

Modeling Protogalactic Collapse and Magnetic Field
Evolution with FLASH

An Undergraduate Thesis by Chris Orban

December 18, 2004

Abstract

1 Introduction

Understanding galactic magnetic fields is one of the most interesting theoretical and observational challenges in modern astrophysics. Radio astronomy has revealed a fascinating variety of strengths and distributions in the magnetic field structures of other galaxies (Beck 2000). The magnetic field of our galaxy can be mapped with other methods, revealing complicated magnetic loops, small scale structure, and reversals on large scales (Binney and Merrifield 1998). Magnetic fields are also a relevant parameter in star formation.

Observations of spiral galaxies, which have magnetic field strengths ranging from $5 - 15 \mu\text{Gauss}$, can be explained with the $\alpha\omega$ dynamo model. It suggests that presently observed magnetic fields are the result of the amplification of large scale toroidal and poloidal modes of the magnetic field over time. Turbulence combined with the differential rotation of the galaxy stretches flux lines causing this amplification (Widrow 2002, Orban poster presentation 2004). However, the $\alpha\omega$ dynamo only explains the mechanism for amplification. It does not explain the origin of the field which is amplified. Thus a “seed” field to the dynamo must be explained.

A variety of candidates for this seed field have been proposed. Cosmological magnetic fields from phase transitions in the early universe have been considered, but observations of the cosmic microwave background are not sensitive enough to place tight limits on the magnetic field contributions of such exotic mechanisms. Ejection of magnetic flux by jets from super-massive black holes in active galactic nuclei (AGN) are strong candidates for the seed field, but so far only order-of-magnitude estimates have been made. (Widrow 2002)

The Biermann battery is also a strong candidate for the seed field. This effect generates magnetic fields when shocks in the gas cause the ions and electrons in the

plasma to separate, generating currents and hence magnetic fields (Biermann 1950). This model is particularly attractive since the seed field in the protogalaxy is generated from simple plasma physics alone, rather than by complex interactions with a jet from an AGN or an external field.

Simulations of the Biermann battery in protogalaxies have indicated that the battery is viable as a candidate for the seed field but more physics needs to be taken into account, and higher resolution achieved (Davies and Widrow 2000), (Ricker, Widrow and Dodelson 2001). Advances in numerical methods, such as adaptive mesh refinement, and advances in computing power have allowed the physics of protogalaxies to be simulated more precisely. We present results from FLASH, an astrophysical code maintained and developed by the University of Chicago, used to simulate a protogalaxy and estimate the magnitude of the Biermann battery.

2 Procedure

A simple numerical model was evolved using FLASH to simulate the gravitational collapse of a protogalaxy. The protogalaxy is treated as a prolate or oblate spheroid of gas (ellipticity = 0.6) with primordial H and He abundances and a dark matter halo. It is placed in an expanding universe with cosmological parameters, $h = 0.7$, $\Omega_m = 0.3$, $\Omega_\Lambda = 0.7$. A visualization of the prolate spheroid is shown in Fig. 1.

The gas is initialized on a Cartesian grid with values of density, temperature and velocity associated with each cell. The gas density distribution through the spheroid is initially uniform. The dark matter halo is simulated with millions of particles that interact gravitationally with the gas and with each other with no collisions. Values of position and velocity are stored for each one. They are also initialized with a uniform

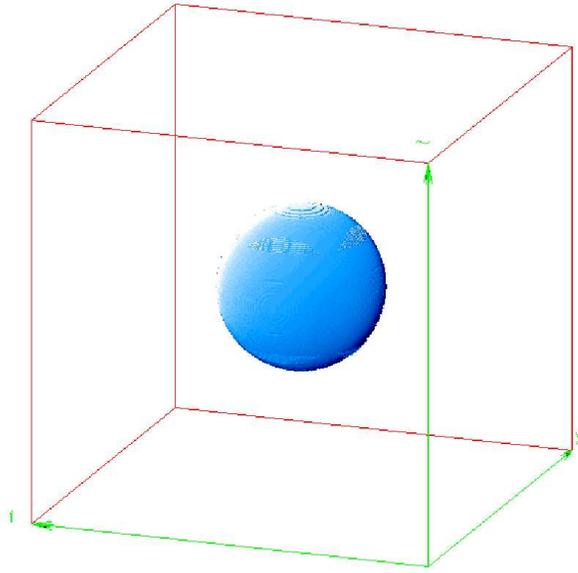


Figure 1: A visualization of the initial prolate spheroidal gas distribution in the protogalactic collapse simulation. The spheroid has an ellipticity of 0.6 with the longer dimension along the z-axis.

density distribution. The particles were evolved with a cosmological leapfrog integration scheme (Ricker et. al. 2004 in preparation) which treats the particle kinematics in co-moving coordinates. The multipole poisson solver was used to determine the gravitational potential at each step. This solver is useful for spherical or nearly spherical problems.

The mass per particle varies with the mass of the protogalaxy (an independent variable) and the total number of particles which is limited by practical considerations of computer memory. In our simulations the practical limit to the number of particles is on the order of millions, which corresponds to a particle mass of order 10^{40} g. Leading theories and observations seem to support the idea that dark matter is composed of some sort of gravitationally interacting fundamental particle. Despite the many orders of magnitude difference that likely exists between the mass of the simulated particles and the much smaller mass of the dark matter particles, the approximation can be thought

of as a Monte Carlo integration of the actual distribution of the dark matter particles. The details of individual particle trajectories are unimportant for the problem at hand.

The gas is evolved using the Piecewise-Parabolic-Method (PPM) which is known to accurately handle shocks (Colella and Woodward 1984). This is an important concern since the Biermann battery generates magnetic fields in shocks. Taking advantage of similarities between the equations of magnetohydrodynamics and hydrodynamics (discussed in Davies and Widrow (2000) and Orban preliminary summer report (2004)) the magnetic field can be inferred from the gas dynamics for weak magnetic fields when the magnetic field pressure is overwhelmed by the gas pressure. In those cases a simple relation between the magnetic field and the curl of the velocity field, also referred to as the vorticity is found,

$$B = \alpha\omega \tag{1}$$

$$\omega \equiv \nabla \times \mathbf{v} \tag{2}$$

where α is a constant ω is the vorticity, and \mathbf{v} the velocity field. The vorticity of the gas is calculated from the numerical curl of \mathbf{v} using centered differencing,

$$\omega = \frac{(v_x(i+1, j) - v_x(i-1, j))}{2\Delta x} + \frac{(v_y(i, j+1) - v_y(i, j-1))}{2\Delta y}. \tag{3}$$

The remaining details of the gridding, parallelization, and code structure are taken care of with FLASH. PARAMESH, an adaptive mesh refinement package, is utilized by FLASH to organize the computational domain in such a way that the resolution scales to interesting areas of the simulation. Figure 2 is provided as an example of FLASH output highlighting this feature.

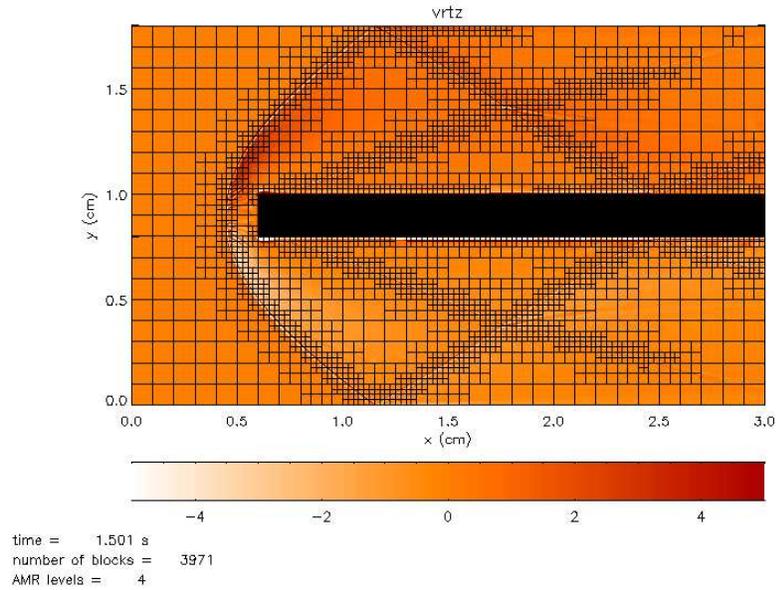


Figure 2: A bow shock simulation with FLASH. Overlaid blocks reveal how the resolution varies in the simulation.

3 Analysis of a 2-D Bow Shock

With the goal of testing the accuracy of vorticity generation, a simple 2-D bowshock simulation was simulated with Mach 3 wind incident on a reflecting step. The bow shock that develops can be compared to analytic solutions. Figure 3 shows a density plot of the bow shock after the shock front has stabilized for the highest resolution run. A test particle is shown to illustrate deflection through the shock.

The data presented here are conclusions from runs at four different levels of refinement. The effective resolution is the equivalent uniform-mesh resolution achieved by the the adaptive mesh refinement.

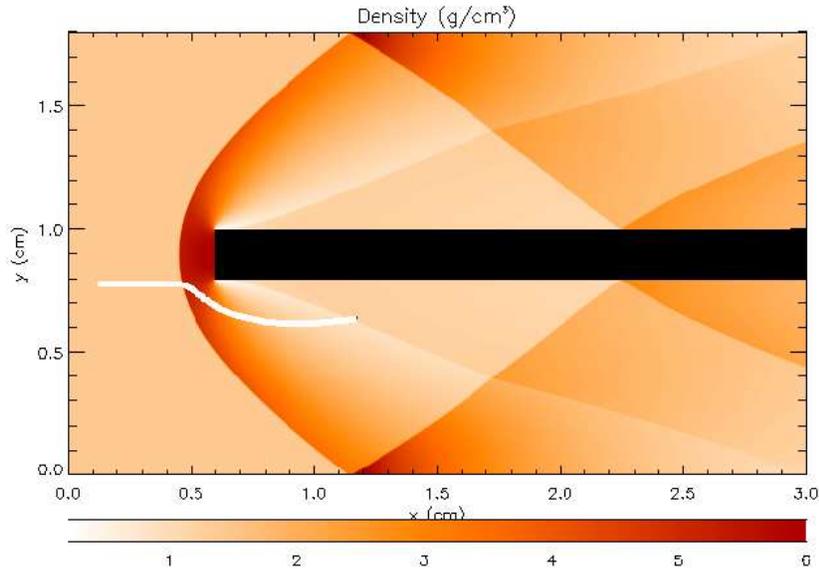


Figure 3: A density plot of a 2-D bow shock simulation. Mach 3 wind is incident on a reflecting step for the highest level of refinement simulated. A particle trajectory is shown to illustrate deflection.

Table 1. The effective resolution for each run.

	x_{cells}	y_{cells}
$l_{refine} = 2$	240	144
$l_{refine} = 3$	480	288
$l_{refine} = 4$	960	576
$l_{refine} = 5$	1920	1152

3.1 Rankine-Hugoniot Jump Conditions

The Rankine-Hugoniot jump conditions are the simplest test of a code’s ability to capture shocks. By combining the equations of mass and momentum conservation across the shock, given the pre-shock density and pressure as well as the shock speed (i.e. the wind velocity), predictions of the post-shock density, pressure and temperature can be made.

At the leading edge of the shock front (i.e. at the point (0.5,0.9) in Fig. 3) because of symmetry the Rankine-Hugoniot jump conditions can be simplified to 1-D to predict

the density, pressure and temperature immediately behind the shock front. For an ideal gas where ρ_{wind} , P_{wind} , and T_{wind} denote the pre-shock values and ρ_{shock} , P_{shock} , and T_{shock} denote post-shock values, the Rankine-Hugoniot jump conditions become

$$\frac{\rho_{shock}}{\rho_{wind}} = \frac{(\gamma + 1)M^2}{(\gamma - 1)M^2 + 2} \quad (4)$$

$$\frac{P_{shock}}{P_{wind}} = \frac{2\gamma M^2 - (\gamma - 1)}{\gamma + 1} \quad (5)$$

$$\frac{T_{shock}}{T_{wind}} = \frac{[2\gamma M^2 - (\gamma - 1)][(\gamma - 1)M^2 + 2]}{(\gamma + 1)^2 M^2} \quad (6)$$

where M represents the Mach number and γ represents the ratio of specific heats for the gas. The Mach number was set to 3 in these simulations and γ was set to 1.4, which is similar to air. The code does a very good job of ensuring the accuracy of the Rankine-Hugoniot jump conditions for the 1920x1152 run. The lower resolutions give similar results.

Table 2. A comparison of analytic values by the Rankine-Hugoniot jump conditions and measured quantities for the 1920x1152 run.

	Wind	Shock Analytic	Shock measured	% Difference
Density	1.4	5.4	5.41351	0.25
Pressure	1.0	10.3333	10.3875	0.5
Temperature	85.9083	230.1494	230.777	0.27

3.2 An Analytic Expression for the Peak Pressure

In the textbook, *Fluid Mechanics* by Landau and Lifshits, an expression for the peak pressure at the leading end of a round body in a bow shock is derived. Their expression is general to a square obstacle so the pressure at the center of the reflecting face of the step (i.e. the point, (0.6,0.9)) can be analytically determined.

$$P_{body} = P_{wind} \left(\frac{\gamma + 1}{2} \right)^{\frac{\gamma+1}{\gamma-1}} \frac{M^2}{(\gamma - (\gamma - 1)/2M^2)^{\frac{1}{\gamma-1}}} \quad (7)$$

The derivation utilizes the fact that a streamline passing through the center of the bow shock would cut perpendicularly through the shock and terminate at the body.

The analytically estimated pressure at this point was $P_{body} = 12.0610$ for $l_{refine} = 5$ and the measured was $P = 12.0588$, a difference of 0.02%.

3.3 Vorticity Generation

Since vorticity generation is incredibly important to understanding the magnetic fields generated by the Biermann Battery, an analytic result for the vorticity jump across a shock was used to check the accuracy of the code in generating vorticity across the bow shock. The vorticity jump follows the relation derived in Kevalahan (1997),

$$\Delta\omega = -\frac{\delta^2}{1+\delta} \frac{\partial v_{\perp}}{\partial \tau} \quad (8)$$

where $\delta = \rho_{shock}/\rho_{wind} - 1$. For a steady bow shock this equation simplifies to

$$\Delta\omega = v_{wind} \frac{\delta^2}{1+\delta} \sin \theta \frac{\partial \theta}{\partial S} \quad (9)$$

where θ is the angle that the normal to the shock front makes with the x-axis. The shape of the bow shock determines the values of $\sin \theta$ and the curvature, $\partial \theta / \partial S$ (where S is the distance from the shock front along the normal). In analysis a 10th order polynomial fit the shape of the shock front was used to infer these values. The densities in the δ variable were sampled ahead and behind the shock along the normal to the shock front. A snapshot of the vorticity after the bow shock has stabilized is shown in Fig. 4.

A comparison of the measured vorticity jump to this analytic expression is shown in Fig. 5. The measured vorticity jump was calculated as the difference in the vorticity variable from ahead of the shock and behind it along the normal to the shock front (at the same sites where the density is sampled for δ).

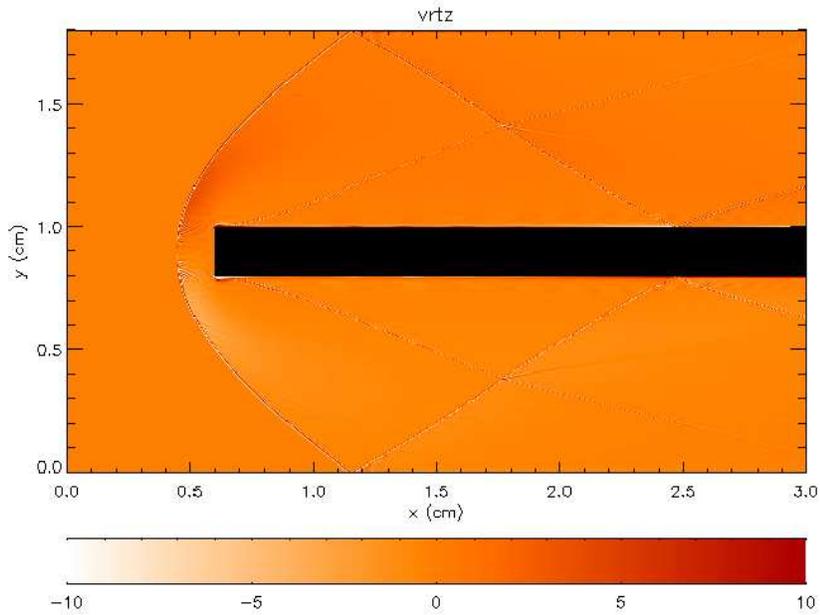


Figure 4: A snapshot of the vorticity in a bow shock simulation at the highest level of refinement.

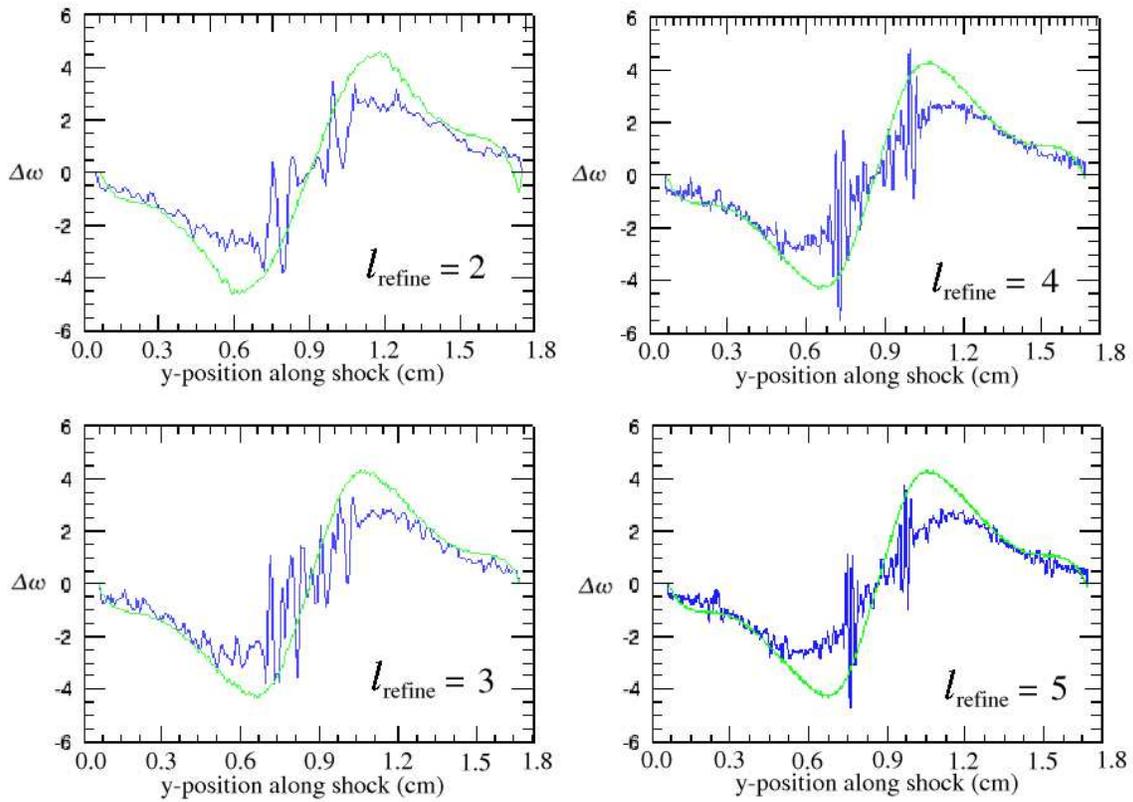


Figure 5: Convergence of the vorticity jump and comparison to an analytic expression.

The results of Fig. 5 are a bit surprising. Overall the measured vorticity jump follows the analytic values, and becomes less noisy with increasing resolution, but the features near 0.7 cm and 1.1 cm persist at high resolutions- a strange result from a smooth-looking shock.

These results are primarily caused by two factors: the pixelation of the shock front and noise from the finite differencing used to determine the vorticity. A closer inspection of Fig. 4 reveals that the pixelation of the shock front creates a sort of rippling effect in the vorticity near the y -values where these features are seen. Figure 6 presents a closer view of the problem.

It is also important to note that error from the higher order terms ignored in the finite differencing also add to the overall noise.

The carbuncle phenomenon was also considered as a possible source of error. The carbuncle phenomenon is a numerical instability that hinders efforts to simulate shock waves in a variety of codes (Dumbser et al. 2003). A severely distorted shape of the shock front is the most recognizable sign of this effect. The cause of the instability, as discussed in Gressier's 1999 Ph.D. Thesis and Mochetta et al. 1995, is due to the eigenvalue of the mode of the vorticity wave.

Though this fact certainly raises an eyebrow in light of the results shown in Fig. 4, the carbuncle phenomenon is not a significant effect in these bow shock simulations for the following reasons:

1. The carbuncle phenomenon is more likely to appear in high-Mach number flows (Quirk 1994). The numerical issues in the measured vorticity still persisted at $M = 1.05$.
2. The shock front is not deformed.

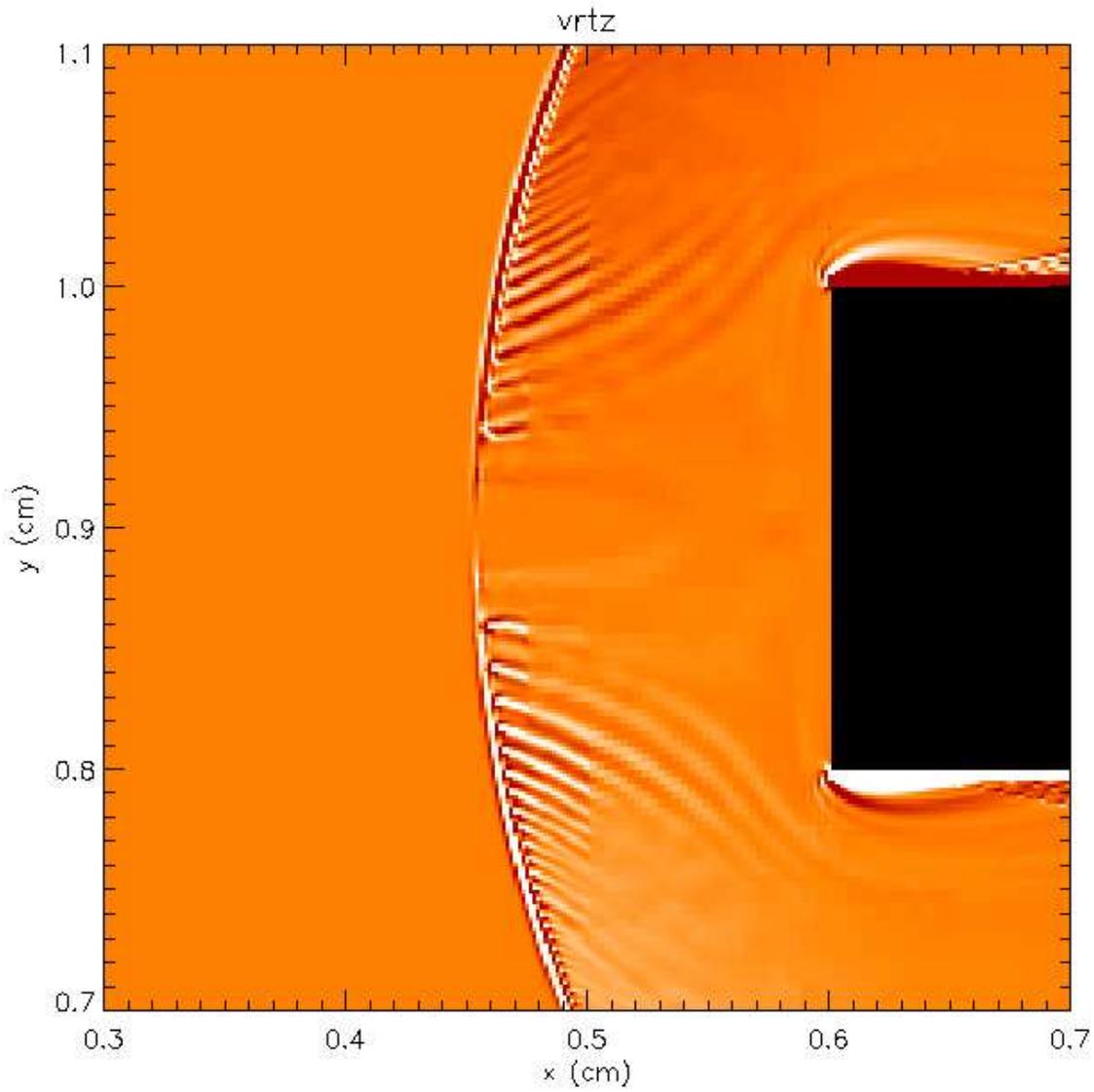


Figure 6: A closer view of the central section of the bow shock from Fig. 4. The shock front creates unphysical ripples of vorticity which explain features near $y = 0.7$ cm and 1.1 cm in Fig. 5.

3. The carbuncle phenomenon is more of a concern when the shock is grid-aligned, which is not the case in these simulations.

Despite this conclusion, the recommended remedy for the carbuncle phenomenon—using a Riemann-HLLE solver on the site of the shock (Quirk 1994)—was still useful for the bow shock simulations, and was already built into the FLASH code. When applied to the bow shock simulation it attenuated the features near $y = 0.7$ cm and 1.1 cm of in measured vorticity jump without significantly increasing computation time. This technique was applied to all of the bow shock simulation results presented in this section. It should be mentioned that the carbuncle phenomenon is a member of a family of “odd-even” problems. The `odd_even` problem in the default FLASH installation is provided as a test case to illustrate this effect.