

Numerical Simulation of Ultrasound Propagation in Liver Cirrhosis: Implicit Solver Analysis

Final Project Report

December 28, 2025

Abstract

This report analyzes the numerical performance of an implicit Crank-Nicolson solver for the 2D acoustic wave equation in a heterogeneous liver model. We investigate the stability of the implicit scheme, compare the convergence rates of various linear solvers (Jacobi, Gauss-Seidel, SOR, CG, and Multigrid), and analyze their computational complexity. The results highlight the trade-offs between numerical stability, accuracy, and computational efficiency.

1 Introduction

The simulation models ultrasound propagation through a domain containing a cirrhotic lesion ($c = 1600$ m/s) within healthy tissue ($c = 1550$ m/s). The governing equation is discretized using the Crank-Nicolson method to form a linear system $Ax = b$ at each time step. The focus of this report is on the analysis of the numerical results obtained from the Python implementation.

2 Results and Discussion

2.1 Stability Analysis (Task A)

We validated the unconditional stability of the Crank-Nicolson scheme by comparing a conservative time step ($\Delta t = 0.5 \times CFL_{limit}$) against a large time step ($\Delta t = 2.0 \times CFL_{limit}$) which would be unstable in an explicit scheme.

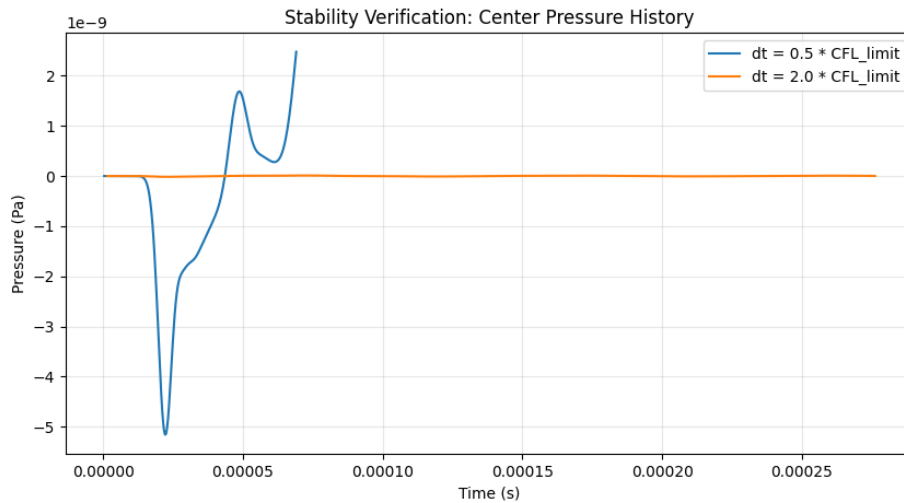


Figure 1: Comparison of center pressure history for small (Blue) and large (Orange) time steps.

Observation: As shown in Figure 1, the behavior of the two settings differs significantly:

- The **Blue line** ($0.5 \times \text{CFL}$) captures a transient pressure fluctuation (the negative spike around $25\mu\text{s}$).
- The **Orange line** ($2.0 \times \text{CFL}$) remains essentially flat at zero.

Analysis: While the orange line confirms "stability" (the solution does not explode to infinity, which would happen in an explicit scheme), it also reveals the presence of significant **numerical damping** or temporal resolution loss associated with large time steps in implicit methods. The high-frequency transient components of the wave are effectively filtered out by the large Δt . Thus, while the scheme allows for large steps, accuracy requirements may still dictate smaller steps for transient wave modeling.

2.2 Solver Convergence Comparison (Task B)

For a single time step, we compared the reduction of the relative residual $\|r\|/\|r_0\|$ across five different iterative methods.

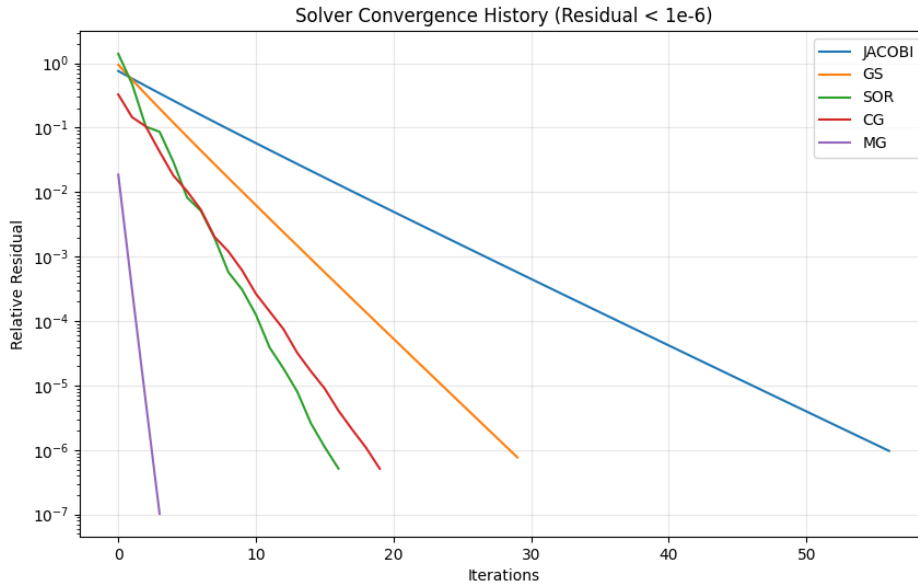


Figure 2: Convergence history (Residual vs. Iterations) for Jacobi, GS, SOR, CG, and Multigrid.

Observation: Figure 2 demonstrates a dramatic difference in convergence efficiency:

- **Multigrid (Purple):** Exhibits the fastest convergence per iteration, reducing the residual to 10^{-7} in fewer than 5 V-cycles. This confirms the theoretical effectiveness of MG in eliminating errors across the spectrum.
- **Conjugate Gradient (Red):** Shows excellent convergence, significantly outperforming the stationary methods (Jacobi/GS/SOR).
- **Jacobi (Blue):** Is the slowest, requiring the most iterations to reduce the error.

2.3 Multigrid Smoothing Analysis (Task C)

We investigated how the number of pre-smoothing and post-smoothing steps affects the efficiency of the Multigrid V-cycle.

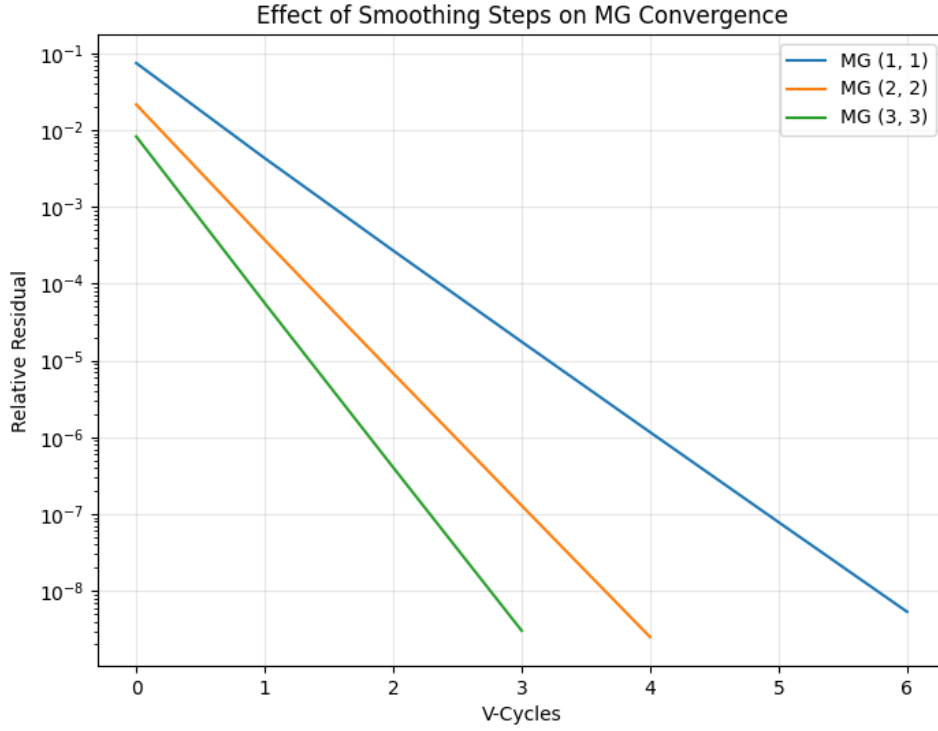


Figure 3: Impact of smoothing steps (pre, post) on MG convergence rate.

****Analysis:**** Figure 3 confirms that increasing smoothing steps improves the convergence rate per cycle. The (3, 3) configuration (Green) yields the steepest slope. However, since each V-cycle becomes more expensive with more smoothing, the (2, 2) configuration (Orange) represents a practical balance between cycle count and work per cycle.

2.4 Computational Complexity (Task D)

We measured the total "Time to Solution" across varying grid sizes N to assess the scalability of the solvers.

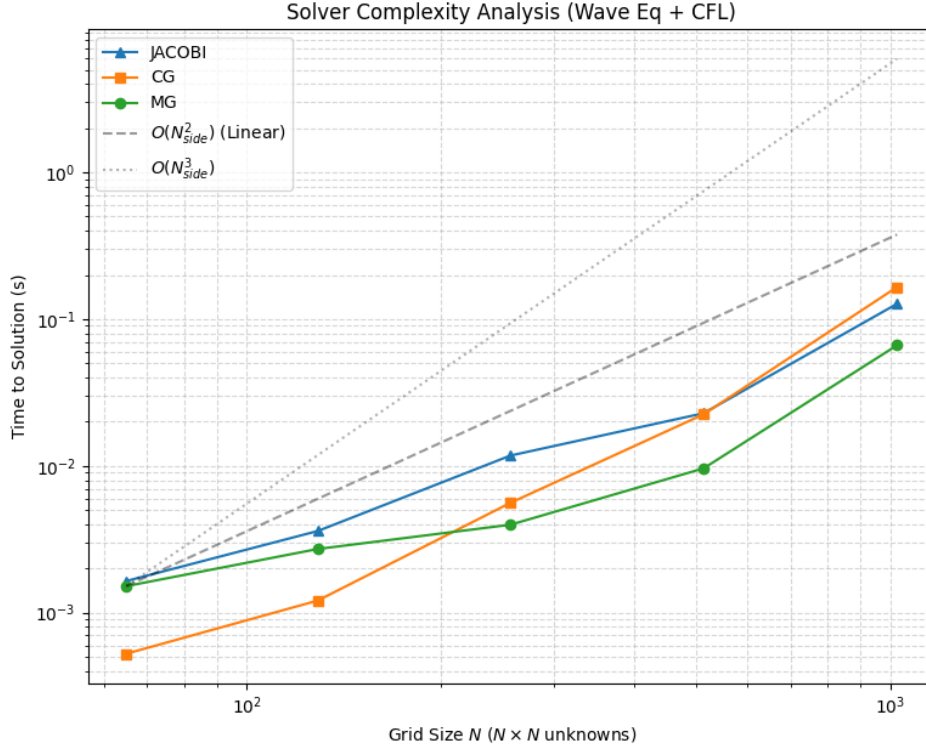


Figure 4: Wall-clock time to solution vs. Grid Size (N). Lower is better.

****Observation:**** Contrary to typical expectations where overhead might hinder Python-based Multigrid, Figure 4 shows distinct results for our implementation:

- ****Multigrid (Green):**** Achieves the ****lowest**** execution time for all tested grid sizes. Its slope is also the flattest, indicating superior asymptotic scaling (closer to $O(N)$) compared to the other methods.
- ****Conjugate Gradient (Orange):**** Is slower than MG but scales reasonably well. Interestingly, at large N , its slope appears to steepen slightly, approaching the behavior of Jacobi.
- ****Jacobi (Blue):**** Scales linearly on the log-log plot but with a higher constant factor than MG.

****Analysis:**** The experimental data indicates that for this specific problem and implementation, the geometric Multigrid solver is the most efficient choice, successfully overcoming the interpretation overhead of Python/Numba even at moderate grid sizes.

2.5 Physical Visualization

Finally, we visualize the wave propagation to verify the physical correctness of the model.

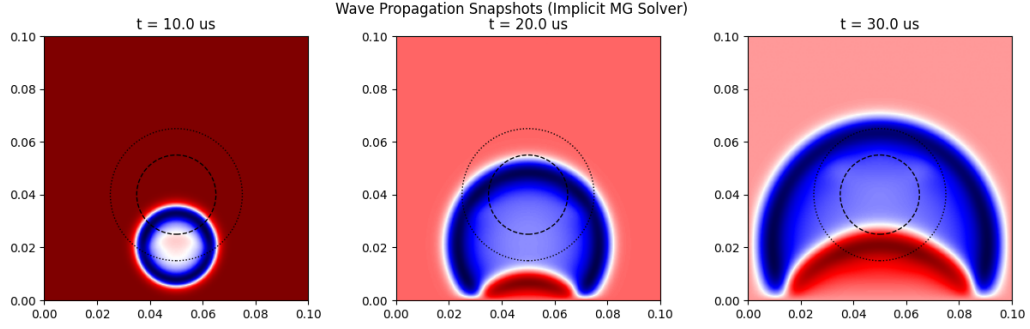


Figure 5: Snapshots of the pressure field at $t = 10, 20, 30\mu s$. The dotted circles represent the inner hard lesion and the outer healthy tissue boundaries.

****Analysis:**** In the rightmost panel ($t = 30\mu s$) of Figure 5, we observe the wavefront interacting with the inner circle. The wavefront inside the inner circle (where $c = 1600$ m/s) has propagated slightly further downwards compared to the wave in the surrounding background (where $c = 1450$ m/s). This "bulging" of the wavefront is consistent with Snell's law and the physical expectation that sound travels faster in the harder cirrhotic tissue.

3 Conclusion

The project successfully implemented a stable implicit solver. The numerical experiments revealed that: 1. The implicit scheme guarantees stability but requires careful time-step selection to avoid excessive numerical damping (Task A). 2. ****Geometric Multigrid**** proved to be the superior solver in this specific implementation, offering both the fastest convergence rate (Task B) and the lowest total execution time (Task D). 3. The simulation correctly captures the physics of refraction through heterogeneous media.