Numerikus módszerek 1.

10. előadás: Részleges LU-felbontás és algoritmus, kerekítési hibák hatása az iterációkra

Krebsz Anna

ELTE IK

Emlékeztető: iterációs módszerek

Általában:

$$Ax = b, \qquad A = P + Q, \qquad (P+Q)x = b,$$

átrendezve:

$$Px = -Qx + b \quad \Longleftrightarrow \quad x = -P^{-1}Qx + P^{-1}b,$$

iterációs alakban írva:

$$x^{(k+1)} = \underbrace{-P^{-1}Q}_{B} \cdot x^{(k)} + \underbrace{P^{-1}b}_{c}.$$

A továbbiakban olyan $P = LU$ felbontást és $-Q$ mátrixot keresünk, melyre $A = P - Q$. Ekkor a P^{-1}-zel való számolás helyettesíthető két háromszög alakú LER megoldásával, vagyis az iteráció könnyen számolható. Ezzel egy módszercsaládot konstruálunk.

Definíció: ILU-felbontás

- Legyen J a mátrix elemek pozícióinak egy részhalmaza, mely nem tartalmazza a főátlót, azaz $(i,i) \notin J \quad \forall\, i$-re.
  A J halmazt *pozícióhalmaz*nak nevezzük.
- Az A mátrixnak a J pozícióhalmazra illeszkedő *részleges LU-felbontásán* (*ILU-felbontásán*) olyan LU-felbontást értünk, melyre $L \in \mathcal{L}_1$ és $U \in \mathcal{U}$ (tehát a szokásos alakúak), továbbá

$$\forall\,(i,j) \in J: \;\; l_{ij} = 0, \;\; u_{ij} = 0 \;\text{ és}$$
$$\forall\,(i,j) \notin J: \;\; a_{ij} = (LU)_{ij}.$$

Algoritmus: ILU-felbontás GE-val

$\widetilde{A}_1 := A$
$k = 1, \dots, n-1$:

(1) Szétbontás: $\widetilde{A}_k = P_k - Q_k$ alakra, ahol

$$\begin{aligned}
(P_k)_{ik} &= 0 \quad (i,k) \in J \\
(P_k)_{kj} &= 0 \quad (k,j) \in J \\
(Q_k)_{ik} &= -\widetilde{a}_{ik}^{(k)} \quad (i,k) \in J \\
(Q_k)_{kj} &= -\widetilde{a}_{kj}^{(k)} \quad (k,j) \in J.
\end{aligned}$$

Ahogy látható, $\widetilde{A}_k$-nak csak k. sorában és k. oszlopában a pozícióhalmazban megadott helyeken változtatunk.

(2) Elimináció P_k-n:

$$\widetilde{A}_{k+1} = L_k P_k$$

Kérdés: az algoritmussal kapott mátrixokból hogyan állítjuk elő az ILU-felbontást?

Tétel: az ILU-felbontásról

Az ILU-felbontás algoritmusával kapott részmátrixokból készítsük el a következőket:

$$U := \widetilde{A}_n,$$
$$L := L_1^{-1} \cdot \ldots \cdot L_{n-1}^{-1} \quad \text{(összepakolással)},$$
$$Q := Q_1 + Q_2 + \ldots + Q_{n-1} \quad \text{(összepakolással)}.$$

Ekkor $A = LU - Q$ és a részleges LU-felbontásra vonatkozó feltételek teljesülnek.

Biz.: A GE $n-1$. lépése után felsőháromszög alakot kapunk, tehát $U := \widetilde{A}_n$ alakja jó és minden $(i,j) \in J, i < j$-re $u_{ij} = 0$. Alkalmazzuk az $n-1$. lépés (2), majd (1) részét:

$$U := \widetilde{A}_n = L_{n-1} P_{n-1} = L_{n-1}\left(\widetilde{A}_{n-1} + Q_{n-1}\right)$$

Az $\widetilde{A}_n$-re kapott rekurziót alkalmazzuk $\widetilde{A}_{n-1}$-re:

$$\widetilde{A}_n = L_{n-1}\left(\widetilde{A}_{n-1} + Q_{n-1}\right) = L_{n-1}\left(L_{n-2}\left[\widetilde{A}_{n-2} + Q_{n-2}\right] + Q_{n-1}\right)$$

Mivel Q_{n-1}-ben az $n-2$. sorban csak nullák vannak, így az $n-2$. GE-s lépés nem változtat rajta, tehát $L_{n-2} Q_{n-1} = Q_{n-1}$. Emiatt Q_{n-1}-et bevihetjük a belső zárójelbe.

$$\widetilde{A}_n = L_{n-1} L_{n-2}\left(\widetilde{A}_{n-2} + Q_{n-2} + Q_{n-1}\right)$$

Biz. folyt.: Folytatva tovább visszafelé a rekurziót

$$U = \widetilde{A}_n = L_{n-1}L_{n-2}\left(\widetilde{A}_{n-2} + Q_{n-2} + Q_{n-1}\right) = \ldots =$$

$$= \underbrace{L_{n-1}L_{n-2}\ldots L_1}_{L^{-1}}\left(A + \underbrace{Q_1 + \ldots + Q_{n-2} + Q_{n-1}}_{Q}\right).$$

$$U = L^{-1}(A+Q) \quad \Leftrightarrow \quad A = LU - Q$$

A kapott mátrixok alakja megfelelő. Az algoritmus (1) lépése garantálja, hogy $\forall\,(i,j) \in J: \;\; l_{ij} = 0, \;\; u_{ij} = 0$, továbbá (2) lépése (GE) miatt $\forall\,(i,j) \notin J: \;\; a_{ij} = (LU)_{ij}$. $\square$

Tétel: szig.diag.dom. mátrix ILU-felbontása

Ha A szigorúan diagonálisan domináns a soraira vagy oszlopaira, akkor a mátrix ILU-felbontása létezik és egyértelmű.

Biz.: az ILU-felbontás (1) lépése a szig. diag. dom. tulajdonságot nem változtatja, mivel átlón kívüli elemet veszünk ki a mátrixból.

A (2) GE-s lépés a szig. diag. dom. tulajdonságot magtartja, lásd GE megmaradási tételek a Schur-komplementerre. $\square$

Megjegyzés:

1. A szig. diag. dom. tulajdonságból következik az összes bal felső részmátrix invertálhatósága, vagyis a főminorok egyike sem nulla.
2. Diff. egyenletek numerikus megoldása során gyakran előforduló M-mátrix osztályra is igaz, hogy egyértelműen létezik az ILU-felbontása.
3. Gyakran csak a főátlót és néhány mellékátlót hagynak ki a J pozícióhalmazból, így a tárigény előre ismert, nem kell a sávon belül feltöltődéssel foglalkozni.

4. Például egy $N^2 \times N^2$-es mátrix esetén, ahol csak a $(-N, -1, 0, 1, N)$ átlókban van nem nulla elem, érdemes J-ből a $(-1, 0, 1)$ átlókat kihagyni.

   Tárolás: L, U csak két-két átlót fog tartalmazni, L átlója egyesekből áll, így 3 db N^2 méretű átlót kell tárolni N^4 elem helyett.

   Műveletigény: az iteráció során a két háromszögmátrixú két átlós LER $2N^2 + \mathcal{O}(1)$ illetve $3N^2 + \mathcal{O}(1)$ művelettel megoldható. (A GE $\frac{2}{3}N^6$-t jelentene.) Gondoljunk arra, hogy $N \approx 10^3$...

1. Példa:

Készítsük el az

$$A = \begin{bmatrix} 4 & 1 & 2 \\ 1 & 4 & 1 \\ 2 & 1 & 4 \end{bmatrix}$$

mátrix $J = \{(1,2),(2,3)\}$ pozícióhalmazhoz illeszkedő ILU-felbontását! A pozícióhalmazt mátrixos alakban is szemléltethetjük, a kinullázandó elemeket $*$-gal jelöljük:

$$\begin{bmatrix} & * & \\ & & * \\ & & \end{bmatrix}.$$

1. lépés: (1) **szétbontás:** olyan pozíciókat keresünk J-ben, melyek az 1. sorhoz illetve az 1. oszlophoz tartoznak: $(1,2)$.
Ezt a pozíciót kinullázzuk P_1-ben és a (-1)-szeresét Q_1-be tesszük.

$$A = \widetilde{A}_1 = \begin{bmatrix} 4 & 1 & 2 \\ 1 & 4 & 1 \\ 2 & 1 & 4 \end{bmatrix} = \begin{bmatrix} 4 & 0 & 2 \\ 1 & 4 & 1 \\ 2 & 1 & 4 \end{bmatrix} - \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} = P_1 - Q_1$$

(2) **Elimináció:** P_1-en elvégezzük az 1. GE-s lépést:

$$\widetilde{A}_2 = L_1 P_1 = \begin{bmatrix} 4 & 0 & 2 \\ 0 & 4 & \frac{1}{2} \\ 0 & 1 & 3 \end{bmatrix}, \quad L_1^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ \frac{1}{4} & 1 & 0 \\ \frac{1}{2} & 0 & 1 \end{bmatrix}.$$

2. lépés: (1) **szétbontás:** olyan pozíciókat keresünk J-ben, melyek a 2. sorhoz illetve az 2. oszlophoz tartoznak: $(2,3)$.
Ezt a pozíciót kinullázzuk P_2-ben és a (-1)-szeresét Q_2-be tesszük.

$$\widetilde{A}_2 = \begin{bmatrix} 4 & 0 & 2 \\ 0 & 4 & \frac{1}{2} \\ 0 & 1 & 3 \end{bmatrix} = \begin{bmatrix} 4 & 0 & 2 \\ 0 & 4 & 0 \\ 0 & 1 & 3 \end{bmatrix} - \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{2} \\ 0 & 0 & 0 \end{bmatrix} = P_2 - Q_2$$

(2) **Elimináció:** P_2-en elvégezzük a 2. GE-s lépést:

$$\widetilde{A}_3 = L_2 P_2 = \begin{bmatrix} 4 & 0 & 2 \\ 0 & 4 & 0 \\ 0 & 0 & 3 \end{bmatrix}, \quad L_2^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \frac{1}{4} & 1 \end{bmatrix}.$$

A tétel alapján összerakjuk az ILU-felbontást:

$$U = \widetilde{A}_3 = \begin{bmatrix} 4 & 0 & 2 \\ 0 & 4 & 0 \\ 0 & 0 & 3 \end{bmatrix}, \quad Q = Q_1 + Q_2 = \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & -\frac{1}{2} \\ 0 & 0 & 0 \end{bmatrix}$$

Mivel az egyes lépésekben a Q_k mátrixok különböző sorait és oszlopait töltjük, így elegedő a gyakorlatban egy Q mátrixot tárolni. Az iterációnál látni fogjuk, hogy Q-ra a végrehajtáshoz nincs szükség.

Összepakolással:

$$L = L_1^{-1} L_2^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ \frac{1}{4} & 1 & 0 \\ \frac{1}{2} & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \frac{1}{4} & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ \frac{1}{4} & 1 & 0 \\ \frac{1}{2} & \frac{1}{4} & 1 \end{bmatrix}.$$

Ellenőrizhetjük, hogy $A = LU - Q$

$$\begin{bmatrix} 4 & 1 & 2 \\ 1 & 4 & 1 \\ 2 & 1 & 4 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ \frac{1}{4} & 1 & 0 \\ \frac{1}{2} & \frac{1}{4} & 1 \end{bmatrix} \cdot \begin{bmatrix} 4 & 0 & 2 \\ 0 & 4 & 0 \\ 0 & 0 & 3 \end{bmatrix}}_{\begin{bmatrix} 4 & 0 & 2 \\ 1 & 4 & \frac{1}{2} \\ 2 & 1 & 4 \end{bmatrix}} - \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & -\frac{1}{2} \\ 0 & 0 & 0 \end{bmatrix}.$$

Teljesíti a ILU-felbontásra tett összes követelményt. $\square$

Tömör írásmódban: Csak egy Q mátrixot tárolunk, ebbe pakoljuk a Q_k mátrixok nem nulla elemeit. Az GE eredményét illetve a GE-s hányadosokat, vagyis az $\widetilde{A}_k, L_k$ mátrixokat is egyben tároljuk. Vonalakkal jelezzük, hogy itt már tárolásról is szó van.

1. lépés: (1) **szétbontás:**

$$A = \widetilde{A}_1 = \begin{bmatrix} 4 & 1 & 2 \\ 1 & 4 & 1 \\ 2 & 1 & 4 \end{bmatrix}, \quad P_1 = \begin{bmatrix} 4 & 0 & 2 \\ 1 & 4 & 1 \\ 2 & 1 & 4 \end{bmatrix}, \quad Q = \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

(2) **Elimináció P_1-en:**

$$\begin{bmatrix} 4 & 0 & 2 \\ 1 & 4 & 1 \\ 2 & 1 & 4 \end{bmatrix} \to \left[\begin{array}{c|cc} 4 & 0 & 2 \\ \hline \frac{1}{4} & 4 & \frac{1}{2} \\ \frac{2}{4} & 1 & 3 \end{array}\right]$$

2. lépés: Ugyanúgy dolgozunk tovább, de most már csak a jobb alsó 2×2-es mátrix részen, a többit változatlanul leírjuk.

(1) **szétbontás:**

$$\left[\begin{array}{c|cc} 4 & 0 & 2 \\ \hline \frac{1}{4} & 4 & \frac{1}{2} \\ \frac{1}{2} & 1 & 3 \end{array}\right] \to \left[\begin{array}{c|cc} 4 & 0 & 2 \\ \hline \frac{1}{4} & 4 & 0 \\ \frac{1}{2} & 1 & 3 \end{array}\right] \qquad Q = \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & -\frac{1}{2} \\ 0 & 0 & 0 \end{bmatrix}$$

(2) **Elimináció:**

$$\left[\begin{array}{c|cc} 4 & 0 & 2 \\ \hline \frac{1}{4} & 4 & 0 \\ \frac{1}{2} & 1 & 3 \end{array}\right] \to \left[\begin{array}{c|c|c} 4 & 0 & 2 \\ \hline \frac{1}{4} & 4 & 0 \\ \cline{2-3} \frac{1}{2} & \frac{1}{4} & 3 \end{array}\right] = L \text{ és } U \text{ együtt}$$

$\square$

2. Példa:

Készítsük el az

$$A = \begin{bmatrix} 4 & 1 & 2 \\ 1 & 4 & 1 \\ 2 & 1 & 4 \end{bmatrix}$$

mátrix $J = \{(1,2),(1,3),(2,1),(2,3),(3,1),(3,2)\}$ pozícióhalmazhoz illeszkedő ILU-felbontását! A pozícióhalmazt mátrixos alakban is szemléltethetjük, a kinullázandó elemeket $*$-gal jelöljük:

$$\begin{bmatrix} & * & * \\ * & & * \\ * & * & \end{bmatrix}.$$

A lehető legbővebb pozícióhalmazt adtuk meg.

1. lépés: (1) **szétbontás:** olyan pozíciókat keresünk J-ben, melyek az 1. sorhoz illetve az 1. oszlophoz tartoznak:
$(1,2),(1,3),(2,1),(3,1)$.
Ezeket a pozíciókat kinullázzuk P_1-ben és a (-1)-szeresüket Q_1-be tesszük.

$$A = \widetilde{A}_1 = \begin{bmatrix} 4 & 1 & 2 \\ 1 & 4 & 1 \\ 2 & 1 & 4 \end{bmatrix} = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 1 \\ 0 & 1 & 4 \end{bmatrix} - \begin{bmatrix} 0 & -1 & -2 \\ -1 & 0 & 0 \\ -2 & 0 & 0 \end{bmatrix} = P_1 - Q_1$$

(2) **Elimináció:** P_1-en elvégezzük az 1. GE-s lépést (valójában nem kell eliminálnunk a kinullázások miatt):

$$\widetilde{A}_2 = L_1 P_1 = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 1 \\ 0 & 1 & 4 \end{bmatrix}, \quad L_1^{-1} = I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

2. lépés: (1) **szétbontás:** olyan pozíciókat keresünk J-ben, melyek a 2. sorhoz illetve az 2. oszlophoz tartoznak: $(2,3),(3,2)$. Ezeket a pozíciókat kinullázzuk P_2-ben és a (-1)-szeresüket Q_2-be tesszük.

$$\widetilde{A}_2 = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 1 \\ 0 & 1 & 4 \end{bmatrix} = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{bmatrix} - \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & -1 & 0 \end{bmatrix} = P_2 - Q_2$$

(2) **Elimináció:** P_2-en elvégezzük a 2. GE-s lépést (valójában nem kell eliminálnunk a kinullázások miatt):

$$\widetilde{A}_3 = L_2 P_2 = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{bmatrix}, \quad L_2^{-1} = I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

A tétel alapján összerakjuk az ILU-felbontást:

$$U = \widetilde{A}_3 = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{bmatrix}, \quad Q = Q_1 + Q_2 = \begin{bmatrix} 0 & -1 & -2 \\ -1 & 0 & -1 \\ -2 & -1 & 0 \end{bmatrix}$$

Mivel az egyes lépésekben a Q_k mátrixok különböző sorait és oszlopait töltjük, így elegedő a gyakorlatban egy Q mátrixot tárolni. Az iterációnál látni fogjuk, hogy Q-ra a végrehajtáshoz nincs szükség.

Összepakolással:

$$L = L_1^{-1} L_2^{-1} = I.$$

Ellenőrizhetjük, hogy $A = LU - Q$

$$\begin{bmatrix} 4 & 1 & 2 \\ 1 & 4 & 1 \\ 2 & 1 & 4 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{bmatrix} - \begin{bmatrix} 0 & -1 & -2 \\ -1 & 0 & -1 \\ -2 & -1 & 0 \end{bmatrix}.$$

Teljesíti a *ILU*-felbontásra tett összes követelményt. $\square$

Tömör írásmódban: Csak egy Q mátrixot tárolunk, ebbe pakoljuk a Q_k mátrixok nem nulla elemeit. Az GE eredményét illetve a GE-s hányadosokat, vagyis az $\widetilde{A}_k, L_k$ mátrixokat is egyben tároljuk. Vonalakkal jelezzük, hogy itt már tárolásról is szó van.

1. lépés: (1) **szétbontás:**

$$A = \widetilde{A}_1 = \begin{bmatrix} 4 & 1 & 2 \\ 1 & 4 & 1 \\ 2 & 1 & 4 \end{bmatrix}, \quad P_1 = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 1 \\ 0 & 1 & 4 \end{bmatrix}, \quad Q = \begin{bmatrix} 0 & -1 & -2 \\ -1 & 0 & 0 \\ -2 & 0 & 0 \end{bmatrix}$$

(2) **Elimináció P_1-en:** valójában nem kell eliminálni.

$$\begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 1 \\ 0 & 1 & 4 \end{bmatrix} \to \left[\begin{array}{c|cc} 4 & 0 & 0 \\ \hline 0 & 4 & 1 \\ 0 & 1 & 4 \end{array}\right]$$

2. lépés: Ugyanúgy dolgozunk tovább, de most már csak a jobb alsó 2×2-es mátrix részen, a többit változatlanul leírjuk.

(1) **szétbontás:**

$$\left[\begin{array}{c|cc} 4 & 0 & 0 \\ \hline 0 & 4 & 1 \\ 0 & 1 & 4 \end{array}\right] \to \left[\begin{array}{c|cc} 4 & 0 & 0 \\ \hline 0 & 4 & {\color{red}0} \\ 0 & {\color{red}0} & 4 \end{array}\right] \qquad Q = \begin{bmatrix} 0 & -1 & -2 \\ -1 & 0 & {\color{red}-1} \\ -2 & {\color{red}-1} & 0 \end{bmatrix}$$

(2) **Elimináció:** valójában nem kell eliminálni.

$$\left[\begin{array}{c|cc} 4 & 0 & 0 \\ \hline 0 & 4 & 0 \\ 0 & 0 & 4 \end{array}\right] \to \left[\begin{array}{c|c|c} 4 & 0 & 0 \\ \hline 0 & 4 & 0 \\ \cline{2-3} 0 & {\color{red}0} & {\color{red}4} \end{array}\right] = L \text{ és } U \text{ együtt}$$

$\square$

3. Példa:

Készítsük el az

$$A = \begin{bmatrix} 4 & 1 & 2 \\ 1 & 4 & 1 \\ 2 & 1 & 4 \end{bmatrix}$$

mátrix $J = \{(1,2),(1,3),(2,3)\}$ pozícióhalmazhoz illeszkedő ILU-felbontását! A pozícióhalmazt mátrixos alakban is szemléltethetjük, a kinullázandó elemeket $*$-gal jelöljük:

$$\begin{bmatrix} & * & * \\ & & * \\ & & \end{bmatrix}.$$

A felsőháromszögrész minden átlón kívüli elemét megjelöltük.

1. lépés: (1) **szétbontás:** olyan pozíciókat keresünk J-ben, melyek az 1. sorhoz illetve az 1. oszlophoz tartoznak: $(1,2),(1,3)$. Ezeket a pozíciókat kinullázzuk P_1-ben és a (-1)-szeresüket Q_1-be tesszük.

$$A = \widetilde{A}_1 = \begin{bmatrix} 4 & 1 & 2 \\ 1 & 4 & 1 \\ 2 & 1 & 4 \end{bmatrix} = \begin{bmatrix} 4 & 0 & 0 \\ 1 & 4 & 1 \\ 2 & 1 & 4 \end{bmatrix} - \begin{bmatrix} 0 & -1 & -2 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} = P_1 - Q_1$$

(2) **Elimináció:** P_1-en elvégezzük az 1. GE-s lépést:

$$\widetilde{A}_2 = L_1 P_1 = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 1 \\ 0 & 1 & 4 \end{bmatrix}, \quad L_1^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ \frac{1}{4} & 1 & 0 \\ \frac{1}{2} & 0 & 1 \end{bmatrix}.$$

2. lépés: (1) **szétbontás:** olyan pozíciókat keresünk J-ben, melyek a 2. sorhoz illetve az 2. oszlophoz tartoznak: $(2, 3)$.
Ezt a pozíciót kinullázzuk P_2-ben és a (-1)-szeresét Q_2-be tesszük.

$$\widetilde{A}_2 = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 1 \\ 0 & 1 & 4 \end{bmatrix} = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 1 & 4 \end{bmatrix} - \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 0 & 0 \end{bmatrix} = P_2 - Q_2$$

(2) **Elimináció:** P_2-en elvégezzük a 2. GE-s lépést:

$$\widetilde{A}_3 = L_2 P_2 = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{bmatrix}, \quad L_2^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \frac{1}{4} & 1 \end{bmatrix}.$$

A tétel alapján összerakjuk az ILU-felbontást:

$$U = \widetilde{A}_3 = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{bmatrix}, \quad Q = Q_1 + Q_2 = \begin{bmatrix} 0 & -1 & -2 \\ 0 & 0 & -1 \\ 0 & 0 & 0 \end{bmatrix}$$

Mivel az egyes lépésekben a Q_k mátrixok különböző sorait és oszlopait töltjük, így elegedő a gyakorlatban egy Q mátrixot tárolni. Az iterációnál látni fogjuk, hogy Q-ra a végrehajtáshoz nincs szükség.

Összepakolással:

$$L = L_1^{-1} L_2^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ \frac{1}{4} & 1 & 0 \\ \frac{1}{2} & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \frac{1}{4} & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ \frac{1}{4} & 1 & 0 \\ \frac{1}{2} & \frac{1}{4} & 1 \end{bmatrix}.$$

Ellenőrizhetjük, hogy $A = LU - Q$

$$\begin{bmatrix} 4 & 1 & 2 \\ 1 & 4 & 1 \\ 2 & 1 & 4 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ \frac{1}{4} & 1 & 0 \\ \frac{1}{2} & \frac{1}{4} & 1 \end{bmatrix} \cdot \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{bmatrix}}_{\begin{bmatrix} 4 & 0 & 0 \\ 1 & 4 & 0 \\ 2 & 1 & 4 \end{bmatrix}} - \begin{bmatrix} 0 & -1 & -2 \\ 0 & 0 & -1 \\ 0 & 0 & 0 \end{bmatrix}.$$

Teljesíti a ILU-felbontásra tett összes követelményt. $\square$

Tömör írásmódban: Csak egy Q mátrixot tárolunk, ebbe pakoljuk a Q_k mátrixok nem nulla elemeit. Az GE eredményét illetve a GE-s hányadosokat, vagyis az $\widetilde{A}_k, L_k$ mátrixokat is egyben tároljuk. Vonalakkal jelezzük, hogy itt már tárolásról is szó van.

1. lépés: (1) **szétbontás:**

$$A = \widetilde{A}_1 = \begin{bmatrix} 4 & 1 & 2 \\ 1 & 4 & 1 \\ 2 & 1 & 4 \end{bmatrix}, \quad P_1 = \begin{bmatrix} 4 & {\color{red}0} & {\color{red}0} \\ 1 & 4 & 1 \\ 0 & 1 & 4 \end{bmatrix}, \quad Q = \begin{bmatrix} 0 & {\color{red}-1} & {\color{red}-2} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

(2) **Elimináció P_1-en:**

$$\begin{bmatrix} 4 & 0 & 0 \\ 1 & 4 & 1 \\ 2 & 1 & 4 \end{bmatrix} \to \left[\begin{array}{c|cc} 4 & 0 & 0 \\ \hline {\color{red}\frac{1}{4}} & {\color{red}4} & {\color{red}1} \\ {\color{red}\frac{1}{2}} & {\color{red}1} & {\color{red}4} \end{array}\right]$$

2. lépés: Ugyanúgy dolgozunk tovább, de most már csak a jobb alsó 2×2-es mátrix részen, a többit változatlanul leírjuk.

(1) **szétbontás:**

$$\left[\begin{array}{c|cc} 4 & 0 & 0 \\ \hline \frac{1}{4} & 4 & 1 \\ \frac{1}{2} & 1 & 4 \end{array}\right] \rightarrow \left[\begin{array}{c|cc} 4 & 0 & 0 \\ \hline \frac{1}{4} & 4 & \color{red}{0} \\ \frac{1}{2} & 1 & 4 \end{array}\right] \qquad Q = \begin{bmatrix} 0 & -1 & -2 \\ 0 & 0 & \color{red}{-1} \\ 0 & 0 & 0 \end{bmatrix}$$

(2) **Elimináció:**

$$\left[\begin{array}{c|cc} 4 & 0 & 0 \\ \hline \frac{1}{4} & 4 & 0 \\ \frac{1}{2} & 1 & 4 \end{array}\right] \rightarrow \left[\begin{array}{c|cc} 4 & 0 & 0 \\ \hline \frac{1}{4} & 4 & 0 \\ \cline{2-3} \frac{1}{2} & \multicolumn{1}{c|}{\color{red}{\frac{1}{4}}} & \color{red}{4} \end{array}\right] = L \text{ és } U \text{ együtt}$$

$\square$

Tartalomjegyzék

Átalakítás:

$$\begin{aligned} Ax &= b, \quad A = P - Q, \quad P = LU \\ (P-Q)x &= b \\ Px &= Qx + b \\ x &= P^{-1}Qx + P^{-1}b \end{aligned}$$

Ezek alapján az iteráció a következő.

Definíció: ILU-algoritmus

$$x^{(k+1)} = \underbrace{P^{-1}Q}_{B_{ILU}} \cdot x^{(k)} + \underbrace{P^{-1}b}_{c_{ILU}} = B_{ILU} \cdot x^{(k)} + c_{ILU}$$

Írjuk fel az iteráció reziduum vektoros alakját!

$$A = P - Q \quad \Leftrightarrow \quad Q = P - A$$

$$\begin{aligned} P \cdot x^{(k+1)} &= Q \cdot x^{(k)} + b = (P - A) \cdot x^{(k)} + b = \\ &= P \cdot x^{(k)} + \left(-Ax^{(k)} + b\right) = P \cdot x^{(k)} + r^{(k)} \\ \Rightarrow \quad x^{(k+1)} &= x^{(k)} + P^{-1} r^{(k)} \end{aligned}$$

Vezessük be az $s^{(k)} := P^{-1} r^{(k)}$ segédvektort, ezzel egy lépésünk alakja:

$$x^{(k+1)} = x^{(k)} + s^{(k)}.$$

Az új reziduum vektor:

$$r^{(k+1)} = b - Ax^{(k+1)} = b - A(x^{(k)} + s^{(k)}) = r^{(k)} - As^{(k)}.$$

Algoritmus: *ILU*-algoritmus

$$r^{(0)} := b - Ax^{(0)}$$

$k = 1, \ldots,$ leállásig

$$s^{(k)} := P^{-1} r^{(k)} \text{ helyett}$$

$$LU\, s^{(k)} = r^{(k)} \text{ (2 db háromszögű LER mo.)}$$

$$x^{(k+1)} := x^{(k)} + s^{(k)}$$

$$r^{(k+1)} := r^{(k)} - As^{(k)}$$

Megjegyzés:

1. Az átmenetmátrix

$$B_{ILU} = P^{-1}Q = P^{-1}(P - A) = I - P^{-1}A.$$

   Legyen P az A-hoz közeli, mert ekkor $\|B_{ILU}\|$ kicsi és így az iteráció gyors.

2. Ha L, U-ban csak kevés nem nulla átló van, akkor az iteráción belüli LER megoldás műveletigénye kicsi.
3. Láttuk, hogy az iteráció végrehajtásakor Q-ra nincs szükségünk.

Általánosítás az *ILU*-algoritmusból:

$$P\,(x^{(k+1)} - x^{(k)}) = r^{(k)} \quad \Leftrightarrow \quad P\,(x^{(k+1)} - x^{(k)}) + A\,x^{(k)} = b.$$

Definíció: általános kétrétegű iterációs eljárás

A

$$P\,(x^{(k+1)} - x^{(k)}) + A\,x^{(k)} = b$$

iterációt *általános kétrétegű iterációs eljárás*nak nevezzük.
P: a *prekondicionáló mátrix*.

Megjegyzés: A korábbi összes iterációs módszerünk ilyen alakú:

$$P\,(x^{(k+1)} - x^{(k)}) + A\,x^{(k)} = b.$$

1. Ha $P = D$, akkor a $J(1)$ iterációt kapjuk.
2. Ha $P = \frac{1}{\omega}\,D$, akkor a $J(\omega)$ iterációt kapjuk.
3. Ha $P = D + L$, akkor az $S(1)$ iterációt kapjuk.
4. Ha $P = D + \omega\,L$, akkor az $S(\omega)$ iterációt kapjuk.
5. Ha $P = \frac{1}{p}\,I$, akkor az $R(p)$ iterációt kapjuk.
6. Ha $P = LU$ az *ILU*-felbontásból, akkor az *ILU* iterációt kapjuk.

Példa:

A korábbi ILU-felbontás példákhoz készítsük el a megfelelő ILU-algoritmusok átmenetmátrixát és hasonlítsuk össze az egyes iterációk gyorsaságát!

1. Példa:

$$P = \begin{bmatrix} 4 & 0 & 2 \\ 1 & 4 & \frac{1}{2} \\ 2 & 1 & 4 \end{bmatrix}, \quad Q = \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & -\frac{1}{2} \\ 0 & 0 & 0 \end{bmatrix}, \quad B_{ILU} = P^{-1}Q,$$

$$\|B_{ILU}\|_2 \approx 0.3601, \quad \|B_{ILU}\|_\infty \approx 0.3438$$

2. Példa: Jacobi-iteráció

$$P = 4\,I, \quad Q = \begin{bmatrix} 0 & -1 & -2 \\ -1 & 0 & -1 \\ -2 & -1 & 0 \end{bmatrix}, \quad B_{ILU} = P^{-1}Q = \frac{1}{4}\begin{bmatrix} 0 & -1 & -2 \\ -1 & 0 & -1 \\ -2 & -1 & 0 \end{bmatrix},$$

$$\|B_{ILU}\|_2 \approx 0.6830, \quad \|B_{ILU}\|_\infty \approx 0.75$$

3. Példa: Gauss–Seidel-iteráció

$$P = \begin{bmatrix} 4 & 0 & 0 \\ 1 & 4 & 0 \\ 2 & 1 & 4 \end{bmatrix}, \quad Q = \begin{bmatrix} 0 & -1 & -2 \\ 0 & 0 & -1 \\ 0 & 0 & 0 \end{bmatrix}, \quad B_{ILU} = P^{-1}Q,$$

$$\|B_{ILU}\|_2 \approx 0.6408, \quad \|B_{ILU}\|_\infty \approx 0.75$$

Látjuk, hogy az 1. példabeli *ILU*-felbontást alkalmazó *ILU*-algoritmus a leggyorsabb a három közül.

Kerekítési hibák hatása az iterációkra

Tekintsük az iteráció szokásos alakját!

$$x^{(k+1)} = Bx^{(k)} + c$$

Vizsgáljuk meg, hogyan változik az iteráció, ha a $k+1$. lépésben *kicsit* $\varepsilon^{(k)}$*-val* megváltoztatjuk! (Számolási pontatlanság, kerekítési hiba, ...)

1. **Eredeti:**

$$x^{(k+1)} = Bx^{(k)} + c$$

2. **Módosult:**

$$y^{(k+1)} = By^{(k)} + c + \varepsilon^{(k)}$$

Nyilván a lépésenkénti $\varepsilon^{(k)}$ hiba miatt *kicsit* más lesz az iteráció ...

Tétel: a kerekítési hibák hatása az iterációkra

Tegyük fel, hogy

- iterációnk bármely kezdőértékre konvergens,
- a lépésenkénti hiba felülről korlátos, vagyis létezik $\varepsilon > 0$, melyre $\left\|\varepsilon^{(k)}\right\| \leq \varepsilon$ minden k-ra.

Ekkor a $z^{(k)}$ hibasorozatra

$$\lim_{k\to\infty} \left\|z^{(k)}\right\| \leq \frac{\varepsilon}{1 - \|B\|}.$$

Biz.: A $z^{(k)} := x^{(k)} - y^{(k)}$ hibavektorra írjuk fel a rekurziót:

$$z^{(k+1)} = x^{(k+1)} - y^{(k+1)} = \left(Bx^{(k)} + c\right) - \left(By^{(k)} + c + \varepsilon^{(k)}\right) =$$

$$= B\left(x^{(k)} - y^{(k)}\right) - \varepsilon^{(k)} = Bz^{(k)} - \varepsilon^{(k)}.$$

Biz. folyt.: A konvergenciából következik, hogy létezik olyan indukált mátrixnorma, melyben $\|B\| < 1$. A hozzá illeszkedő vektornormában becsüljünk:

$$\begin{aligned}
\left\|z^{(k+1)}\right\| &\leq \|B\| \cdot \left\|z^{(k)}\right\| + \left\|\varepsilon^{(k)}\right\| \leq \|B\| \cdot \left\|z^{(k)}\right\| + \varepsilon \leq \\
&\leq \|B\| \left(\|B\| \cdot \left\|z^{(k-1)}\right\| + \varepsilon\right) + \varepsilon \leq \ldots \leq \\
&\leq \|B\|^{k+1} \cdot \left\|z^{(0)}\right\| + \varepsilon \cdot \left(\|B\|^{k} + \ldots + \|B\| + 1\right) < \\
&< \varepsilon \|B\|^{k+1} + \varepsilon \cdot \frac{1}{1 - \|B\|}.
\end{aligned}$$

Innen $k \to \infty$ határátmenettel adódik a bizonyítandó állítás. $\square$