The University of Minnesota / NASA

Implementation of Adaptive Mesh Refinement in an Implicit Unstructured Finite-Volume Flow Solver

Alan Schwing[*], Ioannis Nompelis[†], and Graham V. Candler[‡]

21st AIAA Computational Fluid Dynamics Conference
June 24-27, 2013

[*] Graduate Student, AIAA Student Member

[†] Research Associate, AIAA Senior Member

[‡] Professor, AIAA Fellow

This work was sponsored in part by the NASA Johnson Space Center Academic Fellowship and the Defense National Security Science and Engineering Faculty Fellowship. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of NASA, Johnson Space Center, or the U.S. Government.

Outline of Presentation

Motivation

Numerical Method

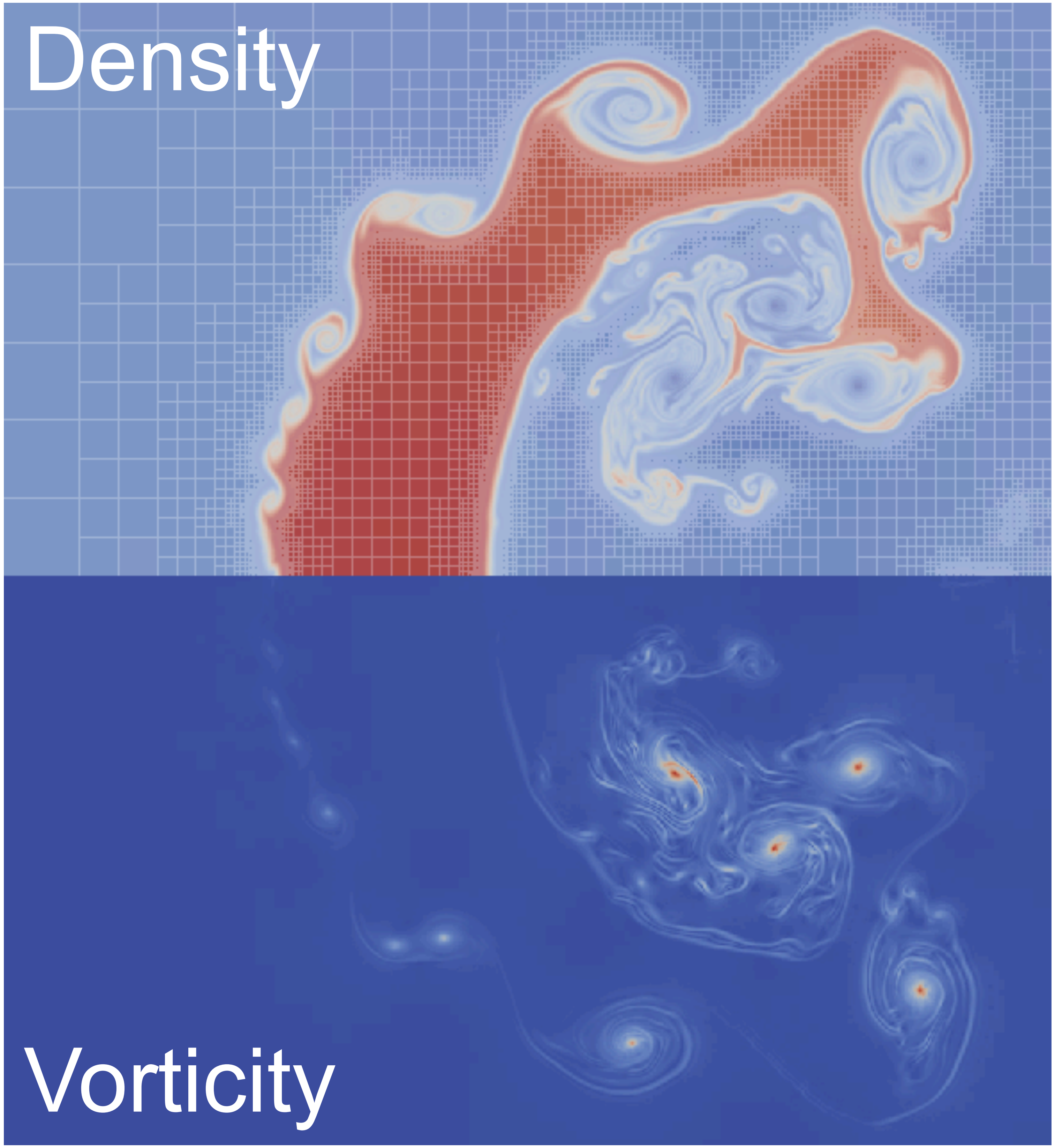
Review of AMR Implementation

High-Order Spatial Flux Validation

Implicit Time Advancement

Double Wedge Results

Conclusions and Future Work



Motivation

Grid generation can be the most expensive portion of a computational analysis - especially for complex shapes with imbedded features. Process is frequently subjective, with user experience influencing a priori assumptions of quality.

Methods used to help 'solve' these problems come with their own limitations.

- Overset Grids
- Unstructured Polyhedra
- Automated translation/smoothing

We are interesting in using solution-driven AMR to place the expensive, grid generation process in the domain of the solver.

Motivational problems include:

- Hypersonic flowfields with strong embedded shocks
- Turbulent flowfields and flows with unsteady, separated wakes

Our first effort in assessing AMR for us in our codes addresses:

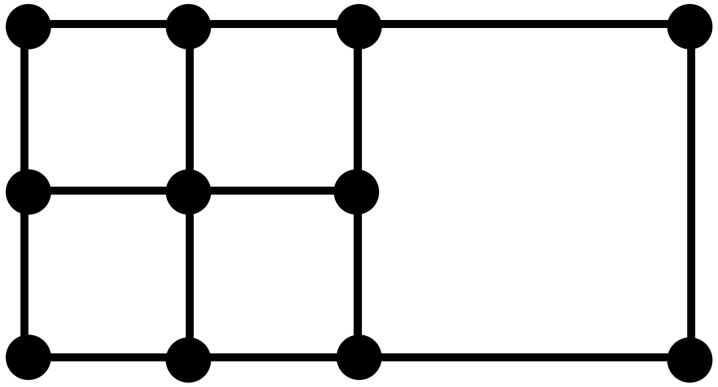
- Understand accuracy of numerical fluxes applied to grids generated with AMR
- Begin working with model problems in order to assess performance and limitations

Finite-Volume Method

$$\frac{\partial U}{\partial t} + \nabla \cdot \left(\vec{F}_c - \vec{F}_v \right) = 0$$

$$V \frac{\partial \bar{U}}{\partial t} + \sum_{faces} \left[\vec{F} \cdot \hat{n} S \right] = 0 \qquad \frac{\partial \bar{U}}{\partial t} = -\frac{1}{V} \sum_{faces} \left[\vec{F} \cdot \hat{n} S \right]$$

The finite-volume formulation effortlessly expresses conservation over arbitrary polyhedra with a generalized number of faces. This same flexibility allows for simple extension to grids with hanging nodes.

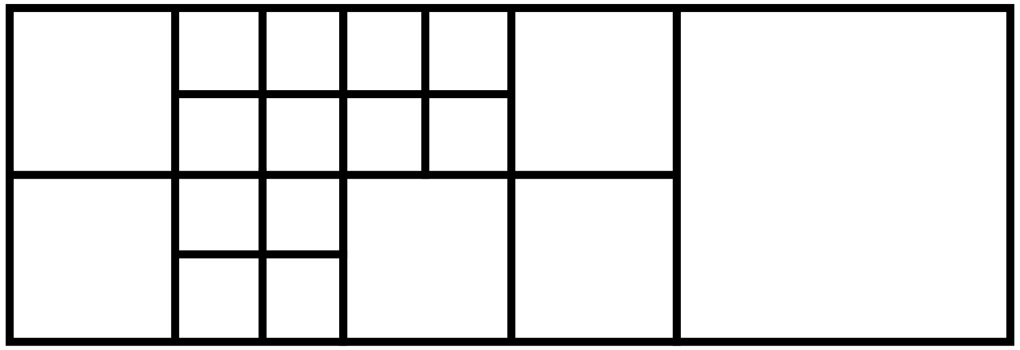


Extending to a first-order time integration with split fluxes:

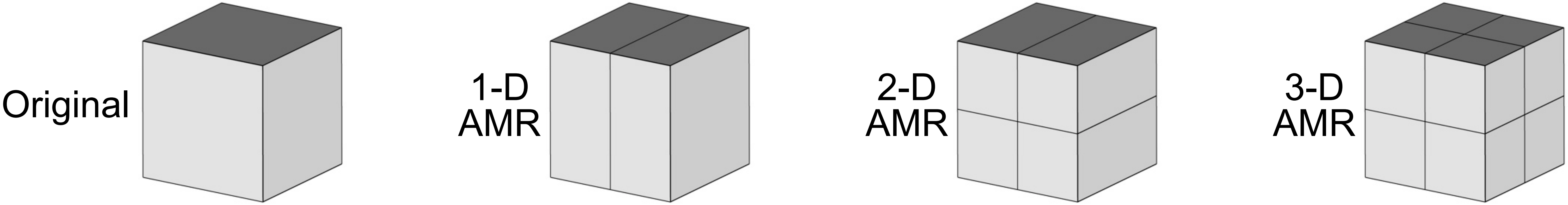
$$\bar{U}^{n+1} = \bar{U}^{n} - \frac{\Delta t}{V} \sum_{faces} \left[\left(\vec{F}_{-} + \vec{F}_{+} \right)^{n} \cdot \hat{n} S \right]$$

AMR Implementation

Our solver accepts a initial grid adds successively finer representations of the grid by replacing coarse elements with finer ones.



We have a 3-D code, but constrain refinement based on boundary conditions.



Each cell is refined independently of the others. With an unstructured framework, this seems the most obvious approach. Additional expense associated with updating connectivity.

For this work, our refinement sensor is an undivided difference in a flow variable across a face. Neighbors are refined if:

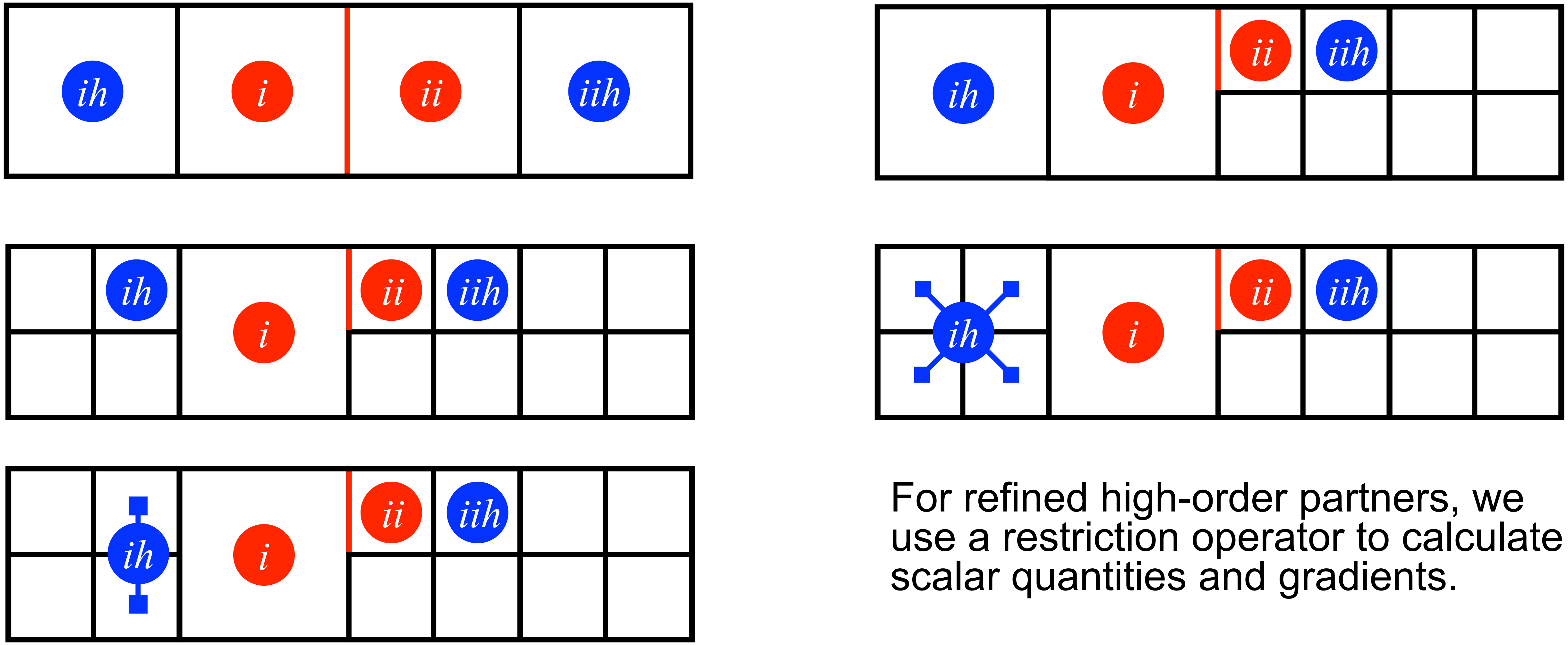
$$\phi_{tol} < \frac{|\phi_i - \phi_{ii}|}{min(\phi_i, \phi_{ii})}$$

High-Order Spatial Fluxes

Problems of interest to our group include problems that involve shock/turbulence interaction and have proven to be sensitive to the method for calculating the numerical flux. It is important to us that we can recover this accuracy on grids constructed with hanging nodes.

The high-order methods require:

- Gradients of flow quantities → weighted-least squares
- Larger numerical stencils → high-order partners



For refined high-order partners, we use a restriction operator to calculate scalar quantities and gradients.

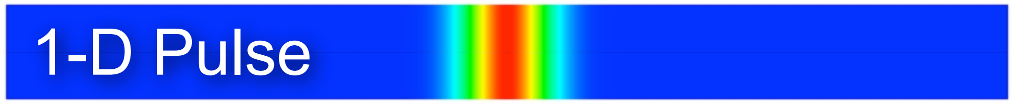
High-Order Spatial Flux Validation

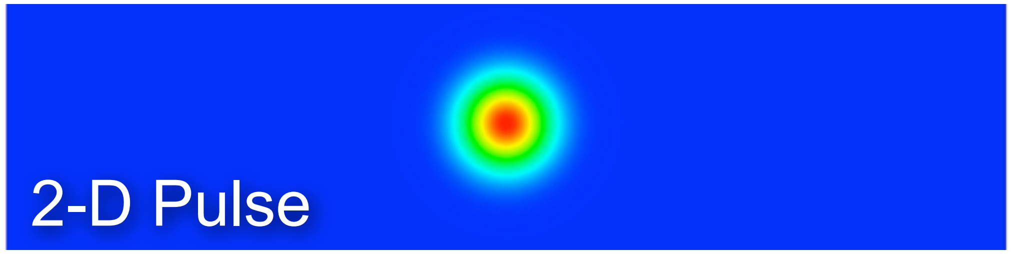
The effect of AMR and hanging nodes on five numerical fluxes are evaluated in this work:

- Modified Steger-Warming fluxes ($\mathcal{O}(\Delta x)$ and $\mathcal{O}(\Delta x^2)$)
- Low-dissipation Kinetic Energy Consistant (KEC) fluxes ($\mathcal{O}(\Delta x^2)$, $\mathcal{O}(\Delta x^4)$, and $\mathcal{O}(\Delta x^6)$)

To limit the effect of the error generated by the time advancement, a third-order RK3 scheme was used with a CFL or 0.1.

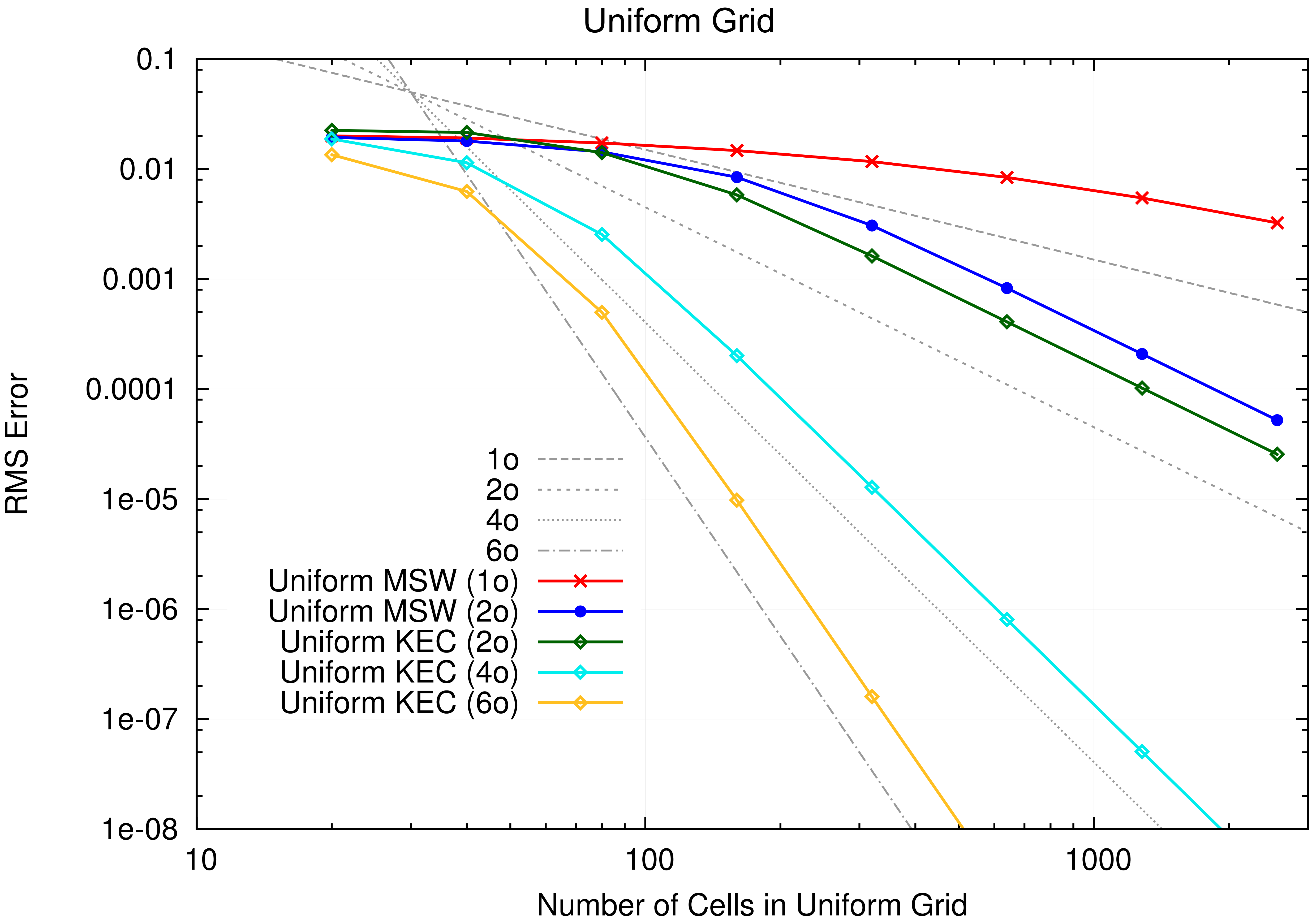
Our test problem is a 1-D and 2-D Gaussian density pulse convecting in a periodic domain.





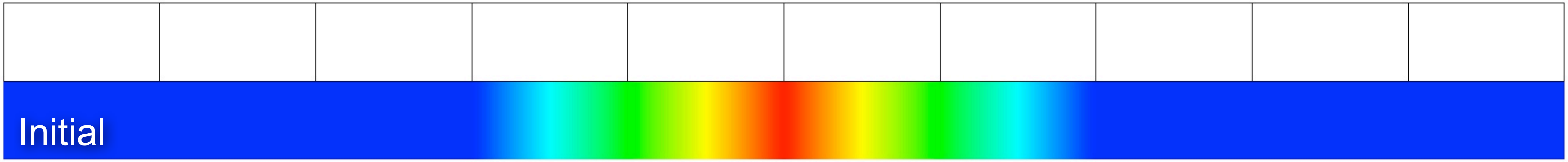
The error in the solution was generated by comparing the numerical results after one cycle. The RMS error from each computational cell is weighted by the cell volume to provide a consistent measure between adapted and uniform grids.

1-D Density Pulse Results

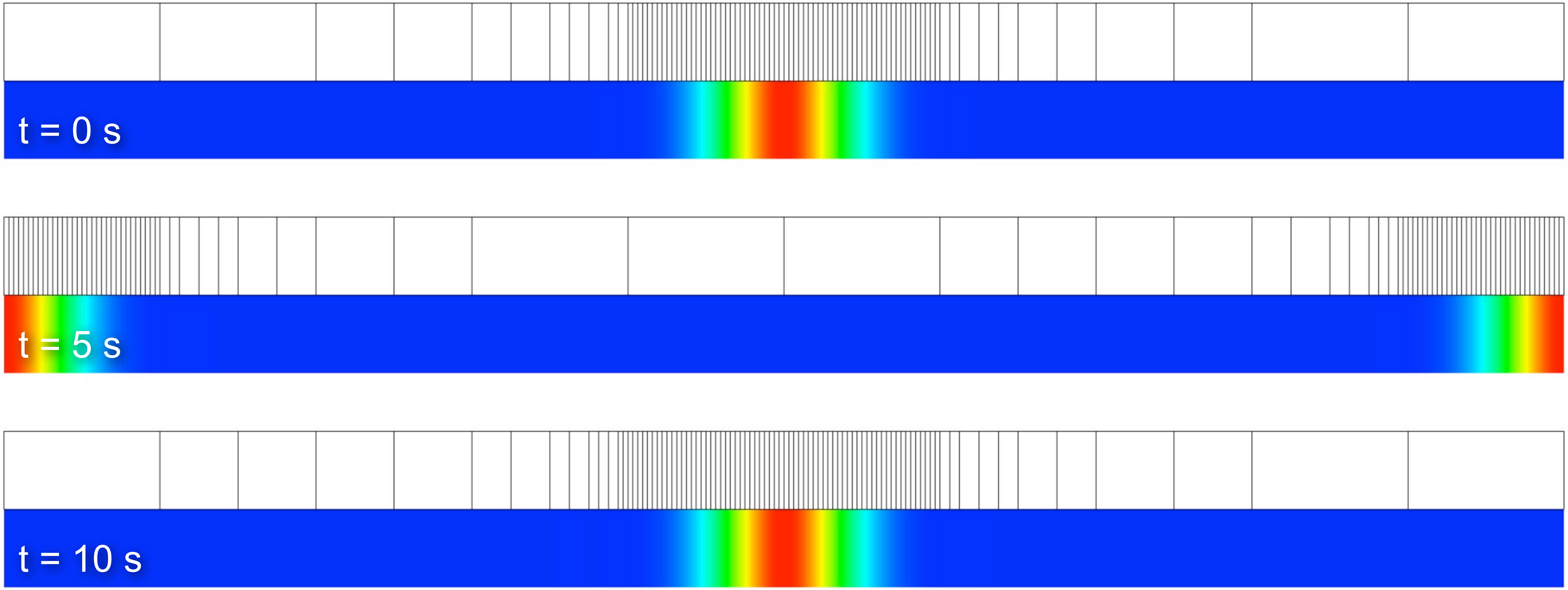


1-D Density Pulse Results

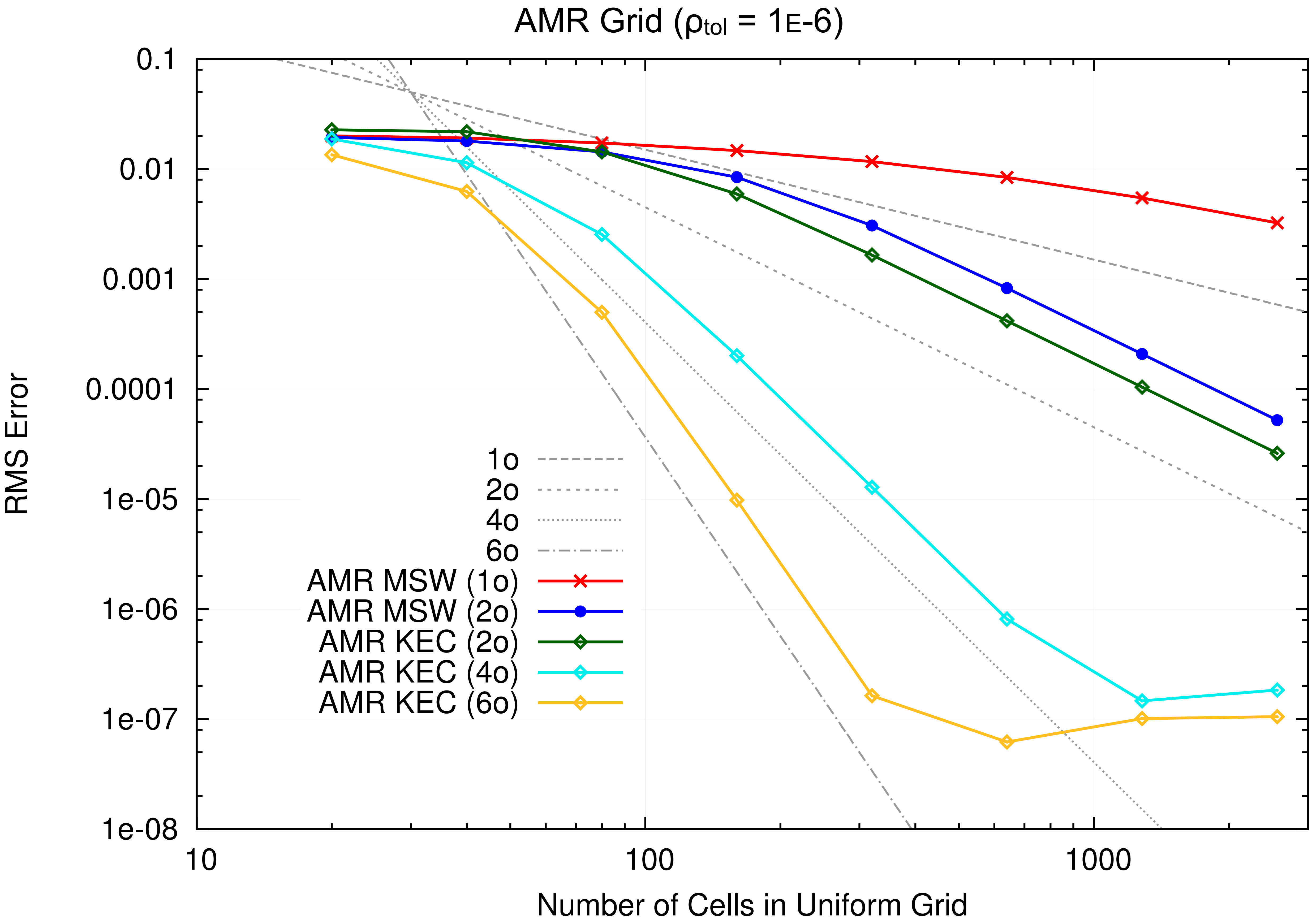
Cases performed with AMR begin with an initial grid having only 10 cells.



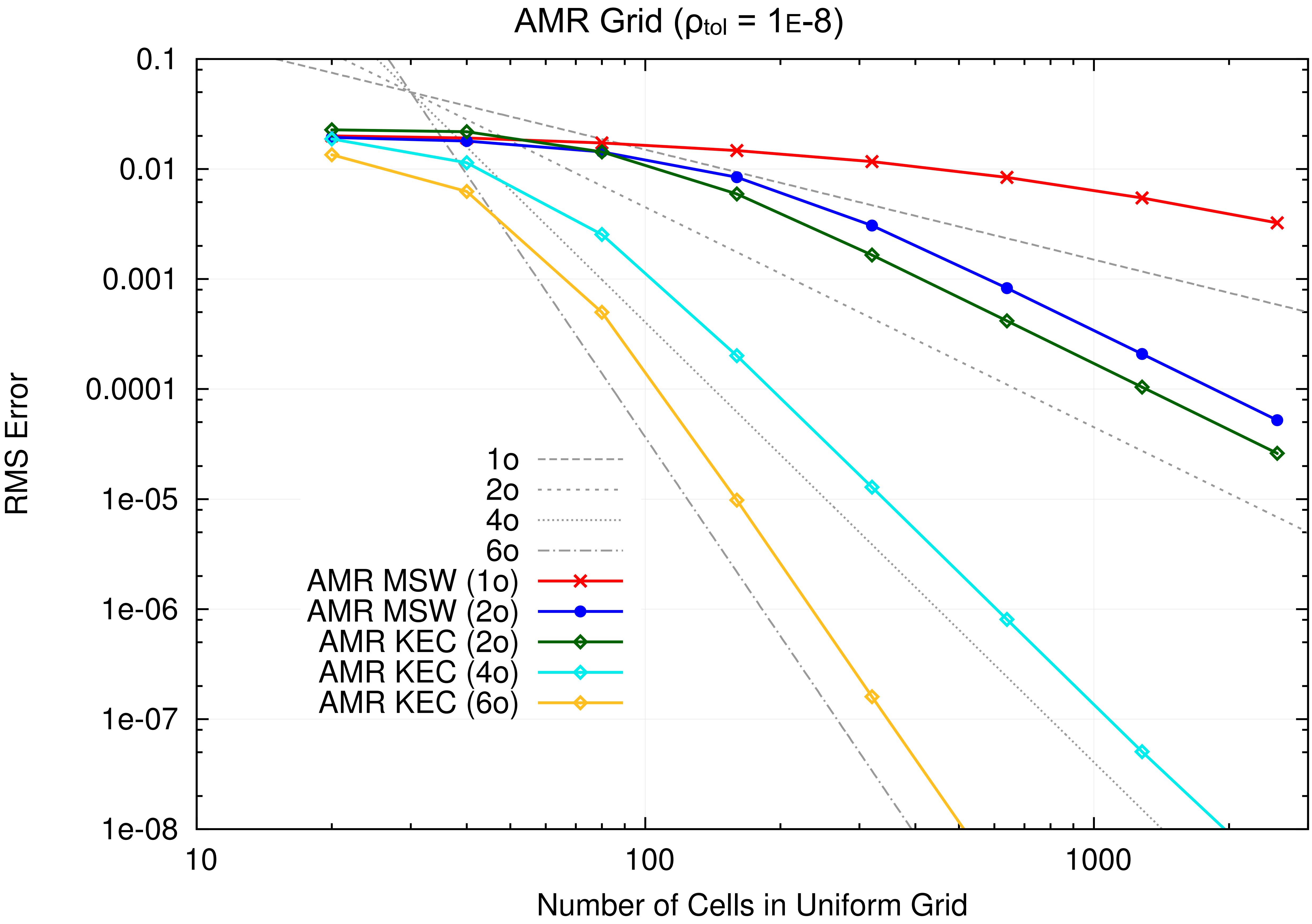
The initial grid is refined to the initial conditions of the density pulse and then allowed to convect.



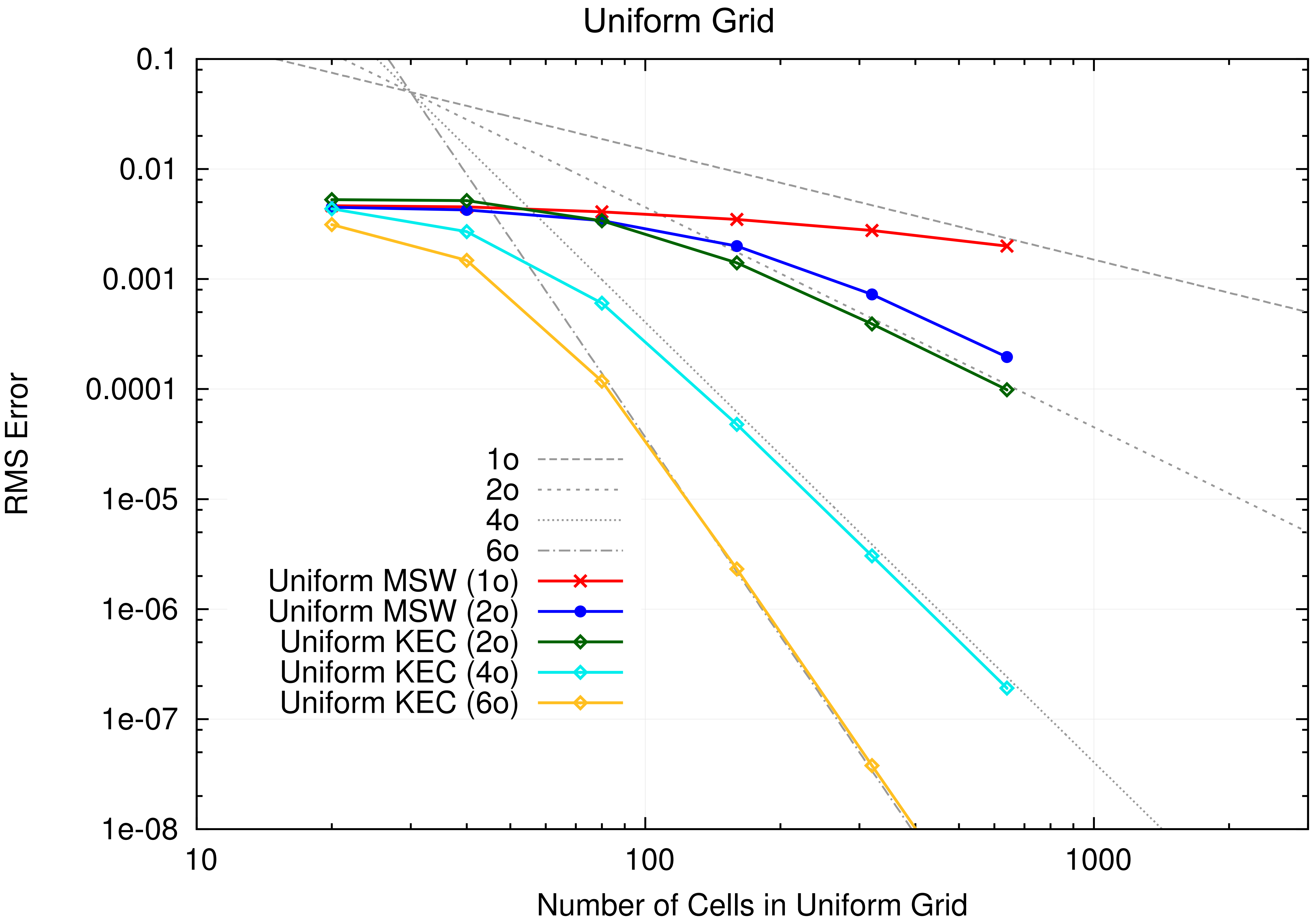
1-D Density Pulse Results



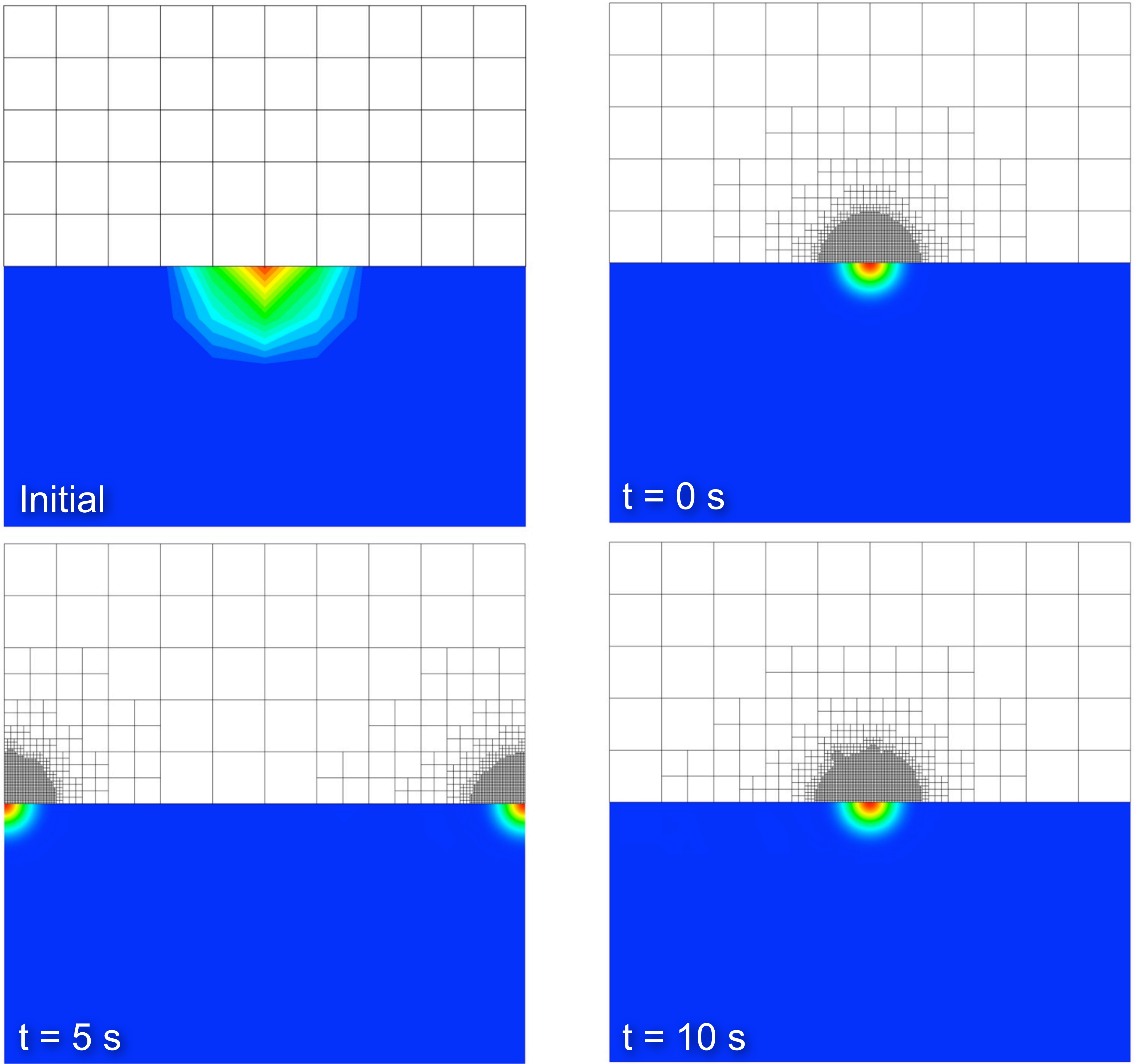
1-D Density Pulse Results



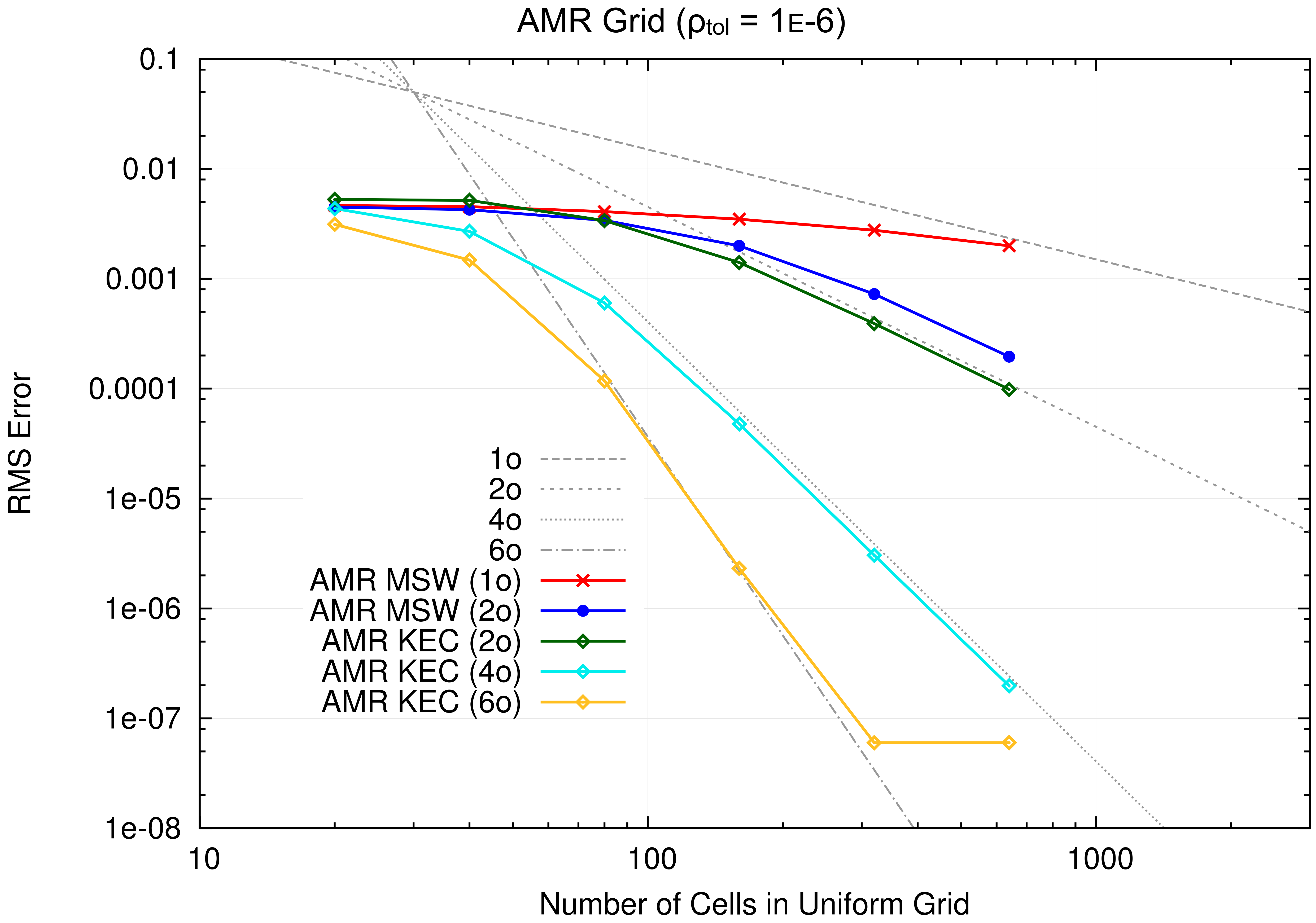
2-D Density Pulse Results



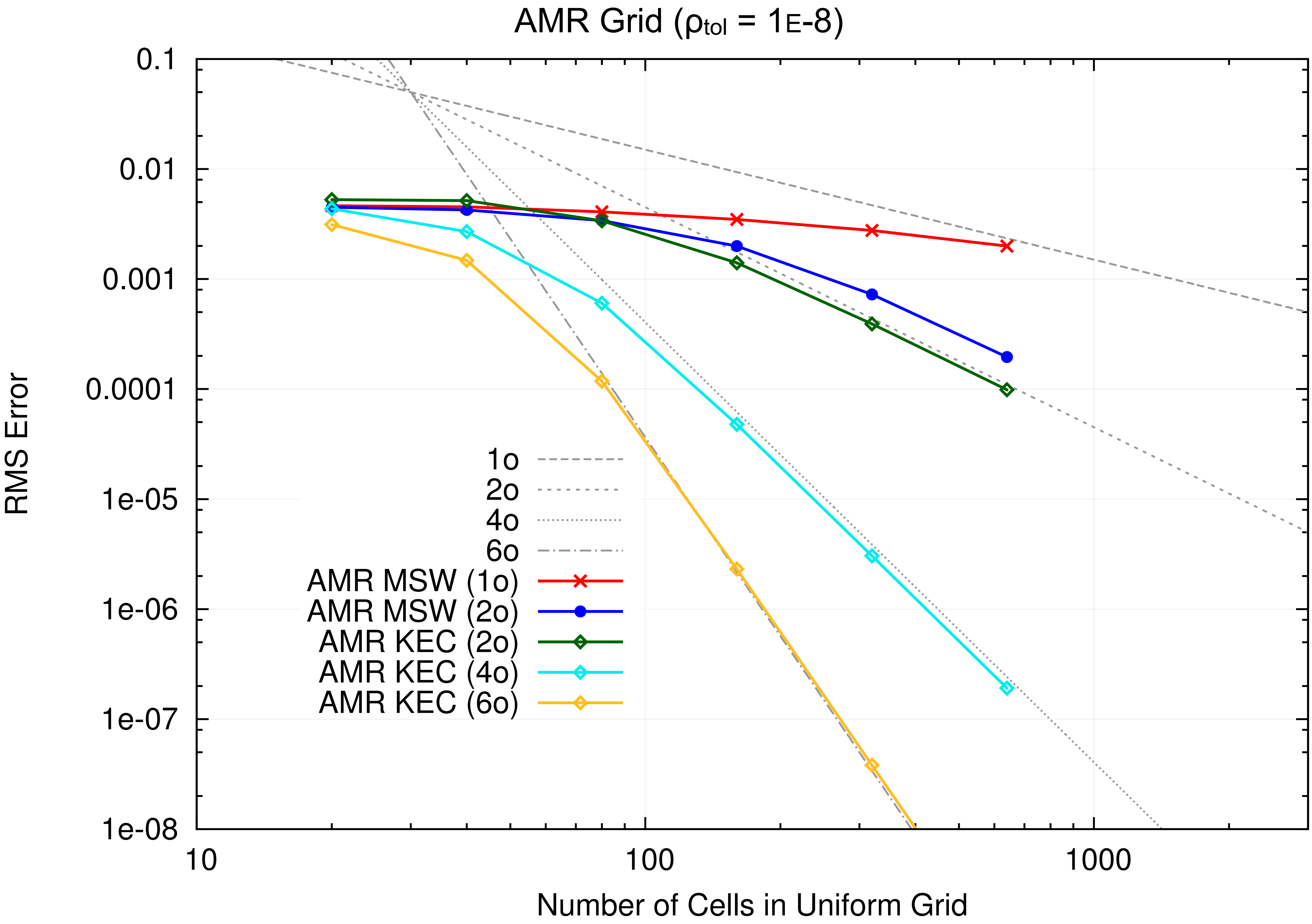
2-D Density Pulse Results



2-D Density Pulse Results



2-D Density Pulse Results



High-Order Spatial Flux Validation

These result indicate that it is possible to recover the accuracy desired for the high-order spatial fluxes using AMR and grids that incorporate hanging nodes.

The minimum attainable error is highly dependent on the choice of refinement threshold. There is still ambiguity about the proper tolerance for the refinement criteria.

- For practical problems, a generalized, automatic sensor would prove beneficial.
- We continue to study feature-based criteria and sensors based on error estimates.

Implicit Time Integration

Moving onto more practical problems, we desire implicit methods to enable timesteps larger than the maximum stable explicit timestep (CFL=1.0).

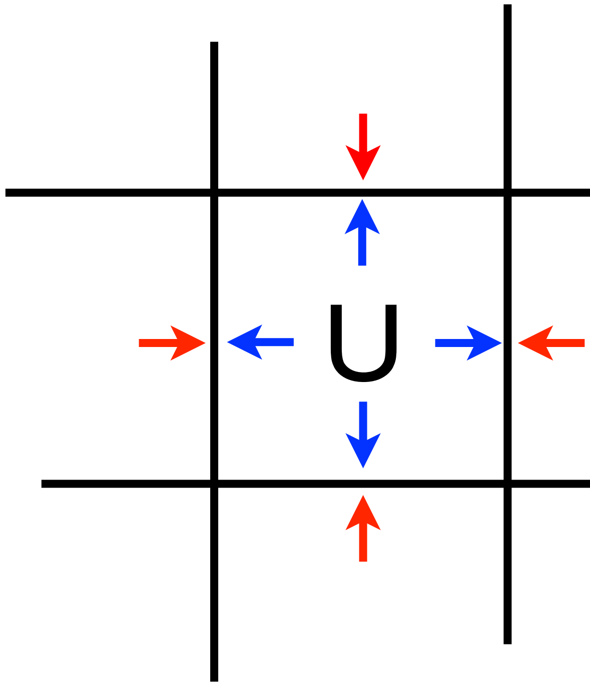
- For steady problems, time to convergence can be reduced if larger timesteps are taken.
- Viscous problems require small cells near the wall that severely limit the maximum explicit timestep.

Certain implicit operators can produce bias in the solution. To avoid this, we implement Full-Matrix Point Relaxation.

Full-Matrix Point Relaxation

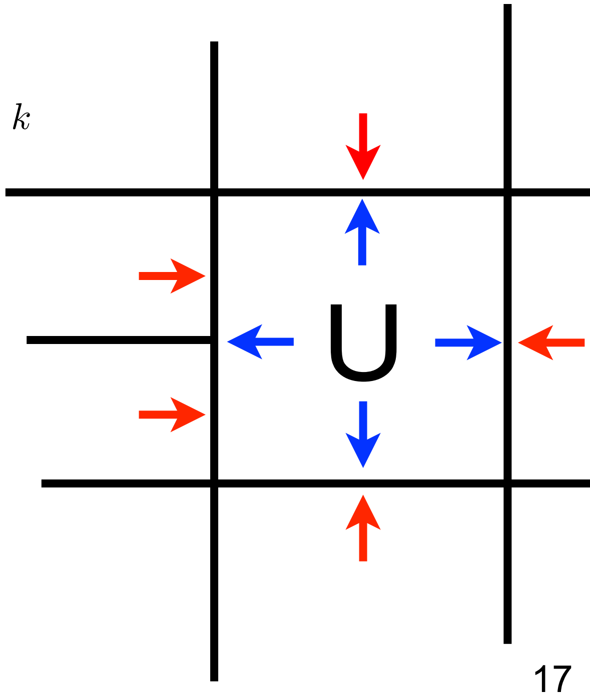
For this work, we construct a block-diagonal matrix that is iteratively solved at each timestep. The influence of the neighboring cells is relaxed and placed on the right-hand side of the system.

$$\begin{bmatrix} \ddots & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \ddots & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \ddots & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \hat{A}_{i,j} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \ddots & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \ddots \end{bmatrix} \begin{pmatrix} \vdots \\ \vdots \\ \vdots \\ \partial U_{i,j} \\ \vdots \\ \vdots \\ \vdots \end{pmatrix}^{k+1} = \begin{pmatrix} \vdots \\ \vdots \\ \vdots \\ \Delta U^n_{i,j} - \hat{B}_{i,j}\partial U_{i,j+1} - \hat{C}_{i,j}\partial U_{i,j-1} - \hat{D}_{i,j}\partial U_{i+1,j} - \hat{E}_{i,j}\partial U_{i-1,j} \\ \vdots \\ \vdots \\ \vdots \end{pmatrix}^{k}$$



By relaxing all of the neighbors, it is easy to expand this to incorporate hanging nodes. Â is updated to include the impact of the new face and another term is added to the right-hand side.

$$\begin{bmatrix} \ddots & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \ddots & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \ddots & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \hat{A}_{i,j} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \ddots & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \ddots \end{bmatrix} \begin{pmatrix} \vdots \\ \vdots \\ \vdots \\ \partial U_{i,j} \\ \vdots \\ \vdots \\ \vdots \end{pmatrix}^{k+1} = \begin{pmatrix} \vdots \\ \vdots \\ \vdots \\ \Delta U^n_{i,j} - \hat{B}_{i,j}\partial U_{i,j+1} - \hat{C}_{i,j}\partial U_{i,j-1} - \hat{D}_{i,j}\partial \tilde{U}_{i+1,j} - \hat{E}_{i,j}\partial \bar{U}_{i+1,j} - \hat{F}_{i,j}\partial U_{i-1,j} \\ \vdots \\ \vdots \\ \vdots \end{pmatrix}^{k}$$



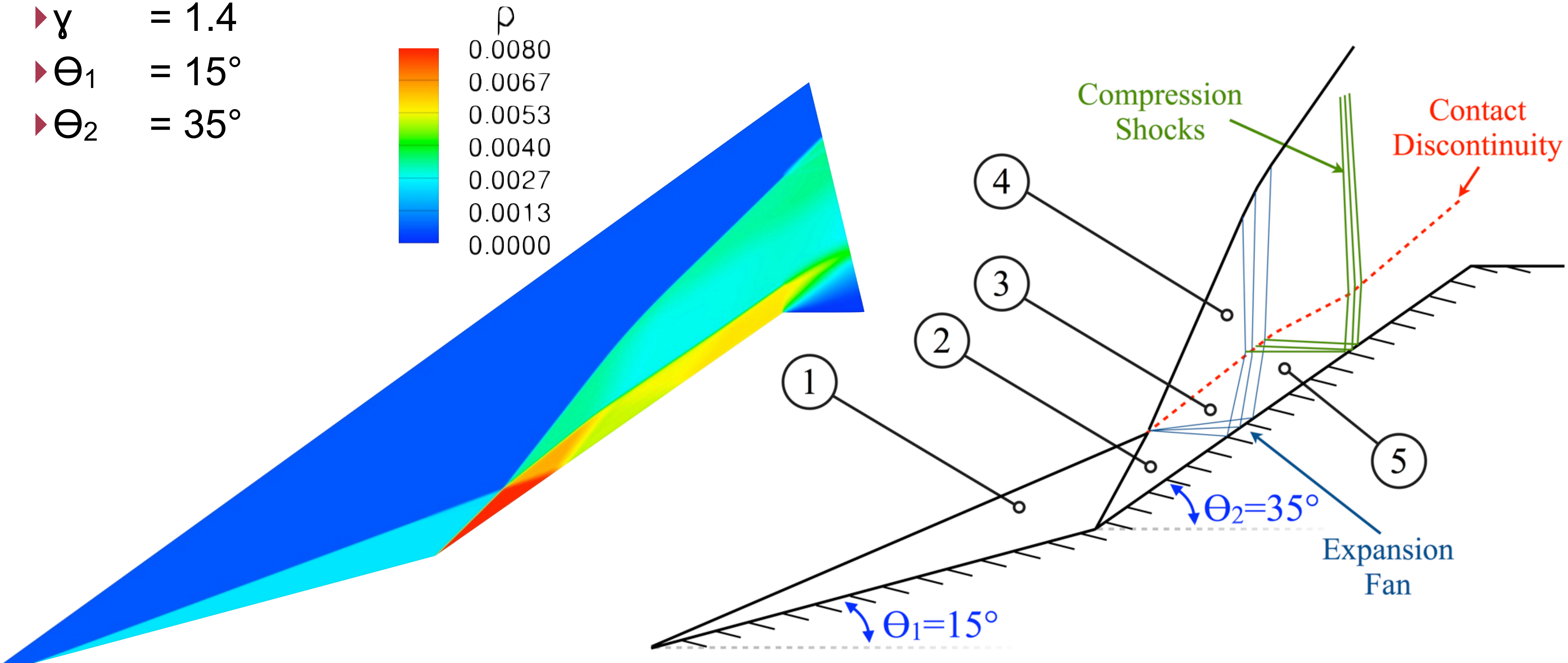
Inviscid Double Wedge

Shock-shock interactions are an important component in many hypersonic flowfields. A popular model problem used to validate computation code is the double wedge.

The flowfield exhibits characteristics found in real-world hypersonic environments and will help us understand how AMR might contribute to problems similar to these.

From the experience of others, we selected a test case exhibiting steady flow with attached shockwaves.

- $M_\infty = 9.0$
- $\gamma = 1.4$
- $\Theta_1 = 15°$
- $\Theta_2 = 35°$



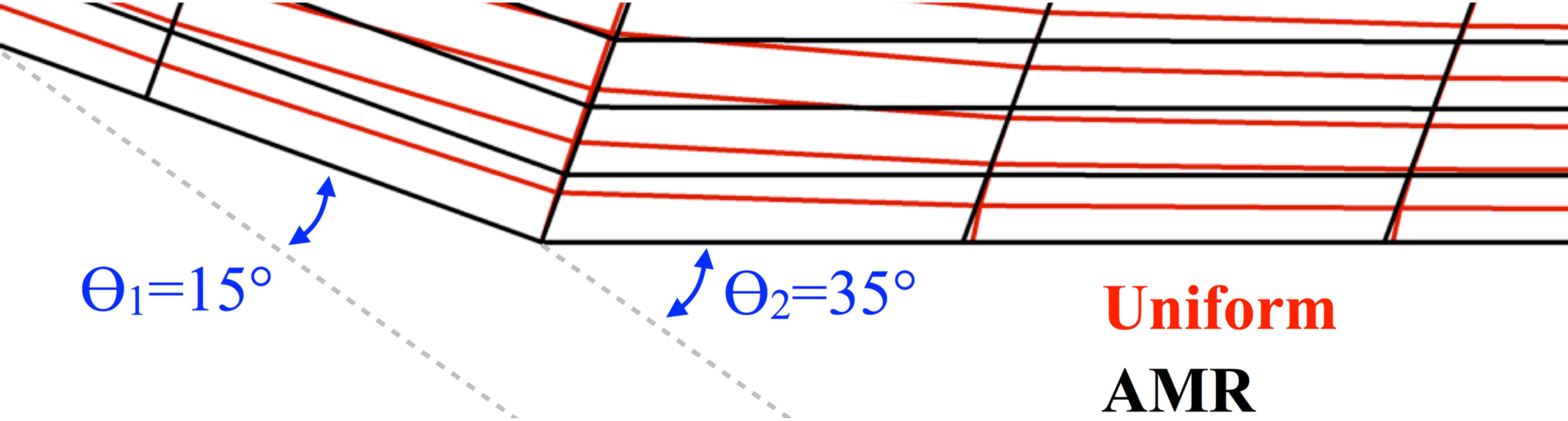
Grid Generation

Previous studies under these conditions showed that 1024x1024 cells were required in order achieve grid converged results.

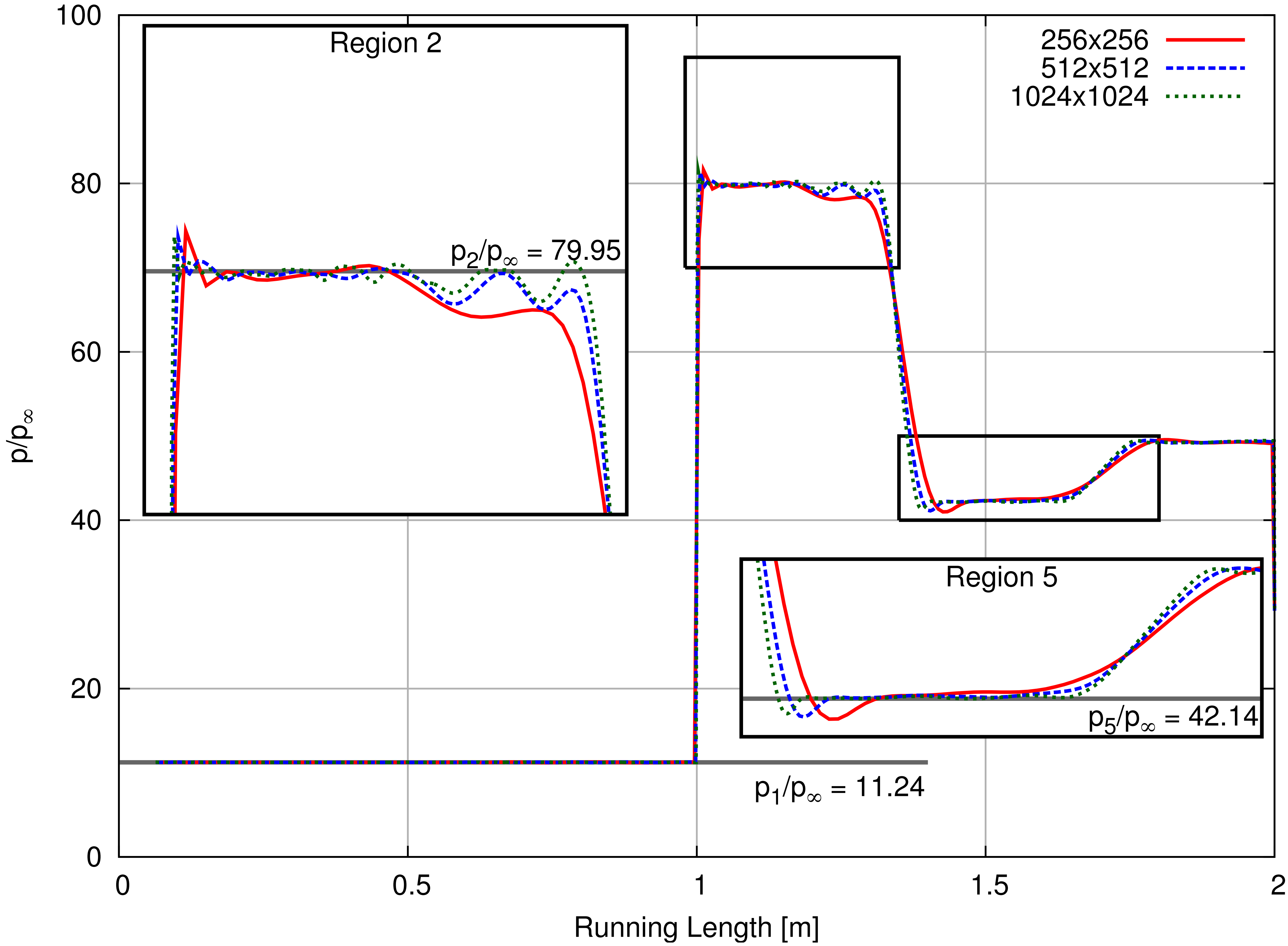
We created grids with 32x32, 256x256, 512x512, and 1024x1024 in order to allow comparison between uniform and AMR grids at various levels of refinement.

When comparing grids created by AMR from the 32x32 grid and the 512x512 uniform grid, discrepancies are evident in the region near the 15°-35° corner.

These differences have noticeable impact on the resulting solution and are caused by our subdivision routine which subdivides cells without clustering or smoothing.



Inviscid Double Wedge Results



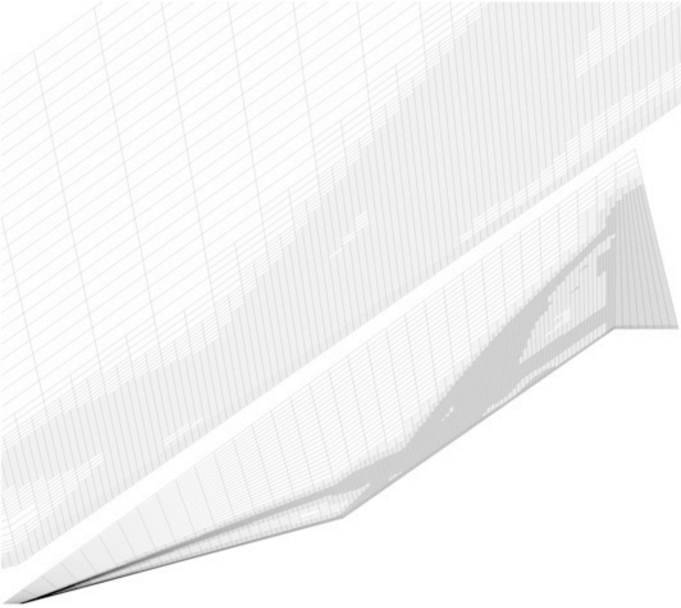
Inviscid Double Wedge Results

AMR results for this case use a combination of two criteria to refine a solution from an initially coarse (32x32) grid.

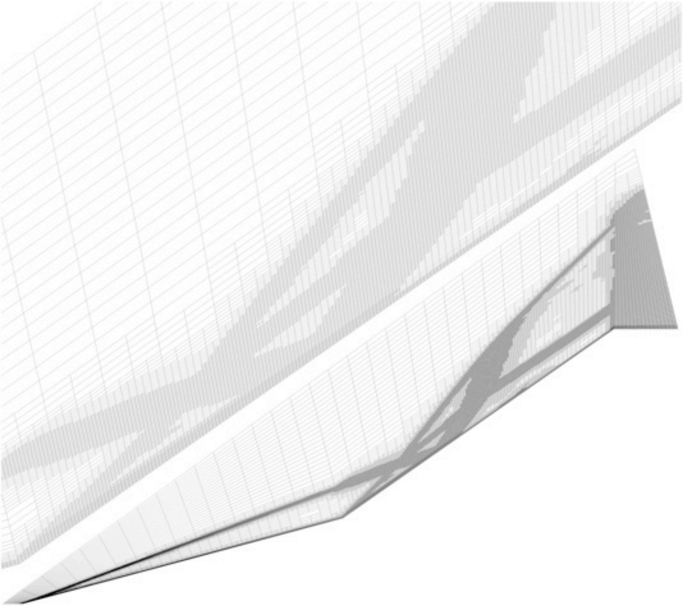
$$\rho_{tol} = 0.01 < \frac{|\rho_i - \rho_{ii}|}{min(\rho_i, \rho_{ii})} \qquad p_{tol} = 0.01 < \frac{|p_i - p_{ii}|}{min(p_i, p_{ii})}$$

Refinement was performed after the density residual had dropped five orders of magnitude from its original value. Once the maximum number of grid levels are obtained, it runs until convergence.

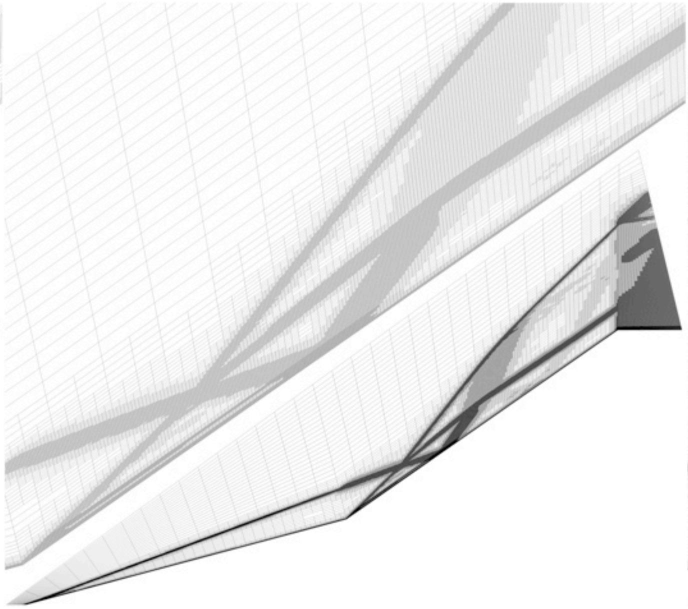
3 Levels of AMR (256x256)



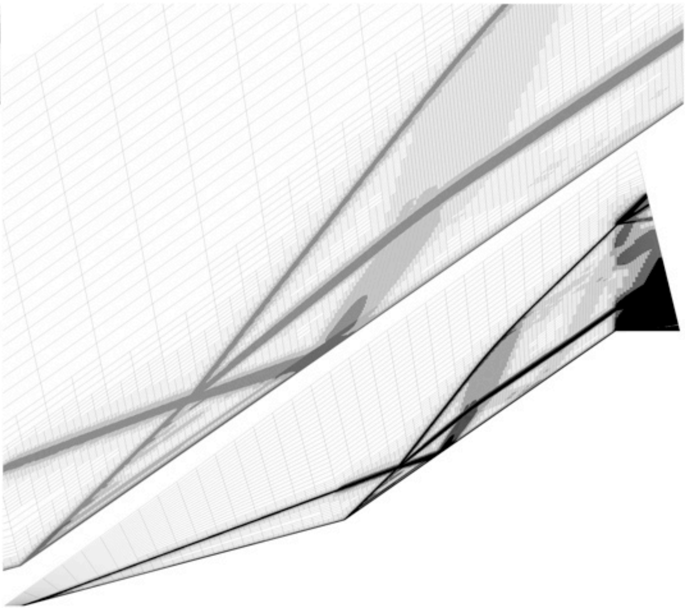
4 Levels of AMR (512x512)



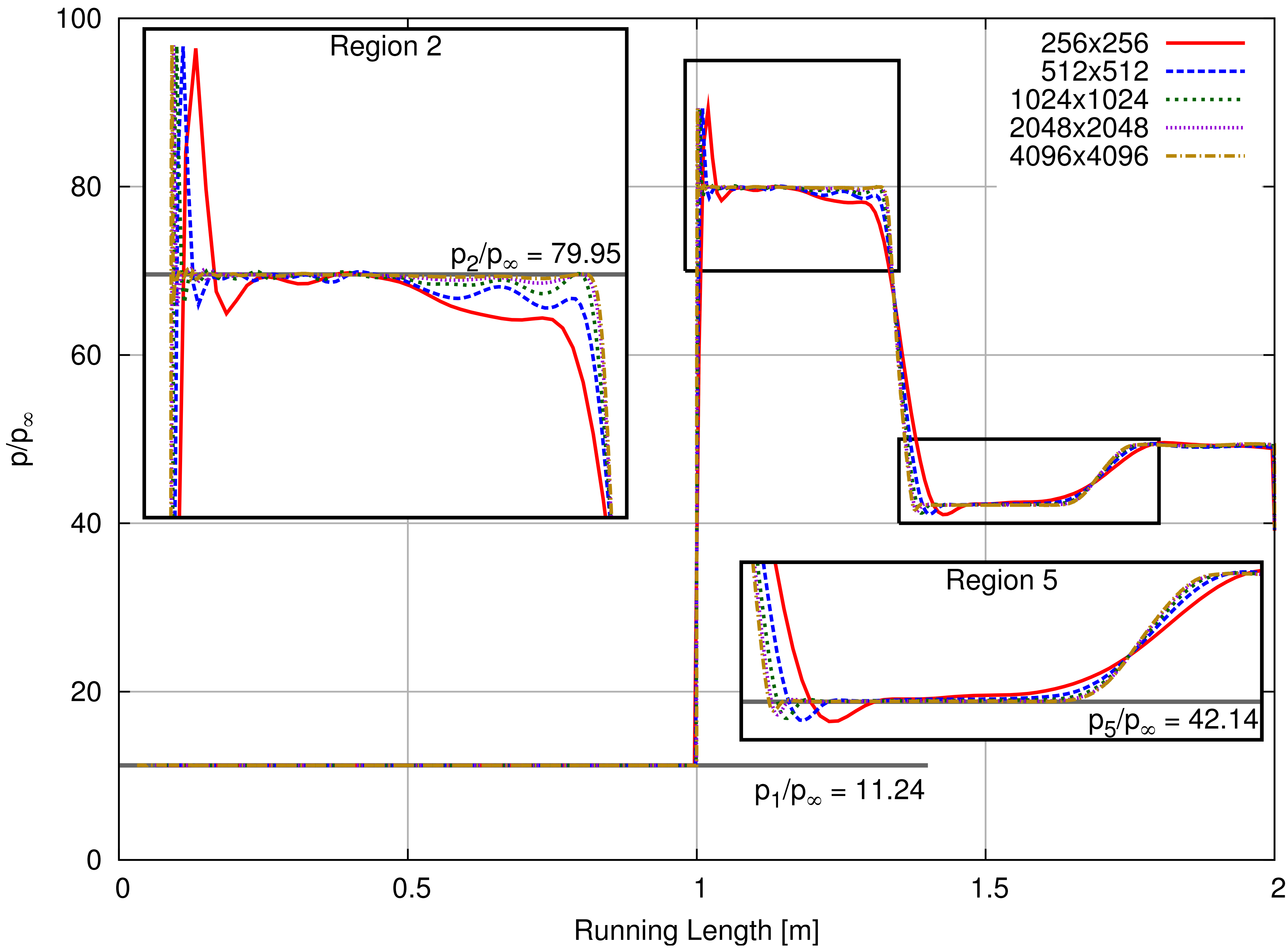
5 Levels of AMR (1024x1024)



5 Levels of AMR (2048x2048)



Inviscid Double Wedge Results



Inviscid Double Wedge Results

Results are promising and illustrate that the AMR method shown here can replicate the physics seen in the inviscid shock-shock interaction.

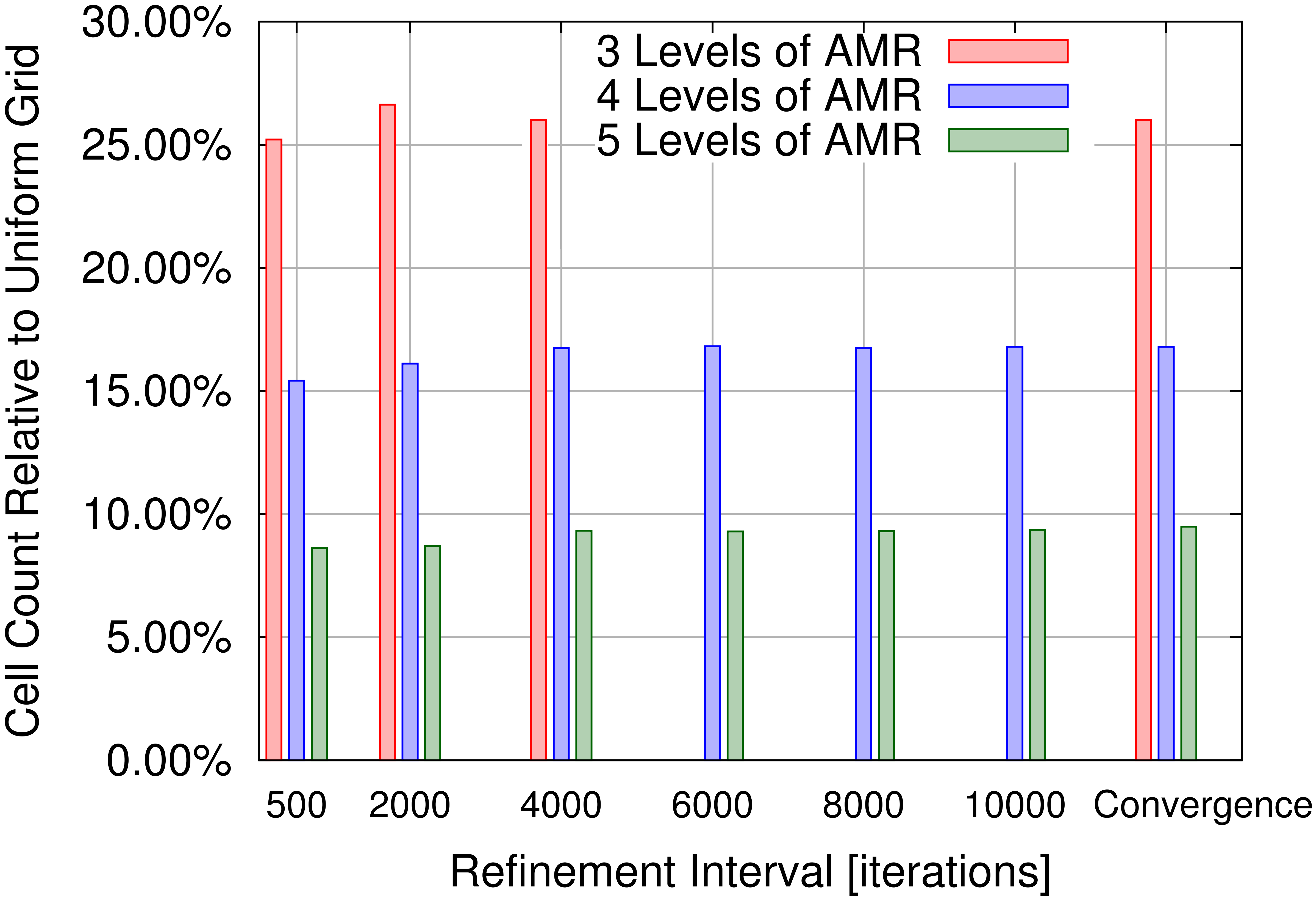
- Predictions closely matched the analytical results in regions 1,2, and 5.
- The AMR simulations were consistent and converged to refined result as additional grid levels were added.

There were differences between the solutions obtained on the uniform grid and those obtained with AMR. These inconstancies are attributed to the method in which we subdivide, without smoothing, when performing refinement.

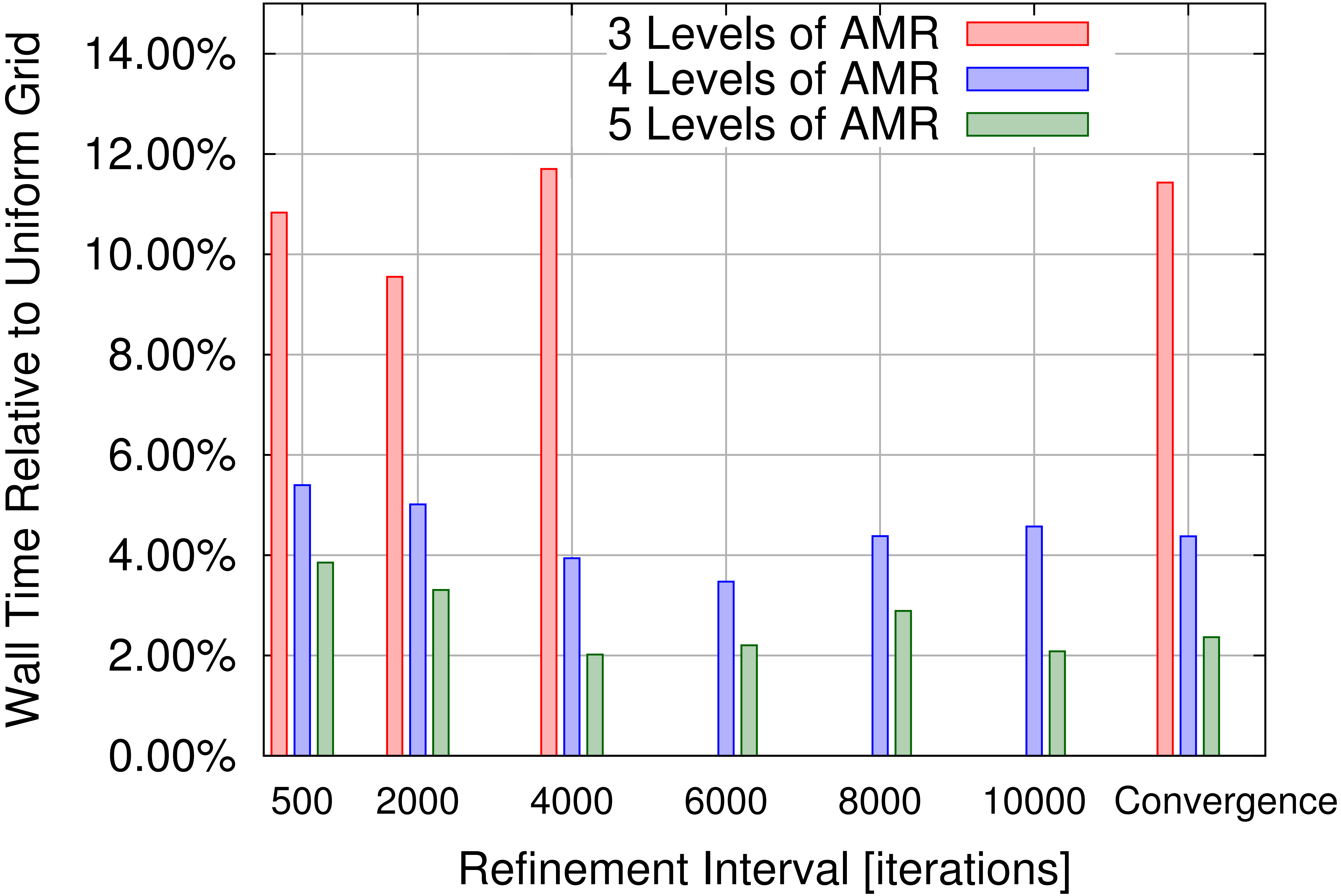
One of the expected impacts of AMR is improved efficiency caused by the reduction in the number of points required for a given simulation.

Additional cases were run that used more frequent invocation of the AMR subroutine. We were interested in understanding if refinement frequency greatly impacted the efficiency of the method.

Inviscid Double Wedge Results



Inviscid Double Wedge Results



Conclusions and Further Work

Grids generated using AMR with hanging nodes are compatible with high-order fluxes created for uniform grid.

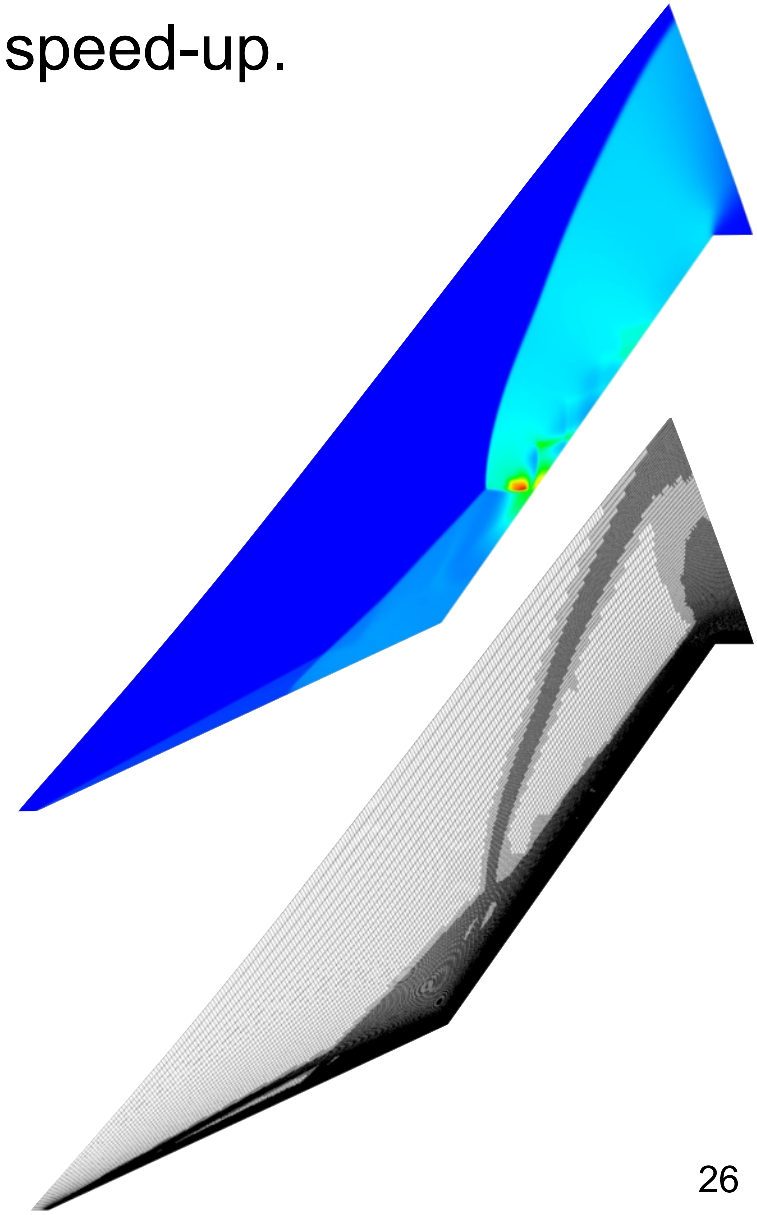
- Handling of gradients and high-order partners appears to be sufficient.
- Work is required in extending the proper refinement tolerance to more general problems.

AMR shows promise at simulating phenomena associated with problems involving shock-shock interaction.

- Drastically reduced element count and significant computational speed-up.
- Hanging nodes do not appear to be a source of error.

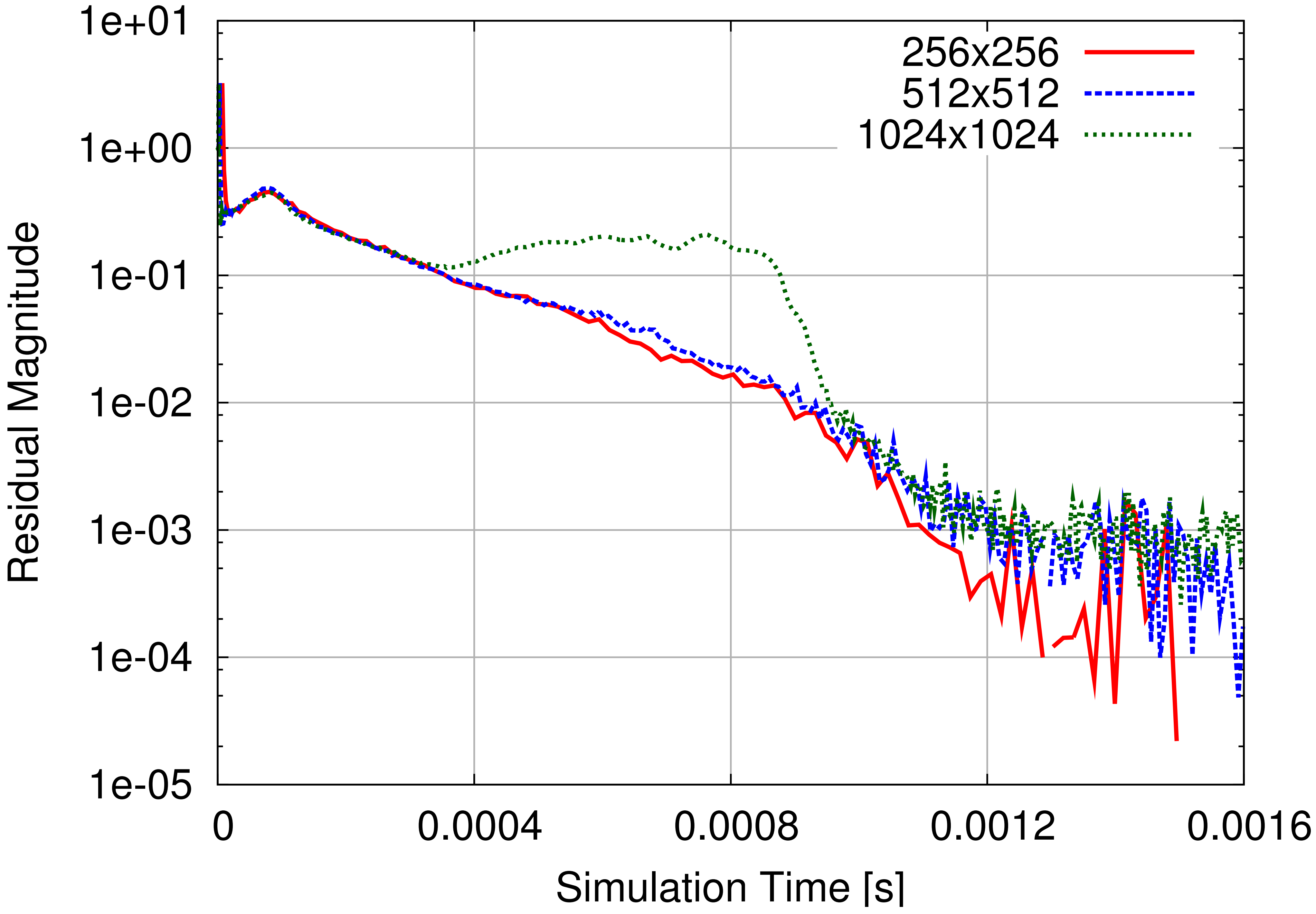
Future work:

- Incorporate the viscous fluxes into the flow solver
- Investigate more advanced implicit methods
- Continue to test other criteria for refinement
- Develop method for surface projection and grid smoothing
- Parallelize by means of OpenMP and MPI to enable more complex and three-dimensional problems

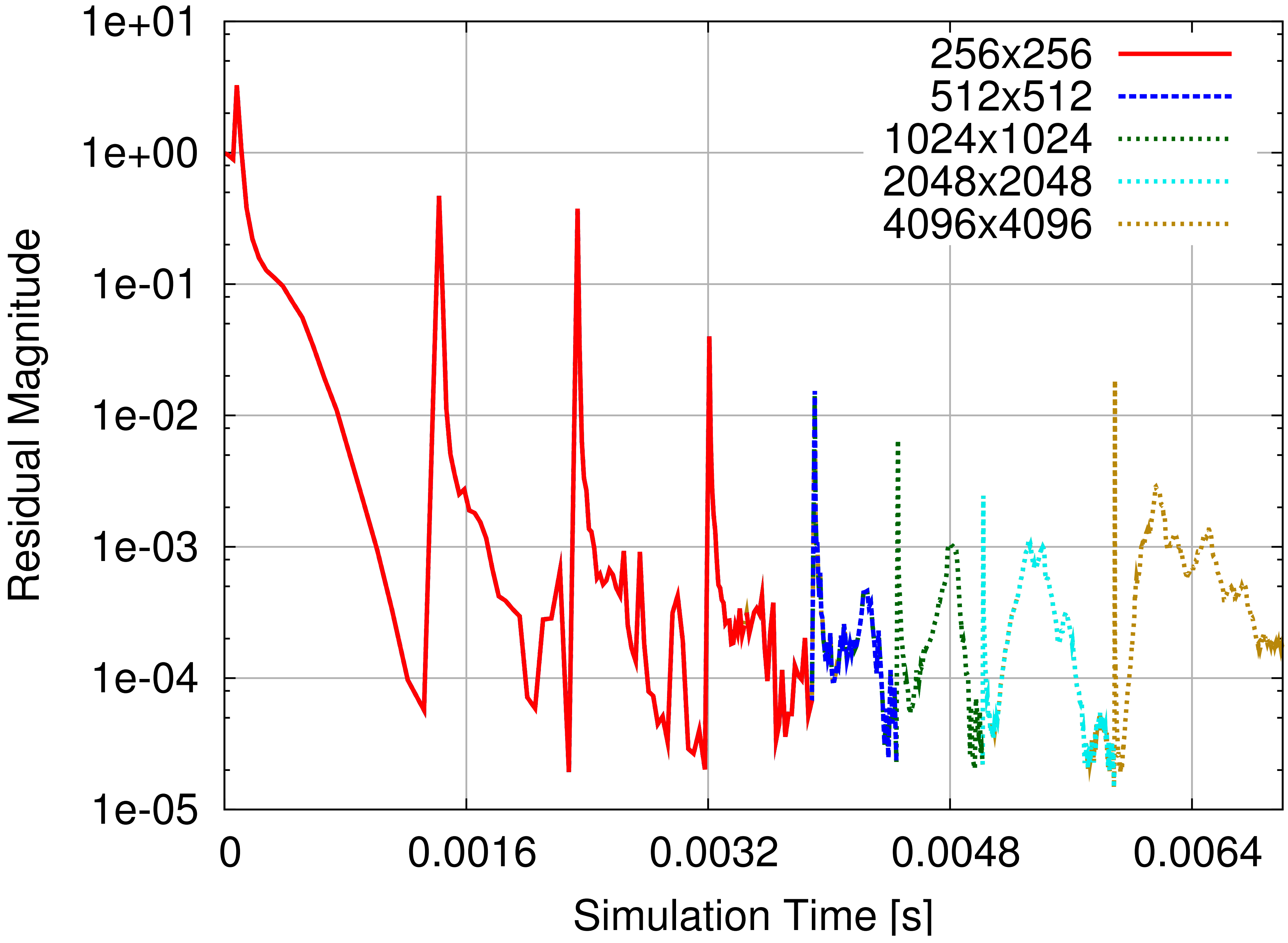


BACKUP

Residual Convergence



Residual Convergence


Residual Magnitude
1e+01
1e+00
1e-01
1e-02
1e-03
1e-04
1e-05
Simulation Time [s]
256x256
512x512
1024x1024
2048x2048
4096x4096