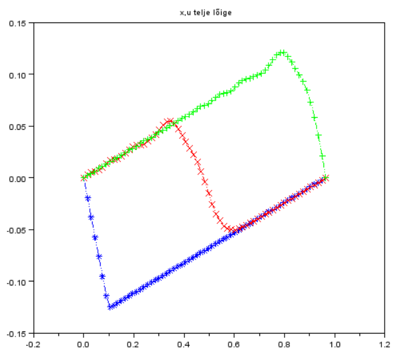
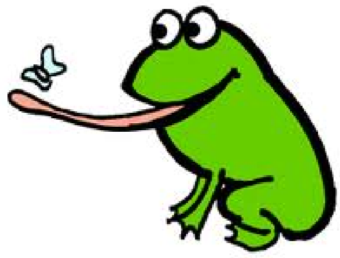
12 Keele väikesed ristvõnkumised

1746. aastal tuletas Jean Le Rond d'Alembert diferentsiaalvõrrandi, mille kohaselt lõpmata pika viiulikeele osakese liikumise kiirus on võrdeline naaberosakeste keskmise nihkega. Sellest - 1727. aastal John Bernoulli initsiatiivil algatatud veidi „mõtetust“ matemaatilisest probleemist - on aja jooksul välja arendatud analoogilised võrrandid mitmemõõtmelise juhu jaoks, mis praktikasse rakendatuna lubab modelleerida vee lainetust, helilaineid, elektromagnetlaineid, kvantlaineid jne. Toodud nimekirjast on lihtne ära tunda, miks nimetatakse keele võnkumise võrrandit ka lainevõrrandiks.





12.1 1-mõõtmeline laine levimise võrrand (keele võnkumise võrrand)

Vaatleme ühemõõtmelist laine levimise võrrandit (vt. [1]-[3])

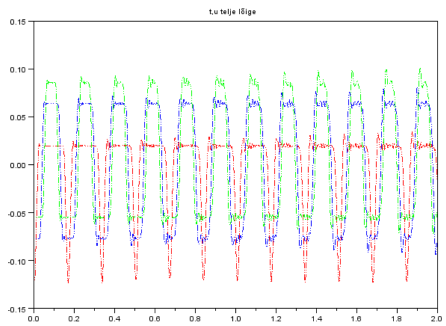
$$\boxed{\frac{\partial^2}{\partial t^2}u(x,t) = c^2 \cdot \frac{\partial^2}{\partial x^2}u(x,t), \quad x \in [0,b], \quad t \geq 0} \tag{12.1}$$

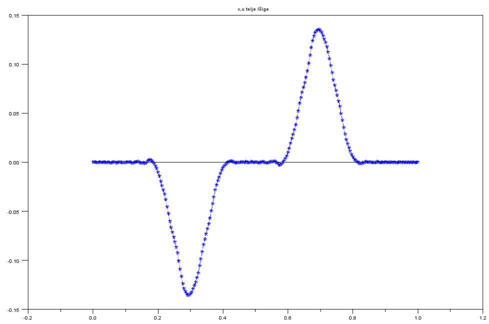
- $u(x,t)$ näitab laine (keele) asendit punktis x ajahetkel t.
- c on laine levimise kiirus (m/sek). Meenutame, et keele korral $c^2 = \frac{T_*}{\rho}$, kus T_* on keele pinge ja ρ keele tihedus, seega c sõltub keele materjali omadustest.
- Antud on algtingimused

$$u(x,0) = \Phi(x), \quad u_t(x,0) = g(x).$$

- Keel pikkusega b on jäigalt kinnitatud, seega meil on homogeensed rajatingimused

$$u(0,t) = 0, \quad u(b,t) = 0.$$





Viited

[1] D. C. Armstead, M. A. Karls. Does the Wave Equation Really Work? Primus, vol. XVI, no. 2, 2006.

[2] K. Celebi, I. Keles. Analysis of One-Dimensional Response of an Elastic Body Under Dynamic Loads. 6th International Advanced Technologies Symposium (IATS'11), 2011.

[3] The Science and Applications of Acoustics. II ed, Springer Science + Business Media, 2006.

12.2 1-mõõtmelise soojusjuhtivuse võrrandi lahendamine võrgumeetodiga

1. Valime lõigul $[0, b]$ osalõikude arvu N_x, sammu pikkuse $h_x = \frac{b}{N_x}$ ja moodustame sõlmed
$$x_i = b \cdot \frac{i}{N_x}, \quad i = 0, \ldots, N_x.$$
2. Valime ajalõigul $[0, T]$ osalõikude arvu N_t, sammu pikkuse $h_t = \frac{T}{N_t}$ ja moodustame sõlmed
$$t_j = T \cdot \frac{j}{N_t}, \quad j = 0, \ldots, N_t.$$
3. Otsime võrrandi
$$u_{tt}(x,t) = c^2 \cdot u_{xx}(x,t) + f(x,t), \quad x \in [0, b], \quad t \in [0, T],$$
lahendi $u(x,t)$ lähisväärtusi u_{ij} sõlmedes $x_0, \ldots, x_{N_x}$ ja $t_0, \ldots, t_{N_t}$. Selleks moodustame $(N_x+1) \times (N_t+1)$-mõõtmelise maatriksi
$$u = \left(\begin{array}{c|cccc} \mathbf{u_{00}} & \mathbf{u_{01}} & \mathbf{u_{02}} & \cdots & \mathbf{u_{0,N_t}} \\ \hline \mathbf{u_{10}} & u_{11} & u_{12} & \cdots & u_{1,N_t} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{u_{N_x-1,0}} & u_{N_x-1,1} & u_{N_x-1,2} & \cdots & u_{N_x-1,N_t} \\ \hline \mathbf{u_{N_x,0}} & \mathbf{u_{N_x,1}} & \mathbf{u_{N_x,2}} & \cdots & \mathbf{u_{N_x,N_t}} \end{array} \right).$$

Esimese veeru elemendid saame algtingimusest $u_{i,0} = \Phi(x_i)$, $i = 0, \ldots, N_x$. Esimese ja viimase rea elemendid saame rajatingimustest.

4. Märgime, et keele algkiiruse tingimusest saame

$$u_t(x_i, x_j) \approx \frac{u_{i,j+1} - u_{i,j-1}}{2 \cdot h_t} \quad \Rightarrow \quad u_{i,-1} \approx u_{i1} - 2 \cdot h_t \cdot g(x_i) = u_{i1}.$$

Viimase seose abil on meil võimalik tuletada u väärtuste leidmise skeem $j = 1$ korral:

$$\begin{pmatrix} u_{1,1} \\ u_{2,1} \\ \vdots \\ u_{N_x-1,1} \end{pmatrix} = \frac{1}{2} \cdot A \cdot \begin{pmatrix} u_{1,0} \\ u_{2,0} \\ \vdots \\ u_{N_x-1,0} \end{pmatrix} + \frac{1}{2} \cdot \sigma^2 \cdot \begin{pmatrix} u_{0,0} \\ 0 \\ \vdots \\ u_{N_x,0} \end{pmatrix} + h_t \cdot \begin{pmatrix} g_{1,0} \\ g_{2,0} \\ \vdots \\ g_{N_x-1,0} \end{pmatrix}.$$

5. Edasi lahendame ülesande võrgumeetodiga, mis näeb maatrikskujul välja järgmine:

$$U_{j+1} = A \cdot U_j - U_{j-1} + \sigma^2 \cdot \tilde{U}_j + h_t^2 \cdot F_j, \quad j = 1, \ldots, N_t - 1,$$

või siis täpsemalt

$$\begin{pmatrix} u_{1,j+1} \\ u_{2,j+1} \\ \vdots \\ u_{N,j+1} \end{pmatrix} = A \cdot \begin{pmatrix} u_{1,j} \\ u_{2,j} \\ \vdots \\ u_{N,j} \end{pmatrix} - \begin{pmatrix} u_{1,j-1} \\ u_{2,j-1} \\ \vdots \\ u_{N,j-1} \end{pmatrix} + \sigma^2 \cdot \begin{pmatrix} u_{0,j} \\ 0 \\ \vdots \\ u_{N_x,j} \end{pmatrix} + h_t^2 \cdot \begin{pmatrix} f(x_1, t_j) \\ f(x_2, t_j) \\ \vdots \\ f(x_N, t_j) \end{pmatrix},$$

kus

$$A = \begin{pmatrix} 2 - 2 \cdot \sigma^2 & \sigma^2 & 0 & \cdots & 0 \\ \sigma^2 & 2 - 2 \cdot \sigma^2 & \sigma^2 & \cdots & 0 \\ 0 & \sigma^2 & 2 - 2 \cdot \sigma^2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 2 - 2 \cdot \sigma^2 \end{pmatrix}$$

ja

$$\sigma = c \cdot \frac{h_t}{h_x}, \quad N = N_x - 1.$$

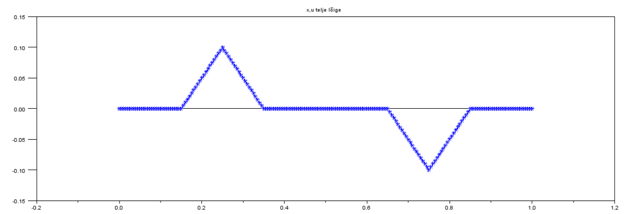
6. Toome siinkohal mõned ideed SciLab'i kasutamiseks.

```
u = zeros( Nx+1 , Nt+1 )
A = zeros( Nx-1 , Nx-1 ) // See tuleb veel õigesti täita
 // ...
  abiu = u( 2:Nx , j )        // maatriksi u j. veerg ilma rajata
 // ...
  u( 2:Nx , j+1 ) = abiu      // maatriksi u (j+1). veerg
```

12.3 Praktikumi ülesanne

1. Kasutame laine levimise kiirust $c = 10\ m/sek$. Keele pikkus b on 1 m. Olgu $t \in [0, 2]$.

2. Keel on oma algasendis välja venitatud.



   Viimast kirjeldab funktsioon

$$u(x,0) = \begin{cases} 0.1 - |x - 0.25| & , x \in [0.15, 0.35], \\ -0.1 + |x - 0.75| & , x \in [0.65, 0.85], \\ 0 & , \text{ muul juhul.} \end{cases}$$

3. Keele algkiirus on 0 m/sek, s.t. $u_t(x,t)\Big|_{t=0} = g(x) = 0.$

4. Millised peaksid olema osalõikude arvud N_x ja N_t, et lahend käituks stabiilselt?

5. Kui palju liigub keele keskpunkt oma algasendist kogu võnkumise vältel?

6. Kuidas liigub keele keskpunkt oma algasendist, kui punktis $x = 0.75$ on keel kaks korda pikemalt välja venitatud, kui punktis $x = 0.25$?

7. Video tegemiseks tuleks kasutada for-tsüklit (mingi hõrendatud sammuga, näiteks $i = 1 : 5 : N_t$) ja kuvada tühjendatud graafikule. Seejuures võib graafikud lasta piltidena salvestada, ja hiljem mingi video programmiga ühendada (näiteks Windows Live Movie Maker). Praktikumis piisab for-tsüklist ilma piltide salvestamiseta.

```
scf(1); clf(1)
plot( xx , yy( : , i ) , 'b:*' )
// xs2jpg(0,'pildid/graafik-'+string(i)+'.jpg')
```

8. 3D graafiku kohta võib kasutada järgmisi käske:

```
plot3d1( x , t , y , theta = -90 , alpha = 70 )
// theta ja alpha on nurgad kraadides graafiku pööramiseks
ff = gcf() \\
ff.color_map = rainbowcolormap( 512 )
```