Mimetic Finite Difference Modeling Of 2D Elastic P-SV Wave Propagation

Otilio J. Rojas

February 2007

Publication Number: CSRCR2007-05

Computational Science & Engineering Faculty and Students Research Articles

Database Powered by the Computational Science Research Center Computing Group

COMPUTATIONAL SCIENCE & ENGINEERING

SAN DIEGO STATE UNIVERSITY

Computational Science Research Center
College of Sciences
5500 Campanile Drive
San Diego, CA 92182-1245
(619) 594-3430

MIMETIC FINITE DIFFERENCE MODELING OF 2D ELASTIC P-SV WAVE PROPAGATION.

Qualifying Exam Report
by

Otilio J. Rojas

(PhD student in Computational Sciences,
San Diego State University – Claremont Graduate University)

April 2006

Abstract

Recently, efforts have been made by Kristek et al.(2002) toward the implementation of a zero-traction boundary condition for an elastic medium by using high-order staggered-grid finite-difference modeling and avoiding symmetry conditions, vacuum formulations, or, other approaches that require grid points located beyond the physical boundary (ghost points). In this work, a new set of numerical differentiators known as "mimetic" finite differences have been used to explicitly solve the exact boundary conditions and fully compute the displacement vector along a planar free surface of a 2D half space. No ghost points are used in our schemes. Two classical grids, the rotated staggered grid (RSG) and the standard staggered grid (SSG), have been enhanced by the inclusion of Compound nodes along the free surface boundary to allow this discretization process and place the displacement vector. Thus, three new algorithms are proposed here, one that works over a RSG, and two implemented using a SSG. Accuracy of these solvers is measured in terms of the dispersion of the numerical Rayleigh wave, and comparisons against Kristek et al.'s algorithm are presented.

1. Introduction

Because of the large size of domains where elastic or acoustic wave propagation must be modeled, explicit finite difference discretizations are still efficient numerical methods for real seismic modeling, even when geometry is treated as regular (Olsen et al., 2006). In particular, actual topography is idealized as planar and a flat free surface boundary condition is used to model the interaction of body waves at this interface. The implementation of the zero-traction conditions is tied up to the type of grid used in the discretization of the medium and wave-stress fields, because the appropriate stress-tensor components must be zeroed at this boundary, independently of their grid location. Thus, the type of grid (cartesian, rotated, nodal, staggered) and the alignment of the free surface with a specific grid-line (2D) or grid-plane (3D) strongly influence the way as planar free surfaces are numerically treated. Thus, the study of these numerical treatments demands reviewing the computational grid underneath.

The most popular grid for finite-difference modeling of seismic wave propagation is the staggered one used by Madariaga (1976), Virieux (1984, 1986), and, Levander (1988) to solve velocity-stress formulations of the 2D first-order elastodynamics equations of motion. Later on, this grid was utilized in 3D applications by multiple authors becoming standard in seismic modeling (Olsen, 1994; Graves, 1996; Pitarka, 1999). In this article, we refer this grid as the *standard staggered grid* (SSG). Because of the staggered distributions of the stress-strain tensor components different implementations of the zero-traction conditions are known (vacuum, stress-imaging, adjusted finite differences formulations). Alternatively, Saenger et al. (2000) employed a *rotated staggered grid* (RSG) to simulate wave propagation in 2D elastic media containing cracks and free surfaces. Based on the fact that stress tensor is placed in just one grid point, such discontinuities are implicitly treated by setting up density and elastic parameters to appropriate values, instead of performing an explicit discretization of contact-boundary conditions (vacuum formulation).

In the classical vacuum formulism, a thin "vacuum" layer is created above this boundary by zeroing Lame parameters (λ,μ) and choosing the density (ρ) value from the range (0, ρ_S], for ρ_S the density of the solid layer underneath. Only second order discretizations of the equation of motion are used to compute locally the wave field, due to higher-order approximations become unstable in the case of a SSG (Graves, 1996), or, strongly fluctuating for RSG implementations (Saenger and Bohlen, 2004). Initial approaches of this formulation were developed on a SSG, but its inaccuracy was proved several times leading to their obsoleteness (Randall, 1989; Zahradnik, 1993; Graves, 1996). For instance, Graves (1996) presents an experiment where a homogeneous half-space is discretized using a SSG and employing a 500m-deep source to excite surface waves. His conclusion is that a low dispersive propagation of the Rayleigh pulse is observed at a horizontal distance of $9\lambda^{MIN}_S$ when 9.2 points per λ^{MIN}_S are employed, but the amplitude is not satisfactory modeled. On the other hand, recently new elastic and viscoelastic solvers built on a RSG have incorporated this technique because of zero-traction conditions could be implicitly imposed on a general topography just by defining the vacuum layer adequately (Saenger et al., 2000; Bohlen and Saenger et al., 2003). Saenger and Bohlen (2004) state that an accurate modeling of the Rayleigh wave propagation excited in a Lamb's problem requires 30 points per P-wavelength in the case of a RSG. Gelis et al. 2005 followed this criterion to precisely model surface waves over shallow cavities.

More accurate free surface implementations on a SSG are the stress imaging formulations introduced in 2D by Levander (1989) and later adopted by some 3D velocity-stress schemes (for instance, Olsen, 1994; Graves, 1996; Pitarka, 1999). Also, its performance in 3D displacement-stress

formulation has been lately researched (Kristek, et al.2002). Due to the staggered distribution of the wave and stress fields on the grid, two different implementations of the zero-traction conditions are possible. Either, the free surface is placed along the grid plane going through normal stresses and horizontal velocity-displacement, or, it shares the same grid plane of (*xz*, *yz)* stress tensor components and (vertical, transversal) velocities-displacements. Kristek et al.2002 baptized the first alternative as *H-formulation*, while *W-formulation* was the name given to the second one. Hereafter, we adopt this nomination. Gottschammer and Olsen (2001) coded velocity-stress versions of both formulations and used a Gaussian-shaped source buried 2 km deep into an uniform halfspace and approximately 6 points per shear wavelength were allowed by the grid spacing. They concluded that the *W-formulation* produces more precise wave fields along the free surface and models more accurately the Rayleigh wave propagation, after the horizontal velocity is interpolated by simply averaging. Kristek et al. (2002) presented other important comparative study. In this case, a Gabor wavelet was employed to excite strong surface waves on a homogeneous halfspace and separate tests were performed for Poisson ratios 0.25 and 0.4. In these experiments, they found that both formulations require a minimum of 10 points per λ^{MIN}_S (λ_S, S-wave wavelength) to propagate Rayleigh waves with low grid dispersion along epicentral distances up to 15 λ^{DOM}_S, and these results are practically independent of the Poisson ratio. In addition, amplitude and speed of the numerical Rayleigh wave vary between each formulation.

As far as we know, the most recent finite difference implementation of planar free surfaces is the *adjusted finite difference formulation* due to Kristek et al. (2002). Similarly to vacuum and stress-imaging techniques, the wave field is calculated locally to this boundary using discrete versions of equations of motion. However, *z*-derivates of the wave and stress fields required in this calculation are obtained by combining the traction-free conditions with one-sided fourth-order staggered finite differences, which avoids any fictitious gridlines above the free surface. This innovative aspect makes this formulation memory-saving and very accurate. After testing both *H* and *W* formulations in the same experiment detailed above, Kristek et al. stated than 6 grid spacing per λ^{MIN}_S are sufficient to accurately model Rayleigh wave propagation in the range of epicentral distances up to 15 λ^{DOM}_S. Phases of the numerical waves are better reproduced by the *H-formulation*, while *W-formulation* yield better amplitudes. In case that all displacement components are required at the free surface, authors suggest using 4^{th}-order extrapolation for the displacement component(s) missing at the free surface in each formulation.

In this work, we propose four staggered-grid solvers of the elastodynamic wave equations in 2D media that include a flat free surface boundary. The full displacement vector is obtained at this boundary and fictitious gridlines are avoided. First at all, both grids the RSG and the SSG have been enhanced by the inclusion of a new set of *compound nodes* (displacement-stress nodes) along the free surface allowing the one-sided discretization of the zero-stress conditions and the computation of the displacement vector. The set of "mimetic" finite-difference operators proposed by Castillo and Grone (2003) has become convenient for this purpose, because it comprises one-sided and centered staggered differentiators with multiple order of accuracy. In section 4, a novel elastic solver is built on a RSG by using second order numerical differentiation along the free surface and fourth-order centered formulas along the rotated axis at the domain's interior. Next in section 5, fourth-order mimetic differentiators (one-sided and centered) are used on a SSG to design two new elastic solvers. Finally, section 6 describes our last fourth-order algorithm designed on staggered grid previously used in solving diffusion phenomena (Shaskov et al. 1996, Hyman et al. 2002), and Maxwell's equations (Hymann et al. 1999).

The efficacy of algorithms mentioned above has been researched through a stringent test presented in the last section of this article. Qualitative comparisons of numerical time series against the exact solution of the classical Lamb's problem are presented, and accuracy is quantitatively determined through the minimum number of points per λ_R (λ_R, Rayleigh wavelength) required per each solver to keep the dispersion error under 1% and 5%. Here, a new procedure to isolate the Rayleigh pulse from surface waves is proposed and used in this dispersion analysis.

2. Mimetic finite difference operators.

Traditional finite difference methods used in the solution of a system of partial differential equations that model physical processes usually present two important drawbacks. The discrete scheme might has little connection with the underlying physical principles of the original problem because it was stated just by discretizating every differential expression present in the system. Thus, important symmetry properties and conservation laws satisfied by the continuous solutions are not fulfilled by the discrete solution, and numerical artifacts could degrade or completely ruin its usefulness (loss of reciprocity, instabilities, and non-conservative solutions). Another deficiency is related with the low order of accuracy used in the numerical treatment of boundary conditions with respect to the discretization of the interior fields, even in regular geometries. Outer-domain points label as fictitious or ghost points are used to allow the simultaneous solution of the interior equations and the boundary conditions, regardless whether they are both valid at the extended boundary nodes.

In a effort to overcome these deficiencies, while keeping simplicity and computational efficiency, a new family of "conservative" numerical differentiators have been introduced and applied to the solution of different models such as Maxwell's equations (Hymann and Shashkov, 1999), Diffusion processes (Shaskov and Steinberg, 1996, Hyman et al., 2002), and, elliptic models (Shaskov and Steinberg, 2003, Castillo and Yasuda, 2005). The main idea is construct discrete divergence(D), gradient(G), and curl(C), that preserve fundamental identities and theorems of vector and tensor calculus, satisfied by the original continuum operators *divergence*, *gradient*, and *curl*. Particularly, the definition of a discrete analog of the Green-Stokes theorem and its fulfillment by operators D and G has become a key goal in the construction. Is it well known that stability and symmetry properties of the continuous solution to many PDEs are due to Stoke's theorem (J. Castillo et. al, 2001). In consequence, discrete solutions obtained by using these new "conservative" or "mimetic" operators should be faithful and physically consistent. Moreover, fourth and sixth order D and G operators have been proposed by Castillo and Grone (2003) with the elegant property that the high-order of accuracy is given by central stencils at the interior, and by one-sided differentiators at boundary points. This last condition ensures that the discretization of the boundary conditions is compatible in accuracy and physics to the interior computation by avoiding any outer-domain nodes. In this work, we use the set of mimetic operators G and D constructed by Castillo and Grone (2003).

To introduce the Castillo-Grone operators let us consider the one-dimensional domain $M = [0,1]$, and $f(x)$ and $v(x)$ as two smooth real-valued functions in M. The multi-dimensional form of the Stoke's theorem is reduced to integration by parts

$$\int_0^1 \frac{dv}{dx} f \, dx + \int_0^1 v \frac{df}{dx} dx = v(1) f(1) - v(0) f(0) \qquad (1)$$

In this context, dv/dx plays the role of the divergence of a vector field $v(x)$, and df/dx plays the role of the gradient of the scalar function $f(x)$. The above equation represents the physical conservation law to be satisfied by the discrete operators. A staggered grid is chosen to place evaluations of v and f. Basically, n equal-sized cells $[x_i, x_{i+1}]$, for $0 \le i \le n-1$ and $h = 1/n$, are defined on M by simply $x_i = i*h$. These points x_i are called nodes, and particularly, x_0 and x_n become the boundaries points. In addition, the cell centers are indexed with half-integer indexes, $x_{i+1/2} = (x_i + x_{i+1})/2$. Thus, nodes, cell centers, and, boundary points comprise the complete set of grid points used in the discretization process. Now, two vectors $\boldsymbol{v}$ and $\boldsymbol{f}$ are defined by using evaluations of functions v at the nodes and boundary points, while values of f are required at the cell centers and boundaries, i.e.,

$$\boldsymbol{v} = (v(x_0), v(x_1), v(x_2), \ldots, v(x_n))^T$$

$$\boldsymbol{f} = (f(x_0), f(x_{1/2}), f(x_{3/2}), \ldots, f(x_{n-1/2}), f(x_n))^T$$

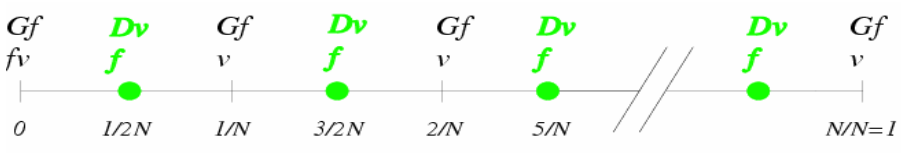


Figure 1: One-dimensional staggered grid for mimetic discretization. Locations of discrete fields f and v are depicted. Approximations of gradient of f and divergence of v are also displayed.

Figure 1 depicts the distribution of the components of vectors $\boldsymbol{v}$ and $\boldsymbol{f}$ along the staggered grid in use. In addition, approximations to the gradient of $f(x)$, G$\boldsymbol{f}$, and divergence of v(x), D$\boldsymbol{v}$, are also illustrated. In this sense, D becomes the mimetic divergence operator that map v-values given at the nodes and boundary points to v'-values located at the cell centers of the grid, and G maps f-evaluations at the cell centers and boundary points to f'-values defined at the nodes and boundary points of the grid (Castillo and Yasuda, 2005). Values $v'(x^*)$ and $f'(x^*)$ denote approximations to $dv(x^*)/dx$ and $df(x^*)/dx$, respectively. As a result, both numerical differentiators D and G, are linear operators expressed as real matrices in (Castillo and Grone, 2003), where D is a $n \times (n+1)$ matrix, and G is a $(n+1) \times (n+2)$ matrix.

The construction methodology for D and G is based on the following discrete version of the conservation law (1),

$$\left\langle \hat{D}\boldsymbol{v}, \boldsymbol{f} \right\rangle_Q + \left\langle G^T \boldsymbol{f}, \boldsymbol{v} \right\rangle_P = \left\langle B\boldsymbol{v}, \boldsymbol{f} \right\rangle \qquad (2)$$

For,

$$\hat{D} = \begin{bmatrix} \boldsymbol{0} \\ D \\ \boldsymbol{0} \end{bmatrix}; \quad B = QD + (PG)^T$$

Basically, matrix $\hat{D}$ is a $(n+2)x(n+1)$ augmented version of D with zeros as its first and last rows with the goal of matching the dimensions of $\boldsymbol{f}$. Weighted inner products were introduced in the computation of the volume integrals by using positive definite matrices Q and P. D and G satisfy the discrete analog of the divergence theorem (2) with respect to the generalized inner products determined by Q and P. Matrix B embodies the global conservation law, and incorporates the sign of the exterior normal vector at the boundaries of M. Operators constructed by Castillo and Grone (2003) are given in appendix 1.

3. Problem Formulation.

In this work, the classical PS-V problem is chosen due to its several interesting features that include dispersionless propagation of Rayleigh pulses in homogeneous media with a known analytical solution. Now, consider an infinite elastic and isotropic half-plane with a horizontal axis x and vertical axis z positive downward. The propagation of elastic waves is described by the elastodynamic wave equations

$$\begin{aligned} \rho \frac{\partial^2 u}{\partial t^2} &= \frac{\partial}{\partial x}[\tau_{xx}] + \frac{\partial}{\partial z}[\tau_{xz}] \\ \rho \frac{\partial^2 w}{\partial t^2} &= \frac{\partial}{\partial x}[\tau_{xz}] + \frac{\partial}{\partial z}[\tau_{zz}] \end{aligned} \tag{3}$$

$$\begin{aligned} \tau_{xx} &= (\lambda + 2\mu)\frac{\partial u}{\partial x} + \lambda \frac{\partial w}{\partial z} \\ \tau_{zz} &= (\lambda + 2\mu)\frac{\partial w}{\partial z} + \lambda \frac{\partial u}{\partial x} \\ \tau_{xz} &= \mu\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right) \end{aligned} \tag{4}$$

In these equations, the variables are the particle displacement vector (u,w), and the symmetric stress tensor components: τ_{xx}, τ_{xz}, τ_{zz}. Equation (4) represents the linear stress-strain relationship for an elastic media also known as Hooke's law. The density $\rho(x,z)$ and Lame parameters $\lambda(x,z)$, and $\mu(x,z)$ constitute the parameters of this model which determine the velocity of the compressional (P) wave velocity α, and the shear (S) wave velocity β through the relations,

$$\alpha = \sqrt{\frac{\lambda + 2\mu}{\rho}} \quad , \qquad \beta = \sqrt{\frac{\mu}{\rho}}$$

A Neumann-type boundary condition is imposed at $z = 0$ to model a free-surface by vanishing the normal and tangential stresses $\tau_{xz} = \tau_{zz} = 0$. In this work, none radiation boundary conditions have been considered along the other three edges of the medium, and the simulation time is adjusted in order to avoid boundary reflections.

An explosive point given by the gaussian pulse $f(t) = exp(-\delta(t-t_0)^2)$ is used in our experiments because of its simple implementation. In the case, a source located at an interior point $f(t)$ is added to both normal stresses, τ_{xx}, and τ_{zz}, while a vertical point force applied at the free surface is modeled by incrementing only τ_{zz} by $f(t)$ (Virieux, 1986). In both staggered grids used in this work, SSG and RSG, normal stresses τ_{xx}, τ_{zz} are defined at the same grid point, implying an unique way to update the stress field due to source contribution.

4. Vacuum formulism over a rotated staggered grid: VRSG algorithm.

The application of a rotated staggered grid to solve the elastic wave equations and model viscoelastic wave propagation has lately increasing been applied to complex problems (Sanger et. al. 2000, Bohlen et. al., 2003, Sanger et. al., 2004, Celis et al. 2004). In 2D, and using a grid with steps *(Δx,Δz)*, this technique introduces a new set of coordinate-axis $(\tilde{x},\tilde{z})$ by rotating the cartesian axis in $tan^{-1}(\Delta z/\Delta x)$ degrees, which implies that $(\tilde{x},\tilde{z})$ coincides with diagonals of the grid cells. The directions of spatial differentiation are aligned with $(\tilde{x},\tilde{z})$. As a result, every physical quantity is distributed on the grid in a proper point, i.e., displacements are placed at the same grid point *(displacement node)*, similarly stresses are assigned to the same location *(stress node)* diagonally half-way from a *displacement node*, and density and Lame parameters are placed at every displacement node, and, stress node, respectively. This elegant arrangement is shown in figure 3. It is claimed, that this distribution of wavefield and material properties allows an accurate and stable modeling of strong viscoelastic heterogeneities, including free surfaces (Bohlen et. al., 2003).

Here, we have implemented a 2D version of the standard rotated-staggered finite-difference elastic solver according to Sanger (Sanger et. al., 2000, Sanger et. al., 2004), and, Gelis (Gelis et al., 2004). A flat free surface passes through the first line of stress nodes with null values of Lame parameters, and the line of displacement nodes above is also zeroed. Thus, by simulating a vacuum on and above the free surface, the zero-traction condition is implicitly imposed. Notice that $\tau_{xx} = 0$ along the free surface, which is not physically correct, and it could affect the accuracy of this approach. Numerical differentiators are applied along diagonals of the cells to approximate derivatives with respect to the rotated coordinates $(\tilde{x},\tilde{z})$: $\partial/\partial\tilde{x}$, $\partial/\partial\tilde{z}$. Then, these approximations are transformed into the original system (x,z) to solve (3), which always remains in (x,z) coordinates. This transformation is given by:

$$\begin{aligned} \tilde{z} &= \frac{\Delta x}{\Delta r}x + \frac{\Delta z}{\Delta r}z \\ \tilde{x} &= \frac{\Delta x}{\Delta r}x - \frac{\Delta z}{\Delta r}z \\ \Delta r^2 &= \Delta x^2 + \Delta z^2 \end{aligned} \quad \Rightarrow \quad \begin{aligned} \frac{\partial}{\partial z} &= \frac{\Delta r}{2\Delta z}\left(\frac{\partial}{\partial \tilde{z}} - \frac{\partial}{\partial \tilde{x}}\right) \\ \frac{\partial}{\partial x} &= \frac{\Delta r}{2\Delta x}\left(\frac{\partial}{\partial \tilde{z}} + \frac{\partial}{\partial \tilde{x}}\right) \end{aligned} \tag{5}$$

The free surface synthetic seismograms are given at the displacement nodes half-way below of the zero-stress line. In our experiments, we verify that using one-sided 4th order finite differences at these displacement nodes numerical instabilities are triggered. Thus, a stable implementation is only reached by using 2nd order finite differences at this line of displacements. Both, 4th order and 2nd order, finite differences could be used along the first interior stress line and the simulation remains stable, and we chose the 4th order case, to implement our *rotated-staggered grid* solver with *vacuum* formulation (VRSG). Fourth order of accuracy is used elsewhere, and, second order centered-nodal finite differencing to discretize time derivates.

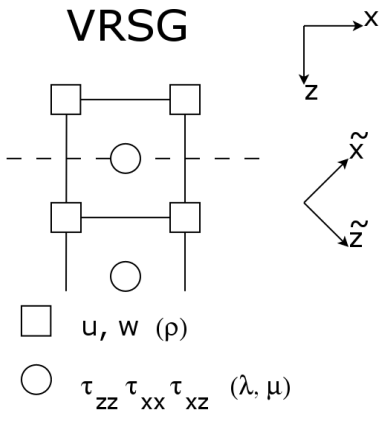


Figure 2. Rotated staggered grid with a free surface boundary condition implemented by vacuum formulition. Extra zero-velocity nodes are defined above the physical boundary.

Numerical stability and grid dispersion of RSG finite-difference schemes was studied by Saenger et al. (2000) for the case of an infinite, elastic, and, homogeneous medium. Different orders of spatial discretization and second order in time were considered, and they proposed equation (6) as the von Neumann stability condition for a (2n)th spatial accurate scheme. Scalars C_K are the coefficients of the stencil for central differentiation. Here, (4) is used to constraint the time step used by VRSG in our numerical simulations, even when a free surface boundary condition is implicitly implemented by a discontinuity in the Lame parameters values, which are zeroed at the first gridline of stresses. We observed stable behavior of VRSG even in cases of heterogeneous media, as it shown in section (??).

$$\frac{\Delta t\,\alpha}{h} \leq \frac{1}{\sqrt{2}\sum|C_k|} \qquad (6)$$

5. Adapted Rotated Staggered Grid for Mimetic Discretization: MRSG algorithm.

As it is shown in Figure 3, a RSG could be easily adapted to perform numerical differentiation along $(\tilde{x},\tilde{z})$ of the displacement and stress fields by using the mimetic operators given in section 2. Notice that along any of the two rotated axis, a RSG is just a one-dimensional staggered grid with two different types of nodes, displacement nodes, and, stress nodes. Respectively, a correspondence with divergence nodes and gradient nodes defined in the grid used in 1D mimetic discretization and depicted in figure 3 is clearly established. However, an extra line of *displacement-stress* nodes must be added along *z=0* to allow the application of one-sided differentiation to *(u,w)* and the explicit treatment of the zero-traction boundary condition. We have called *compound nodes* to these special boundary nodes. Figure 3 shows the adapted rotated staggered grid used here with the inclusion of compound nodes (filled squares).

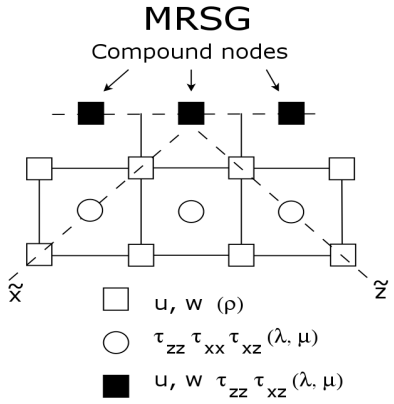


Figure 3. Adapted rotated staggered grid for mimetic discretization. Empty squares represent displacement nodes, circles become stress nodes, and filled squares are compound nodes introduced along the free surface. Density is defined at every interior displacement node, while Lame parameters are assumed available at every stress node.

A new solver for the PS-V model (1), (2) is introduced in this section. It uses mimetic differentiation over the adapted rotated staggered grid (MRSG) along $(\tilde{x},\tilde{z})$ and transform these approximations back to the standard system *(x,z)* by using (3). At every interior grid point being a displacement or a stress node, differentiation by using operators from section 1 yield 4^{th} order of accuracy. The novel aspect of MRSG relies in the implementation of the free-surface boundary condition. Let us introduce the following notation for displacement and compound nodes in a grid of *[0,Nx]x[0,Nz]* points, basically (x_i,z_j), $0 \le i \le Nx$, $1 \le j \le Nz$ correspond to a displacement node, while (x_i,z_0), $0 \le i \le Nx$, represent a compound node. For instance, at (x_i,z_0), $1 \le i \le Nx-1$, first partial derivates of u could be approximated by using $G_{2\text{-}4\text{-}2}$ (given in appendix 1) and (5),

$$u_z|_{0,j} = \frac{\Delta r}{2\Delta z}\left\{2\, g_{11}\, u_{0,j} + g_{12}\left(u_{1,j+1} + u_{1,j}\right) + g_{13}\left(u_{2,j+2} + u_{2,j-1}\right)\right\}$$
$$u_x|_{0,j} = \frac{\Delta r}{2\Delta x}\left\{g_{12}\left(u_{1,j+1} - u_{1,j}\right) + g_{13}\left(u_{2,j+2} - u_{2,j-1}\right)\right\}$$

Substitution of equations above and ones equivalent for w into the zero-traction condition $\tau_{xz} = \tau_{zz} = 0$, lead to a linear dependence of u_{i0}, and, w_{i0} on displacements defined at (x_i,z_j), $1 \le j$, i.e.,

$$u_{0j} = -\frac{1}{2g_{11}}\left\{g_{12}\left(u_{1,j+1} + u_{1,j}\right) + g_{13}\left(u_{2,j+2} + u_{2,j-1}\right) + \frac{2\Delta z}{\Delta r} w_x|_{0,j}\right\} \qquad (7)$$
$$w_{0j} = -\frac{1}{2g_{11}}\left\{g_{12}\left(w_{1,j+1} + w_{1,j}\right) + g_{13}\left(w_{2,j+2} + w_{2,j-1}\right) + \frac{2\Delta z}{\Delta r} \gamma u_x|_{0,j}\right\}$$

Where $\gamma = \lambda/(\lambda+2\mu)$. Now, we sketch a time iteration of the MRSG scheme in the following framework:

MRSG algorithm: Two-step time evolution.

STEP I (Interior): Solve (3) to get (u,w) for t = (n+1)Δt at every displacement node. Accuracy is 4^{th}-order in space and 2^{nd}-order in time.

STEP II (Free Surface): Compute (u,w) for t = (n+1) Δt at every compound node using equation (7) and interior wavefield from STEP I.

An important point is related with the influence of the half-way line of compound nodes on the stability property of the MRSG scheme, and how they compare against the standard rotated finite difference algorithm VRSG. Here, we tested the validity of the von Neumann condition (4) and empirically we found that it is still valid.

Equation (7) represents a simple 2^{nd} order implementation of a planar free surface boundary condition. Naturally, similar treatments were tried using higher order formulas by using $G_{4\text{-}4\text{-}4}$. However, exponential growing instabilities were seen in the discrete quantities u_{i0} and w_{i0} after few thousands of iterations, when Lamb's problem was treated. In these experiments, the time step was adjusted below the upper limit given by the condition (6), exhaustively. Thus, the only stable implementation of the MRSG scheme now available is based on differentiating by $G_{2\text{-}4\text{-}2}$, and, $D_{2\text{-}4\text{-}2}$.

6. Standard Staggered Grid Finite-Difference Solvers: H-AFDA, W-MSSG, and, H-MSSG.

In this section, we propose two new solvers for the elastic model (3)-(4) based on using a SSG. Two important principles have been followed in their design:

i) No outer-domain nodes are used above the free-surface and the zero traction conditions are explicitly imposed at this boundary.

ii) Fourth-order differencing along both directions (x,z) is used to approximate spatial derivates of the wave and stress fields along the free surface and gridlines nearby. Then, the equations of motions are used to calculate the displacements at this boundary.

To our knowledge, only two numerical schemes have been previously proposed that fulfill these two conditions. The adjusted finite difference approximations (AFDA) presented by Kristek et. al. (2002) showed an efficient performance by avoiding any ghost points and requiring of only 6 points per shear wavelength to accurately model Rayleigh wave propagation in the range of epicentral distances up to $15\lambda DOM_S$. Based on the location of the free surface along z-constant gridlines, two different implementations of AFDA were developed and tested: H-AFDA where this boundary was collocated along the same gridline as the horizontal displacement and the normal stresses are defined, and, W-AFDA that places the free surface at the same level of the shear stress and the vertical displacement. Figure 4 depicts the two possible collocations of the free surface in a 2D SSG. Obviously, every of these formulations yields just one displacement component at the exact free-surface boundary, and the other component has to be approximated somehow. This is the only drawback possibly argued against AFDA techniques. Section 6.1 presents the implementation details of a 2D version of the original 3D H-AFDA. The availability of u along $z = 0$ after solving the elastic model (3)-(4) in each time step justifies our interest in H-AFDA. Dispersion analysis of the numerical Rayleigh wave based on the spectral properties of the computed $u(x, z=0)$ are performed and presented in section 8, and H-AFDA is compared with our own PS-V solvers.

In an effort to develop an algorithm that satisfy (i) and (ii), but at the same time produces both displacements (u,w) along the free surface, a slight modification in the way of imposing the zero-traction conditions could be combined with a set of fourth-order differentiators like those given in section 1. The result is a high-order and full description of the wave field at this boundary. Similarly to AFDA cases, the two possible schemes are stated according the initial collocation of the boundary. The implementation details are given below in sections 6.2 and 6.3.

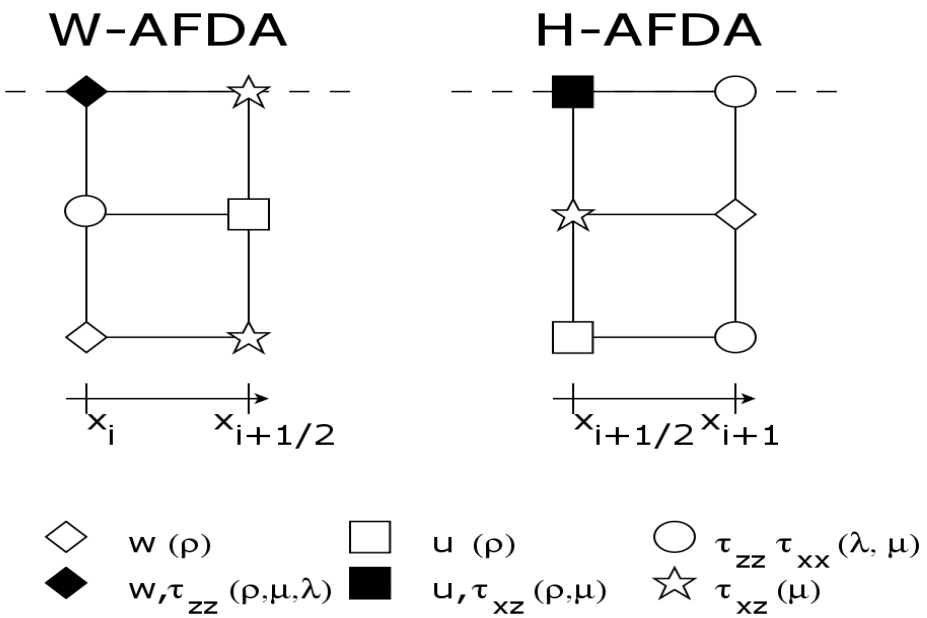


Figure 4. Ajusted finite difference approximations proposed by Krisket et. al (2002). Each algorithm only yield one displacement component at the free surface, W-AFDA computes the vertical displacement, while the horizontal displacement is given by H-AFDA.

6.1. Horizontal adjusted finite difference approximation (H-AFDA; Kristek et al. (2002)).

The calculation performed by H-AFDA along gridlines $z = 0, h/2, h$ is based on one-sided finite differencing, as described below, while the computation in the interior uses the classical and centered staggered differencing. The original formulas proposed by Kristek et al. (2002) (formulas (1) – (4)) and the classical central differentiator (formula (0)) are summarized in Table 1.

2D version of Algorithm H-AFDA:

STEP I: Impose boundary conditions: $\tau_{zz}(x_i, 0, n\Delta t)= 0$, $\tau_{xz}(x_{i+1/2}, 0, n\Delta t)= 0$.

STEP II: Calculate $u_x(x_i, 0, n\Delta t)$ by using formula 0. The condition $\tau_{zz}= 0$ implies that $w_z = - \gamma u_x$ for $\gamma=\lambda/(\lambda+2\mu)$. Stress component $\tau_{xx}(x_i, 0, n\Delta t)$ is given by Hooke's law (4).

STEP III: Compute $u_z(x_{i+1/2}, h/2, n\Delta t)$ by using formula 2, and, $w_x(x_{i+1/2}, h/2, n\Delta t)$ with formula 0. Stress component $\tau_{xz}(x_{i+1/2}, h/2, n\Delta t)$ is obtained from (4).

STEP IV: Obtain $w_z(x_i, h, n\Delta t)$ by using formula 3, and, calculate $u_x(x_i, h, n\Delta t)$ with formula 0. Hooke's law allows to compute $\tau_{zz}(x_i, h, n\Delta t)$ and $\tau_{xx}(x_i, h, n\Delta t)$.

STEP V: By using formula 1 compute $\tau_{xz'z}(x_{i+1/2}, 0, n\Delta t)$, and $\tau_{xx'x}(x_{i+1/2}, 0, n\Delta t)$ is obtained with formula 0. The first equation of motion is used to get $u(x_{i+1/2}, 0, (n+1)\Delta t)$.

STEP VI: Calculate $\tau_{zz'z}(x_i, h/2, n\Delta t)$ with formula 2, and $\tau_{xz'x}(x_i, h/2, n\Delta t)$ is obtained by using formula 0. The second equation of motion allows to compute $w(x_i, h/2, (n+1)\Delta t)$.

STEP VII: Compute $\tau_{xz'z}(x_{i+1/2}, h, n\Delta t)$ with formula 4, and $\tau_{xx'x}(x_{i+1/2}, h, n\Delta t)$ is obtained by using formula 0. The first equation of motion yields $u(x_{i+1/2}, h, (n+1)\Delta t)$.

Formula 0 :	$f'(\xi)=\frac{1}{h}\left[\frac{1}{24}f\left(\xi-\frac{3}{2}h\right)-\frac{9}{8}f\left(\xi-\frac{h}{2}\right)+\frac{9}{8}f\left(\xi+\frac{h}{2}\right)-\frac{1}{24}f\left(\xi+\frac{3}{2}h\right)\right]+O\left(h^4\right)$
Formula 1 :	$f'(\xi)=\frac{1}{h}\left[-\frac{352}{105}f(\xi)+\frac{35}{8}f\left(\xi+\frac{h}{2}\right)-\frac{35}{24}f\left(\xi+\frac{3}{2}h\right)+\frac{21}{40}f\left(\xi+\frac{5}{2}h\right)-\frac{5}{56}f\left(\xi+\frac{7}{2}h\right)\right]+O\left(h^4\right)$
Formula 2 :	$f'(\xi)=\frac{1}{h}\left[-\frac{11}{12}f\left(\xi-\frac{h}{2}\right)+\frac{17}{24}f\left(\xi+\frac{h}{2}\right)+\frac{3}{8}f\left(\xi+\frac{3}{2}h\right)-\frac{5}{24}f\left(\xi+\frac{5}{2}h\right)+\frac{1}{24}f\left(\xi+\frac{7}{2}h\right)\right]+O\left(h^4\right)$
Formula 3 :	$f'(\xi)=\frac{1}{h}\left[-\frac{h}{22}f'(\xi-h)-\frac{577}{528}f\left(\xi-\frac{h}{2}\right)+\frac{201}{176}f\left(\xi+\frac{h}{2}\right)-\frac{9}{176}f\left(\xi+\frac{3}{2}h\right)+\frac{1}{528}f\left(\xi+\frac{5}{2}h\right)\right]+O\left(h^4\right)$
Formula 4 :	$f'(\xi)=\frac{1}{h}\left[\frac{16}{105}f(\xi-h)-\frac{31}{24}f\left(\xi-\frac{h}{2}\right)+\frac{29}{24}f\left(\xi+\frac{h}{2}\right)-\frac{3}{40}f\left(\xi+\frac{3}{2}h\right)+\frac{1}{168}f\left(\xi+\frac{5}{2}h\right)\right]+O\left(h^4\right)$

Table 1. Adjusted finite difference formulas used by Krisket et. al (2002) in the implementation of W-AFDA and H-AFDA techniques.

6.2. Vertical Mimetic scheme over a SSG: W-MSSG.

Similarly to the case of W-AFDA, in this scheme the free surface is collocated along a *z*-gridline where *w* and τ_{xz} are initially defined. The difference between these two techniques relies on using the set of values $w(x_i,0,t)$ to approximate $w_x(x_{i+1/2},0,t)$ at certain time level *t*, and then the discretization of the boundary condition $\tau_{xz}= 0$ at every grid point $(x_{i+1/2}, h)$ yields the set of values $u(x_{i+1/2},0,t)$. Figure 5 displayed this new layout where (u,w) is computed at z = 0. The calculation is shown below where the collection of numerical differentiators comprise in $G_{4\text{-}4\text{-}4}$ (given in appendix 1) are used. Basically,

$$u_z(x_{i+1/2},0,t)+w_x(x_{i+1/2},0,t)=0$$

$$g_{11}u(x_{i+1/2},0,t)+g_{12}u(x_{i+1/2},h/2,t)+g_{13}u(x_{i+1/2},3h/2,t)+g_{14}u(x_{i+1/2},5h/2,t)+$$
$$g_{15}u(x_{i+1/2},7h/2,t)+g_{16}u(x_{i+1/2},9h/2,t) \qquad =-\Delta z\, w_x(x_{i+1/2},0,t)$$

This implies that,

$$u(x_{i+1/2},0,t)=-\frac{1}{g_{11}}\left[g_{12}u(x_{i+1/2},h/2,t)+\ldots+g_{16}u(x_{i+1/2},9h/2,t)+\Delta z\, w_x(x_{i+1/2},0,t)\right] \qquad (8)$$

Equation (8) shows that the implementation of this simple enhancement requires the previous computation of the horizontal displacement at some interior nodes. Actually, at every time step W-MSSG evolves the wave field at every interior grid point by solving the equations of motion before the zero-stress conditions are treated. The detailed treatment performed by W-MSSG at the free surface is sketched below.

Algorithm W-MSSG:

STEP I: Set up boundary condition: $\tau_{zz}(x_i,0,n\Delta t)= 0$. Calculate $\tau_{zz'z}(x_i,0,n\Delta t)$ by using one-sided differentiator defined by coefficients $g_{1j}/\Delta z$.

STEP II: Compute $w(x_i,0,(n+1)\Delta t)$ through the simplified equation of motion, $\rho w_{tt} = \tau_{zz'z}$ ($\tau_{xz'x} = 0$ at $z=0$, due to boundary condition $\tau_{xz}= 0$).

STEP III: Obtain $w_x(x_{i+1/2},0,(n+1)\Delta t)$ with central differentiating defined by $g_{4j}/\Delta x$.

STEP IV: Use equation (8) to calculate $u(x_{i+1/2},0,(n+1)\Delta t)$.

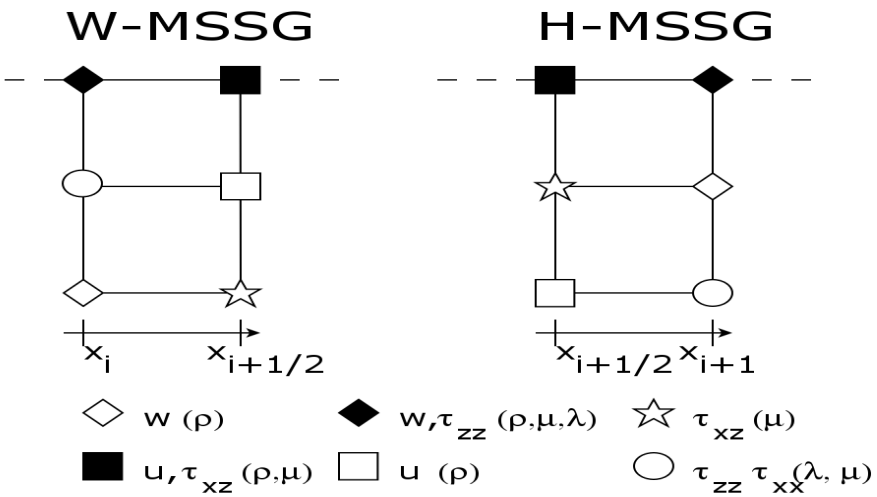


Figure 5. Extension of a SSG by the inclusion of compound nodes(filled symbols) to place the displacement vector *(u,w)* at z = 0. Free surface is collocated along a (w, τ_{xz})-gridline in the design of W-MSSG, while this boundary is defined along a $(u, \tau_{xx}, \tau_{zz})$- gridline in the case of H-MSSG.

6.3. Horizontal Mimetic scheme over a SSG: H-MSSG.

As the reader expects from the acronyms H-MSSG, the free surface is placed at the same level of *u* and normal stresses in a SSG, which is the case of H-AFDA shown in figure 5. The vertical displacement is obtained at this gridline by using the boundary condition $\tau_{zz}= 0$ at every grid point $(x_i,0)$ combined with a mimetic approximation to $u_x(x_i,0,t)$, i.e.,

$$(\lambda+2\mu)w_z(x_i,0,t) + \lambda u_x(x_i,0,t) = 0$$
$$g_{11}w(x_i,0,t)+g_{12}w(x_i,h/2,t)+g_{13}w(x_i,3h/2,t)+g_{14}w(x_i,5h/2,t)+g_{15}w(x_i,7h/2,t)+g_{16}w(x_i,9h/2,t) = -\Delta z\gamma u_x(x_i,0,t)$$

or equivalently,

$$w(x_i,0,t) = -\frac{1}{g_{11}}[g_{12}w(x_i,h/2,t)+...+g_{16}w(x_i,9h/2,t)+\Delta z\,\gamma u_x(x_i,0,t)] \qquad (9)$$

Similar to the case of W-MSSG, the updating of the wave field at every interior grid point must precede the application of (9) for the same time level. Remaining details of the numerical treatment of the free surface are given in the next sketch.

Algorithm H-MSSG:

STEP I: Set up boundary condition: $\tau_{xz}(x_{i+1/2},0,n\Delta t)= 0$. Calculate $\tau_{xz'z}(x_i,0,n\Delta t)$ by using one-sided differentiator defined by coefficients $g_{1j}/\Delta z$.

STEP II: Compute $u(x_{i+1/2},0,(n+1)\Delta t)$ through the first equation of motion. Central differentiating using $g_{4j}/\Delta x$ is used to approximate $\tau_{xx'x}(x_{i+1/2},0,(n+1)\Delta t)$.

STEP III: Obtain $u_x(x_i,0,(n+1)\Delta t)$ along the free surface.

STEP IV: Use equation (9) to calculate $w(x_i,0,(n+1)\Delta t)$.

7. Mimetic discretization using a classical staggered grid: MCSG algorithm.

In this section, the last fourth-order algorithm designed on classical staggered grid previously used in solving diffusion phenomena (Shaskov et al. 1996, Hyman et al. 2002), and Maxwell's equations (Hymann et al. 1999) is described. Figure 6 (left chart) shows a classical rectangular grid where both components of (u,w) are defined at every cell center and the treatment of the traction-free boundary conditions yields the value of (u,w) on the top face of boundary cells at z =0. Using the spatial steps $(\Delta x,\Delta z)$ and the pair of indices (i, j), $1\le i\le Nx$, $1\le j\le Nz$, every cell is denoted by a tuple (i,j), where cell (1,1) is located at the upper left corner. For sake of simplicity, the discrete analog $(u_{i,j},w_{i,j})$ corresponds to (u,w) at $((i-1/2)\Delta x,(j-1/2)\Delta z)$ which is the position of the center point of the cell (i,j). Finally, as it will be explained later, the application of the free surface boundary condition leads to boundary values (u_{i0}, w_{i0}), $1\le i\le Nx$, with a physical location of $((i-1/2)\Delta x, 0)$.

The classical grid given by figure 6 has a constraint for the solution of the elastic system (1)-(2), it does not allow fully computing any of the components of the strain tensor at any grid position. Actually, u_x and w_x could be computed at the center of both left and right faces of any cell, but u_z and w_z cannot be calculated at these locations. In other hand, u_z and w_z are available at the center of the top and bottom faces, but u_x and w_x are not. To overcome this liability, 4th order Lagrange interpolation of the closest values is used to approximate the unavailable quantity at every face and then strain and stress tensors are conveniently defined. Left cell in Figure 4 shows approximations to x-derivates and z-derivates computed by finite difference formulas as no-hat symbols u_x, w_x, u_z, w_z, while hat symbols are used to denote approximation by interpolating $\hat{u}_x,\hat{u}_z,\hat{w}_x,\hat{w}_z$. Figure 6 (right chart) depicts the computation cell with displacement, stress, and, material properties locations.

Boundary values (u_{i0}, w_{i0}), $1\le i\le Nx$, at $z=0$ are calculated by solving the zero-traction conditions and using staggered differentiators along every $x = i\Delta x$ gridline and nodal differentiators along $z=0$, in particular MCSG was implemented using 4th order operators $G_{4\text{-}4\text{-}4}$ and $N_{4\text{-}4\text{-}4}$ (given in appendix 1), respectively. Let us define the column vectors $U_j = (u_{1j},\ldots, u_{Nx\,j})^T$, and $W_j = (w_{1j}, \ldots, w_{Nx\,j})^T$ for $j =1,2,3,4,5,6$. In addition, consider the matrix $I_\gamma = diag(\gamma_{10},\ldots, \gamma_{Nx\,0})$ given by values of γ at every boundary gridpoint. Thus, $\tau_{xz} = \tau_{zz} = 0$ could be respectively discretized as,

$$(u_z + w_x)\big|_{0,j} = 0 \quad \Rightarrow \frac{1}{\Delta z}\sum_{j=1}^{6} g_{1j}U_{j-1} + \frac{1}{\Delta x} N_{4-4-4} W_0 = 0$$

$$(\gamma u_x + w_z)\big|_{0,j} = 0 \quad \Rightarrow \frac{1}{\Delta x} I_\gamma N_{4-4-4} U_0 + \frac{1}{\Delta z}\sum_{j=1}^{6} g_{1j}W_{j-1} = 0$$

By re-writing equations above we get,

$$U_0 + \kappa N_{4-4-4} W_0 = -\sum_{j=2}^{6} \left(g_{1j} \big/ g_{11} \right) U_{j-1} \tag{10.1}$$

$$W_0 + \kappa I_\gamma N_{4-4-4} U_0 = -\sum_{j=2}^{6} \left(g_{1j} \big/ g_{11} \right) W_{j-1} \tag{10.2}$$

Where, $\kappa = \Delta z / (\Delta x\, g11)$. In case that U_j, and, W_j for $j \geq 2$ are known, boundary values collected in U_0, and, W_0 could be determined by solving the system above. However, after subtracting (10.2) from (10.1) times $\kappa I_\gamma N_{4\text{-}4\text{-}4}$, this system is decoupled leading to

$$\left(\kappa I_\gamma N_{4-4-4}^{2} - I \right) W_0 = \left(\kappa I_\gamma N_{4-4-4} \right) b - c \tag{11}$$

In (11), vectors b and c are the right hand side of (10.1) and (10.2), respectively. Notice that (11) represents a system of Nx linear equations to determine W_0 according to boundary conditions and discrete values of the wavefield at interior. Later, (10.1) could be used to compute U_0. This is the basic idea of the boundary treatment performed by MCSG at every time iteration, whose efficiency relies on the structure of the matrix in (11) and the factorization technique used. Basically, this matrix is non-symmetric, banded, and its entries depend only on κ and γ, thus a standard LU factorization is applied in the initial stage of simulation, and both factors are used at every time iteration to solve (11) by forward and backward substitutions. Therefore, the computational complexity of the combined boundary treatment (11) followed by (10.1) remains as the same as the numerical calculation of the interior wavefield, which is $O(Nx^2)$. A time iteration of the MCSG scheme is sketched in the following framework:

MCSG algorithm:

STEP I (Interior): Solve (3)-(4) to get (u,w) for t = (n+1)Δt at every interior node. Accuracy is 4th-order in space and 2nd-order in time. Then, compute vectors b and c.

STEP II (Free Surface): Solve (11) for W_0 at t = (n+1)Δt by using backward/forward substitutions.

STEP III (Free Surface): Calculate U_0 for t = (n+1)Δt from equation (10.1).

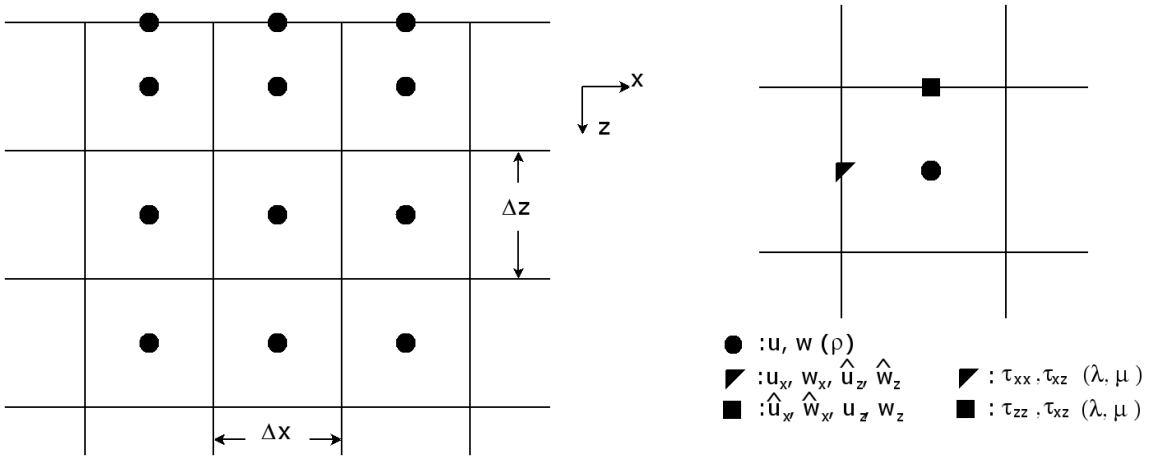


Figure 6. Distribution of displacement vector along the classical staggered grid used by MCSG (left chart). Both components (u,w) are defined at every cell center and also along the free surface. Locations of partial derivates u_x, u_z, w_x, w_z calculated by staggered differentiation and approximations $\hat{u}_x, \hat{u}_z, \hat{w}_x, \hat{w}_z$ given by interpolation are illustrated in the right chart of this figure.

8. Lamb's Problem: Test in a homogeneous medium.

Lamb's problem has been extensively used to assess the efficacy of numerical implementations of a free surface boundary condition (Virieux, 1986, Ma et al., 2004, Min et al., 2004). First at all, the response of an elastic half-plane due to the application of a vertical point force f(t) at (x_0,z=0) is known and given by the convolution of f(t) with the fundamental solution obtained by Cagniard-De Hoop's technique (Aki and Richards, 1980). In addition, sources near the surface tend to excite strong Rayleigh waves, which are dispersionless in the case of a homogeneous domain. Thus, the time series of the exact displacements and its spectral properties are used to analyze the accuracy of numerical solutions at some superficial locations (receivers). Artificial dispersion of the numerical Rayleigh wave measured through phase speed is the metric in this analysis.

Here, we have performed a similar experiment to the one proposed by Ma et al. 2004. The vertical point force is a narrowbanded Gaussian-type pulse given by f(t)= exp[-500(t – 0.25)^2] with an aproximate frequency spectrum of [0,20] Hz. Figure 7 shows the distribution of receivers along the free surface of a elastic half space with density 2500 kg/m^3 and S-wave speed of 1500 m/s. In the case of a Poisson solid ($\alpha = \sqrt{3}\beta$), figure 8 depicts comparisons among the numerical solutions computed by algorithms described in sections 3-7, and the analytical solution, for both horizontal and vertical displacements. Note the effect of the numerical dispersion as waves propagate away from the source. Algorithms with fourth-order free surface treatment (H-AFDA, H-MSSG, W-MSSG, MCSG) use 6 points (h = 12m) per minimum S-wavelength ($\lambda_S^{\min}$), as opposed to the 10 points (h = 6m) per $\lambda_S^{\min}$ employed by second-order algorithms (VRSG, MRSG). The time step was chosen as $dt = 0.5*dx/\alpha$.

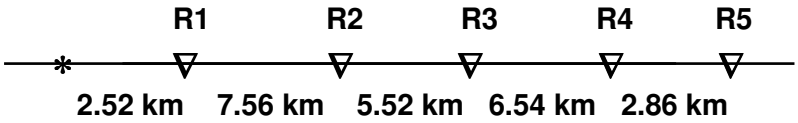


Figure 7. 2D homogeneous half-space with ρ = 2500 kg/m^3 and β = 1500m/s. Domain and receivers for Lamb's problem.

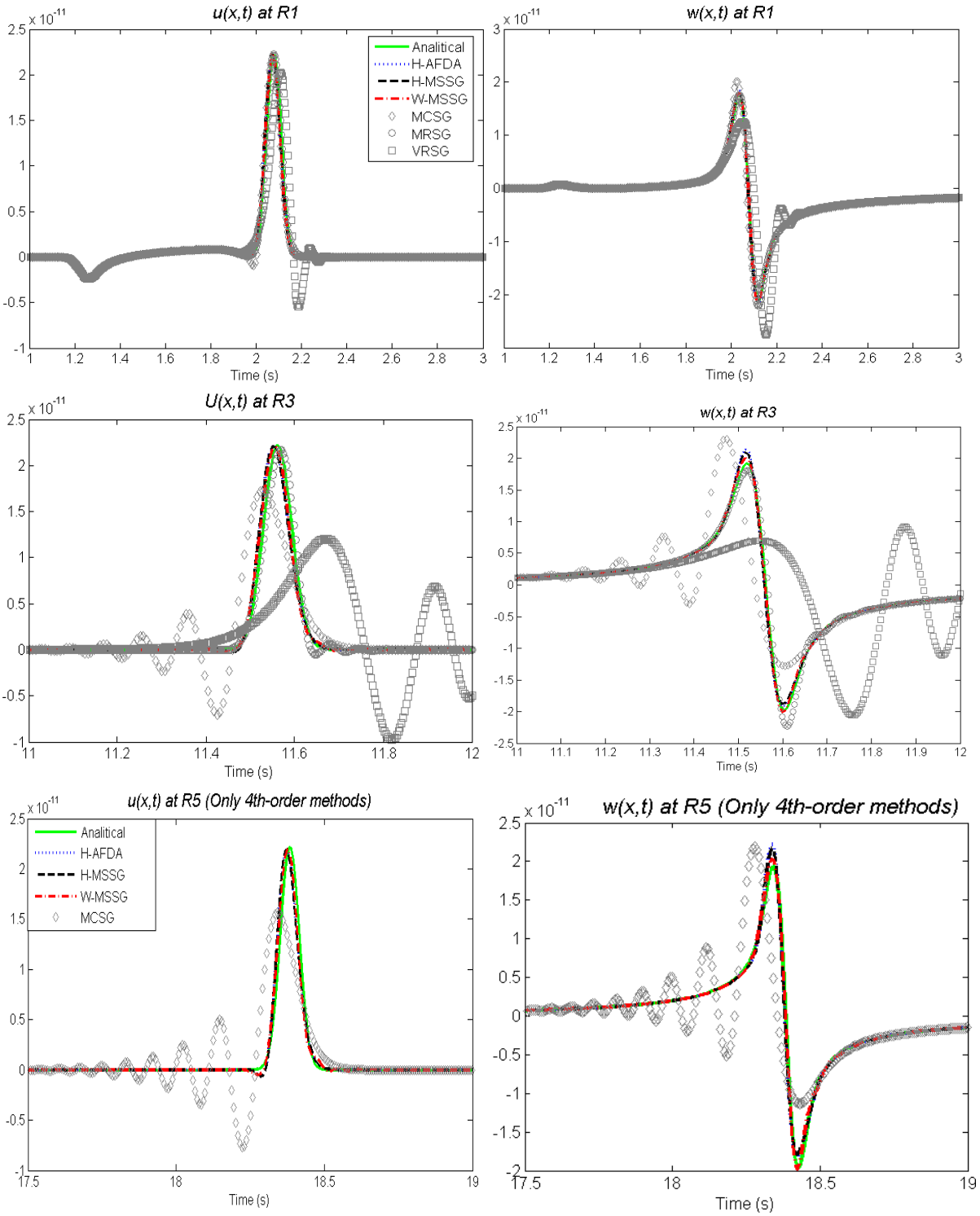


Figure 8. Comparisons among analytical and numerical displacements for Lamb's problem at receivers R1, R3 and R5. In the case of R5, only 4th order solutions are shown. The domain is a Poisson solid. Algorithms H-AFDA, H-MSSG, W-MSSG, and, MCSG use h = 12 m while VRSG, and, MRSG employ h = 6 m.

9. Dispersion analysis of the numerical Rayleigh wave.

Although in a Lamb's experiment the signal recorded by a superficial station is mainly the Rayleigh pulse, here a basic approach has been introduced to isolate this pulse from body-wave

arrivals. Figure 9 depicts both analytical time series, $u(x_3,0,t)$ and $w(x_3,0,t)$, at receiver R3. Comparing these signals, $u(x_3,0,t)$ reaches very low values during a time gap of approximately 0.7 seconds between the S-wave($T_{S;R2}$) and the Rayleigh-wave($T_{R;R2}$) arrivals, which represents a natural separation of these pulses. On other hand, $w(x_3,0,t)$ does not present this behavior. Thus, the first basic idea of this analysis is defining a cut-off time ($T_{C;R3}$) to truncate the time series of $u(x_3,0,t)$ and keep records that only describe the Rayleigh pulse, i.e., the energy content of $u(x_3,0,t \geq T_{C;R3})$ would not have any interference from P or S waves. Obviously, $T_{S;R3} < T_{C;R3} < T_{R;R3}$, and one simple definition is stated through finding the minimum magnitude of $u(x_3,0,t)$ for $T_{S;R3} < t < T_{R;R3}$, and choosing a particular t in case of non-uniqueness. Here, $T_{C;R3}$ is defined as,

$$T_{C;R3} = \min\left\{ t^* : u(x_3,0,t^*) = \min_{T_{S;R3}<t<T_{R;R3}} \left| u(x_3,0,t) \right| \right\} \tag{12}$$

The reason of considering the magnitude of $u(x_3,0,t)$ is clearly exposed by the seismogram produced by MCSG. It shows how low-magnitude oscillations reach R3 earlier than the important part of the Rayleigh pulse, illustrating a typical effect of numerical dispersion in fourth-order finite-difference solutions where high-frequency components travel faster than the low frequency ones. In addition, the fact that t* is not unique is also depicted in this case. For the same reasons, equation (12) is also convenient for solutions given by the others fourth-order algorithms H-AFDA, H-MSSG, and, W-MSSG.

Numerical dispersion affects solutions provided by VRSG and MRSG in an opposite way than the one seen in fourth-order solutions, and high-frequency modes are delayed as a result of the second-order free-surface treatment. In this sense, equation (12) or any other trivial choice for $T_{C;R3}$ that follows $T_{S;R3} < T_{C;R3} < T_{R;R3}$ would allow to define properly the truncated series $u(x_3,0,t \geq T_{C;R3})$. Actually, the important consideration in this case is that the simulation time should be long enough to allow most of these low-amplitude oscillations to reach and be recorded at R3. Finally, the algorithm PhS_Rayleigh is proposed to perform dispersion analysis of the numerical Rayleigh wave computed by either forth-order or second-order solvers. Even, when time series recorded at receivers R3 and R2 are explicitly mentioned, PhS_Rayleigh could be used for another couple of receivers located away enough from the source.

Algorithm PhS_Rayleigh: Phase Speed of Rayleigh Wave.

STEP I: Obtain truncated time series at R3, $u(x_3,0,t \geq T_{C;R3})$, for $T_{C;R3}$ given by (12). Compute phase spectrum $\phi_{R3}(f)$.

STEP II: Obtain an equal-sized truncated time series at R2, $u(x_2,0, T \geq t \geq T_{C;R2})$, for $T_{C;R2}$ given by (12), and T is chosen such that: size{ $u(x_2,0, T \geq t \geq T_{C;R2})$} = size { $u(x_3,0,t \geq T_{C;R3})$}. Compute phase spectrum $\phi_{R2}(f)$.

STEP III: Calculate phase speed velocity, C(f), using the phase difference (Lay & Wallace, 1995):

$$\phi_{R3}(f) - \phi_{R2}(f) = 2\pi f \left[T_{C;R3} - T_{C;R2} \right] + \frac{2\pi f}{C(f)} \left(x_3 - x_2 \right)$$

The simplicity of this algorithm is a consequence of the idealized domain used in this experiment. The speed of the exact Rayleigh(C_0) wave excited in a homogenous half-plane is a known function of β and Poisson's ratio(σ) and given by the unique real root of the following cubic equation that satisfies $C_0 < \beta$ (Lay & Wallace, 1995),

$$\varepsilon^3 - 8\varepsilon^2 + \left(24 - 16/k\right)\varepsilon - 16\left(1 - 1/k\right) = 0$$

where,

$$\varepsilon = \left(C_0/\beta\right)^2, \qquad k = \frac{2(\sigma - 1)}{2\sigma - 1}$$

Table 2 shows 6-digit approximations of $\sqrt{\varepsilon}$ for a representative set of values to σ. Steps I and II of PhS_Rayleigh algorithm use C_0 to compute the analytical arrival times $T_{R;R2}$ and $T_{R;R3}$. The consistency of this algorithm is depicted in figures 10 and 11, where exact time series were used as input, and the resulting C(f) equaled C_0, within round-off errors. Figure10 show phase speed distortions introduced by the second-order, and, fourth-order, free-surface numerical solvers in the case of σ=0.25. Results for different Poisson ratios σ = 0.2, 0.3, 0.35 are depicted in Figure 11.

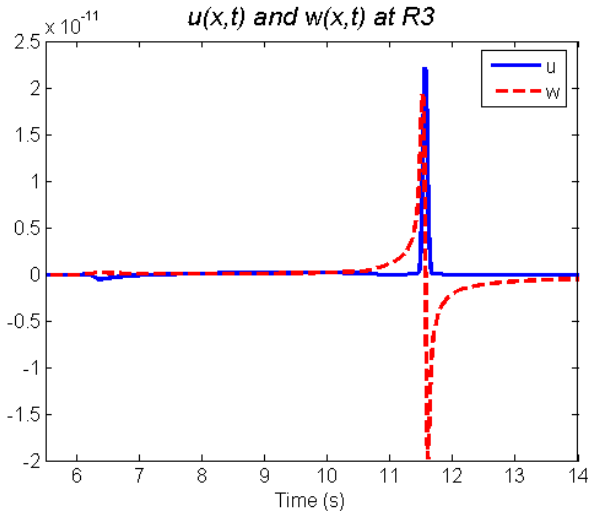


Figure 9. Horizontal(*u*) and vertical(*w*) analytical displacements at R3. P-wave arrival time is 6.0s, S-wave arrival time($T_{S;R3}$) = 10.40s, and, Rayleigh wave arrival time ($T_{R;R3}$) = 11.31s.

σ	$\sqrt{\varepsilon}$
0.20	**0.910996…**
0.25	**0.919402…**
0.30	**0.927413…**
0.35	**0.935013…**

TABLE 2. Proportionality between Rayleigh wave speed(C_0) and S-wave speed(β) for different Poisson's ratios: $C_0 = \sqrt{\varepsilon}\ \beta$.

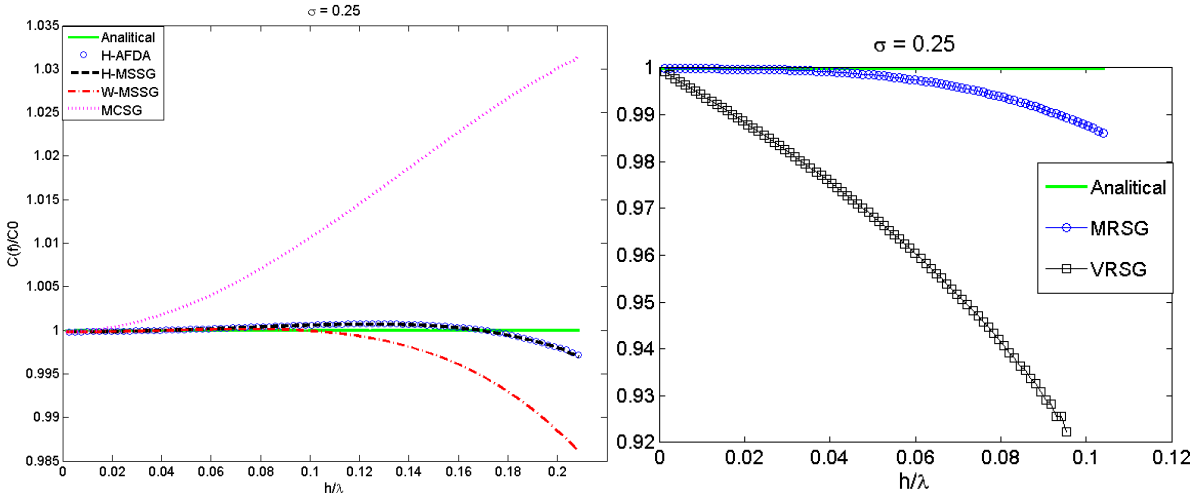
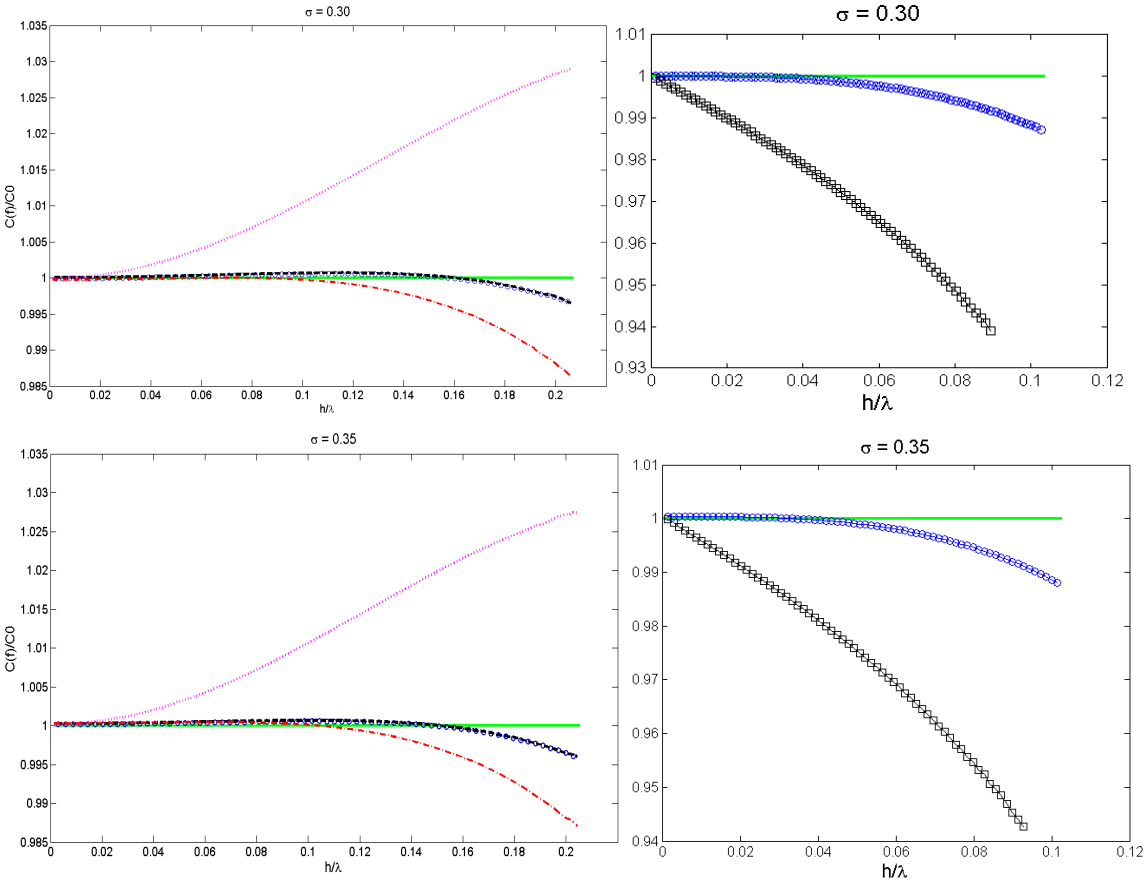


Figure 10. Normalized phase velocity of the numerical Rayleigh wave in the case of σ = 0.25. PhS_Rayleigh algorithm uses time series recorded at R5 and R4 in the case of 4th-order solvers (left chart), while seismograms from R3 and R2 are employed in the case of 2nd-order schemes (right chart).



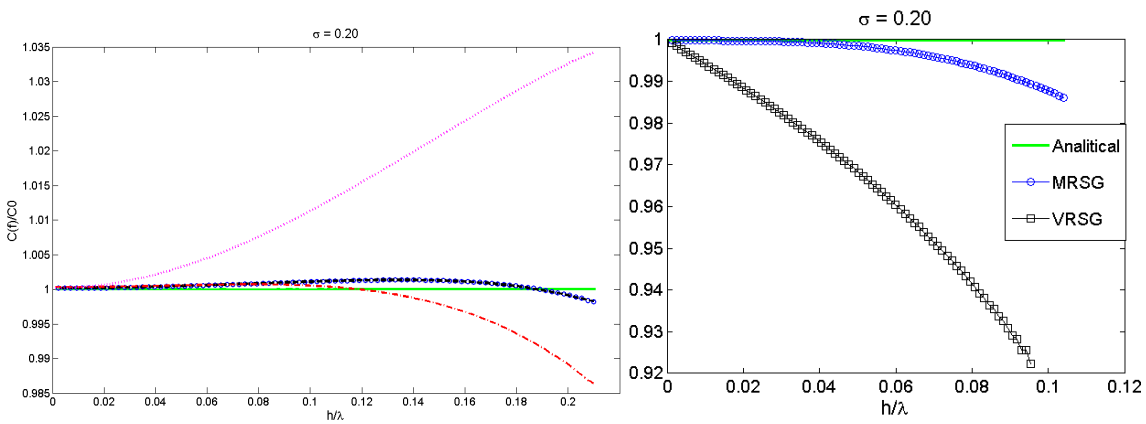


Figure 11. Normalized phase velocity of the numerical Rayleigh wave in cases of σ = 0.20, 0.30, 0.35. Similarly to the case of σ = 0.25, time series recorded at R5 and R4 are used in cases of 4th-order solvers (left charts), while seismograms from R3 and R2 are employed for 2nd-order schemes (right charts).

In addition to the numerical dispersion curves presented in figures 10 and 11, results from PhS_Rayleigh algorithm allow to estimate the number of points per λ_R required to keep the dispersion error below certain bound. Because of the low dispersive performance shown by H-AFDA, H-MSSG, and, W-MSSG this bound is chosen as 1%. Table 3 summarizes these interesting according to the set of values of Poisson ratio considered in this dispersion analysis.

Scheme	Points per λ_R			
	σ = 0.20	σ = 0.25	σ = 0.30	σ = 0.35
H-AFDA	4.8	4.8	4.9	4.9
H-MSSG	4.8	4.8	4.9	4.9
W-MSSG	5.1	5.2	5.2	5.2
MCSG	10.7	10.3	10.3	10.3
MRSG	10.7	10.7	10.6	10.4
VRSG	59.1	58.3	50.6	45.9

TABLE 3. Number of points per λ_R required by each numerical scheme to propagate the Rayleigh pulse with 1% of dispersion error at most. Results are listed for different values to Poisson ratio (σ).

The dispersion analysis given above leads to the next important conclusions valid for the discrete set of values of Poisson ratio 0.20, 0.25, 0.30, and, 0.35. Algorithms H-MSSG, W-MSSG, and H-AFDA are equally efficient requiring 5 points per λ_R to accurately simulate the propagation of Rayleigh waves (≤ 1% of dispersion error). This remarkable behavior makes these algorithms the least dispersive schemes in their class (4th-order SSG-FD) available today. Similar accuracy could be achieved by schemes MCSG and MRSG if 10 points per λ_R are used in the simulation, being comparable to traditional 4th order free-surface FD techniques based on symmetry conditions (Gottschammer and Olsen, 2001). Because of the free-surface treatment performed by MRSG avoids ghost points and does not involve the simultaneous solution of linear systems, this scheme becomes computationally economic compared to its competitors. Finally, VRSG showed a high dispersive performance demanding more than 40 nodes per λ_R for a precise modeling in the context exposed above. In this sense, even when MRSG is only applicable to cases of flat free surfaces, it becomes a new choice to be used by modelers who prefer rotated staggered grids.

11. Conclusions.

Low dispersive modeling of surface waves propagating along a flat free surface has been achieved thanks to the incorporation of compound nodes at this boundary and the high-order one-

sided differentiation of wave fields. New algorithms H-MSSG and W-MSSG designed for a SSG are equally efficient than their precursor H-AFDA requiring only 5 points per λ_R for accurate propagation of Rayleigh waves ($\leq 1\%$ of dispersion error). However, H-MSSG and W-MSSG yield both components of the wave field at the surface, while H-AFDA requires an additional extrapolation for the vertical displacement. Similarly, the MRSG scheme represents a more accurate alternative to implement zero-tractions conditions over a RSG than its commonly used competitor VRSG, where the former demands approximately of 10 nodes per λ_R compared to at least 40 gridpoints required by the later to achieve the same precision.

Two potential extensions of algorithms presented here are worth to mention. In one hand, exploiting the one-side high-order mimetic differentiation to model alternative and challenging boundary conditions Neumann type such as fault jump conditions for frictional sliding of contiguous elastic plates look feasible and promising. One the other, the effect of real topography on surface and body waves could de studied by adapting these algorithms to non-cartesian grids.

12. Acknowledgments.

We are really grateful to Dr. Luis Angel Dalguer for his valuable advice and foreseeing the promising use of rotated staggered grid in this work. We also want to thank Dr. Kim Olsen for his helpful comments and providing the SSVS code (this code was developed by Yi Luo, University of Utah (5/6/89), and modified by Jerry Schuster (5/19/89). Many thanks to Dr. Shuo Ma for kindly offering a code to compute the analytical solution for the Lamb's problem (originally coded by Dr. Eric Dunham). Funding for this work was provided by NSF Grant ATM 0325033.

13. References.

Bohlen, Th., Saenger, E. (2003). 3-D viscoelastic finite-difference modeling using the rotated staggered grid – tests of accuracy. EAGE 65th conference & Exhibition – Stavanger, Norway (2-5 June).

Castillo, J., and Grone, R., A matrix analysis approach to higher-order approximations for divergence and gradients satisfying a global conservation law. SAIM J. Matrix Anal. Appl. 25(2003) 128-142.E.

Castillo, J., Hyman, J., Shashkov, M., and Steinberg, S. Fourth and sixth-order conservative finite difference approximations of the divergence and gradient. Appl. Numer. Math. 37(2001) 171-187.

Castillo, J., Hyman ,J., Shashkov, M., and Steinberg, S. The sensitivity and accuracy of fourth order finite-difference schemes on non-uniform grids in one dimension, Comput. Math. Appl., 30-8(1995) 41-55.

Castillo, J., Yasuda, M. On the solvability of the Castillo-Grone mimetic discretization, submitted to *Klewer Academic Publishers* (2005).

J. Hyman, and M. Shashkov, Mimetic discretization for Maxwell's equations, J. Comput. Phys., 151(1999) 881-909.

J. Hyman, J. Morel, M. Shashkov, and S. Steinberg. Mimetic finite difference methods for diffusion equations, Comput. Geosciences, 6(2002) 333-352.

Lay, T., and, Wallace, T., 1995, Modern global seismology: Academic Press.

Levander, A. Fourth-order finite-difference P-SV seismograms. Geophysics 53(1988) 1425-1436.

Kristek, J., Moczo, P., Archuleta, R.,(2002). Efficient methods to simulate planar free surface in the 3D 4th-order staggered-grid finite-difference schemes, Stud. Geophys. Geod., 46, 355-381.

Gelis, C., Leparoux, D., Virieux, J., Bitri, A., Operto, S., Grandjean, G., 2004. (PRE-PRINT).

Graves, R., Simulating seismic wave propagation in 3D elastic media using staggered-grid finite differences: Bull. Seis. Soc. Am. 86(1996) 1091-1106.

Gottschammer, E., and, Olsen, K. B., 2001, Accuracy of the explicit planar free-surface boundary condition implemented in a fourth-order staggered-grid velocity-stress finite-difference scheme: Bull. Seis. Soc. Am., 91, 3, 617-623.

Ma Shuo, R. Archuleta, and P. Liu, Hybrid modeling of elastic P-SV wave motion: A combined finite-element and staggered-grid finite-difference approach, Bull. Seis. Soc. Am. 94(2004) 1557-1563.

Madariaga, R. Dynamics of an expanding circular fault. Bull. Seis. Soc. Am. 66(1976) 163-182.

Min, D., Shin, C., and, Yoo, H., 2004, Free-surface boundary condition in finite-difference elastic wave modeling: Bull. Seis. Soc. Am., 94, 1, 237-250.

Olsen, K. B., S. M. Day, J. B. Minster, Y. Cui, A. Chourasia, M. Faerman, R. Moore, P. Maechling, and T. Jordan (2006). Strong Shaking in Los Angeles Expected From Southern San Andreas Earthquake, *Geophys. Res. Lett., Vol 33,* L07305, doi:10.1029/2005GRL025472.

Olsen, K. B., (1994). Simulation of three-dimensional wave propagation in the Salt Lake Basin, Ph.D. Thesis, University of Utah.

Pitarka, Arben, 1999. 3D finite-difference modeling of seismic motion using staggered grids with nonuniform spacing: Bull. Seis. Soc. Am., 89, 1, 54-68.

Randall, C. J., 1989. Absorbing boundary condition for the elastic wave equation: velocity-stress formulation. Geophysics, 54, 1141-1152.

Saenger, N. Gold, and S. Shapiro. Modeling the propagation or elastic waves using a modified finite-difference grid. Wave Motion 31(2000) 77-92.

Saenger, E., and, Bohlen, T., 2004, Finite-difference modeling of viscoelastic and anisotropic wave propagation using the rotated staggered grid: Geophysics, 69, 583-591.

Shashkov, M., and S. Steinberg, Solving diffusion equations with rough coefficients in rough grids, J. Comput. Phys., 129(1996) 383-405.

Shashkov, M., and S. Steinberg, Support-Operator finite-difference algorithms for general elliptic problems, Personal communication (2003).

Virieux, J. P-SV wave propagation in heterogeneous media: Velocity-stress finite-difference method. Geophysics 49(1986) 1933-1957.

Virieux, J. SH-wave propagation in heterogeneous media: Velocity-stress finite-difference method. Geophysics 51(1986) 886-901.

Zahrzdnik, J., Moczo, P., Hron, F. 1993. Testing four finite-difference schemes for behavior at discontinuities: Bull. Seis. Soc. Am., 83, 1, 107-129.

Appendix 1.

A three-parametric(α,β,γ) family of uniformly fourth-order accurate G operators and a three-parametric(α',β',γ') family of uniformly fourth-order accurate D operators, that both satisfy the discrete conservation law (2) for any six-tuple of real values (α,β,γ, α',β',γ'). These operators take the form,

$$G=\begin{bmatrix} g_{11} & g_{12} & g_{13} & g_{14} & g_{15} & g_{16} & 0 & 0 & \dots \\ \frac{16}{105}-\frac{128}{35}\alpha & -\frac{31}{24}+9\alpha & \frac{29}{24}-12\alpha & -\frac{3}{40}+\frac{54}{5}\alpha & \frac{1}{168}-\frac{36}{7}\alpha & \alpha & 0 & 0 & \dots \\ -\frac{128}{35}\beta & \frac{1}{24}+9\beta & -\frac{27}{24}-12\beta & \frac{27}{24}+\frac{54}{5}\beta & -\frac{1}{24}-\frac{36}{7}\beta & \beta & 0 & 0 & \dots \\ -\frac{16}{105}-\frac{128}{35}\gamma & \frac{3}{8}+9\gamma & -\frac{11}{24}-12\gamma & -\frac{27}{40}+\frac{54}{5}\gamma & \frac{51}{56}-\frac{36}{7}\gamma & \gamma & 0 & 0 & \dots \\ 0 & 0 & 0 & \frac{1}{24} & -\frac{27}{24} & \frac{27}{24} & -\frac{1}{24} & 0 & \dots \end{bmatrix}$$

$$g_{11} = -\frac{124832}{42735} + \frac{16512}{1295}\alpha + \frac{18816}{2035}\beta + \frac{13696}{1295}\gamma,$$
$$g_{12} = \frac{10789}{3256} - \frac{1161}{37}\alpha - \frac{9261}{407}\beta - \frac{963}{37}\gamma,$$
$$g_{13} = -\frac{421}{9768} + \frac{1548}{37}\alpha + \frac{12348}{407}\beta + \frac{1284}{37}\gamma,$$
$$g_{14} = -\frac{12189}{16280} - \frac{6966}{185}\alpha - \frac{55566}{2035}\beta - \frac{5778}{185}\gamma,$$
$$g_{15} = \frac{11789}{22792} + \frac{4644}{256}\alpha + \frac{5292}{407}\beta + \frac{3852}{259}\gamma,$$
$$g_{16} = -\frac{48}{407} - \frac{129}{37}\alpha - \frac{1029}{407}\beta - \frac{107}{37}\gamma.$$

$$D = \begin{bmatrix} d_{11} & d_{12} & d_{13} & d_{14} & d_{15} & d_{16} & 0 & 0 & \dots \\ \frac{1}{24} - \alpha' & -\frac{27}{24} + 5\alpha' & \frac{27}{24} - 10\alpha' & -\frac{1}{24} + 10\alpha' & -5\alpha' & \alpha' & 0 & 0 & \dots \\ -\beta' & \frac{1}{24} + 5\beta' & -\frac{27}{24} - 10\beta' & \frac{27}{24} + 10\beta' & -\frac{1}{24} - 5\beta' & \beta' & 0 & 0 & \dots \\ -\frac{1}{24} - \gamma' & \frac{5}{24} + 5\gamma' & -\frac{3}{8} - 10\gamma' & -\frac{17}{24} + 10\gamma' & \frac{11}{12} - 5\gamma' & \gamma' & 0 & 0 & \dots \\ 0 & 0 & 0 & \frac{1}{24} & -\frac{27}{24} & \frac{27}{24} & -\frac{1}{24} & 0 & \dots \end{bmatrix}$$

$$d_{11} = -\frac{6851}{7788} + \frac{39}{59}\alpha' + \frac{675}{649}\beta' + \frac{551}{649}\gamma',$$
$$d_{12} = \frac{8153}{15576} - \frac{195}{59}\alpha' - \frac{3375}{649}\beta' - \frac{2755}{649}\gamma',$$
$$d_{13} = \frac{3867}{5192} + \frac{390}{59}\alpha' + \frac{6750}{649}\beta' + \frac{5510}{649}\gamma',$$
$$d_{14} = -\frac{9005}{15576} - \frac{390}{59}\alpha' - \frac{6750}{649}\beta' - \frac{5510}{649}\gamma',$$
$$d_{15} = \frac{3529}{15576} + \frac{195}{59}\alpha' + \frac{3375}{649}\beta' + \frac{2755}{649}\gamma',$$
$$d_{16} = -\frac{24}{649} - \frac{39}{59}\alpha' - \frac{675}{649}\beta' - \frac{551}{649}\gamma'.$$

In terms of bandwidth of operators G and D, a convenient choice is given by the set of parameters (α,β,γ)= (α',β',γ')= (0,0,-1/24), where,

$$G_{4-4-4} = \begin{bmatrix} -\frac{1775}{528} & \frac{1790}{407} & -\frac{2107}{1415} & \frac{1496}{2707} & -\frac{272}{2655} & \frac{25}{9768} & 0 & \dots \\ \frac{16}{105} & -\frac{31}{24} & \frac{29}{24} & -\frac{3}{40} & \frac{1}{168} & 0 & 0 & \dots \\ 0 & \frac{1}{24} & -\frac{27}{24} & \frac{27}{24} & -\frac{1}{24} & 0 & 0 & \dots \\ 0 & 0 & \frac{1}{24} & -\frac{27}{24} & \frac{27}{24} & -\frac{1}{24} & 0 & \dots \end{bmatrix}$$

$$D_{4-4-4} = \begin{bmatrix} -\frac{1045}{1142} & \frac{909}{1298} & \frac{201}{514} & -\frac{1165}{5192} & \frac{129}{2596} & -\frac{25}{15576} & 0 & \dots \\ \frac{1}{24} & -\frac{27}{24} & \frac{27}{24} & -\frac{1}{24} & 0 & 0 & 0 & \dots \\ 0 & \frac{1}{24} & -\frac{27}{24} & \frac{27}{24} & -\frac{1}{24} & 0 & 0 & \dots \end{bmatrix}$$

In this paper, we use the sub-index 4-4-4 in operators D_{4-4-4} and G_{4-4-4} to denote the fourth order of accuracy at interior and boundary points. Another set of mimetic operators D_{2-4-2} and G_{2-4-2} have become useful in this work, where fourth order of accuracy has been kept at interior, but second formulas are given at boundary points. The reason of using this particular set will be given in section 6, but its existence shows the flexibility of the construction methodology. Specifically,

$$D_{2-4-2} = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 & \dots \\ \frac{1}{23} & -\frac{26}{23} & \frac{26}{23} & -\frac{1}{23} & 0 & 0 & 0 & \dots \\ 0 & \frac{1}{24} & -\frac{27}{24} & \frac{27}{24} & -\frac{1}{24} & 0 & 0 & \dots \\ 0 & 0 & \frac{1}{24} & -\frac{27}{24} & \frac{27}{24} & -\frac{1}{24} & 0 & \dots \end{bmatrix} \qquad G_{2-4-2} = \begin{bmatrix} -\frac{8}{3} & 3 & -\frac{1}{3} & 0 & 0 & 0 & 0 & \dots \\ \frac{4}{39} & -\frac{31}{26} & \frac{44}{39} & -\frac{1}{26} & 0 & 0 & 0 & \dots \\ 0 & \frac{1}{24} & -\frac{27}{24} & \frac{27}{24} & -\frac{1}{24} & 0 & 0 & \dots \\ 0 & 0 & \frac{1}{24} & -\frac{27}{24} & \frac{27}{24} & -\frac{1}{24} & 0 & \dots \end{bmatrix}$$

Finally, Castillo and Grone (2003) proposed uniformly fourth-order approximations to the derivate on equally-spaced nodal grid by using one-sided and centered formulas. The resulting operator N_{4-4-4} takes the form,

$$N_{4-4-4} = \begin{bmatrix} -\frac{33989}{13640} & \frac{49453}{8184} & -\frac{28993}{4092} & \frac{7391}{1364} & -\frac{18763}{8184} & \frac{16717}{40920} & 0 & \dots \\ -\frac{1}{4} & -\frac{5}{6} & \frac{3}{2} & -\frac{1}{2} & \frac{1}{12} & 0 & 0 & \dots \\ \frac{1}{12} & -\frac{287}{348} & \frac{55}{87} & -\frac{49}{174} & \frac{191}{348} & -\frac{55}{348} & 0 & \dots \\ 0 & \frac{1}{12} & -\frac{2}{3} & 0 & \frac{2}{3} & -\frac{1}{12} & 0 & \dots \end{bmatrix}$$