Az áramlástani problémák diszkretizálásával kapott algebrai egyenletrendszer megoldása

Dr. Kristóf Gergely

2014.10.27.

A Poisson-egyenletet minden időlépésnél meg kell oldani...

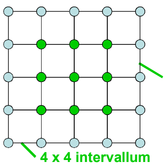
Ezt a feladatot nem tudjuk elkerülni inkompresszibilis áramlások esetén.

Ψ-ω módszer esetében: $\Delta\psi = -\omega \longrightarrow \psi$

Nyomásalapú megoldók esetében: $\Delta P = \nabla \cdot \underline{f} \longrightarrow P$

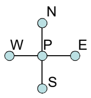
Egyszerű 2D példa

A számítási tartomány:



$$\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} = Q$$

Diszkretizáljuk véges differenciák módszerével:

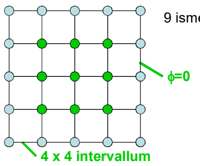


$$\frac{1}{\Delta x}\left(\frac{\phi_E - \phi_P}{\Delta x} - \frac{\phi_P - \phi_W}{\Delta x}\right) + \frac{1}{\Delta y}\left(\frac{\phi_N - \phi_P}{\Delta y} - \frac{\phi_P - \phi_S}{\Delta y}\right) = Q_P$$

Pl: $\Delta x = \Delta y = h$ esetén: $\boxed{\phi_S + \phi_W - 4\phi_P + \phi_E + \phi_N = h^2 Q_P}$

Mátrixos alakban

$$\phi_S + \phi_W - 4\phi_P + \phi_E + \phi_N = h^2 Q_P$$



9 ismeretlenünk van.

$$A_{i,j} = \begin{bmatrix} -4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & -4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & -4 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & -4 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & -4 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & -4 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & -4 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & -4 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & -4 \end{bmatrix}$$

Az egyenletrendszer mátrixos alakban:

$$\boxed{A_{i,j}\,\varphi_i = Q_i}$$

Pl. 101 x 101 es háló esetén az ismeretlenek száma $N=10^4$, **A** elemeinek száma pedig 10^8.

Gauss-elimináció

Általános mátrix esetében épp olyan jó, mint bármilyen más módszer, viszont a mátrix kedvező tulajdonságait nem használja ki.

1. lépés Elimináció:

$$\begin{pmatrix} A_{1,1} & A_{1,2} & A_{1,3} \\ A_{2,1} & A_{2,2} & A_{2,3} \\ A_{3,1} & A_{3,2} & A_{3,3} \end{pmatrix}$$

Az első sor A_{21}/A_{11}-szeresét kivonjuk a második sorból, így ott az első elem 0 lesz. Ugyanez minden további sorra. } Minden további oszlopra az N-1 –edik oszlopig.

2. lépés Visszahelyettesítés:

$$\begin{pmatrix} U_{1,1} & U_{1,2} & U_{1,3} \\ 0 & U_{2,2} & U_{2,3} \\ 0 & 0 & U_{3,3} \end{pmatrix}$$

$$\phi_n = \frac{Q_n}{U_{nn}}$$

$$\phi_i = \frac{Q_i - \sum_{k=k+1}^{N} U_{k,i}\,\phi_k}{U_{i,i}}$$

A műveletigény $N^3/3$, de ebből a visszahelyettesítés csak $N^2/2$.
Hiába ritka A mátrix, az U mátrix már nem ritka. Memóriaigény 101 x 101 es hálón kb. 400 Mb. Továbbá: **Nem is szükséges nagyon pontos megoldás, mert a diszkretizációs hiba jelentős.**

Iteratív módszerek

A megoldást lépésenként finomítjuk. ϕ közelítése az n-edik lépésben ϕ^n.

Elhagyva a vektorindexeket: $A\,\phi^n = Q - \rho^n$ **ρ^n : reziduum**

A hiba: $\varepsilon^n = \phi - \phi^n$

$$A\,\varepsilon^n = A\left(\phi - \phi^n\right) = Q - \left(Q - \rho^n\right) = \rho^n$$

Tehát ha megoldjuk A mátrixot a hibára, n=1 lépésben megkapjuk a pontos megoldást. Viszont **közelíthetjük** is A-t!

Iteratív módszerek: $M\,\phi^{n+1} = N\phi^n + Q$

Bekonvergált megoldásra: $\phi^{n+1} = \phi^n = \phi$, ezért: $A = M - N$

Mindkét oldalból vonjunk le $M\phi^n$ –et:

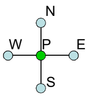
$$M\underbrace{\left(\phi^{n+1} - \phi^n\right)}_{\delta^n} = N\,\phi^n + Q - M\,\phi^n = Q - A\,\phi^n = \rho^n$$

korrekció: δ^n

$M\,\delta^n = \rho^n$ Korrekciós egyenlet.

Minél jobban közelíti M az A mátrixot, annál gyorsabban konvergál.
M lehet pl. diagonál, tridiagonál, vagy Δ mátrix. **δ-nak sem kell pontosnak lenni...**

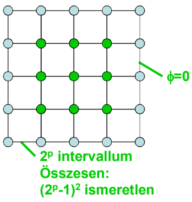
Jacobi-iteráció



$$\phi_S^n + \phi_W^n - 4\phi_P^{n+1} + \phi_E^n + \phi_N^n = h^2 Q_P$$

M diagonál mátrix lesz.

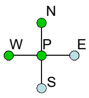
$$\phi_P^{n+1} = \frac{1}{4}\left(\phi_S^n + \phi_W^n + \phi_E^n + \phi_N^n - h^2 Q_P\right)$$



Példaprogram:

- A program...
- Az eredmény jellege...
- Szükséges iterációszám...

Gauss-Seidel iteráció



$$\underbrace{\phi_S^n + \phi_W^{n+1}}_{} - 4\phi_P^{n+1} + \underbrace{\phi_E^n + \phi_N^{n+1}}_{} = h^2 Q_P$$

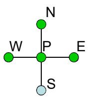
M Δ mátrix lesz.

ezeket már ismerjük a számítási sorrend miatt (lexikografikus séma)

$$\phi_P^{n+1} = \frac{1}{4}\left(\phi_S^n + \phi_W^{n+1} + \phi_E^n + \phi_N^{n+1} - h^2 Q_P\right)$$

- Fele annyi iterációt igényel
- Nem kell új tömb a változóknak
- A hiba aszimmetrikus.

Vonalrelaxáció



$$\phi_S^n + \underbrace{\phi_W^{n+1} - 4\phi_P^{n+1} + \phi_E^{n+1}}_{\text{ezeket egyszerre határozzuk meg Thomas algoritmus segítségével.}} + \underbrace{\phi_N^{n+1}}_{\text{ezt már ismerjük}} = h^2 Q_P$$

Ugyancsak tridiagonál megoldóra épül az ADI (más néven operator splitting) módszer, ami ennél sokkal hatékonyabb megoldást tesz lehetővé.

Probléma:
Az eddigi módszerek csak simítanak, ezért a peremek hatása nagyon lassan terjed be a finom hálókon. → Durvább rácsokat is használni kell.
A korrekciós egyenletet kell durvább rácsra levinni, mert ezt pontatlanul (nagyobb relatív hibával) is megoldhatjuk.

Multigrid módszer

Vegyünk például egy egyszerű egydimenziós feladatot:

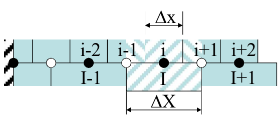
$$\frac{\partial^2 \phi}{\partial x^2} = Q$$

$$\frac{1}{\Delta x^2}\left(\phi_{i-1} - 2\phi_i + \phi_{i+1}\right) = Q_i$$

$$\frac{1}{\Delta x^2}\left(\phi_{i-1}^n - 2\phi_i^n + \phi_{i+1}^n\right) = Q_i - \rho_i^n$$

$$\frac{1}{\Delta x^2}\left(\varepsilon_{i-1}^n - 2\varepsilon_i^n + \varepsilon_{i+1}^n\right) = \rho_i^n$$

Elhagyjuk az iterációs indexet: $\frac{1}{\Delta x^2}\left(\varepsilon_{i-1} - 2\varepsilon_i + \varepsilon_{i+1}\right) = \rho_i$



$$\frac{1}{\Delta x^2}\left(\frac{1}{2}\varepsilon_{i-2} \underbrace{-\varepsilon_{i-1}}_{\text{ezek kiesnek}} + \frac{1}{2}\varepsilon_i \underline{+\varepsilon_{i-1}} - 2\varepsilon_i \underline{+\varepsilon_{i+1}} + \frac{1}{2}\varepsilon_i \underline{-\varepsilon_{i+1}} + \frac{1}{2}\varepsilon_{i+2}\right) =$$

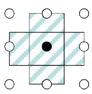
$$= \frac{1}{2}\rho_{i-1} + \rho_i + \frac{1}{2}\rho_{i+1}$$

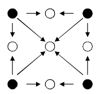
$$\frac{1}{4\Delta x^2}\left(\varepsilon_{i-2} - 2\varepsilon_i + \varepsilon_{i+2}\right) = \frac{1}{4}\left(\rho_{i-1} + 2\rho_i + \rho_{i+1}\right)$$

$$\frac{1}{\Delta X^2}\left(\varepsilon_{I-1} - 2\varepsilon_I + \varepsilon_{I+1}\right) = \rho_I$$

restrikciós séma (2D és 3D esetekben közelítő jellegű, ezért ε-t jelölhetnénk δ-val is.)

Általánosítás 2D esetre:

Restrikció: 

Prolongálás: 

1. ρ_i restrikciója → ρ_I
2. ε_I számítása ¼ annyi ismeretlen meghatározása (és ¼ annyi iteráció). Az időigény szinte elhanyagolható.
3. ε_I prolongálása a finom rácsra (ε_i), majd egy simítás a finom rácson.

Miért ne mennénk le még durvább rácsokra?

1. Reziduumok kiszámítása a legfinomabb rácson
2. Reziduumok restrikciója minden durvább rácsra
3. Egyenletrendszer megoldása a legdurvább rácson
4. Minden finomabb rácsra:
   - Korrekció prolongálása
   - Utósimítás

A megoldás műveletigénye

Szükséges iterációk száma 2D-ben:

p	Nsor	N	Jacobi	G-S	Vonalrelax	Multigrid
3	7	49	40	20	10	11
4	15	225	160	80	40	15
5	31	961	640	320	160	37
6	63	3969	2560	1280	640	43
7	127	16129	10240	5120	2560	44

Műveletigény / N:

p	Nsor	N	Jacobi	G-S	Vonalrelax	Multigrid
3	7	49	200	100	50	220
4	15	225	800	400	200	300
5	31	961	3200	1600	800	740
6	63	3969	12800	6400	3200	860
7	127	16129	51200	25600	12800	880

finom háló