UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

COMPUTER BASED ENGINEERING MATHEMATICS

SUMMER SEMESTER 2018

Project : Vibration of a string

Cordinators:

Prof. Dr. Johannes Gottschling
Dipl.-Math. Robert Martin
Institute for Applied Material technology
Universität Duisburg-Essen

Group members:

Md Abu Saym (3022015)
Sadek Dewan (3056001)
Mohammad khademul Amin(3022194)
Azahar Hossain(3056371)
Abdullah Al Mamun(xxxxx)

Table of Contents

UNIVERSITÄT DUISBURG ESSEN

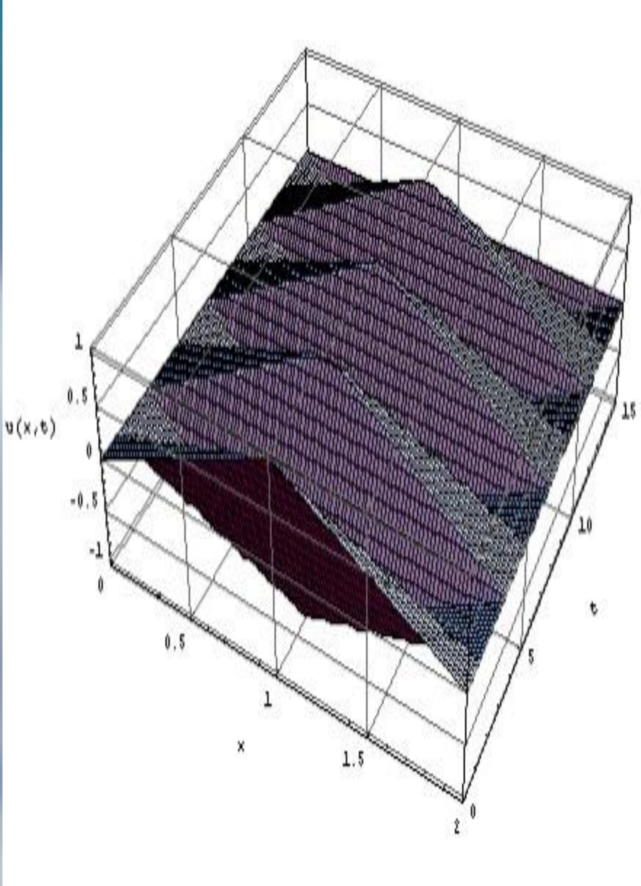
Offen im Denken

Project 1 – Vibration of a string

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

$$\begin{cases} \dfrac{\partial^2 u}{\partial t^2}(x,t) - c^2\,\dfrac{\partial^2 u}{\partial x^2}(x,t) = 0\,, \\[2ex] u(0,t) = u(L,t) = 0 \quad \forall\, t \geq 0\,, \\[2ex] u(x,0) = f(x),\; u_t(x,0) = g(x) \quad \forall\, x \in [0;L] \end{cases}$$



Task 1:

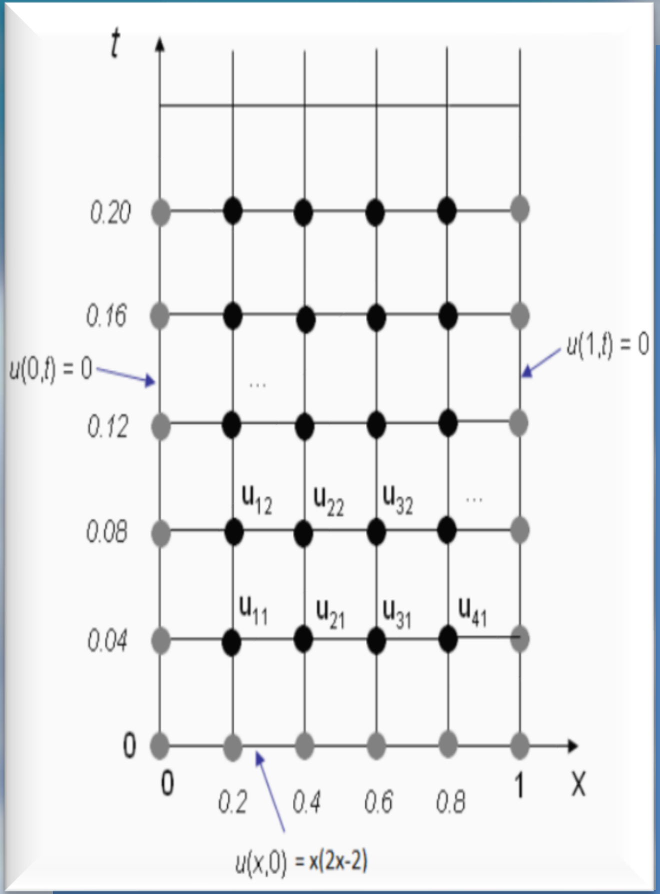
UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

Using the methods introduced in Section 1.1 of the script (“Motivation”), find (an approximation of) the displacement function *u* of a vibrating string for

- $c = 1$
- $L = 1$
- the initial velocity zero,
- the initial displacement *f* with
$f(x) = x \cdot (x - 2) \cdot (x - 1)$.

Plot the (three-dimensional) graph of $u(\cdot,\cdot)$ and the (two-dimensional) graphs of $u(\cdot,0)$, $u(\cdot,\pi)$ and $u(\cdot,2\pi)$.



Task 2:

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

Write a MATLAB function which takes the input arguments

- f (the initial displacement function),
- h (the grid size in x-direction),
- k (the grid size in t-direction),
- T (the endpoint of the time interval)

and returns the (approximate) solution to problem (1) for $c = 1$, $L = 1$, zero initial velocity and the given initial displacement f on the grid given by h, k and T

Task 2 (Cont'd)

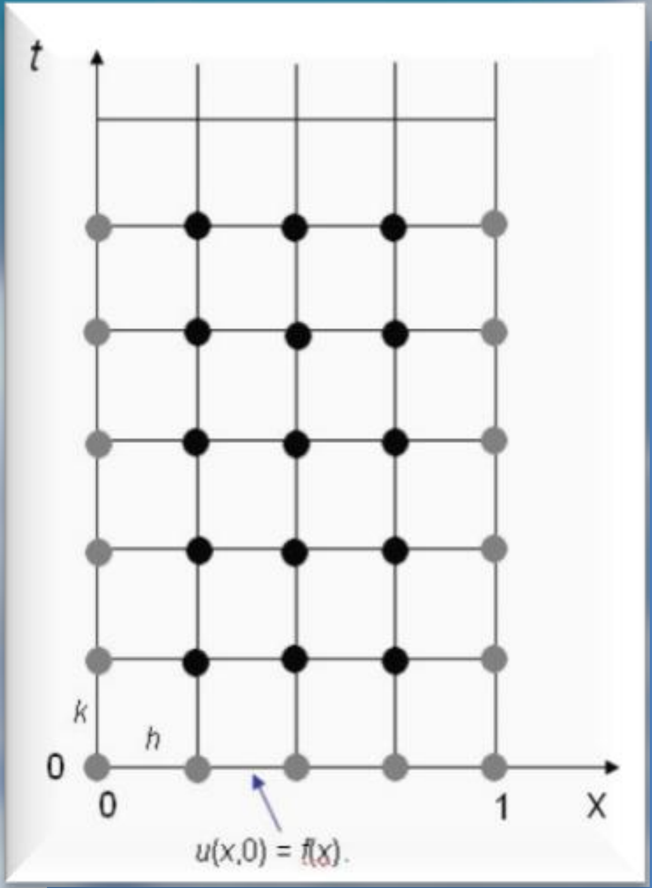
UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

Your project solution must be a file called

waveSolution.m of the following form:

```
function [ u ] = waveSolution( f, h, k, T )
    % your code% ...
endaa
```



THE WAVE EQUATION

Mathematical Model

The wave equation in 1D:

- Physical phenomenon: small vibrations on a string
- Mathematical model: the wave equation

$$\begin{cases} \dfrac{\partial^2 u}{\partial t^2}(x,t) - c^2 \dfrac{\partial^2 u}{\partial x^2}(x,t) = 0\,, \\[2ex] u(0,t) = u(L,t) = 0 \quad \forall\, t \geq 0\,, \\[2ex] u(x,0) = f(x),\ u_t(x,0) = g(x) \quad \forall\, x \in [0;L] \end{cases}$$

Mathematical Model (Cont'd)

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

Initial conditions on $U(x,0)$ and $U_t(x,0)$

For small vibrations($\partial u/\partial x \approx 0$) this simplifies to:

$$\frac{\partial^2 u}{\partial t^2} = c^2 \frac{\partial^2 u}{\partial x^2} \qquad \ldots\ldots c^2 = \frac{T}{\rho}$$

Initial and boundary conditions:

•String fixed at the ends:

$$u(a,t)=u(b,t)=0$$

•String initially at rest:

$$u(x,0)= l(x), \qquad u_t(x,0)=0$$

Mathematical Model (Cont'd)

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

After a scaling, the equation becomes:

$$\frac{\partial^2 u}{\partial t^2} = c^2 \frac{\partial^2 u}{\partial x^2}, \qquad x \in (0,1), \qquad t > 0$$

$$u(x,0) = I(x), \qquad x \in (0,1)$$

$$u_t(x,0) = 0, \qquad x \in (0,1)$$

$$u(0,t) = 0, \qquad t > 0$$

$$u(1,t) = 0, \qquad t > 0$$

Mathematical Model (Cont'd)

This program solves the 1D wave equation of the form:

Utt = c2 Uxx

over the spatial interval [X1,X2] and time interval [T1,T2], with initial conditions:

U(T1,X) = U_T1(X),
Ut(T1,X) = UT_T1(X),

Mathematical Model (Cont'd)

and boundary conditions of Dirichletian type:

U(T,X1) = U_X1(T)

U(T,X2) = U_X2(T)

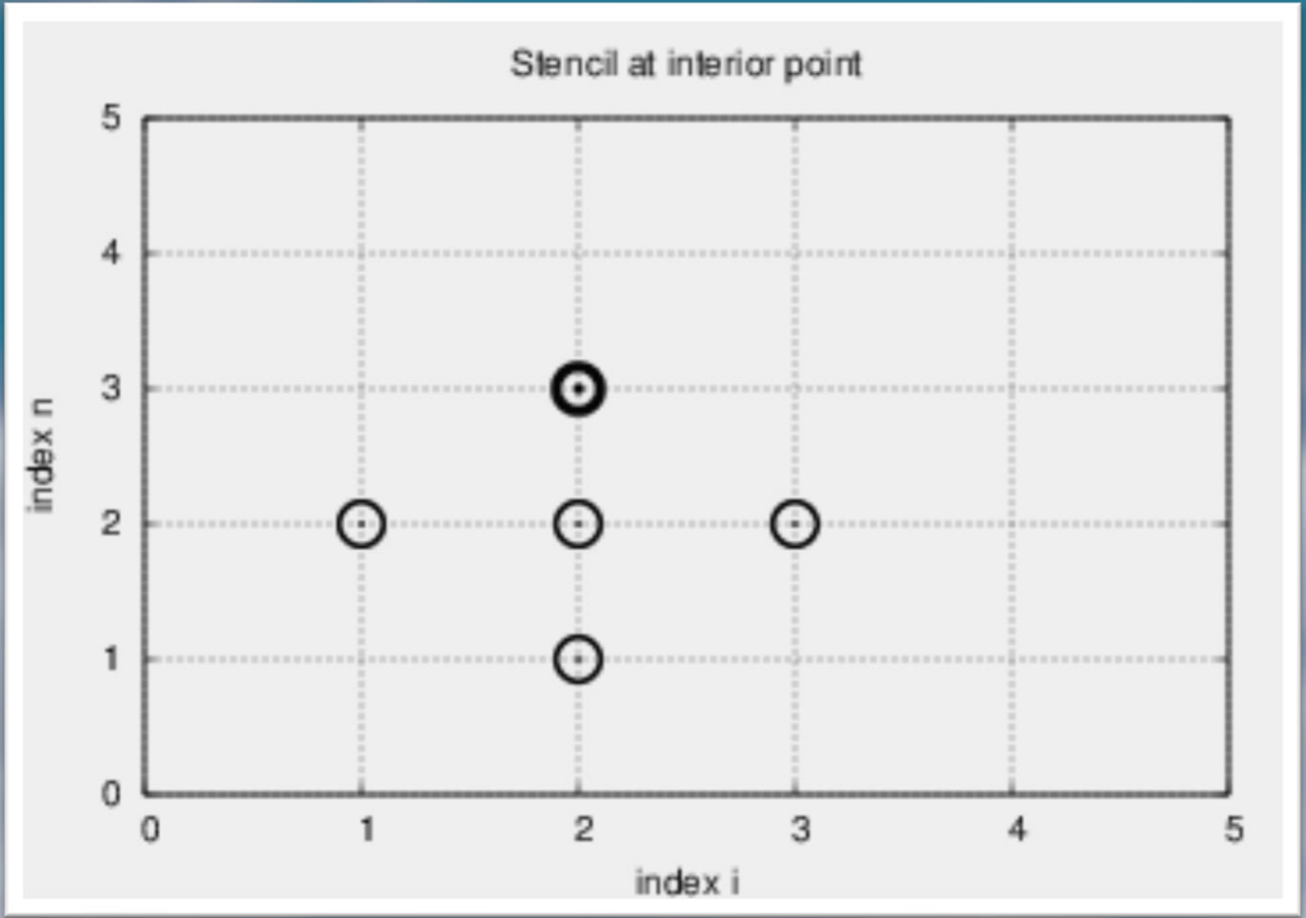
Uxx(T,X)=(U(T,X+dX)-2U(T,X)+U(T,X-dX))/dX2

Utt(T,X)=(U(T+dt,X)-2U(T,X)+U(T-dt,X))/dT2

Mathematical Model (Cont'd)

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken



Program (Cont'd)

U(T+dt,X)={(c2 * dT2/dX2) * U(T,X+dx)}+
2*(1–c2*dT2/dX2)* U(T, X)+
{(c2 * dT2/dX2)* U(T,X-dX)}-{U(T-dt,X)}

(Equation to advance from time T to time T+dT, except for FIRST step!)

Program (Cont'd)

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

dU/dT(T,X)=(U(T+dT,X)-U(T-dT,X))/(2*dT)

so we can estimate U(T-dT,X) as

U(T-dT,X)=U(T+dT,X)-2*dT*dU/dT(T,X)

U(T+dT,X)=1/2*(c2*dT2/dX2)*U(T,X+dX)
+(1– c2*dT2/dX2)*U(T,X)
+1/2*(c2*dT2/dX^2)*U(T,X-dX)
+dT * dU/dT(T,X)

Program (Cont'd)


```
clc
clear all
close all

u = 1; % wave-velocity
r = 1; % aspect-ratio
delx = 0.1; % x-step
delt = (delx*sqrt(r))/u; % t-step

x = 0:delx:1; % defining x-scale
t = 0:delt:1; % defining t-scale

%% Implementing Numerical Solution of wave
equation. PHI(x,t)=sin(pi*x)*cos(pi*t)
```

Program (Cont'd)


```
PHI = (cos(pi*t))'*sin(pi*x); % Analytical solution

%% Implementing Analytical Solution of wave equation PHItt = PHIxx by Finite
% Difference Explicit method

PHIN = zeros(length(t),1)*zeros(1,length(x)); % initialising solution matrix, row as 'x' and column as 't'

PHIN(:,1) = 0; % Applying boundry condition, PHIN(0,t) = 0
PHIN(:,length(x)) = 0; % Applying boundry condition, PHIN(1,t) = 0
```

Program (Cont'd)


```
PHIN(1,:) = sin(pi*x); % Applying initial condition,PHIN(x,0) = sin(pi*x)

% calculting second row, i.e. t = delt corresponding to j = 1, utilising
% initial condition PHINt(x,0) = 0 for 0<x<1

for n = 2:length(x)-1 % n is used as column index

 PHIN(2,n)= (PHIN(1,n-1)+PHIN(1,n+1))/2;

end
```

Program (Cont'd)

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

```
% Calculating third and higher order time row
for m = 3:length(t) % m is used as row index
   for n = 2:length(x)-1 % n is used as column index

 PHIN(m,n) = PHIN(m-1,n+1)+PHIN(m-1,n-1)-PHIN(m-2,n); % calculating time 't'-row

   end
end
```

Program (Cont'd)


```
mesh(x,t,PHI,'FaceLighting','gouraud','LineWidth',2) % Plotting 3-D analytical solution
hold on
mesh(x,t,PHIN) % Plotting 3-D numerical solution
legend('Analytical Solution','Numerical Solution')
set(gca,'FontSize',16)
xlabel('Distance(x) \rightarrow')
ylabel('\leftarrow Time(t)')
zlabel('Amplitude(Phi) \rightarrow')
title('Solution of wave equation \phi_t_t = \phi_x_x - Asssignment Number I')
```

Program (Cont'd)


```
colorbar

Error_A_N = abs(PHIN-PHI); % Calculationg Error between Numerical and Analytical Solution

figure,
mesh(x,t,Error_A_N,'FaceLighting','gouraud','LineWidth',2) % Plotting Error

set(gca,'FontSize',16)
xlabel('Distance(x) \rightarrow')
```

Program (Cont'd)


```
ylabel('\leftarrow Time(t)')
zlabel('Error \rightarrow')
title('Error between Numerical and Analytical Solution of wave equation
\phi_t_t = \phi_x_x')
colorbar
```