Laboratorijske vaje Numerične metode

6. Vaja

B. Jurčič Zlobec[1], A. Perne[1]

[1]Univerza v Ljubljani,
Fakulteta za Elektrotehniko
1000 Ljubljana, Tržaška 25, Slovenija

Numerične metode FE, Ljubljana, 13. november 2012

Nelinearne enačbe

Pot svetlobnega žarka

Svetlobni žarek začne pot v točki $T_1 = (1/2, 2)$ in jo konča v točki $T_2 = (2, 1/2)$, glej sliko 1. Svetlobna hitrost je na svetlejšem delu območja enaka $c_1 = 1$, na temnejšem delu pa je enaka $c_2 = 0.3$. Mejo območja določa funkcija $f(x) = \sqrt{x}$.

Fermatov princip

Žarek izbere med dvema točkama takšno pot, da je čas potovanja najkrajši.

Najkrajši čas

Čas potovanja žarka

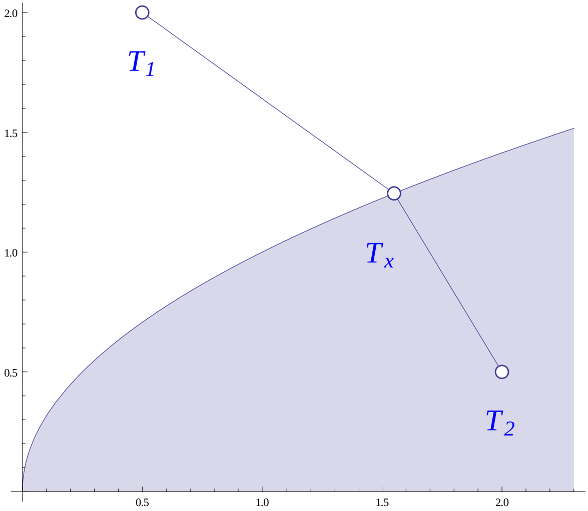
Točki $T_1 = (x_1, y_1)$, $T_2 = (x_2, y_2)$ in meja $y = f(x)$.
Če žarek prečka mejo v točki $T_x = (x, f(x))$,
je čas potovanja enak:

$$t(x) = \frac{\sqrt{(x_1 - x)^2 + (y_1 - f(x))^2}}{c_1} + \frac{\sqrt{(x - x_2)^2 + (f(x) - y_2)^2}}{c_2} \tag{1}$$

Najkrajši čas

$$t'(x) = 0$$

Slika 1



Slika: Pot žarka

Program

```
clear;
f=inline('sqrt(x)','x');
T=[0.5,2;2,0.5]; c=[1,0.3]; eps=1e-8;
t=inline('norm(T(1,:)-[x,f(x)])/c(1)\
+norm([x,f(x)]-T(2,:))/c(2)','x');
dt=inline('(t(x+eps)-t(x-eps))/2/eps','x');
x1=1; x2=2;
for i=1:100
        xi=x2-dt(x2)*(x2-x1)/(dt(x2)-dt(x1));
        x1=x2; x2=xi;
        if abs(x1-x2)<eps, break, end;
end;
printf('Tx=(%0.6f,%0.6f)\n',xi,f(xi));
```

```
Tx=(1.550027,1.245001)
```

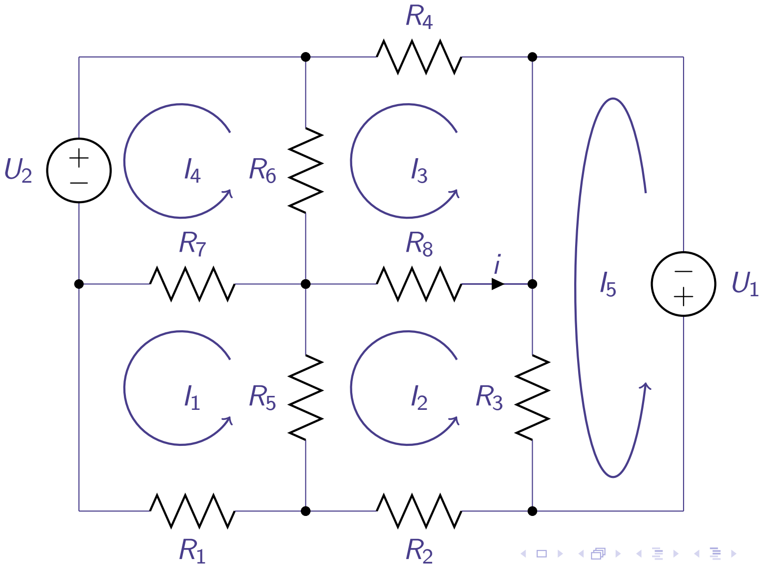
Sistemi linearnih nenačb

Električno vezje

Za vezje na sliki uporabi Kirchoffova zakona in zapiši sistem linearnih enačb. Koliko je i?
Vektor uporov $R = (20, 30, 40, 25, 10, 15, 34, 20)$,
napetosti $U_1 = 40$ in $U_2 = 35$.

$$
\begin{aligned}
0 &= R_7(I_1 - I_4) + R_5(I_1 - I_2) + R_1 I_1 \\
0 &= R_5(I_2 - I_1) + R_2 I_2 + R_3(I_2 - I_5) + R_8(I_2 - I_3) \\
0 &= R_4 I_3 + R_6(I_3 - I_4) + R_8(I_3 - I_2) \\
-U_2 &= R_7(I_4 - I_1) + R_6(I_4 - I_3) \\
-U_1 &= R_3(I_5 - I_2)
\end{aligned}
$$

Slika 2



Matrična oblika sistema $R\ I = U$

$$R =$$

$$\begin{bmatrix} R_1 + R_5 + R_7 & -R_5 & 0 & -R_7 & 0 \\ -R_5 & R_2 + R_3 - R_5 + R_8 & -R_8 & 0 & -R_3 \\ 0 & -R_8 & R_4 + R_6 + R_8 & -R_6 & 0 \\ -R_7 & 0 & -R_6 & R_6 + R_7 & 0 \\ 0 & -R_3 & 0 & 0 & R_3 \end{bmatrix}$$

$$I = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_5 \end{bmatrix}, \quad U = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -U_2 \\ -U_1 \end{bmatrix}, \quad i = I_3 - I_2$$

Program

```
r=[20,30,40,25,10,15,34,20];
U=[0; 0; 0; -35; -40];
R=[r(1)+r(5)+r(7),-r(5),0,-r(7),0;...
-r(5),r(2)+r(3)+r(5)+r(8),-r(8),0,-r(3);...
0,-r(8),r(4)+r(6)+r(8),-r(6),0;...
-r(7),0,-r(6),r(6)+r(7),0;...
0,-r(3),0,0,r(3)];
I=R\U;
printf('i=%0.6f\n',I(3)-I(2));
```

```
i=0.314720
```