Computational Seismology

Lecture 3: Finite-difference Method

February 24, 2021

University of Toronto

TABLE OF CONTENTS

1. History of FD
2. Finite-difference approximation to derivatives
3. FD for 1D Acoustic wave equation
4. FD for 2D Acoustic wave equation
5. Elastic wave propagation in 2D
6. FD for 3D wave propagation
7. Miscellaneous Subjects related to FD

History of FD

Finite-difference method

1. Conceptually simple: brute-force (strong formulation)
2. Quickly adaptable to specific problems (exploration geophysics, strong ground motion dynamic rupture simulations, FWI).

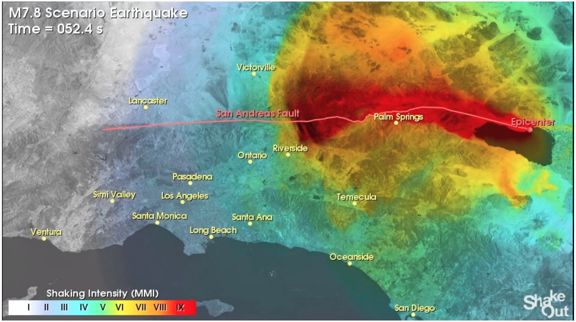


Figure 1: A magnitude 7.8 scenario eqk (South California ShakeOut) ruptures 300 km of the San Andreas fault with final slip ranging from 2-7 m).

History of Finite-difference - 1

- First applications of FD: layered medium in cylindrical coordinates (Alterman and Karal 1968); simulate Love waves (snapshots) by Boore (1970)
- Acoustic equations (Alford et al 1974), **elastic equations** (Kelly et al 1976)
- **Staggered-grid formulation**: introduced to solve ruture propagation problem (Madariaga 1976, Virieux and Madariaga, 1982), elastic SH/P-SV waves (Virieux, 1984, 1986)
- Parallel computing allowed for **3D applications**: Frankel and Vidale (1992), Graves (1993), Olsen and Archuleta (1996), and Pitarka and Irikura (1996)
- Other rheology: viscoelastic (Day and Minster, 1984, Emmerich and Korn, 1987, Robertsson et al., 1994) and anisotropic (Igel et al., 1995)

History of Finite-difference - 2

- **Spherical coordinates** for global waves: first with the axisymmetric approximation (Igel and Weber, 1995, Igel and Weber, 1996, Chaljub and Tarantola, 1997), 3D spherical sections (Igel et al., 2002).
- **Frictional boundary condition** for dyanmic rupture analysis (Olsen et al., 1997); Failed node based on threshold criterion (Nielsen and Tarantola 1992)
- Free-surface boundary with **strong topography**: volcanology (Ohminato and Chouet, 1997), viscoelastic (Robertsson and Holliger, 1997), modified operators or hybrid schemes (Moczo et al., 2014); strongly heterogeneous media (Moczo et al. 2002)
- **FWI**: in 2D Crase et al. (1990), and in 3D (Chen et al., 2007). FD is the prevailing method for forward solver for FWI in exploration seismology (Virieux and Operto, 2009).

Finite-difference approximation to derivatives

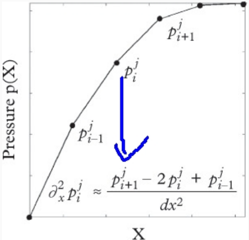
Finite-difference method: introduction

In a nutshell, space and time are both discretized (usually) on regular space–time grids in FD. It is a grid-based method as field values are ONLY known at these grid points. Partial derivatives are replaced by finite-difference formulas.

$$\partial_t^2 p(x,t) = c^2(x)\partial_x^2 p(x,t) + s(x,t) \quad (1)$$

$$\partial_t^2 p(x,t) \approx \frac{p(x,t+dt) - 2p(x,t) + p(x,t-dt)}{dt^2} \quad (2)$$

Extrapolation in time: The pressure field at $t + dt$ updated from field at t and t-dt at the nearest neighbours (easily adaptable in parallel).



Finite Differencing formulas

Forward differencing

$$\frac{df^+}{dx} \approx \frac{f(x+dx)-f(x)}{dx} \tag{3}$$

Centered differencing

$$\frac{df^c}{dx} \approx \frac{f(x+dx)-f(x-dx)}{2dx} \tag{4}$$

Backward differencing

$$\frac{df^-}{dx} \approx \frac{f(x)-f(x-dx)}{dx} \tag{5}$$

How accurate are these differencing formulas?

Accuracy of differencing formulas

Based on Taylor expansion

$$f(x+dx) = f(x) + f'(x)dx + \frac{1}{2}f''(x)dx^2 + O(dx^3) \tag{6}$$

Hence the central differencing scheme is an order of magnitude more accurate (converges more rapidly as $dx \to 0$):

$$d_x f^+ = f'(x) + O(dx), \quad d_x f^- = f'(x) + O(dx), \quad d_x f^c = f'(x) + O(dx^2),$$

Higher-order derivatives

$$\partial_x^2 f \approx \frac{f(x+dx) - 2f(x) + f(x-dx)}{dx^2} \tag{7}$$

which can be obtained through Taylor expansion by solving for a, b, c that

$$\frac{af(x+dx) + bf(x) + cf(x-dx)}{dx^2} \propto f''(x) + O(dx^2) \tag{8}$$

where $[a, b, c] = [1, -2, 1]$.

General differencing formulas

More neighbouring points (*domain of influence*) can be used.
For example, a five-point approximation

$$f''(x) + O(dx^4) =$$
$$[af(x + 2dx) + bf(x + dx) + cf(x) + df(x - dx) + ef(x - 2dx)]/dx^2$$

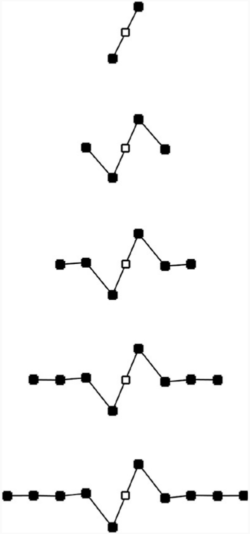
Coefficients can be determined by matching the coefficients of different orders in the Taylor expansion

$$[a, b, c, d, e] = [-1/12, 4/3, -5/2, 4/3, -1/12] \tag{9}$$

In practice, always use 5-point operator (4-point) for 2nd (1st) order derivatives.

Differencing formula

The differential weights rapidly decrease with distance from the central point of evaluation (white).



FD for 1D Acoustic wave equation

1D Acoustic waves for pressure $p(x,t)$ or waves on a 1D string:

$$\partial_t^2 p(x,t) = c(x)^2 \partial_x^2 p(x,t) + s(x,t) \tag{10}$$

with I.C., $p(x,t=0) = \partial_t p(x,t=0) = 0$ and B.C. such as $p(x)|_{x=0,L} = 0$ or others to be specified later.

Discretization in time and space $x_j = jdx, j = 0, \cdots, j_{max}$, $t_n = ndt, n = 0, \cdots n_{max}$, $p_j^n = p(x_j, t_n)$ (upper index for time discretization and lower index for spatial discretization).

Discretization of the PDE by replacing the PD with FD at p_j^n:

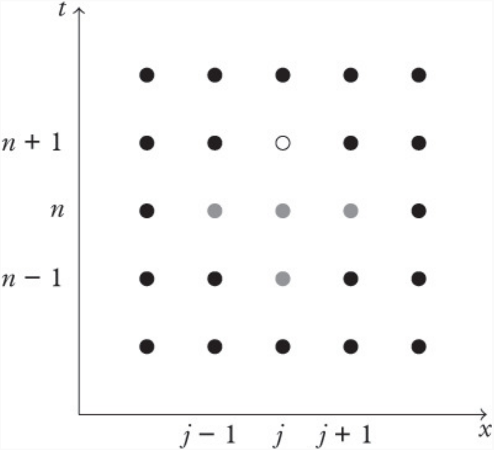
$$\frac{p_j^{n+1} - 2p_j^n + p_j^{n-1}}{dt^2} = c_j^2 \left[\frac{p_{j+1}^n - 2p_j^n + p_{j-1}^n}{dx^2} \right] + s_j^n \tag{11}$$

which can be reorganized into a formula for advancing in time

$$p_j^{n+1} = c_j^2 \frac{dt^2}{dx^2} [p_{j+1}^n - 2p_j^n + p_{j-1}^n] + 2p_j^n - p_j^{n-1} + dt^2 s_j^n \tag{12}$$

Advancing of wavefield in time

The field at $n+1$'th time can be computed based on the field at n'th time and $n-1$'th time, starting from the I.C.



Source time function

The wavefield is generated by enacting the source $s(x,t)$, e.g., source for Greens function $s(x,t) = \delta(x - x_s)\delta(t - t_s)$, or a band-limited point source $s(x,t) = \delta(x - x_s)f(t)$.

Discretization of the spatial delta into boxcar

$$\delta(x) \rightarrow \delta_{bc}(x) = \begin{cases} 1/dx & |x| < dx/2 \\ 0 & \text{elsewhere} \end{cases} \tag{13}$$

and discretization of temporal delta into Gaussian functions

$$\delta(t) \rightarrow \delta_a(t) = \frac{1}{\sqrt{2\pi a}} e^{-t^2/(2a)} \tag{14}$$

How to choose discretization

What determine the discretization scheme?

- what is the **dominant frequency** of the waves to be simulated?
- What is the minimum velocity inside the medium? Hence what is the minimum spatial wavelength that propagates inside the medium? $c = \frac{\omega}{k} = \frac{\lambda}{T} = \lambda f$, $\lambda = cT = c/f$.
- What is the propagation distance (in terms of number of dominant wavelength)?

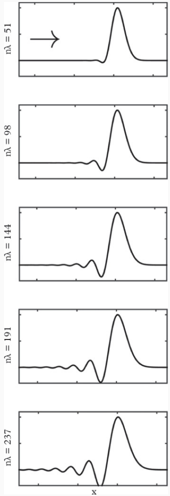
Example: sound waves with $f_0 = 20$ Hz (or 50 Hz) propagating in atomsphere with $c = 343$ m/s gives $\lambda = 17$m (or $\lambda = 7$ m), a choice of $dx = 0.5$ m gives ~ 34 (or 14) points-per-wavelength.

Python codes and snapshots for 1D acoustic waves by FD

Input STF: 1st der of Gaussian; *nt*

$nx = 20,000$ total number grid points in x; numerical dispersion

```
# Time extrapolation
for it in range(nt):
    # calculate partial derivatives (omit boundaries)
    for i in range(1, nx - 1):
        d2p[i] = (p[i + 1] - 2 * p[i]\
            + p[i - 1]) / dx ** 2
    # Time extrapolation
    pnew = 2 * p - pold + dt ** 2 * c ** 2 * d2p
    # Add source term at isrc
    pnew[isrc] = pnew[isrc] +  dt ** 2 * src[it] / dx
    # Remap time levels
    pold, p = p, pnew
```



Stability of the numerical solution

von Neumann analysis: assuming a trial solution $p(x,t) = e^{i(kx-\omega t)}$ and plug it into the FD formula (such that $p_j^{n+1} \to p_j^n e^{-i\omega\, dt}$ and $p_{j+1}^n \to p_j^n e^{ik\, dx}$) and ignoring the source term:

$$\sin(\omega\, dt/2) = c\frac{dt}{dx}\sin(k\, dx/2) \tag{15}$$

- Courant-Friedrichs-Lewy **(CFL) stability condition** on the dependency of space-time discretization (necessary but not sufficient)

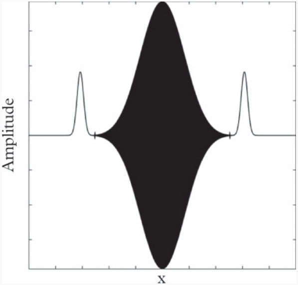
  $$\epsilon = c\frac{dt}{dx} \leq 1 \tag{16}$$

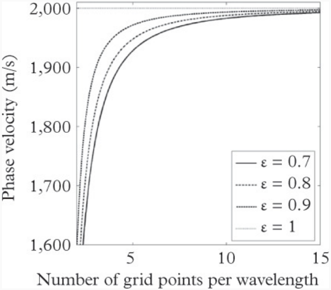
  Therefore, num-pts-per-wavelength (NPW) determines dx, and CFL then determines dt.

Stability analysis

- (unphysical) **numerical dispersion**: for a physical wavenumber k, FD will result in frequency ω of waves that depends on the choice of dx ($kdx = 2\pi/\text{npw}$). The phase velocity of these waves is dispersive and not identical to c. As npw $\uparrow$, $k\,dx \downarrow$, $c^{num}(k) \rightarrow c$.

$$c^{num}(k) = \frac{\omega}{k} = \frac{2}{k\,dt}\sin^{-1}[\epsilon \sin(k\,dx/2)] \neq c \qquad (18)$$





Simulation accuracy: depends on the NPW and overall propagation distance (more error accumulation). Convergence: $dt, dx \rightarrow 0$, $c^{num}(k) \rightarrow c$

FD for 2D Acoustic wave equation

Acoustic wave equation in 2D (with constant velocity)

Acoustic wave propagation in X-Z (vertical) plane

$$\partial_t^2 p(x,z,t) = c(x,z)^2(\partial_x^2 p(x,z,t) + \partial_x^2 p(x,z,t)) + s(x,z,t) \quad (19)$$

Discretization $p(x,z,t) \to p_{j,k}^n = p(ndt, jdx, kdz)$. Again using central differencing formula (for both time and space) for $p_{j,k}^n$ on a regular grid

$$\frac{p_{j,k}^{n+1} - 2p_{j,k}^n + p_{j,k}^{n-1}}{dt^2} = c_j^2(\partial_x^2 p + \partial_z^2 p) + s_{j,k}^n \quad (20)$$

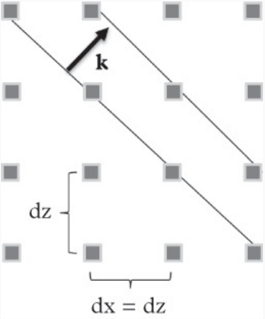
where

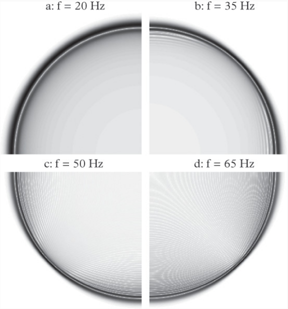
$$\partial_x^2 p = \frac{p_{j+1,k}^n - 2p_{j,k}^n + p_{j-1,k}^n}{dx^2}, \quad \partial_z^2 p = \frac{p_{j,k+1}^n - 2p_{j,k}^n + p_{j,k-1}^n}{dz^2} \quad (21)$$

2D Example: P-wave propagation

P-wave propagation in a reservoir scale model: $c_{max} = 5$ km/s and $c_{min} = 3$ km/s, $f_{dom} = 20$ Hz (energy up to 50 Hz can be present in the waveforms), dominant wavelength is at least λ_{dom}=c_{min}/f_{dom}=150 m.

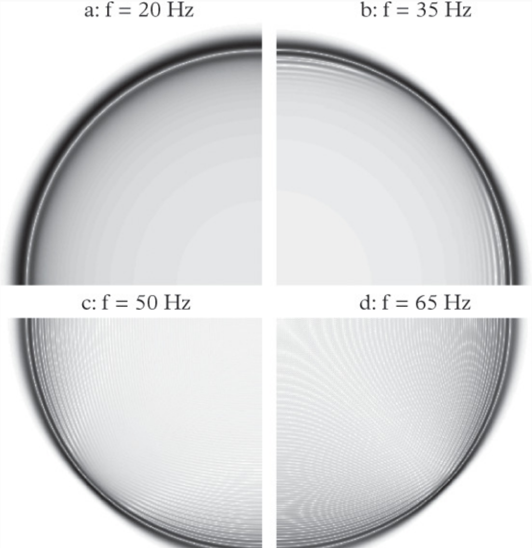
Simulation domain 5km $\times$ 5km, grid spacin $dx = 10$ m, resulting in 15 NPW for the dominant frequency.





Numerical Anisotropy

Numerical anisotropy can be observed for high-frequency waves: in certain directions the wavefield deteriorates faster

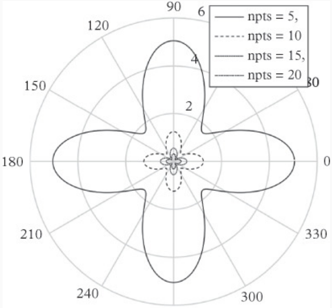


Stability analysis: Numerical anisotropy

Assuming $p(x,z,t) = e^{i(\mathbf{k}\cdot\mathbf{x}-\omega t)} = e^{i(k_x x + k_z z - \omega t)}$, we obtain numerical dispersion relationship

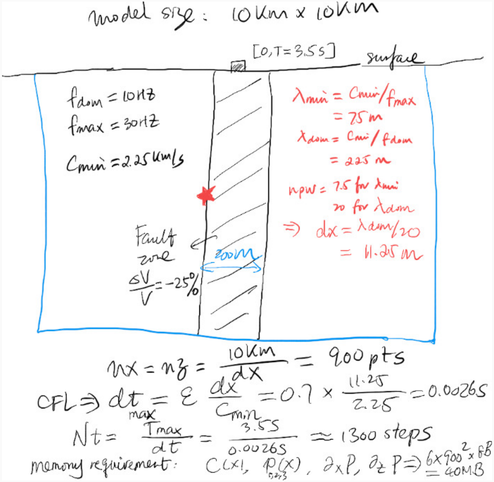
$$c^{num}(k_x,k_z) = \frac{2}{|k|dt}\sin^{-1}\left[\epsilon\left(\sin^2(\frac{k_x dx}{2}) + \sin^2(\frac{k_z dz}{2})\right)^{1/2}\right]$$

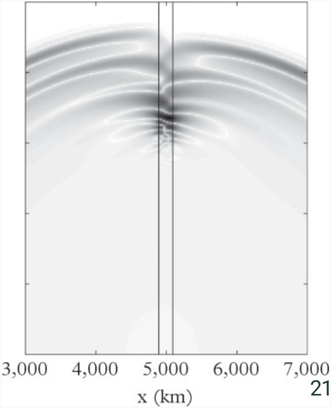
Phase velocity errors as a function of propagation direction $\mathbf{k} = [k_x, k_z] = |k|[\cos\alpha, \sin\alpha]$. Error decrease with increasing NPW, most accurate at $\alpha = 45°$. **Physical vs numerical** dispersion.



Example: fault-zone trapped waves

Trapped waves can be observed right above fault zones. Model setup and snapshot at $t = 2$ s (head waves at the edge of host medium, and amplified trapped waves in the fault zone)





1D elastic wave equations

Constitutive relationship in 1D for $u_y(x,t)$

$$\sigma_{ij} = \lambda \epsilon_{kk} \delta_{ij} + 2\mu \epsilon_{ij} \tag{22}$$

becomes

$$\sigma_{xy} = \sigma_{yx} = 2\mu \epsilon_{xy} = \mu \partial_x u_y \tag{23}$$

For simplicity, just use $u(x,t)$ to represent $u_y(x,t)$, and the 1D elastic wave equation becomes

$$\rho \partial_t^2 u = \partial_x (\mu \partial_x u) + f \tag{24}$$

and under discretization $u_i^j = u(idx, jdt)$,

$$\rho_i \frac{u_i^{j+1} - 2u_i^j + u_i^{j-1}}{dt^2} = \frac{\mu_{i+1} u_{i+2}^j - \mu_{i+1} u_i^j - \mu_{i-1} u_i^j + \mu_{i-1} u_{i-2}^j}{4\, dx^2} + f_i^j$$

Note u values on $i \pm 1$ are not used due to the asymmetry in central difference formula for 1st order derivatives. This inefficiency can be improved by **velocity-stress** formulation.

Velocity–stress formulation

Goal: as error $\sim O(dx^2)$, reducing dx by half, will give 1/4 of the error. Rewrite the wave equation into a coupled first-order PDE system for (v, σ)

$$\rho \partial_t v = \partial_x \sigma + f \tag{25}$$

$$\partial_t \sigma = \mu \partial_x v \tag{26}$$

and still discretize on the regular grid in time and space: centered at $(v_i^j, \sigma_{i+1/2}^j$ by **staggered-grid** for $v_i^{j+1/2} = v(i\,dx, (j+1/2)\,dt)$, and $\sigma_{i+1/2}^j = \sigma((i+1/2)\,dx, j\,dt)$.

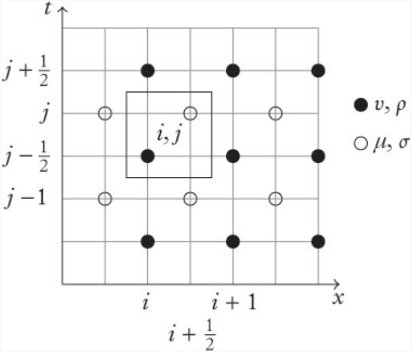
$$\frac{v_i^{j+1/2} - v_i^{j-1/2}}{dt} = \frac{\sigma_{i+1/2}^j - \sigma_{i-1/2}^j}{\rho_i dx} + \frac{f_i^j}{\rho_i} \tag{27}$$

$$\frac{\sigma_{i+1/2}^{j+1} - \sigma_{i+1/2}^j}{dt} = \mu_{i+1/2} \frac{v_{i+1}^{j+1/2} - v_i^{j+1/2}}{dx} \tag{28}$$

Extrapolation scheme

$$v_i^{j+1/2} = v_i^{j-1/2} + \frac{dt}{\rho_i dx}\left(\sigma_{i+1/2}^{j} - \sigma_{i-1/2}^{j}\right) + \frac{dt}{\rho_i} f_i^j$$

$$\sigma_{i+1/2}^{j+1} = \sigma_{i+1/2}^{j} + \frac{\mu_{i+1/2} dt}{dx}\left(v_{i+1}^{j+1/2} - v_i^{j+1/2}\right) \qquad (29)$$




```
# Time extrapolation
for it in range(nt):
    # Stress derivative
    for i in range(1, nx-1):
        ds[i] = (s[i+1] - s[i])/dx
    # Velocity extrapolation
    v = v + dt/rho*ds
    # Add source term at isx
    v[isx] = v[isx] + dt*src[it]/(dx*rho
    # Velocity derivative
    for i in range(1, nx-1):
        dv[i] = (v[i] - v[i-1])/dx
    # Stress extrapolation
    s = s + dt*mu*dv
```

Numerical Dispersion for velocity-stress formulation

Plug in $v = e^{i(kx-\omega t)}$ to the velocity-stress FD formulas

$$\sin(\frac{\omega dt}{2}) = \sqrt{\frac{\mu}{\rho}}\frac{dt}{dx}\sin\left(\frac{kdx}{2}\right) \tag{30}$$

which gives the numerical dispersion relation

$$c^{num}(k) = \frac{\omega}{k} = \frac{\lambda}{\pi dt}\sin^{-1}\left(c_0\frac{dt}{dx}\sin\frac{\pi dx}{\lambda}\right) \neq c_0 \equiv \sqrt{\frac{\mu}{\rho}} \tag{31}$$

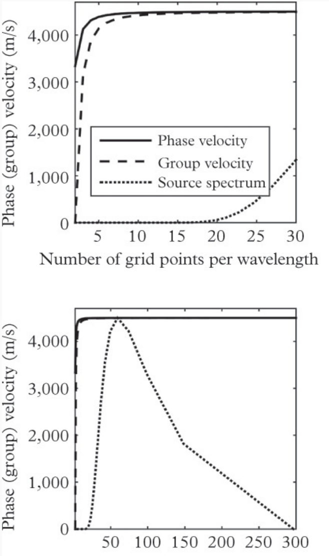
Or in terms of group velocity dispersion

$$c_g(k) = \frac{d\omega}{dk} = \frac{c_0\cos\frac{\pi dx}{\lambda}}{\left[1-\left(c_0\frac{dt}{dx}\sin\frac{\pi dx}{\lambda}\right)^2\right]^{1/2}} \tag{32}$$

Numerical dispersion

Plot numerical dispersion $c(k)$ and $c_g(k)$ as a function of NPW ($= \lambda/dx = 2\pi/kdx$)

Source spectrum with dominant frequency at $f_0 = 1/15$ Hz (npw=66). The results can be substantially improved by using a 4-point operator for derivatives.



Elastic wave propagation in 2D

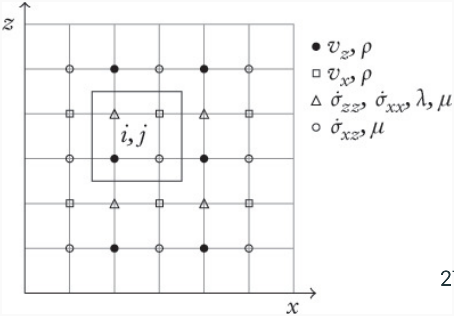
Staggered grid formulation

In 2D, $v_{x,z}(x, z; t)$, the time-derivative of Stress-strain relation $\partial_t \sigma_{ij} = \lambda \partial_t \epsilon_{kk} \delta_{ij} + 2\mu \epsilon_{ij}$, becomes ($\epsilon$ is the strain-rate tensor)

$$\partial_t \sigma_{xx} = (\lambda + 2\mu)\partial_x v_x + \lambda \partial_z v_z$$
$$\partial_t \sigma_{zz} = (\lambda + 2\mu)\partial_z v_z + \lambda \partial_x v_x$$
$$\partial_t \sigma_{xz} = \mu(\partial_x v_z + \partial_z v_x)$$

And EOM becomes

$$\begin{aligned} \rho \partial_t v_x &= \partial_x \sigma_{xx} + \partial_z \sigma_{xz} \\ \rho \partial_t v_z &= \partial_z \sigma_{xz} + \partial_z \sigma_{zz} \end{aligned}$$



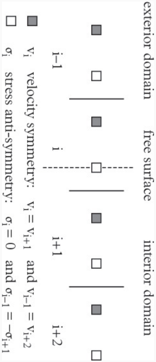
Boundary Condition

Free-surface boundary condition assumes zero traction at the surface $z = 0$

$$\partial_t \sigma_{zz} = \lambda \partial_x v_x + (\lambda + 2\mu) \partial_z v_z = 0$$

$$\partial_t \sigma_{zz} = \mu(\partial_x v_z + \partial_z v_x) = 0$$

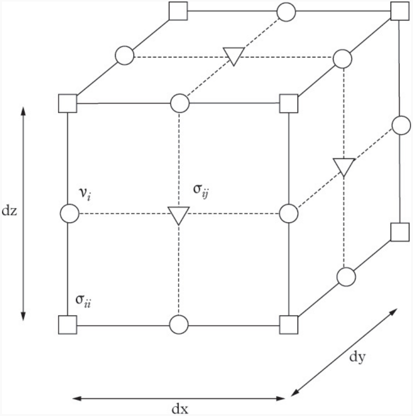
FD implementation: medium is extended beyond (above) the interior domain for as many points as required by the length of the FD operator.
Velocities are imposed to be symmetric (so that traction vanishes at the surface); stresses are extended beyond the free surface in an anti-symmetric way.
Not accurate enough for surface waves: one-sided approximation, hybrid solution etc



FD for 3D wave propagation

Staggered-grid in 3D

The classic 3D staggered grid



Miscellaneous Subjects related to FD

Higher-order time extrapolation scheme

So far the time extrapolation scheme is of lowest order. Take the first-order system equation as an example

$$\partial_t q = L(q,t) \equiv c\partial_x q(x,t) + s(x,t) \tag{33}$$

The lowerest order time extrapolation is equivalent to the **Euler's method**

$$q^{n+1} = q^n + L(q^n, t_n)dt \tag{34}$$

Extending it to high-order scheme (most packages use second-order extrapolation), such as the family of Runge-Kutta methods. One of them is the **predictor-corrector method**

$$\begin{aligned} q^+ &= q^n + L(q^n, t_n)dt \\ k_1 &= L(q^n, t) \quad \text{predictor} \\ k_2 &= L(q^+, t+dt) \quad \text{corrector} \\ q^{n+1} &= q^n + (k_1 + k_2)dt/2 \end{aligned} \tag{35}$$

Newmark scheme

The Newmark scheme (1959) is also a method of integration used to solve differential equations and used widely in finite-element analysis for structural dynamics. For the second-order equation

$$\ddot{u} = L(u, \dot{u}, t) \tag{36}$$

the explicit **Newmark scheme** (with special choice of parameters) is as follows:

1. Given $u^n, \dot{u}^n, \ddot{u}^n$, first compute the intermediate velocity and displacement $\dot{u}^+ = \dot{u}^n + \ddot{u}^n * dt/2$, and $u^+ = u^n + \dot{u}^n * dt$
2. then calculate the new acceleration based on the 2nd-order equation $\ddot{u}^{n+1} = L(u^+, \dot{u}^+, t_n)$
3. Update velocity and displacement $\dot{u}^{n+1} = \dot{u}^+ + \ddot{u}^{n+1} * dt/2$ and $u^{n+1} = u^+ + \ddot{u}^n * dt^2/2$

Explicit vs. implicit scheme

For simplicity, let us first consider the heat equation

$$Q_t = D\nabla^2 Q, \tag{37}$$

where $D = \frac{k}{\rho C}$ is the *thermal diffusivity*. In 1D,

$$Q_t = DQ_{xx}. \tag{38}$$

We introduce a compact way of writing all the wavefield at one time instance as a vector $\mathbf{Q}^n = \{Q_j^n\}, j = 1, \cdots, N$, we can then write the approximate spatial derivative $\frac{\partial^2}{\partial x^2}$ as

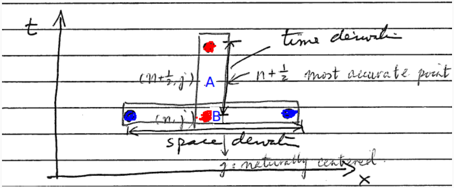
$$\frac{\partial^2 Q}{\partial x^2} \approx \frac{1}{h^2}[1, -2, 1] * Q = \frac{1}{h^2} TQ \tag{39}$$

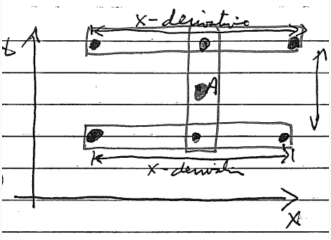
where T is the tri-diagonal matrix

$$\begin{bmatrix} -2 & 1 & & 0 \\ 1 & \ddots & \ddots & \\ & \ddots & \ddots & 1 \\ 0 & & 1 & -2 \end{bmatrix} \tag{40}$$

Implicit scheme

To get more centered derivatives, use central difference for the spatial derivatives





$$\frac{Q^{n+1} - Q^n}{\Delta t} = \frac{D}{h^2} T \left(\frac{Q^n + Q^{n+1}}{2} \right) \tag{41}$$

which upon rearrangement gives the **Crank-Nicholson methods**

$$\left(I - \frac{D\Delta t}{2h^2} T \right) Q^{n+1} = \left(I + \frac{D\Delta t}{2h^2} T \right) Q^n . \tag{42}$$

Both matrices in blue are tri-diagonal, and the solution at the same level (x) are solved simultaneously $\rightarrow$ **implicit method**.

Cauchy-Kowaleski procedure

Ignoring the source term, the derivatives of q also satisfies the equation

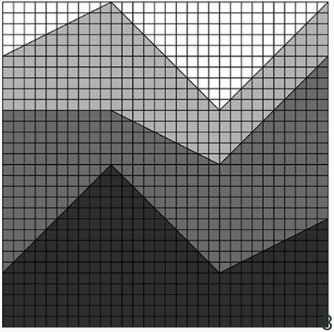
$$\partial_t^{j+1} q(x,t) = c\partial_x[\partial_t^j q(x,t)] \tag{43}$$

The time derivative of $q(x,t)$ of any order can be replaced by the spatial derivative recursively. It has been used to **Arbitrary high-orDER (ADER) schemes** for the finite-volume and discontinuous Galerkin methods (e.g. Titarev and Toro 2002; Dumbser and Munz, 2005a)

Heteogeneous Earth Model

For cases where interfaces (i.e. material discontinuities) are not aligned with the regular grid, the geometry of the interfaces are not accurately modelled. One solution is the finite (spectral)-element method. But there are remedies that can improve the FD approach

- Equivalent medium theory (Muir et al 1992): isotropic $\rightarrow$ anisotropic parameters
- discontinuous FD grid (Moczo et al 2010)
- spatially varying time steps for very heterogeneous models (Tessmer 2000)
- Homogenization (Capdeville et al 2010ab)



Optimization Operators

Artificially make errors in the space derivatives that compensate for the errors committed by the time extrapolation, to obtain a truly high-order scheme (Emmerich and Korn 1987). Another approach is to use derivative operators that are optimized for seismic wave equation:

Conventional ($1/dt^2$)

t+dt		1	
t		−2	
t−dt		1	
	x−dx	x	x+dx

Optimal ($1/dt^2$)

t+dt	1/12	10/12	1/12
t	−2/12	−20/12	−2/12
t−dt	1/12	10/12	1/12
	x−dx	x	x+dx

Conventional ($1/dx^2$)

t+dt			
t	1	−2	1
t−dt			
	x−dx	x	x+dx

Optimal ($1/dx^2$)

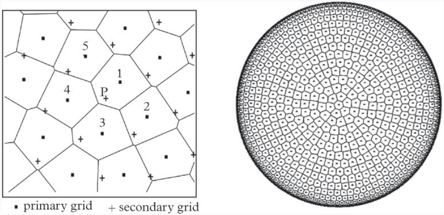
t+dt	1/12	−2/12	1/12
t	10/12	−20/12	10/12
t−dt	1/12	−2/12	1/12
	x−dx	x	x+dx

Smearing out conventional operator in space and time: leading to a locally implicit scheme, but also one order of magnitude more accurate.

Irregular grids for FD?

Is it possible to use irregular grid for FD schemes?

- in 2D: two staggered equilateral triangular grids; does not exist in 3D.
- differential weights for unstructured grid: near neighbour coordinates (Braun and Sambridge 1995, Kaser et al 2001, etc): low-order accuracy $\rightarrow$ **discontinous Galerkin method** for unstructured tetrahedral mesh.



FD on unstructured grids using Delauney triangulation and Voronoi cells.

Other coordinate systems

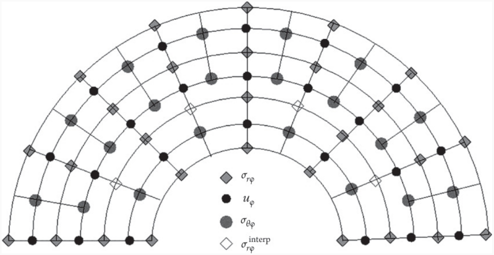
For the acoustic wave equation, assuming spherical coordinate system (r, θ, ϕ), and a model $c(r, \theta)$ and source $s(r, \theta)$ invariant in ϕ (i.e., zonal model or axisymmetric)

$$\ddot{p} = c^2 \left[\frac{1}{r^2} \partial_r (r^2 \partial_r p) + \frac{1}{r^2 \sin\theta} \partial_\theta (\sin\theta \partial_\theta p) \right] + s \qquad (44)$$

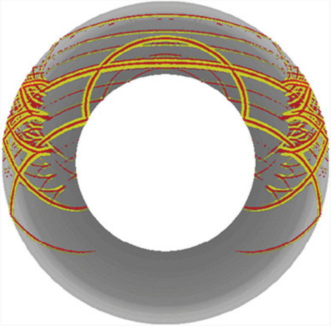
which is much more complicated than the cartesian case.

- Singularity at $\theta = 0$ (the pole)
- Regular discretization on *r* and *theta* leads to grid spacing decrease with depth, while given the velocity in the mantle, we want grid spacing to increase with depth $\rightarrow$ grid refinement towards the surface, smaller time steps

FD in spherical coordinate system



Axisymmetric mesh for staggered-grid 2D FD.



SH waves in the Earth's mantle (2D FD accurate to 25 s).

More wave simulation animations can be found online.