The Astrophysical Journal Supplement Series, 176:467–483, 2008 June
© 2008. The American Astronomical Society. All rights reserved. Printed in U.S.A.

RELATIVISTIC HYDRODYNAMIC FLOWS USING SPATIAL AND TEMPORAL ADAPTIVE STRUCTURED MESH REFINEMENT

PENG WANG, TOM ABEL, AND WEIQUN ZHANG
Kavli Institute for Particle Astrophysics and Cosmology, Stanford Linear Accelerator Center and Stanford Physics Department, Menlo Park, CA 94025; pengwang@stanford.edu, tabel@stanford.edu, wqzhang@slac.stanford.edu

ABSTRACT

Astrophysical relativistic flow problems require high-resolution three-dimensional (3D) numerical simulations. In this paper, we describe a new parallel 3D code for simulations of special relativistic hydrodynamics (SRHD) using both spatially and temporally structured adaptive mesh refinement (AMR). We used the method of lines to discretize the SRHD equations spatially and a total variation diminishing (TVD) Runge-Kutta scheme for time integration. For spatial reconstruction, we have implemented piecewise linear method (PLM), piecewise parabolic method (PPM), third-order convex essentially nonoscillatory (CENO) and third- and fifth-order weighted essentially nonoscillatory (WENO) schemes. Flux is computed using either direct flux reconstruction or approximate Riemann solvers including HLL, modified Marquina flux, local Lax-Friedrichs flux formulas, and HLLC. The AMR part of the code is built on top of the cosmological Eulerian AMR code enzo. We discuss the coupling of the AMR framework with the relativistic solvers. Via various test problems, we emphasize the importance of resolution studies in relativistic flow simulations because extremely high resolution is required especially when shear flows are present in the problem. We also present the results of two 3D simulations of astrophysical jets: AGN jets and GRB jets. Resolution study of those two cases further highlights the need of high resolutions to calculate accurately relativistic flow problems.

Subject headings: hydrodynamics — methods: numerical — relativity

1. INTRODUCTION

Relativistic flow problems are important in many astrophysical phenomena including gamma-ray bursts (GRBs), active galactic nuclei (AGNs), as well as microquasar and pulsar wind nebulae, among others. Apparent superluminal motion is observed in many jets of extragalactic radio sources associated with AGNs. According to the currently accepted standard model, this implies the jet flow velocities as large as 99% of the speed of light (Blandford et al. 1977; Begelman et al. 1984). Similar phenomena are also seen in microquasars such as GRS 1915+105 (Mirabel & et al. 1977; Begelman et al. 1984). Similar phenomena include gamma-ray bursts (GRBs), active galactic nuclei (AGNs), as well as microquasar and pulsar wind nebulae, among others. Apparent superluminal motion is observed in many jets of extragalactic radio sources associated with AGNs. According to the currently accepted standard model, this implies the jet flow velocities as large as 99% of the speed of light (Blandford et al. 1977; Begelman et al. 1984). Similar phenomena are also seen in microquasars such as GRS 1915+105 (Mirabel & et al. 1977; Begelman et al. 1984). This new code we call enzo.

Jim Wilson and collaborators pioneered the numerical solution of relativistic hydrodynamics equations (Wilson 1972; Centrella & Wilson 1984; Hawley et al. 1984a, 1984b). Starting with these earlier papers, this has typically been done in the context of general relativistic problems such as accretion onto black holes and supernovae explosions. The problem was recognized to be difficult to solve when the Lorentz factor becomes large (Norman & Winkler 1986), and a solution with an implicit adaptive scheme was demonstrated in one dimension. Unfortunately, this approach is not generalizable to multiple dimensions. However, in the past two decades accurate solvers based on Godunov’s scheme have been designed that have adopted shock-capturing schemes for Newtonian fluid to the relativistic fluid equations in conservation form (for a review see Marti & Müller 2003). Such schemes, called high-resolution shock-capturing (HRSC) methods, have been proven to be very useful in capturing strong discontinuities with a few numerical zones without serious numerical oscillations. We will discuss a number of them in § 3.

Studies involving astrophysical fluid dynamics in general are benefiting tremendously from using spatial and temporal adaptive techniques. Smoothed particle hydrodynamics (Gingold & Monaghan 1977; Lucy 1977) is a classic example by being a Lagrangian method. Increasingly, variants of the Berger & Colella (1989) structured adaptive mesh technique are also being implemented. This is also true in relativistic hydrodynamics (Hughes et al. 2002; Anninos et al. 2005; Zhang & MacFadyen 2006; Morsony et al. 2006; Meliani et al. 2007), where certainly the work of Hughes et al. (2002) showed that a serial AMR code could solve problems even highly efficient parallel fixed grid codes would have difficulty with. In this paper we discuss our implementation of different hydrodynamics solvers with various reconstruction schemes as well as different time integrators on top of the enzo framework previously developed for cosmology (Bryan & Norman 1997a, 1997b; Bryan et al. 2001; O’Shea et al. 2004). This new code we call enzo is adaptive in time and space and is a dynamically load-balanced parallel using the standard message passing interface.

In the following we briefly summarize the equations being solved before we give details on the different solvers we have implemented. Section 4 discusses the adaptive mesh refinement strategy and implementation. We then move on to describe various test problems for relevant combinations of solvers, reconstruction schemes in one, two, and three dimensions, with and without AMR. Section 6 presents an application of our code to

1 Currently at: Center for Cosmology and Particle Physics, Department of Physics, New York University, 4 Washington Place, New York, NY, 10003.
3D relativistic and supersonic jet propagation problem. Section 7 discusses 3D GRB jet simulation. We summarize our main conclusions in § 8.

2. EQUATIONS OF SPECIAL RELATIVISTIC HYDRODYNAMICS

The basic equations of special relativistic hydrodynamics (SRHD) are conservation of rest mass and energy momentum:

\[
(\rho u^\mu)_{\mu i} = 0, \tag{1}
\]

and

\[
T^\mu{}_{\nu i} = 0, \tag{2}
\]

where \(\rho\) is the rest mass density measured in the fluid frame, \(u^\mu = W(1, v^i)\) is the fluid four-velocity (assuming the speed of light \(c = 1\)), \(W\) is the Lorentz factor, \(v^i\) is the coordinate three-velocity, \(T^\mu{}_{\nu}\) is the energy-momentum tensor of the fluid, and the semicolon denotes a covariant derivative.

For a perfect fluid the energy-momentum tensor is

\[
T^\mu{}_{\nu} = \rho u^\mu u^\nu + pg^{\mu\nu}, \tag{3}
\]

where \(h = 1 + \epsilon + p/\rho\) is the relativistic specific enthalpy, \(\epsilon\) is the specific internal energy, \(p\) is the pressure, and \(g^{\mu\nu}\) is the spacetime metric.

SRHD equations can be written in the form of conservation laws

\[
\frac{\partial U}{\partial t} + \sum_{j=1}^{3} \frac{\partial F_j}{\partial x^j} = 0, \tag{4}
\]

where the conserved variable \(U\) is given by

\[
U = (D, S^1, S^2, S^3, \tau)^T, \tag{5}
\]

and the fluxes are given by

\[
F_j = (Dv^i, S^1 v^i + pb^i_1, S^2 v^i + pb^i_2, S^3 v^i + pb^i_3, S^i - Dv^i)^T. \tag{6}
\]

Anile (1989) has shown that system (4) is hyperbolic for causal EOS, i.e., those satisfying \(c_s < 1\), where the local sound speed \(c_s\) is defined as

\[
c_s^2 = \frac{1}{h} \left[ \frac{\partial p}{\partial \rho} + \left( \frac{\rho \tau}{\rho^2} \right) \frac{\partial p}{\partial \rho} \right]. \tag{7}
\]

The eigenvalues and left and right eigenvectors of the characteristic matrix \(\partial F/\partial U\), which are used in some of our numerical schemes, are given by Donat et al. (1998).

The conserved variables \(U\) are related to the primitive variables by

\[
D = \rho W, \tag{8}
\]

\[
S^i = \rho W^2 v^i, \tag{9}
\]

\[
\tau = \rho W^2 - p - \rho W, \tag{10}
\]

where \(j = 1, 2,\) and \(3\). The system (eq. [4]) is closed by an equation of state (EOS) given by \(p = p(\rho, \epsilon)\). For an ideal gas, the EOS is

\[
p = (\Gamma - 1)\rho\epsilon, \tag{11}
\]

where \(\Gamma\) is the adiabatic index.

3. NUMERICAL SCHEMES FOR SRHD

3.1. Time Integration

We use a method of lines to discretize the system (eq. [4]) spatially:

\[
\frac{dU_{i,j,k}}{dt} = -\frac{F_{x,i+1/2,j,k} - F_{x,i-1/2,j,k}}{\Delta x} \tag{12}
\]

\[
-\frac{F_{y,i,j+1/2,k} - F_{y,i,j-1/2,k}}{\Delta y} - \frac{F_{z,i,j,k+1/2} - F_{z,i,j,k-1/2}}{\Delta z} = L(U), \tag{13}
\]

where \(i, j, k\) refers to the discrete cell index in the \(x, y, z\) directions, respectively; \(F_{x,i+1/2,j,k}\), \(F_{y,i,j+1/2,k}\), and \(F_{z,i,j,k+1/2}\) are the fluxes at the cell interface.

As discussed by Shu & Osher (1988) if using a high-order scheme to reconstruct flux spatially, one must also use the appropriate multilevel total variation diminishing (TVD) Runge-Kutta schemes to integrate the ODE system (eq. [13]). Thus, we implemented the second- and third-order TVD Runge-Kutta schemes coupled with AMR.

The second-order TVD Runge-Kutta scheme reads

\[
U^{(1)} = U^n + \Delta tL(U^n), \tag{14}
\]

\[
U^{n+1} = \frac{1}{2} U^n + \frac{1}{2} U^{(1)} + \frac{1}{2} \Delta tL(U^{(1)}), \tag{15}
\]

and the third-order TVD Runge-Kutta scheme reads

\[
U^{(1)} = U^n + \Delta tL(U^n), \tag{16}
\]

\[
U^{(2)} = \frac{3}{4} U^n + \frac{1}{4} U^{(1)} + \frac{1}{4} \Delta tL(U^{(1)}), \tag{17}
\]

\[
U^{n+1} = \frac{1}{3} U^n + \frac{2}{3} U^{(2)} + \frac{2}{3} \Delta tL(U^{(2)}), \tag{18}
\]

where \(U^{n+1}\) is the final value after advancing one time step from \(U^n\).

For an explicit time integration scheme, the time step is constrained by the Courant-Friedrichs-Lewy (CFL) condition. The time step is determined as

\[
\Delta t = C\min_i \left( \frac{\Delta x_i}{\alpha i} \right), \tag{19}
\]

where \(C\) is a parameter called the CFL number and \(\alpha_i\) is the local fastest speed of propagation of characteristics in the direction \(i\), whose explicit expression can be found in Donat et al. (1998).

3.2. Reconstruction Method

Generally speaking, there are two classes of spatially reconstruction schemes (see, e.g., LeVeque 2002). One is reconstructing the unknown variables at the cell interfaces and then using an exact or approximate Riemann solver to compute the fluxes. Another is direct flux reconstruction, in which we reconstruct the flux directly from fluxes at the cell centers. To explore the coupling of different schemes with AMR as well as exploring which method is most suitable for a specific astrophysical problem, we implement several different schemes in both classes.

To reconstruct unknown variables, we have implemented the piecewise linear method (PLM; Van Leer 1979), the piecewise parabolic method (PPM; Colella & Woodward 1984; Marti &
Müller 1996), and the third-order convex essentially nonoscil-

laratory scheme (CENO; Liu & Osher 1998). These are used to
reconstruct the primitive variables since reconstructing the cons-
served variables can produce unphysical values in SRHD. Fur-
thermore, unphysical values of three-velocities may arise during
the reconstruction especially for ultrarelativistic flows. So we
either use \(v^\prime W\) to do the reconstruction or we also recon-
struct the Lorentz factor and use it to renormalize the reconstructed
three-velocity when they are unphysical.

For direct flux reconstruction, we have implemented PLM and
the third- and fifth-order WENO scheme of Jiang & Shu (1996).
Direct flux reconstruction using WENO was first used to solve
SRHD problems by Zhang & MacFadyen (2006). They showed
that the fifth-order WENO scheme works well with the third-order
Runge-Kutta time integration. In our implementation, we followed
their description closely. For the PLM and CENO schemes, we
used a generalized minmod slope limiter (Kurganov & Tadmor
2000). For given \(v_{i-1}, v_i, v_{i+1}\), \(v_{i+1/2} = v_i + 0.5\text{minmod}(|v_{i+1} - v_{i-1}|, 0.5(v_{i+1} - v_i), \theta(v_{i+1} - v_i))\), where \(1 \leq \theta \leq 2\). For \(\theta = 2\) it
reduces to the monotonized central-difference limiter of
Van Leer (1977). We found that this generalized minmod slope
limiter behaves much better than a traditional minmod limiter
(LeVeque 2002), especially for strong shear flows. In our calcu-
lation, \(\theta = 1.5\) is used by default. For the PPM scheme, we used
the parameters proposed by Marti & Müller (1996) for all the test
problems. For WENO, we used the parameters suggested in the
original paper of Jiang & Shu (1996).

3.3. Riemann Solvers

In the first class of reconstruction methods, given the recon-
structed left and right primitive variables at interfaces, the flux
cross each interface is calculated by solving the Riemann prob-
lem defined by those two states. An exact Riemann solver is quite
expensive in SRHD (Marti & Müller 1994; Pons et al. 2000).
Thus, we have implemented several approximate Riemann solv-
ers including HLL (Harten et al. 1983; Schneider et al. 1993;
Kurganov et al. 2001), HLLC (Toro et al. 1994; Mignone & Bodo
2005), local Lax-Friedrichs (LLF; Kurganov & Tadmor 2000),
and the modified Marquina flux (Aloy et al. 1999).

The HLLC scheme is an extension of the HLL solver devel-
oped by Toro et al. (1994) for Newtonian flow, which was exten-
ted to two-dimensional (2D) relativistic flows by Mignone &
Bodo (2005). The improvement of HLLC over HLL is restoring
the full wave structure by constructing the two approximate states
inside the Riemann fan. The two states can be found by the
Rankine-Hugoniot conditions between those two states and the
reconstructed states. With this modification, HLLC indeed be-
haves better than other Riemann solvers in some one-dimensional
(1D) (§5.1.7) and 2D (§5.2) test problems. But when we apply
HLLC to 3D jet simulation, we found that HLLC suffers from the
so called “carbuncle” artifact well known in the computational
fluid dynamics literature (Quirk 1994). We have used HLLC to
run many other 2D test problems designed to detect the carbuncle
artifact and confirmed this shortcomings. We found that the HLLC
solver is unsuitable for many multidimensional problems. The
discussion of these problems will be presented elsewhere. In
this work, we will only apply HLLC to two test problems showing
that the HLLC solver has less smearing at contact discontinu-
ties than other schemes.

Lucas-Serrano et al. (2004) has compared the HLL scheme,
the LLF scheme, and the modified Marquina flux formula using
1D and 2D test problems. They found that those three schemes
give similar results. However, the modified Marquina flux formula
is not as stable as HLL in problems with strong transverse flows, and
LLF is more diffusive than HLL. So in the following discussion we
will only show the results using HLL in most of the tests if there is
no difference among those three schemes.

In the following discussion, we will denote a specific hydro
algorithm by \(X - Y\), where \(X\) is the flux formula and \(Y\) is the re-
construction scheme. For example, F-WENO5 denotes direct flux
reconstruction using fifth-order WENO. We used the third-order
Runge-Kutta method for all the tests in this work.

3.4. Converting Conserved Variables to Primitive Variables

Since primitive variables are needed in the reconstruction pro-
cess, after every RK time step, we need to convert conserved vari-
ables to primitive variables. While conserved variables can be
computed directly from primitive variables using equations (8),
(9), and (10), the inverse operation is not straightforward. One
needs to solve a quartic equation for the ideal gas EOS and a non-
linear equation for more complicated EOS. Iteration methods are
used even for ideal gas EOS, because computing the solution of a
quartic is expensive. Following Aloy et al. (1999) we have used a
Newton-Raphson (NR) iteration to solve a nonlinear equation for
pressure to recover primitive variables from conserved variables.
Typically, the NR iteration needs only 2–3 steps to converge.

3.5. Curvilinear Coordinates

We have also implemented cylindrical and spherical coordi-
nates following the description of Zhang & MacFadyen (2006).
This affects three parts of the code. First, the geometric factors
are incorporated into the flux when updating the conserved var-
iables. Second, there will be geometric source terms. Third, the
flux correction in AMR (§4.4) is modified by geometric factors.

4. ADAPTIVE MESH REFINEMENT

4.1. Overview

Structured adaptive mesh refinement (AMR) was developed
by Berger & Oliger (1984) and Berger & Colella (1989) to achieve
high spatial and temporal resolution in regions where fixed grid
resolution is insufficient. In structured AMR, a subgrid will be
created in regions of its parent grid needing higher resolution. The
hierarchy of grids is a tree structure. Each grid is evolved as a se-
parate initial boundary value problem, while the whole grid hier-
archy is evolved recursively.

The renzo code is built on top of the AMR framework of enzo
(Bryan & Norman 1997a; O’Shea et al. 2004). The enzo code’s
implementation of AMR follows closely the Berger & Colella pa-
aper and has been shown to be very efficient for very high dynamic
range cosmological simulations (see, e.g., Abel et al. 2002). The
psuedocode of the main loop for the second-order Runge-Kutta
method reads as follows:

\[
\text{EvolveLevel } 1
\]
\[
\text{SetBoundaryCondition}
\]
\[
\text{while}(t_1 < t_{i-1})
\]
\[
\text{ComputeTimeStep } dt_1
\]
\[
\text{for every grid patch on this level}
\]
\[
\text{Runge-Kutta Grst step : Eq.}(14)
\]
\[
\text{ComputeFlux}
\]
\[
\text{SweepX, Y, Z}
\]
\[
\text{ChooseHydroAlgorithm}
\]
SaveSubgridFlux
UpdateConservedVariables
ConservedToPrimitive
UpdateTime \( t_{l+1} = t_l + dt_l \)
SetBoundaryCondition (§ 4.2)
for every grid patch on this level
Runge-Kutta second step : Eq.(15)
ComputeFlux
UpdateConservedVariables
ConservedToPrimitive
SetBoundaryCondition
EvolveLevel \( l + 1 \)
UpdateFromFinerGrid (§ 4.4)
FluxCorrection (§ 4.4)
RebuildHierarchy for level \( 1 \)

The RebuildHierarchy function called at the end of every time step is at the heart of AMR. Its pseudocode as implemented originally in enzo reads as follows:

RebuildHierarchy for level \( 1 \)
for \( il_{\text{level}} = 1 \) to MaximumLevel - 1
for every grid on \( il_{\text{level}} \)
FlagCellsForRefinement (§ 4.3)
CreateSubgrids
AddLevel (\( il_{\text{level}} + 1 \))
for every new subgrid
InterpolateFieldValuesFromParent (§ 4.2)
CopyFromOldSubgrids
LoadBalanceGrids

4.2. Interpolation

When a new subgrid is created, the initial values on that grid are obtained by interpolating spatially from its parent grid. In this case, we apply the conservative second-order interpolation routine provided by enzo to conserved variables. But in this process sometimes the interpolated values can violate the constraint \((r + D)^2 > S^2 + D^2\). If this happens, we will then use the first-order method for that subgrid.

Before the first Runge-Kutta step for a grid at level \( l \) in the following discussion, we use the convention that top grid has level 0, we will need the boundary condition at time \( t_l \), which is derived by interpolating from its parent grid. Then at the later steps of the Runge-Kutta scheme, one needs the boundary condition at time \( t_l + dt_l \). Since the variables of its parent grid have already been evolved to time \( t_{l-1} + dt_{l-1} \), which is greater than time \( t_l + dt_l \), we can obtain the boundary conditions at time \( t_l + dt_l \) for a grid at level \( l \) by interpolating both temporally and spatially from its parent grid. There are two exceptions to this procedure. First, if a cell of fine grid abuts the box boundary, then we just use the specified boundary condition for that cell. Second, if a cell abuts another grid at the same level, we copy the value from that grid. Because of the above mentioned problem for interpolating conserved variables, when interpolating boundary values, we apply the second-order interpolation to primitive variables. Since for ultrarelativistic flows spatially interpolating three-velocity can lead to unphysical values, we also interpolate the Lorentz factor and then use it to renormalize the interpolated three-velocity.

4.3. Refinement Criteria

In the test problems discussed in § 5, we mainly used two general purpose refinement criteria that have been widely used in AMR code (Zhang & MacFadyen 2006).

In the first one, we compute the slope

\[ S_i = \frac{|u_i+1 - u_i| - |u_i - u_i-1|}{\max (|u_i|, \epsilon)} \]

where \( u_i \) is typically density, pressure, and velocities, and \( \epsilon \) is a small number typically taken to be \( 10^{-10} \). When \( S_i \) is larger than a minimum slope, typically 1, a cell will be flagged for refinement.

In the second one, for every cell we compute

\[ E_i = \frac{|u_{i+2} - 2u_i + u_{i-2}|}{|u_{i+2} - u_{i+1}| + |u_{i+1} - u_{i-2}| + \epsilon (|u_{i+2} + 2|u_i| + |u_{i-2}|)} \]

which is the ratio of the second and first derivatives with a safety factor in the denominator. Unless otherwise stated, we use \( \epsilon = 0.01 \). When \( E_i \) is larger than a critical value \( E_{\text{crit}} \), a cell will be flagged for refinement. Typically, we use \( E_{\text{crit}} = 0.8 \).

To fully exploit AMR, it is desirable to design more specific refinement criteria that are most efficient for a specific astrophysical problems.

4.4. Flux Correction

When a cell is overlaid by a finer level grid, then the coarse grid value is just the conservative average of the fine grid values. On the other hand, when a cell abuts a fine grid interface but is not itself covered by any fine grid, we will do flux correction for that cell, i.e., we will use the fine grid flux to replace the coarser grid flux in the interface abutting the fine grid (see Berger & Colella [1989] for a more detailed description of flux correction). For this purpose, note that the second-order Runge-Kutta method can be rewritten as

\[ U^{n+1} = U^n + \frac{1}{2} \Delta t (U^n) + \frac{1}{2} \Delta t (U^{(1)}) \]

and the third-order Runge-Kutta method can be rewritten as

\[ U^{n+1} = U^n + \frac{1}{6} \Delta t (U^n) + \frac{1}{6} \Delta t (U^{(1)}) + \frac{2}{3} \Delta t (U^{(2)}) \]

Thus, for example, when we do flux correction in the x-direction for the interface \( i + 1/2 \), we will use \( F_{i+1/2}^x (U^n)/2 + F_{i+1/2}^x (U^{(1)})/2 \) and \( F_{i+1/2}^x (U^n)/6 + F_{i+1/2}^x (U^{(1)})/6 + 2F_{i+1/2}^x (U^{(2)})/3 \) to correct the coarser grid conserved variables for the second- and third-order Runge-Kutta method, respectively.

4.5. Parallelism

The enzo code uses the enzo parallel framework, which uses dynamically load balancing using the Message Passing Interface (MPI) library. At run time, the code will move grids among processors according to the current load of every processor to achieve a balanced distribution of computational load among processors.
The computational load of a processor is defined as the total number of active cells on that processor and level.

5. Code Tests

5.1. One-Dimensional Test

Relativistic Riemann problems have analytical solutions (Pons et al. 2000), and thus they are ideal for testing SRHD codes. In the following discussion, subscripts "L" and "R" refer to the left and right initial states, respectively. The initial discontinuity is always at $x = 0.5$. We will report the error between numerical solutions and analytical solutions using the $L_1$ norm defined as $L_1 = \sum |u_i - u(x^i)|\Delta x^i$, where $u_i$ is the numerical solution, $u(x^i)$ is the analytical solution, and $\Delta x^i$ is the cell width.

5.1.1. Relativistic Blast Wave I

This test and the following one are fairly standard, and all modern SRHD codes can match the analytical solution quite well (see Marti & Müller 2003 for a summary of different codes’ performance on those two tests).

The initial left and right states for this problem are $p_L = 13.33$, $\rho_L = 10.0$, $v_L = 0.0$ and $p_R = 10^{-6}$, $\rho_R = 1.0$, $v_R = 0.0$, respectively. The gas is assumed to be ideal with an adiabatic index $\Gamma = 5/3$. This test is only mildly relativistic, with a postshock velocity 0.72 and shock velocity 0.83. The results using four hydro solvers are shown in Figure 1. The CFL number used is 0.5. The $L_1$ errors are shown in Figure 2. We can see that for this problem PPM and WENO have smaller $L_1$ errors than PLM.

5.1.2. Relativistic Blast Wave II

The initial left and right states for this problem are $p_L = 1000.0$, $\rho_L = 1.0$, $v_L = 0.0$ and $p_R = 10^{-2}$, $\rho_R = 1.0$, $v_R = 0.0$, respectively. The gas is assumed to be ideal with an adiabatic index $\Gamma = 5/3$. This test is more relativistic than the previous one. While the wave structure is the same, the thermodynamically relativistic initial left state gives rise to a relativistic shock propagating at a Lorentz factor $W \approx 6$ and a very thin dense shell behind the shock with width $\approx 0.01056$ at $t = 0.4$. The CFL number used is 0.5. The results using four hydro algorithms are shown in Figure 3. The $L_1$ errors are shown in Figure 4. We can see that for this problem PPM and WENO have smaller $L_1$ errors than PLM.

---

Fig. 1.—Relativistic blast wave I at $t = 0.4$ with uniform resolution $N = 400$ for (a) HLL-PLM, (b) HLL-PPM, (c) HLL-CENO, and (d) F-WENO5. Numerical profiles of density (squares), pressure (crosses), and velocity (diamonds) are shown, as well as the analytical solution (solid lines). The CFL number used is 0.5.

Fig. 2.—$L_1$ errors in rest mass density for the relativistic blast wave I problem for six different uniform grid resolutions. The symbols denote HLL-PLM (plus signs), HLL-PPM (diamonds), HLL-CENO (squares), and F-WENO5 (crosses). The dashed line indicates first order of global convergence.
and CENO. This is due to their ability to better resolve the thin shell.

5.1.3. Planar Jet Propagation

The initial left and right states for this test are $p_L = 1.0$, $p_L = 1.0$, $v_L = 0.0$, $v_L = 0.0$, $v_L = 0.9$, and $v_L = 0.9$, respectively. The gas is assumed to be ideal with an adiabatic index $\gamma = 4/3$. This test mimics the interaction of a planar jet head with the ambient medium. The decay of the initial discontinuity gives rise to a strong reverse shock propagating to the left, a forward shock propagating to the right, and a contact discontinuity in-between. The results are shown in Figure 5. The CFL number is 0.5. The $L_1$ errors are shown in Figure 6. As can be seen in Figures 5 and 6, for this problem PPM and WENO behave better than PLM and CENO: there is almost no oscillation behind the reverse shock, and they capture the contact discontinuity with fewer cells. PPM especially captures the contact discontinuity with only four cells and has the smallest $L_1$ error.

5.1.4. Blast Wave with Transverse Velocity I

For this and the following two problems, we will consider nonzero transverse velocities in the initial states. The initial state is identical to blast wave problem II except for the presence of transverse velocities. Those problems were first discussed analytically by Pons et al. (2000). Since then various groups have shown that when transverse velocities are nonzero, in some cases those problems become very difficult to solve numerically unless very high spatial resolution is used (Mignone et al. 2005; Zhang & MacFadyen 2006; Morsony et al. 2006). In realistic astrophysical phenomena transverse velocities are usually very important (see, e.g., Aloy & Rezzolla 2006), and thus solving those problems accurately is of great importance.

As an easy first case, we will consider nonzero transverse velocity only in the low-pressure region. The initial left and right states are $p_L = 1000.0$, $p_L = 1.0$, $v_{xL} = 0.0$, $v_{yL} = 0.0$, and $p_R = 10^{-2}$, $p_R = 1.0$, $v_{xR} = 0.0$, $v_{yR} = 0.99$, respectively. The gas is assumed to be ideal, with an adiabatic index $\gamma = 5/3$. The results are shown in Figure 7. The CFL number is 0.5. The $L_1$ errors are shown in Figure 8. We can see that all four hydro algorithms behaves similarly well, except that PLM and CENO show some small oscillations around the contact discontinuity.

5.1.5. Blast Wave with Transverse Velocity II

Next, we consider nonzero transverse velocity in the high-pressure region. In this case, the problem becomes more difficult to solve numerically (Mignone et al. 2005; Zhang & MacFadyen 2006). The initial left and right states for this problem are $p_L = 1000.0$, $p_L = 1.0$, $v_{xL} = 0.0$, $v_{yL} = 0.9$, and $p_R = 10^{-2}$, $p_R = 1.0$, 

![Fig. 3.—Relativistic blast wave II at $t = 0.4$ with uniform resolution $N = 400$ for (a) HLL-PLM, (b) HLL-PPM, (c) HLL-CENO, and (d) F-WENO5. Numerical profiles of density (squares), pressure (crosses), and velocity (diamonds) are shown as well as the analytical solution (solid lines). The CFL number used is 0.5.](image1)

![Fig. 4.—$L_1$ errors in rest mass density for the relativistic blast wave II problem for six different uniform grid resolutions. The symbols denote HLL-PLM (plus signs), HLL-PPM (diamonds), HLL-CENO (squares), and F-WENO5 (crosses). The dashed line indicates first order of global convergence.](image2)
\[ v_{xR} = 0.0, \quad v_{yR} = 0.0, \] respectively. The gas is assumed to be ideal, with an adiabatic index \( \Gamma = 5/3 \). The high-pressure region is connected to the intermediate state by a rarefaction wave. Since the initial normal velocity in the high-pressure region is zero, the slope of the adiabat increases rapidly with transverse velocity, and thus a large initial transverse velocity will lead to a small intermediate pressure and a small mass flux.

The results using a uniform grid and two AMR runs are shown in Figure 9. The hydro solver used for this figure is HLL-PLM. The CFL number is 0.4. The \( L_1 \) errors are shown in Figure 10. It can be seen that for the run with 400 uniform grid cells, the numerical solution is inadequate, as previously found by Mignone et al. (2005). This is mainly due to the poor capture of the contact discontinuity. We have tried to run this problem with various algorithms but only obtained accurate solutions by dramatically increasing the resolution.

5.1.6. Blast Wave with Transverse Velocity III

Now we introduce transverse velocity in both regions. The initial left and right states for this problem are
\[ p_L = 1000.0, \quad \rho_L = 0.0; \quad v_{xL} = 0.0, \quad v_{yL} = 0.9; \quad p_R = 10^{-2}, \quad \rho_R = 1.0, \quad v_{xR} = 0.0, \quad v_{yR} = 0.9, \] respectively. The gas is assumed to be ideal with an adiabatic index \( \Gamma = 5/3 \). This problem is more difficult than the previous one due to the formation of an extremely thin shell between the rarefaction wave tail and the contact discontinuity (Zhang & MacFadyen 2006).

The results with a uniform grid and two AMR runs are shown in Figure 11. The hydro solver used for this run is HLL-PLM. The CFL number used is 0.5. The \( L_1 \) errors are shown in Figure 12.

Table 1 shows the equivalent resolution and the actual number of cells used for this and the previous tests. It can be seen for the highest-resolution calculation our code uses about 400 times fewer grid cells than the corresponding uniform grid calculation. Thus, AMR allows us to achieve very high resolution while significantly reducing the computational cost.

5.1.7. Jet-Cocoon Interaction

For this test we set up a 1D Riemann problem that mimics the interaction of a jet with an overpressured cocoon. The initial left and right states for this problem are
\[ p_L = 0.00017, \quad \rho_L = 0.01, \quad v_{xL} = 0, \quad v_{yL} = 0.99; \quad p_R = 0.017, \quad \rho_R = 0.1, \quad v_{xR} = 0, \quad v_{yR} = 0, \] respectively. The gas is assumed to be ideal with \( \Gamma = 5/3 \). Those values mimic the conditions of the jet-cocoon boundary in model C2 of Marti et al. (1997). The results using four different Riemann solvers with PLM on uniform grid are shown in Figure 13. It can be seen that solutions using HLL, LLF, and direct flux reconstruction have large positive fluctuations in the normal velocity at the rarefaction wave. It is interesting to note that only

---

**Fig. 5.**—Planar jet problem at \( t = 0.4 \) with uniform resolution \( N = 400 \) for (a) HLL-PLM, (b) HLL-PPM, (c) HLL-CENO, and (d) F-WENO5. Numerical profiles of density (squares), pressure (crosses), and velocity (diamonds) are shown, as well as the analytical solution (solid lines). The CFL number used is 0.5.

**Fig. 6.**—\( L_1 \) errors in rest mass density for the planar jet problem for six different uniform grid resolutions. The symbols denote HLL-PLM (plus signs), HLL-PPM (diamonds), HLL-CENO (squares), and F-WENO5 (crosses). The dashed line indicates first order of global convergence.
HLLC does not suffer from this shortcoming, which is probably due to the ability of HLLC to resolve the contact discontinuity compared to the other Riemann solvers in the code. If those fluctuations also happen in higher dimensional jet simulation, then one would expect that the normal velocity fluctuation seen in this test would lead to an artificially extended cocoon.

In Figure 14 the result of using HLL-PLM and HLLC-PLM with AMR is shown. It can been seen that the fluctuation in the HLL scheme becomes smaller with higher resolution. Figure 15 shows the $L_1$ error for those two schemes with different levels of refinement.

5.2. Two-Dimensional Test: Shock Tube

To test the code in higher dimensions, we first study the 2D shock tube problem suggested by Del Zanna & Bucciantini (2002) and later also used by various groups (see, e.g., Zhang & MacFadyen 2006; Mignone & Bodo 2005; Lucas-Serrano et al. 2004). This test is done in a 2D Cartesian box divided into four equal-area-constant states:

\[
\begin{align*}
(p, v_x, v_y, p)_{\text{NE}} &= (0.1, 0, 0, 0.01), \\
(p, v_x, v_y, p)_{\text{NW}} &= (0.1, 0, 0.99, 1), \\
(p, v_x, v_y, p)_{\text{SW}} &= (0.5, 0, 0, 1), \\
(p, v_x, v_y, p)_{\text{SE}} &= (0.1, 1, 0, 0.99, 1),
\end{align*}
\]

where NE means northeast corner, and so on. The grid is uniform $400 \times 400$. The gas is assumed to be ideal with an adiabatic index $\Gamma = 5/3$. We use outflow boundary conditions in all four directions, and the CFL number is 0.5.

The results are shown in Figure 16 for four schemes. This problem does not have analytical solutions to compare with, but comparing our result with other groups’ result shows good agreement. The crosses in the lower left corners of (a) HLL-PLM and (d) F-WENO is a numerical artifact due to the inability to maintain a contact discontinuity perfectly, which are absent in the results using the HLLC solver (c). This agreed with the result of Mignone & Bodo (2005) that the HLLC solver behaves better in this problem than other Riemann solvers because of its ability of resolve contact discontinuities.

5.3. Three-Dimensional Tests

5.3.1. Relativistic Spherical Shock Reflection

We first study a test problem with uniform grid in Cartesian coordinates that has been used by several groups to test the symmetric properties of a 3D SRHD code (Aloy et al. 1999; Mignone et al. 2005). The initial setup of this test consists of a cold spherical
inflow with initially constant density \( \rho_0 \) and constant velocity \( v_1 \) colliding at the box center. This problem is run using 3D Cartesian coordinates, so it allows one to evaluate the symmetry properties of the code (Marti et al. 1997; Aloy et al. 1999; Mignone et al. 2005). When gas collides at the center, a reflection shock forms. Behind the shock, the kinetic energy will be converted completely into internal energy. Thus, the downstream velocity \( v_2 \) is zero and the specific internal energy is given by the upstream specific kinetic energy,

\[
\epsilon_2 = W_1 - 1. \tag{26}
\]

Using the shock jump condition, the compression ratio \( \rho_2/\rho_1 \) and the shock velocity can be found to be

\[
\frac{\rho_2}{\rho_1} = \frac{\Gamma + 1}{\Gamma - 1} + \frac{\Gamma}{\Gamma - 1} \epsilon_2 \tag{27}
\]

\[
V_s = \frac{(\Gamma - 1)W_1 |v_1|}{W_1 + 1}. \tag{28}
\]

In the unshocked region \( r > V_s t \), the gas flow will develop a self-similar density distribution,

\[
\rho_1 = \left(1 + \frac{v_1 t}{r}\right)^2 \rho_0. \tag{29}
\]

The initial states are \( p_1 = 7 \times 10^{-6}, \rho_0 = 1, v_1 = -0.9 \). We chose a small value for pressure because a grid-based code cannot handle zero pressure. A CFL number of 0.1 is used for this problem, as in other groups (Aloy et al. 1999). We chose to use LLF-PLM for this problem because this turns out to be the most stable solver for this problem. Figure 17 shows the 1D cut through the axis and diagonal direction, and Figure 18 shows a contour through \( z = 0.5 \) plane, both at \( t = 0.4 \). It can be seen from those plots that our code keeps the original spherical symmetry quite well. Since in a Cartesian box the simple outflow boundary condition is inconsistent with the initial spherical inflow setup, we evolve this problem only to \( t = 0.4 \), at which point all the mass in the original box has just entered the shocked region (see Fig. 17). After that time, the evolution would be affected by the unphysical boundary condition.

5.3.2. Relativistic Blast Wave I

In this test, we study a spherical blast wave in 3D Cartesian coordinates. There is no analytical solution for this problem. Thus,
for the sake of comparison, we set up the same problem as other groups (Del Zanna & Bucciantini 2002; Zhang & MacFadyen 2006). The center of the blast wave source is located at the corner (0, 0, 0) of the box. The initial conditions are

\[(\rho, v_r, p) = \begin{cases} (1, 0, 1000), & r \leq 0.4, \\ (1, 0, 1), & r > 0.4, \end{cases}\]

where \( r \) is the distance to the center (0, 0, 0).

An ideal gas with an adiabatic index of \( \Gamma = 5/3 \) is assumed. The left boundaries at the \( x, y, \) and \( z \)-directions are reflecting, while others are outflow. We use a top grid of 128\(^3\) zones with two more levels of refinement and a refinement factor of 2 for this calculation (equivalent resolution 512\(^3\)). F-PLM is used for the result shown, and the CFL number used is 0.1.

The results are given in Figure 19, which shows the cut along \( x \)-axis and the diagonal direction. For comparison, we run a high-resolution 1D simulation using spherical coordinates. The 3D run in Cartesian coordinates agrees with the 1D high-resolution run. Furthermore, it can be seen that the spherical symmetry of the initial condition is preserved rather well in the 3D Cartesian case.

### 5.3.3. Relativistic Blast Wave II

Finally, we study another blast wave problem for which the center of the blast wave source is located at the box center. This problem also does not have analytical solution, but it has been studied by Hughes et al. (2002) so our result can be compared to theirs. The initial conditions are

\[(\rho, v_r, p) = \begin{cases} (1, 0, 10^4), & r \leq 0.05, \\ (0.1, 0, 10), & r > 0.05. \end{cases}\]

### Table 1

| Equivalent Resolution | BT II | BT III |
|-----------------------|-------|--------|
| 400                   | 400   | 400    |
| 800                   | 448   | 421    |
| 1600                  | 455   | 442    |
| 3200                  | 470   | 468    |
| 6400                  | 501   | 473    |
| 12800                 | 518   | 476    |
| 25600                 | 562   | 520    |
An ideal gas EOS with an adiabatic index of $\Gamma = 4/3$ is used. We stop the run at $t = 0.12$, roughly the same ending time as Hughes et al. (2002). A top grid of $64^3$ zones with four levels of refinement and a refinement factor of 2 is used (equivalent resolution $1024^3$). Thus, our resolution is roughly 1.5 times that of Hughes et al. (2002). We used HLL-PLM and a CFL number of 0.5 for this calculation.

Figure 20 plots the numerical solution for all cells centered on the highest level in the 2D slice at $y = 0.5$ at $t = 0.12$ as a function of radius from the center $(0.5, 0.5, 0)$. The position and amplitude of the high-density shell agrees with the calculation of Hughes et al. (2002). And it can be seen that the spherical symmetry is preserved rather well in our code.

6. ASTROPHYSICAL APPLICATION I: RELATIVISTIC SUPERSONIC JET PROPAGATION

Having validated our code using various test problems, we can apply it to astrophysical problems. We will study two typical astrophysical relativistic flow problems in this work, AGN jets and GRB jets. Both topics have been studied extensively with 2D simulations before, but very few 3D calculations have been done in both cases. Consequently, we will focus on 3D simulations here.

In this section, we study a relativistic supersonic jet in three dimensions. We set up the problem using the same parameters as model C2 of Marti et al. (1997). This model has also been studied in two dimensions by Zhang & MacFadyen (2006) and in three dimensions by Aloy et al. (1999). The jet parameters are $\rho_b = 0.01, v_b = 0.99,$ and $r_b = 0.02$. The jet has a classical Mach number $M_b = v_b/c_s = 6$, so the pressure is $p_b = 0.000170305$. The parameters for the medium are $\rho_m = 1.0, v_m = 0,$ and $p_m = p_b$. The EOS is assumed to be ideal with $\Gamma = 5/3$. The jet is injected from the low-$z$ boundary centered at $(0.5, 0.5, 0)$ with radius $r_b$. Outflow boundary conditions are used at the other part of the boundary.

Figure 21 shows the result at $t = 60r_b/c$ for the three runs: HLL-PPM with three levels of refinement (HLL-PPM-I3), HLL-PLM with three levels of refinement (HLL-PLM-I3), and HLL-PLM with four levels of refinement (HLL-PLM-I4). The top grid resolution is $64^3$ zones. Thus, the first two runs have an equivalent resolution of $512^3$ zones, while the last one has $1024^3$ zones. For the first two, turbulence in the cocoon is not fully developed, so the cocoon is still symmetric even in 3D. The HLL-PPM-I3 run has a slightly more turbulent cocoon due to the higher spatial reconstruction order of PPM. Thus, the HLL-PPM-I3 jet propagates slightly more slowly than the HLL-PLM-I3 jet. On the other hand, for the HLL-PLM-I4 jet, the resolution is 20 cells per beam radius, comparable to the resolution used in the 2D study by Marti et al. (1997). The cocoon turbulence is much more developed in this case, as in the 2D case (Marti et al. 1997). Consequently, the HLL-PLM-I4 jet propagates more slowly than the two lower resolution
Furthermore, the HLL-PLM-l4 case does not show axisymmetry because instability quickly develops in the lateral motion and consequently lateral motion also becomes turbulent.

Since we found the jet-cocoon structure differs significantly at the higher resolution run and consequently the jet propagation speed decreases, we conclude that even at 1024^3 effective resolution of our 3D jet simulations the correct solution remains elusive. Moreover, different solvers give disparate answers.

7. ASTROPHYSICAL APPLICATION II:
THREE-DIMENSIONAL SIMULATION
OF COLLAPSR JETS

The idea that “long-soft” gamma-ray bursts are associated with the deaths of massive stars has been supported by various observations recently (see, e.g., Woosley & Bloom (2006) for a recent review). So it is of great interest to calculate how a relativistic jet propagates through and breaks out of a massive star. There have been extensive 2D calculations recently (Zhang et al. 2003, 2004; Mizuta et al. 2006; Morsony et al. 2006). But very few 3D calculations have been reported in the literature so far. Since there is significant turbulent motion and mixing in the jet-cocoon system in 2D calculations, it is important to model this process in 3D.

We use the progenitor HE16A of Woosley & Heger (2006), which is a stripped-down helium core with initial mass 16 $M_\odot$ (see Fig. 22). The radius of the star at onset of collapse is 3.86 $\times$ 10^{10} cm. In our simulation, the mass inside 5 $\times$ 10^{9} cm is removed and replaced by a point mass of 9.3 $M_\odot$. The jet has power $\dot{E} = 3 \times 10^{50}$ ergs s^{-1}, initial Lorentz factor $\Gamma_0 = 5$, and the ratio of its total energy (excluding rest mass energy) to its kinetic energy.
Fig. 16.—Shock tube problem at $t = 0.4$ for (a) HLL-PLM, (b) HLL-PPM, (c) HLLC-PLM, and (d) F-WENO5. Thirty equally spaced contours of the logarithm of density are plotted. The CFL number is 0.3.
is \( f_0 = 40 \). This corresponds to a jet with initial density \( 5.937 \times 10^{-3} \text{ g cm}^{-3} \) and pressure \( 4.156 \times 10^{19} \text{ ergs cm}^{-3} \). The jet is injected parallel to the \( z \)-axis with an initial radius \( 8.73 \times 10^8 \text{ cm} \). Those model parameters are similar to previous 2D calculations (Zhang et al. 2004; Mizuta et al. 2006).

We use a simulation box of \( 3.2 \times 10^{11} \text{ cm} \) in order to follow the propagation of the jet after breakout. We use HLL-PLM as this should be the most stable and reliable scheme to carry out resolution study. The top grid resolution is \( 128 \times 128 \). We run the simulation using four and five refinement levels, which correspond to resolutions of \( 5.6 \) and \( 11 \) cells per jet beam radius, respectively. In order to always have high resolution for the jet material, we designed a color field refinement strategy in addition to the standard refinement criterion designed for discontinuities. More specifically, we use two color fields to keep track of the injected jet material and the stellar material. Those two color fields give us a fraction of jet material at every cell. Then, whenever a cell contains more than 0.1% of jet material, we flag that cell for refinement. This ensures that we also have high resolution when mixing between jet material and star material happens.

The results for those two runs are show in Figure 23. It can been seen that the high-resolution run gives qualitatively different jet dynamics. While in the low-resolution run the jet breaks out of the star successfully, in the high-resolution run the jet head bifurcates at the stellar edge. Interestingly, this behavior has also been seen in 2D calculations (Morsony et al. 2006). But since we...
did not get convergent behavior so far, we cannot conclude at this stage whether this behavior is physical or purely numerical. However, it is safe to conclude that much higher resolutions will be needed to model the jet breakout in 3D.

8. CONCLUSIONS AND DISCUSSIONS

In this paper we have described a new code that solves the special relativistic hydrodynamics equations with both spatially and temporally adaptive mesh refinement. It includes direct flux reconstruction and four approximate Riemann solvers including HLLC, HLL, LLF, and a modified Marquina flux formula. It contains several reconstruction routines: PLM, PPM, third-order CENO, and third- and fifth-order WENO schemes. A modular code structure makes it easy to include more physics modules and new algorithms.

From our test problems and two astrophysical applications, it is clear that relativistic flow problems are more difficult than the Newtonian case. One key reason is that in the presence of ultra-relativistic speed, nonlinear structures such as shocked shells are typically much thinner and thus require the use of very high spatial resolution. SRHD problems also become difficult to solve accurately when significant transverse velocities are present in the problem, as we have shown using several 1D problems. One reason for this difficulty is that in SRHD velocity components are coupled nonlinearly via the Lorentz factor. In studying astrophysical jet problems, we have demonstrated the need of both high resolution achievable
Fig. 21.—Density slice through $y = 0.5$ for 3D relativistic jet at $t = 60\rho_3/c$ for HLL-PPM with three levels of refinement (left), HLL-PLM with three levels of refinement (middle), and HLL-PLM with four levels of refinement (right). The top grid resolution is $64^3$ zones, and a refinement factor of 2 is used. Thus, the left and middle panels have an equivalent resolution of $512^3$ zones, while the right one has $1024^3$ zones. The CFL number is 0.4.

Fig. 22.—Density profile and pressure profile of the HE16A progenitor model adopted in our GRB jet calculation.

Fig. 23.—Density slice through $y = 0.5$ for 3D relativistic GRB jet at $t = 10.99$ s for HLL-PPM with four levels of refinement (left) and five levels of refinement (right). The top grid resolution is $128^3$ zones, and a refinement factor of 2 is used. Thus, the left and middle panels have an equivalent resolution of $2048^3$ zones, while the right one has $4096^3$ zones. The CFL number is 0.4.
only through AMR and careful choice of hydrodynamic algorithms. In addition to validating our AMR code, the most important implications of the calculations we have done is that in relativistic flow simulations, resolution studies are crucial.

We thank Greg Bryan and Michael Norman for sharing enzo with the astrophysical community, without which this study could not have been carried out. We would also like to thank Miguel Aloy for very helpful comment on the draft, and Roger Blandford and Łukasz Stawarz for helpful discussions. Furthermore, we also thank Ralf Kießel for help with optimizing the code and SLAC’s computing servers for maintaining an SGI Altix supercomputer on which the reported calculations were carried out. This work was partially supported by NSF CAREER award AST-0239709. P. W. acknowledges support by the Stanford Graduate Fellowship. W. Z. has been supported by NASA through Chandra Postdoctoral Fellowship PF4-50036 awarded by the Chandra X-Ray Observatory Center, and the DOE Program for Scientific Discovery through Advanced Computing (SciDAC).

REFERENCES

Abel, T., Bryan, G., & Norman, M. 2002, Science, 295, 93
Aloy, M.-A., Ibáñez, J. M., Martí, J. M., & Müller, E. 1999, ApJS, 122, 151
Aloy, M. A., & Rezzolla, L. 2006, ApJ, 640, L115
Anile, A. M. 1989, Relativistic Fluids and Magnetofluids (Cambridge: Cambridge Univ. Press)
Anninos, P., Fragile, P. C., & Salmonson, J. D. 2005, ApJ, 635, 723
Begelman, M. C., Blandford, R. D., & Rees, M. J. 1984, Rev. Mod. Phys., 56, 255
Berger, M. J., & Colella, P. 1989, J. Comput. Phys., 82, 64
Berger, M. J., & Oliker, J. 1984, J. Comput. Phys., 53, 484
Blandford, R. D., McKee, C. F., & Rees, M. J. 1977, Nature, 267, 211
Bryan, G. L., & Norman, M. L. 1997a, preprint (astro-ph/9710187)
———. 1997b, in ASP Conf. Ser. 123, Computational Astrophysics, ed. D. A. Clarke & M. J. West (San Francisco: ASP), 363
Bryan, G. L., Abel, T., & Norman, M. L. 2001, preprint (astro-ph/0112089)
Centrella, J., & Wilson, J. R. 1984, ApJS, 54, 229
Colella, P., & Woodward, P. R. 1984, J. Comput. Phys., 54, 174
Del Zanna, L., & Bucciantini, N. 2002, A&A, 390, 177
Donat, R., Font, J. A., Ibáñez, J. M., & Marquina, A. 1998, J. Comput. Phys., 146, 58
Gehrels, N., et al. 2004, ApJ, 611, 1005
Gingold, R. A., & Monaghan, J. J. 1977, MNRAS, 181, 375
Harten, A., Lax, P. D., & van Leer, B. 1983, SIAM Rev., 25, 35
Hawley, J. F., Wilson, J. R., & Smarr, L. L. 1984a, ApJ, 277, 296
———. 1984b, ApJS, 55, 211
Hughes, P. A., Miller, M. A., & Duncan, G. C. 2002, ApJ, 572, 713
Jiang, G.-S., & Shu, C.-W. 1996, J. Comput. Phys., 126, 202
Kurganov, A., & Tadmor, E. 2000, J. Comput. Phys., 160, 241
Kurganov, A., Noelle, S., & Petrova, G. 2001, SIAM J. Sci. Comput., 23, 707
LeVeque, R. J. 2002, Finite Volume Methods for Hyperbolic Problems (Cambridge: Cambridge Univ. Press)
Lithwick, Y., & Sari, R. 2001, ApJ, 555, 540
Liu, X.-D., & Osher, S. 1998, J. Comput. Phys., 142, 304
Lucas-Serrano, A., Font, J. A., Ibáñez, J. M., & Martí, J. M. 2004, A&A, 428, 703
Lucy, L. B. 1977, AJ, 82, 1013
Martí, J. M., & Müller, E. 1994, J. Fluid Mech., 258, 317
———. 1996, J. Comput. Phys., 123, 1
———. 2003, Living Rev. Relativ., 6, 7
Martí, J. M. A., Mueller, E., Font, J. A., Ibáñez, J. M. A., & Marquina, A. 1997, ApJ, 479, 151
Meliani, Z., Keppens, R., Casse, F., & Giannios, D. 2007, preprint (astro-ph/0701434)
Mignone, A., & Bodo, G. 2005, MNRAS, 364, 126
Mignone, A., PLEWA, T., & Bodo, G. 2005, ApJS, 160, 199
Mirabel, I. F., & Rodriguez, L. F. 1994, Nature, 371, 46
Mizuta, A., Yamasaki, T., Nagataki, S., & Mineshige, S. 2006, ApJ, 651, 960
Morsony, B. J., Lazzati, D., & Begelman, M. C. 2006, preprint (astro-ph/0609254)
Norman, M. L., & Winkler, K.-H. A. 1986, in NATO Advanced Research Workshop on Astrophysical Radiation Hydrodynamics, ed. K.-H. A. Winkler & M. L. Norman (Dordrecht: Kluwer), 449
O’Shea, B. W., Bryan, G., Bordner, J., Norman, M. L., Abel, T., Harkness, R., & Kritsuk, A. 2004, In Adaptive Mesh Refinement—Theory and Applications, ed. T. Plewa, T. Linde & V. G. Weirs (Berlin: Springer), in press
Quirk, J. 1994, Int. J. Numer. Methods Fluids, 18, 555
Pons, J. A., Martí, J. M., & Müller, E. 2000, J. Fluid Mech., 422, 125
Schneider, V., Katscher, U., Rischke, D. H., Waldausser, B., Maruhn, J. A., & Munz, C. D. 1993, J. Comput. Phys., 105, 92
Shu, C.-W., & Osher, S. 1988, J. Comput. Phys., 77, 439
Tingay, S. J., et al. 1995, Nature, 374, 141
Toro, E. F., Spruce, M., & Speares, W. 1994, Shock Waves, 4, 25
Van Leer, B. 1977, J. Comput. Phys., 23, 276
———. 1979, J. Comput. Phys., 32, 101
Wilson, J. R. 1972, ApJ, 173, 431
Woosley, S. E., & Bloom, J. S. 2006, ARA&A, 44, 507
Woosley, S. E., & Heger, A. 2006, ApJ, 637, 914
Zhang, W., & MacFadyen, A. I. 2003, ApJ, 586, 356