FINAL PROJECT REPORT

ADVANCED SCIENTIFIC COMPUTING (CPS 6320)

June 17, 2025

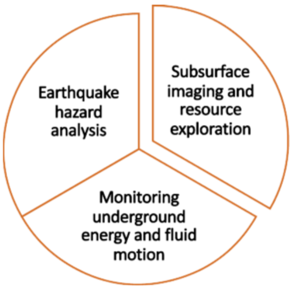
Submitted By
Solayman Hossain Emon
Mustafizur Rahman
Joy Saha

Contents

Seismic waves modeling & analysis using WaveQLab3D

1 Project Overview

This project focuses on the numerical simulation and analysis of seismic wave propagation through using the WaveQLab3D platform. We employ a first-order velocity-stress formulation of the elastic wave equation, which is foundational in seismic modeling and geophysical applications. Simulating seismic wavefields over 3D domains. Visualizing wave propagation patterns and validating the physical consistency of results.



Conservation of momentum:

$$\boxed{\begin{aligned}
\rho\frac{\partial v_x}{\partial t} &= \frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{xy}}{\partial y} + \frac{\partial \sigma_{xz}}{\partial z}\\
\rho\frac{\partial v_y}{\partial t} &= \frac{\partial \sigma_{xy}}{\partial x} + \frac{\partial \sigma_{yy}}{\partial y} + \frac{\partial \sigma_{yz}}{\partial z}\\
\rho\frac{\partial v_z}{\partial t} &= \frac{\partial \sigma_{xz}}{\partial x} + \frac{\partial \sigma_{yz}}{\partial y} + \frac{\partial \sigma_{zz}}{\partial z}
\end{aligned}}$$

Let $(x, y, z) \in \Omega$ denote the spatial variables, $t > 0$ be the time variable, $\sigma := (\sigma_{xx}, \sigma_{yy}, \sigma_{zz}, \sigma_{xy}, \sigma_{xz}, \sigma_{yz})^T$ be the vector of stresses, and $\mathbf{v} := (v_x, v_y, v_z)^T$ be the particle velocities. Throughout, we assume that the velocities and stresses are bounded functions of space and time.

Time derivative of Hooke's law in 3D:

$$\begin{pmatrix} \frac{\partial \sigma_{xx}}{\partial t} \\ \frac{\partial \sigma_{yy}}{\partial t} \\ \frac{\partial \sigma_{zz}}{\partial t} \\ \frac{\partial \sigma_{xy}}{\partial t} \\ \frac{\partial \sigma_{xz}}{\partial t} \\ \frac{\partial \sigma_{yz}}{\partial t} \end{pmatrix} = \begin{pmatrix} \frac{\partial v_x}{\partial x} \\ \frac{\partial v_y}{\partial y} \\ \frac{\partial v_z}{\partial z} \\ \frac{\partial v_x}{\partial y} + \frac{\partial v_y}{\partial x} \\ \frac{\partial v_x}{\partial z} + \frac{\partial v_z}{\partial x} \\ \frac{\partial v_y}{\partial z} + \frac{\partial v_z}{\partial y} \end{pmatrix}$$

Skew-symmetric form of the equation:

$$P^{-1}\frac{\partial Q}{\partial t} = \nabla \cdot F(Q) + \sum_{\xi \in \{x,y,z\}} B_\xi(\nabla Q)$$

where $Q := (\mathbf{v}, \sigma)^T$, $F(\mathbf{Q})$ and $\mathbf{B}_\xi(\nabla \mathbf{Q})$ are the flux and non-conservative terms respectively.

$$P = \begin{pmatrix} \rho^{-1} & 0^T \\ 0 & C \end{pmatrix}, \quad S^{-1} := C := \begin{pmatrix} 2\mu+\lambda & \lambda & \lambda & 0 & 0 & 0 \\ \lambda & 2\mu+\lambda & \lambda & 0 & 0 & 0 \\ \lambda & \lambda & 2\mu+\lambda & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu & 0 & 0 \\ 0 & 0 & 0 & 0 & \mu & 0 \\ 0 & 0 & 0 & 0 & 0 & \mu \end{pmatrix}$$

There are three parameters –

- ρ = density of the medium
- μ = shear resistance of the medium
- λ = first Lamé parameter

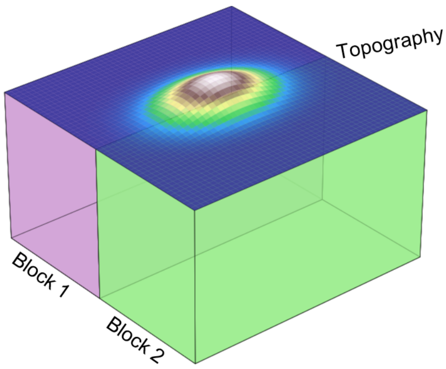


Figure 1: Overview of the considered seismic framework

2 Computational Tools

2.1 WaveQLab3D

For modeling and analyzing 3D seismic waves, we utilized WaveQLab3D computational tool which is designed for simulating 3D seismic wave propagation and earthquake rupture dynamics. It has the capability of solving the elastic wave equation in curvilinear coordinates, enabling modeling of complex geometries, and includes a possibly nonplanar frictional fault interface. The current version of WaveQLab3D implementation could be found here: https://bitbucket.org/ericmdunham/waveqlab3d/src/master/

2.2 Programming Languages

- **Python:** Process the large-scaled data
- **Matlab:** Visualization

2.3 HPC infrastructure

For high-scale parallel computations, we utilized UTEP's HPC cluster Jakar. Jakar is the high performance computing (hpc) environment and available for all purposes of computational processing. The system is expanding as more requirements for hpc resources arise. UTEP's HPC cluster Jakar has around CPU cores: ~ 400. However, we have only accessed for **educ** partition with limited number of cores particularly for this project.

3 Outcomes

3.1 High Scaled Simulation

We abled to produced output from Jakar after running around 8 hrs. After producing the desired outcomes from HPC Cluster, we extract physical coordinates based on logical grid indices.

- **Logical Indices**: i_x, i_y, i_z, representing the grid's logical indices in the x, y, and z directions.

- **Grid Dimensions**: n_x, n_y, n_z representing the number of grid points in the x, y, and z-directions.

- **Velocity Data**: The velocity data containing time steps and corresponding velocity components v_x, v_y, and v_z.

The physical coordinates of a grid point corresponding to logical indices (i_x, i_y, i_z) are computed as:

$$\boxed{\begin{aligned} x &= a_x + (i_x - 1) \cdot \Delta x \\ y &= a_y + (i_y - 1) \cdot \Delta y \\ z &= a_z + (i_z - 1) \cdot \Delta z \end{aligned}}$$

The spacing $\Delta x, \Delta y, \Delta z$ between grid points in each direction is computed using the following formulas:

$$\Delta x = \frac{b_x - a_x}{n_x - 1}, \quad \Delta y = \frac{b_y - a_y}{n_y - 1}, \quad \Delta z = \frac{b_z - a_z}{n_z - 1}$$

The velocity components v_x, v_y, and v_z are extracted from the files. The magnitude of the velocity vector at each time step is computed as:

$$|\mathbf{v}| = \sqrt{v_x^2 + v_y^2 + v_z^2}$$

Where: v_x, v_y, v_z are the velocity components in the x, y, and z-directions, respectively.

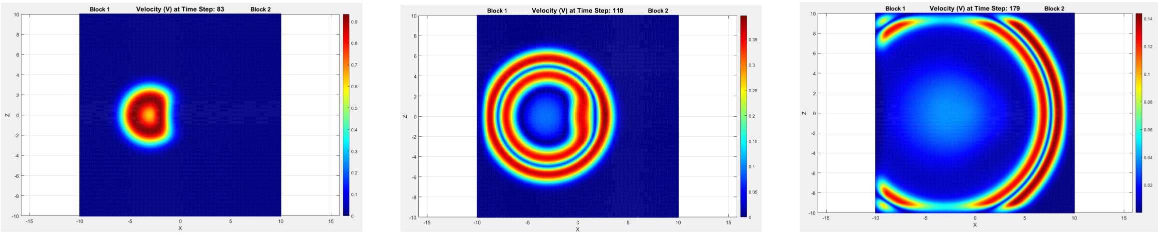


Figure 2: Visualization of velocity (V) at different times steps on X-Z system

The acceleration is calculated by differentiating the magnitude of the velocity with respect to time:

$$a(t) = \frac{\Delta|\mathbf{v}|}{\Delta t}$$

Where: $\mathbf{v}$ is the velocity vector. Δt is the time step between consecutive velocity measurements.

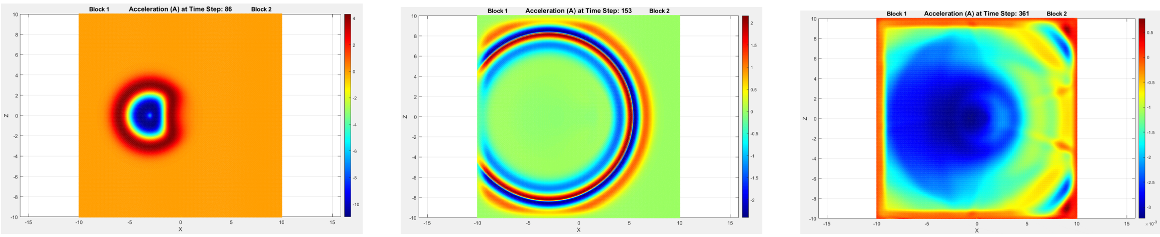


Figure 3: Visualization of Acceleration (A) at different times steps on X-Z system

The displacement is calculated by integrating the velocity over time using the cumulative summation method, which approximates the integral by the trapezoidal rule:

$$d(t) = \sum_{t_1}^{t_n} |\mathbf{v}(t)| \cdot \Delta t$$

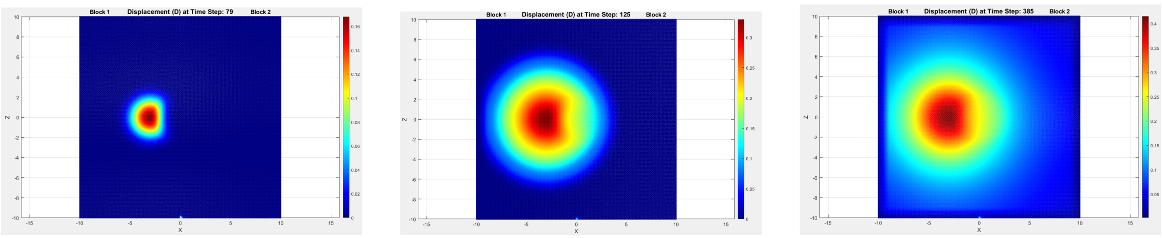
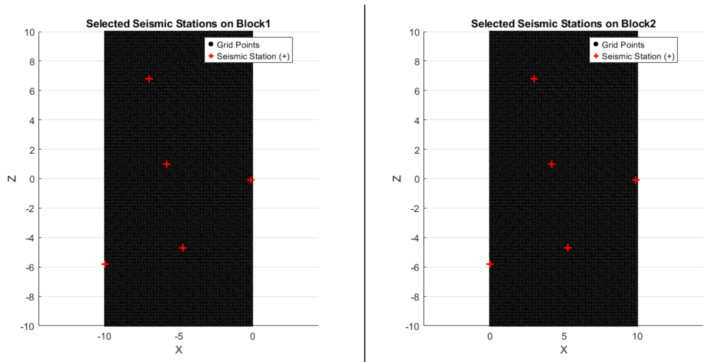


Figure 4: Visualization of displacement (D) at different times steps on X-Z system



(a) Block 1 (b) Block 2

Figure 5: Selected seismic stations from two blocks

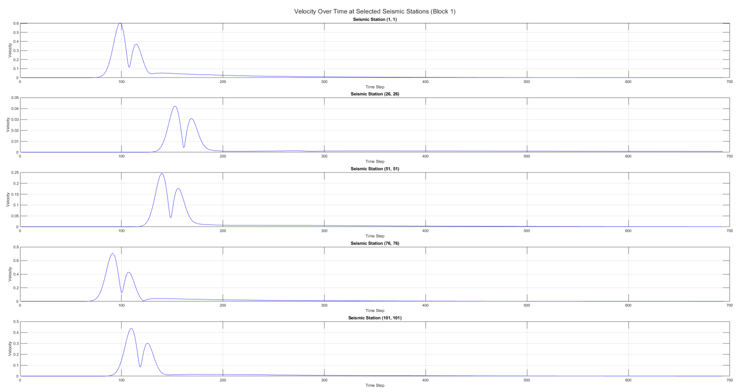


Figure 6: Velocity analysis of seismic stations from block 1

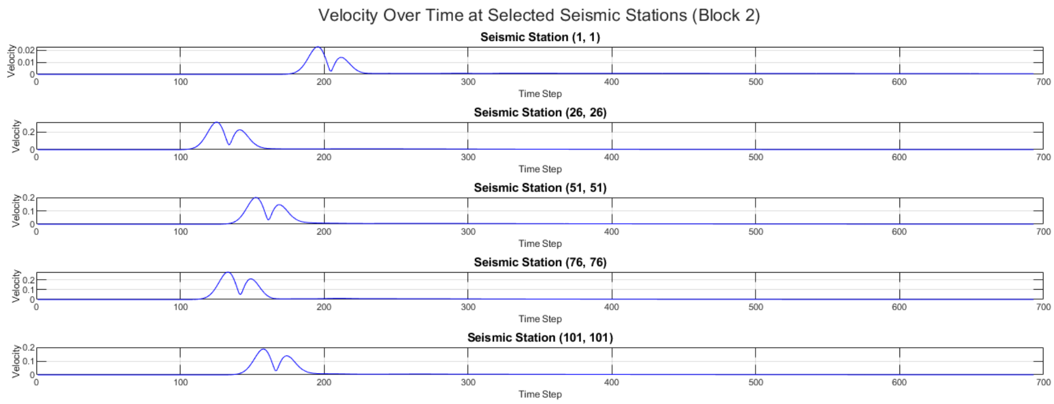


Figure 7: Velocity analysis of seismic stations from block 2

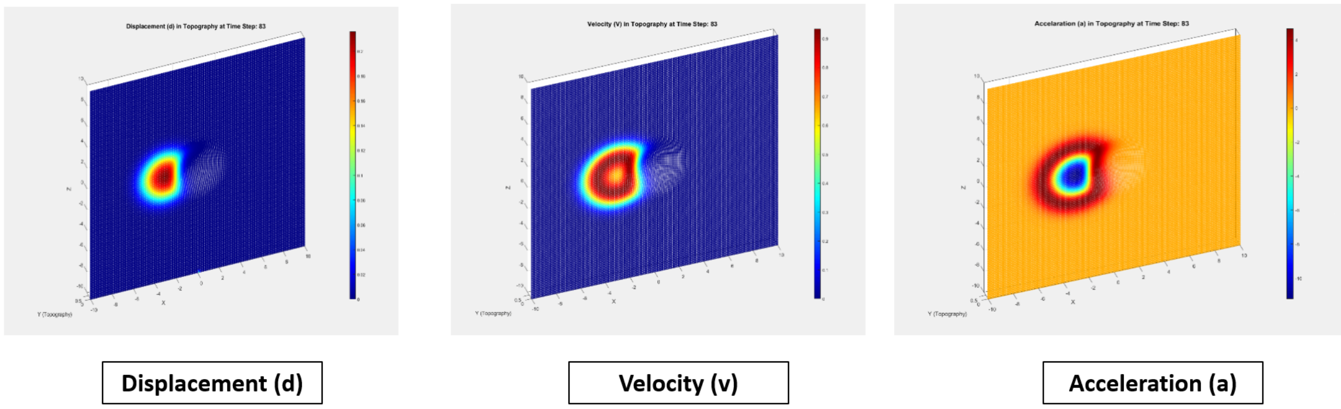


Figure 8: Simulations on Topography

4 Conclusion

4.1 Insights

- **Monitor Seismic Activities:** we were able to monitor seismic activity in real-time across different seismic stations.
- **Real-time Alert System:** By analyzing wave velocity, displacement, and acceleration data from these stations, we can implement an micro-level analysis that leads to better earthquake alert system.

4.2 Future Directions

- **Extend the topograhpy analysis on more realistic scenarios**
- **Extend the analysis for stress variables** (σ)
- **Dynamic earthquake rupture simulations**

5 References

[1] Duru, K., & Dunham, E. M. (2016). Dynamic earthquake rupture simulations on nonplanar faults embedded in 3D geometrically complex, heterogeneous elastic solids. Journal of Computational Physics, 305, 185-207.

[2] Duru, K., Fung, F., & Williams, C. (2022). Dual-pairing summation by parts finite difference methods for large scale elastic wave simulations in 3D complex geometries. Journal of Computational Physics, 454, 110966.