Computational Seismology: Simulating Seismic Wavefields for AlpArray

Heiner Igel

Department of Earth and Environmental Sciences
Ludwig-Maximilians-University Munich

Introduction

Goals of lecture

- Introduction to seismic waves in a discrete world
- Understand methods that allow the calculation of **seismic wavefields in heterogeneous media**
- Know **the dangers, traps, and risks of using simulation tools** (as black boxes -> turning black boxes into white boxes)
- Providing you with basic knowledge about common **numerical methods**
- Knowing **application domains** of the various methods and guidelines what method works best for various problems
- ... and having fun simulating waves ...

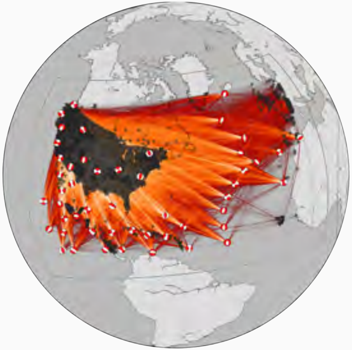
What is Computational Seismology?

We define **computational seismology** such that it **involves the complete solution of the seismic wave propagation (and rupture) problem for arbitrary 3-D models by numerical means.**

What is not covered ... but you can do tomography with ...

- Ray-theoretical methods
- Quasi-analytical methods (e.g., normal modes, reflectivity method)
- Frequency-domain solutions
- Boundary integral equation methods
- Discrete particle methods

These methods are important for benchmarking numerical solutions!



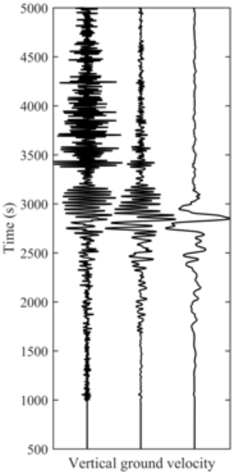
Why numerical methods?

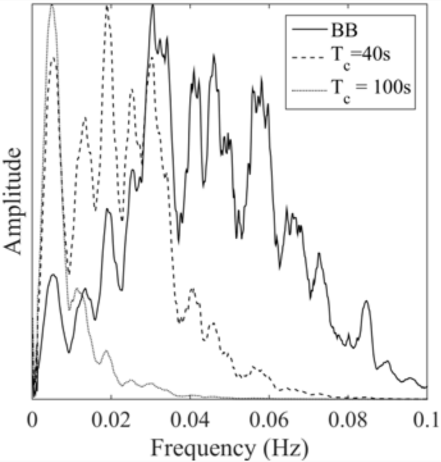
Computational Seismology, Memory, and Compute Power

Numerical solutions necessitate the discretization of Earth models. Estimate how much memory is required to store the Earth model and the required displacement fields.

Are we talking laptop or supercomputer?

Seismic Wavefield Observations





Exercise: Sampling a global seismic wavefield

- The highest frequencies that we observe for global wave fields is 1Hz.
- We assume a homogeneous Earth (radius 6371km).
- P velocity $v_p = 10 km/s$ and the v_p/v_s ratio is $\sqrt{3}$
- We want to use 20 **grid points (cells) per wavelength**
- How many grid cells would you need (assume cubic cells).
- What would be their size?
- How much memory would you need to store one such field (e.g., density in single precision).

You may want to make use of

$$c = \frac{\lambda}{T} = \lambda f = \frac{\omega}{k}$$

Exercise: Solution (Matlab)

% **Earth volume**
$v_e = 4/3 * pi * 6371^3$;
% **smallest velocity (ie, wavelength)**
vp=10; vs=vp/sqrt(3);
% **Shortest Period**
T=10;
% **Shortest Wavelength**
lam=vs*T;
% **Number of points per wavelength and**
% **required grid spacing**
$nplambda = 20$;
$dx = lam/nplambda$;
% **Required number of grid cells**
$nc = v_e/(dx^3)$;
% **Memory requirement (TBytes)**
$mem = nc * 8/1000/1000/1000/1000$;

Results ($@T = 1s$) : 360 TBytes
Results ($@T = 10s$) : 360 GBytes
Results ($@T = 100s$) : 360 MBytes

Computational Seismology, Memory, and Compute Power

1960: 1 MFlops

1970: 10MFlops

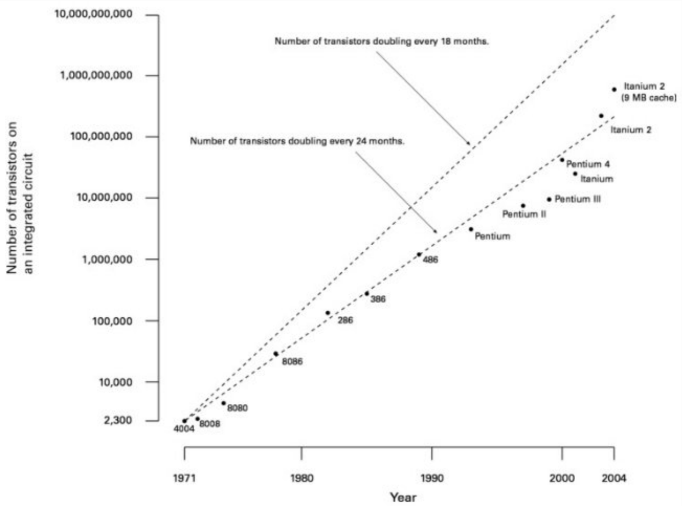
1980: 100MFlops

1990: 1 GFlops

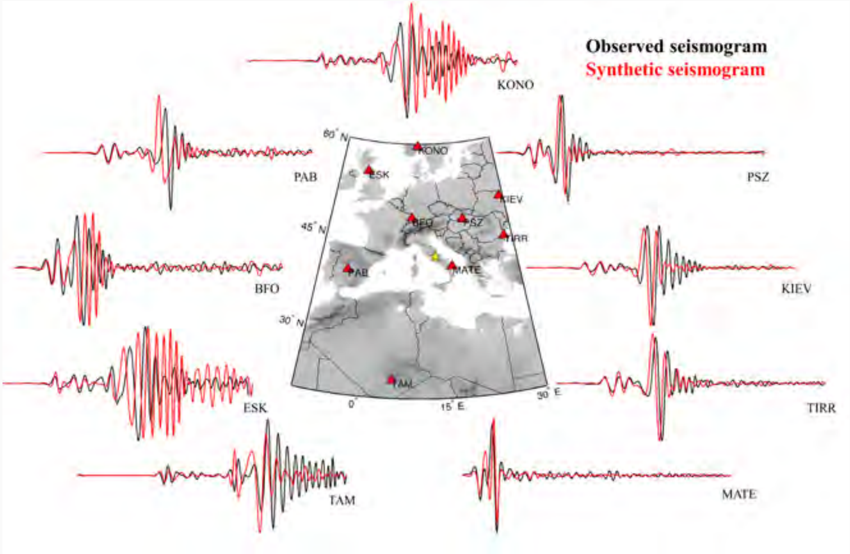
1998: 1 TFlops

2008: 1 Pflops

20??: 1 EFlops



The Utimate Goal: Matching Wavefield Observations



A Bit of Wave Physics

Acoustic wave equation: no source

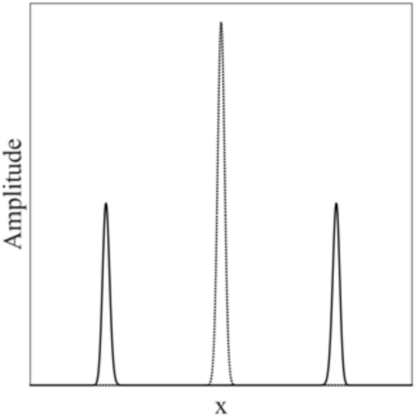
Acoustic wave equation

$$\partial_t^2 p \;=\; c^2 \Delta p + s$$

$p \rightarrow p(\mathbf{x}, t)$, pressure
$c \rightarrow c(\mathbf{x})$, velocity
$s \rightarrow s(\mathbf{x}, t)$, source term

Initial conditions

$$p(\mathbf{x}, t = 0) \;=\; p_0(\mathbf{x}, t)$$
$$\partial_t p(\mathbf{x}, t = 0) \;=\; 0$$



Snapshot of $p(\mathbf{x}, t)$ (solid line) after some time for initial condition $p_0(\mathbf{x}, t)$ (Gaussian, dashed line), 1D case.

Acoustic wave equation: external source

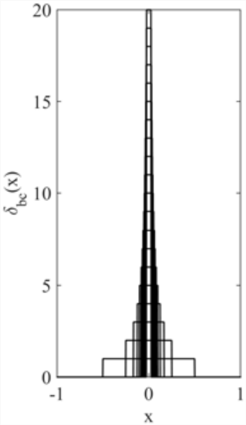
Green's Function G

$$\partial_t^2 G(\mathbf{x}, t; \mathbf{x}_0, t_0) - c^2 \Delta G(\mathbf{x}, t; \mathbf{x}_0, t_0) = \delta(\mathbf{x} - \mathbf{x}_0)\delta(t - t_0)$$

Delta function δ

$$\delta(x) = \begin{cases} \infty & x = 0 \\ 0 & x \neq 0 \end{cases}$$

$$\int_{-\infty}^{\infty} \delta(x) dx = 1 \,, \int_{-\infty}^{\infty} f(x)\delta(x) dx = f(0)$$



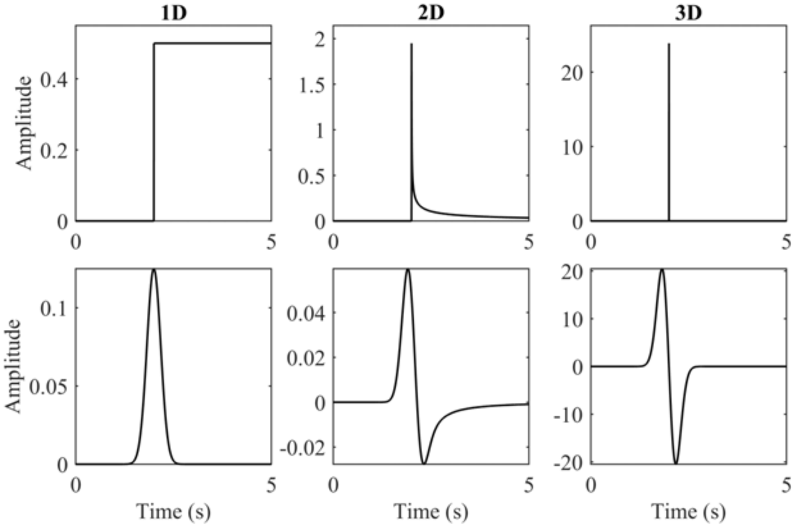
δ-generating function using boxcars.

Acoustic wave equation: analytical solutions

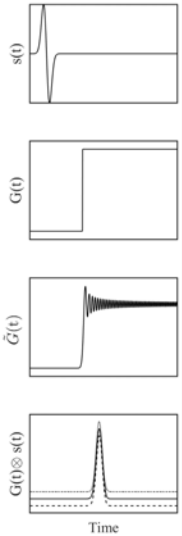
Green's functions for the inhomogeneous acoustic wave equation for all dimensions. $H(t)$ is the Heaviside function.

1D	2D	3D
$\frac{1}{2c} H(t - \frac{\lvert r \rvert}{c})$	$\frac{1}{2\pi c^2} \frac{H(t - \frac{\lvert r \rvert}{c})}{\sqrt{t^2 - \frac{r^2}{c^2}}}$	$\frac{1}{4\pi c^2 r} \delta(t - r/c)$
$r = x$	$r = \sqrt{x^2 + y^2}$	$r = \sqrt{x^2 + y^2 + z^2}$

Acoustic wave equation: analytical solutions



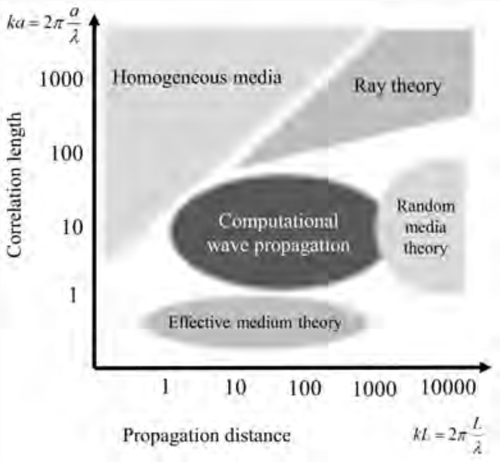
Wave Equation as Linear System



- Accurate Green's functions cannot be calculated numerically
- A numerical solver is a **linear system**
- The convolution theorem applies
- Inaccurate simulations can be filtered afterwards
- Source time functions can be altered afterwards
- ... provided the sampling is good enough ...

Numerical Methods for Wave Propagation Problems

Spatial Scales, Scattering, Solution Strategies



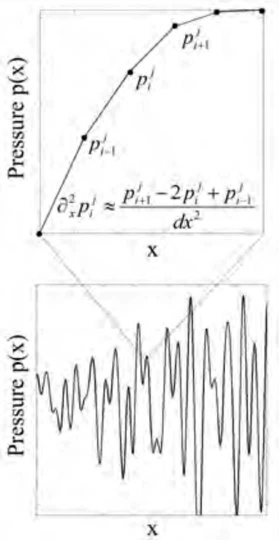
- Recorded seismograms are affected by ...
- ... the ratio of seismic wavelength λ and structural wavelength *a* ...
- ... how many wavelengths are propagated ...
- strong scattering when $a \approx \lambda \rightarrow$ numerical methods
- ray theory works when $a >> \lambda$
- random medium theory necessary for strong scattering media (and long distances)

What’s on the market

- **The finite-difference method**
- The pseudospectal method
- The finite-element method
- **The spectral-element method**
- The finite-volume method
- **The discontinuous Galerkin method**

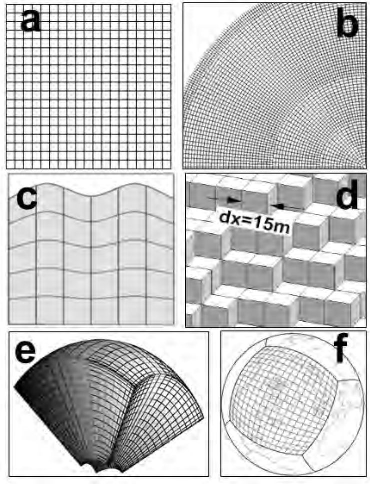
The Finite-Difference Method

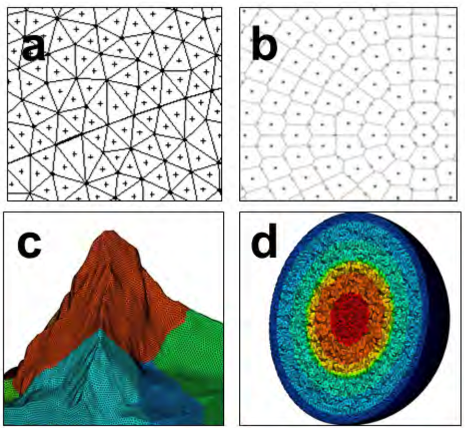
Finite differences in a Nutshell



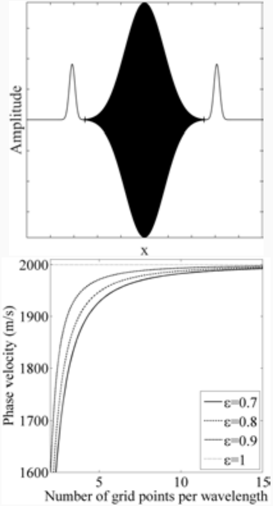
- Direct numerical approximation of partial derivatives using finite-differences
- Local computational scheme $\rightarrow$ efficient parallelisation
- Very efficient on regular grids, cumbersome for strongly heterogeneous models
- Boundary conditions difficult to implement with high-order accuracy
- The method of choice for models with flat topo and moderate velocity perturbations
- Highly efficient extensions possible, but rarely used!

Meshes, grids, structured, unstructured



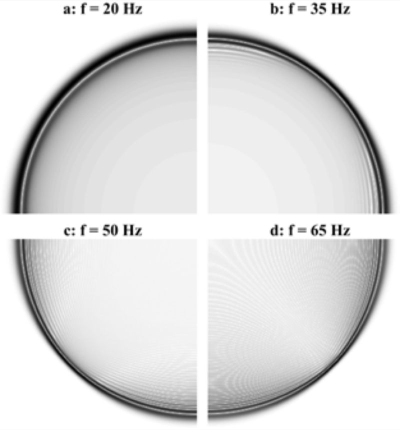


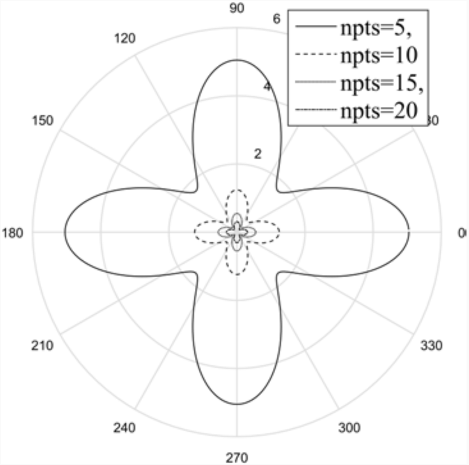
von Neumann Analysis, Stability, Dispersion



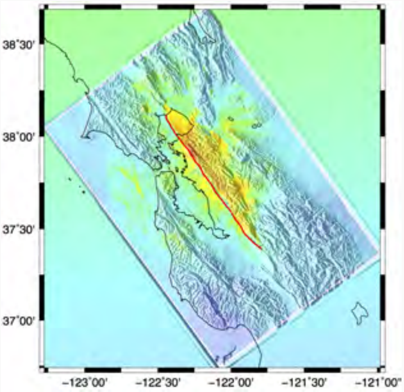
- Plane waves in a discrete world
- $p(x,t) = e^{kj\,\mathbf{dx}x - wn\,\mathbf{dt}}$
- Simulations are conditionally stable
- $c\,\frac{dt}{dx} \leq \epsilon \approx 1$ *CFL - criterion*
- Simulated phase velocity becomes numerically dispersive!
- The more points per wavelength the more accurate
- How to check?

Numerical Anisotropy





Applications, recent , community codes

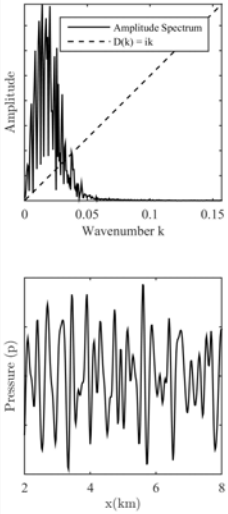


Source: geodynamics.org (SW4)

- Method of choice for flat surfaces and body wave problems (exploration)
- Problems for strong topography
- Very accurate (optimal) operators possible, but ...
- Summation-by-parts approach (better for topography)
- Combination with homogenisation (regular grid revival)
- Community codes: SW4 (CIG), SOFI3D (Karlsruhe)

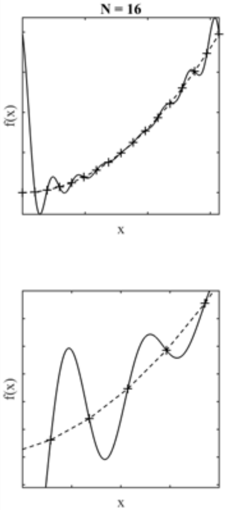
The Pseudospectral Method: the road to spectral elements

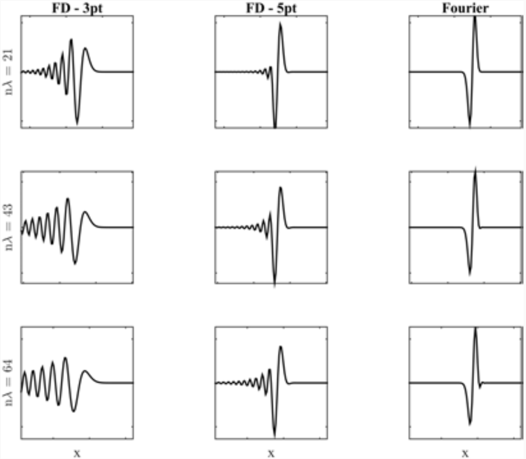
The Pseudospectral Method in a Nutshell



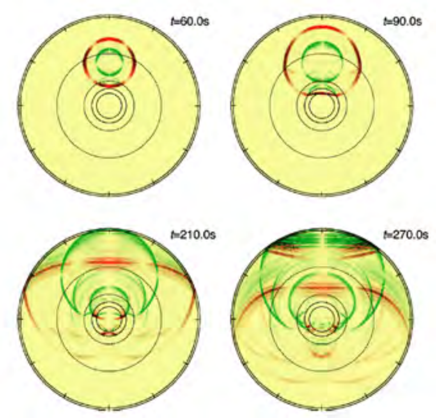
- Calculation of *exact* derivatives in spectral domain
- Less dispersive than the finite-difference method (isotropic errors)
- Boundary conditions hard to implement
- Global communication scheme → inefficient parallelisation
- Combinations with FD possible
- Hardly in use today, but concepts used in the spectral-element method

Exact interpolation/derivative: Fourier Series





Applications, recent developments

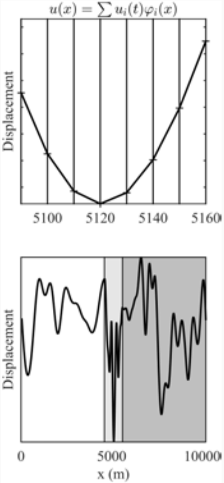


Seismic wave simulation in the Moon (Wang et al., GJI, 2013)

- Axisymmetric wave propagation (Group Prof. Furumura)
- Implementation in spherical coordinates
- Pseudospectral approach in θ direction
- Finite-difference approach in radial direction
- Used in combination with axisem ($\rightarrow$ axisem3d)

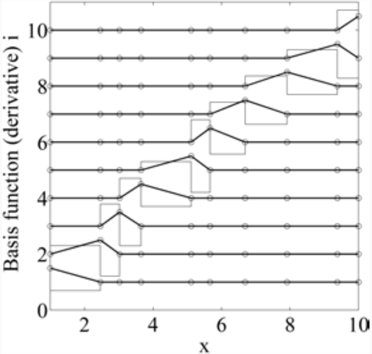
The Finite-Element Method

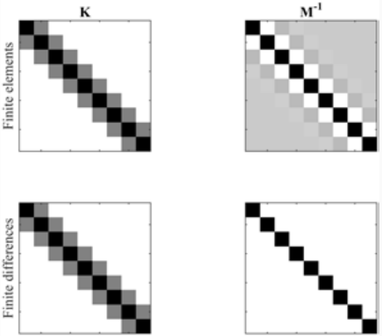
The Finite-Element Method in a Nutshell



- Solution of the *weak form* of the wave equation
- Wavefield is interpolated with (linear) orthogonal basis functions
- Global linear system of equations has to be solved (matrix inversion)
- **Free surface condition implicitly fullfilled**
- Works on hexahedral or tetrahedral meshes

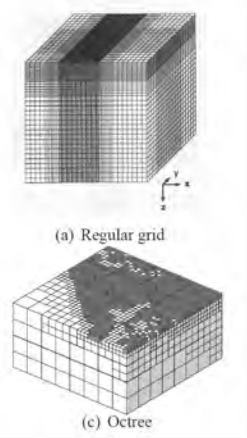
Linear basis functions, system matrices





Linear finite-element method and low-order finite-difference method are basically identical

Applications, Recent Developments



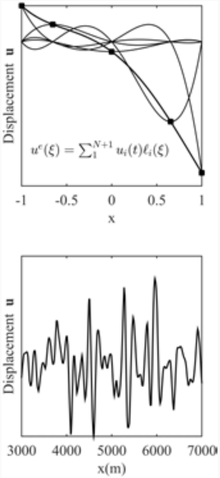
Finite element mesh, octree approach

(Bielak et al.)

- Requires linear algebra libraries for matrix inversion
- Suboptimal for parallelization
- Allows arbitrary geometric complexity
- Curbed element boundaries possible
- Standard in engineering applications
- Hardly used in seismology (why?)

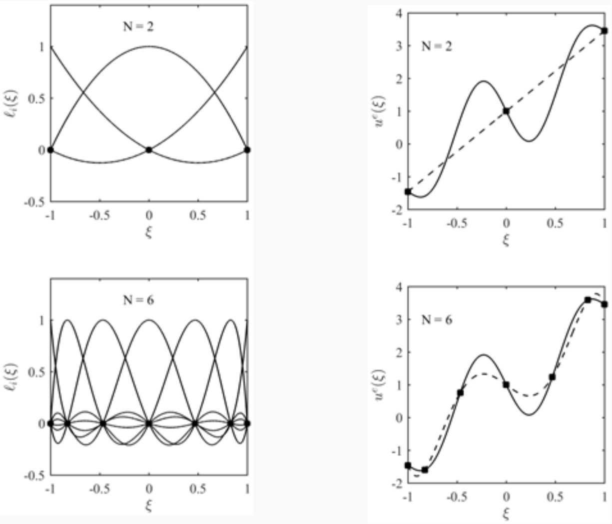
The Spectral-Element Method

The Spectral-Element Method in a Nutshell



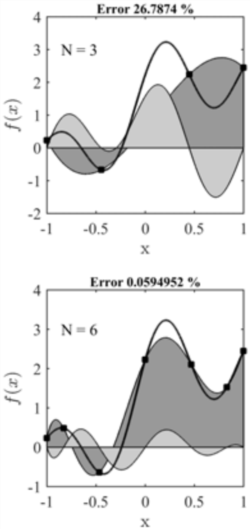
- Same mathematical derivation as the finite-element method
- Lagrange polynomial representation of wave field
- Gauss-Lobatto-Legendre collocation points (stability!)
- **Diagonal mass matrix** $\rightarrow$ trivially inverted
- Explicit extrapolation scheme $\rightarrow$ efficient parallelisation
- Method of choice for global wave propagation (specfem, axisem)
- Meshing required

Lagrange polynomials, interpolation



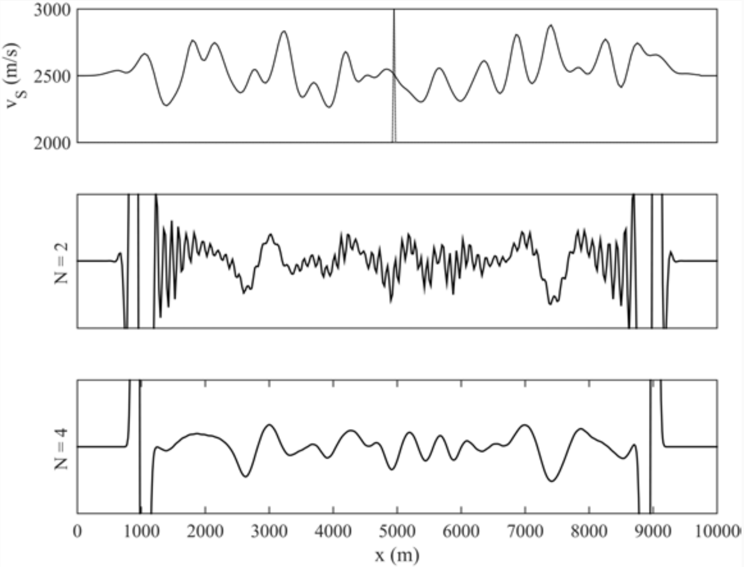
- Uneven grid spacing for high-order polynomials → time-stepping
- Maximum order usually N=4
- Chebyshev was first (why?)

Numerical integration

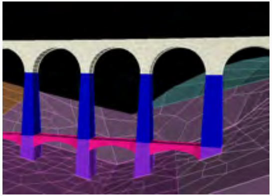


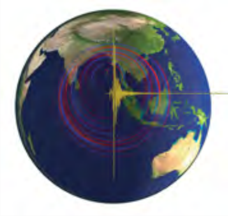
- Galerkin methods necessitate integral calculations over elements
- They are the only source of errors in the space discretization (SE)
- Integration with Chebyshev polynomials would be exact up to order N
- Integration with same collocation points as interpolation

Example, convergence



Applications, Recent Developments

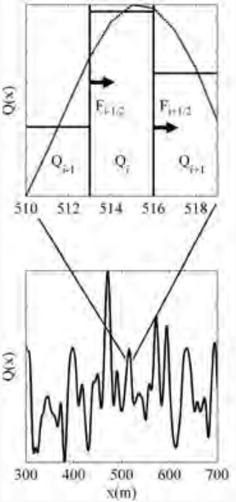




- Many applications in regional and global seismology
- Method of choice whenever surface waves are involved
- Spherical geometry with cubed sphere approach
- Applications to soil-structure interaction
- Works for hexahedral and tetrahedral meshes ($\rightarrow$ salvus)
- specfem maybe most widely used community software for global seismology (3D)

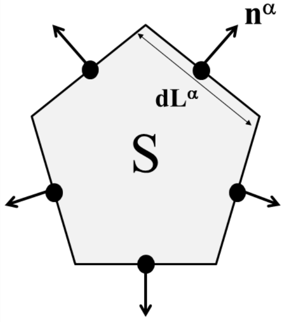
The Finite-Volume Method

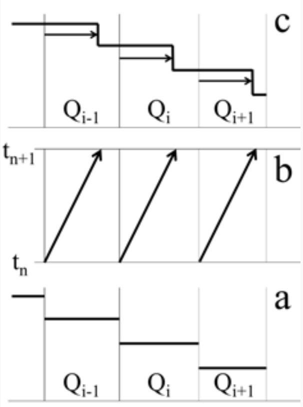
The Finite-Volume Method in a Nutshell



- Mathematically derived as a conservation law
- Spatial discretization with arbitrary volumes
- Extreme geometric flexibility (e.g., shock waves)
- Voronoi cells, tetrahedra, polygons
- Entirely local formulation (cell based)
- Communication with neighbours through flux scheme
- Hardly used in seismology

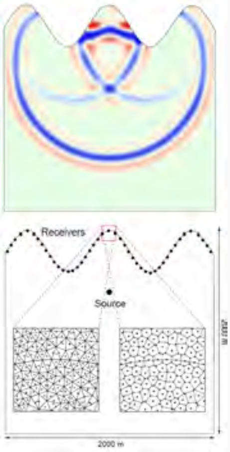
Fluxes, Riemann problem





The finite-volume approach allows derivation of acoustic wave equation from first principles (i.e., mass conservation)

Applications, Recent Developments

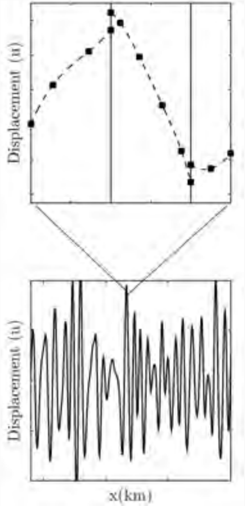


Kaeser et al., 2000

- Method of choice for conservation problems with strongly discontinuous solutions
- Many applications in geophysical fluid dynamics
- Relatively simple, finite-difference style algorithms
- Linear extrapolation schemes strongly diffusive
- Recent general extensions to higher order
- Potential for seismology not fully explored

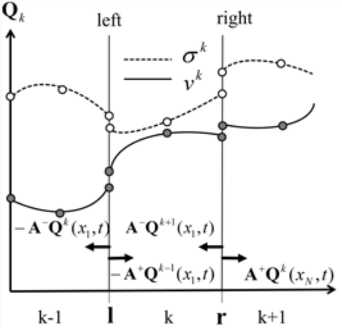
The Discontinuous Galerkin Method

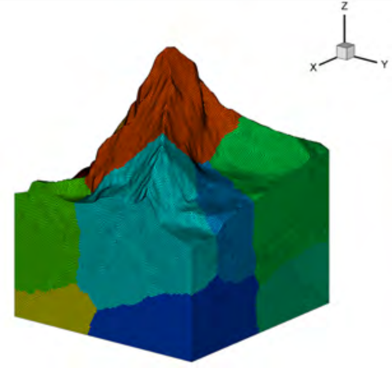
The Discontinuous Galerkin Method in a Nutshell



- Numerical solution of first-order systems
- Developed for hyperbolic problems (e.g., neutron diffusion)
- Local formulation for each element
- Solution of *weak form* of wave equation
- Communication between elements through fluxes $\rightarrow$ FV
- Explicit time extrapolation $\rightarrow$ efficient parallelisation
- Nodal and modal approaches for hexahedral and tetrahedral meshes

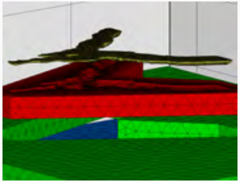
Fluxes, tetrahedral meshes

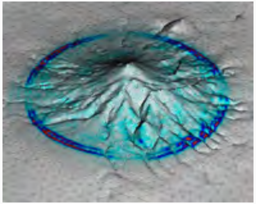




The first competitive scheme for tetrahedral meshes in seismology, but ...

Applications, Recent Developments

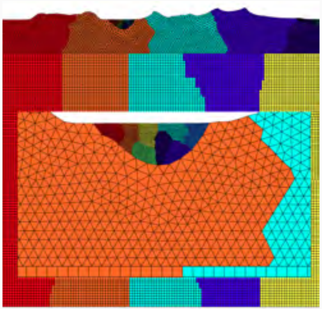




- Applications in exploration, volcanology, shaking hazard, earthquake physics
- Inefficient with tetrahedral meshes (for smooth problems)
- Method of choice for dynamic rupture simulations
- Extremely well scalable (Gordon Bell finalist 2015)
- New modal, octree approach developed in ExaHype project
- Community codes: *seissol* (munich), *nex3d* (Bochum)

"Meet the future ..."

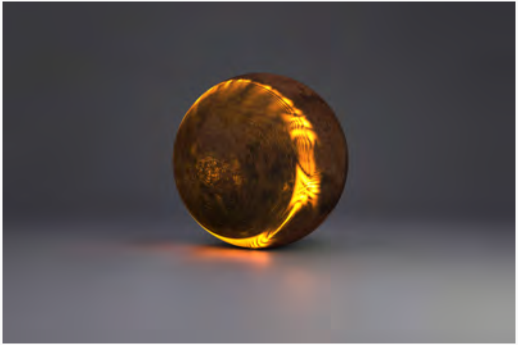
Challenges - Meshing



Human time	Simulation workflow	cpu time
15%	Design	0%
80% (weeks)	Geometry creation, meshing	10%
5%	Solver	90%

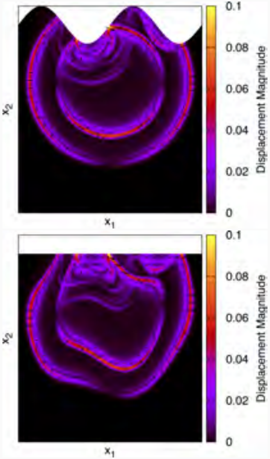
- Meshing work flow not well defined
- Still major bottleneck for simulation tasks with complex geometries
- Tetrahedral meshes easier, but ...
- Salvus?

Spectral element method - *Salvus*



- In development at ETH
- Spectral-element implementation
- tetrahedral and hexahedral meshes
- built on top of community libraries (e.g., PetSc)
- Meshing routines for some model classes

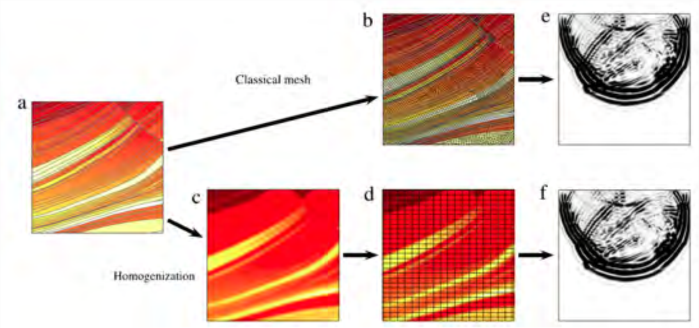
Future Strategies - Alternative Formulations



Al-Attar and Crawford GJI 2016

- Particle relabelling
- Summation-by-parts
- Mapping geometrical complexity onto regular grids
- Smart pre-processing rather than meshing?
- Similar concept used in summation-by-parts (SBP) algorithms (SW4)

Future Strategies - Homogenization



- We only *see* low-pass filtered Earth
- So why simulate models with infinite frequencies?
- Homogenisation of discontinuous model
- Renaissance of regular grid methods?

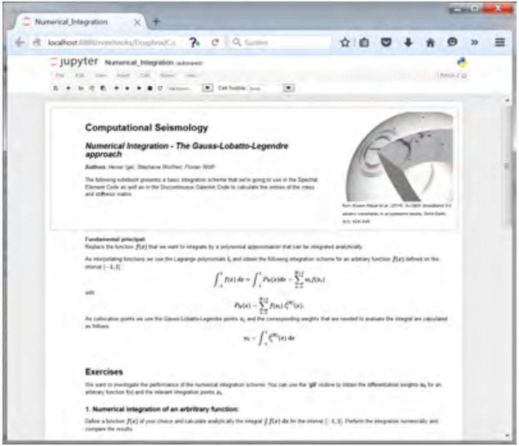
Challenges - Community Platforms

www.verce.eu

- Science gateways for basic simulation tasks
- High level model initialization
- Large scale simulations - hidden supercomputers
- Complex admission protocols
- Black boxes
- Great idea, but ...

Computational Seismology, Practical Exercises, Jupyter Notebooks

- *Jupyter notebooks* are interactive documents that work in any browser
- Simple text editing
- Inclusion of graphics
- Equations with Latex
- Executable code cells with Python (or else)
- The coolest thing since ...
- Many examples on: *www.seismo-live.org*
- *Computational Seismology: A Practical Introdcution* (Oxford University Press)



Try it out!

Conclusions

The forward problem is solved, but ...