Acoustic wave equation

Helmholtz (wave) equation (time-dependent)

- Regular grid
- Irregular grid

Numerical Examples

The Acoustic Wave Equation 1-D

How do we solve a time-dependent problem such as the acoustic wave equation?

$$\partial_t^2 u - v^2 \Delta u = f$$

where *v* is the wave speed.
using the same ideas as before we multiply this equation with an arbitrary function and integrate over the whole domain, e.g. [0,1], and after partial integration

$$\int_0^1 \partial_t^2 u \varphi_j dx - v^2 \int_0^1 \nabla u \nabla \varphi_j dx = \int_0^1 f \varphi_j dx$$

.. we now introduce an approximation for u using our previous basis functions...

The Acoustic Wave Equation 1-D

$$u \approx \tilde{u} = \sum_{i=1}^{N} c_i(t)\varphi_i(x)$$

note that now our coefficients are time-dependent!
... and ...

$$\partial_t^2 u \approx \partial_t^2 \tilde{u} = \partial_t^2 \sum_{i=1}^{N} c_i(t)\varphi_i(x)$$

together we obtain

$$\left[\sum_i \partial_t^2 c_i \int_0^1 \varphi_i \varphi_j dx\right] + v^2 \left[\sum_i c_i \int_0^1 \nabla\varphi_i \nabla\varphi_j dx\right] = \int_0^1 f\varphi_j$$

which we can write as ...

Time extrapolation

$$\left[\sum_i \partial_t^2 c_i \int_0^1 \varphi_i \varphi_j dx\right] + v^2 \left[\sum_i c_i \int_0^1 \nabla\varphi_i \nabla\varphi_j dx\right] = \int_0^1 f\varphi_j$$

M — mass matrix

A — stiffness matrix

b

... in Matrix form ...

$$M^T \ddot{c} + v^2 A^T c = g$$

... remember the coefficients c correspond to the actual values of u at the grid points for the right choice of basis functions ...

How can we solve this time-dependent problem?

FD extrapolation

$$M^T \ddot{c} + v^2 A^T c = g$$

... let us use a finite-difference approximation for the time derivative ...

$$M^T \left(\frac{c_{k+1} - 2c + c_{k-1}}{dt^2} \right) + v^2 A^T c_k = g$$

... leading to the solution at time t_{k+1}:

$$c_{k+1} = \left[(M^T)^{-1} (g - v^2 A^T c_k) \right] dt^2 + 2c_k - c_{k-1}$$

we already know how to calculate the matrix A but how can we calculate matrix M?

The mass matrix

$$\left[\sum_i \partial_t^2 c_i \int_0^1 \varphi_i \varphi_j dx\right] + v^2 \left[\sum_i c_i \int_0^1 \nabla \varphi_i \nabla \varphi_j dx\right] = \int_0^1 f \varphi_j$$

... let's recall the definition of our basis functions ...

$$M_{ij} = \int_0^1 \varphi_i \varphi_j dx$$

$$\varphi_i(\tilde{x}) = \begin{cases} \dfrac{\tilde{x}}{h_{i-1}} + 1 & \mathit{for} \quad -h_{i-1} < \tilde{x} \le 0 \\ 1 - \dfrac{\tilde{x}}{h_i} & \mathit{for} \quad 0 < \tilde{x} < h_i \\ 0 & \mathit{elsewhere} \end{cases} \quad , \tilde{x} = x - x_i$$

i=1	2	3	4	5	6	7
+	+	+	+	+	+	+
h_1	h_2	h_3	h_4	h_5	h_6	

... let us calculate some element of M ...

The mass matrix - some elements

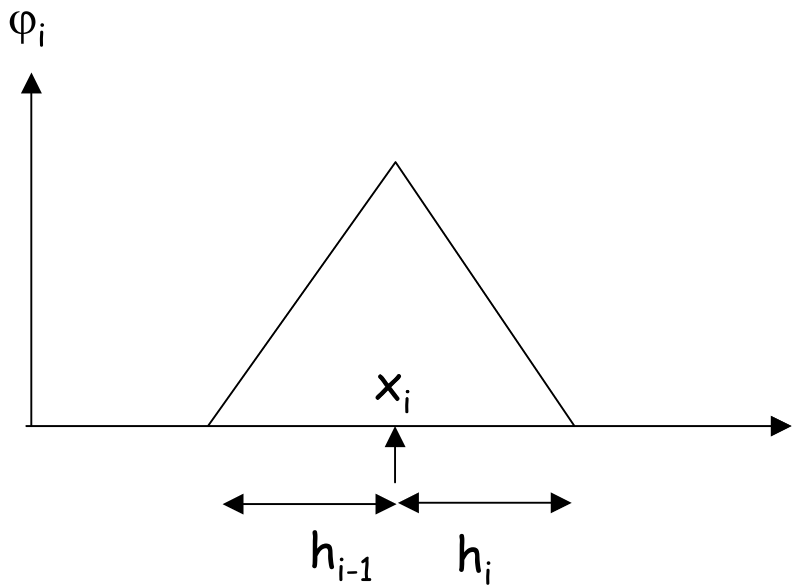
$$\varphi_i(\tilde{x}) = \begin{cases} \frac{\tilde{x}}{h_{i-1}} + 1 & \text{for} \quad -h_{i-1} < \tilde{x} \le 0 \\ 1 - \frac{\tilde{x}}{h_i} & \text{for} \quad 0 < \tilde{x} < h_i \\ 0 & \text{elsewhere} \end{cases}$$

Diagonal elements: M_{ii}, i=2,n-1

$$M_{ii} = \int_0^1 \varphi_i \varphi_i dx = \int_0^{h_{i-1}} \left(\frac{x}{h_{i-1}} \right)^2 dx + \int_0^{h_i} \left(1 - \frac{x}{h_i} \right)^2 dx$$

$$= \frac{h_{i-1}}{3} + \frac{h_i}{3}$$

i=1	2	3	4	5	6	7
+	+	+	+	+	+	+
h_1	h_2	h_3	h_4	h_5	h_6	



Matrix assembly

M_{ij}

```
% assemble matrix Mij

M=zeros(nx);

for i=2:nx-1,
   for j=2:nx-1,
      if i==j,
         M(i,j)=h(i-1)/3+h(i)/3;
      elseif j==i+1
         M(i,j)=h(i)/6;
      elseif j==i-1
         M(i,j)=h(i)/6;
      else
         M(i,j)=0;
      end
   end
end
```

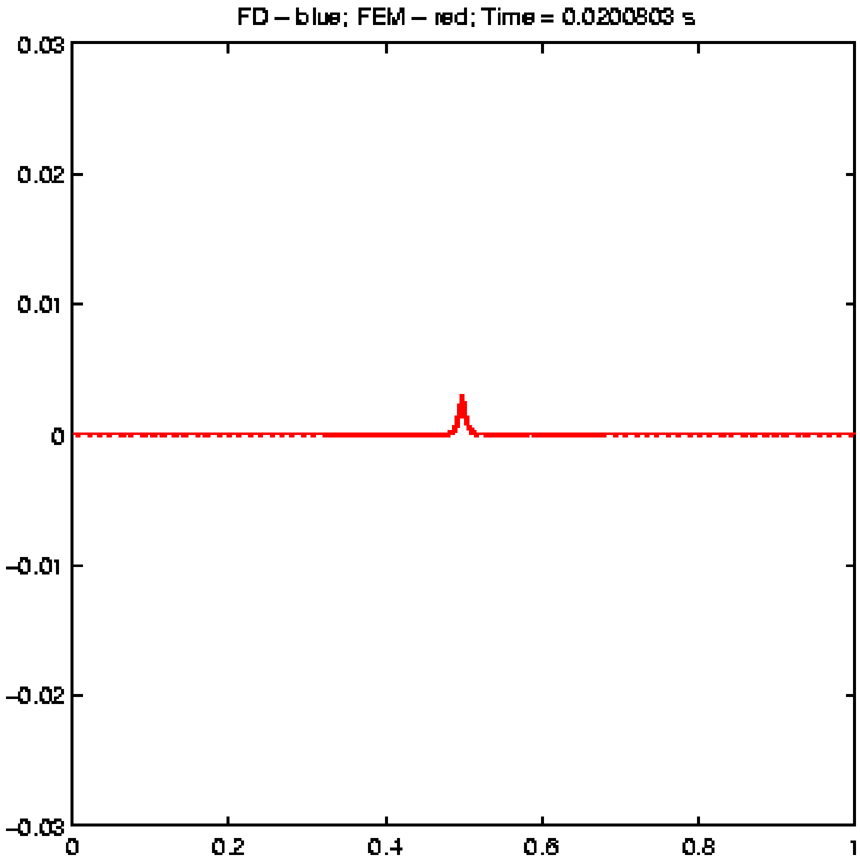
A_{ij}

```
% assemble matrix Aij

A=zeros(nx);

for i=2:nx-1,
   for j=2:nx-1,
      if i==j,
         A(i,j)=1/h(i-1)+1/h(i);
      elseif i==j+1
         A(i,j)=-1/h(i-1);
      elseif i+1==j
         A(i,j)=-1/h(i);
      else
         A(i,j)=0;
      end
   end
end
```

Numerical example - regular grid



Implicit Time extrapolation

Let is recall the *ODE*:

$$\frac{dT}{dt} = f(T,t)$$

Before we used a forward difference scheme, what happens if we use a backward difference scheme?

$$\frac{T_j - T_{j-1}}{dt} + O(dt) = f(T_j, t_j)$$

$$\Rightarrow \boxed{T_j \approx T_{j\text{-}1} + \mathrm{dt} f(T_j, t_j)}$$

Implicit schemes - stability

or

$$T_j \approx T_{j-1}(1+\frac{dt}{\tau})^{-1}$$

$$T_j \approx T_0(1+\frac{dt}{\tau})^{-j}$$

Is this scheme *convergent*?

Does it tend to the exact solution as dt->0? YES, it does (exercise)

Is this scheme *stable,* i.e. does T decay monotonically? This requires

$$0 < \frac{1}{1+\frac{dt}{\tau}} < 1$$

What is an implicit method?

Let is recall the *ODE*:

$$\frac{dT}{dt} = f(T,t)$$

Before we used a forward difference scheme, what happens if we use a backward difference scheme?

$$\frac{T_j - T_{j-1}}{dt} + O(dt) = f(T_j, t_j)$$

$$\Rightarrow \boxed{T_j \approx T_{j\text{-}1} + \mathrm{dt} f(T_j, t_j)}$$

FD extrapolation - implicit

$$M^T \ddot{c} + v^2 A^T c = g$$

... let us use an implicit finite-difference approximation for the time derivative ...

$$M^T \left(\frac{c_{k+1} - 2c + c_{k-1}}{dt^2} \right) + v^2 A^T c_{k+1} = g$$

... leading to the solution at time t_{k+1}:

$$c_{k+1} = \left[M^T + v^2 dt^2 A^T \right]^{-1} \left(g dt^2 + M^T \left(2c - c_{k-1} \right) \right)$$

How do the numerical solutions compare?