FINITE DIFFERENCE SIMULATION

Maarten van Walstijn

Aims

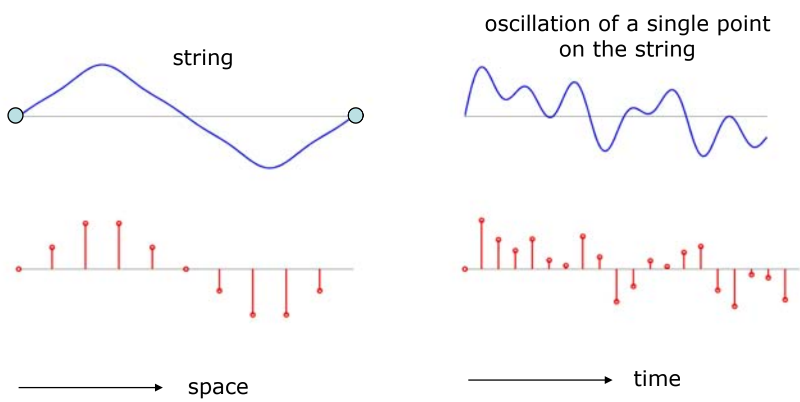
- To gain understanding of finite difference schemes for discretisation of continuous space-time systems (such as strings & membranes).
- The become able in applying basic discretisation methods and implement them in Matlab.

Content

- Discretisation using Finite Differences
- FD String Simulation
- Stability Analysis
- Adding Damping and Excitation Force terms
- FD Membrane Simulation

Discretistation using Finite Differences

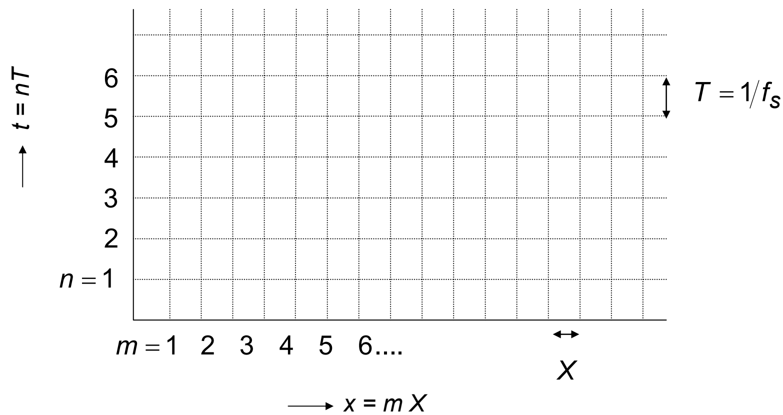
Discrete-time modelling of spatially continuous vibrating systems involves discretisation of both space and time.



Discrete-Time Notation

$$y(x,t) \rightarrow y(mX, nT) \rightarrow y_m^n$$

space-time continuem → space-time grid



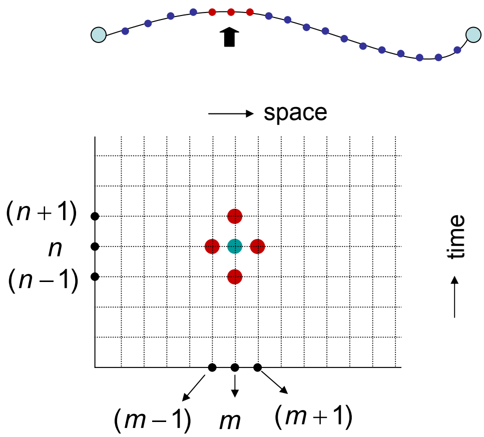
FD String Simulation: Centered Scheme

(lossless) wave equation: $$\frac{\partial^2 y}{\partial t^2} = c^2 \frac{\partial^2 y}{\partial x^2}$$

2nd derivative with respect to *x:* $$\left[\frac{\partial^2 y}{\partial x^2}\right]_m^n \approx \frac{y_{m+1}^n - 2y_m^n + y_{m-1}^n}{X^2}$$

2nd derivative with respect to *t:* $$\left[\frac{\partial^2 y}{\partial t^2}\right]_m^n \approx \frac{y_m^{n+1} - 2y_m^n + y_m^{n-1}}{T^2}$$

FD String Simulation: Centered Scheme (cont.)



FD schemes use "neighbouring points" in the space-time grid

FD String Simulation: Centered Scheme (cont.)

$$\frac{\partial^2 y}{\partial t^2} = c^2 \frac{\partial^2 y}{\partial x^2}$$

Substitution of the derivative approximations into the wave equation yields:

$$\frac{y_m^{n+1} - 2y_m^n + y_m^{n-1}}{T^2} = c^2 \frac{y_{m+1}^n - 2y_m^n + y_{m-1}^n}{X^2}$$

$$\Leftrightarrow$$

$$y_m^{n+1} - 2y_m^n + y_m^{n-1} = \lambda^2 \left(y_{m+1}^n - 2y_m^n + y_{m-1}^n \right)$$

$$\left(\lambda = \frac{cT}{X} \right)$$

FD String Simulation: Centered Scheme (cont.)

By writing the newest value explicit, the final difference equation takes the form:

$$y_m^{n+1} = a_{10} y_m^n + a_{11}\left(y_{m+1}^n + y_{m-1}^n\right) + a_{20} y_m^{n-1}$$

last value at same grid-point

last values at neighbouring grid-points

before-last value at same grid-point

$$\begin{cases} a_{10} = 2 - 2\lambda^2 \\ a_{11} = \lambda^2 \\ a_{20} = -1 \end{cases}$$

For simulation we have to apply some boundary conditions, for example, those for the fixed, fixed string:

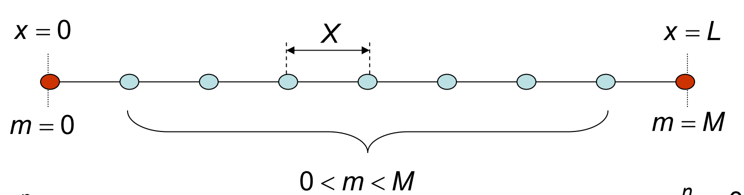
$$\begin{cases} y(0,t) = 0 \\ y(L,t) = 0 \end{cases} \Rightarrow \begin{cases} y_0^n = 0 \\ y_{M+1}^n = 0 \end{cases}$$

$MX = L$ = string length

M = number of string segments

Boundary Conditions: Fixed at Both ends

M string segments → (M+1) grid-points

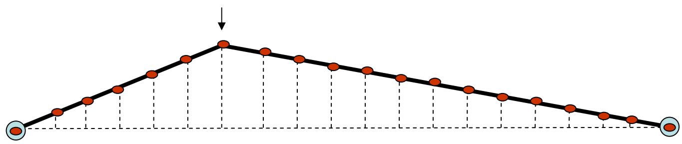


$y_0^n = 0$ $y_M^n = 0$

$$y_m^{n+1} = a_{10} y_m^n + a_{11}\left(y_{m+1}^n + y_{m-1}^n\right) + a_{20} y_m^{n-1}$$

Excitation by Initial Condition

We can set the string into vibration in two different ways. One way is to pluck the string. That is, the string is pulled out of it equilibrium as follows:



In the FD simulation, we simply have set the 'initial state' of the string to that of the spatially sampled pluck shape.

Plucked String Simulation: Implementation

```
%%% index vector for computed grid-points %%%
ind = 2:Np-1;

%%% initiation of variables %%%
Y0 = zeros(Np,1);   % displacement vector at time (n+1)
Y1 = zeros(Np,1);   % displacement vector at time (n)
Y2 = zeros(Np,1);   % displacement vector at time (n-1)

%%% initial pluck %%%
Y0 = pluck; Y1 = pluck; Y2 = pluck;    % pluck = initial state vector

%%% the main loop %%%
for n=1:N
   %%% compute difference equation %%%
   Y0(ind) = a10*Y1(ind) + a11*(Y1(ind + 1) + Y1(ind - 1)) + a20*Y2(ind) + aF*F0(ind);

   %%% memorise the vectors %%%
   Y2 = Y1;
   Y1 = Y0;
end
```

see also example Matlab code FDstring.m

Stability Analysis

Assume solutions of the kind (i.e. wave travelling in positive direction):

$$y(x,t) = e^{st+jkx}$$

but it also works for waves travelling in the negative direction. In discrete space-time, we have:

$$y_m^n = e^{snT+jkmX}$$

We can now formulate how successive values depend on each other in terms of ‘amplification’:

$$\begin{aligned} y_m^{n+1} &= e^{s(n+1)T+jkmX} \\ &= e^{sT}\left[e^{snT+jkmX}\right] \\ &= z\, y_m^n \end{aligned}$$

$$\left(z = e^{sT}\right)$$

amplification factor

Stability Analysis (cont.)

Now let’s see what the other terms in the difference equation look like:

$$\begin{aligned} y_{m+1}^n &= e^{snT+jk(m+1)X} \\ &= e^{kX} e^{snT+jkmX} = e^{jkX} y_m^n \end{aligned}$$

$$\begin{aligned} y_{m-1}^n &= e^{snT+jk(m-1)X} \\ &= e^{jkX} e^{snT+jkmX} = e^{-jkX} y_m^n \end{aligned}$$

Hence we can write:

$$\begin{aligned} y_{m+1}^n + y_{m-1}^n &= \left(ej^{kX} + e^{-jkX}\right) y_m^n \\ &= 2\cos(kX)\, y_m^n \end{aligned}$$

Stability Analysis (cont.)

Insert it all into the difference equation:

$$\underbrace{y_m^{n+1}}_{zy} = a_{10} y_m^n + a_{11} \underbrace{\left(y_{m+1}^n + y_{m-1}^n\right)}_{2\cos(kX) y_m^n} + a_{20} \underbrace{y_m^{n-1}}_{z^{-1} y_m^n}$$

$$\Leftrightarrow$$

$$y_m^n z = a_{10} y_m^n + 2 a_{11} \cos(kx) y_m^n + a_{20} y_m^n z^{-1}$$

$$\Leftrightarrow$$

$$z = a_{10} + 2 a_{11} \cos(kx) + a_{20} z^{-1}$$

$$\Leftrightarrow$$

$$z^2 + 2 \underbrace{\left[-\tfrac{1}{2} a_{10} - a_{11} \cos(kx)\right]}_{B} - \underbrace{a_{20}}_{-1} z^{-1}$$

Stability Analysis (cont.)

Amplification equation:

$$z^2 + 2Bz + 1 = 0$$

where

$$\begin{aligned} B &= -\tfrac{1}{2} a_{10} - a_{11} \cos(kx) \\ &= \lambda^2 - \lambda^2 \cos(kx) - 1 \\ &= \lambda^2 \underbrace{\left(1 - \cos(kx)\right)}_{F_k} - 1 \\ &= \lambda^2 F_k - 1 \end{aligned}$$

$$\begin{cases} a_{10} = 2 - 2\lambda^2 \\ a_{11} = \lambda^2 \end{cases}$$

$$F_k = 1 - \cos(kx)$$

Stability Analysis (cont.)

The solutions of the amplification equation are:

$$z = -B \pm \sqrt{B^2 - 1}$$

For stable, oscillatory solution, we have

Condition 1:

$$|z| \le 1$$

Condition 2:

$$B^2 \le 1$$

Stability Analysis (cont.)

Condition 1:

Assuming condition 2 is satisfied:

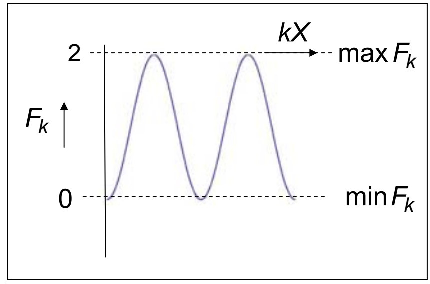
$$\begin{aligned} |z|^2 &= \left(-B - j\sqrt{1-B^2}\right)\left(-B + j\sqrt{1-B^2}\right) \\ &= \left(B^2 - B^2 + 1\right) \\ &= 1 \end{aligned}$$

condition 1 is also satisfied!

Stability Analysis (cont.)

Condition 2:

$$B^2 \le 1$$
$$\Leftrightarrow \left(\lambda^2 F_k - 1\right)^2 \le 1$$
$$\Leftrightarrow \lambda^4 F_k^2 - 2\lambda^2 F_k \le 0$$
$$\Leftrightarrow \lambda^2 F_k \left(\lambda^2 F_k - 2\right) \le 0$$
$$\Leftrightarrow \lambda^2 \le \frac{2}{F_k}$$
$$\Leftrightarrow \lambda \le \sqrt{\frac{2}{\max F_k}}$$
$$\lambda \le \sqrt{\frac{2}{2}}$$
$$\lambda \le 1$$



$$\boxed{\lambda = \frac{cT}{X} \le 1}$$

Excitation by External Force

Another way of exciting the string is by simply exerting a force at one or more positions on the string. In order to realise that, we first have to add a force-term to the wave equation:

$$\frac{\partial^2 y}{\partial t^2} = c^2 \frac{\partial^2 y}{\partial x^2} + F(x,t)$$

We can now re-do our discretisation process. The only change to the difference equation is an extra force term there:

$$y_m^{n+1} = a_{10} y_m^n + a_{11}\left(y_{m+1}^n + y_{m-1}^n\right) + a_{20} y_m^{n-1} + a_F F$$

where $a_F = T^2$

can change when losses
or taken into account

Adding Damping Terms

Damping terms are possible in the wave equation:

$$\frac{\partial^2 y}{\partial t^2} = c^2 \frac{\partial^2 y}{\partial x^2} \underbrace{-2b_1 \frac{\partial y}{\partial t}}_{\text{air damping}} \underbrace{+2b_2 \frac{\partial^3 y}{\partial t \partial x^2}}_{\text{internal friction}}$$

In order to discretise the wave equation with one or both of these damping terms, we can use the centered-difference approximation for the air damping term:

$$\left[\frac{\partial y}{\partial t}\right]_m^n \approx \frac{y_m^{n+1} - y_m^{n-1}}{2T}$$

Adding Damping Terms (cont.)

For the internal friction term, using the centered-difference equation leads to an implicit scheme, with the disadvantages of being more computationally expensive. An explicit scheme can be obtained by using the backward-difference approximation for the differentiation with respect to time:, i.e.

$$\left[\frac{\partial^3 y}{\partial x^2 \partial t}\right]_m^n = \left[\frac{\partial\left(\frac{\partial^2 y}{\partial x^2}\right)}{\partial t}\right]_m^n \approx \frac{\left(\frac{\partial^2 y}{\partial x^2}\right)_m^n - \left(\frac{\partial^2 y}{\partial x^2}\right)_m^{n-1}}{T}$$

where

$$\left[\frac{\partial^2 y}{\partial x^2}\right]_m^n \approx \frac{y_{m+1}^n - 2y_m^n + y_{m-1}^n}{X^2} \qquad \left[\frac{\partial^2 y}{\partial x^2}\right]_m^{n-1} \approx \frac{y_{m+1}^{n-1} - 2y_m^{n-1} + y_{m-1}^{n-1}}{X^2}$$

Some Sound Examples

plucked & hammered

varying position

varying damping (b1 & b2)

varying tension

on-line parameter variation

Some Further Issues Regarding FD String Simulation

- Discretisation artefacts
- Stability Analysis when including damping terms
- Adding stiffness (piano strings)
- Other finite-difference schemes possible (including implicit schemes)

FD Membrane Simulation

2D wave equation:

$$\frac{\partial^2 u}{\partial t^2} = c^2\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right)$$

For standard centered scheme, substitute for FD approximations of derivatives:

$$\left[\frac{\partial^2 u}{\partial t^2}\right]_{m,i}^{n} \approx \frac{u_{m,i}^{n+1} - 2u_{m,i}^{n} + u_{m,i}^{n-1}}{T^2}$$

$$\left[\frac{\partial^2 u}{\partial x^2}\right]_{m,i}^{n} \approx \frac{u_{m+1,i}^{n} - 2u_{m,i}^{n} + u_{m-1,i}^{n}}{X^2}$$

$$\left[\frac{\partial^2 u}{\partial y^2}\right]_{m,i}^{n} \approx \frac{u_{m,i+1}^{n} - 2u_{m,i}^{n} + u_{m,i-1}^{n}}{X^2}$$

FD Membrane Simulation (cont.)

Set $\left(\lambda = \frac{cT}{X}\right)$

$$u_{m,i}^{n+1} - 2u_{m,i}^{n} + u_{m,i}^{n-1} = \lambda^2\left(u_{m+1,i}^{n} - 2u_{m,i}^{n} + u_{m-1,i}^{n} + u_{m,i+1}^{n} - 2u_{m,i}^{n} + u_{m,i-1}^{n}\right)$$

Final difference equation :

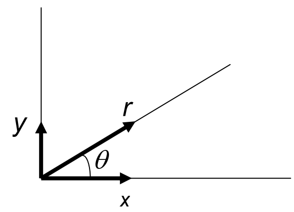
$$u_{m,i}^{n+1} = a_{10}u_{m,i}^{n} + a_{20}u_{m,i}^{n-1} + a_{11}\left(u_{m+1,i}^{n} + u_{m-1,i}^{n} + u_{m,i+1}^{n} + u_{m,i-1}^{n}\right)$$

where $\begin{cases} a_{10} = 2 - 4\lambda^2 \\ a_{20} = -1 \\ a_{11} = \lambda^2 \end{cases}$

Stability Analysis

Assume wave of the form:

$$y(x,t) = e^{st} e^{-jkr}$$



$$r = x\cos(\theta) + y\sin(\theta)$$

Write in terms of x and y coordinates:

$$\begin{aligned} y(x,t) &= e^{st} e^{-jk(\cos(\theta)x + \sin(\theta)y)} \\ &= e^{st} e^{-j(k_x x + k_y y)} \end{aligned}$$

where

$$\begin{aligned} k_x &= k\cos(\theta) \\ k_y &= k\sin(\theta) \end{aligned}$$

Stability Analysis (cont.)

In discrete space-time, we have:

$$\begin{aligned} y_{m,i}^{n} &= e^{snT} e^{-j(k_x mX + k_y iX)} \\ y_{m,i}^{n+1} &= e^{s(n+1)T} e^{-j(k_x mX + k_y iX)} = z y_{m,i}^{n} \\ y_{m,i}^{n-1} &= e^{s(n-1)T} e^{-j(k_x mX + k_y iX)} = z^{-1} y_{m,i}^{n} \\ y_{m+1,i}^{n} &= e^{-snT} e^{-j(k_x (m+1)X + k_y iX)} = e^{-jk_x X} y_m^n \\ y_{m-1,i}^{n} &= e^{-snT} e^{-j(k_x (m-1)X + k_y iX)} = e^{+jk_x X} y_m^n \\ y_{m,i+1}^{n} &= e^{-snT} e^{-j(k_x mX + k_y (i+1)X)} = e^{-jk_y X} y_m^n \\ y_{m,i-1}^{n} &= e^{-snT} e^{-j(k_x mX + k_y (i-1)X)} = e^{+jk_y X} y_m^n \end{aligned}$$

$$\left(z = e^{sT}\right)$$

Stability Analysis (cont.)

Substitution into the difference equation yields amplification equation:

$$z^2 + 2Bz + 1 = 0$$

$$\boxed{\begin{aligned} B &= -\tfrac{1}{2}\left[a_{10} + 2a_{11}\left(\cos(k_x X) + \cos(k_y X)\right)\right] \\ &= \lambda^2 F_k - 1 \\ F_k &= 2 - \cos(k_x X) - \cos(k_y X) \end{aligned}}$$

For stability, we have the condition:

$$B^2 \le 1$$

Which becomes:

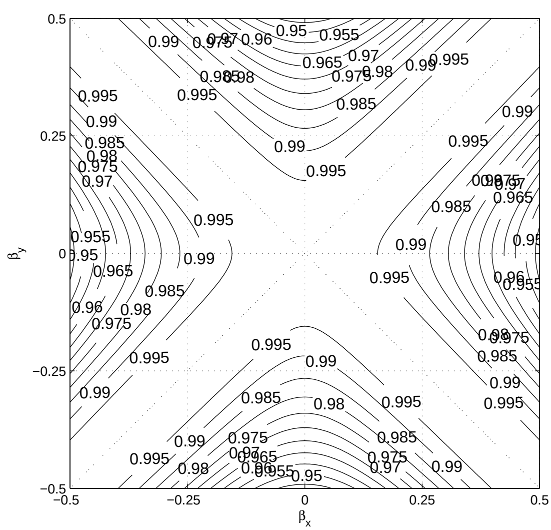
$$\lambda \le \sqrt{\frac{2}{\max F_k}} \qquad \lambda \le \sqrt{\frac{2}{4}} \qquad \boxed{\lambda = \frac{cT}{X} \le \tfrac{1}{2}\sqrt{2}}$$

Dispersion Error

Shown is the relative phase velocity, i.e.

$$\frac{v_{discrete}}{v_{continuous}}$$

The plot shows that the disperion is strongly direction-dependent.



Some Further Issues Regarding FD Membrane Simulation

- Dispersion Error can be reduced using Implicit Schemes
- Stability Analysis when including damping terms
- Adding stiffness -> becomes plate
- 2D wave equation can also be applied to room modelling
- Many different boundary conditions possible.
- Unlike 1D systems, ‘anechoic boundary’ difficult to realise.