Centrum voor Wiskunde en Informatica

REPORT*RAPPORT*

MAS

Modelling, Analysis and Simulation

Modelling, Analysis and Simulation

Error reduction at the contact discontinuity in numerical special relativistic hydrodynamics

D.E.A. van Odyck

REPORT MAS-R0308 JULY 31, 2003

CWI is the National Research Institute for Mathematics and Computer Science. It is sponsored by the Netherlands Organization for Scientific Research (NWO).
CWI is a founding member of ERCIM, the European Research Consortium for Informatics and Mathematics.

CWI's research has a theme-oriented structure and is grouped into four clusters. Listed below are the names of the clusters and in parentheses their acronyms.

Probability, Networks and Algorithms (PNA)

Software Engineering (SEN)

Modelling, Analysis and Simulation (MAS)

Information Systems (INS)

P.O. Box 94079, 1090 GB Amsterdam (NL)
Kruislaan 413, 1098 SJ Amsterdam (NL)
Telephone +31 20 592 9333
Telefax +31 20 592 4199

ISSN 1386-3711

Error Reduction at the Contact Discontinuity in Numerical Special Relativistic Hydrodynamics

D.E.A. van Odyck
CWI
P.O. Box 94079, 1090 GB Amsterdam, The Netherlands

ABSTRACT
In this paper the quasi-1D special relativistic hydrodynamic equations are numerically solved with a Godunov-type scheme. It was found that when there is a jump in the tangential velocity component at the contact discontinuity an $O(1)$ error arises in the solution. An analysis of this phenomenon is given and possible remedies are proposed and tested.

2000 Mathematics Subject Classification: 65M60, 76L05, 76Nxx, 76Y05, 83A05, 85A30
Keywords and Phrases: special relativity, hydrodynamics, Riemann problems.
Note: This research was supported by the Netherlands Organization for Scientific Research (NWO). The work was carried out under CWI-project MAS2.1 "Computational Fluid Dynamics".

1. Introduction

From astrophysics and nuclear physics there is a high interest in fluids moving at relativistic speeds. Relativistic jets (resulting from accretion onto compact objects) and gamma-ray bursts (GRB's, high-energy explosions of not yet determined objects in the universe, at cosmological distances from the earth) can be modeled with the use of special relativistic hydrodynamics. Simulations of relativistic jets have been performed by Aloy et al. [2] and of GRB's by Piran et al. [21]. Also in the field of heavy-ion collisions special relativistic hydrodynamics can be applied, see Csernai [4].

In the past ten years a lot of research effort has been put in the development of high resolution shock capturing (HRSC) schemes to solve the special relativistic hydrodynamic (SRHD) equations. Most of the schemes which were developed to solve the Euler equations of gas dynamics have been extended to SRHD. In the following a concise list of methods is given: Godunov method: Martí & Müller [17]; Glimm's method: Wen [25]; two-shock approximation: Balsara [3] and Dai & Woodward [5]; Roe-type solver: Eulderink et al. [8, 9]; relativistic HLL (Harten, Lax, van Leer [11]) method: Schneider et al. [23]; Marquina's method: Donat & Marquina [6] and Marti et al. [18, 19]; symmetric TVD schemes: Koide et al. [13, 14].

Fortunately, an analytical solution exists for the SRHD Riemann problem without [16] or with [22] a tangential velocity component. Nevertheless, to my knowledge there are no numerical results reported in the case of the quasi-1D Riemann problem. In this paper the quasi-1D Riemann problem is studied and it is found that numerical errors of $O(1)$ arise at the contact discontinuity (CD) in the case of a jump in the tangential velocity component. The reason for this error is analyzed and possible solutions are proposed and tested.

2. SRHD equations

In this paper we consider the quasi-1D SRHD equations. For a detailed derivation of the SRHD equations see for example [15, 20, 24]. In the following the speed of light is put to unity ($c = 1$). Then the SRHD equations can be written in the following conservative form:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} = 0 \ , (x,t) \in (-\infty, \infty) \times (0, \infty)$$

$$\mathbf{U}(x, t=0) = \mathbf{U}_0(x) \ , x \in (-\infty, \infty) \ . \tag{2.1}$$

The conservative variables are defined in terms of the primitive variables density: (ρ), velocity in x-direction (v^x), velocity in y-direction (v^y) and pressure (p).

$$\mathbf{U} = \begin{pmatrix} D \\ S^x \\ S^y \\ \tau \end{pmatrix} = \begin{pmatrix} \rho\Gamma \\ \rho h \Gamma^2 v^x \\ \rho h \Gamma^2 v^y \\ \rho h \Gamma^2 - p - \rho\Gamma \end{pmatrix} , \mathbf{F} = \begin{pmatrix} D v^x \\ S^x v^x + p \\ S^y v^x \\ (\tau + p) v^x \end{pmatrix} , \tag{2.2}$$

where $\Gamma = \frac{1}{\sqrt{1-(v^x)^2-(v^y)^2}}$ is the Lorentz factor, $h = 1 + \bar{\gamma}\frac{p}{\rho}$ is the enthalpy and $\bar{\gamma} = \frac{\gamma}{(\gamma-1)}$. In the limit $c \to \infty$ the Euler equations are recovered. The primitive variables are calculated from the conservative variables by solving the following equation for p:

$$f(p) \equiv \tau + p + D(1-\Gamma) - \bar{\gamma} p \Gamma^2 = 0 \ . \tag{2.3}$$

The pressure can then be used to calculate the density and the velocity components according to:

$$\begin{array}{lcl} \rho & = & \frac{D}{\Gamma} \\ v^x & = & \frac{S^x}{\tau+p+D} \\ v^y & = & \frac{S^y}{\tau+p+D} \end{array} \ . \tag{2.4}$$

In this paper four model problems are studied. They are defined by the initial conditions in Table 1. The left (right) state is defined on the domain $x \leq 0.5$ ($x > 0.5$).

	Problem 1		Problem 2		Problem 3		Problem 4	
	L	R	L	R	L	R	L	R
p	13.3	0.01	1000.0	0.01	1.0	1.0	1.0	1.0
ρ	10.0	1.0	1.0	1.0	2.0	1.0	1.0	1.0
v^x	0.0	0.0	0.0	0.0	0.4	0.4	0.4	0.0
v^y	0.9	0.0	0.6	0.0	0.9	0.0	0.9	0.0

Table 1: *Left and right initial values for Problem 1, 2, 3 and 4.*

Problem 1 and 2 are frequently used in the literature to test algorithms for solving the SRHD equations, but with $(v^y)_L = 0$. A useful quantity to locate a CD is:

$$\psi = h\Gamma v^y \ . \tag{2.5}$$

In Pons [22] it is shown that the quantity ψ only jumps across a CD.

3. Numerical solution

The starting point to discretize equation (2.1) is the Godunov approach [10]. A mesh is defined in the (x,t)-plane in order to discretize equation (2.1). The computational domain is restricted to $x \in [0,1]$. The points on the mesh are at locations ($x_i = i\Delta x, \ t^n = n\Delta t$) with $i = 0,..,N_x$ and $n = 0,..,N_t$. The discrete values of $\vec{U}(x,t)$ at ($i\Delta x, \ n\Delta t$) will be denoted by $\vec{U}_i^n$ and $\vec{U}_i^n$ is defined by:

$$\vec{U}_i^n = \frac{1}{\Delta x} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \mathbf{U}(s, n\Delta t) ds \ . \tag{3.1}$$

The conserved variables $\vec{U}(x,t)$ are advanced in time in the following way:

$$\vec{U}_i^{n+1} = \vec{U}_i^n + \frac{\Delta t}{\Delta x} \left(\vec{F}_{i-\frac{1}{2}} - \vec{F}_{i+\frac{1}{2}} \right) , \tag{3.2}$$

where $\vec{F}_{i+\frac{1}{2}}$ is the flux through the cell face located at $x_{i+\frac{1}{2}}$. To ensure that no waves interact within a cell, the time step must satisfy the condition:

$$\Delta t = \sigma \frac{\Delta x}{S^n_{max}} \ , \ 0 < \sigma \leq 1 \ , \tag{3.3}$$

where σ is the Courant number and S^n_{max} the largest signal velocity in the domain at a certain time step t^n. In all following calculations we take for simplicity $S^n_{max} = 1$, the speed of light.

In this paper different ways of approximating the flux on the boundary of the cell are used. The eigenvalues (λ_p) and eigenvectors ($\mathbf{r}_p$) of $\frac{\partial \mathbf{F}}{\partial \mathbf{U}}$ are known, see Donat [7]. They can be used in the Lax-Friedrichs (LF) scheme. The flux is computed according to:

$$\vec{F}_{i+\frac{1}{2}} = \frac{1}{2}\left(\vec{F}^n_i + \vec{F}^n_{i+1} - C^{max}_{i+\frac{1}{2}}(\vec{U}^n_{i+1} - \vec{U}^n_i)\right) \ , \tag{3.4}$$

where

$$C^{max}_{i+\frac{1}{2}} = max[C^{max}_i, C^{max}_{i+1}] \ , \ C^{max}_i = (max_p[|\lambda_p|])_i \ . \tag{3.5}$$

The Roe-approach developed to calculate the fluxes in the case of the Euler equations can be extended to the SRHD equations as was done in Eulderink [8, 9]. The fluxes are computed according to:

$$\vec{F}_{i+\frac{1}{2}} = \frac{1}{2}\left(\vec{F}^n_i + \vec{F}^n_{i+1} - \sum_p |\hat{\lambda}_p| \alpha_p \vec{e}_p\right) \ , \tag{3.6}$$

and

$$\vec{U}^n_{i+1} - \vec{U}^n_i = \sum_p \alpha_p \vec{e}_p \ . \tag{3.7}$$

The $\hat{\lambda}_p$ and $\vec{e}_p$ are the eigenvalues and eigenvectors of the Roe-matrix defined as $\hat{A}(\vec{U}^n_{i+1}, \vec{U}^n_i)(\vec{U}^n_{i+1} - \vec{U}^n_i) = \vec{F}^n_{i+1} - \vec{F}^n_i$. The explicit expressions for $\hat{\lambda}_p$, $\vec{e}_p$ and α_p are given in Eulderink [8] pg. 19.

The Roe solver is complemented with the Harten [12] sonic entropy fix to eliminate expansion shocks. It has the simple form:

$$|\hat{\lambda}_p| \rightarrow |\hat{\lambda}_p| \text{ if } |\hat{\lambda}_p| \geq \beta \ ,$$

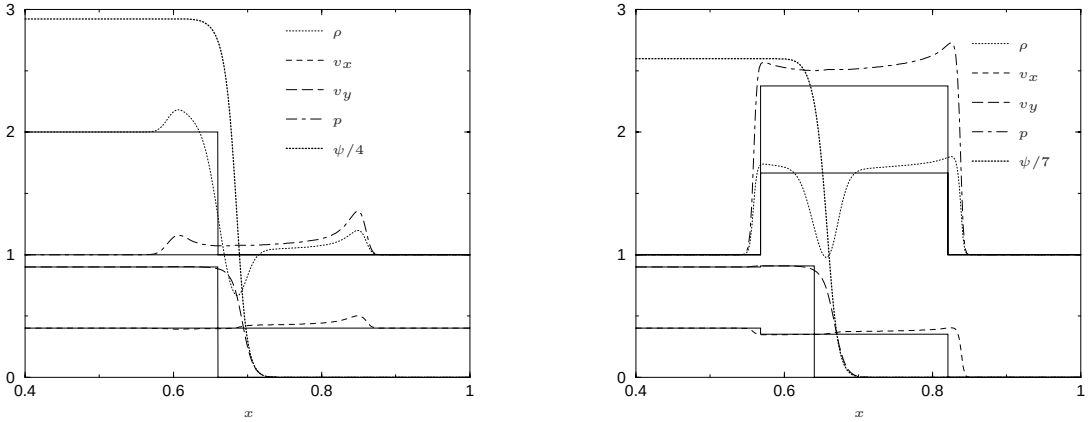
$$|\hat{\lambda}_p| \rightarrow \frac{1}{2}(\frac{\hat{\lambda}^2_p}{\beta} + \beta) \text{ if } |\hat{\lambda}_p| < \beta \ , \ \beta > 0 \ , \tag{3.8}$$

for the p which represent the non-entropy waves. In the following we take $\beta = 0.5$. To the discretized SRHD equations, the in/outflow boundary conditions at $x = 0, 1$: $\vec{U}^n_{-1} = \vec{U}^n_1$ and $\vec{U}^n_{N_x+1} = \vec{U}^n_{N_x}$ are added.

4. Moving contact discontinuity

Problem 3 and 4 have been solved with both the Roe and the LF scheme. Both give the same type of solution. At the CD there are quantities (p, v^x) which are not constant in contrast to the theory [22] which states that across a CD $[p] = 0$ and $[v^x] = 0$. The bracket indicates the difference of the quantity across a shock or a CD. It was found numerically that the non-constant behavior of some quantities only exists if the CD is moving and if there is a jump in v^y at the CD. In Figure 1 the Roe solution is plotted for Problem 3 and 4. The LF solution is not plotted because it shows the same behavior as the Roe solution. It is observed that for Problem 4 at the location of the CD the density is not constant in contrast to the exact solution. The minimal value of the density dip is mesh

independent, the dip only narrows with decreasing mesh size. Already Problem 3, a CD moving from left to right, shows a large error arising at the CD. Figure 1 also shows that only at the CD ψ changes rapidly. It was found, in the case of Problem 3, that if the numerical solution is diffusion free, take $\Delta t = \frac{\Delta x}{\text{speed of CD}}$, there are no oscillations in the pressure and in the velocity v^x. In the following section Problem 3 is further analysed.



a. Solution of Problem 3. *b. Solution of Problem 4.*

Figure 1: *Solid line discrete exact solution.* $\Delta x = \Delta t$, $T = 0.4$, $N_x = 400$.

4.1 Analysis of moving CD

The oscillations in the primitive variables already exist after performing one time step with equation (3.2) independent of the used flux formula: Roe or LF. When computing from the conservative variables the primitive variables the result does not give $[p] = 0$ and $[v_x] = 0$ at the CD. In the following, Problem 3 with $\rho_L = \rho_R = 1$ is analyzed in more detail. At $t = n\Delta t$ the CD is at the cell face $x_{i-\frac{1}{2}}$ of the i^{th} cell. The exact solution of this problem is known in advance: the left state $\mathbf{U}_L$ is propagated with speed $v_s = v_L^x$ from left to right. The discrete exact solution can be computed from:

$$\begin{aligned} \vec{U}_i^{n+1} &= \tfrac{1}{\Delta x}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \mathbf{U}(s,(n+1)\Delta t)\; ds \\ &= \tfrac{1}{\Delta x}\left\{\int_{x_{i-\frac{1}{2}}}^{x_{i-\frac{1}{2}}+v_s\Delta t} \mathbf{U}_L\; ds + \int_{x_{i-\frac{1}{2}}+v_s\Delta t}^{x_{i+\frac{1}{2}}} \mathbf{U}_R\; ds\right\} \\ &= \tilde{\sigma}\mathbf{U}_L + (1-\tilde{\sigma})\mathbf{U}_R \quad ,\tilde{\sigma} = v_s\tfrac{\Delta t}{\Delta x}\;,\; 0 \le \tilde{\sigma} \le 1. \end{aligned} \tag{4.1}$$

In order to study how good the computed primitive variables approach the exact values we insert the exact solution for the pressure ($p_i^{n+1} = 1$) and the density ($\rho_i^{n+1} = 1$) in equations (2.3) and (2.4). This gives:

$$(v^x)_i^{n+1} = \frac{(S^x)_i^{n+1}}{\tau_i^{n+1} + D_i^{n+1} + 1} = v_s\;, \tag{4.2}$$

$$(v^y)_i^{n+1} = \frac{(S^y)_i^{n+1}}{\tau_i^{n+1} + D_i^{n+1} + 1} = \frac{\tilde{\sigma} v_L^y (1 - v_s^2)}{1 - v_s^2 - (1-\tilde{\sigma})(v_L^y)^2}\;. \tag{4.3}$$

The $(v^x)_i^{n+1}$ is correctly calculated and it can be shown that $0 < (v^y)_i^{n+1} \le v_L^y$. The equation for the density gives

$$f_1 \equiv 1 - D_i^{n+1}\sqrt{(1 - (v_L^x)^2 - ((v^y)_i^{n+1})^2)} = 0\;, \tag{4.4}$$

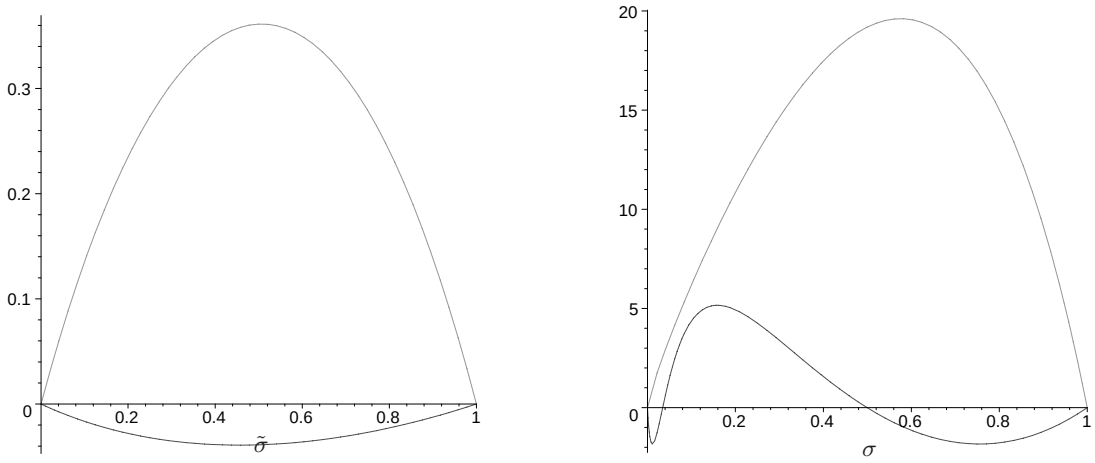
and the equation for the pressure gives:

$$f_2 \equiv \tau_i^{n+1} + 1 + D_i^{n+1}(1-\Gamma) - \bar{\gamma}\Gamma^2 = 0 \ , \tag{4.5}$$

with

$$\Gamma = \frac{1}{\sqrt{1-(v_L^x)^2 - ((v^y)_i^{n+1})^2}} \ . \tag{4.6}$$

In Figure 2 two different situations ($v_L^y = 0.5$ and $v_L^y = 0.9$) have been plotted. It is clear that not for all $\tilde{\sigma}$ both f_1 and f_2 are zero. It was also found that $f_1 = 0$ and $f_2 = 0$ if $v_L^y = v_R^y \geq 0$ for $0 \leq \sigma \leq 1$. This was expected because we already mentioned that only if there is a jump in v^y the solution of Problem 3 contains oscillations in the pressure and the speed.



a. $v_l^y = 0.5$. *b.* $v_l^y = 0.9$.

Figure 2: f_1 *and* f_2 *as function of* $\tilde{\sigma}$.

5. Fixes for moving CD

In the following two sections different fixes for the problems mentioned in the last section are introduced: the averaging method and the Simple Single Fluid (SSF) algorithm.

5.1 Averaging, fix for Problem 3

Let us first focus on Problem 3. In the following, use is made of the fact that the behavior of the solution at the CD is known. The CD is located at the point where $[\psi] = [h\Gamma v_y] = [S^y/D] \neq 0$. In a numerical calculation the location of the CD is not a point but as a result of numerical diffusion a region of finite width. The algorithm for calculating the moving contact discontinuity in Problem 3 is:

1. First distinguish the region G where the CD is located with:

$$\left|\psi_i^n - \psi_{i-1}^n\right| > \epsilon \, , \ \epsilon > 0 \ . \tag{5.1}$$

2. Do a calculation with $v^y = 0$. This gives $(\rho, v^x, p)_i^{n+1}$. Because Problem 3 consists only of a moving CD a correct calculation with $v^y \neq 0$ would give the same result for $(\rho, v^x, p)_i^{n+1}$.

3. Do a calculation with $v^y \neq 0$. This gives $(S^y)_i^{n+1}$. It is observed that the combination of $(D, S^y, \tau)_i^{n+1}$ calculated with $v^y = 0$ and $(S^y)_i^{n+1}$ calculated with $v^y \neq 0$ does not give the desired values of the primitive variables in the CD region.

4. Assume that with the above calculations $(\rho, v^x, p)_i^{n+1}$ and $(S^y)_i^{n+1}$ are correctly computed. Now demand that $\forall i \in G$:

$$(v^x)_i^{n+1} = \frac{(S^x)_i^{n+1}}{\tau_i^{n+1} + D_i^{n+1} + p_i^{n+1}} \ , \tag{5.2}$$

$$\rho_i^{n+1} = D_i^{n+1}\sqrt{1 - ((v^x)_i^{n+1})^2 - \left(\frac{(S^y)_i^{n+1}}{\tau_i^{n+1} + D_i^{n+1} + p_i^{n+1}}\right)^2} \ , \tag{5.3}$$

$$\tau_i^{n+1} + D_i^{n+1} + p_i^{n+1} - D_i^{n+1}\Gamma_i^{n+1} - \bar{\gamma} p_i^{n+1}(\Gamma_i^{n+1})^2 = 0 \ . \tag{5.4}$$

5. Equations (5.2), (5.3) and (5.4) are solved for $(S_x, D, \tau)_i^{n+1}$:

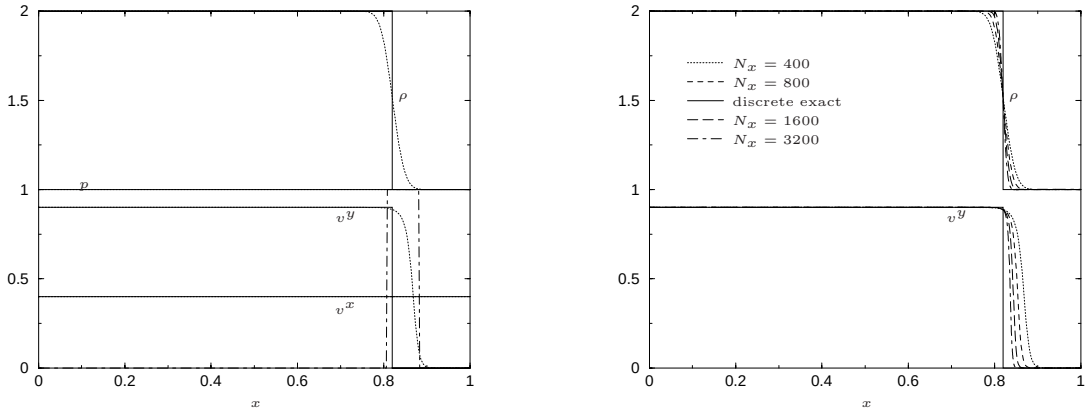
$$D_i^{n+1} = \frac{\rho_i^{n+1}}{\sqrt{2}\sqrt{1 - ((v^x)_i^{n+1})^2}}\sqrt{1 + \sqrt{1 + 4((S^y)_i^{n+1})^2 \frac{1 - ((v^x)_i^{n+1})^2}{\rho_i^{n+1} + \bar{\gamma} p_i^{n+1}}}} \ , \tag{5.5}$$

$$\tau_i^{n+1} = -D_i^{n+1} - p_i^{n+1} + \frac{|(S^y)_i^{n+1}| D_i^{n+1}}{\sqrt{(D_i^{n+1})^2(1 - ((v^x)_i^{n+1})^2) - (\rho_i^{n+1})^2}} \ , \tag{5.6}$$

$$(S^x)_i^{n+1} = (\tau_i^{n+1} + D_i^{n+1} + p_i^{n+1})(v^x)_i^{n+1} \ . \tag{5.7}$$

The calculated values for $(\rho, v^x, p)_i^{n+1}$ in Step 2 are only relevant in this specific problem because the exact solutions for p, ρ, v^x are the same for $v^y = 0$ and for $v^y \neq 0$.

In Figure 3(a) the solution to Problem 3 is plotted using the above described algorithm. The pressure and the x-component of the velocity are on the whole domain constant. Figure 3(b) shows that an increase of points results in a better approximation of the numerical solution. The above algorithm



a. Dotted-dashed line indicates region G. $N_x = 400$. *b. Solution for different N_x.*

Figure 3: *Problem 3, solid line represents discrete exact solution. $T = 0.8$, $DT = DX$ and $\epsilon = 0.1$.*

is not conservative anymore as a result of Step 5. This is illustrated in Figure 4 where a numerical solution for Problem 3 with $v_L^y = 0.915872$, this corresponds to a Lorentz factor $\Gamma_{max} = 29.1$, is plotted. It is clear that the jump in v^y is not located at the correct point. In the case of Problem 3 the method gives satisfactory solutions. In the next section the method will be extended to situations where it is not possible to calculate $(\rho, v^x, p)_i^{n+1}$ with $v^y = 0$.

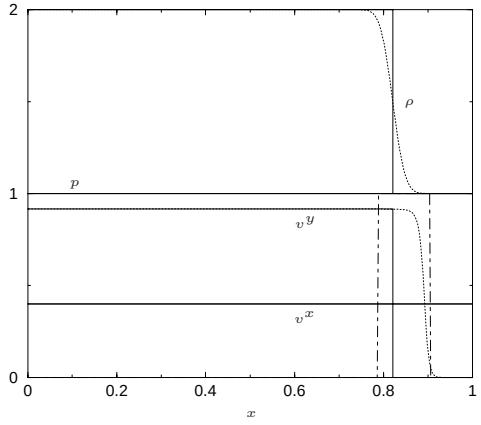


Figure 4: *Solution of Problem 3. Solid line represents discrete exact solution. Dotted-dashed line indicates region G.* $(v^y)_L = 0.915872, \Gamma_{max} = 29.1$*,* $N_x = 400, T = 0.8, DT = DX$*,* $\epsilon = 0.1$*.*

5.2 Averaging, fix for Problem 4

Almost the same procedure as in the case of Problem 3 can be used. Only $(p, \rho, v^x)_i^{n+1}$ are calculated in a different way. A more general criterion is used to define the region G. Therefore, the maximum in the quantity ψ at time step n is introduced:

$$\left|\frac{\partial \psi}{\partial x}\right|_{max}^{n} = max_i \left(\left|\frac{\psi_{i+1}^n - \psi_{i-1}^n}{2\Delta x}\right|\right) \ , \ i = 1..N_x. \tag{5.8}$$

This maximum is labeled with $i = i_{max}$. The algorithm then contains the following steps:

1. First do a time step with a Godunov-type scheme. This gives $\vec{U}_i^{n+1}$.

2. Distinguish the region where the CD is located. This region G is defined by the points which fulfill the condition:

$$\text{IF } \left(\frac{|\psi_i - \psi_{i-1}|}{\Delta x} > \epsilon \left|\frac{\partial \psi}{\partial x}\right|_{max}^{n} \text{ AND } \frac{|\psi_{i+1} - \psi_i|}{\Delta x} > \epsilon \left|\frac{\partial \psi}{\partial x}\right|_{max}^{n}\right) \text{ THEN } i \in G \ . \tag{5.9}$$

   It was found that as a result of diffusion the location of the CD is not exactly at the point $(|\partial\psi/\partial x|_{max}^n$. So a criterion to find the CD region can not only be built on the behavior of $\partial\psi/\partial x$. To the above criterion the following criterion is added:

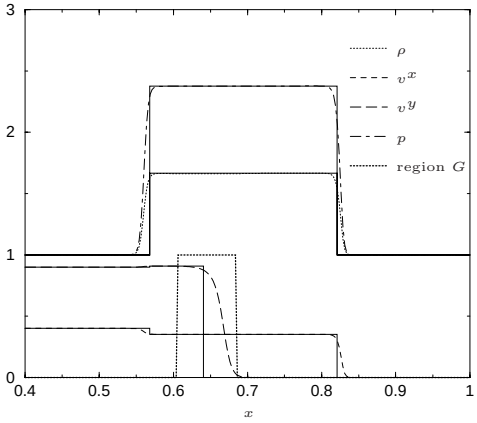
$$\text{IF } (i > i_{max} \text{ AND } (v^y)_i > \delta) \text{ THEN } i \in G \ , \tag{5.10}$$

   where unless otherwise stated $\delta = 10^{-6}$. This criterion is only relevant for the model problems used in this paper.

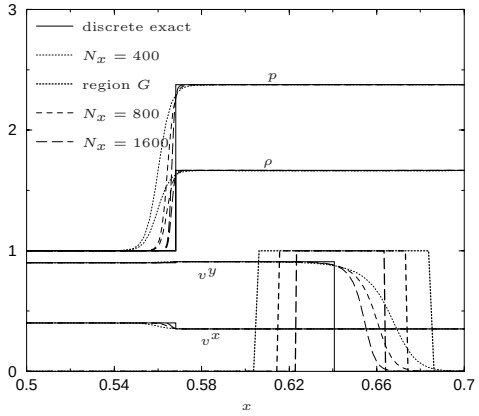
3. In this region averaged values for $(p, \rho, v^x)_i^{n+1}$ are calculated according to:

$$p_i^{n+1} = \frac{\sum_{j\in G} p_j^{n+1}}{\sum_{j\in G}} \ , \ \rho_i^{n+1} = \frac{\sum_{j\in G} \rho_j^{n+1}}{\sum_{j\in G}} \ , \ (v^x)_i^{n+1} = \frac{\sum_{j\in G} (v^x)_j^{n+1}}{\sum_{j\in G}} \ , i \in G \ . \tag{5.11}$$

4. Step 5 described in Section 5.1 is used to calculate from $(p, \rho, v^x, S^y)_i^{n+1}$ the values for $(D, S^x, S^y, \tau)_i^{n+1}$, $\forall i \in G$.

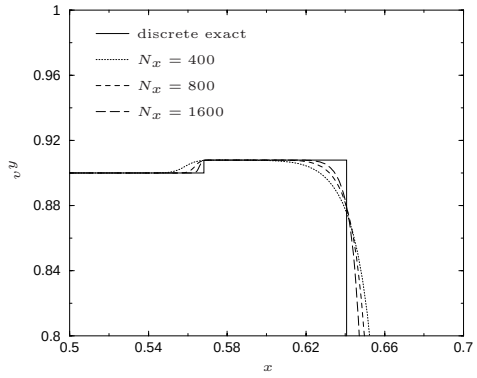


a. $N_x = 400$.

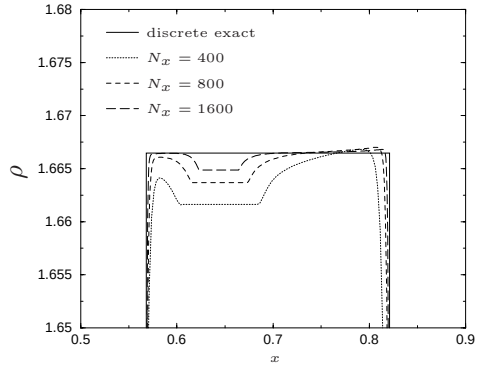


b. Solution for different N_x.

Figure 5: *Problem 4 with* $T = 0.4, \Delta t = \Delta x$, $\epsilon = 0.1$.



a. v^y *as function of* x.



b. ρ *as function of* x.

Figure 6: *Problem 4 with* $T = 0.4, \Delta t = \Delta x$, $\epsilon = 0.1$.

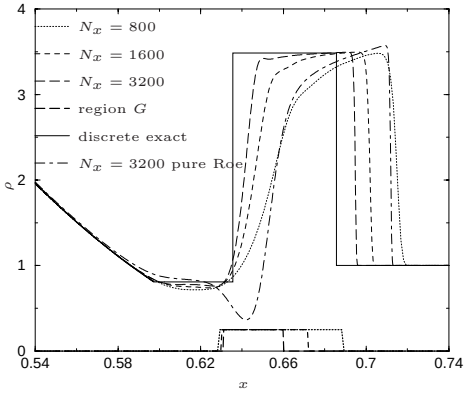
In Figure 5 and 6 the corrected solutions for Problem 4 are plotted. They show good agreement with the analytical solution.

Finally, the method is used to solve a more realistic problem. Consider Problem 1 with $(v^y)_L = 0.9$. For this problem the density is not averaged in the region G but the original density, as calculated in Step 1, is used. It was found that especially in the beginning the function $\partial\psi/\partial x$ fluctuates in the region of interest. Therefore, the first ten time steps no fix is used. With this choice of the region G Problem 1 can be solved successfully. This is illustrated in Figure 7. It is observed that the location of the shock is improved substantially with respect to the solution without the special treatment of the pressure and the velocity.

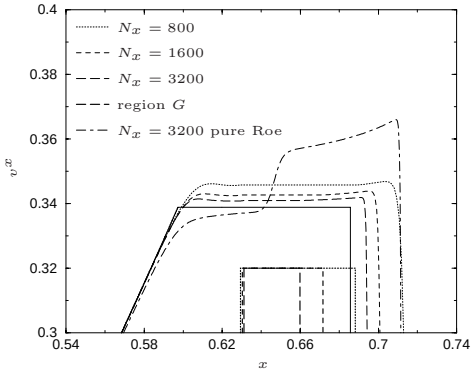
6. Simple single fluid algorithm

Another approach to overcome the pressure oscillations is to use the "Simple single-fluid" (SSF) algorithm introduced in [1]. It uses the fact that for constant v^y the numerical solution does not have oscillations. The SRHD equations are split into two parts:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} = 0 \ , \tag{6.1}$$



a. Density as function of x.



b. v^x as function of x.

Figure 7: *Problem 1 with $v_L^y = 0.9$. $\sigma = 0.8$, $T = 0.4$.*

where

$$\mathbf{U} = \begin{pmatrix} D \\ S^x \\ \tau \end{pmatrix} \ , \ \mathbf{F} = \begin{pmatrix} Dv^x \\ S^x v^x + p \\ (\tau + p)v^x \end{pmatrix} \ , \tag{6.2}$$

and

$$\frac{\partial \psi}{\partial t} + v^x \frac{\partial \psi}{\partial x} = 0 \ . \tag{6.3}$$

In this method two fluxes are calculated:

$$\vec{F}^L_{i+\frac{1}{2}}(\vec{U}^n_i, \vec{U}^n_{i+1}, (v^y)^n_i) \ , \ \vec{F}^R_{i+\frac{1}{2}}(\vec{U}^n_i, \vec{U}^n_{i+1}, (v^y)^n_{i+1}) \ . \tag{6.4}$$

The Godunov scheme now looks like:

$$\vec{U}^{n+1}_i = \vec{U}^n_i - \frac{\Delta t}{\Delta x}\left(\vec{F}^L_{i+\frac{1}{2}}(\vec{U}^n_i, \vec{U}^n_{i+1}, (v^y)^n_i) - \vec{F}^R_{i-\frac{1}{2}}(\vec{U}^n_i, \vec{U}^n_{i-1}, (v^y)^n_i)\right) \ , \tag{6.5}$$

and ψ is advected. It is observed that the characteristic decomposition for equation (6.1) is not the same as for pure 1D SRHD because v^y is now involved. After a long and elaborate calculation it was found that the eigenvalues of $\frac{\partial \mathbf{F}}{\partial \mathbf{U}}$ are:

$$\lambda_0 = v^x \ , \ \lambda_\pm = \frac{v^x(1 + \frac{1}{2}(v^y)^2(c_s^2 + \gamma - 1) - c_s^2)}{1 - (v^y)^2(1-\gamma) - (v^x)^2 c_s^2} +$$

$$\frac{\pm\sqrt{\frac{1}{4}(v^x)^2(v^y)^4(c_s^2 + \gamma - 1)^2 + c_s^2((v^x)^2 + (v^y)^2 - 1)((v^y)^2(1-\gamma) + (v^x)^2 - 1)}}{1 - (v^y)^2(1-\gamma) - (v^x)^2 c_s^2} \ . \tag{6.6}$$

The SSF-algorithm is used together with the LF-flux. We proceed as follows:

1. Initialize the primitive variables ($\mathbf{W} = (\rho, v^x, p)^T$) and the conservative variables ($\mathbf{U} = (\rho\Gamma, \rho h \Gamma^2 v^x, \rho h \Gamma^2 - p - \rho\Gamma)^T$) with $\Gamma = \frac{1}{\sqrt{1-(v^x)^2-(v^y)^2}}$.

2. Do a time-step according to equation (6.5). Take for the Lax-Friedrichs flux:

$$\vec{F}^L_{i+\frac{1}{2}}(\vec{U}^n_i, \vec{U}^n_{i+1}, (v^y)^n_i) = \frac{1}{2}\left(\vec{F}^n_i + \vec{F}^n_{i+1} - C^{max}_{i+\frac{1}{2}}(\vec{U}^n_{i+1} - \vec{U}^n_i)\right)_{v^y = (v^y)^n_i} \ . \tag{6.7}$$

3. Calculate the primitive variables in each cell from $\mathbf{U}_i^{n+1}$ and $(v^y)_i^n$. Solve for p_i^{n+1}:

$$\tau_i^{n+1} + p_i^{n+1} + D_i^{n+1}(1 - \Gamma_i^{n+1}) - \bar{\gamma} p_i^{n+1} (\Gamma_i^{n+1})^2 = 0 \ , \tag{6.8}$$

with

$$\Gamma_i^{n+1} = \frac{1}{\sqrt{1 - (\frac{(S^x)_i^{n+1}}{\tau_i^{n+1} + D_i^{n+1} + p_i^{n+1}})^2 - ((v^y)_i^n)^2}} \ . \tag{6.9}$$

4. Calculate ψ^{n+1} (upwind):

$$\psi_i^{n+1} = \psi_i^n - \frac{\Delta t}{\Delta x} |(v^x)_i^{n+1}| (\frac{1}{2}(1 - sign((v^x)_i^{n+1}))(\psi_{i+1}^n - \psi_i^n) + \frac{1}{2}(1 + sign((v^x)_i^{n+1}))(\psi_i^n - \psi_{i-1}^n)) \ . \tag{6.10}$$

5. Use $\vec{W}_i^{n+1}$ and ψ_i^{n+1} to calculate $(v^y)_i^{n+1}$:

$$(v^y)_i^{n+1} = \frac{\psi_i^{n+1}}{h_i^{n+1}} \sqrt{\frac{1 - ((v^x)_i^{n+1})^2}{1 + (\frac{\psi_i^{n+1}}{h_i^{n+1}})^2}} \ . \tag{6.11}$$

6. Use $\vec{W}_i^{n+1}$ and $(v^y)_i^{n+1}$ to calculate $\vec{U}_i^{n+1}$
7. Go to 2

The results for Problem 3 are shown in Figure 8. It is clear that the method keeps p and v^x constant. The jump in v^y is a little out of phase with respect to the exact solution. This is not a surprise because the method lost its conservative character. The results for Problem 4 are shown in Figure 9.

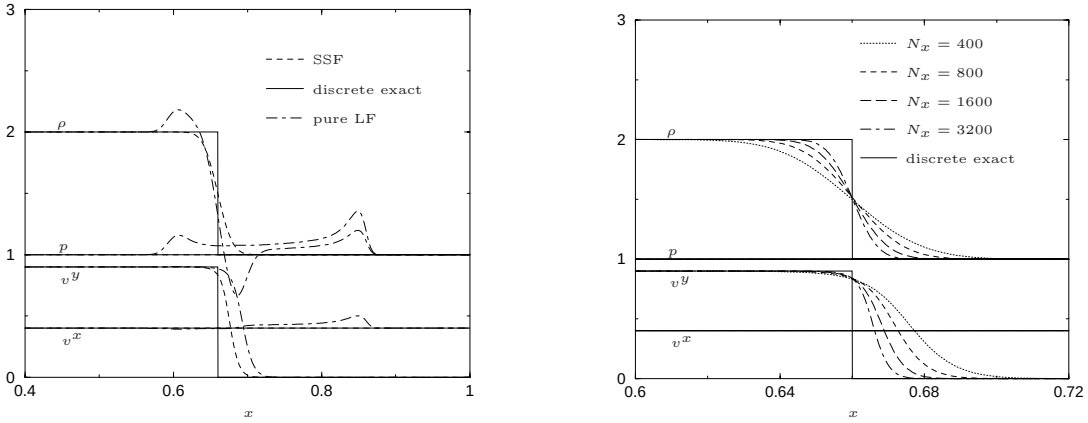
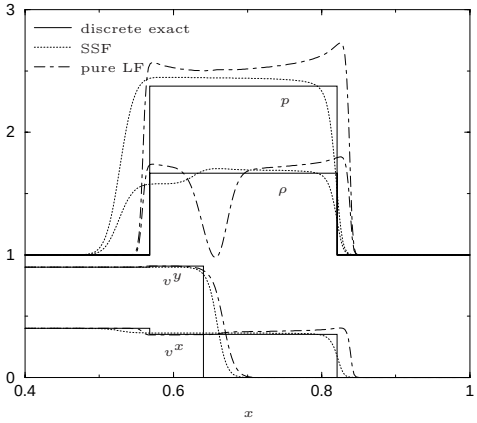


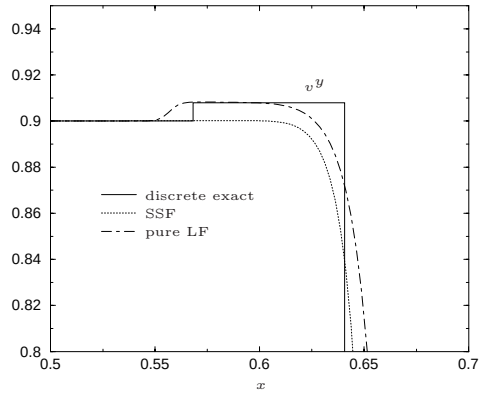
Figure 8: *Problem 3.* $T = 0.4, \Delta t = \Delta x$.

The slow-moving shock is not captured well. Especially the jump in v^y is not captured. To overcome this inaccuracy the SSF-algorithm is only used in the region of the CD. This is achieved by defining the CD region G as in equation (5.9). The results for Problem 4 are plotted in Figure 10. This figure shows that the numerical solution tends to the analytical solution.

Unfortunately, also this algorithm has a drawback. It is demonstrated in the next example, take Problem 3 with $\rho_L = 1.0$, $\rho_R = 10.0$ and $v_L^y = 0.6$. Figure 11 shows a bump in the v^y profile. The p

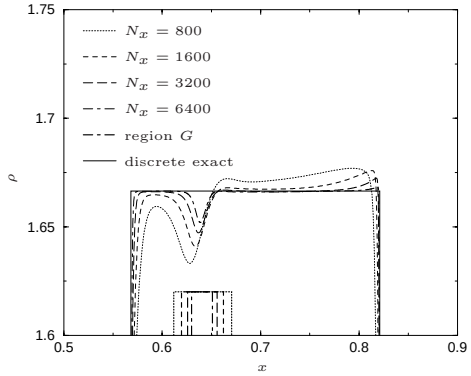


a. ρ, v^x, v^y, p *as function of* x.

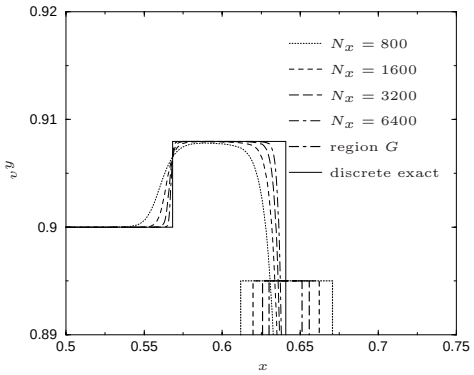


b. v^y *as function of* x.

Figure 9: *Problem 4.* $T = 0.4, \Delta t = \Delta x, N_x = 400$.



a. Density as function of x.



b. v^y *as function of* x.

Figure 10: *Problem 4.* $T = 0.4, \Delta t = \Delta x$, $\epsilon = 0.01$.

and v^x are still constant. If instead of updating ψ, S^y is updated with an upwind algorithm then this bump disappears. So in step 4 of the SSF algorithm ψ is exchanged for S^y. And step 5 now becomes:

$$(v^y)_i^{n+1} = \frac{-1 + \sqrt{1 + 4a^2(1 - ((v^x)_i^{n+1})^2}}{2a} \ , \ a = \frac{(S^y)_i^{n+1}}{(\rho h)_i^{n+1}} \ , \tag{6.12}$$

and $(v^y)_i^{n+1} = 0$ if $(S^y)_i^{n+1} = 0$. To define the region G it is used that $\psi_i^{n+1} = (S^y/D)_i^{n+1}$.

How good is the method in solving a more realistic problem? Take Problem 2 in Table 1. The results are plotted in Figure 12. Two different definitions of region G are used. The results show that this choice of region G has got a great impact on the solution. The region G_1 is defined with equation (5.9) and region G_2 with equations (5.9) and (5.10). The region G_1 defined by ψ in equation (5.9) is too small compared to the actual, smeared CD region. Therefore an unpractical algorithm is introduced. It is only used to show that the SSF method works in principle, if the CD region is correctly defined. The region G_2 is extended to the points defined by equation (5.10). The figure shows that with this definition of the region G the numerical solution is less out of phase as is the case for the pure LF solution and for the solution with the region G_1 defined by only equation (5.9). In Figure 13 a mesh refinement study is shown, indicating convergence to the analytic solution.

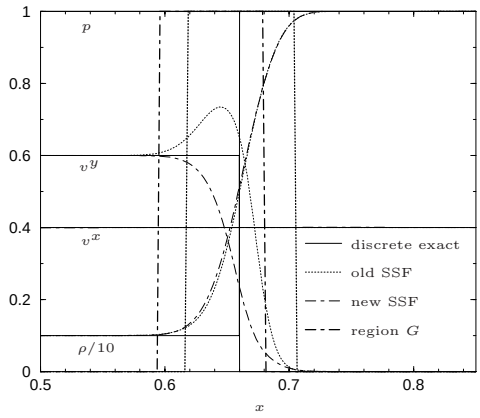


Figure 11: *Problem 3. In* old SSF ψ_i^{n+1} *is calculated. In* new SSF $(S^y)_i^{n+1}$ *is calculated.* $T = 0.4, \Delta t = \Delta x$, $\epsilon = 0.01$ *and for new fix* $\epsilon = 0.1$.

7. Conclusions

In the case of a CD with a jump in the tangential velocity component a conservative scheme cannot calculate correctly the solution. Strict conservation has to be abandoned. Two schemes are proposed: the averaging scheme and the SSF scheme. Both schemes can handle the CD but need an accurate location of the CD. Especially, for more realistic problems the solution is sensitive to this parameter dependent choice of locating the CD. It was observed that both schemes can dramatically improve the shock location in the case of Problem 1 and 2.

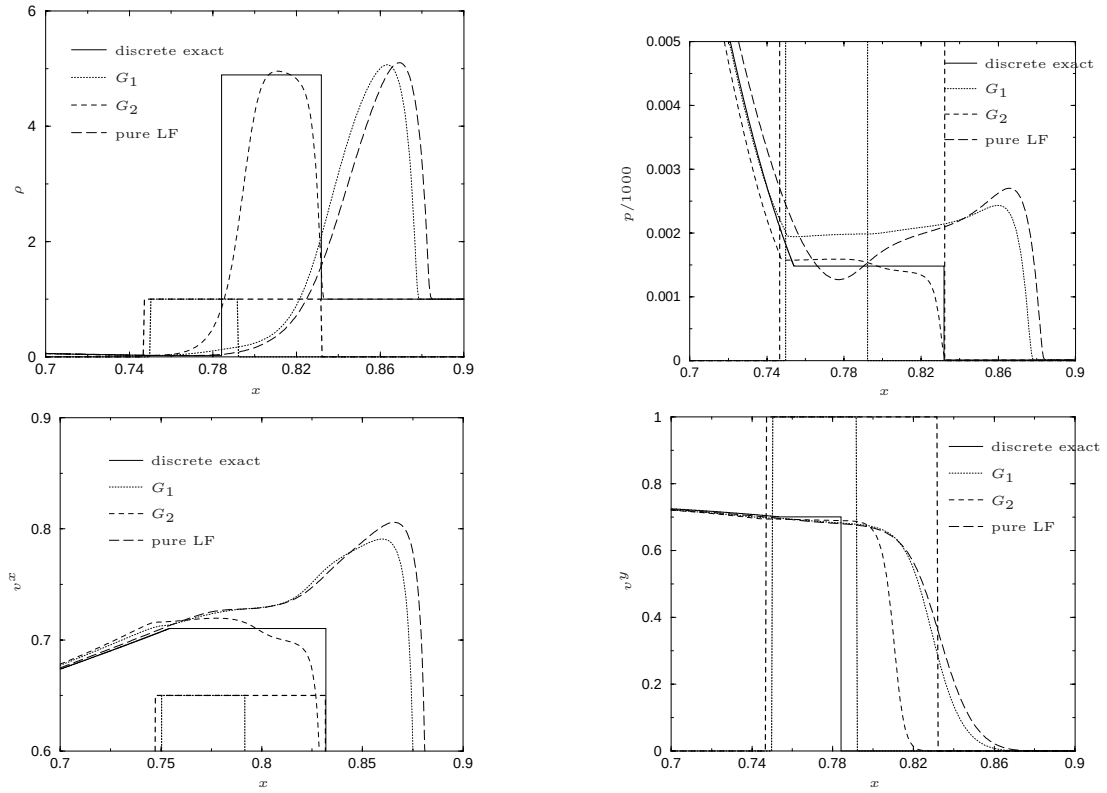


Figure 12: *Exact and LF solution for Problem 2. G_1: Region G is defined by (5.9). G_2: Region G is defined by (5.9) and (5.10). pure LF: Pure LF calculation.* $N = 1600$, $T = 0.4$, $\sigma = 0.2$, $\epsilon = 0.2$ *and* $\delta = 0.2$

References

1. R. Abgrall and S. Karni, Computations of compressible multifluids, *J. Comput. Phys.*, **169**, 594–623 (2001).
2. M.A. Aloy, J.M. Ibáñez, J.M. Martí, J.L. Gómez and E. Müller, High-resolution three-dimensional simulations of relativistic jets, *Astrophys. J.*, **523**, L125–L128 (1999).
3. D.S. Balsara, Riemann solver for relativistic hydrodynamics, *J. Comput. Phys.*, **114**, 284–297 (1994).
4. L.P. Csernai, *Introduction to Relativistic Heavy Ion Collisions*, Wiley, Chichester (1994).
5. W. Dai and P.R. Woodward, An iterative Riemann solver for relativistic hydrodynamics, *SIAM J. Sci. Comput.*, **18**, 982–995 (1997).
6. R. Donat and A. Marquina, Capturing shock reflections: An improved flux formula, *J. Comput. Phys.*, **125**, 42–58 (1996).
7. R. Donat, J.A. Font, J.M. Ibáñez and A. Marquina, A flux-split algorithm applied to relativistic flows, *J. Comput. Phys.*, **146**, 58–81 (1998).
8. F. Eulderink, Numerical relativistic hydrodynamics, *PhD thesis*, State University of Leiden, (1993).
9. F. Eulderink and G. Mellema, General relativistic hydrodynamics with a Roe solver, *Astron. Astrophys. Suppl.*, **110**, 587–623 (1995).

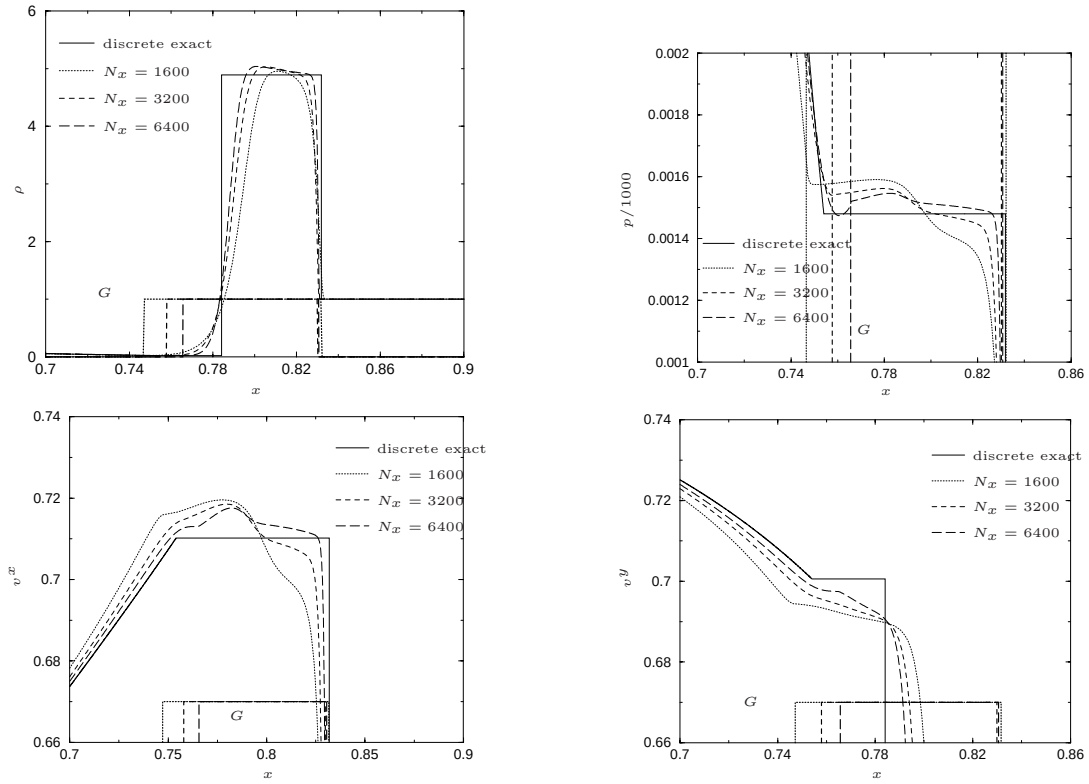


Figure 13: *Exact and LF solution for Problem 2 with region G defined by equations (5.9), (5.10) for different N. T* = 0.4, $\sigma = 0.2$, $\epsilon = 0.2$ *and* $\delta = 0.2$

10. S.K. Godunov, A finite difference method for the computation of discontinuous solutions of the equations of fluid dynamics, *Mat. Sb.*, **47**, 357–393 (1959).
11. A. Harten, P.D. Lax and B. van Leer, On upstream differencing and Godunov-type schemes for hyperbolic conservation laws, *SIAM Rev.*, **25**, 35–61 (1983).
12. A. Harten, High resolution total-variation-stable finite-difference schemes, *SIAM J. Num. Anal.*, **21**, 1–23 (1984).
13. S. Koide, K.I. Nishikawa and R.L. Muttel, A two-dimensional simulation of a relativistic magnetized jet, *Astrophys. J.*, **463**, L71–L74 (1996).
14. S. Koide, A two-dimensional simulation of a relativistic jet bent by an oblique magnetic field, *Astrophys. J.*, **487**, 66–69 (1997).
15. L.D. Landau and E.M. Lifshitz, *Fluid Mechanics*, Pergamon Press, London (1959).
16. J.M. Martí and E. Müller, The analytical solution of the Riemann problem in relativistic hydrodynamics, *J. Fluid Mech.*, **258**, 317–333 (1994).
17. J.M. Martí and E. Müller, Extension of the piecewise parabolic method to one-dimensional relativistic hydrodynamics, *J. Comput. Phys.*, **123**, 1–14 (1996).
18. J.M. Martí, E. Müller, J.A. Font and J.M. Ibáñez, Morphology and dynamics of highly supersonic relativistic jets, *Astrophys. J.*, **448**, L105–L108 (1995).
19. J.M. Martí, E. Müller, J.A. Font, J.M. Ibáñez and A. Marquina, Morphology and dynamics of relativistic jets, *Astrophys. J.*, **479**, 151–163 (1997).

20. D.E.A. van Odyck, Review of numerical special relativistic hydrodynamics, *Report MAS-R0212*, CWI, Amsterdam (2002).

21. T. Piran, A. Shemi and R. Narayan, Hydrodynamics of relativistic fireballs, *Mon. Not. R. Astron. Soc.*, **263**, 861–867 (1993).

22. J.A. Pons, J.M. Martí and E. Müller, The exact solution of the Riemann problem with non-zero tangential velocities in relativistic hydrodynamics, *J. Fluid Mech.*, **422**, 125–139 (2000).

23. V. Schneider, U. Katscher, D.H. Rischke, B. Waldhauser, J.A. Maruhn and C.D. Munz, New algorithms for ultra-relativistic numerical hydrodynamics, *J. Comput. Phys.*, **105**, 92–107 (1993).

24. B.F. Schutz, *A First Course in General Relativity*, Cambridge University Press, Cambridge (1985).

25. L. Wen, A. Panaitescu and P. Laguna, A shock-patching code for ultra-relativistic fluid flows, *Astrophys. J.*, **486**, 919–927 (1997).

Table of Contents