Seismic Wave Propagation

Mark Asch - ML-PREP

2025

wave propagation

Program

1. Harmonic waves (ODEs).
2. Acoustic wave equation.
3. Waves in elastic media.
4. Numerical methods for wave propagation:
   (a) Finite differences.
   (b) Finite elements.
   (c) Spectral element method.

CONTEXT

The Vibrating Planet

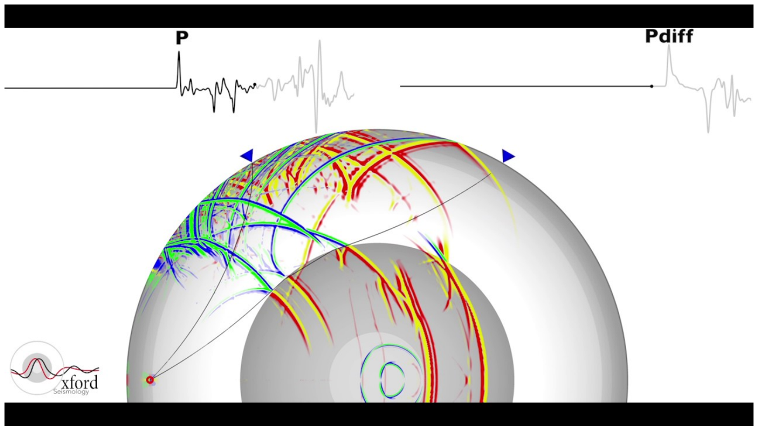
Our planet is permanently vibrating, excited by oceans, atmosphere, earthquakes, or man-made sources. Earth's physical properties are such that these vibrations—elastic waves to be more specific—often propagate to large distances carrying information on the medium they encounter along the way. There are two major challenges:

1. model and simulate this (seismic) wave propagation problem;

2. make an educated guess at the subsurface structure from observations of ground motions.

We call this last problem seismic tomography, inspired by CT scanning in medicine.

Types of Waves



- Primary, P-waves: acoustic, longitudinal, pressure waves.
- Secondary, S-waves: elastic, transversal, shear waves.
- Other waves: Surface, Rayleigh, Stoneley.

Seismology

Seismology is the science based on data from seismograms, records of mechanical vibrations of the Earth, that are caused by earthquakes and volcanic eruptions.

HARMONIC WAVES

Harmonic Waves - a simplification

- Before diving into the full, 2D-3D, elastic, partial differential wave equation (PDE) that is used for seismic wave propagation, we will study the very basic case, where the waves are considered to be harmonic and can be described by an ordinary differential equation (ODE).

- Vibration problems lead to differential equations with solutions that oscillate in time, typically in a damped or undamped sinusoidal fashion.

- Vibration problems occur throughout mechanics and physics, but the methods discussed in this lecture are also fundamental for constructing successful algorithms for partial differential equations of wave nature in multiple spatial dimensions, of which seismic waves are an important example.

Harmonic Waves - model

- The simplest model of a vibrating mechanical system has the following form:

$$u'' + \omega^2 u = 0,$$
$$u(0) = I,\ u'(0) = 0,\ t \in (0, T] \qquad (1)$$

where, ω, the wave's angular frequency and I, the initial amplitude, are given constants.

- The exact solution of (1) is

$$u(t) = I \cos(\omega t)$$

- That is, u oscillates with constant amplitude I and angular frequency ω. The corresponding period of oscillations (i.e., the time between two neighboring peaks in the cosine function) is $P = 2\pi/\omega$. The number of periods per second is $f = \omega/(2\pi)$ and measured in the unit Hz. Both f and ω are referred to as frequency, but ω is more precisely named *angular frequency*, measured in rad/s.

- In vibrating mechanical systems modeled by (1), $u(t)$ represents a position or a displacement of a particular point in the system. The derivative $u'(t)$ then has the interpretation of velocity, and $u''(t)$ is the associated acceleration. All these quantities are of importance for modelling seismically induced landslides.

Harmonic Waves - Finite Difference Scheme

- To solve (1) numerically, we use a finite difference method, consisting of 4 steps.
  - ⇒ Step 1: Discretize the domain.
  - ⇒ Step 2: Write the system at the discrete points.
  - ⇒ Step 3: Replace derivatives by finite differences.
  - ⇒ Step 4: Formulate a recursive algorithm, or a linear system to be solved for the numerical solution.

- Step 1: Define a uniformly partitioned time mesh,

$$t_n = n\Delta t, \quad n = 0, 1, \ldots, N, \quad \Delta t = T/N_t$$

  and a mesh function u^n, for $n = 0, 1, \ldots, N_t$ that approximates the exact solution at the mesh points.

- Step 2: the ODE to be satisfied at each mesh point is then

$$u''(t_n) + \omega^2 u(t_n) = 0, \quad n = 1, \ldots, N_t \qquad (2)$$

- Step 3: Replace the derivative $u''(t_n)$ by a centered, second-order accurate[1] finite difference approximation

$$u''(t_n) \approx \frac{u^{n+1} - 2u^n + u^{n-1}}{\Delta t^2}$$

and insert this approximation into (2),

$$\frac{u^{n+1} - 2u^n + u^{n-1}}{\Delta t^2} = -\omega^2 u^n.$$

We also need to approximate the derivative initial condition by a centered difference of the same accuracy,

$$\frac{u^1 - u^{-1}}{2\Delta t} = 0. \qquad (3)$$

- Step 4: Assume we know u^{n-1} and u^n, then we can solve for

$$u^{n+1} = 2u^n - u^{n-1} - \Delta t^2 \omega^2 u^n \qquad (4)$$

[1] Proof by Taylor expansion...

by applying a computational algorithm successively for $n = 1, 2, \ldots, N_t - 1$. This in known as the leapfrog scheme.

- Initial step: when we have a second-order equation, we need a special treatment to compute the first step. for this, we can define a "ghost" value u^{-1} at $t = -\Delta t$ and use (3) to compute

$$u^1 = 2u^0 - u^1 - \Delta t^2 \omega^2 u^0,$$

since $u^{-1} = u^1$, implying

$$u^1 = u^0 - \frac{1}{2}\Delta t^2 \omega^2 u^0. \qquad (5)$$

Harmonic Waves - Computational Algorithm

The steps for solving (1) numerically are:

1. $u^0 = I$
2. Compute u^1 from (5).
3. For $n = 1, 2, \ldots, N_t - 1$ compute u^{n+1} from (4).

This can be very easily coded in Python in just 7 lines:

```
t = np.linspace(0, T, Nt+1)  # mesh points in time
dt = t[1] - t[0]             # constant time step
u = np.zeros(Nt+1)           # solution
u[0] = I
u[1] = u[0] - 0.5*dt**2*w**2*u[0]
for n in range(1, Nt):
    u[n+1] = 2*u[n] - u[n-1] - dt**2*w**2*u[n]
```

Harmonic Waves - 1st Order System

- It is often judicious to rewrite a second-order equation as a system of two first-order equations.

  $\Rightarrow$ Consider the 2nd order ODE

$$u'' + \omega^2 u = 0, \quad u(0) = I,\ u'(0) = 0,$$

  $\Rightarrow$ Define the auxiliary variable $v = u'$ and rewrite the ODE in terms of first-order derivatives of u and v :

$$\begin{aligned} u' &= v, \\ v' &= -\omega^2 u \end{aligned}$$

  $\Rightarrow$ The initial conditions become $u(0) = I$ and $v(0) = 0$.

- We can now apply a very basic, forward Euler scheme to approximate this system

$$u^{n+1} = u^n + \Delta t v^n,$$
$$v^{n+1} = v^n - \Delta t \omega^2 u^n$$

- Other FD schemes are possible:

  $\Rightarrow$ backward Euler
  $\Rightarrow$ Crank-Nicolson
  $\Rightarrow$ Runge-Kutta

- Generalizations: this basic, 2nd order ODE can be generalized to take into account

  $\Rightarrow$ damping
  $\Rightarrow$ nonlinearity
  $\Rightarrow$ excitation

WAVE EQUATIONS

Wave Equation in 1D

The wave equation describes numerous propagation phenomena and is widely used in acoustics, seismics and vibration modeling, in general. It is second-order in both time and space, and in the one-dimensional case is defined by the initial boundary value problem

$$\begin{cases} \dfrac{\partial^2 u}{\partial t^2} - c^2 \dfrac{\partial^2 u}{\partial x^2} = s(x,t), \quad x \in (a,b), \quad 0 < t \leq T, \\ u(x,0) = f(x), \quad \dfrac{\partial u}{\partial t}(x,0) = g(x), \quad x \in (a,b), \\ u(a,t) = 0, \quad u(b,t) = 0, \quad t > 0, \end{cases} \tag{6}$$

- Initial conditions are required on u and $\partial u / \partial t$.
- Boundary conditions can be:

$\Rightarrow$ Dirichlet, where the amplitude u is prescribed.

- $\Rightarrow$ Neumann, where the normal derivative of u is prescribed.
- $\Rightarrow$ Mixed, where linear combination of Dirichlet and Neumann, is used.
- $\Rightarrow$ Absorbing, where the waves can pass straight through the boundaries without any reflections. This is used in practice to limit the size of the computational domain.

Wave Equation - Finite Difference Scheme

Following the same 4 steps as for the harmonic wave equation, we get a second-order accurate leapfrog scheme with centered differences in both time and space,

$$\frac{u_i^{n+1} - 2u_i^n + u_i^{n-1}}{k^2} = c^2 \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{h^2},$$

which gives the explicit recursion

$$u_i^{n+1} = r^2 u_{i-1}^n + 2(1 - r^2) u_i^n + r^2 u_{i+1}^n - u_i^{n-1}, \quad (7)$$

where the following stability condition must be imposed

$$r = c\frac{k}{h} \leq 1.$$

- In practice, we fix the spatial discretization step, h, then we impose the stability constraint on the time discretization step

$$k \leq \frac{\beta}{\sqrt{d}}\frac{h}{c}, \quad d = 1, 2, 3,$$

where d is the space dimension, and $\beta \approx 0.75$ is a safety factor that is necessary to prevent numerical round-off, and errors due to discontinuous material properties.

Wave Equation - Computational Algorithm

```
# Given mesh points as arrays x and t (x[i], t[n])
dx = x[1] - x[0], dt = t[1] - t[0]
C = c*dt/dx            # Courant number
Nt = len(t)-1
# Set initial condition u(x,0) = I(x)
for i in range(0, Nx+1):
    u_n[i] = I(x[i])
# Apply special formula for first step,  incorporating du/dt=0
for i in range(1, Nx):
    u[i] = u_n[i] - \
           0.5*C**2(u_n[i+1] - 2*u_n[i] + u_n[i-1])
u[0] = 0;  u[Nx] = 0   # Enforce boundary conditions
# Switch variables before next step
u_nm1[:], u_n[:] = u_n, u
for n in range(1, Nt):
    # Update all inner mesh points at time t[n+1]
    for i in range(1, Nx):
        u[i] = 2u_n[i] - u_nm1[i] - \
               C**2(u_n[i+1] - 2*u_n[i] + u_n[i-1])
    # Insert boundary conditions
    u[0] = 0;  u[Nx] = 0
    # Switch variables before next step
    u_nm1[:], u_n[:] = u_n, u
```

- As above, for the harmonic wave equation, the steps are identical and the numerical finite-difference problem can be solved with about 10 lines of code only.

- A vectorized version can be written and will have very high computaional efficiency - for example, solving a problem with $N_x = 10^4$ unknowns and $N_t = 1.5 \times 10^3$ time steps, takes only 0.5 seconds on a laptop.

- The generalization to higher dimensions (2 or 3) is straightforward, following exactly the same steps.

FD - Matrix Formulation

- Everything can be formulated in matrix-vector form, yielding a linear system of equations for each time-step.

- Recall the explicit recursion (7) for the wave equation

$$u_i^{n+1} = r^2 u_{i-1}^n + 2(1-r^2)u_i^n + r^2 u_{i+1}^n - u_i^{n-1},$$

- Define the FD matrix A and the vector unknown $\mathbf{U}^n$ at time n,

$$A = \begin{bmatrix} 2(1-r^2) & r^2 & & & \\ r^2 & \ddots & \ddots & & \\ & \ddots & \ddots & r^2 \\ & & r^2 & 2(1-r^2) \end{bmatrix}, \; \mathbf{U}^n = \begin{bmatrix} U_1^n \\ U_2^n \\ \vdots \\ U_{I-2}^n \\ U_{I-1}^n \end{bmatrix},$$

then the recursion becomes

$$\mathbf{U}^{n+1} = AU^n - \mathbf{U}^{n-1}.$$

FD - Matrix Formulation in 2D

- Suppose that $u(x, y)$ is the solution of the two-dimensional Poisson equation

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = f \tag{8}$$

over a bounded domain, Ω.

- We discretize both spatial derivatives with a second-order centered difference

$$\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{h^2} = f_{i,j},$$

which can be rewritten

$$\frac{1}{h^2}\left[-4u_{i,j} + u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1}\right] = f_{i,j},$$

where $i, j = 1, \ldots, I$ supposing that we have a constant step size $h = 1/(I + 1)$.

- This is known as the *five-point stencil* for the Laplacian.

- We can rewrite the problem in classical matrix-vector form,

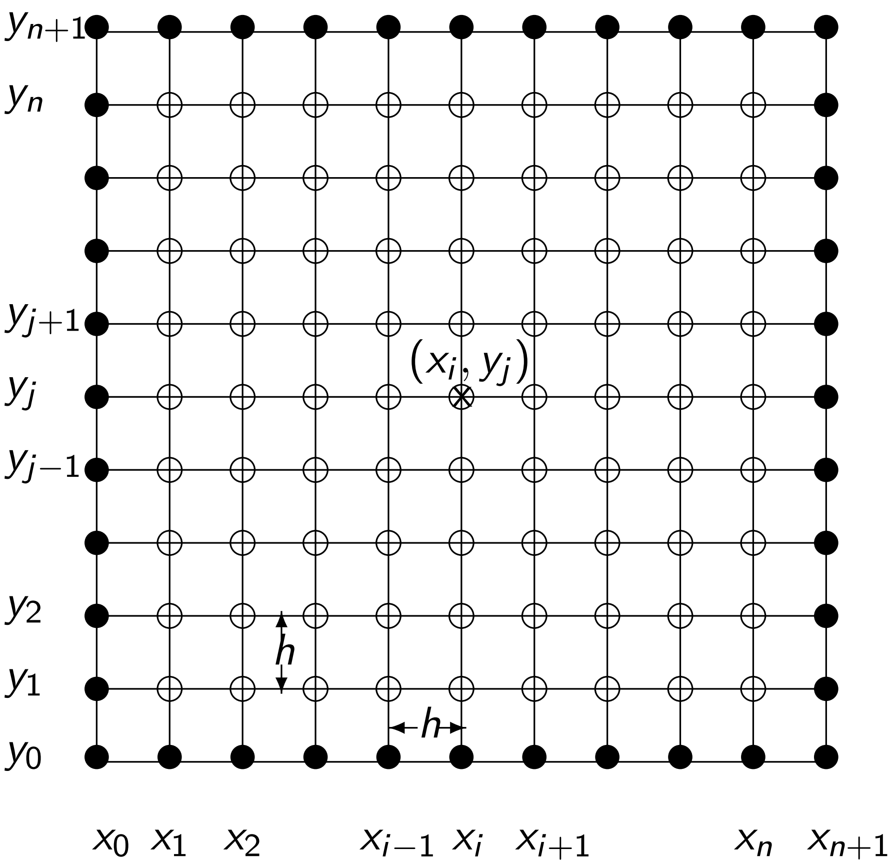
$$A\mathbf{u} = \mathbf{b}, \tag{9}$$

with

$$A = \frac{1}{h^2}\begin{bmatrix} -4 & 1 & 0 & 1 & 0 \\ 1 & -4 & 1 & \ddots & 1 \\ 0 & 1 & \ddots & \ddots & 0 \\ 1 & \ddots & \ddots & \ddots & 1 \\ 0 & 1 & 0 & 1 & -4 \end{bmatrix}, \ \mathbf{u} = \begin{bmatrix} u_{1,1} \\ \vdots \\ \vdots \\ \vdots \\ u_{n,n} \end{bmatrix},$$

$$\mathbf{b} = \begin{bmatrix} f_{1,1} \\ \vdots \\ \vdots \\ \vdots \\ f_{n,n} \end{bmatrix},$$

where the entries of $\mathbf{u}$ and $\mathbf{b}$ are in lexicographic order, left to right from bottom to top---see Figure below.



y_{n+1}
y_n
y_{j+1}
y_j
y_{j-1}
y_2
y_1
y_0
(x_i, y_j)
h
h
x_0 x_1 x_2 x_{i-1} x_i x_{i+1} x_n x_{n+1}

Seismic Wave Equation

- Seismic waves are used to infer properties of subsurface geological structures.

- The physical model is a heterogeneous, elastic medium where sound is propagated by small elastic vibrations.

- The general mathematical model for deformations in an elastic medium is based on: Newton's second law and a constitutive law relating stress to displacement.

- We define:

  ⇒ $\boldsymbol{u} = (u_1, u_2, u_3)^{\mathrm{T}}$ as the displacement in the x-, y- and z-direction,
  ⇒ λ and μ are the Lamé coefficients,
  ⇒ ρ is the medium density,
  ⇒ and $\boldsymbol{f}$ is an initial impulse that represents the acoustic source.

$$\rho \frac{\partial^2 u_i}{\partial t^2} = \frac{\partial \sigma_{ij}}{\partial x_j} + f_i$$

$$\sigma_{ij} = \lambda \delta_{ij} \epsilon_{kk} + 2\mu \epsilon_{ij}$$

$$\epsilon_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right)$$

- The pressure (primary) and shear (secondary) wave speeds are related to the Lamé coefficients,

$$c_p^2 = \frac{\lambda + 2\mu}{\rho} \qquad \text{and} \qquad c_s^2 = \frac{\mu}{\rho},$$

- Non-isotropic rheologies are important for realistic applications:

⇒ viscoelastic material,
⇒ anisotropic material, and
⇒ poro-elasticity.

- Seismic sources: in addition to the structural parameters of the Earth model, the physical description of the seismic source parameters will affect the resulting wavefield.

$\Rightarrow$ Forces and moments.

$\Rightarrow$ Point sources.

NUMERICAL METHODS

Categories

- Finite differences (seen above): for researchers who are interested in understanding partial differential equations, the finite-difference method offers an efficient and fast way to develop numerical approximations that allow the investigation of some of the main characteristics of the underlying modeled problem. Finite differences are always used for the time-discretization. For the spatial interpolation problem, there is an additional class of methods.

- Finite elements:

  $\Rightarrow$ classical - low-order, continuous across element boundaries.

  $\Rightarrow$ **spectral** - combination of Lagrange polynomials as interpolants and an integration scheme based on Gauss quadrature defined on the GLL points for the elastic wave equation. This leads to a diagonal mass matrix that can be trivially inverted.

⇒ discontinuous Galerkin - combines spectral with finite volume flux scheme.

FINITE ELEMENTS

Background and Motivation

- FEM is essentially an approximation of spatial derivatives (see below)

- Wave propagation is time- and space-dependent

- How do we use FEM for Wave Propagation?

  $\Rightarrow$ FEM for the spatial operators (usually a Laplacian)
  $\Rightarrow$ FDM for the time evolution (see above!)

- Why use FEM?

  $\Rightarrow$ flexible and versatile implementation (operators, discontinuities, complex geometries)
  $\Rightarrow$ solid theoretical basis (functional analysis)
  $\Rightarrow$ software packages for seismic wave propagation (SPECFEM)

Formulation: strong

We consider the general boundary value problem

$$\begin{cases} -\sum_{i,j=1}^{n} \frac{\partial}{\partial x_j}\left(a_{ij}(x)\frac{\partial u}{\partial x_j}\right) + \sum_{i=1}^{n} b_i(x)\frac{\partial u}{\partial x_j} + c(x)u = f(x), \\ u = 0, \quad x \in \partial\Omega, \end{cases} \tag{10}$$

where the conditions on the parameters for a weak solution are (for a classical/stong solution)

- Ω bounded domain in $\mathbb{R}^n$, $n = 1, 2, 3$:

  $\Rightarrow$ $a_{ij}(x) \in L_\infty(\Omega)$; $(C^1(\Omega))$
  $\Rightarrow$ $b_i(x) \in L_\infty(\Omega)$; $(C(\Omega))$
  $\Rightarrow$ $c \in L_\infty(\Omega)$, $f \in L_2(\Omega)$. $(C(\Omega))$

- i.e. all coefficients need only to be bounded and integrable

Formulation: weak

The weak formulation of problem (10) is:

Weak BVP

Find $u \in H_0^1(\Omega)$ such that

$$a(u,v) = l(v), \quad \forall v \in H_0^1(\Omega), \tag{11}$$

- The bilinear form (using integration by parts on the 1st term, and zero BC on $\partial\Omega$)

$$a(u,v) = \sum_{i,j=1}^{n} \int_\Omega a_{ij} \frac{\partial u}{\partial x_i}\frac{\partial v}{\partial x_j} \mathrm{d}x + \sum_{i=1}^{n} \int_\Omega b_i \frac{\partial u}{\partial x_i} v \, \mathrm{d}x + \int_\Omega cuv \, \mathrm{d}x \tag{12}$$

- and the linear form

$$l(v) = \int_\Omega f(x)v(x)\,\mathrm{d}x.$$

- The function space, H_0^1, is the space of square integrable functions, zero on the boundary, whose first derivatives are also square integrable.

Weak formulation

The weak, or variational formulation, being based on an **integral**, has weaker requirements on the smoothness of the functions, and can be shown to be equivalent to a minimization of an energy formulation, hence it is also variational. Hence, discontinuous functions and solutions can be rigorously dealt with, which is not the case (in principle) with finite difference methods.

Formulation: existence and uniqueness

We have a rigorous theorem for the existence and uniqueness of a weak solution to (11) .

Theorem 1 (Lax-Milgram Lemma). *If*

$$c(x) - \frac{1}{2}\sum_{i=1}^{n} \frac{\partial b_i}{\partial x_i} \geq 0, \quad x \in \Omega$$

then $\exists!\, u \in H_0^1(\Omega)$ *solution of (11) and we have the estimation*

$$\|u\|_{H^1(\Omega)} \leq \frac{1}{c_0} \|f\|_{L_2(\Omega)}$$

where c_0 *depends on a constant from the Poincaré-Friedrichs inequality that provides a bound for the function by its derivatives.*

Formulation: discretization

- The geometrical domain Ω is subdivided into a finite number of subdomains, called finite elements, and the solution is approximated on each element by a simple polynomial function. The overall solution is then the sum of these piecewise solutions. The elements themselves have simple geometrical shapes, usually triangles in 2D and tetrahedrons in 3D. The system that results is, as for FD, an algebraic one, a linear or nonlinear system of n equations in n unknowns that are the coefficients of the polynomial approximation.

- For the discrete formulation:
  - ⇒ Let V_h be the space of piecewise continuous polynomials, a finite-dimensional subspace of $H_0^1(\Omega)$.
  - ⇒ Then the approximation by finite elements of (11) is: "Find $u_h \in V_h$ such that

$$a(u_h, v_h) = l(v_h), \quad \forall\, v_h \in V_h." \tag{13}$$

- Existence and uniqueness of a solution is obtained, once again, from the Lax-Milgram Lemma, since $V_h \subset H_0^1(\Omega)$.

FEM for Wave Equation

- Model wave equation

$$\begin{aligned} \ddot{u} - c^2 \triangle u &= f, \quad \text{in } \Omega \times J, \\ u &= 0, \quad \text{on } \partial\Omega \times J, \\ u &= u_0, \quad \text{in } \Omega,\ t = 0, \\ \dot{u} &= u_1, \quad \text{in } \Omega,\ t = 0, \end{aligned}$$

  where $J =]0, T]$, c^2 is a parameter, f is a given source function, u_0 and u_1 are given initial condition

- Multiply by a test function $v \in V_0 = \left\{v \colon v \in H_0^1(\Omega)\right\}$, integrate by parts using Green's formula in dimension $n = 2, 3$

- Variational formulation: find $u \in V_0$ t such that for all $t \in J$ fixed,

$$\int_\Omega \ddot{u} v \,\mathrm{d}x + c^2 \int_\Omega \nabla u \cdot \nabla v \,\mathrm{d}x = \int_\Omega f v \,\mathrm{d}x, \quad v \in V_0,\ t \in J$$

- Spatial discretization

$$\int_\Omega \ddot{u}_h \phi_i \,\mathrm{d}x + c^2 \int_\Omega \nabla u_h \cdot \nabla \phi_i \,\mathrm{d}x = \int_\Omega f \phi_i \,\mathrm{d}x,$$
$$i = 1, \ldots, n_i, \ t \in J$$

where ϕ_i are piecewise-linear ("hat") basis functions for $V_{h,0}$ and n_i is the number of interior nodes

- Galerkin approximation

$$u_h(x,t) = \sum_{j=1}^{n_i} u_j(t) \phi_j(x)$$

- Substitute in the discretized formulation

$$\int_0^L f \phi_i \,\mathrm{d}x = \sum_{j=1}^{n_i} \ddot{u}_j(t) \int_\Omega \phi_j \phi_i \,\mathrm{d}x$$
$$+ c^2 \sum_{j=1}^{n_i} u_j(t) \int_\Omega \nabla \phi_j \cdot \nabla \phi_i \,\mathrm{d}x,$$
$$i = 1, \ldots, n_i, \ t \in J$$

- Introduce the notation

$$M_{ij} = \int_I \phi_j \phi_i \,\mathrm{d}x$$

$$K_{ij} = \int_I \phi_j' \phi_i' \,\mathrm{d}x$$

$$b_i = \int_0^L f \phi_i \,\mathrm{d}x$$

- Finally, we obtain the matrix system

$$\boxed{M\ddot{\mathbf{u}}(t) + c^2 K\mathbf{u}(t) = \mathbf{b}(t), \quad 0 < t \leq T}$$

with

⇒ the mass matrix, M,
⇒ the stiffness matrix, K and
⇒ the load vector $\mathbf{b}(t)$.

- **Remarks**:

⇒ at each time step, the matrix M has to be inverted

⇒ for a stationary problem, with no time evolution, this equation becomes

$$c^2 K\mathbf{u} = \mathbf{b}$$

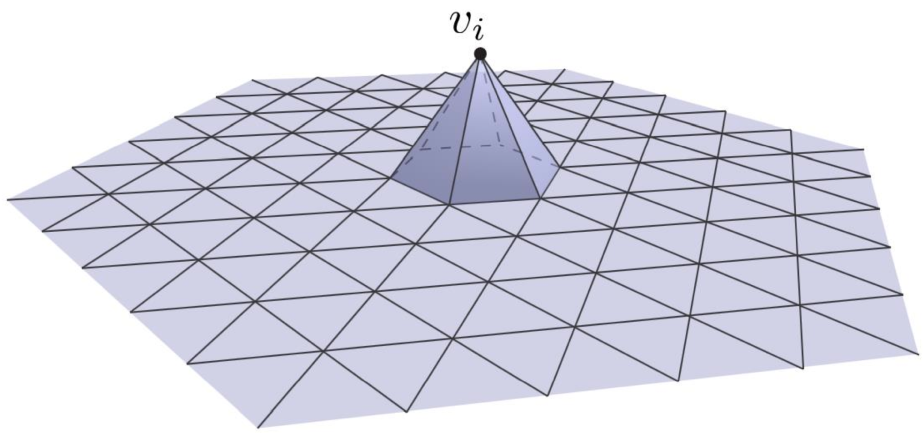
which has the same structure as the matrix-vector formulation (9) for Poisson's equation seen above in the finite-difference section.

⇒ for a heterogeneous medium, where c is a function (even discontinuous) of x, we simply put the function under the integral, and that's all.

⇒ the time evolution is discretized using finite-difference stencils or ODE-inspired schemes, and this mixture of FDM-FDM is called the "method of lines."

⇒ there are 2 topics that we have not discussed: (please consult the references for these)

1. mesh generation—usually performed by specialized software,
2. polynomial interpolation (so-called shape, or basis functions) on which the FEM approximation is based, and which produces a rigorous convergence theorem that describes how the FEM discretization error tends to zero as a function of the mesh size.



Example of a hat function—piecewise linear polynomial—with local support.

SPECTRAL FINITE ELEMENTS

Background and Motivation

- The spectral finite element method uses high-order piecewise polynomial approximations of Lagrange type, usually of order four to nine, in the weak formulation of time-evolution problems.

- When coupled with Gauss-Lobatto-Legendre quadrature rules for evaluation of the integrals, and hexahedral elements, the resulting mass matrix, M, is diagonal and thus trivial to invert.

- A further advantage is that the mesh size can be larger than what is normally required, in particular when the number of mesh points per wavelength is critical in wave propagation problems of high frequency signals.

- There is, however, much more computational work at the element level.

Formulation

Consider the 1D elastic wave equation in a domain $\Omega \times (0, T]$,

$$\rho \frac{\partial^2 u}{\partial t^2} = \frac{\partial}{\partial x}\left(\mu \frac{\partial u}{\partial x}\right) + f, \tag{14}$$

where

- displacement u, external force f, mass density ρ, shear modulus μ depend on x
- u and f depend on t also.
- boundary conditions are:

  $\Rightarrow$ stress-free condition on the Earth's surface

  $$\sigma_{ij} n_j = 0,$$

  where σ_{ij} is the symmetric stress tensor and n_j is the unit normal vector in direction x_j

$\Rightarrow$ spatial boundary conditions on the boundary $\partial\Omega$, either of Neumann type

$$\left|\mu\frac{\partial u}{\partial x}\right|_{x=0,L} = 0,$$

or an absorbing (transparent) condition.

Formulation - steps

1. Formulate the weak form of the wave equation (14) and its boundary conditions.

2. Transform the weak equation to the element level, using a Jacobian.

3. The discretization of our system comes with the approximation of the unknown function u using Lagrange polynomials as interpolants.

4. The formulation also requires the evaluation of the first derivatives of the Lagrange polynomials, the calculation of which requires the Legendre polynomials.

5. The numerical integration scheme based on GLL quadrature allows us to calculate all system matrices at elemental level.

6. Assemble in a final step to obtain the global system of equations that is integrated over time using a simple finite-difference scheme.

Formulation - weak form

As in the classical finite element method,

- we multiply both sides of (14) by a time-independent test function $v(x)$

$$\int_\Omega \rho \ddot{u} v \,\mathrm{d}x - \int_\Omega \frac{\partial}{\partial x}\left(\mu \frac{\partial u}{\partial x}\right) v \,\mathrm{d}x = \int_\Omega f v \,\mathrm{d}x \quad (15)$$

- integrate (15) by parts and use the Neumann boundary conditions

$$\int_\Omega \rho \ddot{u} v \,\mathrm{d}x + \int_\Omega \mu \frac{\partial u}{\partial x}\frac{\partial v}{\partial x} \,\mathrm{d}x = \int_\Omega f v \,\mathrm{d}x \qquad (16)$$

- approximate the displacement field by a linear combination (superposition) of basis functions

$$u(x,t) \approx u^h(x,t) = \sum_{i=1}^{N_p} u_i(t)\phi_i(x) \qquad (17)$$

- substitute this approximation (17) in the weak form (16) to obtain an equation for the unknown coefficients $u_i(t)$

$$
\begin{aligned}
&\sum_{i=1}^{N_p} \ddot{u}_i(t) \overbrace{\int_\Omega \rho(x)\phi_i(x)\phi_j(x)\,\mathrm{d}x}^{M_{ij}} \\
&+ \sum_{i=1}^{N_p} u_i(t) \overbrace{\int_\Omega \mu(x)\frac{\partial \phi_i(x)}{\partial x}\frac{\partial \phi_j(x)}{\partial x}\,\mathrm{d}x}^{K_{ij}} \\
&= \underbrace{\int_\Omega f(x,t)\phi_i(x)\,\mathrm{d}x}_{f_i}
\end{aligned}
$$

where the Galerkin method uses the same expansion for the test function v in terms of the basis functions ϕ.

Formulation - system

- This is the well-known equation for time-dependent finite-element problems, which can be rewritten in matrix-vector notation as

$$M\ddot{\mathbf{u}}(t) + K\mathbf{u}(t) = \mathbf{f}(t)$$

- The mass matrix , M, is diagonal, thus its inversion is trivial.

- The stiffness matrix, K, has a banded structure in this case with the bandwidth depending on the number of basis functions that are required inside each element.

- A simple centred finite-difference approximation of the second derivative in time gives the time-stepping scheme

$$\mathbf{u}^{n+1} = (\Delta t)^2 \left[M^{-1}\left(\mathbf{f} - K\mathbf{u}^n\right)\right] + 2\mathbf{u}^n - \mathbf{u}^{n-1}$$

Formulation - element level

Without going into details (see the References):

- the basis functions $\phi(x)$ are defined element-by-element, with local support which ensures sparsity of the resulting stiffness matrix
- the interpolation used is based on Lagrange polynomials which guarantee high-order, very accurate interpolation
- the integration, element-by-element, is performed using a special Gauss-Lobatto-Legendre (GLL) quadrature rule
- finally, the element matrices are assembled into a sparse, global matrix

Remark 1. The resulting code is very complex and should never be coded from scratch, unlike the finite difference method that is far simpler and can indeed be coded "by hand."

References

1. H.P. Langtangen. *Finite Difference Computing with PDEs: A Modern Software Approach.* Springer Cham. 2017. Download pdf

2. H. Igel. *Computational Seismology: A Practical Introduction.* Oxford University Press, 2017. website

3. SPECFEM2D

4. SPECFEM3D