12	-5/12	1/12	
	1	-1/12	7/12	7/12	-1/12	
	2	1/12	-5/12	13/12	1/4	

Continued

	3	-1/4	13/12	-23/12	25/12	
	-1	137/60	-163/60	137/60	-21/20	1/5
	0	1/5	77/60	-43/60	17/60	-1/20
*5	1	-1/20	9/20	47/60	-13/60	1/30
	2	1/30	-13/60	47/60	9/20	-1/20
	3	-1/20	17/60	-43/60	77/60	1/5
	4	1/5	-21/20	137/60	-163/60	137/60

Table A2. optimal weights $d_{k,s}$.

	$s=0$	$s=1$	$s=2$	$s=3$	$s=4$
$k=2$	2/3	1/3			
$k=3$	3/10	3/5	1/10		
$k=4$	4/35	18/35	12/35	1/35	
$k=5$	5/126	20/63	10/21	10/23	1/126

C. WENO (2k - 1)-JS Scheme (k = 2, 3, 4, 5)

For $k = 2$, (23) yields [11]

$$\beta_0^{JS}(u, i) = (u_{i+1} - u_i)^2, \quad \beta_1^{JS}(u, i) = (u_i - u_{i-1})^2. \tag{30}$$

For $k = 3$, (23) yields [11]

$$\begin{aligned} \beta_0^{JS}(u, i) &= \frac{13}{12}(u_i - 2u_{i+1} + u_{i+2})^2 + \frac{1}{4}(3u_i - 4u_{i+1} + u_{i+2})^2, \\ \beta_1^{JS}(u, i) &= \frac{13}{12}(u_{i-1} - 2u_i + u_{i+1})^2 + \frac{1}{4}(u_{i-1} - u_{i+1})^2, \\ \beta_2^{JS}(u, i) &= \frac{13}{12}(u_{i-2} - 2u_{i-1} + u_i)^2 + \frac{1}{4}(u_{i-2} - 4u_{i-1} + 3u_i)^2. \end{aligned} \tag{31}$$

For $k = 4$, (23) yields [25]

$$\begin{aligned} \beta_0^{JS}(u, i) &= \frac{1}{240} [u_i (2107u_i - 9402u_{i+1} + 7042u_{i+2} - 1854u_{i+3}) + u_{i+1} (11003u_{i+1} \\ &\quad - 17246u_{i+2} + 4642u_{i+3}) + u_{i+2} (7043u_{i+2} - 3882u_{i+3}) + 547u_{i+3}^2], \\ \beta_1^{JS}(u, i) &= \frac{1}{240} [u_{i-1} (547u_{i-1} - 2522u_i + 1922u_{i+1} - 494u_{i+2}) + u_i (3443u_i \\ &\quad - 5966u_{i+1} + 1602u_{i+2}) + u_{i+1} (2843u_{i+1} - 1642u_{i+2}) + 267u_{i+2}^2], \\ \beta_2^{JS}(u, i) &= \frac{1}{240} [u_{i-2} (267u_{i-2} - 1642u_{i-1} + 1602u_i - 494u_{i+1}) + u_{i-1} (2843u_{i-1} \\ &\quad - 5966u_i + 1922u_{i+1}) + u_i (3443u_i - 2522u_{i+1}) + 547u_{i+1}^2], \\ \beta_3^{JS}(u, i) &= \frac{1}{240} [u_{i-3} (547u_{i-3} - 3882u_{i-2} + 4642u_{i-1} - 1854u_i) + u_{i-2} (7043u_{i-2} \\ &\quad - 17246u_{i-1} + 7042u_i) + u_{i-1} (11003u_{i-1} - 9402u_i) + 2107u_i^2], \end{aligned} \tag{32}$$

where β_i^{JS} is equal to $\frac{1}{240} \text{IS}_{3-i}^4$, in [25], $i = 0, 1, 2, 3$.

For $k = 5$, (23) yields [25]

$$\begin{aligned} \beta_0^{JS}(u, i) &= \frac{1}{5040} \left[u_i (107918u_i - 649501u_{i+1} + 758823u_{i+2} - 411487u_{i+3} + 86329u_{i+4}) \right. \\ &\quad + u_{i+1} (1020563u_{i+1} - 2462076u_{i+2} + 1358458u_{i+3} - 288007u_{i+4}) + u_{i+2} (1521393u_{i+2} \\ &\quad - 1704396u_{i+3} + 364863u_{i+4}) + u_{i+3} (482963u_{i+3} - 208501u_{i+4}) + 22658u_{i+4}^2 \left. \right], \\ \beta_1^{JS}(u, i) &= \frac{1}{5040} \left[u_{i-1} (22658u_{i-1} - 140251u_i + 165153u_{i+1} - 88297u_{i+2} + 18079u_{i+3}) \right. \\ &\quad + u_i (242723u_i - 611976u_{i+1} + 337018u_{i+2} - 70237u_{i+3}) + u_{i+1} (406293u_{i+1} \\ &\quad - 464976u_{i+2} + 99213u_{i+3}) + u_{i+2} (138563u_{i+2} - 60871u_{i+3}) + 6908u_{i+3}^2 \left. \right], \\ \beta_2^{JS}(u, i) &= \frac{1}{5040} \left[u_{i-2} (6908u_{i-2} - 51001u_{i-1} + 67923u_i - 38947u_{i+1} + 8209u_{i+2}) \right. \\ &\quad + u_{i-1} (104963u_{i-1} - 299076u_i + 179098u_{i+1} - 38947u_{i+2}) + u_i (231153u_i \\ &\quad - 299076u_{i+1} + 67923u_{i+2}) + u_{i+1} (104963u_{i+1} - 51001u_{i+2}) + 6908u_{i+2}^2 \left. \right], \tag{33} \\ \beta_3^{JS}(u, i) &= \frac{1}{5040} \left[u_{i-3} (6908u_{i-3} - 60871u_{i-2} + 99213u_{i-1} - 70237u_i + 18079u_{i+1}) \right. \\ &\quad + u_{i-2} (138563u_{i-2} - 464976u_{i-1} + 337018u_i - 88297u_{i+1}) + u_{i-1} (406293u_{i-1} \\ &\quad - 611976u_i + 165153u_{i+1}) + u_i (242723u_i - 140251u_{i+1}) + 22658u_{i+1}^2 \left. \right], \\ \beta_4^{JS}(u, i) &= \frac{1}{5040} \left[u_{i-4} (22658u_{i-4} - 208501u_{i-3} + 364863u_{i-2} - 288007u_{i-1} + 86329u_i) \right. \\ &\quad + u_{i-3} (482963u_{i-3} - 1704396u_{i-2} + 1358458u_{i-1} - 411487u_i) + u_{i-2} (1521393u_{i-2} \\ &\quad - 2462076u_{i-1} + 758823u_i) + u_{i-1} (1020563u_{i-1} - 649501u_i) + 107918u_i^2 \left. \right], \end{aligned}$$

where β_i^{JS} is equal to $\frac{1}{5040} \text{IS}_{4-i}^5$, in [25], $i = 0, \dots, 4$.

D. WENO5-NS Scheme

According to [14], the new SIs, β_s^{NS} , are defined as follows:

$$\begin{aligned} \beta_s^{NS} = \beta_s^{NS}(u, i) &= \xi \left[(1-s)u_{i-2+s} + (2s-3)u_{i-1+s} + (2-s)u_{i+s} \right] \\ &\quad + \left[u_{i-2+s} - 2u_{i-1+s} + u_{i+s} \right], \quad s = 0, 1, 2, \quad \xi \in (0, 1]. \tag{34} \end{aligned}$$

We assign $\xi = 0.4$ in accordance with [14].

E. WENO (2k - 1)-ZS Scheme (k = 2, 3, 4, 5)

E.1. The Reconstruction Polynomials

In order to maintain uniform notation in accordance with the present study, we made small modifications to the notation used in the original work by Zhu and Shu [18].

The reconstruction polynomials $p_i^{ZS(k,s)}(x)$, $k = 2, 3, 4, 5$, with varying degrees are defined as

$$p_i^{ZS(k,s)}(x) = \begin{cases} p_i^{(1,0)}(x) & \text{if } s = 0, \\ \frac{1}{\gamma_{s,s}} p_i^{(2s+1,s)}(x) - \sum_{j=0}^{s-1} \frac{\gamma_{j,s}}{\gamma_{s,s}} p_i^{ZS(k,j)}(x) & \text{if } s = 1, \dots, k-1, \end{cases} \tag{35}$$

with $\sum_{j=0}^s \gamma_{j,s} = 1$ and $\gamma_{s,s} \neq 0$. In these expressions, $\gamma_{j,s}$ for $j=0, \dots, s$ and $s=1, \dots, k-1$ are the linear weights, and $p_i^{(k,s)}(x)$ is the reconstruction polynomial that was mentioned in 3.3.1. Hence,

$$\deg\left(p_i^{ZS(k,s)}\right) \begin{cases} = 0 & \text{if } s = 0, \\ \leq 2s & \text{if } s = 1, \dots, k-1. \end{cases}$$

According to [18], the linear weights are assigned as

$$\gamma_{j,s} = \frac{\bar{\gamma}_{j,s}}{\sum_{l=0}^s \bar{\gamma}_{l,s}}, \text{ where } \bar{\gamma}_{j,s} = 10^j, \text{ for } j=0, \dots, s \text{ and } s=1, \dots, k-1. \quad (36)$$

As an illustration, when $k=3$, the reconstruction polynomials $p_i^{ZS(3,s)}(x)$, $s=0, 1, 2$, are defined as

$$p_i^{ZS(3,0)}(x) = p_i^{(1,0)}(x),$$

$$p_i^{ZS(3,1)}(x) = \frac{1}{\gamma_{1,1}} p_i^{(3,1)}(x) - \frac{\gamma_{0,1}}{\gamma_{1,1}} p_i^{ZS(3,0)}(x), \quad \gamma_{0,1} + \gamma_{1,1} = 1, \quad \gamma_{1,1} \neq 0,$$

$$p_i^{ZS(3,2)}(x) = \frac{1}{\gamma_{2,2}} p_i^{(5,2)}(x) - \frac{\gamma_{0,2}}{\gamma_{2,2}} p_i^{ZS(3,0)}(x) - \frac{\gamma_{1,2}}{\gamma_{2,2}} p_i^{ZS(3,1)}(x), \quad \sum_{j=0}^2 \gamma_{j,2} = 1, \quad \gamma_{2,2} \neq 0.$$

By applying (36), we derive the linear weights as follows: $\bar{\gamma}_{0,1} = \bar{\gamma}_{0,2} = 1$, $\bar{\gamma}_{1,1} = \bar{\gamma}_{1,2} = 10$, $\bar{\gamma}_{2,2} = 100$, and $\gamma_{0,1} = 1/11$, $\gamma_{1,1} = 10/11$, $\gamma_{0,2} = 1/111$, $\gamma_{1,2} = 10/111$, and $\gamma_{2,2} = 100/111$.

Table A3 below displays the linear weights $\gamma_{j,s}$ for $j=0, \dots, s$, and $s=1, 2, 3, 4$.

Table A3. Linear weights $\gamma_{j,s}$ in (36).

	$j=0$	$j=1$	$j=2$	$j=3$	$j=4$
$s=1$	1/11	10/11			
$s=2$	1/111	10/111	100/111		
$s=3$	1/1111	10/1111	100/1111	1000/1111	
$s=4$	1/11111	10/11111	100/11111	1000/11111	10000/11111

E.2. The Smoothness Indicators (SIs)

The SIs β_0^{ZS} and β_1^{ZS} are universal for all values of k . The SI β_2^{ZS} is universal for $k=3, 4, 5$. The SI β_3^{ZS} is universal for $k=4, 5$. Lastly, the SI β_4^{ZS} is necessary only for $k=5$. Their precise expression is specified below. In order to abbreviate the expressions, we employed the following substitutes:

$$\mu_1 = \gamma_{1,1}\gamma_{2,2}, \quad \mu_2 = \gamma_{1,3}\gamma_{2,2} - \gamma_{1,2}\gamma_{2,3}, \quad \mu_3 = \mu_1\gamma_{3,3}, \quad \mu_4 = \gamma_{2,4}\gamma_{3,3} - \gamma_{2,3}\gamma_{3,4},$$

$$\mu_5 = \gamma_{2,2}(\gamma_{1,4}\gamma_{3,3} - \gamma_{1,3}\gamma_{3,4}) - \gamma_{1,2}\mu_4, \quad \mu_6 = \gamma_{3,3}\gamma_{3,4}, \quad \mu_7 = -5 + 6\gamma_{0,1}.$$

In order to determine β_0^{ZS} , the subsequent calculations must be performed:

$$\varsigma_0 = \varsigma_0(u, i) = (u_i - u_{i-1})^2, \quad \varsigma_1 = \varsigma_1(u, i) = (u_{i+1} - u_i)^2,$$

$$\bar{\gamma}_{-1,0} = \begin{cases} 1, & \varsigma_0 \geq \varsigma_1, \\ 10, & \text{otherwise,} \end{cases} \quad \bar{\gamma}_{0,0} = 11 - \bar{\gamma}_{-1,0},$$

$$\gamma_{-1,0} = \frac{\bar{\gamma}_{-1,0}}{\bar{\gamma}_{-1,0} + \bar{\gamma}_{0,0}}, \quad \gamma_{0,0} = 1 - \gamma_{-1,0},$$

$$\sigma_0 = \gamma_{-1,0} \left(1 + \frac{|\varsigma_0 - \varsigma_1|^{k-1}}{\varsigma_0 + \epsilon} \right), \quad \sigma_1 = \gamma_{0,0} \left(1 + \frac{|\varsigma_0 - \varsigma_1|^{k-1}}{\varsigma_1 + \epsilon} \right), \quad \sigma = \sigma_0 + \sigma_1.$$

Ultimately, the definition of β_0^{ZS} is as follows

$$\beta_0^{ZS} = \beta_0^{ZS}(u, i) = \frac{1}{\sigma^2} (\sigma_0 (u_i - u_{i-1}) + \sigma_1 (u_{i+1} - u_i))^2. \tag{37}$$

For $k = 2$,

$$\beta_1^{ZS} = \beta_1^{ZS}(u, i) = \sum_{l=1}^2 \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \Delta x^{2l-1} \left(\frac{d^l p_i^{ZS(2,1)}(x)}{dx^l} \right)^2 dx$$

$$= \frac{1}{3\gamma_{1,1}^2} [4u_{i-1}^2 + 13u_i^2 + 4u_{i+1}^2 + 5u_{i-1}u_{i+1} - 13u_i(u_{i-1} + u_{i+1})] \tag{38}$$

For $k = 3$,

$$\beta_2^{ZS} = \beta_2^{ZS}(u, i) = \sum_{l=1}^4 \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \Delta x^{2l-1} \left(\frac{d^l p_i^{ZS(3,2)}(x)}{dx^l} \right)^2 dx$$

$$= \frac{1}{5040\mu_1^2} \left\{ \gamma_{1,1} u_{i-2} [6908\gamma_{1,1} u_{i-2} + (1344\gamma_{1,2} - 51001\gamma_{1,1}) u_{i-1} \right.$$

$$+ 3(22641\gamma_{1,1} - 616\gamma_{1,2}) u_i + (504\gamma_{1,2} - 38947\gamma_{1,1}) u_{i+1} + 8209\gamma_{1,1} u_{i+2}]$$

$$+ u_{i-1} [(104963\gamma_{1,1}^2 - 17976\gamma_{1,1}\gamma_{1,2} + 6720\gamma_{1,2}^2) u_{i-1} - 12(24923\gamma_{1,1}^2$$

$$- 4718\gamma_{1,1}\gamma_{1,2} + 1820\gamma_{1,2}^2) u_i + 2(89549\gamma_{1,1}^2 - 11256\gamma_{1,1}\gamma_{1,2} + 4200\gamma_{1,2}^2) u_{i+1} \tag{39}$$

$$+ \gamma_{1,1} (504\gamma_{1,2} - 38947\gamma_{1,1}) u_{i+2}] + u_i [3(77051\gamma_{1,1}^2 - 18256\gamma_{1,1}\gamma_{1,2}$$

$$+ 7280\gamma_{1,2}^2) u_i + 12(4718\gamma_{1,1}\gamma_{1,2} - 24923\gamma_{1,1}^2 - 1820\gamma_{1,2}^2) u_{i+1}$$

$$+ 3\gamma_{1,1} (22641\gamma_{1,1} - 616\gamma_{1,2}) u_{i+2}] + u_{i+1} [(104963\gamma_{1,1}^2 - 17976\gamma_{1,1}\gamma_{1,2}$$

$$+ 6720\gamma_{1,2}^2) u_{i+1} + \gamma_{1,1} (1344\gamma_{1,2} - 51001\gamma_{1,1}) u_{i+2}] + 6908\gamma_{1,1}^2 u_{i+2}^2 \left. \right\}$$

For $k = 4$,

$$\beta_3^{ZS} = \beta_3^{ZS}(u, i) = \sum_{l=1}^6 \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \Delta x^{2l-1} \left(\frac{d^l p_i^{ZS(4,3)}(x)}{dx^l} \right)^2 dx$$

$$= \frac{1}{59875200(\mu_1\gamma_{3,3})^2} \left\{ \mu_1 u_{i-3} [84070496\mu_1 u_{i-3} + 3\gamma_{1,1} u_{i-2} (-323333323\gamma_{2,2}$$

$$+ 10090784\gamma_{2,3}) + 3u_{i-1} [\gamma_{1,1} (761142961\gamma_{2,2} - 35635226\gamma_{2,3}) - 823680\mu_2] \right.$$

$$+ 2u_i [\gamma_{1,1} (-1403126266\gamma_{2,2} + 67308021\gamma_{2,3}) + 1473120\mu_2]$$

$$+ 6u_{i+1} [\gamma_{1,1} (317088638\gamma_{2,2} - 11933383\gamma_{2,3}) - 79200\mu_2] \left. \right\}$$

$$\begin{aligned}
 &+ 3u_{i+2}\gamma_{1,1}(-225623953\gamma_{2,2} + 4539194\gamma_{2,3}) + 99022657\mu_1u_{i+3} \\
 &+ 3\gamma_{1,1}u_{i-2} [9\gamma_{1,1}u_{i-2} [108444169\gamma_{2,2}^2 + 22\gamma_{2,3}(-554063\gamma_{2,2} + 138160\gamma_{2,3})] \\
 &- 3u_{i-1} [-31680\mu_2(101\gamma_{2,2} - 56\gamma_{2,3}) + \gamma_{1,1}(1588486617\gamma_{2,2}^2 \\
 &+ 22\gamma_{2,3}(-11054161\gamma_{2,2} + 3060060\gamma_{2,3}))] + 4u_i [-23760\mu_2(139\gamma_{2,2} - 77\gamma_{2,3}) \\
 &+ \gamma_{1,1}(1506944981\gamma_{2,2}^2 + 22\gamma_{2,3}(-10835023\gamma_{2,2} + 3056535\gamma_{2,3}))] \\
 &- 9u_{i+1} [-10560\mu_2(38\gamma_{2,2} - 21\gamma_{2,3}) + \gamma_{1,1}(464678369\gamma_{2,2}^2 \\
 &+ 44\gamma_{2,3}(-1412311\gamma_{2,2} + 389470\gamma_{2,3}))] + 6\gamma_{1,1}u_{i+2} [252791247\gamma_{2,2}^2 \\
 &+ 22\gamma_{2,3}(-989309\gamma_{2,2} + 246270\gamma_{2,3})] - \mu_1u_{i+3} (225623953\gamma_{2,2} - 4539194\gamma_{2,3}) \\
 &+ 3u_{i-1} [90u_{i-1} [-1056\gamma_{1,1}\mu_2(844\gamma_{2,2} - 749\gamma_{2,3}) + 295680\mu_2^2 \\
 &+ \gamma_{1,1}^2(67162828\gamma_{2,2}^2 + 11\gamma_{2,3}(-1320195\gamma_{2,2} + 419852\gamma_{2,3}))] \\
 &- 10u_i [-3168\gamma_{1,1}(7852\gamma_{2,2} - 7077\gamma_{2,3})\mu_2 + 8648640\mu_2^2 \\
 &+ \gamma_{1,1}^2(1581062354\gamma_{2,2}^2 + 11\gamma_{2,3}(-33061655\gamma_{2,2} + 10766736\gamma_{2,3}))] \\
 &+ 15u_{i+1} [-12672\gamma_{1,1}\mu_2(529\gamma_{2,2} - 469\gamma_{2,3}) + 2217600\mu_2^2 + \gamma_{1,1}^2(751570629\gamma_{2,2}^2 \\
 &+ 44\gamma_{2,3}(-3412339\gamma_{2,2} + 1074588\gamma_{2,3}))] - 9\gamma_{1,1}u_{i+2} [-10560\mu_2(38\gamma_{2,2} - 21\gamma_{2,3}) \\
 &+ \gamma_{1,1}(464678369\gamma_{2,2}^2 + 44\gamma_{2,3}(-1412311\gamma_{2,2} + 389470\gamma_{2,3}))] \\
 &+ 2\mu_1u_{i+3} [\gamma_{1,1}(317088638\gamma_{2,2} - 11933383\gamma_{2,3}) - 79200\mu_2] \\
 &+ 2u_i [10u_i [-9504\gamma_{1,1}\mu_2(3733\gamma_{2,2} - 3423\gamma_{2,3}) + 12972960\mu_2^2 \\
 &+ \gamma_{1,1}^2(1607739169\gamma_{2,2}^2 + 198\gamma_{2,3}(-2066797\gamma_{2,2} + 693459\gamma_{2,3}))] \\
 &- 15u_{i+1} [-3168\gamma_{1,1}\mu_2(7852\gamma_{2,2} - 7077\gamma_{2,3}) + 8648640\mu_2^2 \\
 &+ \gamma_{1,1}^2(1581062354\gamma_{2,2}^2 + 11\gamma_{2,3}(-33061655\gamma_{2,2} + 10766736\gamma_{2,3}))] \\
 &+ 6\gamma_{1,1}u_{i+2} [-23760\mu_2(139\gamma_{2,2} - 77\gamma_{2,3}) + \gamma_{1,1}(1506944981\gamma_{2,2}^2 \\
 &+ 22\gamma_{2,3}(-10835023\gamma_{2,2} + 3056535\gamma_{2,3}))] \\
 &+ \mu_1u_{i+3} [\gamma_{1,1}(-1403126266\gamma_{2,2} + 67308021\gamma_{2,3}) + 1473120\mu_2] \\
 &+ 3u_{i+1} [90u_{i+1} [-1056\gamma_{1,1}\mu_2(844\gamma_{2,2} - 749\gamma_{2,3}) + 295680\mu_2^2 \\
 &+ \gamma_{1,1}^2(67162828\gamma_{2,2}^2 + 11\gamma_{2,3}(-1320195\gamma_{2,2} + 419852\gamma_{2,3}))] \\
 &- 3\gamma_{1,1}u_{i+2} [-31680\mu_2(101\gamma_{2,2} - 56\gamma_{2,3}) \\
 &+ \gamma_{1,1}(1588486617\gamma_{2,2}^2 + 22\gamma_{2,3}(-11054161\gamma_{2,2} + 3060060\gamma_{2,3}))] \\
 &+ \mu_1u_{i+3} [\gamma_{1,1}(761142961\gamma_{2,2} - 35635226\gamma_{2,3}) - 823680\mu_2] \\
 &+ 3\gamma_{1,1}^2u_{i+2} [9u_{i+2} [108444169\gamma_{2,2}^2 + 22\gamma_{2,3}(-554063\gamma_{2,2} + 138160\gamma_{2,3})] \\
 &+ \gamma_{2,2}u_{i+3}(-323333323\gamma_{2,2} + 10090784\gamma_{2,3})] + 84070496\mu_1^2u_{i+3}^2 \}
 \end{aligned}$$

(40)

For $k = 5$,

$$\begin{aligned}
 \beta_4^{ZS} &= \beta_4^{ZS}(u, i) = \sum_{l=1}^8 \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \Delta x^{2l-1} \left(\frac{d^l P_i^{ZS(5,4)}(x)}{dx^l} \right)^2 dx \\
 &= \frac{1}{1307674368000(\mu_3 \gamma_{4,4})^2} \left\{ \mu_3 u_{i-4} \left[1904918771192 \mu_3 u_{i-4} + \mu_1 u_{i-3} (-29871456729101 \gamma_{3,3} \right. \right. \\
 &\quad + 975087457920 \gamma_{3,4}) + 5 \gamma_{1,1} u_{i-2} \left[\gamma_{2,2} (20216075320673 \gamma_{3,3} - 1096248578370 \gamma_{3,4}) \right. \\
 &\quad - 25337282880 \mu_4] + u_{i-1} \left[\gamma_{1,1} (\gamma_{2,2} (-192700060973723 \gamma_{3,3} + 12578111025930 \gamma_{3,4}) \right. \\
 &\quad + 431640507480 \mu_4) + 9859449600 \mu_5] + 5 u_i \left[\gamma_{1,1} (\gamma_{2,2} (45272942020727 \gamma_{3,3} \right. \\
 &\quad - 3011533902588 \gamma_{3,4}) - 103120611048 \mu_4) - 2075673600 \mu_5] \\
 &\quad + u_{i+1} \left[\gamma_{1,1} (\gamma_{2,2} (-167888314942259 \gamma_{3,3} + 9918999944940 \gamma_{3,4}) + 252370776840 \mu_4) \right. \\
 &\quad + 518918400 \mu_5] + \gamma_{1,1} u_{i+2} \left[\gamma_{2,2} (76858903972891 \gamma_{3,3} - 3412405211850 \gamma_{3,4}) \right. \\
 &\quad - 41721814680 \mu_4] + 5 \mu_1 u_{i+3} (-3976300410337 \gamma_{3,3} + 95823837570 \gamma_{3,4}) \\
 &\quad + 2227506474493 \mu_3 u_{i+4}] + \mu_1 u_{i-3} \left[\mu_1 u_{i-3} (119979314906981 \gamma_{3,3}^2 - 10976930658570 \mu_6 \right. \\
 &\quad + 1836099632640 \gamma_{3,4}^2) - 2 \gamma_{1,1} u_{i-2} \left[-595140 (1335547 \gamma_{3,3} - 555456 \gamma_{3,4}) \mu_4 \right. \\
 &\quad + \gamma_{2,2} (414718505225338 \gamma_{3,3}^2 - 53990687515995 \mu_6 + 10592399661480 \gamma_{3,4}^2) \left. \right] \\
 &\quad + 10 u_{i-1} \left[-103783680 (119 \gamma_{3,3} - 52 \gamma_{3,4}) \mu_5 + \gamma_{1,1} (-2184 (256807433 \gamma_{3,3} - 106905678 \gamma_{3,4}) \mu_4 \right. \\
 &\quad + \gamma_{2,2} (161084839253509 \gamma_{3,3}^2 - 24189551470794 \mu_6 + 4987008680472 \gamma_{3,4}^2) \left. \right] \\
 &\quad + 10 u_i \left[207567360 (71 \gamma_{3,3} - 31 \gamma_{3,4}) \mu_5 - \gamma_{1,1} (-13104 (53913605 \gamma_{3,3} - 22436007 \gamma_{3,4}) \mu_4 \right. \\
 &\quad + \gamma_{2,2} (192310346872991 \gamma_{3,3}^2 - 29393569930323 \mu_6 + 6128855529888 \gamma_{3,4}^2) \left. \right] \\
 &\quad + 2 u_{i+1} \left[-518918400 \mu_5 (23 \gamma_{3,3} - 10 \gamma_{3,4}) + \gamma_{1,1} (-10920 \mu_4 (172252127 \gamma_{3,3} - 71600298 \gamma_{3,4}) \right. \\
 &\quad + \gamma_{2,2} (723357784442063 \gamma_{3,3}^2 - 101179854539310 \mu_6 + 20775647561760 \gamma_{3,4}^2) \left. \right] \\
 &\quad + 2 \gamma_{1,1} u_{i+2} \left[5460 (65581237 \gamma_{3,3} - 27235164 \gamma_{3,4}) \mu_4 - \gamma_{2,2} (335381350482226 \gamma_{3,3}^2 \right. \\
 &\quad - 37910478663105 \mu_6 + 7391440700280 \gamma_{3,4}^2) \left. \right] + 2 \mu_1 u_{i+3} (87766773846323 \gamma_{3,3}^2 \\
 &\quad - 6491576601750 \mu_6 + 1081327414440 \gamma_{3,4}^2) + 5 \mu_3 u_{i+4} (-3976300410337 \gamma_{3,3} \\
 &\quad + 95823837570 \gamma_{3,4}) \left. \right] + \gamma_{1,1} u_{i-2} \left[7 \gamma_{1,1} u_{i-2} \left[-1560 \gamma_{2,2} (928599317 \gamma_{3,3} - 658226844 \gamma_{3,4}) \mu_4 \right. \right. \\
 &\quad + 256049164800 \mu_4^2 + \gamma_{2,2} (209113420631753 \gamma_{3,3}^2 - 38126807480520 \mu_6 \\
 &\quad + 9135336796560 \gamma_{3,4}^2) \left. \right] + 14 u_{i-1} \left[74131200 (\gamma_{2,2} (809 \gamma_{3,3} - 606 \gamma_{3,4}) - 336 \mu_4) \mu_5 \right. \\
 &\quad - \gamma_{1,1} (-2340 \gamma_{2,2} (1951860751 \gamma_{3,3} - 1459149252 \gamma_{3,4}) \mu_4 + 945191332800 \mu_4^2 \\
 &\quad + \gamma_{2,2} (414118316081794 \gamma_{3,3}^2 - 87253085764125 \mu_6 + 22302352102680 \gamma_{3,4}^2) \left. \right] \\
 &\quad + 140 u_i \left[-14826240 (\gamma_{2,2} (557 \gamma_{3,3} - 417 \gamma_{3,4}) - 231 \mu_4) \mu_5 \right. \\
 &\quad + \gamma_{1,1} (-78 \gamma_{2,2} (7582721225 \gamma_{3,3} - 5720892144 \gamma_{3,4}) \mu_4 + 125880337440 \mu_4^2
 \end{aligned}$$

$$\begin{aligned}
 & + \gamma_{2,2}^2 \left(50317355051084\gamma_{3,3}^2 - 10876947937377\mu_6 + 2821001004432\gamma_{3,4}^2 \right) \Big] \\
 & + 70u_{i+1} \left[14826240 \left(\gamma_{2,2} \left(305\gamma_{3,3} - 228\gamma_{3,4} \right) - 126\mu_4 \right) \mu_5 - \gamma_{1,1} \left(-468\gamma_{2,2} \left(1498341157\gamma_{3,3} \right. \right. \right. \\
 & \left. \left. \left. - 1118550312\gamma_{3,4} \right) \mu_4 + 144359392320\mu_4^2 + \gamma_{2,2}^2 \left(76901526633986\gamma_{3,3}^2 \right. \right. \right. \\
 & \left. \left. \left. - 15412987720467\mu_6 + 3914450580456\gamma_{3,4}^2 \right) \right) \right] + 14u_{i+2}\gamma_{1,1} \left[-1560\gamma_{2,2} \left(553209863\gamma_{3,3} \right. \right. \right. \\
 & \left. \left. \left. - 391766364\gamma_{3,4} \right) \mu_4 + 152135755200\mu_4^2 + \gamma_{2,2}^2 \left(180786151740479\gamma_{3,3}^2 \right. \right. \right. \\
 & \left. \left. \left. - 29915214958920\mu_6 + 7098378215760\gamma_{3,4}^2 \right) \right] + 2\mu_4u_{i+3} \left[5460 \left(65581237\gamma_{3,3} \right. \right. \right. \\
 & \left. \left. \left. - 27235164\gamma_{3,4} \right) \mu_4 - \gamma_{2,2} \left(335381350482226\gamma_{3,3}^2 - 37910478663105\mu_6 \right. \right. \right. \\
 & \left. \left. \left. + 7391440700280\gamma_{3,4}^2 \right) \right] + \mu_3u_{i+4} \left[\gamma_{2,2} \left(76858903972891\gamma_{3,3} - 3412405211850\gamma_{3,4} \right) \right. \right. \\
 & \left. \left. - 41721814680\mu_4 \right] + u_{i-1} \left[7u_{i-1} \left[-148262400\gamma_{1,1} \left(\gamma_{2,2} \left(5407\gamma_{3,3} - 5064\gamma_{3,4} \right) - 4494\mu_4 \right) \mu_5 \right. \right. \right. \\
 & \left. \left. \left. + 249080832000\mu_5^2 + \gamma_{1,1}^2 \left(-3120\gamma_{2,2} \left(4855374941\gamma_{3,3} - 3920979150\gamma_{3,4} \right) \mu_4 \right. \right. \right. \right. \\
 & \left. \left. \left. + 3890516572800\mu_4^2 + \gamma_{2,2}^2 \left(836484368637131\gamma_{3,3}^2 - 205803487578870\mu_6 \right. \right. \right. \right. \\
 & \left. \left. \left. + 56577966307200\gamma_{3,4}^2 \right) \right) \right] + 70u_i \left[29652480\gamma_{1,1} \left(13\gamma_{2,2} \left(639\gamma_{3,3} - 604\gamma_{3,4} \right) - 7077\mu_4 \right) \mu_5 \right. \right. \\
 & \left. \left. - 80951270400\mu_5^2 - \gamma_{1,1}^2 \left(-6864\gamma_{2,2} \left(605844418\gamma_{3,3} - 495924825\gamma_{3,4} \right) \mu_4 \right. \right. \right. \\
 & \left. \left. \left. + 1108543138560\mu_4^2 + \gamma_{2,2}^2 \left(207139067201783\gamma_{3,3}^2 - 52892178362667\mu_6 \right. \right. \right. \right. \\
 & \left. \left. \left. + 14798743633440\gamma_{3,4}^2 \right) \right) \right] + 14u_{i+1} \left[-148262400\gamma_{1,1} \left(\gamma_{2,2} \left(3391\gamma_{3,3} - 3174\gamma_{3,4} \right) - 2814\mu_4 \right) \mu_5 \right. \right. \\
 & \left. \left. + 155675520000\mu_5^2 + \gamma_{1,1}^2 \left(-21840\gamma_{2,2} \left(599670701\gamma_{3,3} - 482602230\gamma_{3,4} \right) \mu_4 \right. \right. \right. \\
 & \left. \left. \left. + 3319187414400\mu_4^2 + \gamma_{2,2}^2 \left(805195803373277\gamma_{3,3}^2 - 192849849114090\mu_6 \right. \right. \right. \right. \\
 & \left. \left. \left. + 52760258155800\gamma_{3,4}^2 \right) \right) \right] + 70\gamma_{1,1}u_{i+2} \left[14826240 \left(\gamma_{2,2} \left(305\gamma_{3,3} - 228\gamma_{3,4} \right) - 126\mu_4 \right) \mu_5 \right. \right. \\
 & \left. \left. - \gamma_{1,1} \left(-468\gamma_{2,2} \left(1498341157\gamma_{3,3} - 1118550312\gamma_{3,4} \right) \mu_4 + 144359392320\mu_4^2 \right. \right. \right. \\
 & \left. \left. \left. + \gamma_{2,2}^2 \left(76901526633986\gamma_{3,3}^2 - 15412987720467\mu_6 + 3914450580456\gamma_{3,4}^2 \right) \right) \right] \\
 & + 2\mu_4u_{i+3} \left[-518918400 \left(23\gamma_{3,3} - 10\gamma_{3,4} \right) \mu_5 + \gamma_{1,1} \left(-10920 \left(172252127\gamma_{3,3} - 71600298\gamma_{3,4} \right) \mu_4 \right. \right. \right. \\
 & \left. \left. \left. + \gamma_{2,2} \left(723357784442063\gamma_{3,3}^2 - 101179854539310\mu_6 + 20775647561760\gamma_{3,4}^2 \right) \right) \right] \\
 & + \mu_3u_{i+4} \left[\gamma_{1,1} \left(\gamma_{2,2} \left(-167888314942259\gamma_{3,3} + 9918999944940\gamma_{3,4} \right) + 252370776840\mu_4 \right) \right. \right. \\
 & \left. \left. + 518918400\mu_5 \right] + 5u_i \left[35u_i \left[-23721984\gamma_{1,1} \left(\gamma_{2,2} \left(3908\gamma_{3,3} - 3733\gamma_{3,4} \right) - 3423\mu_4 \right) \mu_5 \right. \right. \right. \\
 & \left. \left. \left. + 32380508160\mu_5^2 + \gamma_{1,1}^2 \left(-1872\gamma_{2,2} \left(655806001\gamma_{3,3} - 545634408\gamma_{3,4} \right) \mu_4 + 342712985472\mu_4^2 \right. \right. \right. \right. \\
 & \left. \left. \left. + \gamma_{2,2}^2 \left(52297392889139\gamma_{3,3}^2 - 14048901736824\mu_6 + 4012916965824\gamma_{3,4}^2 \right) \right) \right] \\
 & + 14u_{i+1} \left[29652480\gamma_{1,1} \left(13\gamma_{2,2} \left(639\gamma_{3,3} - 604\gamma_{3,4} \right) - 7077\mu_4 \right) \mu_5 - 80951270400\mu_5^2 \right. \right. \\
 & \left. \left. - \gamma_{1,1}^2 \left(-6864\gamma_{2,2} \left(605844418\gamma_{3,3} - 495924825\gamma_{3,4} \right) \mu_4 + 1108543138560\mu_4^2 \right. \right. \right. \\
 & \left. \left. \left. + \gamma_{2,2}^2 \left(207139067201783\gamma_{3,3}^2 - 52892178362667\mu_6 + 14798743633440\gamma_{3,4}^2 \right) \right) \right] \\
 & + 28\gamma_{1,1}u_{i+2} \left[-14826240 \left(\gamma_{2,2} \left(557\gamma_{3,3} - 417\gamma_{3,4} \right) - 231\mu_4 \right) \mu_5 + \gamma_{1,1} \left(-78\gamma_{2,2} \left(7582721225\gamma_{3,3} \right. \right. \right.
 \end{aligned}$$

$$\begin{aligned}
 & -5720892144\gamma_{3,4})\mu_4 + 125880337440\mu_4^2 + \gamma_{2,2}^2 (50317355051084\gamma_{3,3}^2 \\
 & - 10876947937377\mu_6 + 2821001004432\gamma_{3,4}^2)) - 2\mu_1 u_{i+3} [-207567360(71\gamma_{3,3} - 31\gamma_{3,4})\mu_5 \\
 & + \gamma_{1,1}(-13104(53913605\gamma_{3,3} - 22436007\gamma_{3,4})\mu_4 + \gamma_{2,2}(192310346872991\gamma_{3,3}^2 \\
 & - 29393569930323\mu_6 + 6128855529888\gamma_{3,4}^2))] + \mu_3 u_{i+4} [\gamma_{1,1}(\gamma_{2,2}(45272942020727\gamma_{3,3} \\
 & - 3011533902588\gamma_{3,4}) - 103120611048\mu_4) - 2075673600\mu_5] \\
 & + u_{i+1} [7u_{i+1} [-148262400\gamma_{1,1}(\gamma_{2,2}(5407\gamma_{3,3} - 5064\gamma_{3,4}) - 4494\mu_4)\mu_5 + 249080832000\mu_5^2 \\
 & + \gamma_{1,1}^2(-3120\gamma_{2,2}(4855374941\gamma_{3,3} - 3920979150\gamma_{3,4})\mu_4 + 3890516572800\mu_4^2 \\
 & + \gamma_{2,2}^2(836484368637131\gamma_{3,3}^2 - 205803487578870\mu_6 + 56577966307200\gamma_{3,4}^2))] \\
 & + 14\gamma_{1,1}u_{i+2} [74131200(\gamma_{2,2}(809\gamma_{3,3} - 606\gamma_{3,4}) - 336\mu_4)\mu_5 \\
 & - \gamma_{1,1}(-2340\gamma_{2,2}(1951860751\gamma_{3,3} - 1459149252\gamma_{3,4})\mu_4 + 945191332800\mu_4^2 \\
 & + \gamma_{2,2}^2(414118316081794\gamma_{3,3}^2 - 87253085764125\mu_6 + 22302352102680\gamma_{3,4}^2))] \\
 & + 10\mu_1 u_{i+3} [-103783680(119\gamma_{3,3} - 52\gamma_{3,4})\mu_5 + \gamma_{1,1}(-2184(256807433\gamma_{3,3} \\
 & - 106905678\gamma_{3,4})\mu_4 + \gamma_{2,2}(161084839253509\gamma_{3,3}^2 - 24189551470794\mu_6 \\
 & + 4987008680472\gamma_{3,4}^2))] + \mu_3 u_{i+4} [\gamma_{1,1}(\gamma_{2,2}(-192700060973723\gamma_{3,3} \\
 & + 12578111025930\gamma_{3,4}) + 431640507480\mu_4 + 9859449600\mu_5] \\
 & + \gamma_{1,1}^2 u_{i+2} [7u_{i+2} [-1560\gamma_{2,2}(928599317\gamma_{3,3} - 658226844\gamma_{3,4})\mu_4 \\
 & + 256049164800\mu_4^2 + \gamma_{2,2}^2(209113420631753\gamma_{3,3}^2 - 38126807480520\mu_6 \\
 & + 9135336796560\gamma_{3,4}^2)] + 2\gamma_{2,2}u_{i+3} [595140(1335547\gamma_{3,3} - 555456\gamma_{3,4})\mu_4 \\
 & - \gamma_{2,2}(414718505225338\gamma_{3,3}^2 - 53990687515995\mu_6 + 10592399661480\gamma_{3,4}^2)] \\
 & + 5\gamma_{2,2}\gamma_{3,3}u_{i+4} [\gamma_{2,2}(20216075320673\gamma_{3,3} - 1096248578370\gamma_{3,4}) - 25337282880\mu_4] \\
 & + \mu_1^2 u_{i+3} [u_{i+3}(119979314906981\gamma_{3,3}^2 - 10976930658570\mu_6 + 1836099632640\gamma_{3,4}^2) \\
 & + \gamma_{3,3}u_{i+4}(-29871456729101\gamma_{3,3} + 975087457920\gamma_{3,4})] + 1904918771192\mu_3^2 u_{i+4}^2 \}
 \end{aligned} \tag{41}$$

By replacing the linear weights $\gamma_{j,s}$ with the values provided in **Table A3**, we obtain the following simplified equations for the SIs β_s^{ZS} :

$$\begin{aligned}
 \beta_1^{ZS} &= \frac{121}{300} [u_{i-1}(4u_{i-1} - 13u_i + 5u_{i+1}) + 13u_i(u_i - u_{i+1}) + 4u_{i+1}^2] \\
 \beta_2^{ZS} &= \frac{1}{16800000} [111u_{i-2}(255596u_{i-2} - 1882109u_{i-1} + 2506375u_i \\
 & - 1439191u_{i+1} + 303733u_{i+2}) + u_{i-1}(424037849u_{i-1} \\
 & - 1206143300u_i + 726731902u_{i+1} - 159750201u_{i+2}) \\
 & + 25u_i(37117427u_i - 48245732u_{i+1} + 11128305u_{i+2}) \\
 & + u_{i+1}(424037849u_{i+1} - 208914099u_{i+2}) + 28371156u_{i+2}^2]
 \end{aligned}$$

$$\beta_3^{ZS} = \frac{1}{544320000000} \left[1111u_{i-3} (8491120096u_{i-3} - 97664521317u_{i-2} + 229547541705u_{i-1} - 282073107490u_i + 191433202530u_{i+1} - 68226643977u_{i+2} + 10001288357u_{i+3}) + 3u_{i-2} (108318401819397u_{i-2} - 526783972204035u_{i-1} + 665995002622780u_i - 463180938156135u_{i+1} + 168768132312474u_{i+2} - 25266600486149u_{i+3}) + 15u_{i-1} (132817919788890u_{i-1} - 346936096805390u_i + 248112598843305u_{i+1} - 92636187631227u_{i+2} + 14178819200722u_{i+3}) + 10u_i (351943966663390u_i - 520404145208085u_{i+1} + 199798500786834u_{i+2} - 31338322242139u_{i+3}) + 15u_{i+1} (132817919788890u_{i+1} - 105356794440807u_{i+2} + 17001821255617u_{i+3}) + 33u_{i+2} (9847127438127u_{i+2} - 3288038884339u_{i+3}) + 9433634426656u_{i+3}^2 \right]$$

$$\beta_4^{ZS} = \frac{1}{13076743680000000000} \left[11111u_{i-4} (2116552466714312u_{i-4} - 330818433551292091u_{i-3} + 1117014403587143165u_{i-2} - 2127116096129228023u_{i-1} + 2498409223132612145u_i - 1854387058384611409u_{i+1} + 850188099852426551u_{i+2} - 220371067878570685u_{i+3} + 24749824438091723u_{i+4}) + u_{i-3} (14678728436577047871371u_{i-3} - 101090775024726955052366u_{i-2} + 195941714347760860810270u_{i-1} - 233862646070102292844760u_i + 176156582666533561821346u_{i+1} - 81890842384883649463442u_{i+2} + 21512776142651584290346u_{i+3} - 2448542935198798881035u_{i+4}) + u_{i-2} (177496072207200489844871u_{i-2} - 701052011634493421392406u_{i-1} + 851355174080462444206660u_i - 651587620251860128618250u_{i+1} + 307416343785383671515586u_{i+2} - 81890842384883649463442u_{i+3} + 9446439977460311408161u_{i+4}) + u_{i-1} (705578585708975218148867u_{i-1} - 1745629288613829197913320u_i + 1359251933617741158923578u_{i+1} - 651587620251860128618250u_{i+2} + 176156582666533561821346u_{i+3} - 20604094605711417365399u_{i+4}) + 5u_i (220075387145048518601665u_i - 349125857722765839582664u_{i+1} + 170271034816092488841332u_{i+2} - 46772529214020458568952u_{i+3} + 5551964975645290708619u_{i+4}) + u_{i+1} (705578585708975218148867u_{i+1} - 701052011634493421392406u_{i+2} + 195941714347760860810270u_{i+3} - 23634386944091852563553u_{i+4}) + u_{i+2} (177496072207200489844871u_{i+2} - 101090775024726955052366u_{i+3} + 12411147038256747706315u_{i+4}) + u_{i+3} (14678728436577047871371u_{i+3} - 3675723615188406423101u_{i+4}) + 235170453457662720632u_{i+4}^2 \right]$$

E.3. The Values of the Reconstruction Polynomials at the Cell Interfaces

The definition for the new final reconstruction polynomial $p_i^{ZS,k}(x)$,

$k = 2, 3, 4, 5$, in cell I_i is stated as

$$p_i^{ZS,k}(x) = \sum_{s=0}^{k-1} \omega_{s,k}^{ZS} p_i^{ZS(k,s)}(x). \tag{42}$$

The $(2k - 1)$ -th order reconstruction is achieved by assigning

$$u_{i+\frac{1}{2}}^- = p_i^{ZS,k} \left(x_{i+\frac{1}{2}} \right), \quad u_{i-\frac{1}{2}}^+ = p_i^{ZS,k} \left(x_{i-\frac{1}{2}} \right).$$

The latter thus yields $u_{i+\frac{1}{2}}^+ = p_{i+1}^{ZS,k} \left(x_{i+\frac{1}{2}} \right)$. Consequently, in order to determine the values $p_i^{ZS,k} \left(x_{i\pm\frac{1}{2}} \right)$, it is necessary to first compute the values $p_i^{ZS(k,s)} \left(x_{i\pm\frac{1}{2}} \right)$, as per (42). These values are derived from (35) and are presented below. We employed an algebraic manipulator (Mathematica) to obtain them.

By examining the definition of $p_i^{ZS(k,s)}(x)$ in (35), it becomes evident that $p_i^{ZS(k,s)} = p_i^{ZS(k-1,s)}$ for $s = 0, \dots, k-2$ and $k = 3, 4, 5$. Therefore, the left extrapolated value $u_{i+\frac{1}{2}}^-$ will only require the values $p_i^{ZS(2,0)} \left(x_{i+\frac{1}{2}} \right)$, $p_i^{ZS(2,1)} \left(x_{i+\frac{1}{2}} \right)$, $p_i^{ZS(3,2)} \left(x_{i+\frac{1}{2}} \right)$, $p_i^{ZS(4,3)} \left(x_{i+\frac{1}{2}} \right)$, and $p_i^{ZS(5,4)} \left(x_{i+\frac{1}{2}} \right)$, while the right extrapolated value $u_{i+\frac{1}{2}}^+$ will only require the values $p_{i+1}^{ZS(2,0)} \left(x_{i+\frac{1}{2}} \right)$, $p_{i+1}^{ZS(2,1)} \left(x_{i+\frac{1}{2}} \right)$, $p_{i+1}^{ZS(3,2)} \left(x_{i+\frac{1}{2}} \right)$, $p_{i+1}^{ZS(4,3)} \left(x_{i+\frac{1}{2}} \right)$, and $p_{i+1}^{ZS(5,4)} \left(x_{i+\frac{1}{2}} \right)$, depending on the value of k .

For $k = 2, s = 0, 1$ (35) yields

$$p_i^{ZS(2,0)} \left(x_{i+\frac{1}{2}} \right) = u_i, \quad p_i^{ZS(2,1)} \left(x_{i+\frac{1}{2}} \right) = \frac{1}{6\gamma_{1,1}} (-u_{i-1} - \mu_7 u_i + 2u_{i+1}) \quad (43)$$

$$p_{i+1}^{ZS(2,0)} \left(x_{i+\frac{1}{2}} \right) = u_{i+1}, \quad p_{i+1}^{ZS(2,1)} \left(x_{i+\frac{1}{2}} \right) = \frac{1}{6\gamma_{1,1}} (2u_i - \mu_7 u_{i+1} - u_{i+2}) \quad (44)$$

For $k = 3, s = 2$ (35) results in

$$p_i^{ZS(3,2)} \left(x_{i+\frac{1}{2}} \right) = \frac{1}{60\mu_1} \left\{ 2\gamma_{1,1} u_{i-2} + (-13\gamma_{1,1} + 10\gamma_{1,2}) u_{i-1} + [\gamma_{1,1} (47 - 60\gamma_{0,2}) + 10\gamma_{1,2} \mu_7] u_i + (27\gamma_{1,1} - 20\gamma_{1,2}) u_{i+1} - 3\gamma_{1,1} u_{i+2} \right\}, \quad (45)$$

$$p_{i+1}^{ZS(3,2)} \left(x_{i+\frac{1}{2}} \right) = \frac{1}{60\mu_1} \left\{ -3\gamma_{1,1} u_{i-1} + (27\gamma_{1,1} - 20\gamma_{1,2}) u_i + [\gamma_{1,1} (47 - 60\gamma_{0,2}) + 10\gamma_{1,2} \mu_7] u_{i+1} + (-13\gamma_{1,1} + 10\gamma_{1,2}) u_{i+2} + 2\gamma_{1,1} u_{i+3} \right\}. \quad (46)$$

For $k = 4, s = 3$ (35) gives

$$p_i^{ZS(4,3)} \left(x_{i+\frac{1}{2}} \right) = \frac{1}{420\mu_3} \left\{ -3\mu_1 u_{i-3} + \gamma_{1,1} (25\gamma_{2,2} - 14\gamma_{2,3}) u_{i-2} + [\gamma_{1,1} (-101\gamma_{2,2} + 91\gamma_{2,3}) + 70\mu_2] u_{i-1} + [\gamma_{1,1} (\gamma_{2,2} (319 - 420\gamma_{0,3}) + 7\gamma_{2,3} (-47 + 60\gamma_{0,2})) + 70\mu_7 \mu_2] u_i + [\gamma_{1,1} (214\gamma_{2,2} - 189\gamma_{2,3}) - 140\mu_2] u_{i+1} + \gamma_{1,1} (-38\gamma_{2,2} + 21\gamma_{2,3}) u_{i+2} + 4\mu_1 u_{i+3} \right\}, \quad (47)$$

$$\begin{aligned}
 p_{i+1}^{ZS(4,3)} \left(x_{i+\frac{1}{2}} \right) &= \frac{1}{420\mu_3} \left\{ 4\mu_1 u_{i-2} + \gamma_{1,1} (-38\gamma_{2,2} + 21\gamma_{2,3}) u_{i-1} \right. \\
 &\quad + \left[\gamma_{1,1} (214\gamma_{2,2} - 189\gamma_{2,3}) - 140\mu_2 \right] u_i \\
 &\quad + \left[\gamma_{1,1} (\gamma_{2,2} (319 - 420\gamma_{0,3}) + 7\gamma_{2,3} (-47 + 60\gamma_{0,2})) \right. \\
 &\quad \left. + 70\mu_7 \mu_2 \right] u_{i+1} + \left[\gamma_{1,1} (-101\gamma_{2,2} + 91\gamma_{2,3}) + 70\mu_2 \right] u_{i+2} \\
 &\quad \left. + \gamma_{1,1} (25\gamma_{2,2} - 14\gamma_{2,3}) u_{i+3} - 3\mu_4 u_{i+4} \right\}
 \end{aligned} \tag{48}$$

For $k = 5, s = 4$ (35) produces

$$\begin{aligned}
 p_i^{ZS(5,4)} \left(x_{i+\frac{1}{2}} \right) &= \frac{1}{2520\mu_3\gamma_{4,4}} \left\{ 4\mu_3 u_{i-4} + \mu_1 (-41\gamma_{3,3} + 18\gamma_{3,4}) u_{i-3} \right. \\
 &\quad + \gamma_{1,1} \left[\gamma_{2,2} (199\gamma_{3,3} - 150\gamma_{3,4}) - 84\mu_4 \right] u_{i-2} \\
 &\quad + \left[\gamma_{1,1} (\gamma_{2,2} (-641\gamma_{3,3} + 606\gamma_{3,4}) + 546\mu_4) + 420\mu_5 \right] u_{i-1} \\
 &\quad + \left[\gamma_{1,1} (\gamma_{2,2} ((1879 - 2520\gamma_{0,4})\gamma_{3,3} + 6(-319 + 420\gamma_{0,3})\gamma_{3,4}) \right. \\
 &\quad \left. + 42(-47 + 60\gamma_{0,2})\mu_4) + 420\mu_7\mu_5 \right] u_i \\
 &\quad + \left[\gamma_{1,1} (\gamma_{2,2} (1375\gamma_{3,3} - 1284\gamma_{3,4}) - 1134\mu_4) - 840\mu_5 \right] u_{i+1} \\
 &\quad + \gamma_{1,1} \left[\gamma_{2,2} (-305\gamma_{3,3} + 228\gamma_{3,4}) + 126\mu_4 \right] u_{i+2} \\
 &\quad \left. + \mu_1 (55\gamma_{3,3} - 24\gamma_{3,4}) u_{i+3} - 5\mu_3 u_{i+4} \right\},
 \end{aligned} \tag{49}$$

$$\begin{aligned}
 p_{i+1}^{ZS(5,4)} \left(x_{i+\frac{1}{2}} \right) &= \frac{1}{2520\mu_3\gamma_{4,4}} \left\{ -5\mu_3 u_{i-3} + \mu_1 (55\gamma_{3,3} - 24\gamma_{3,4}) u_{i-2} \right. \\
 &\quad + \gamma_{1,1} \left[\gamma_{2,2} (-305\gamma_{3,3} + 228\gamma_{3,4}) + 126\mu_4 \right] u_{i-1} \\
 &\quad + \left[\gamma_{1,1} (\gamma_{2,2} (1375\gamma_{3,3} - 1284\gamma_{3,4}) - 1134\mu_4) - 840\mu_5 \right] u_i \\
 &\quad + \left[\gamma_{1,1} (\gamma_{2,2} ((1879 - 2520\gamma_{0,4})\gamma_{3,3} + 6(-319 + 420\gamma_{0,3})\gamma_{3,4}) \right. \\
 &\quad \left. + 42(-47 + 60\gamma_{0,2})\mu_4) + 420\mu_7\mu_5 \right] u_{i+1} \\
 &\quad + \left[\gamma_{1,1} (\gamma_{2,2} (-641\gamma_{3,3} + 606\gamma_{3,4}) + 546\mu_4) + 420\mu_5 \right] u_{i+2} \\
 &\quad + \gamma_{1,1} \left[\gamma_{2,2} (199\gamma_{3,3} - 150\gamma_{3,4}) - 84\mu_4 \right] u_{i+3} \\
 &\quad \left. + \mu_1 (-41\gamma_{3,3} + 18\gamma_{3,4}) u_{i+4} + 4\mu_3 u_{i+5} \right\}.
 \end{aligned} \tag{50}$$

Substituting the linear weights $\gamma_{j,s}$ with the values from **Table A3** yields the subsequent simpler formulations:

$$p_i^{ZS(2,1)} \left(x_{i+\frac{1}{2}} \right) = \frac{1}{60} (-11u_{i-1} + 49u_i + 22u_{i+1}),$$

$$p_{i+1}^{ZS(2,1)} \left(x_{i+\frac{1}{2}} \right) = \frac{1}{60} (22u_i + 49u_{i+1} - 11u_{i+2}),$$

$$p_i^{ZS(3,2)} \left(x_{i+\frac{1}{2}} \right) = \frac{1}{6000} (222u_{i-2} - 1333u_{i-1} + 4667u_i + 2777u_{i+1} - 333u_{i+2}),$$

$$p_{i+1}^{ZS(3,2)}\left(x_{i+\frac{1}{2}}\right) = \frac{1}{6000}(-333u_{i-1} + 2777u_i + 4667u_{i+1} - 1333u_{i+2} + 222u_{i+3}),$$

$$p_i^{ZS(4,3)}\left(x_{i+\frac{1}{2}}\right) = \frac{1}{420000}(-3333u_{i-3} + 26221u_{i-2} - 102110u_{i-1} + 317890u_i + 216775u_{i+1} - 39887u_{i+2} + 4444u_{i+3}),$$

$$p_{i+1}^{ZS(4,3)}\left(x_{i+\frac{1}{2}}\right) = \frac{1}{420000}(4444u_{i-2} - 39887u_{i-1} + 216775u_i + 317890u_{i+1} - 102110u_{i+2} + 26221u_{i+3} - 3333u_{i+4}),$$

$$p_i^{ZS(5,4)}\left(x_{i+\frac{1}{2}}\right) = \frac{1}{25200000}(44444u_{i-4} - 435553u_{i-3} + 2044439u_{i-2} - 6448885u_{i-1} + 18751115u_i + 13851101u_{i+1} - 3135547u_{i+2} + 584441u_{i+3} - 55555u_{i+4}),$$

$$p_{i+1}^{ZS(5,4)}\left(x_{i+\frac{1}{2}}\right) = \frac{1}{25200000}(-55555u_{i-3} + 584441u_{i-2} - 3135547u_{i-1} + 13851101u_i + 18751115u_{i+1} - 6448885u_{i+2} + 2044439u_{i+3} - 435553u_{i+4} + 44444u_{i+5}).$$