Inverse Lax-Wendroff Procedure for Numerical Boundary Conditions of Hyperbolic Equations

Chi-Wang Shu

Division of Applied Mathematics
Brown University

Joint work with Sirui Tan, Francois Vilar, Tingting Li, Ling Huang and Mengping Zhang
• Introduction

• Steady state Hamilton-Jacobi equations

• Time dependent conservation laws

• Compressible inviscid flows involving complex moving geometries

• Conclusions and future work
Introduction

For finite difference schemes approximating PDEs, there are two major difficulties associated with numerical boundary conditions:

- High order finite difference schemes involve a wide stencil, hence there are several points near the boundary (either as ghost points outside the computational domain or as the first few points inside the computational domain near the boundary) which need different treatment.
For example, if we have the following scheme

\[ u_j^{n+1} = au_{j-2}^n + bu_{j-1}^n + cu_j^n + du_{j+1}^n \]

with suitably chosen constants \(a, b, c \) and \(d\) (which depend on \( \lambda = \frac{\Delta t}{\Delta x} \)), approximating the PDE

\[ u_t + u_x = 0, \]

\[ u(x, 0) = f(x), \quad u(0, t) = g(t) \]

to third order accuracy, then either a ghost point \( u_{-1}^n \) is needed, or the scheme cannot be used to compute \( u_1^{n+1} \).
The boundary of the computational domain may not coincide with grid points.

For example, in 1D, we may have the physical boundary \( x = 0 \) located anywhere between two grid points. While this seems artificial, it is unavoidable for a moving boundary computed on a fixed grid.

This difficulty is more profound in 2D (complicated geometry computed on Cartesian meshes).

One of the major difficulties is the small cell near the boundary and the resulting small time step required for stability.
Previous work on numerical boundary conditions:

- $h$-box method of Berger, Helzel and LeVeque (SINUM 2003): suitable flux computation based on cells of size $h$. This method can overcome the difficulty of small time step for stability, but is somewhat complicated in 2D and for high order accuracy.

- Reflecting or symmetry boundary conditions for ghost points: suitable for solid walls or symmetry lines which are straight lines but lead to large errors for curved walls not aligned with meshes.
• Extrapolation to obtain ghost point values (Kreiss et al SINUM 2002, 2004; SISC 2006; Sjögreen and Petersson CiCP 2007). A GKS stability analysis must be performed to assess its stability. Second order is fine but higher order is more complicated to analyze. It is not stable if the physical boundary is too close to a grid point.

• Converting spatial derivative near the boundary to temporal derivatives (Goldberg and Tadmor, Math Comp 1978, 1981 for one-dimensional linear hyperbolic initial-boundary value problems).

• Compatible boundary conditions for boundaries and interfaces (Henshaw, Kreiss and Reyna, Computers & Fluids 1994; Henshaw, SISC 2006; Henshaw and Chand, JCP 2009).
Review on the traditional Lax-Wendroff procedure for solving, e.g.

\[ u_t + u_x = 0 \]

- Taylor expansion in time

\[ u_j^{n+1} = u_j + (u_t)_j \Delta t + \frac{1}{2}(u_{tt})_j \Delta t^2 + \ldots \]

- Replace the time derivatives by spatial derivatives by repeatedly using the PDE:

\[ (u_t)_j = -(u_x)_j \]

\[ (u_{tt})_j = -((u_x)_t)_j = -((u_t)_x)_j = (u_{xx})_j \]

\[ \ldots \]

- Approximate the spatial derivatives by finite differences of suitable order of accuracy.
We now look at the basic idea of the inverse Lax-Wendroff procedure, by switching the roles of \( x \) and \( t \) in the traditional Lax-Wendroff procedure. Suppose we are solving

\[
    u_t + u_x = 0, \quad u(0, t) = g(t)
\]

and suppose the boundary \( x = 0 \) is of distance \( a \Delta x \) from \( x_1 \) (with a constant \( a \)), the inverse Lax-Wendroff procedure to determine \( u_1 \) is as follows:
- Taylor expansion in space

\[ u_1 = u(0, t) + u_x(0, t)a\Delta x + \frac{1}{2}u_{xx}(0, t)(a\Delta x)^2 + \ldots \]

- Replace the spatial derivatives by time derivatives by repeatedly using the PDE:

\[ u_x = -u_t; \quad u_x(0, t) = -u_t(0, t) = -g'(t) \]

\[ u_{xx} = (-u_t)_x = -(u_x)_t = u_{tt}; \]

\[ u_{xx}(0, t) = u_{tt}(0, t) = g''(t) \]

\[ \ldots \]

- Compute \( g'(t), g''(t) \), etc. either analytically or by finite difference.
We are interested in the steady state solution of the Hamilton-Jacobi equation

\[ H(\phi_x, \phi_y) = f(x, y) \]  

(1)

together with suitable boundary conditions.

We can use Runge-Kutta or other methods to march in time for the time dependent PDE

\[ \phi_t + H(\phi_x, \phi_y) = f(x, y) \]  

(2)

until steady state is reached, but that is rather slow.
One class of effective numerical methods is the fast sweeping method (Boué and Dupuis, SINUM 1999; Zhao, Math Comp 2005). For high order finite difference fast sweeping methods (Zhang, Zhao and Qian, JSC 2006), the first few points near an inflow boundary are usually prescribed to be the exact solution. This is not practical for problems with unknown exact solutions.

To fix the ideas, let us assume that the left boundary

$$\Gamma = \{(x, y) : x = 0, \ 0 \leq y \leq 1\}$$

of the computational domain $[0, 1]^2$ is the inflow boundary, on which the solution is given as

$$\phi(0, y) = g(y), \quad 0 \leq y \leq 1.$$
We would like to obtain a high order approximation to the solution value
\[
\phi_{i,j} \approx \phi(x_i, y_j) \text{ for } i = 1, 2 \text{ and a fixed } j, \text{ which corresponds to a point }
(x_i, y_j) \text{ near the inflow boundary which cannot be computed by the high }
\text{order WENO scheme. A simple Taylor expansion gives, for } i = 1, 2,
\[
\phi(x_i, y_j) = \phi(0, y_j) + i h \phi_x(0, y_j) + \frac{(i h)^2}{2} \phi_{xx}(0, y_j) + O(h^3)
\]
hence our desired approximation for the third order WENO scheme is
\[
\phi_{i,j} = \phi(0, y_j) + i h \phi_x(0, y_j) + \frac{(i h)^2}{2} \phi_{xx}(0, y_j).
\]
We already have \( \phi(0, y_j) = g(y_j) \). The PDE (1), evaluated at the point \((0, y_j)\), becomes

\[
H(\phi_x(0, y_j), g'(y_j)) = f(0, y_j)
\]

(4)
in which the only unknown quantity is \( \phi_x(0, y_j) \). Solving this (usually nonlinear) equation should give us \( \phi_x(0, y_j) \). There might be more than one root, in which case we should choose the root so that

\[
\partial_u H(\phi_x(0, y_j), g'(y_j)) > 0
\]

(5)
where \( \partial_u \) refers to the partial derivative with respect to the first argument in \( H(u, v) \). The condition (5) guarantees that the boundary \( \Gamma \) in (3) is an inflow boundary. If the condition (5) still cannot pin down a root, then we would choose the root which is closest to the value from the first order fast sweeping solution at the same grid point.
To obtain \( \phi_{xx}(0, y_j) \), we first take the derivative with respect to \( y \) on the original PDE (1), and then evaluate it at the the point \((0, y_j)\), which yields

\[
\partial_u H(\phi_x(0, y_j), g'(y_j)) \phi_{xy}(0, y_j) + \partial_v H(\phi_x(0, y_j), g'(y_j)) g''(y_j) = f_y(0, y_j),
\]

where \( \partial_u \) and \( \partial_v \) refer to the partial derivatives with respect to the first and second arguments in \( H(u, v) \), respectively. In this equation the only unknown quantity is \( \phi_{xy}(0, y_j) \), hence we obtain easily its value, thanks to (5).
We then take the derivative with respect to $x$ on the original PDE (1), and evaluate it at the point $(0, y_j)$ to obtain

$$\partial_u H(\phi_x(0, y_j), g'(y_j)) \phi_{xx}(0, y_j) + \partial_v H(\phi_x(0, y_j), g'(y_j)) \phi_{xy}(0, y_j) = f_x(0, y_j)$$

This time, the only unknown quantity is $\phi_{xx}(0, y_j)$, which we can obtain readily from this equality.

It is clear that this procedure can be carried out to any desired order of accuracy. Also, the inflow boundary $\Gamma$ in (3) can be any piece of a smooth curve and does not need to be aligned with the mesh points: we only need to change the $x$ and $y$ partial derivatives to normal and tangential derivatives with respect to $\Gamma$. However, for this approach to work, $\Gamma$ cannot consist of a single point.
Example 1. We solve the Eikonal equation with $f(x, y) = 1$. The computational domain is $[-1, 1]^2$, and the inflow boundary $\Gamma$ is the unit circle of center $(0,0)$ and radius 0.5, that is

$$\Gamma = \left\{ (x, y) : x^2 + y^2 = \frac{1}{4} \right\}.$$

The boundary condition $\phi(x, y) = 0$ is prescribed on $\Gamma$. The exact solution for this problem is the distance function to the circle $\Gamma$. This exact solution has a singularity at the center of the circle to which the characteristics converge, hence we exclude the box $[-0.15, 0.15]^2$ when measuring the errors. Notice that for this example, the domain boundary $\Gamma$ is not aligned with the Cartesian mesh. We again use third order WENO scheme with the fast sweeping method.
Table 1: Example 1. Lax-Wendroff type procedure for the inflow boundary. $N$ is the number of mesh points in each direction. The errors are measured in the computational domain but outside the box $[-0.15, 0.15]^2$.

<table>
<thead>
<tr>
<th>N</th>
<th>$L^1$ error</th>
<th>order</th>
<th>$L^\infty$</th>
<th>order</th>
<th>iteration number</th>
</tr>
</thead>
<tbody>
<tr>
<td>80</td>
<td>0.573E-05</td>
<td></td>
<td>0.129E-03</td>
<td></td>
<td>25</td>
</tr>
<tr>
<td>160</td>
<td>0.122E-05</td>
<td>2.23</td>
<td>0.407E-05</td>
<td>4.98</td>
<td>32</td>
</tr>
<tr>
<td>320</td>
<td>0.191E-06</td>
<td>2.68</td>
<td>0.122E-05</td>
<td>1.74</td>
<td>46</td>
</tr>
<tr>
<td>640</td>
<td>0.246E-07</td>
<td>2.95</td>
<td>0.161E-06</td>
<td>2.92</td>
<td>62</td>
</tr>
</tbody>
</table>
References:


The same idea we mentioned in the introduction can be used to strongly hyperbolic conservation laws for \( U = U(x, y, t) \in \mathbb{R}^2 \)

\[
\begin{align*}
\begin{cases}
U_t + F(U)_x + G(U)_y &= 0 \quad (x, y) \in \Omega, \quad t > 0, \\
U(x, y, 0) &= U_0(x, y) \quad (x, y) \in \bar{\Omega},
\end{cases}
\end{align*}
\]

(7)
on a bounded domain \( \Omega \) with appropriate boundary conditions prescribed on \( \partial \Omega \) at time \( t \). We assume \( \Omega \) is covered by a uniform Cartesian mesh \( \Omega_h = \{(x_i, y_j) : 0 \leq i \leq N_x, 0 \leq j \leq N_y\} \) with mesh size \( \Delta x = \Delta y \).
One difficulty of this procedure, especially for nonlinear systems in multiple-dimensions, is that the algebra becomes very heavy for higher order derivatives.

In (Tan, Wang, Shu and Ning, JCP 2012), a simplified version of this inverse Lax-Wendroff procedure is adopted. This procedure is used only to compute the first spatial derivative $u_x$, subsequent derivatives $u_{xx}$ etc. are obtained by standard extrapolation with suitable order of accuracy.

The computational examples in (Tan, Wang, Shu and Ning, JCP 2012) are for physical boundaries aligned with the mesh points. For such cases and for fifth order WENO schemes, this simplified inverse Lax-Wendroff procedure works very well with stable results in very demanding detonation problems.
In (Vilar and Shu, *M2AN* 2015), we perform a rigorous stability analysis using the GKS (Gustafsson, Kreiss and Sundström) theory, using the class of central compact schemes in (Liu, Zhang, Zhang and Shu, JCP 2013) as examples. This analysis is also performed for upwind-biased finite difference schemes (prototypes of WENO schemes with linear weights) in (Li, Shu and Zhang, JCAM submitted).

This analysis gives explicit guidance on how many terms of $u_x$, $u_{xx}$, ... are required to be treated by the inverse Lax-Wendroff procedure in order to maintain stability (for the fully discrete case, under the same CFL number as in the periodic case) for arbitrary location of the boundary in relation to the nearest grid point.
Two different techniques, one based on normal mode GKS analysis and the other based on eigenstructure analysis of amplification matrices, are performed, and are shown to lead to identical conclusions regarding stability when both work.
Table 2: Minimum number of leading terms with ILW procedure required by the different RK3-CCS-explicit schemes to remain stable under the same CFL as that for periodic boundary conditions.

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Required leading terms by ILW</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCS-E4</td>
<td>3</td>
</tr>
<tr>
<td>CCS-E6</td>
<td>4</td>
</tr>
<tr>
<td>CCS-E8</td>
<td>5</td>
</tr>
<tr>
<td>CCS-E10</td>
<td>5</td>
</tr>
</tbody>
</table>
Table 3: Minimum number of leading terms with ILW procedure required by the different RK3-CCS-tridiagonal schemes to remain stable under the same CFL as that for periodic boundary conditions.

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Required leading terms</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCS-T4</td>
<td>3</td>
</tr>
<tr>
<td>CCS-T6</td>
<td>3</td>
</tr>
<tr>
<td>CCS-T8</td>
<td>5</td>
</tr>
<tr>
<td>CCS-T10</td>
<td>8</td>
</tr>
<tr>
<td>CCS-T12</td>
<td>9</td>
</tr>
</tbody>
</table>
Table 4: Minimum number of leading terms with ILW procedure required by the different upwind-biased schemes with RK3 time discretization to remain stable under the same CFL as that for periodic boundary conditions.
At the outflow boundary, extrapolation of appropriate order is used. Either a regular or a WENO type extrapolation is appropriate depending on whether the outflow solution is smooth or contains shocks.

For the outflow boundary condition, we can show that the scheme with the extrapolation is stable for all order $s$.

We remark that the time step restriction of solving the system of ODEs with our boundary treatment is not more severe than the pure initial value problem. The standard CFL conditions determined by the interior schemes are used in the numerical examples.
Example 2. We test the Burgers equation

\[
\begin{cases}
    u_t + \left( \frac{1}{2} u^2 \right)_x = 0 & x \in (-1, 1), \quad t > 0, \\
    u(x, 0) = 0.25 + 0.5 \sin(\pi x) & x \in [-1, 1], \\
    u(-1, t) = g(t) & t > 0.
\end{cases}
\] (8)

Here \(g(t) = w(-1, t)\), where \(w(x, t)\) is the exact solution of the initial value problem on \((-1, 1)\) with periodic boundary conditions. For all \(t\), the left boundary \(x = -1\) is an inflow boundary and the right boundary \(x = 1\) is an outflow boundary.
Table 5: Errors of the Burgers equation (8). $\Delta x = 2/N$ and $t = 0.3$.

<table>
<thead>
<tr>
<th>$N$</th>
<th>$L^1$ error</th>
<th>order</th>
<th>$L^\infty$ error</th>
<th>order</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>9.11E-05</td>
<td></td>
<td>3.56E-04</td>
<td></td>
</tr>
<tr>
<td>80</td>
<td>3.10E-06</td>
<td>4.88</td>
<td>1.35E-05</td>
<td>4.72</td>
</tr>
<tr>
<td>160</td>
<td>1.31E-07</td>
<td>4.57</td>
<td>6.51E-07</td>
<td>4.38</td>
</tr>
<tr>
<td>320</td>
<td>3.97E-09</td>
<td>5.05</td>
<td>2.68E-08</td>
<td>4.60</td>
</tr>
<tr>
<td>640</td>
<td>1.02E-10</td>
<td>5.29</td>
<td>8.34E-10</td>
<td>5.00</td>
</tr>
<tr>
<td>1280</td>
<td>2.86E-12</td>
<td>5.15</td>
<td>2.62E-11</td>
<td>5.00</td>
</tr>
</tbody>
</table>
Figure 1: Burgers equation (8), $\Delta x = 1/40$. Solid line: exact solution; Symbols: numerical solution.
Example 3. Euler equations, blast wave example. We consider the interaction of two blast waves. The initial data are

\[ U(x, 0) = \begin{cases} 
    U_L & 0 < x < 0.1, \\
    U_M & 0.1 < x < 0.9, \\
    U_R & 0.9 < x < 1, 
\end{cases} \]

where \( \rho_L = \rho_M = \rho_R = 1 \), \( u_L = u_M = u_R = 0 \), \( p_L = 10^3 \), \( p_M = 10^{-2} \), \( p_R = 10^2 \). There are solid wall boundary conditions at both \( x = 0 \) and \( x = 1 \). This problem involves multiple reflections of shocks and rarefactions off the walls. There are also multiple interactions of shocks and rarefactions with each other and with contact discontinuities.
Figure 2: The density profiles of the blast wave problem. Solid lines: reference solution computed by the fifth order WENO scheme with $\Delta x = 1/16000$; Symbols: numerical solutions by our boundary treatment.
Example 4. We test the 2D Burgers equation

\[
\begin{cases}
  u_t + \frac{1}{2} (u^2)_x + \frac{1}{2} (u^2)_y = 0 & (x, y) \in \Omega, \ t > 0, \\
  u(x, y, 0) = 0.75 + 0.5 \sin [\pi (x + y)] & (x, y) \in \bar{\Omega}, \\
  u(x, y, t) = g(x, y, t) & (x, y) \in \Gamma, \ t > 0,
\end{cases}
\]

(9)

where

\[
\begin{align*}
  \Omega &= (-1, 1) \times (-1, 1), \\
  \Gamma &= \{(x, y) : x = -1 \text{ or } y = -1\},
\end{align*}
\]

or

\[
\begin{align*}
  \Omega &= \{(x, y) : x^2 + y^2 < 0.5\}, \\
  \Gamma &= \{(x, y) : x^2 + y^2 = 0.5 \text{ and } x + y \leq 0\}.
\end{align*}
\]
Here $g(x, y, t) = w(x, y, t)$, where $w(x, y, t)$ is the exact solution of the initial value problem on $(-1, 1) \times (-1, 1)$ with periodic boundary conditions. Notice that in the second case the domain boundary is not aligned with the Cartesian meshes.

Table 6: Errors of the 2D Burgers equation (9). $\Delta x = 2/N_x, \Delta y = 2/N_y, t = 0.15.$

<table>
<thead>
<tr>
<th>$N_x = N_y$</th>
<th>on a square</th>
<th>on a disk</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$L^1$ error</td>
<td>$L^\infty$ error</td>
</tr>
<tr>
<td>40</td>
<td>1.55E-04</td>
<td>9.86E-03</td>
</tr>
<tr>
<td>80</td>
<td>1.06E-05</td>
<td>1.80E-03</td>
</tr>
<tr>
<td>160</td>
<td>4.93E-07</td>
<td>2.38E-04</td>
</tr>
<tr>
<td>320</td>
<td>3.47E-08</td>
<td>2.83E-05</td>
</tr>
<tr>
<td>640</td>
<td>2.72E-09</td>
<td>2.85E-06</td>
</tr>
</tbody>
</table>
Figure 3: 2D Burgers equation (9). $\Delta x = \Delta y = 1/40$. Cut along the diagonal. Solid line: exact solution; Symbols: numerical solution.
INVERSE LAX-WENROFF PROCEDURE FOR NUMERICAL BOUNDARY CONDITIONS OF HYPERBOLIC EQUATIONS

(a) on a disk, $t = 0.55$

(b) on a disk, $t = 6$

Figure 4: Continued.
Example 5. We are most interested in applying our method to the solid wall boundary conditions \((u, v) \cdot n = 0\), when the wall is not aligned with the grid and can be curved. Our first example of this kind is the double Mach reflection problem. This problem is initialized by sending a horizontally moving shock into a wedge inclined by a 30° angle. In order to impose the solid wall condition by the reflection technique, people usually solve an equivalent problem that puts the solid wall horizontal and puts the shock 60° angle inclined to the wall. Another way to avoid the trouble of imposing boundary conditions is to use a multidomain WENO method. With the use of our method, we are able to solve the original problem with a uniform mesh in a single domain.
Figure 5: Left: The computational domain (solid line). The dashed line indicates the computational domain used in the traditional finite difference solvers. The square points indicate some of the grid points. Right: Density contour of double Mach reflection. $\Delta x = \Delta y = \frac{1}{320}$.
Figure 6: Density contours of double Mach reflection, 30 contours from 1.731 to 20.92. Zoomed-in near the double Mach stem. The plots in the left column (our computation with the new boundary condition treatment) are rotated and translated for comparison.
Inverse Lax-Wendroff Procedure for Numerical Boundary Conditions of Hyperbolic Equations

\[ \Delta x = \Delta y = \frac{1}{640}, \text{ original problem} \]
\[ \Delta x = \Delta y = \frac{\sqrt{3}}{960}, \text{ equivalent problem} \]

Figure 7: Continued
**Example 6.** This example involves a curved wall which is a circular cylinder of unit radius positioned at the origin on a $x-y$ plane. The problem is initialized by a Mach 3 flow moving toward the cylinder from the left. In order to impose the solid wall boundary condition at the surface of the cylinder by the reflection technique, a particular mapping from the unit square to the physical domain is usually used in traditional finite difference methods. Using our method, we are able to solve this problem directly in the physical domain.
Figure 8: Physical domain of flow past a cylinder. The square points indicate some of the grid points near the cylinder. Illustrative sketch, not to scale.
(a) $\Delta x = \Delta y = \frac{1}{20}$

(b) $\Delta x = \Delta y = \frac{1}{40}$

Figure 9: Pressure contour of flow past a cylinder.

Division of Applied Mathematics, Brown University
Reference:


We extend the high order accurate numerical boundary condition based on finite difference methods to simulations of compressible inviscid flows involving complex moving geometries.

- For problems in such geometries, it is difficult to use body-fitted meshes which conform to the moving geometry.
- Instead, methods based on fixed Cartesian meshes have been successfully developed. For example, the immersed boundary (IB) method introduced by Peskin (JCP 1972) is widely used. One of the challenges of the IB method is the representation of the moving objects which cut through the grid lines in an arbitrary fashion.
• To solve compressible inviscid flows in complex moving geometries, most methods in the literature are based on finite volume schemes. The challenge mainly comes from the so-called “small-cell” problem. Namely, one obtains irregular cut cells near the boundary, which may be orders of magnitude smaller than the regular grid cells, leading to a severe time step restriction.

• In terms of accuracy, most finite volume schemes in the literature are at most second order. In particular, the errors at the boundaries sometimes often fall short of second order.

• Our inverse Lax-Wendroff procedure can be extended to such situations with moving geometries. The only change is to obtain relationships between the temporal and spatial derivatives via the PDE in moving Lagrangian framework.
Example 7. We consider a gas confined between two rigid walls. The right wall is fixed at $x_r = 1.0$ while the left wall is moving. We assume the left wall is positioned at $x_l(t) = 0.5(1 - t)$. The initial conditions are

\[
\begin{align*}
\rho(x, 0) &= 1 + 0.2 \cos [2\pi (x - 0.5)], \\
u(x, 0) &= x - 1, \\
p(x, 0) &= \rho(x, 0)^\gamma,
\end{align*}
\]

such that the initial entropy $s(x, 0) = 1$. As long as the solution stays smooth, we have isentropic flow, i.e., $s(x, t) = 1$. Thus the numerical value of the entropy can be used for the analysis of convergence.
Table 7: Entropy errors and convergence rates of Example 9

<table>
<thead>
<tr>
<th>$h$</th>
<th>$x_l(t) = 0.5(1 - \sin t)$</th>
<th>$L^1$ error</th>
<th>order</th>
<th>$L^\infty$ error</th>
<th>order</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/40</td>
<td>7.26E-07</td>
<td>1.32E-06</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1/80</td>
<td>1.15E-08</td>
<td>5.98</td>
<td>2</td>
<td>2.82E-08</td>
<td>5.55</td>
</tr>
<tr>
<td>1/160</td>
<td>3.43E-10</td>
<td>5.07</td>
<td>6.19E-10</td>
<td>5.51</td>
<td></td>
</tr>
<tr>
<td>1/320</td>
<td>9.90E-12</td>
<td>5.11</td>
<td>2.49E-11</td>
<td>4.64</td>
<td></td>
</tr>
</tbody>
</table>
Example 8. This is a 1D problem involving shocks and rarefaction waves. A piston with width $10h$ is initially centered at $x = -5h$ inside a shock tube. Here $h$ is the mesh size. The piston instantaneously moves with a constant velocity $u_p = 2$ into an initially quiescent fluid with $\rho = 1$ and $p = 5/7$. This problem is equivalent to two independent Riemann problems and thus the exact solution can be obtained. A shock forms ahead of the piston and a rarefaction wave forms in the rear.
Figure 10: Density and pressure profiles of Example 10. The piston is represented by the rectangle. Solid lines: exact solutions; Symbols: numerical solutions with $h = 0.25$. 
Example 9. We now move on to 2D examples. We first test a 2D version of Example 9. A gas is confined in a rectangular region whose boundaries are rigid walls. The top and bottom walls are fixed at \( y = 0 \) and \( y = 1 \) respectively. The right wall is fixed at \( x = 1 \). The left moving wall is positioned at \( x_l(t) = 0.5(1 - \sin t) \). The initial conditions are

\[
\rho(x, y, 0) = 1 + 0.2 \cos [2\pi (x - 0.5)] + 0.1 \cos [2\pi(y - 0.5)] ,
\]

\[
u(x, y, 0) = x - 1 ,
\]

\[
v(x, y, 0) = y(1 - y) \cos(\pi x) ,
\]

\[
p(x, y, 0) = \rho(x, y, 0)^{\gamma} ,
\]

such that the initial entropy \( s(x, y, 0) = 1 \). We use our high order boundary treatment at the left moving wall and the reflection technique at the fixed walls.
Table 8: Entropy errors and convergence rates of Example 11.

<table>
<thead>
<tr>
<th>$h$</th>
<th>$L^1$ error</th>
<th>order</th>
<th>$L^\infty$ error</th>
<th>order</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/80</td>
<td>2.50E-08</td>
<td></td>
<td>3.28E-07</td>
<td></td>
</tr>
<tr>
<td>1/160</td>
<td>1.10E-09</td>
<td>4.50</td>
<td>3.06E-08</td>
<td>3.42</td>
</tr>
<tr>
<td>1/320</td>
<td>9.70E-11</td>
<td>3.50</td>
<td>6.17E-09</td>
<td>2.31</td>
</tr>
<tr>
<td>1/640</td>
<td>9.87E-12</td>
<td>3.30</td>
<td>7.06E-10</td>
<td>3.13</td>
</tr>
</tbody>
</table>
Example 10. Our next example involves 2D flows in complex moving geometries. The computational domain is $[-4, 4] \times [-4, 4]$ with all the boundaries as rigid walls. A rigid cylinder with radius $R = 1$ is initially centered at $(0, 0)$ and starts moving. The center of the cylinder is positioned at $X_c(t)$. We use our high order boundary treatment at the surface of the moving cylinder and the reflection technique at the fixed walls.

In our first case, we take $X_c = (-0.5 \sin t, 0)$ such that the cylinder moves horizontally. In the second case, we take $X_c = (-0.5 \sin t, 0.3t)$ such that the cylinder moves in the 2D space.
INVERSE LAX-WENDROFF PROCEDURE FOR NUMERICAL BOUNDARY CONDITIONS OF HYPERBOLIC EQUATIONS

Figure 11: Density contours of Example 12. $h = 1/40$, $t = 0.4$.

(a) $X_c = (-0.5 \sin t, 0)$

(b) $X_c = (-0.5 \sin t, 0.3t)$
Table 9: Entropy errors and convergence rates of Example 12. $t = 0.4$.

<table>
<thead>
<tr>
<th>$h$</th>
<th>$x_c = (-0.5 \sin t, 0), t = 0.7$</th>
<th>$x_c = (-0.5 \sin t, 0.3t), t = 0.5$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$L^1$ error</td>
<td>order</td>
</tr>
<tr>
<td>1/5</td>
<td>4.11E-03</td>
<td>2.47E-03</td>
</tr>
<tr>
<td>1/10</td>
<td>3.86E-04</td>
<td>3.41</td>
</tr>
<tr>
<td>1/20</td>
<td>1.21E-05</td>
<td>5.00</td>
</tr>
<tr>
<td>1/40</td>
<td>2.43E-07</td>
<td>5.64</td>
</tr>
</tbody>
</table>
**Example 11.** The last example shows that our high order method can also treat a rigid body whose motion is induced by the fluid. We test the so-called cylinder lift-off problem. In this problem, a rigid cylinder initially resting on the floor of a 2D channel is driven and lifted by a strong shock. The computational domain is $[0, 1] \times [0, 0.2]$. A rigid cylinder with radius 0.05 and density 10.77 is initially centered at $(0.15, 0.05)$. A Mach 3 shock starts at $x = 0.08$ moving towards the cylinder. The density and pressure of the resting gas are $\rho = 1.4$ and $p = 1.0$ respectively. The top and bottom of the domain are rigid walls. The left boundary is set to the post-shock state and the right boundary is supersonic outflow.
Table 10: Center of the cylinder of Example 13.

<table>
<thead>
<tr>
<th>$h$</th>
<th>$t = 0.1641$</th>
<th>$t = 0.30085$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$x$-coordinate</td>
<td>$y$-coordinate</td>
</tr>
<tr>
<td>1/160</td>
<td>3.7058E-01</td>
<td>8.1140E-02</td>
</tr>
<tr>
<td>1/320</td>
<td>3.6153E-01</td>
<td>8.3219E-02</td>
</tr>
<tr>
<td>1/640</td>
<td>3.5706E-01</td>
<td>8.3680E-02</td>
</tr>
<tr>
<td>1/1280</td>
<td>3.5539E-01</td>
<td>8.4133E-02</td>
</tr>
<tr>
<td>1/2560</td>
<td>3.5461E-01</td>
<td>8.4258E-02</td>
</tr>
</tbody>
</table>
Figure 12: Pressure contours at $t = 0.1641$. 53 contours from 2 to 28.

(a) $h = 1/640$

(b) $h = 1/1280$

Division of Applied Mathematics, Brown University
Figure 13: Pressure contours at $t = 0.30085$. 53 contours from 2 to 28.
Reference:


Concluding remarks

- We have demonstrated an inverse Lax-Wendroff procedure for boundary treatment, which yields stable discretization with the same CFL number as the inner scheme and allows us to compute problems on arbitrary domains using Cartesian meshes.

- The technique can be applied to inviscid flows with complex moving geometries, yielding stable and high order accurate solutions.

- Future work would involve a generalization of this technique to other schemes such as the discontinuous Galerkin method, and to viscous problems and to problems with deformable structures.
The End

THANK YOU!