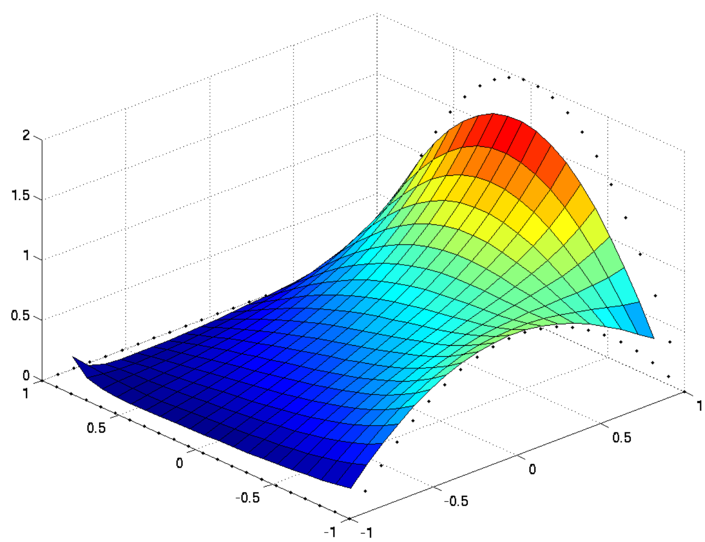
Linearni sistemi: iterativne metode

22. marec 2005

Naloga: Žično zanko s pravokotnim tlorisom potopimo v milnico tako, da se nanjo napne milni mehurček (glej sliko). Poišči obliko milnega mehurčka razpetega na žični zanki.

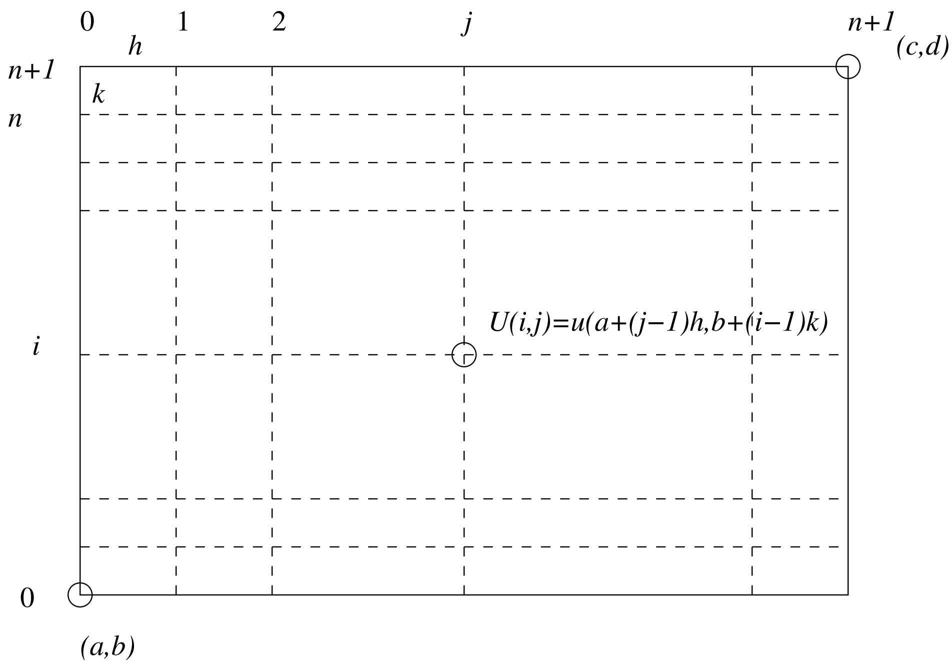


Slika 1: Graf funkcije, ki opisuje minico napeto na žično zanko.

Rešitev:

Ploskev milnega mehurčka lahko opišemo s funkcijo $u(x, y)$ dveh spremenljivk, ki predstavlja višino ploskve nad točko (x, y). Poiskati moramo funkcijo dveh spremenljivk $u(x, y)$ na tlorisu žične mreže, ki čim bolje opiše milnico.

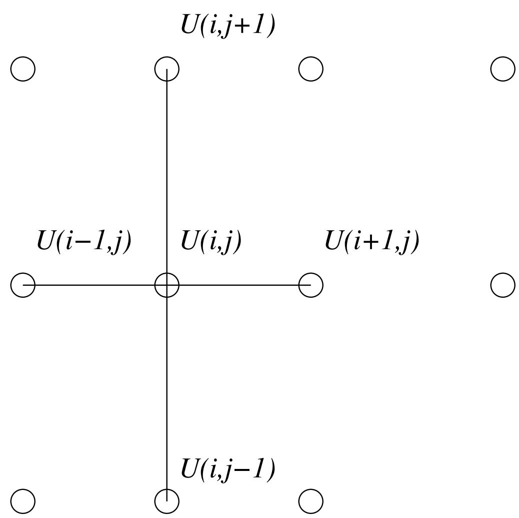
Problema se bomo lotili numerično tako, da bomo vrednosti $u(x, y)$ poiskali le v končno mnogo točkah: problem bomo *diskretizirali*. Najpreprosteje je uporabiti enakomerno razporejeno pravokotno (še bolje kvadratno) mrežo točk, v katerih računamo vrednosti u. Točke na mreži bomo imenovali vozlišča. Zaradi enostavnosti bomo obravnavali le mreže na kvadratu z enakim številom točk v obeh koordinatnih smereh.



Slika 2: Mreža vozlišč na tlorisu žične zanke

Namesto funkcije $u(x, y)$ iščemo matriko U dimenzije $n \times n$, v kateri bodo približne vrednosti u. Z (a, b) označimo levo spodnje, z (c, d), pa desno zgornje oglišče tlorisa zanke. Element $U(i, j)$ matrike U vsebuje vrednost $u(b + h(i-1), a + k(j-1))$, kjer je $h = \frac{d-b}{2}$ in $k = \frac{c-a}{2}$. Vrednosti funkcije u na robu pravokotnika opisujejo obliko žične zanke in so podane kot vhodni podatek. Poznane so torej vrednosti $U(0, j), U(n+1, j), U(i, 0)$ in $U(i, n+1)$ za poljubne vrednosti indeksov i in j.

Izpeljimo enačbo, ki povezuje vrednosti v U. Predpostavimo, da na vrednost $U(i, j)$ vplivajo le vrednosti najbližjih sosedov v mreži, in sicer $U(i-1, j)$, $U(i, j+1)$, $U(i+1, j)$ in $U(i, j-1)$.



Slika 3: Sosednja vozlišča, ki jih v enačbi upoštevamo.

V ravnovesni legi bo vsota vseh sil, ki delujejo na dano točko (x_i, y_i) enaka nič. Sila med dvema sosednjima točkama je premo sorazmerna njuni višinski razliki (glej sliko).

Vsota vseh sil, ki delujejo na točko (x_i, y_i) je enaka

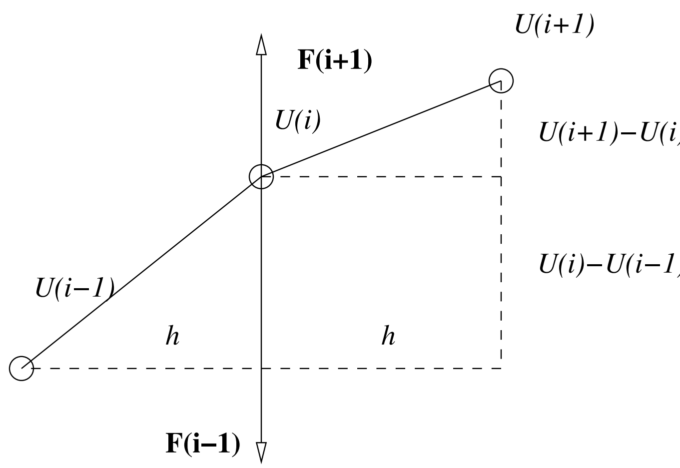
$$\begin{aligned} k((U_{i-1,j} - U_{i,j}) + (U_{i,j+1} - U_{i,j}) + (U_{i+1,j} - U_{i,j}) + (U_{i,j-1} - U_{i,j})) &= 0 \\ U_{i-1,j} + U_{i,j-1} + U_{i+1,j} + U_{i,j+1} - 4U_{i,j} &= 0 \end{aligned}$$

V vsakem notranjem vozlišču mreže dobimo eno linearno enačbo, ki skupaj tvorijo sistem n^2 linearnih enačb z n^2 neznankami. Ne pozabimo, da so vrednosti U znane v vozliščih na robu.

$$\begin{aligned} -4U_{1,1} + U_{1,2} + U_{2,1} = -U_{1,0} - U_{0,1} &= b_1 \\ U_{1,1} - 4U_{1,2} + U_{1,3} + U_{2,2} = -U_{0,2} &= b_2 \\ &\vdots \\ U_{n-1,n} + U_{n,n-1} - 4U_{n,n} = -U_{n,n+1} - U_{n+1,n} &= b_n \end{aligned} \tag{1}$$

Če želimo sistem zapisati v matrični obliki, moramo elemente matrike $U_{i,j}$ zložiti v stolpec. Stolpce matrike $U_{i,j}$ razvrstimo enega pod drugim in dobimo vektor u dolžine n^2.

$$u = [U_{1,1}, U_{2,1}, U_{3,1}, \ldots, U_{1,2}, U_{2,2}, \ldots, U_{n,1}, U_{n,2}, \ldots, U_{n,n}]^T$$



Slika 4: Sile med sosednjimi točkami v mreži.

Element matrike $U_{i,j} = u(j(n-1)+i)$. Enačbe (1) se sedaj glasijo

$$
\begin{aligned}
-4u_1 + u_2 + u_{n+1} &= b_1 \\
u_1 - 4u_2 + u_3 + u_{n+2} &= b_2 \\
&\vdots \\
u_1 - 4u_{n+1} + u_{n+2} + u_{2n+1} &= b_{n+1} \\
&\vdots \\
u_i + u_{n+i-1} - 4u_{n+i} + u_{n+i+1} + u_{2n+i} &= b_{n+i} \\
&\vdots \\
u_{n^2-n} + u_{n^2-1} - 4u_{n^2} &= b_{n^2}
\end{aligned}
$$

Matrika sistema je bločna

$$
A = \begin{pmatrix}
L & I & 0 & \cdots & \cdots & 0 \\
I & L & I & 0 & \cdots & 0 \\
0 & I & L & I & \ddots & \vdots \\
\vdots & 0 & \ddots & \ddots & \ddots & 0 \\
\vdots & \vdots & \ddots & I & L & I \\
0 & \cdots & \cdots & 0 & I & L
\end{pmatrix}
$$

z bloki velikosti $n \times n$, kjer so matrike I identične $n \times n$ matrike in L

Laplaceove matrike oblike

$$L = \begin{pmatrix} -4 & 1 & 0 & \cdots & 0 \\ 1 & -4 & 1 & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & 1 & -4 & 1 \\ 0 & \cdots & 0 & 1 & -4 \end{pmatrix}.$$

Za rešitev sistema potrebujemo še vektor desnih strani, ki ga izračunamo iz robnih pogojev.

$$\begin{aligned} b = -[U_{0,1} + U_{1,0}, U_{0,2}, \ldots, U_{0,n} + U_{1,n+1}, \\ U_{2,0}, 0, \ldots, 0, U_{2,n+1}, \\ U_{3,0}, \ldots, \\ \ldots \\ U_{n,0} + U_{n+1,1}, U_{n+1,2}, \ldots, U_{n+1,n} + U_{n,n+1}]^T \end{aligned} \qquad (2)$$

Reševanje sistema z Jacobijevo iteracijo

Ker je matrika sistema skoraj prazna, je smiselno uporabiti katero od iteracijskih metod. Predvsem prostorski prihranek bo tako precejšen in bo omogočil raševanje sistema tudi za zelo velike dimenzije, ko bi sicer zmanjkalo pomnilnika. Matrika sistema je *diagonalno dominantna* po vrsticah in po stolpcih, ni pa strogo diagonalno dominantna. Zato ne moremo pričakovati hitre konvergence.

Ker elementi matrike ne spreminjajo veliko, bomo matriko sistema vgradili kar v algoritem.

Rešitev sistema $Ax = b$ z *Jacobijevo* iteracijo dobimo kot zaporedje približkov

$$Dx_{n+1} = b - (S + Z)x_n,$$

kjer je D diagonala matrike A, $S+Z$ pa preostanek matrike A brez diagonale.

$$x_i^{k+1} = \frac{1}{a_{i,i}} \left(b_i - \sum_{j=1, j \neq i}^{n} a_{i,j} x_j^k \right) \qquad (3)$$

Za našo matriko bodo v vsoti nastopali največ štirje členi

$$x_i^{k+1} = \frac{1}{-4} \left(b_i - x_{i-1}^k - x_{i+1}^k - x_{i-n}^k - x_{i+n}^k \right), \quad i \neq kn+1, kn-1$$

$$x_i^{k+1} = \frac{1}{-4}\left(b_i - x_{i+1}^k - x_{i-n}^k - x_{i+n}^k\right), \quad i = kn+1$$

$$x_i^{k+1} = \frac{1}{-4}\left(b_i - x_{i-1}^k - x_{i-n}^k - x_{i+n}^k\right), \quad i = kn-1$$

pri čemer so elementi x_j z indeksom $j < 0$ ali $j > n^2$ enaki 0.

Gauss-Seidlova iteracija

Konvergenco iteracije lahko izboljšamo, če pri Jacobijevi iteraciji v formuli (3) uporabimo tiste komponente novega približka $\mathbf{x}^{k+1}$, ki jih že poznamo. Tako dobimo *Gauss-Seidlovo iteracijo*

$$x_i^{k+1} = \frac{1}{a_{i,i}}\left(b_i - \sum_{j=1}^{i-1} a_{i,j}x_j^{k+1} - \sum_{j=i+1}^{n} a_{i,j}x_j^k\right) \tag{4}$$

Za našo matriko, dobimo naslednjo formulo za izračun naslednjega približka.

$$x_i^{k+1} = \frac{1}{-4}\left(b_i - x_{i-1}^{k+1} - x_{i+1}^k - x_{i-n}^{k+1} - x_{i+n}^k\right), i \neq kn \pm 1.$$

Podobno naredimo še v ostalih vrsticah.

SOR iteracija

Kot smo se prepričali na zgornjem primeru, je konvergenca Jacobieve in Gauss-Seidlove iteracije včasih kaj klavrna. Zato so pred nekaj desetletji razvili iteracijsko shemo *SOR*. Pri tej metodi približke dobljene z Gauss-Seidlovo iteracijo "relaksiramoš prejšnjim približkom. Formule (4) popravimo nov približek po Gauss-Seidlu pomnožimo z ω in mu prištejemo $(1-\omega)$ krat prejšnji približek.

$$x_i^{k+1} = \omega\left(\frac{1}{a_{i,i}}\left(b_i - \sum_{j=1}^{i-1} a_{i,j}x_j^{k+1} - \sum_{j=i+1}^{n} a_{i,j}x_j^k\right)\right) + (1-\omega)x_i^k. \tag{5}$$

Tako dobimo celo družino iteracij parametrizirano z ω. Za $\omega = 1$ dobimo Gauss-Seidlovo iteracijo. Konvergenca je seveda odvisna od izbire ω in mogoče je pokazati, da SOR ne konvergira za $\omega \notin (0, 2)$.

Primer

Poglejmo si še primer, ki je na začetni sliki. Tloris zanke je kvadrat $[-1,1] \times [-1,1]$. Ploskev je na robu kvadrata podana s funkcijami $u(x,1) = 1 - x^2$, $u(1,y) = 2 - 2x^2$ in $u(x,-1) = u(-1,y) = 0$. Slika na začetku je izračunana na mreži 20×20.