LARGE TIME STEP HLL AND HLLC SCHEMES*

MARIN PREBEG†, TORE FLÅTTEN‡, AND BERNHARD MÜLLER†

Abstract. We present Large Time Step (LTS) extensions of the Harten-Lax-van Leer (HLL) and Harten-Lax-van Leer Contact (HLLC) schemes. Herein, LTS denotes a class of explicit methods stable for Courant numbers greater than one. The original LTS method [R. J. LeVeque, *SIAM J. Numer. Anal.*, 22 (1985), pp. 1051–1073] was constructed as an extension of the Godunov scheme, and successive versions have been developed in the framework of Roe's approximate Riemann solver.

We first formulate the LTS extension of the original HLL scheme in conservation form. Next, we provide explicit expressions for the flux-difference splitting coefficients and the numerical viscosity coefficients. We then formulate the LTS extension of the HLLC scheme in conservation form.

We apply the new schemes to the one dimensional Euler equations and compare them to their non-LTS counterparts. As test cases, we consider the classical Sod shock tube problem and the Woodward-Colella blast-wave problem. It is shown that the LTS-HLL scheme smears out the contact discontinuity, while the LTS-HLLC scheme improves the resolution of both shocks and contact discontinuities. In addition, we numerically demonstrate that for the right choice of wave velocity estimates both schemes calculate entropy satisfying solutions.

Key words. Large Time Step, HLL, HLLC, Euler equations, Riemann solver

AMS subject classifications. 65M08, 35L65, 65Y20

1. Introduction.

We consider the hyperbolic system of conservation laws:

$$\mathbf{U}_t + \mathbf{F}(\mathbf{U})_x = 0, \tag{1.1a}$$

$$\mathbf{U}(x,0) = \mathbf{U}_0(x), \tag{1.1b}$$

where $\mathbf{U} \in \mathbb{R}^N$ is the vector of conserved variables, $\mathbf{F}(\mathbf{U})$ is the flux function and $\mathbf{U}_0$ is the initial data. We are interested in solving (1.1) with an explicit finite volume method not limited by the CFL (Courant-Friedrichs-Lewy) condition.

1.1. Large Time Step scheme.

A class of such methods has been proposed by LeVeque in a series of papers [11, 12, 13] in the 1980s. Therein, the Godunov scheme was extended to the LTS-Godunov scheme and applied to the Euler equations. The CFL condition is relaxed by allowing the waves from each Riemann problem to travel more than one cell during a single time step. Each wave is treated as a discontinuity, and the interactions between the waves are assumed to be linear. Through the years this idea has been used by a number of authors. For the shallow water equations, Murillo, Morales-Hernandez and co-workers [23, 20, 22, 21] applied the LTS-Roe scheme and Xu et al. [38] applied the LTS-Godunov scheme. Further applications of the LTS-Godunov scheme include the 3D Euler equations by Qian and Lee [26, 27], high speed combustion waves by Tang et al. [32], and Maxwell's equations by Makwana and Chatterjee [18]. Lindqvist and Lund [16] and Prebeg et al. [25] applied the LTS-Roe scheme to two-phase flow models. Lindqvist et al. [15] also studied the TVD properties of LTS methods and showed that the LTS-Roe scheme and the LTS-Lax-Friedrichs scheme are the least and most diffusive TVD LTS schemes,

*The authors were supported by the Research Council of Norway (234126/30) through the SIMCOFLOW project.

†Department of Energy and Process Engineering, Norwegian University of Science and Technology, Kolbjørn Hejes vei 2, NO-7491 Trondheim, Norway (marin.prebeg@ntnu.no, bernhard.muller@ntnu.no)

‡SINTEF Materials and Chemistry, P. O. Box 4760 Sluppen, NO-7465 Trondheim, Norway (toref@math.uio.no).

respectively. All the methods discussed above share the feature of starting from a Godunov or Roe-type Riemann solver and extending it to the LTS framework.

1.2. HLL and HLLC schemes. The original Riemann solver proposed by Godunov [7] involves a computationally costly procedure, especially for complex equations of state. In order to reduce the computational time, different approximate Riemann solvers have been developed. A very simple approximate Riemann solver, proposed by Harten, Lax and van Leer [9] in the 1980s, has became known as the HLL solver. The original paper [9] assumes a two-wave structure of the solution and constructs the approximate Riemann solver by using estimates of the velocities of the slowest and the fastest waves. The choice for these velocity estimates has been studied for instance by Davis [4], Einfeldt and co-workers [5, 6] and Batten et al. [1]. The original HLL solver may poorly resolve certain physics in systems where the solution structure consists of more than two waves. For the Euler equations, Toro et al. [36] proposed the HLLC solver in which the contact discontinuity is reconstructed by assuming a three-wave structure of the solution. Today, HLL and HLLC solvers are widely used in a number of different fields, such as multiphase flow modeling [39, 34, 33, 24, 3, 2, 17] and magnetohydrodynamics [10, 19].

1.3. Outline of the paper. In this paper, we follow LeVeque's approach [13] to derive Large Time Step extensions of the HLL and HLLC schemes, denoted as LTS-HLL and LTS-HLLC, respectively. In section 2 we outline the problem and the standard (non-LTS) numerical methods on which we will build. In section 3 we present the standard HLL scheme and extend it to the LTS framework. We then write the LTS-HLL scheme in numerical viscosity and flux-difference splitting form. Section 4 presents the LTS extension of the HLLC scheme. In section 5 we present numerical investigations for the one-dimensional Euler equations. The resulting LTS-HLL(C) schemes are seen to improve the efficiency of standard HLL(C) schemes while also providing improved robustness compared to previously studied Large Time Step methods. In section 6 we close with conclusions and final remarks.

2. Preliminaries.

2.1. Problem outline. As a special example of (1.1) we consider the Euler equations where the vector of conserved variables $\mathbf{U}$ and the flux function $\mathbf{F}(\mathbf{U})$ are defined as:

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho u \\ E \end{pmatrix}, \qquad \mathbf{F}(\mathbf{U}) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ u(E+p) \end{pmatrix}, \tag{2.1}$$

where ρ, u, E, p denote the density, velocity, total energy density and pressure, respectively. The system is closed by the relation for the total energy density, $E = \rho e + \rho u^2/2$, and an equation of state for perfect gas, $e = p/(\rho(\gamma - 1))$. Throughout the paper we will use $\gamma = 1.4$ for air. Alternatively, we can write (1.1) in a quasi-linear form as:

$$\mathbf{U}_t + \mathbf{A}(\mathbf{U})\mathbf{U}_x = 0, \qquad \mathbf{A}(\mathbf{U}) = \mathbf{F}(\mathbf{U})_{\mathbf{U}}. \tag{2.2}$$

We assume that the system of equations (2.2) is hyperbolic, i.e. the Jacobian matrix $\mathbf{A}$ has real eigenvalues and linearly independent eigenvectors.

2.2. Numerical methods. We discretize (1.1) by the explicit Euler method in time and the finite volume method in space:

$$\mathbf{U}_j^{n+1} = \mathbf{U}_j^n - \frac{\Delta t}{\Delta x}\left(\mathbf{F}_{j+1/2}^n - \mathbf{F}_{j-1/2}^n\right), \tag{2.3}$$

where $\mathbf{U}_j^n$ is a piecewise constant approximation of $\mathbf{U}$ in the cell with center at x_j at time level n and $\mathbf{F}_{j+1/2}^n$ is a numerical approximation of the flux function at the cell interface $x_{j+1/2}$ at time level n.

2.2.1. Standard 3-point schemes. In the case that the numerical flux depends only on the neighboring cell values, we can with no loss of generality write the scheme in the numerical viscosity form [8, 31]:

$$\mathbf{F}_{j+1/2}^n = \mathbf{F}\left(\mathbf{U}_j^n, \mathbf{U}_{j+1}^n\right) = \frac{1}{2}\left(\mathbf{F}_j^n + \mathbf{F}_{j+1}^n\right) - \frac{1}{2}\mathbf{Q}_{j+1/2}^n\left(\mathbf{U}_{j+1}^n - \mathbf{U}_j^n\right), \tag{2.4}$$

where $\mathbf{F}_j^n = \mathbf{F}\left(\mathbf{U}_j^n\right)$ and $\mathbf{Q}_{j+1/2}^n$ is the numerical viscosity matrix. To simplify the notation, the time level n will be implicitly assumed in the absence of a temporal index. The choice of the numerical viscosity matrix $\mathbf{Q}$ determines the finite volume scheme we use, i.e. for the Lax-Friedrichs scheme $\mathbf{Q}_{\mathrm{LxF}} = \mathrm{diag}(\Delta x/\Delta t)$, and for the Roe scheme $\mathbf{Q}_{\mathrm{Roe}} = |\hat{\mathbf{A}}|$ where $\hat{\mathbf{A}}$ is the Roe matrix [28]. $\hat{\mathbf{A}}$ can be diagonalized as:

$$\hat{\mathbf{A}} = \hat{\mathbf{R}}\hat{\mathbf{\Lambda}}\hat{\mathbf{R}}^{-1}, \tag{2.5}$$

where $\hat{\mathbf{R}}$ is the matrix of right eigenvectors and $\hat{\mathbf{\Lambda}} = \mathrm{diag}(\lambda_1, \ldots, \lambda_N)$ is the matrix of eigenvalues. We note that in the Lax-Friedrichs and the Roe schemes, the numerical viscosity matrix $\mathbf{Q}$ acts independently on each characteristic field. In that case, $\mathbf{Q}$ can be diagonalized as:

$$\mathbf{Q} = \hat{\mathbf{R}}\mathbf{\Omega}\hat{\mathbf{R}}^{-1}, \tag{2.6}$$

where $\mathbf{\Omega} = \mathrm{diag}(\omega_1, \ldots, \omega_N)$ is the matrix of eigenvalues of $\mathbf{Q}$, and $\mathbf{Q}$ and $\hat{\mathbf{A}}$ have the same eigenvectors. The numerical viscosities of the Lax-Friedrichs and the Roe scheme are then obtained by:

$$\mathbf{\Omega}_{\mathrm{LxF}} = \frac{\Delta x}{\Delta t}\mathbf{I}, \quad \mathbf{\Omega}_{\mathrm{Roe}} = |\hat{\mathbf{\Lambda}}|. \tag{2.7}$$

An alternative way to discretize (1.1) is by the explicit Euler method in time and flux-difference splitting in space:

$$\mathbf{U}_j^{n+1} = \mathbf{U}_j^n - \frac{\Delta t}{\Delta x}\left(\hat{\mathbf{A}}_{j-1/2}^{+}\left(\mathbf{U}_j^n - \mathbf{U}_{j-1}^n\right) + \hat{\mathbf{A}}_{j+1/2}^{-}\left(\mathbf{U}_{j+1}^n - \mathbf{U}_j^n\right)\right), \tag{2.8}$$

where $\hat{\mathbf{A}}^{\pm}$ represent a splitting of the Roe matrix (2.5) according to:

$$\hat{\mathbf{A}}^{\pm} = \hat{\mathbf{R}}\hat{\mathbf{\Lambda}}^{\pm}\hat{\mathbf{R}}^{-1}. \tag{2.9}$$

Herein, $\hat{\mathbf{\Lambda}}^{\pm}$ are obtained by transforming each diagonal entry of $\hat{\mathbf{\Lambda}}$:

$$\lambda_{\mathrm{LxF}}^{\pm} = \frac{1}{2}\left(\lambda \pm \frac{\Delta x}{\Delta t}\right), \quad \lambda_{\mathrm{Roe}}^{\pm} = \pm\max(0, \pm\lambda). \tag{2.10}$$

For 3-point schemes, the size of the time step in discretizations (2.3) and (2.8) is limited by the CFL condition:

$$C = \max_{k,x}|\lambda_k(x,t)|\,\frac{\Delta t}{\Delta x} \leq 1, \tag{2.11}$$

where λ_k are the eigenvalues of the Jacobian matrix $\mathbf{A}$ in (2.2). In this paper, we consider explicit methods that are not limited by the constraint (2.11).

2.2.2. Large Time Step method. The natural LTS extension of the numerical viscosity formulation (2.4) is [15]:

$$\mathbf{F}_{j+1/2} = \frac{1}{2}\left(\mathbf{F}_j + \mathbf{F}_{j+1}\right) - \frac{1}{2}\sum_{i=-\infty}^{\infty} \mathbf{Q}^i_{j+1/2+i}\Delta\mathbf{U}_{j+1/2+i}, \tag{2.12}$$

and the natural LTS extension of the flux-difference splitting formulation (2.8) is [15]:

$$\mathbf{U}_j^{n+1} = \mathbf{U}_j - \frac{\Delta t}{\Delta x}\sum_{i=0}^{\infty}\left(\hat{\mathbf{A}}^{i+}_{j-1/2-i}\Delta\mathbf{U}_{j-1/2-i} + \hat{\mathbf{A}}^{i-}_{j+1/2+i}\Delta\mathbf{U}_{j+1/2+i}\right), \tag{2.13}$$

where we introduced the notation $\Delta\mathbf{U}_{j+1/2} = \mathbf{U}_{j+1} - \mathbf{U}_j$. We note that (2.12) differs from [15] in a sense that we scale $\mathbf{Q}^i$ with $\Delta x/\Delta t$. Herein, the upper indices denote the relative cell interface position. These will be further clarified in subsection 3.2. Lindqvist et al. [15] provided the partial viscosity coefficients $\mathbf{Q}^i$ and the flux-difference splitting coefficients $\hat{\mathbf{A}}^{i\pm}$ for the LTS-Godunov, LTS-Roe and LTS-Lax-Friedrichs schemes. For the LTS-Roe scheme [15], the partial viscosity coefficients are defined through the eigenvalues of $\mathbf{Q}^i$:

$$\mathbf{Q}^i_{j+1/2} = (\hat{\mathbf{R}}\boldsymbol{\Omega}^i\hat{\mathbf{R}}^{-1})_{j+1/2}, \tag{2.14}$$

where the eigenvalues are defined as:

$$\omega^0_{\mathrm{Roe}} = |\lambda|\,, \tag{2.15a}$$

$$\omega^{\mp i}_{\mathrm{Roe}} = 2\max\left(0, \pm\lambda - i\frac{\Delta x}{\Delta t}\right), \quad \text{for } i > 0, \tag{2.15b}$$

and the flux-difference splitting coefficients are defined through the eigenvalues of $\hat{\mathbf{A}}^{i\pm}$:

$$\hat{\mathbf{A}}^{i\pm}_{j+1/2} = (\hat{\mathbf{R}}\hat{\boldsymbol{\Lambda}}^{i\pm}\hat{\mathbf{R}}^{-1})_{j+1/2}, \tag{2.16}$$

where the eigenvalues are defined as:

$$\lambda^{i\pm}_{\mathrm{Roe}} = \pm\max\left(0, \min\left(\mp\lambda - i\frac{\Delta x}{\Delta t}, \frac{\Delta x}{\Delta t}\right)\right). \tag{2.17}$$

In the following section we determine these coefficients for the LTS-HLL scheme.

3. HLL scheme.

We start by presenting the standard HLL scheme of Harten et al. [9]. Then we formulate the natural LTS extension of the HLL scheme and provide explicit expressions for the flux-difference splitting and the numerical viscosity coefficients.

3.1. The standard HLL scheme.

We consider the cell interface Riemann problem:

$$\mathbf{U}(x,0) = \begin{cases} \mathbf{U}_j & \text{if} \quad x < 0, \\ \mathbf{U}_{j+1} & \text{if} \quad x > 0. \end{cases} \tag{3.1}$$

The original HLL scheme by Harten et al. [9] solves the Riemann problem approximately by assuming a single state between the left and right states:

$$\widetilde{\mathbf{U}}(x,t) = \begin{cases} \mathbf{U}_j & \text{if} \quad x < S_{\mathrm{L}}t, \\ \mathbf{U}^{\mathrm{HLL}}_{j+1/2} & \text{if} \quad S_{\mathrm{L}}t < x < S_{\mathrm{R}}t, \\ \mathbf{U}_{j+1} & \text{if} \quad x > S_{\mathrm{R}}t, \end{cases} \tag{3.2}$$

where S_{L} and S_{R} are approximations of the smallest and the largest wave velocities at the interface $x_{j+1/2}$. As for now, we leave these unspecified and return to them in section 5. The intermediate state $\mathbf{U}^{\mathrm{HLL}}_{j+1/2}$ is defined such that the Riemann solver is consistent with the integral form of the conservation law (1.1), see [9, 5]:

$$\mathbf{U}^{\mathrm{HLL}}_{j+1/2} = \frac{S_{\mathrm{R}}\mathbf{U}_{j+1} - S_{\mathrm{L}}\mathbf{U}_j + \mathbf{F}_j - \mathbf{F}_{j+1}}{S_{\mathrm{R}} - S_{\mathrm{L}}}. \tag{3.3}$$

Next, we use $\mathbf{U}^{\mathrm{HLL}}_{j+1/2}$ to determine the flux function $\mathbf{F}_{j+1/2}$. This is defined as:

$$\mathbf{F}_{j+1/2} = \begin{cases} \mathbf{F}_j & \text{if} \quad 0 < S_{\mathrm{L}}, \\ \mathbf{F}^{\mathrm{HLL}}_{j+1/2} & \text{if} \quad S_{\mathrm{L}} < 0 < S_{\mathrm{R}}, \\ \mathbf{F}_{j+1} & \text{if} \quad 0 > S_{\mathrm{R}}. \end{cases} \tag{3.4}$$

In the interesting case, $S_{\mathrm{L}} < 0 < S_{\mathrm{R}}$, the flux function has the form [35]:

$$\mathbf{F}^{\mathrm{HLL}}_{j+1/2} = \mathbf{F}_j + S_{\mathrm{L}}\left(\mathbf{U}^{\mathrm{HLL}}_{j+1/2} - \mathbf{U}_j\right), \tag{3.5}$$

$$\mathbf{F}^{\mathrm{HLL}}_{j+1/2} = \mathbf{F}_{j+1} + S_{\mathrm{R}}\left(\mathbf{U}^{\mathrm{HLL}}_{j+1/2} - \mathbf{U}_{j+1}\right). \tag{3.6}$$

These two equations are equivalent and by using (3.3) in any of them we obtain:

$$\mathbf{F}^{\mathrm{HLL}}_{j+1/2} = \frac{S_{\mathrm{R}}\mathbf{F}_j - S_{\mathrm{L}}\mathbf{F}_{j+1} + S_{\mathrm{L}}S_{\mathrm{R}}\left(\mathbf{U}_{j+1} - \mathbf{U}_j\right)}{S_{\mathrm{R}} - S_{\mathrm{L}}}. \tag{3.7}$$

Further, the equations (3.4) and (3.7) can be written more compactly as:

$$\mathbf{F}_{j+1/2} = \frac{S^+_{\mathrm{R}}\mathbf{F}_j - S^-_{\mathrm{L}}\mathbf{F}_{j+1} + S^-_{\mathrm{L}}S^+_{\mathrm{R}}\left(\mathbf{U}_{j+1} - \mathbf{U}_j\right)}{S^+_{\mathrm{R}} - S^-_{\mathrm{L}}}, \tag{3.8}$$

where $S^-_{\mathrm{L}} = \min(S_{\mathrm{L}}, 0)$ and $S^+_{\mathrm{R}} = \max(S_{\mathrm{R}}, 0)$. Equation (3.8) is then used in (2.3). For more information and more detailed derivation we refer to [1, 4, 5, 9, 35]. Einfeldt [5] showed that the numerical flux (3.8) can be recovered from the numerical viscosity framework (2.4) by setting:

$$\mathbf{Q}^{\mathrm{HLL}}_{j+1/2} = \frac{S^+_{\mathrm{R}} + S^-_{\mathrm{L}}}{S^+_{\mathrm{R}} - S^-_{\mathrm{L}}}\hat{\mathbf{A}}_{j+1/2} - 2\frac{S^-_{\mathrm{L}}S^+_{\mathrm{R}}}{S^+_{\mathrm{R}} - S^-_{\mathrm{L}}}\mathbf{I}. \tag{3.9}$$

Following the framework introduced in (2.7), we define the HLL scheme through the diagonal entries of $\mathbf{\Omega}$ as:

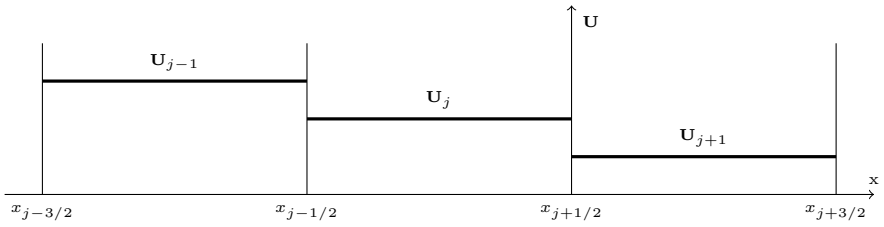
$$\omega_{\mathrm{HLL}} = \frac{S^+_{\mathrm{R}}(\lambda - S^-_{\mathrm{L}}) - S^-_{\mathrm{L}}(S^+_{\mathrm{R}} - \lambda)}{S^+_{\mathrm{R}} - S^-_{\mathrm{L}}}. \tag{3.10}$$

The HLL scheme can also be written in the flux-difference splitting framework (2.10) by modifying the diagonal entries of $\hat{\mathbf{\Lambda}}^{\pm}$ as:

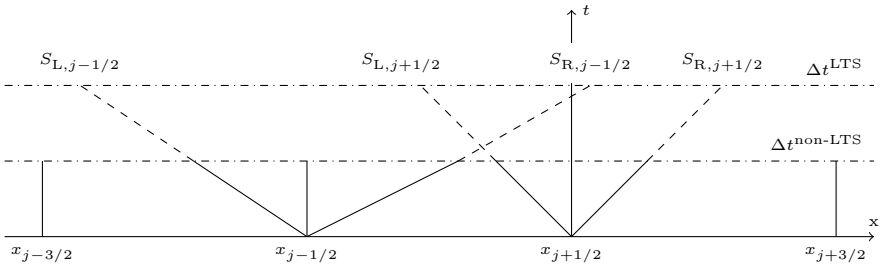
$$\lambda^+_{\mathrm{HLL}} = \frac{\lambda - S^-_{\mathrm{L}}}{S^+_{\mathrm{R}} - S^-_{\mathrm{L}}}S^+_{\mathrm{R}} = \frac{\lambda - S_{\mathrm{L}}}{S_{\mathrm{R}} - S_{\mathrm{L}}}S^+_{\mathrm{R}} + \frac{S_{\mathrm{R}} - \lambda}{S_{\mathrm{R}} - S_{\mathrm{L}}}S^+_{\mathrm{L}}, \tag{3.11}$$

$$\lambda^-_{\mathrm{HLL}} = \frac{S^+_{\mathrm{R}} - \lambda}{S^+_{\mathrm{R}} - S^-_{\mathrm{L}}}S^-_{\mathrm{L}} = \frac{\lambda - S_{\mathrm{L}}}{S_{\mathrm{R}} - S_{\mathrm{L}}}S^-_{\mathrm{R}} + \frac{S_{\mathrm{R}} - \lambda}{S_{\mathrm{R}} - S_{\mathrm{L}}}S^-_{\mathrm{L}}. \tag{3.12}$$

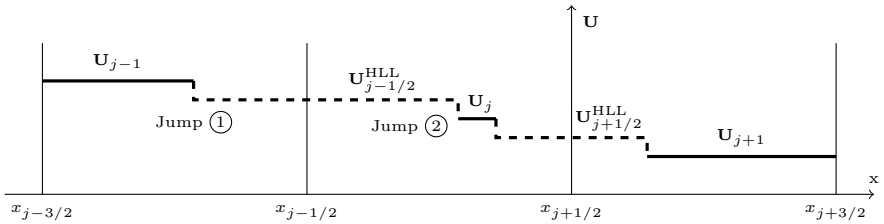
3.2. The LTS-HLL scheme. We want to construct the LTS extension of the numerical flux function (3.8). Consider the Figure 1a and the Riemann problem at the interface $x_{j+1/2}$. First, we consider the wave structure when $C \leq 1$, denoted in Figure 1b as $\Delta t^{\text{non-LTS}}$. In this case, the Riemann problem at $x_{j+1/2}$ is completely defined by $\mathbf{U}_j, \mathbf{U}_{j+1}$ and velocities $S_{\text{L},j+1/2}$ and $S_{\text{R},j+1/2}$ being emitted from the interface $x_{j+1/2}$, see (3.2)–(3.8). Next, we consider the case when $C > 1$, denoted in Figure 1b as Δt^{LTS}. For this case, the wave emitted from the interface $x_{j-1/2}$ and associated with velocity $S_{\text{R},j-1/2}$ passes through the interface $x_{j+1/2}$.



(a) Riemann problems at $x_{j-1/2}$ and $x_{j+1/2}$



(b) Characteristics with slopes $\mathrm{d}x(t)/\mathrm{d}t = S_{\text{L,R}}$ at $x_{j\mp1/2}$



(c) Approximate solutions of Riemann problems at $x_{j\mp1/2}$ with HLL scheme

Fig. 1: Wave structure in the LTS-HLL scheme

This wave violates the CFL condition (2.11) since we allowed the wave to travel more than one cell during a single time step. However, we may relax the CFL condition (2.11) if we modify (3.8) by taking into account this additional contribution. We start by assuming that the interactions between the waves are linear and we note that:

- The flux function (3.8) at the interface $x_{j+1/2}$ is increased by the contribution from the *jump 2* moving to the right with the velocity $S_{\text{R},j-1/2}$.
- The contribution from the *jump 2* does not start passing through the face $x_{j+1/2}$ immediately, i.e. it has to travel through the cell x_j before it starts to pass through the face $x_{j+1/2}$.

Based on this, we modify (3.8) as:

$$\mathbf{F}^{\text{LTS-HLL}}_{j+1/2} = \mathbf{F}^{0}_{j+1/2} + S^{-1}_{\text{R},j-1/2}\left(\mathbf{U}^{\text{HLL}}_{j-1/2} - \mathbf{U}_j\right), \tag{3.13}$$

where we denoted (3.8) as $\mathbf{F}^{0}_{j+1/2}$, and:

$$S^{-1}_{\text{R},j-1/2} = S_{\text{R},j-1/2} - \frac{\Delta x}{\Delta t}. \tag{3.14}$$

The purpose of this modification is to take into the account the fact that the wave has to travel one cell before it starts contributing to the flux function (3.13). In the general case, we allow for an arbitrarily large time step size Δt, therefore allowing the waves to travel several cells during a single time step. In addition, we note that each interface may emit waves where each of the local wave speeds S_{L} and S_{R} may be either negative, zero or positive. Therefore, the general formula for the flux function of the LTS-HLL scheme has the form:

$$\mathbf{F}^{\text{LTS-HLL}}_{j+1/2} = \mathbf{F}^{0}_{j+1/2} + \sum_{i=1}^{\infty} \mathbf{F}^{-i}_{j+1/2-i} + \sum_{i=1}^{\infty} \mathbf{F}^{+i}_{j+1/2+i}, \tag{3.15}$$

where the additional terms under the sum signs represent the information reaching the face $x_{j+1/2}$ from neighboring Riemann problems on the left and on the right, respectively. The newly introduced terms in (3.15) are:

$$\mathbf{F}^{-i}_{j+1/2-i} = S^{-i}_{\text{R},j+1/2-i}\left(\mathbf{U}^{\text{HLL}}_{j+1/2-i} - \mathbf{U}_{j+1-i}\right) + S^{-i}_{\text{L},j+1/2-i}\left(\mathbf{U}_{j-i} - \mathbf{U}^{\text{HLL}}_{j+1/2-i}\right), \tag{3.16}$$

$$\mathbf{F}^{+i}_{j+1/2+i} = S^{+i}_{\text{L},j+1/2+i}\left(\mathbf{U}^{\text{HLL}}_{j+1/2+i} - \mathbf{U}_{j+i}\right) + S^{+i}_{\text{R},j+1/2+i}\left(\mathbf{U}_{j+1+i} - \mathbf{U}^{\text{HLL}}_{j+1/2+i}\right), \tag{3.17}$$

where the modified wave velocities are:

$$S^{-i}_{[\text{L,R}],j+1/2-i} = \max\left(S_{[\text{L,R}],j+1/2-i} - i\frac{\Delta x}{\Delta t}, 0\right), \tag{3.18}$$

$$S^{+i}_{[\text{L,R}],j+1/2+i} = \min\left(S_{[\text{L,R}],j+1/2+i} + i\frac{\Delta x}{\Delta t}, 0\right). \tag{3.19}$$

Equation (3.15) is then used in (2.3).

3.2.1. The LTS-HLL scheme in numerical viscosity form. We can now write the LTS-HLL scheme in the numerical viscosity form (2.12).

PROPOSITION 1. *Given the Roe matrix:*

$$\hat{\mathbf{A}}_{j+1/2} = \left(\hat{\mathbf{R}}\hat{\mathbf{\Lambda}}\hat{\mathbf{R}}^{-1}\right)_{j+1/2} \quad \forall j, \tag{3.20}$$

where $\hat{\mathbf{\Lambda}}$ is the diagonal matrix of eigenvalues, the LTS-HLL scheme defined by (3.13)–(3.19) can be written in the numerical viscosity form (2.12) with coefficients:

$$\mathbf{Q}^{i}_{j+1/2} = \left(\hat{\mathbf{R}}\mathbf{\Omega}^{i}\hat{\mathbf{R}}^{-1}\right)_{j+1/2}, \tag{3.21}$$

where $\mathbf{\Omega}^i(\hat{\mathbf{\Lambda}}, S_\mathrm{L}, S_\mathrm{R})$ *is the diagonal matrix with entries given by:*

$$\omega^0_{HLL} = \frac{S^+_\mathrm{R}(\lambda - S^-_\mathrm{L}) - S^-_\mathrm{L}(S^+_\mathrm{R} - \lambda)}{S^+_\mathrm{R} - S^-_\mathrm{L}}, \tag{3.22a}$$

$$\begin{aligned}\omega^{\mp i}_{HLL} &= 2\frac{S_\mathrm{L} - \lambda}{S_\mathrm{R} - S_\mathrm{L}} \max\left(0, \pm S_\mathrm{R} - i\frac{\Delta x}{\Delta t}\right) \\ &\quad + 2\frac{\lambda - S_\mathrm{R}}{S_\mathrm{R} - S_\mathrm{L}} \max\left(0, \pm S_\mathrm{L} - i\frac{\Delta x}{\Delta t}\right) \quad \textit{for} \quad i > 0.\end{aligned} \tag{3.22b}$$

Proof. The coefficient $\mathbf{Q}^0$ has already been determined by (3.9). We obtain the coefficients $\mathbf{Q}^i$ for $i \neq 0$ by equalizing (2.12) and (3.15), while using the Roe condition [28]:

$$\hat{\mathbf{A}}_{j+1/2}\left(\mathbf{U}_{j+1} - \mathbf{U}_j\right) = \mathbf{F}(\mathbf{U}_{j+1}) - \mathbf{F}(\mathbf{U}_j). \tag{3.23}$$

∎

We point out the similarity of the LTS-HLL numerical viscosity coefficients (3.22) to the partial viscosity coefficients of the LTS-Roe scheme, (2.15).

3.2.2. The LTS-HLL scheme in flux-difference splitting form. We have built the LTS-HLL method by heuristic arguments as an extension of the standard HLL scheme, following LeVeque's general approach of treating all wave interactions as linear [13]. We now derive the flux-difference splitting formulation in a more formal way, starting with LeVeque's general updating formula [13]:

$$\mathbf{U}^{n+1}_j = \frac{\Delta t}{\Delta x}\sum_{i=-\infty}^{\infty}\int_{(i-1)\frac{\Delta x}{\Delta t}}^{i\frac{\Delta x}{\Delta t}} \widetilde{\mathbf{U}}_{j+1/2-i}(\zeta_i)\,\mathrm{d}\zeta_i - \sum_{\ell=-\infty}^{\infty}\mathbf{U}_\ell, \tag{3.24}$$

where $\widetilde{\mathbf{U}}_{j+1/2-i}(\zeta_i)$ is the solution to the Riemann problem at $x_{j+1/2-i}$. Herein:

$$\zeta_i = \frac{x - x_{j+1/2-i}}{t - t^n}. \tag{3.25}$$

Proposition 2. *Given the Roe matrix:*

$$\hat{\mathbf{A}}_{j+1/2} = \left(\hat{\mathbf{R}}\hat{\mathbf{\Lambda}}\hat{\mathbf{R}}^{-1}\right)_{j+1/2} \quad \forall j, \tag{3.26}$$

where $\hat{\mathbf{\Lambda}}$ *is the diagonal matrix of eigenvalues, the LTS-HLL scheme can be written in the flux-difference splitting form* (2.13) *with coefficients:*

$$\hat{\mathbf{A}}^{i\pm}_{j+1/2} = \left(\hat{\mathbf{R}}\hat{\mathbf{\Lambda}}^{i\pm}\hat{\mathbf{R}}^{-1}\right)_{j+1/2}, \tag{3.27}$$

where $\hat{\mathbf{\Lambda}}^{i\pm}(\hat{\mathbf{\Lambda}}, S_\mathrm{L}, S_\mathrm{R})$ *is the diagonal matrix with entries given by:*

$$\begin{aligned}\lambda^{i\pm}_{HLL} = &\pm \frac{\lambda - S_\mathrm{L}}{S_\mathrm{R} - S_\mathrm{L}} \max\left(0, \min\left(\pm S_\mathrm{R} - i\frac{\Delta x}{\Delta t}, \frac{\Delta x}{\Delta t}\right)\right) \\ &\pm \frac{S_\mathrm{R} - \lambda}{S_\mathrm{R} - S_\mathrm{L}} \max\left(0, \min\left(\pm S_\mathrm{L} - i\frac{\Delta x}{\Delta t}, \frac{\Delta x}{\Delta t}\right)\right).\end{aligned} \tag{3.28}$$

Proof. The HLL Riemann solver (3.2) can be written as:

$$\begin{aligned}\widetilde{\mathbf{U}}_{j+1/2}(\zeta) &= \mathbf{U}_j + H(\zeta - S_{\mathrm{L}})\left(\mathbf{U}^{\mathrm{HLL}}_{j+1/2} - \mathbf{U}_j\right) + H(\zeta - S_{\mathrm{R}})\left(\mathbf{U}_{j+1} - \mathbf{U}^{\mathrm{HLL}}_{j+1/2}\right) \\ &= \mathbf{U}_{j+1} - H(S_{\mathrm{L}} - \zeta)\left(\mathbf{U}^{\mathrm{HLL}}_{j+1/2} - \mathbf{U}_j\right) - H(S_{\mathrm{R}} - \zeta)\left(\mathbf{U}_{j+1} - \mathbf{U}^{\mathrm{HLL}}_{j+1/2}\right),\end{aligned} \tag{3.29}$$

where H is the Heaviside function. Using (3.3) we can rewrite this as:

$$\widetilde{\mathbf{U}}_{j+1/2}(\zeta) = \mathbf{U}_j + \left(\frac{H(\zeta - S_{\mathrm{L}})}{S_{\mathrm{R}} - S_{\mathrm{L}}}(\mathbf{S}_{\mathrm{R}} - \hat{\mathbf{A}}) + \frac{H(\zeta - S_{\mathrm{R}})}{S_{\mathrm{R}} - S_{\mathrm{L}}}(\hat{\mathbf{A}} - \mathbf{S}_{\mathrm{L}})\right)(\mathbf{U}_{j+1} - \mathbf{U}_j) \tag{3.30a}$$

$$= \mathbf{U}_{j+1} - \left(\frac{H(S_{\mathrm{L}} - \zeta)}{S_{\mathrm{R}} - S_{\mathrm{L}}}(\mathbf{S}_{\mathrm{R}} - \hat{\mathbf{A}}) + \frac{H(S_{\mathrm{R}} - \zeta)}{S_{\mathrm{R}} - S_{\mathrm{L}}}(\hat{\mathbf{A}} - \mathbf{S}_{\mathrm{L}})\right)(\mathbf{U}_{j+1} - \mathbf{U}_j), \tag{3.30b}$$

where $\mathbf{S}_{\mathrm{L}} = S_{\mathrm{L}}\mathbf{I}$ and $\mathbf{S}_{\mathrm{R}} = S_{\mathrm{R}}\mathbf{I}$. We then use (3.30a) in (3.24) and note that for $i \leq 0$ we can write:

$$\int_{(i-1)\frac{\Delta x}{\Delta t}}^{i\frac{\Delta x}{\Delta t}} \widetilde{\mathbf{U}}_{j+1/2-i}(\zeta_i)\,\mathrm{d}\zeta_i = \frac{\Delta x}{\Delta t}\mathbf{U}_{j-i} - \hat{\mathbf{A}}^{(-i)-}_{j+1/2-i}\left(\mathbf{U}_{j+1-i} - \mathbf{U}_{j-i}\right), \tag{3.31}$$

where:

$$\hat{\mathbf{A}}^{i-} = \hat{\mathbf{R}}\hat{\mathbf{\Lambda}}^{i-}\hat{\mathbf{R}}^{-1}, \tag{3.32}$$

and $\hat{\mathbf{\Lambda}}^{i-}$ is the diagonal matrix with values:

$$\begin{aligned}\lambda^{i-} =& \frac{\lambda - S_{\mathrm{L}}}{S_{\mathrm{R}} - S_{\mathrm{L}}}\min\left(0, \max\left(S_{\mathrm{R}} + i\frac{\Delta x}{\Delta t}, -\frac{\Delta x}{\Delta t}\right)\right) \\ &+ \frac{S_{\mathrm{R}} - \lambda}{S_{\mathrm{R}} - S_{\mathrm{L}}}\min\left(0, \max\left(S_{\mathrm{L}} + i\frac{\Delta x}{\Delta t}, -\frac{\Delta x}{\Delta t}\right)\right).\end{aligned} \tag{3.33}$$

Similarly, we use (3.30b) in (3.24) and note that for $i \geq 1$ we can write:

$$\int_{(i-1)\frac{\Delta x}{\Delta t}}^{i\frac{\Delta x}{\Delta t}} \widetilde{\mathbf{U}}_{j+1/2-i}(\zeta_i)\,\mathrm{d}\zeta_i = \frac{\Delta x}{\Delta t}\mathbf{U}_{j+1-i} - \hat{\mathbf{A}}^{(i-1)+}_{j+1/2-i}\left(\mathbf{U}_{j+1-i} - \mathbf{U}_{j-i}\right), \tag{3.34}$$

where:

$$\hat{\mathbf{A}}^{i+} = \hat{\mathbf{R}}\hat{\mathbf{\Lambda}}^{i+}\hat{\mathbf{R}}^{-1}, \tag{3.35}$$

and $\hat{\mathbf{\Lambda}}^{i+}$ is the diagonal matrix with values:

$$\begin{aligned}\lambda^{i+} =& \frac{\lambda - S_{\mathrm{L}}}{S_{\mathrm{R}} - S_{\mathrm{L}}}\max\left(0, \min\left(S_{\mathrm{R}} - i\frac{\Delta x}{\Delta t}, \frac{\Delta x}{\Delta t}\right)\right) \\ &+ \frac{S_{\mathrm{R}} - \lambda}{S_{\mathrm{R}} - S_{\mathrm{L}}}\max\left(0, \min\left(S_{\mathrm{L}} - i\frac{\Delta x}{\Delta t}, \frac{\Delta x}{\Delta t}\right)\right).\end{aligned} \tag{3.36}$$

Substituting (3.31) and (3.34) into (3.24) we recover the LTS flux-difference splitting equation (2.13). ∎

PROPOSITION 3. *The flux-difference splitting formulation (3.27)–(3.28) and the numerical viscosity formulation (3.21)–(3.22) are equivalent.*

Proof. Lindqvist et al. [15] derived the following one-to-one mapping between the numerical viscosity and flux-difference splitting coefficients:

$$\mathbf{A}^{0\pm} = \frac{1}{2}\frac{\Delta x}{\Delta t}\left(\frac{\Delta t}{\Delta x}\mathbf{A} \pm \mathbf{Q}^{0} \mp \mathbf{Q}^{\mp 1}\right), \quad \mathbf{A}^{i\pm} = \pm\frac{1}{2}\frac{\Delta x}{\Delta t}\left(\mathbf{Q}^{\mp i} - \mathbf{Q}^{\mp(i+1)}\right). \tag{3.37}$$

By using(3.21)–(3.22) in (3.37) we obtain (3.27)–(3.28). ∎

We point out the similarity of the LTS-HLL flux-difference splitting coefficients (3.28) to the flux-difference splitting coefficients of the LTS-Roe scheme, (2.17).

4. HLLC scheme.

In this section we propose a direct extension from the HLLC scheme to the LTS-HLLC scheme, following the approaches from section 3.

4.1. Standard HLLC scheme.

We recall that the standard HLL scheme assumes a two wave structure of the solution with a single, uniform state $\mathbf{U}^{\mathrm{HLL}}$ between the waves. This is a correct assumption for hyperbolic systems consisting of only two equations (such as the one-dimensional shallow water equations). However, for the Euler equations this assumption leads to neglecting the contact discontinuity. The approach to recover the missing contact discontinuity was first presented by Toro et al. [36]. Herein, we outline an approach to reconstruct the missing wave following the approach described by Toro in [35].

The standard HLLC scheme is given in the form similar to the HLL scheme defined by equations (3.2) and (3.4), but with the state $\mathbf{U}^{\mathrm{HLL}}$ being split into two states separated by a contact discontinuity:

$$\widetilde{\mathbf{U}}(x,t) = \begin{cases} \mathbf{U}_j & \text{if} \quad x < S_{\mathrm{L}}t, \\ \mathbf{U}_{\mathrm{L}}^{\mathrm{HLLC}} & \text{if} \quad S_{\mathrm{L}}t < x < S_{\mathrm{C}}t, \\ \mathbf{U}_{\mathrm{R}}^{\mathrm{HLLC}} & \text{if} \quad S_{\mathrm{C}}t < x < S_{\mathrm{R}}t, \\ \mathbf{U}_{j+1} & \text{if} \quad x > S_{\mathrm{R}}t. \end{cases} \tag{4.1}$$

Based on this, the numerical flux function is defined as:

$$\mathbf{F}_{j+1/2} = \begin{cases} \mathbf{F}_j & \text{if} \quad 0 < S_{\mathrm{L}}, \\ \mathbf{F}_{\mathrm{L},j+1/2}^{\mathrm{HLLC}} & \text{if} \quad S_{\mathrm{L}} < 0 < S_{\mathrm{C}}, \\ \mathbf{F}_{\mathrm{R},j+1/2}^{\mathrm{HLLC}} & \text{if} \quad S_{\mathrm{C}} < 0 < S_{\mathrm{R}}, \\ \mathbf{F}_{j+1} & \text{if} \quad 0 > S_{\mathrm{R}}. \end{cases} \tag{4.2}$$

In the interesting case, $S_{\mathrm{L}} < 0 < S_{\mathrm{R}}$, the numerical flux function has the form:

$$\mathbf{F}_{\mathrm{L},j+1/2}^{\mathrm{HLLC}} = \mathbf{F}_j + S_{\mathrm{L}}\left(\mathbf{U}_{\mathrm{L},j+1/2}^{\mathrm{HLLC}} - \mathbf{U}_j\right), \tag{4.3}$$

$$\mathbf{F}_{\mathrm{R},j+1/2}^{\mathrm{HLLC}} = \mathbf{F}_{j+1} + S_{\mathrm{R}}\left(\mathbf{U}_{\mathrm{R},j+1/2}^{\mathrm{HLLC}} - \mathbf{U}_{j+1}\right), \tag{4.4}$$

where the intermediate states are determined according to [35]:

$$\mathbf{U}_{\mathrm{K}}^{\mathrm{HLLC}} = \rho_{\mathrm{K}}\left(\frac{S_{\mathrm{K}} - u_{\mathrm{K}}}{S_{\mathrm{K}} - S_{\mathrm{C}}}\right)\begin{bmatrix} 1 \\ S_{\mathrm{C}} \\ \frac{E_{\mathrm{K}}}{\rho_{\mathrm{K}}} + (S_{\mathrm{C}} - u_{\mathrm{K}})\left(S_{\mathrm{C}} + \frac{p_{\mathrm{K}}}{\rho_{\mathrm{K}}(S_{\mathrm{K}} - u_{\mathrm{K}})}\right) \end{bmatrix}, \tag{4.5}$$

where index K denotes left (L) or right (R) state in (4.1). The contact discontinuity velocity is given by [35]:

$$S_{\mathrm{C}} = \frac{p_{\mathrm{R}} - p_{\mathrm{L}} + \rho_{\mathrm{L}} u_{\mathrm{L}}(S_{\mathrm{L}} - u_{\mathrm{L}}) - \rho_{\mathrm{R}} u_{\mathrm{R}}(S_{\mathrm{R}} - u_{\mathrm{R}})}{\rho_{\mathrm{L}}(S_{\mathrm{L}} - u_{\mathrm{L}}) - \rho_{\mathrm{R}}(S_{\mathrm{R}} - u_{\mathrm{R}})}. \tag{4.6}$$

For details on the derivation of these formulae we refer to the book by Toro [35].

4.2. LTS-HLLC scheme. Following the approaches of section 3, we obtain the following expression for the numerical flux to be used in (2.3):

PROPOSITION 4. *The numerical flux of the LTS-HLLC scheme (4.2) is:*

$$\mathbf{F}^{LTS\text{-}HLLC}_{j+1/2} = \mathbf{F}^{0}_{j+1/2} + \sum_{i=1}^{\infty} \mathbf{F}^{-i}_{j+1/2-i} + \sum_{i=1}^{\infty} \mathbf{F}^{+i}_{j+1/2+i}, \tag{4.7}$$

where $\mathbf{F}^{0}_{j+1/2}$ is the standard HLLC flux given by (4.2), and the additional terms are:

$$\begin{aligned} \mathbf{F}^{-i}_{j+1/2-i} &= S^{-i}_{R,j+1/2-i}\left(\mathbf{U}^{HLLC}_{R,j+1/2-i} - \mathbf{U}_{j+1-i}\right) \\ &+ S^{-i}_{C,j+1/2-i}\left(\mathbf{U}^{HLLC}_{L,j+1/2-i} - \mathbf{U}^{HLLC}_{R,j+1/2-i}\right) \\ &+ S^{-i}_{L,j+1/2-i}\left(\mathbf{U}_{j-i} - \mathbf{U}^{HLLC}_{L,j+1/2-i}\right), \end{aligned} \tag{4.8}$$

$$\begin{aligned} \mathbf{F}^{i+}_{j+1/2+i} &= S^{+i}_{L,j+1/2+i}\left(\mathbf{U}^{HLLC}_{L,j+1/2+i} - \mathbf{U}_{j+i}\right) \\ &+ S^{+i}_{C,j+1/2+i}\left(\mathbf{U}^{HLLC}_{R,j+1/2+i} - \mathbf{U}^{HLLC}_{L,j+1/2+i}\right) \\ &+ S^{+i}_{R,j+1/2+i}\left(\mathbf{U}_{j+1+i} - \mathbf{U}^{HLLC}_{R,j+1/2+i}\right). \end{aligned} \tag{4.9}$$

Herein, the modified velocities are:

$$S^{-i}_{[L,C,R],j+1/2-i} = \max\left(S_{[L,C,R],j+1/2-i} - i\frac{\Delta t}{\Delta x}, 0\right), \tag{4.10}$$

$$S^{+i}_{[L,C,R],j+1/2+i} = \min\left(S_{[L,C,R],j+1/2+i} + i\frac{\Delta t}{\Delta x}, 0\right). \tag{4.11}$$

Proof. The HLLC Riemann solver (4.1) can be written as:

$$\begin{aligned} \widetilde{\mathbf{U}}_{j+1/2}(\zeta) &= \mathbf{U}_j + H(\zeta - S_{\mathrm{L}})\left(\mathbf{U}^{\mathrm{HLLC}}_{\mathrm{L}} - \mathbf{U}_j\right) \\ &+ H(\zeta - S_{\mathrm{C}})\left(\mathbf{U}^{\mathrm{HLLC}}_{\mathrm{R}} - \mathbf{U}^{\mathrm{HLLC}}_{\mathrm{L}}\right) + H(\zeta - S_{\mathrm{R}})\left(\mathbf{U}_{j+1} - \mathbf{U}^{\mathrm{HLLC}}_{\mathrm{R}}\right), \end{aligned} \tag{4.12}$$

or equivalently:

$$\begin{aligned} \widetilde{\mathbf{U}}_{j+1/2}(\zeta) &= \mathbf{U}_{j+1} - H(S_{\mathrm{L}} - \zeta)\left(\mathbf{U}^{\mathrm{HLLC}}_{\mathrm{L}} - \mathbf{U}_j\right) \\ &- H(S_{\mathrm{C}} - \zeta)\left(\mathbf{U}^{\mathrm{HLLC}}_{\mathrm{R}} - \mathbf{U}^{\mathrm{HLLC}}_{\mathrm{L}}\right) - H(S_{\mathrm{R}} - \zeta)\left(\mathbf{U}_{j+1} - \mathbf{U}^{\mathrm{HLLC}}_{\mathrm{R}}\right), \end{aligned} \tag{4.13}$$

where H is the Heaviside function and ζ is given by (3.25). We then use (4.12) in

(3.24) and note that for $i \leq 0$ we can write:

$$\begin{aligned}(4.14)\quad \int_{(i-1)\frac{\Delta x}{\Delta t}}^{i\frac{\Delta x}{\Delta t}} \widetilde{\mathbf{U}}_{j+1/2-i}(\zeta_i)\,\mathrm{d}\zeta_i &= \frac{\Delta x}{\Delta t}\mathbf{U}_{j-i} \\ &+ \left(\min\left(0, S_\mathrm{L} - (i-1)\frac{\Delta x}{\Delta t}\right) - \min\left(0, S_\mathrm{L} - i\frac{\Delta x}{\Delta t}\right)\right)\left(\mathbf{U}_\mathrm{L}^\mathrm{HLLC} - \mathbf{U}_{j-i}\right) \\ &+ \left(\min\left(0, S_\mathrm{C} - (i-1)\frac{\Delta x}{\Delta t}\right) - \min\left(0, S_\mathrm{C} - i\frac{\Delta x}{\Delta t}\right)\right)\left(\mathbf{U}_\mathrm{R}^\mathrm{HLLC} - \mathbf{U}_\mathrm{L}^\mathrm{HLLC}\right) \\ &+ \left(\min\left(0, S_\mathrm{R} - (i-1)\frac{\Delta x}{\Delta t}\right) - \min\left(0, S_\mathrm{R} - i\frac{\Delta x}{\Delta t}\right)\right)\left(\mathbf{U}_{j+1-i} - \mathbf{U}_\mathrm{R}^\mathrm{HLLC}\right).\end{aligned}$$

Similarly, we use (4.13) in (3.24) and note that for $i \geq 1$ we can write:

$$\begin{aligned}(4.15)\quad \int_{(i-1)\frac{\Delta x}{\Delta t}}^{i\frac{\Delta x}{\Delta t}} \widetilde{\mathbf{U}}_{j+1/2-i}(\zeta_i)\,\mathrm{d}\zeta_i &= \frac{\Delta x}{\Delta t}\mathbf{U}_{j+1-i} \\ &+ \left(\max\left(0, S_\mathrm{L} - (i-1)\frac{\Delta x}{\Delta t}\right) - \max\left(0, S_\mathrm{L} - i\frac{\Delta x}{\Delta t}\right)\right)\left(\mathbf{U}_\mathrm{L}^\mathrm{HLLC} - \mathbf{U}_{j-i}\right) \\ &+ \left(\max\left(0, S_\mathrm{C} - (i-1)\frac{\Delta x}{\Delta t}\right) - \max\left(0, S_\mathrm{C} - i\frac{\Delta x}{\Delta t}\right)\right)\left(\mathbf{U}_\mathrm{R}^\mathrm{HLLC} - \mathbf{U}_\mathrm{L}^\mathrm{HLLC}\right) \\ &+ \left(\max\left(0, S_\mathrm{R} - (i-1)\frac{\Delta x}{\Delta t}\right) - \max\left(0, S_\mathrm{R} - i\frac{\Delta x}{\Delta t}\right)\right)\left(\mathbf{U}_{j+1-i} - \mathbf{U}_\mathrm{R}^\mathrm{HLLC}\right).\end{aligned}$$

Herein, the index $_{j+1/2-i}$ is implicitly assumed on the parameters $S_{[\mathrm{L,C,R}]}$ and $\mathbf{U}_{[\mathrm{L,R}]}^\mathrm{HLLC}$. Using (4.14) and (4.15) in (3.24) we can write the LTS-HLLC scheme as:

$$(4.16)\qquad \mathbf{U}_j^{n+1} = \mathbf{U}_j^n + \frac{\Delta t}{\Delta x}\left(\mathbf{F}_{j-1/2}^\text{LTS-HLLC} - \mathbf{F}_{j+1/2}^\text{LTS-HLLC}\right). \qquad \square$$

We note that (4.8) and (4.9) are very similar to the corresponding numerical flux functions for the LTS-HLL scheme, (3.16) and (3.17), but with the addition of the middle wave associated with S_C.

5. Results. In this section we compare the new schemes with their non-LTS counterparts and the LTS-Roe scheme. Until now, we did not discuss how to choose the wave velocity estimates for S_L and S_R in the HLL and HLLC schemes (the choice also applies to the LTS framework). For our investigations, the choice of wave velocity estimates for S_L and S_R is made according to Einfeldt [5]:

$$(5.1)\qquad S_{\mathrm{L},j+1/2} = \min\left(\lambda_1(\mathbf{U}_j), \lambda_1(\widehat{\mathbf{U}}_{j+1/2})\right),$$

$$(5.2)\qquad S_{\mathrm{R},j+1/2} = \max\left(\lambda_3(\widehat{\mathbf{U}}_{j+1/2}), \lambda_3(\mathbf{U}_{j+1})\right),$$

where $\widehat{\mathbf{U}}$ denotes the Roe average of conserved variables. For the Euler equations, the eigenvalues are defined as $\lambda_1 = u - c$ and $\lambda_3 = u + c$, where u and c are the velocity and speed of sound, respectively. We note that the choice of wave velocity estimates is not a trivial matter and refer to Davis [4], Einfeldt [5] and Toro et al. [36] for detailed discussions about a number of different estimates and their properties. Herein, we choose (5.1) and (5.2) based on our own experience, where this choice

yielded very good results, especially when it came to calculating entropy satisfying solutions. A more rigorous comparison between different wave velocity estimates in the LTS framework may be very fruitful, but at the moment it remains outside the scope of this paper.

In all the numerical experiments below, the input discretization parameters were the Courant number C and Δx. Then, the time step Δt was evaluated at each time step according to:

$$\Delta t = \frac{C\Delta x}{\max\limits_{k,x} |\lambda_k(x,t)|}, \tag{5.3}$$

where λ_k are the eigenvalues of the Jacobian matrix $\mathbf{A}$ in (2.2).

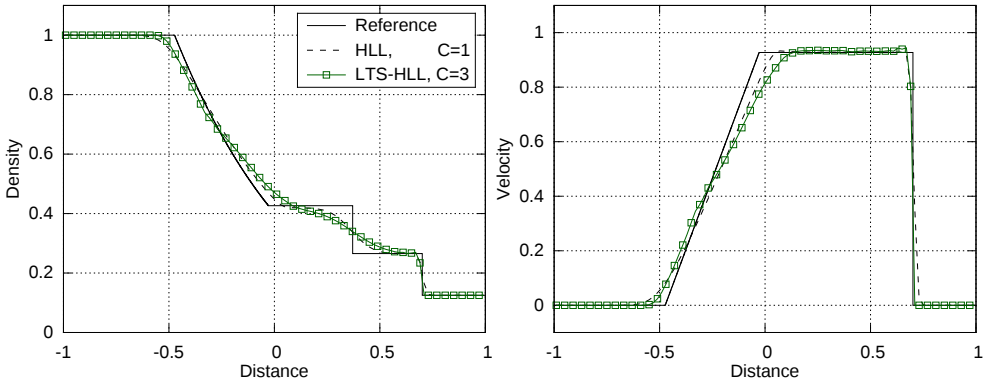
5.1. Sod shock tube. As a first test case we consider the classic Sod shock tube problem [29], with initial data $\mathbf{V}(x,0) = (\rho, u, p)^{\mathrm{T}}$:

$$\mathbf{V}(x,0) = \begin{cases} (1,0,1)^{\mathrm{T}} & \text{if} \quad x < 0, \\ (0.125,0,0.1)^{\mathrm{T}} & \text{if} \quad x > 0, \end{cases} \tag{5.4}$$

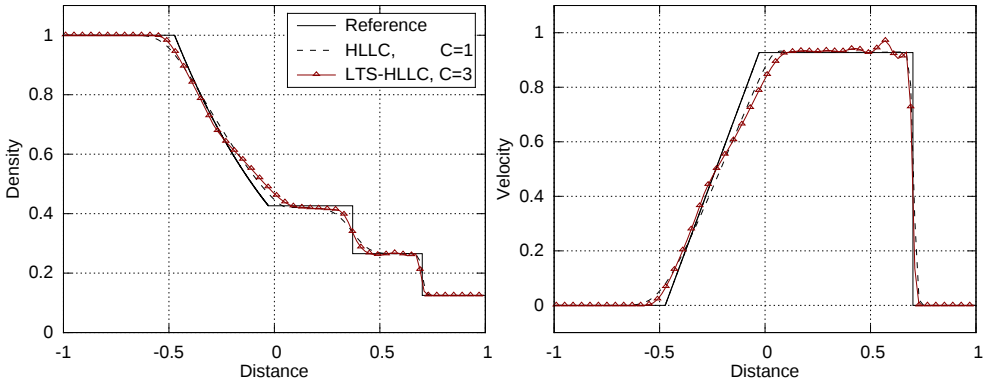
where the solution is evaluated at $t = 0.4$ on a grid with 100 cells. Figure 2 shows the results obtained with HLL(C) and LTS-HLL(C) schemes with $C = 1$ and $C = 3$. We observe that the LTS-HLL scheme (Figure 2a) increases the accuracy of the shock and the left going part of the rarefaction wave, while increasing the diffusion of the contact discontinuity. This is due to the fact that the standard HLL scheme assumes a two wave structure of the solution and neglects the contact discontinuity, leading to excessive diffusion. Since the LTS-HLL scheme maintains the two wave assumption, it can be seen that the increase in the time step leads to further smearing of the contact discontinuity. The LTS-HLLC scheme (Figure 2b) also improves the accuracy of the shock and the rarefaction wave. In addition, the LTS-HLLC scheme also improves the accuracy of the contact discontinuity, because the HLLC scheme resolves the wave missing in the HLL scheme. The velocity profiles show that the LTS-HLLC scheme produces more spurious oscillations than the LTS-HLL scheme.

Next, we compare the performance of the LTS schemes to each other. We consider the same test case and also include the results obtained with the LTS-Roe scheme [15]. Figure 3 shows that the LTS-Roe scheme produces spurious oscillations in both density and internal energy. Further, we observe that the LTS-Roe scheme violates the entropy condition, while both LTS-HLL and LTS-HLLC schemes produce entropy satisfying solutions. We note that the LTS-HLL(C) schemes produce entropy satisfying solution, because we use the wave velocity estimates (5.1) and (5.2).

We also compare the error estimates and the convergence rates for the standard HLL(C) scheme, HLL(C) scheme with the superbee wave limiter (*HLL(C)+WL*) and the LTS-HLL(C) scheme at different Courant numbers and grid sizes. Table 1 shows that the grid refinement indicates convergence of the LTS-HLL(C) schemes, and that the convergence rate tends to increase as we increase the Courant number. This suggests that as we refine the grid the higher Courant numbers will achieve more accurate solutions. A similar behavior is observed for the accuracy and the convergence rate of the other variables as well. The convergence tables for all variables (density, velocity, pressure and internal energy) for both LTS-HLL(C) schemes can be found in the Supplement (Tables S1 to S8). Last, we investigate the computational times for the LTS-HLL(C) schemes at different Courant numbers and different grids, see Figure 4. We observe that for any grid, the CPU time decreases as we increase the Courant



(a) Standard HLL and LTS-HLL scheme



(b) Standard HLLC and LTS-HLLC scheme

Fig. 2: Comparison between the standard HLL(C) and the LTS-HLL(C) schemes for the problem (5.4)

number. However, by looking at the CPU time required to reach the same error we observe that the HLL scheme tends to be more efficient than the LTS-HLL scheme, and that the LTS-HLLC scheme tends to be more efficient than the HLLC scheme.

Remark 5. The CPU times are obtained with the MATLAB tic-toc function and averaged over a number of simulations. The computational times in Figure 4 correspond to implementation in the framework (2.3) with the numerical flux functions evaluated with (3.15) for the LTS-HLL and (4.7) for the LTS-HLLC scheme. We note that for the LTS-HLL scheme the similar computational efficiency trends are observed for implementations in the numerical viscosity framework (2.12) with (3.22), and the flux-difference splitting framework (2.13) with (3.28). Similar computational efficiency trends were reported by Lindqvist and Lund [16] and Prebeg et al. [25].

5.2. Woodward-Colella blast-wave problem. We consider the Woodward-Colella blast-wave problem [37]. The initial data is given by uniform density $\rho(x,0) = 1$,

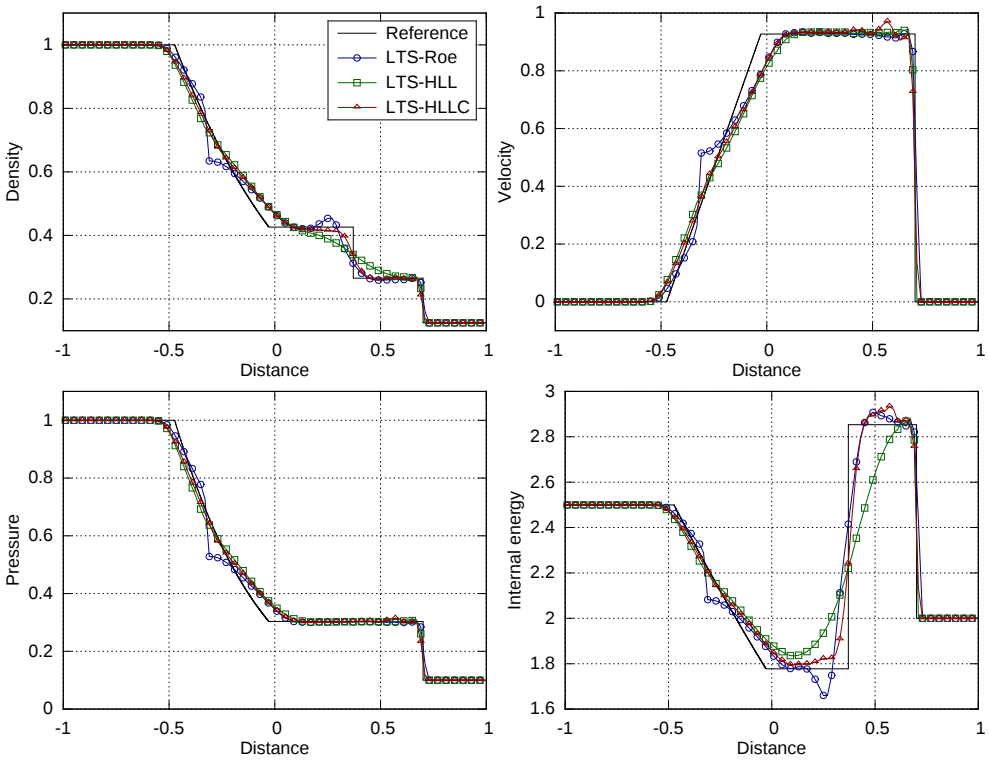


Fig. 3: Comparison between different LTS schemes at $C = 3$ for problem (5.4)

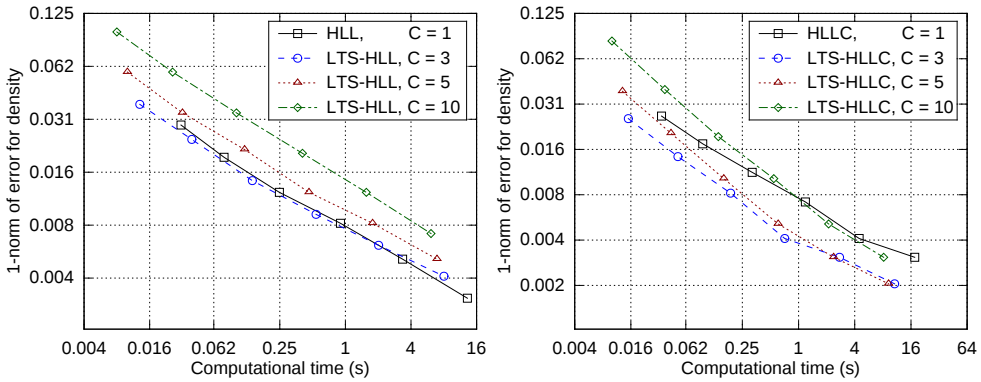


Fig. 4: Computational time vs. error estimate $\mathcal{E}$ for density with the LTS-HLL(C) schemes for the problem (5.4) with 100, 200, 400, 800, 1600 and 3200 cells

uniform velocity $u(x,0) = 0$, and two discontinuities in the pressure:

$$p(x,0) = \begin{cases} 1000 & \text{if} \quad 0 < x < 0.1, \\ 0.01 & \text{if} \quad 0.1 < x < 0.9, \\ 100 & \text{if} \quad 0.9 < x < 1. \end{cases} \tag{5.5}$$

Table 1: 1-norm error estimates $\mathcal{E}$ ($\times 10^{-2}$) and convergence rates $\mathcal{L}$ of density for problem (5.4) with LTS-HLL(C) schemes

(a) LTS-HLL

	HLL		HLL+WL		LTS-HLL		LTS-HLL		LTS-HLL	
$C =$	1		1		3		5		10	
n	$\mathcal{E}_n$	$\mathcal{L}_n$	$\mathcal{E}_n$	$\mathcal{L}_n$	$\mathcal{E}_n$	$\mathcal{L}_n$	$\mathcal{E}_n$	$\mathcal{L}_n$	$\mathcal{E}_n$	$\mathcal{L}_n$
100	2.886	–	1.553	–	3.781	–	5.836	–	9.802	–
200	1.916	0.591	0.998	0.638	2.399	0.656	3.415	0.773	5.801	0.757
400	1.202	0.672	0.609	0.713	1.429	0.747	2.054	0.734	3.360	0.788
800	0.753	0.675	0.381	0.677	0.873	0.711	1.220	0.750	2.005	0.745
1600	0.484	0.638	0.259	0.557	0.561	0.638	0.763	0.678	1.203	0.737
3200	0.307	0.655	0.172	0.587	0.363	0.627	0.483	0.659	0.743	0.695

(b) LTS-HLLC

	HLLC		HLLC+WL		LTS-HLLC		LTS-HLLC		LTS-HLLC	
$C =$	1		1		3		5		10	
n	$\mathcal{E}_n$	$\mathcal{L}_n$	$\mathcal{E}_n$	$\mathcal{L}_n$	$\mathcal{E}_n$	$\mathcal{L}_n$	$\mathcal{E}_n$	$\mathcal{L}_n$	$\mathcal{E}_n$	$\mathcal{L}_n$
100	2.610	–	0.753	–	2.456	–	3.762	–	8.243	–
200	1.749	0.577	0.392	0.941	1.399	0.812	1.981	0.925	3.865	1.093
400	1.104	0.663	0.182	1.109	0.761	0.879	1.027	0.947	1.943	0.992
800	0.689	0.680	0.087	1.068	0.434	0.810	0.536	0.938	0.977	0.992
1600	0.443	0.638	0.049	0.805	0.266	0.704	0.295	0.861	0.517	0.917
3200	0.280	0.663	0.023	1.080	0.159	0.744	0.162	0.868	0.275	0.911

The solution is evaluated at $t = 0.038$ on a grid with 500 cells. The solution consists of contact discontinuities at $x = 0.6$, $x = 0.76$ and $x = 0.8$ and shock waves at $x = 0.65$ and $x = 0.87$, see [14]. The boundary walls at $x = 0$ and $x = 1$ are modeled as reflective boundary condition. The reference solution was obtained by the Roe scheme with the superbee wave limiter on the grid with 16000 cells.

Figure 5 shows the results obtained with the standard HLLC scheme at $C = 1$ and different LTS schemes at $C = 5$. We observe that both LTS-Roe and LTS-HLLC schemes are more accurate than the standard HLLC scheme. Next, we observe that all schemes correctly capture the positions of both shocks and contact discontinuities. As expected, all schemes resolve the shocks much more accurately than the contact discontinuities, especially the LTS-HLL scheme which introduces very strong diffusion at the contact discontinuities.

We again compare the error estimates and the convergence rates for different schemes at the different Courant numbers and grid sizes. Table 2 shows that the grid refinement indicates convergence of the LTS-HLL(C) schemes, and that the convergence rate tends to increase as we increase the Courant number. A similar behavior is observed for the accuracy and the convergence rate of the other variables as well. The convergence tables for all variables (density, velocity, pressure and internal energy) for both LTS-HLL(C) schemes can be found in the Supplement (Tables S9 to S16). Last, we investigate the computational time and convergence rate for the LTS-HLL(C) schemes for different Courant numbers and different grids, see Figure 6. We observe that for any grid, the CPU time decreases as we increase the Courant number. For the LTS-HLL scheme, the optimal choice of the Courant number depends on the grid size. The LTS-HLLC scheme is always more efficient than the HLLC scheme. The observations made in Remark 5 also apply for Figure 6.

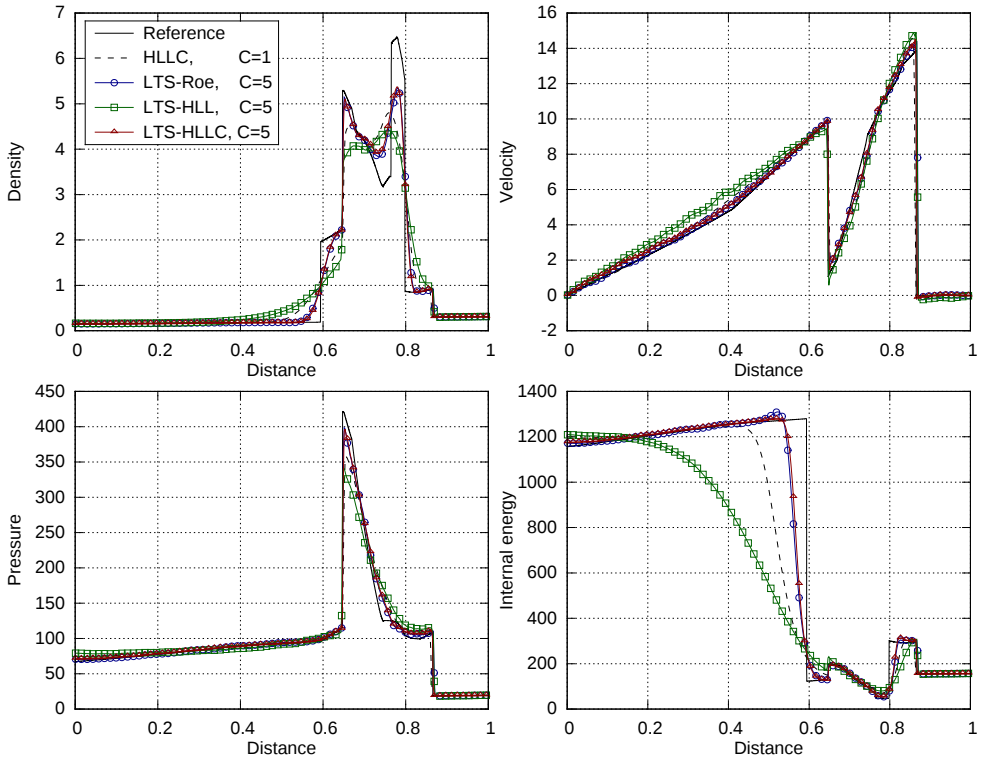


Fig. 5: Comparison between the standard HLLC and different LTS schemes for problem (5.5)

Table 2: 1-norm error estimates $\mathcal{E}$ ($\times 10^{-1}$) and convergence rates $\mathcal{L}$ of density for problem (5.5) with LTS-HLL(C) schemes

(a) LTS-HLL

	HLL		HLL+WL		LTS-HLL		LTS-HLL	
$C=$	1		1		3		5	
n	$\mathcal{E}_n$	$\mathcal{L}_n$	$\mathcal{E}_n$	$\mathcal{L}_n$	$\mathcal{E}_n$	$\mathcal{L}_n$	$\mathcal{E}_n$	$\mathcal{L}_n$
100	3.711	–	3.032	–	4.266	–	4.713	–
200	3.267	0.184	2.236	0.439	3.555	0.263	4.085	0.206
400	2.715	0.267	1.582	0.499	2.836	0.326	3.329	0.295
800	2.152	0.335	1.038	0.608	2.148	0.400	2.555	0.382
1600	1.629	0.402	0.691	0.588	1.580	0.443	1.888	0.436
3200	1.172	0.475	0.450	0.617	1.126	0.487	1.356	0.478

(b) LTS-HLLC

	HLLC		HLLC+WL		LTS-HLLC		LTS-HLLC	
$C=$	1		1		3		5	
n	$\mathcal{E}_n$	$\mathcal{L}_n$	$\mathcal{E}_n$	$\mathcal{L}_n$	$\mathcal{E}_n$	$\mathcal{L}_n$	$\mathcal{E}_n$	$\mathcal{L}_n$
100	3.603	–	2.160	–	2.658	–	2.334	–
200	3.207	0.168	1.373	0.654	2.253	0.239	1.953	0.257
400	2.649	0.275	0.684	1.004	1.795	0.327	1.490	0.390
800	2.068	0.357	0.350	0.968	1.358	0.403	1.082	0.462
1600	1.541	0.425	0.194	0.854	1.005	0.435	0.796	0.423
3200	1.095	0.492	0.093	1.064	0.713	0.494	0.561	0.504

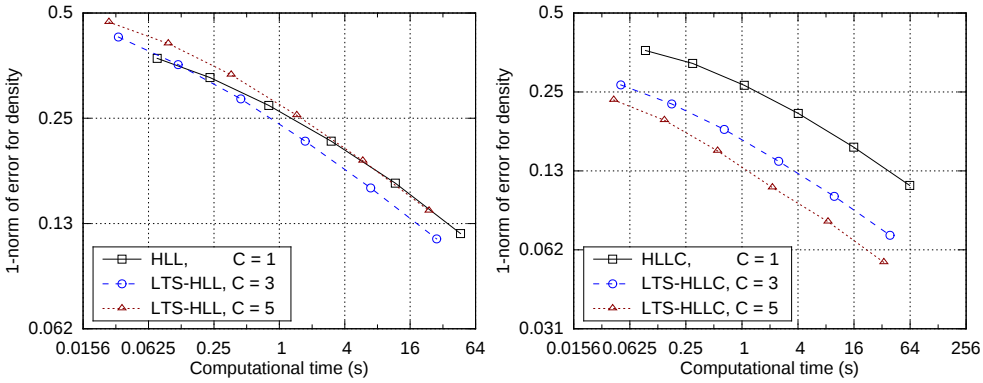


Fig. 6: Computational time vs. error estimate $\mathcal{E}$ for density with the LTS-HLL(C) schemes for the problem (5.5) with 100, 200, 400, 800, 1600 and 3200 cells

6. Conclusions. We constructed the Large Time Step extensions of the HLL and HLLC schemes. Main results of this paper are Propositions 2 and 3 where we determine the explicit expressions for the flux-difference splitting coefficients and the numerical viscosity coefficients of the LTS-HLL scheme.

We applied the LTS-HLL(C) schemes to a one dimensional test cases for the Euler equations. At moderate Courant numbers the LTS-HLL scheme leads to increased accuracy of shocks and rarefaction waves, and further decreases the resolution of the contact discontinuity. At the same time, the LTS-HLLC scheme leads to an increased accuracy of shocks, rarefaction waves and contact discontinuities. Further, for an appropriate choice of the wave velocity estimates both schemes yielded entropy satisfying solutions. This is a notable improvement compared to the existing LTS-Roe scheme for which entropy violations are observed for even more cases than the standard Roe scheme. In addition to this, the new schemes are able to handle a combination of very strong shocks, interaction of multiple waves and reflection of waves from walls, as was demonstrated by the example of the Woodward-Colella blast-wave problem. The LTS-HLLC scheme tends to be more efficient than the standard HLLC scheme. By further increasing the Courant number, both schemes produced spurious oscillations and the accuracy decreased.

The problem of spurious oscillations in the LTS-Roe was investigated by Lindqvist et al. [15] and Solberg [30]. Therein, the oscillations are reduced by introducing numerical diffusion by taking convex combinations between the LTS-Roe and the LTS-Lax-Friedrich scheme. It may be more convenient to add numerical diffusion in the framework of LTS-HLL(C) schemes, since the choice of the wave velocity estimates provides greater flexibility in the amount of numerical diffusion we introduce.

Standard HLL(C) schemes have the nice property of being positivity preserving for an appropriate choice of the wave velocity estimates [6, 1]. It remains to be explored under which conditions LTS-HLL(C) schemes preserve this valuable property.

REFERENCES

[1] P. Batten, N. Clarke, C. Lambert, and D. Causon, *On the choice of wave speeds for the HLLC Riemann solver*, SIAM J. Sci. Comput., 18 (1997), pp. 1553–1570, doi:10.1137/S1064827593260140.

[2] F. Daude and P. Galon, *On the computation of the Baer-Nunziato model using ALE formulation with HLL- and HLLC-type solvers towards fluid-structure interactions*, J. Comput. Phys., 304 (2016), pp. 189–230, doi:10.1016/j.jcp.2015.09.056.

[3] F. Daude, P. Galon, Z. Gao, and E. Blaud, *Numerical experiments using a HLLC-type scheme with ALE formulation for compressible two-phase flows five-equation models with phase transition*, Comput. Fluids, 94 (2014), pp. 112–138, doi:10.1016/j.compfluid.2014.02.008.

[4] S. Davis, *Simplified second-order Godunov-type methods*, SIAM J. Sci. Stat. Comput., 9 (1988), pp. 445–473, doi:10.1137/0909030.

[5] B. Einfeldt, *On Godunov-type methods for gas dynamics*, SIAM J. Numer. Anal., 25 (1988), pp. 294–318, doi:10.1137/0725021.

[6] B. Einfeldt, C. Munz, P. Roe, and B. Sjögreen, *On Godunov-type methods near low densities*, J. Comput. Phys., 92 (1991), pp. 273–295, doi:10.1016/0021-9991(91)90211-3.

[7] S. Godunov, *A finite difference method for the computation of discontinuous solutions of the equations of fluid dynamics*, Mat. Sb., 89 (1959), pp. 271–306.

[8] A. Harten, *High resolution schemes for hyperbolic conservation laws*, J. Comput. Phys., 49 (1983), pp. 379–399, doi:10.1016/0021-9991(83)90136-5.

[9] A. Harten, P. D. Lax, and B. van Leer, *On upstream differencing and Godunov-type schemes for hyperbolic conservation laws*, SIAM Rev., 25 (1983), pp. 35–61, doi:10.1137/1025002.

[10] P. Janhunen, *A positive conservative method for magnetohydrodynamics based on HLL and Roe methods*, J. Comput. Phys., 160 (2000), pp. 649–661, doi:doi:10.1006/jcph.2000.6479.

[11] R. LeVeque, *Large time step shock-capturing techniques for scalar conservation laws*, SIAM J. Numer. Anal., 19 (1982), pp. 1091–1109, doi:10.1137/0719080.

[12] R. LeVeque, *Convergence of a large time step generalization of Godunov's method for conservation laws*, Comm. Pure Appl. Math., 37 (1984), pp. 463–477, doi:10.1002/cpa.3160370405.

[13] R. LeVeque, *A large time step generalization of Godunov's method for systems of conservation laws*, SIAM J. Numer. Anal., 22 (1985), pp. 1051–1073, doi:10.1137/0722063.

[14] R. LeVeque, *Finite Volume Methods for Hyperbolic Problems*, Cambridge University Press, 2002, doi:10.1017/CBO9780511791253.

[15] S. Lindqvist, P. Aursand, T. Flåtten, and A. Solberg, *Large Time Step TVD schemes for hyperbolic conservation laws*, SIAM J. Numer. Anal., 54 (2016), pp. 2775–2798, doi:10.1137/15M104935X.

[16] S. Lindqvist and H. Lund, *A Large Time Step Roe scheme applied to two-phase flow*, in VII European Congress on Computational Methods in Applied Sciences and Engineering, M. Papadrakakis, V. Papadopoulos, G. Stefanou, and V. Pleveris, eds., Crete Island, Greece, 2016.

[17] H. Lochon, F. Daude, P. Galon, and J.-M. Hérard, *HLLC-type Riemann solver with approximated two-phase contact for the computation of the Baer-Nunziato two-fluid model*, J. Comput. Phys., 326 (2016), pp. 733–762, doi:10.1016/j.jcp.2016.09.015.

[18] N. N. Makwana and A. Chatterjee, *Fast solution of time domain Maxwell's equations using large time steps*, in 2015 IEEE International Conference on Computational Electromagnetics (ICCEM 2015), Institute of Electrical and Electronics Engineers (IEEE), 2015, pp. 330–332, doi:10.1109/COMPEM.2015.7052651.

[19] T. Miyoshi and K. Kusano, *A multi-state HLL approximate Riemann solver for ideal magnetohydrodynamics*, J. Comput. Phys., 208 (2005), pp. 315–344, doi:10.1016/j.jcp.2005.02.017.

[20] M. Morales-Hernandez, P. García-Navarro, and J. Murillo, *A large time step 1D upwind explicit scheme (CFL>1): Application to shallow water equations*, J. Comput. Phys., 231 (2012), pp. 6532–6557, doi:10.1016/j.jcp.2012.06.017.

[21] M. Morales-Hernandez, M. Hubbard, and García-Navarro, *A 2D extension of a large time step explicit scheme (CFL>1) for unsteady problems with wet/dry boundaries*, J. Comput. Phys., 263 (2014), pp. 303–327, doi:10.1016/j.jcp.2014.01.019.

[22] M. Morales-Hernández, J. Murillo, P. García-Navarro, and J. Burguete, *A large time step upwind scheme for the shallow water equations with source terms*, in Numerical Methods for Hyperbolic Equations, E. V. Cendón, A. Hidalgo, P. García-Navarro, and L. Cea, eds., CRC Press, 2012, pp. 141–148, doi:10.1201/b14172-17.

[23] J. Murillo, P. García-Navarro, P. Brufau, and J. Burguete, *Extension of an explicit finite volume method to large time steps (CFL>1): application to shallow water flows*, International Journal for Numerical Methods in Fluids, 50 (2006), pp. 63–102, doi:10.1002/fld.1036.

[24] M. Pelanti and K.-M. Shyue, *A mixture-energy-consistent six-equation two-phase numerical*

model for fluids with interfaces, cavitation and evaporation waves, J. Comput. Phys., 259 (2014), pp. 331–357, doi:10.1016/j.jcp.2013.12.003.

[25] M. Prebeg, T. Flåtten, and B. Müller, *Large Time Step Roe scheme for a common 1D two-fluid model*. submitted, 2016.

[26] Z. Qian and C.-H. Lee, *A class of large time step Godunov schemes for hyperbolic conservation laws and applications*, J. Comput. Phys., 230 (2011), pp. 7418–7440, doi:10.1016/j.jcp.2011.06.008.

[27] Z. Qian and C.-H. Lee, *On large time step TVD scheme for hyperbolic conservation laws and its efficiency evaluation*, J. Comput. Phys., 231 (2012), pp. 7415–7430, doi:10.1016/j.jcp.2012.07.015.

[28] P. Roe, *Approximate Riemann solvers, parameter vectors, and difference schemes*, J. Comput. Phys., 43 (1981), pp. 357–372, doi:10.1016/0021-9991(81)90128-5.

[29] G. A. Sod, *A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws*, J. Comput. Phys., 27 (1978), pp. 1–31, doi:10.1016/0021-9991(78)90023-2.

[30] A. A. Solberg, *Large Time Step explicit schemes for partial differential evolution equations*, master's thesis, Dept. of Energy and Process Engineering, Norwegian University of Science and Technology, 2016, http://hdl.handle.net/11250/2409951.

[31] E. Tadmor, *Numerical viscosity and the entropy condition for conservative difference schemes*, Math. Comp, 43 (1984), pp. 369–381, doi:10.1090/s0025-5718-1984-0758189-x.

[32] K. Tang, A. Beccantini, and C. Corre, *Combining Discrete Equations Method and upwind downwind-controlled splitting for non-reacting and reacting two-fluid computations: One dimensional case*, Comput. Fluids, 93 (2014), pp. 74–90, doi:10.1016/j.compfluid.2014.01.017.

[33] B. Tian, E. F. Toro, and C. E. Castro, *A path-conservative method for a five-equation model of two-phase flow with an HLLC-type Riemann solver*, Comput. Fluids, 46 (2011), pp. 122–132, doi:10.1016/j.compfluid.2011.01.038.

[34] S. A. Tokareva and E. F. Toro, *HLLC-type Riemann solver for Baer-Nunziato equations of compressible two-phase flow*, J. Comput. Phys., 229 (2010), pp. 3573–3604, doi:10.1016/j.jcp.2010.01.016.

[35] E. F. Toro, *Riemann Solvers and Numerical Methods for Fluid Dynamics*, Springer Berlin Heidelberg, 3th ed., 2009, doi:10.1007/b79761.

[36] E. F. Toro, M. Spruce, and W. Speares, *Restoration of the contact surface in the HLL-Riemann solver*, Shock Waves, 4 (1994), pp. 25–34, doi:10.1007/BF01414629.

[37] P. Woodward and P. Colella, *The numerical simulation of two-dimensional fluid flow with strong shocks*, J. Comput. Phys., 54 (1984), pp. 115–173, doi:10.1016/0021-9991(84)90142-6.

[38] R. Xu, D. Zhong, B. Wu, X. Fu, and R. Miao, *A large time step Godunov scheme for free-surface shallow water equations*, Chinese Sci. Bull., 59 (2014), pp. 2534–2540, doi:10.1007/s11434-014-0374-7.

[39] G.-S. Yeom and K.-S. Chang, *Two-dimensional two-fluid two-phase flow simulation using an approximate Jacobian matrix for HLL scheme*, Numer. Heat Tr. B-Fund., 56 (2010), pp. 372–392, doi:10.1080/10407790903507998.