A High Resolution Scheme for the MHD Equations

Matthew Bedford

December 9, 2012

Background

To generate a numerical model for the propagation of shock waves in a plasma, it is necessary to solve a system of equations consisting of conservation laws for mass, momentum, and energy, along with Maxwell's equations. Such a system in one dimension can be written in vector form as

$$\frac{\partial \vec{U}}{\partial t} + \frac{\partial \vec{F}(\vec{U})}{\partial x} = 0, \tag{1}$$

where $\vec{F}$ is the flux of the conservative variables $\vec{U}$.

Applying the chain rule, this system can be written as

$$\frac{\partial \vec{U}}{\partial t} + \mathbf{A}(\vec{U})\frac{\partial \vec{U}}{\partial x} = 0, \tag{2}$$

where the matrix $\mathbf{A}(\vec{U}) = \frac{\partial \vec{F}(\vec{U})}{\partial \vec{U}}$ is diagonalizable. Here the vectors depend on the density ρ, velocity $\vec{v} = (u, v, w)$, pressure p, and magnetic field $\vec{B} = (B_x, B_y, B_z)$.

$$\vec{U} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ e \\ B_x \\ B_y \\ B_z \end{pmatrix}, \quad \vec{F}(\vec{U}) = \begin{pmatrix} \rho u \\ \rho u^2 + p_0 - \frac{B_x^2}{4\pi} \\ \rho uv - \frac{B_x B_y}{4\pi} \\ \rho uw - \frac{B_x B_z}{4\pi} \\ (e + p_0)u - \frac{B_x}{4\pi}(\vec{v} \cdot \vec{B}) \\ 0 \\ uB_y - vB_x \\ uB_z - wB_x \end{pmatrix}, \quad \text{and} \quad \vec{u} = \begin{pmatrix} \rho \\ u \\ v \\ w \\ p \\ B_x \\ B_y \\ B_z \end{pmatrix}, \tag{3}$$

where $e = \frac{p}{\gamma - 1} + \frac{\rho(u^2+v^2+w^2)}{2} + \frac{B_x^2+B_y^2+B_z^2}{8\pi}$ is the total energy, $\gamma = \frac{5}{3}$ is the adiabatic index, $p_0 = p + \frac{B^2}{8\pi}$ is the total pressure, and $\vec{u}$ are the primitive variables, which are often more convenient to use.

Since **A** is diagonalizable, there exist real eigenvalues λ_k and a complete set of eigenvectors. Multiplying (2) by the left eigenvector $\vec{l}_k$ yields

$$\vec{l}_k \cdot \left(\frac{\partial \vec{U}}{\partial t} + \lambda_k \frac{\partial \vec{U}}{\partial x} \right) = \vec{l}_k \cdot \left(\frac{d\vec{U}}{dt} \right)_k = 0, \tag{4}$$

where the total time derivative is taken in the characteristic direction $\frac{dx}{dt} = \lambda_k$. If the matrix **A** is constant, then the solution to (4) propagates along straight lines in the x-t plane:

$$x = \lambda_k t + \text{constant}. \tag{5}$$

The eigenvalues are the characteristic velocities of the plasma: $u, u \pm v_A, u \pm v_f$, and $u \pm v_s$. Here v_A is the Alfven velocity, and $v_{f,s}$ are the fast and slow magnetosonic velocities.

Numerical Model

The problem solved here is the time evolution of a one-dimensional MHD system with discontinuous initial conditions: on the spatial interval $(0, 1)$ the primitive variables are given by

$$\vec{u}(x) = \begin{cases} \vec{u}_L & \text{if } x < 0.5 \\ \vec{u}_R & \text{if } x \geq 0.5, \end{cases} \quad \text{where } \vec{u}_L = \begin{pmatrix} 0.18405 \\ 3.8964 \\ 0.5361 \\ 2.4866 \\ 0.3641 \\ 4.0 \\ 2.394 \\ 1.197 \end{pmatrix} \text{ and } \vec{u}_R = \begin{pmatrix} 0.1 \\ -5.5 \\ 0.0 \\ 0.0 \\ 0.1 \\ 4.0 \\ 2.0 \\ 1.0 \end{pmatrix}. \tag{6}$$

The x-component of the magnetic field must be continuous across the boundary in order to satisfy $\vec{\nabla} \cdot \vec{B} = 0$: in one dimension, $\vec{\nabla} \to \frac{\partial}{\partial x}\hat{x}$. Results are given at $t = 0.15$.

The initial conditions (6) were chosen because they lead to both MHD shocks and a contact discontinuity.

To solve (2) numerically on the spatial interval $(0,1)$, the first step is to discretize the interval: designate N equally spaced grid points $x_i = i\Delta x$ separated by $\Delta x = \frac{1}{N} = 0.0025$ for $N = 400$. Each time step will advance the solution by an increment $\Delta t \leq 0.8\frac{\Delta x}{\lambda_{max}}$ satisfying the CFL condition. In this case it is sufficient to choose $\Delta t = 0.00025$. At the grid point (x_i, t_k) the variables are written $\vec{U}_i^k$.

The method used for advancing the time interval is an explicit two-step "predictor-corrector" scheme that assumes the Jacobian matrix $\mathbf{A}$ to be constant within each computational cell, so that the characteristics are piecewise linear. The result is accurate to second order in both space and time.

The first step predicts the value at half a time step:

$$\vec{U}_i^{k+1/2} = \vec{U}_i^k - (\vec{F}_{i+1/2}^k - \vec{F}_{i-1/2}^k)\frac{\Delta t}{2\Delta x}, \tag{7}$$

where the flux at the cell boundary is given by

$$\vec{F}_{i+1/2}^k = \frac{1}{2}[(\vec{F}_{i+1}^k + \vec{F}_i^k - (\Omega_R|\Lambda|\Omega_L)_{i+1/2}^k(\vec{U}_{i+1}^k - \vec{U}_i^k)]. \tag{8}$$

This is the Roe flux, designed to satisfy $\mathbf{A}(\vec{U}_{i+1} - \vec{U}_i) \approx \vec{F}_{i+1} - \vec{F}_i$ [2]. Here Ω_R is the column matrix of right eigenvectors, $\Omega_L = \Omega_R^{-1}$ is the row matrix of left eigenvectors, and $|\Lambda| = \text{diag}(|\lambda_k|)$. The derivation of the matrices is long and cumbersome; the corresponding matrices ω_R, ω_L for the primitive variables are given in [1]. They are related by $\Omega_R = \frac{\partial \vec{U}}{\partial \vec{u}} \omega_R$. The Jacobian matrix $\frac{\partial \vec{U}}{\partial \vec{u}}$ and its inverse can be calculated by hand relatively easily. These matrices are functions of the primitive variables; the spatial half index means to interpolate the primitive variables to the cell boundary. Here the interpolation is simply the arithmetic average of the adjacent cells.

The advance of half a time step leads to a first order accurate estimation, which is insufficient to model discontinuities such as in the initial conditions. The next step uses a discontinuous piecewise linear flux by interpolating $\vec{U}$ and $\vec{F}$ to the left and right sides of each cell boundary separately, rather than a simple averaging at the boundary. For $\vec{u}$, the procedure is as follows:

$$\vec{u}^R_{i+1/2} = \vec{u}^{k+1/2}_{i+1} - \frac{1}{4}((1-\eta)\tilde{\Delta}_{i+3/2} + (1+\eta)\tilde{\tilde{\Delta}}_{i+1/2}) \tag{9}$$

$$\vec{u}^L_{i+1/2} = \vec{u}^{k+1/2}_{i} + \frac{1}{4}((1-\eta)\tilde{\tilde{\Delta}}_{i-1/2} + (1+\eta)\tilde{\Delta}_{i+1/2}) \tag{10}$$

$$\tilde{\Delta}_{i+1/2} = \text{minmod}(\Delta_{i+1/2}, \omega\Delta_{i-1/2}) \tag{11}$$

$$\tilde{\tilde{\Delta}}_{i+1/2} = \text{minmod}(\Delta_{i+1/2}, \omega\Delta_{i+3/2}). \tag{12}$$

Here $\Delta_{1+1/2} = \vec{u}_{i+1}^{k+1/2} - \vec{u}_{i}^{k+1/2}$. The minmod function returns the argument with the least absolute value if the signs are the same; otherwise it returns zero. This limits the slope near discontinuities, preventing spurious oscillations. Smoothly changing variables are not affected. Other slope limiters are available, the choice of which depends on the type of discontinuity to be sharpened. The parameter ω is a number between 1 and 2 that determines the level of artificial compression, allowing better resolution discontinuities. The parameter η is a number between -1 and 1 that weights the spacial dependence of the surrounding cells. Increasing ω results in a small increase in shock resolution, while varying η can have a much more dramatic effect.

The interpolated primitive variables above are then used in the corrector step to achieve second order accuracy:

$$\vec{U}_i^{k+1} = \vec{U}_i^k - (\vec{F}_{i+1/2}^{k+1/2} - \vec{F}_{i-1/2}^{k+1/2})\frac{\Delta t}{\Delta x}, \text{ where} \tag{13}$$

$$\vec{F}_{i+1/2}^{k+1/2} = \frac{1}{2}[\vec{F}(\vec{U}_{i+1/2}^R) + \vec{F}(\vec{U}_{i+1/2}^L) - (\Omega_R|\Lambda|\Omega_L)_{i+1/2}^{k+1/2}(\vec{U}_{i_1/2}^R - \vec{U}_{i+1/2}^L)]. \tag{14}$$

Both the predictor and corrector are applied for $2 \leq i \leq N-2$ since the flux limiters in the corrector step require a four-point stencil. The two points at each boundary are kept constant at the initial conditions. This condition holds at least for time $t = 0.15$, but if the solution were allowed to propagate much further the shocks would reach the boundaries and render this assumption invalid.

Results

The scheme was run for 600 time steps at fixed $\Delta t = 0.00025$. Using Ubuntu with a Pentium 4 3.8GHz CPU, this took 10 to 12 seconds. Plots of results are presented below. Note that the x component of magnetic field remains unchanged after 600 time steps in accordance with the condition $\vec{\nabla} \cdot \vec{B} = 0$. It is possible for this to be violated as numerical error accumulates.

The first plot shows the reconstructed primitive variables with $\omega = 2$ and $\eta = 1$, which corresponds to a central difference scheme. Most of the discontinuities are spread over a few cells. The first type, a contact discontinuity, occurs at $x = 0.5$. Here all quantities except for density are continuous, which occurs if there is no particle flux through the discontinuity surface. The others are MHD shocks.

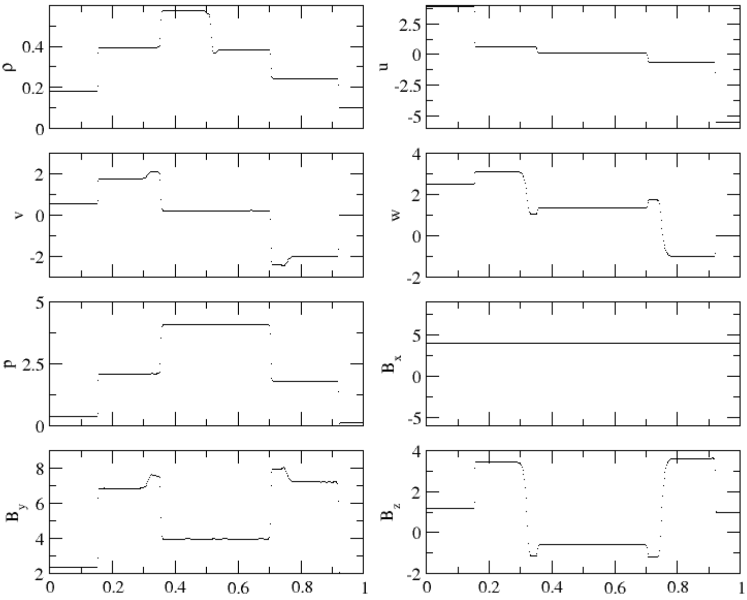

Figure: Primitive variables at time $t = 0.15$ with $\omega = 2$ and $\eta = 1$.

With $\eta = -1$, however, high frequency oscillations appear in the vicinity of the discontinuities. This is because $\eta = -1$ corresponds to an upwind scheme, where the stencil consists entirely of points in the opposite direction of the shock velocity (”upwind” of the shock). Upwind schemes tend not do do well near discontinuities [2], especially in MHD [1].

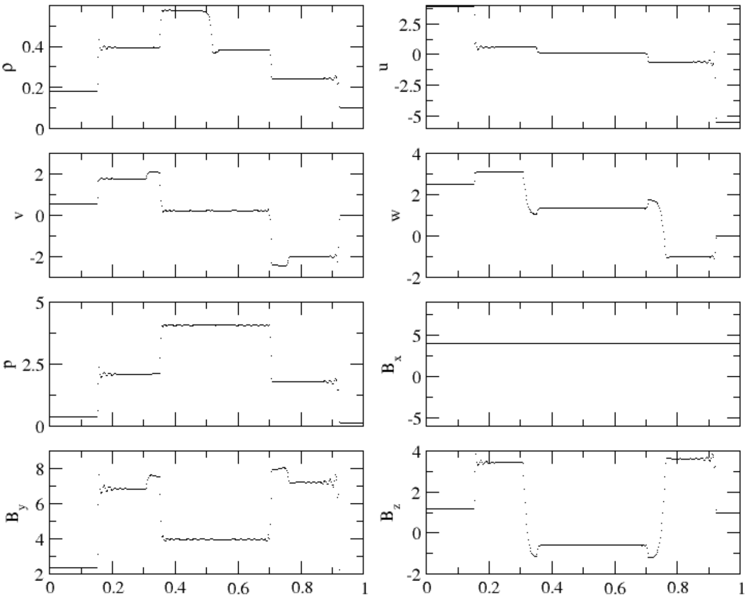

Figure: Primitive variables at time $t = 0.15$ with $\omega = 2$ and $\eta = -1$.

The minmod function is used as a slope limiter to lower the resolution of the scheme in the vicinity of discontinuities to prevent oscillations. Different limiters can be used to sharpen certain types of shock profiles: for example, the superbee limiter

$$\Delta_i^{SB} = S \times \max\left[0, \min(2|\Delta_{i+1/2}|, S\Delta_{i-1/2}), \min(|\Delta_{i+1/2}|, 2S\Delta_{i-1/2})\right], \tag{15}$$

where $S = \Delta_{i+1/2}$, is used for rotational discontinuities where the magnitudes of the velocity and magnetic field are continuous but the individual components are not. Then (9) is written instead as

$$\vec{u}^R_{i+1/2} = \vec{u}^{k+1/2}_{i+1} - \frac{1}{2}\Delta^{SB}_{i+1}, \tag{16}$$

$$\vec{u}^L_{i+1/2} = \vec{u}^{k+1/2}_{i} + \frac{1}{2}\Delta^{SB}_{i}. \tag{17}$$

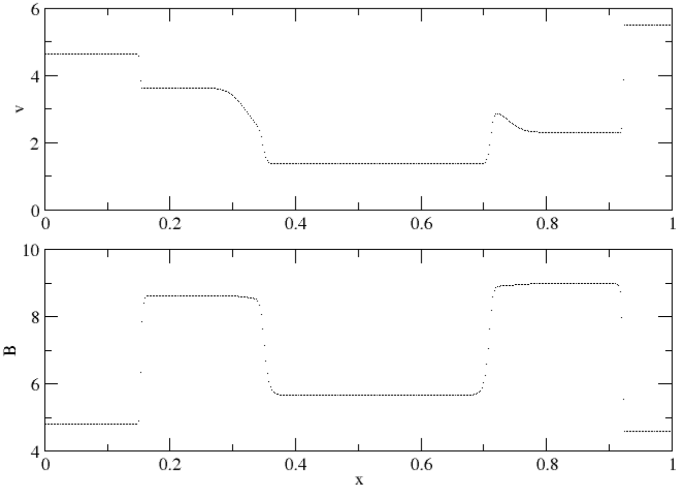

Figure: Magnitudes of v and B at time $t = 0.15$ with $\omega = 2$ and $\eta = 1$ using superbee limiter.

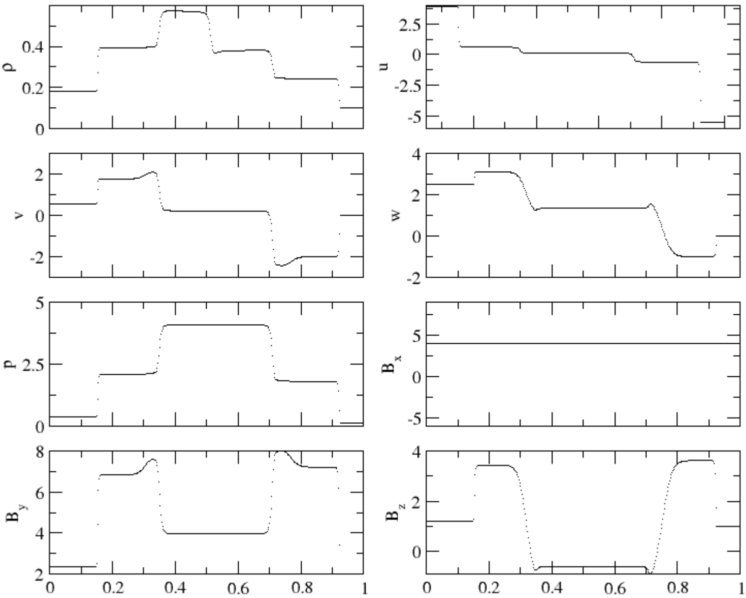

Figure: Primitive variables at time $t = 0.15$ with $\omega = 2$ and $\eta = 1$ using superbee limiter.

However, the initial conditions (6) do not result in rotational discontinuities: comparison of Figs. 3 and 4 shows that the discontinuities in the components of v and B are also present in their magnitudes. From Fig. 4 it is apparent that the shocks are spread over more computational cells, so the superbee limiter is worse than the minmod limiter for this problem.

These plots show that the numerical solution can vary depending on the choice of slope limiter, including the compression parameter ω and weight η, for this approach to the problem. Here the flux at the cell boundaries was found by interpolating the primitive variables and then using them to calculate the flux. It is also possible to find the flux at cell boundaries for the corrector step without explicit reference to the primitive variables:

$$\vec{F}^{(2)}_{i+1/2} = \vec{F}^{k+1/2}_{i+1/2} + \frac{1}{4}\left[(1-\eta)\delta\tilde{\tilde{\vec{F}}}{}^{+}_{i-1/2} + (1+\eta)\delta\tilde{\vec{F}}{}^{+}_{i+1/2}\right] \tag{18}$$

$$- \frac{1}{4}\left[(1-\eta)\delta\tilde{\vec{F}}{}^{-}_{i+3/2} + (1+\eta)\delta\tilde{\tilde{\vec{F}}}{}^{-}_{i+1/2}\right], \tag{19}$$

$$\delta\tilde{\vec{F}}_{i+1/2} = \mathrm{minmod}(\delta\vec{F}_{i+1/2}, \omega\delta\vec{F}_{i-1/2}), \tag{20}$$

$$\delta\tilde{\tilde{\vec{F}}}_{i+1/2} = \mathrm{minmod}(\delta\vec{F}_{i+1/2}, \omega\delta\vec{F}_{i+3/2}), \tag{21}$$

$$\delta\vec{F}^{-}_{i+1/2} = \vec{F}^{k}_{i+1/2} - \vec{F}^{k}_{i}, \tag{22}$$

$$\delta\vec{F}^{+}_{i+1/2} = \vec{F}^{k}_{i} - \vec{F}^{k}_{i-1/2}. \tag{23}$$

Summary

Since the results of the equation (2) with initial conditions (6) for many different parameters are given in [1], it is easy to verify that the attached code leads to the correct numerical solution. The main lesson is that for hyperbolic systems with discontinuous initial conditions, it is important to test a variety of slope limiters and interpolation procedures to find a combination that spreads discontinuities across few computational cells and minimizes oscillations.

The most surprising part of this challenging programming exercise was discovering that such simple initial conditions, constant apart from a single discontinuity, can lead to a complicated solution with several shocks, all moving at different speeds. This is not at all obvious from the equations. Writing the code, seeing how certain errors change the results, and experimenting with the parameters leads to a deeper understanding of the problem than analytical tools alone can provide. In addition, writing a large, complex code with many parts is excellent practice for working with even more complex code libraries used in research, e.g. adaptive mesh refinement. The same is true for the diversity of methods used in the class assignments.

References

A. Kulikovskii, N. Pogorelov, A. Semenov, *Mathematical Aspects of Numerical Solution of Hyperbolic Systems*. Chapman and Hall/CRC Monographs and Surveys in Pure and Applied MAthematics; 118, 2001.

A. Morton, D. Mayers, *Numerical Solution of Partial Differential Equations*. Cambridge University Press, 2005.