1. Gépi szám, hibák

Áttekintjük a gépi aritmetika néhány jellegzetességét és szemügyre vesszük a számításokat terhelő hibafajtákat.

1.1. A gépi számok

A gépi számok leggyakrabban 2-es alapú (vagy bináris), előjeles normalizált számok, így elsősorban ezekkel fogunk foglalkozni. Alakjuk

$$\underset{\text{előjel, } t \text{ db bináris jegy}}{\pm .101\ldots01} \cdot 2^{\overset{\nwarrow \text{ kitevő}}{k}} = \pm m \cdot 2^k \tag{1.1}$$

A nemzérus mantissza mindig 1-gyel kezdődik, emiatt $0.5 \le m < 1,\ m \neq 0$. Ha az alap nem 2, akkor a 10-es és a 16-os (hexadecimális) számok fordulnak még elő a gyakorlatban.

A bináris gépi számok halmazát jelölje $M(t,k^-,k^+)$, ahol t a mantisszahossz, k^- a legkisebb kitevő, k^+ pedig a legnagyobb kitevő. Az általunk használt PC-kben, - személyi számítógépekben a szimplapontos szám 4 *bájt* = 32 *bit* területet foglal el a memóriában és az egyes funkciók kiosztása a következő:

1	8	23

1 bit jut az előjelre, 8 bit a kitevőre és 23 a mantisszára. Ezen számok pontossága kb. 7 decimális jegynek felel meg ($23\log_{10} 2 \approx 6.923$, azaz kb. 0.3-del szorzandó a bitek száma) és a nagyságrend 10^{-38}-tól 10^{38}-ig terjedhet. A duplapontos (kétszeres pontosságú) számok 64 biten helyezkednek el:

1	11	52

előjel: 1 bit, kitevő 11 bit és a mantisszahossz: 52 bit. Most a pontosság kb. 15 decimális jegy, és az ábrázolható számok nagyságrendje 10^{-307}-től 10^{307}-ig terjed. Egyes programnyelvek megengedik a négyszeres pontosságú számokat is.

A konkrét megvalósításban kihasználható, hogy a nemzérus mantissza első bitje mindig 1, emiatt elhagyható. Ezzel a fogással még plusz 1 bithez lehet jutni, aminek jelentősége az aritmetika tulajdonságainak javításában van. Ekkor viszont meg kell tudni különböztetni a zérust 0.5-től. Erre többféle lehetőség van, hiszen zérus mantissza mellett a kitevő bitjei extra információt hordozhatnak. Az igen nagy abszolút értékű, a gépi számokkal nem ábrázolható számok jelölésére is ki lehet alakítani egy bit-kombinációt. A már nem ábrázolható nagy számokra a ∞ jelet fogjuk használni. Szokás még az NaN jelölés: „not-a-number": *nem szám*, értsd: nem gépi szám. Egyes programnyelvekben ezt kapjuk eredményül, ha zérussal próbálunk osztani. Ha NaN-nel ezután bármilyen aritmetikai műveletet végzünk, az eredmény NaN, mégha zérussal szoroztunk, akkor is.

1.2. Nevezetes gépi számok

A legkisebb pozitív mantissza: ½. A legnagyobb mantissza: $.\overbrace{11\ldots1}^{t \text{ db 1-es}} = 1 - 2^{-t}$. $M(t,k^-,k^+)$-ban a legkisebb pozitív szám: $\varepsilon_0 = .10\ldots0 \cdot 2^{k^-} = 1/2 \cdot 2^{k^-}$.

A másik nevezetes szám ε_1, az a legkisebb pozitív szám, amelyet 1-hez hozzáadva 1-nél nagyobb gépi számot kapunk: $1 + \varepsilon_1 = .10\ldots01 \cdot 2^{+1}$, innen $\varepsilon_1 = 2^{-t+1}$. A legnagyobb ábrázolható szám: $M_\infty = (.11\ldots1 \cdot 2^{k^+}) = (1 - 2^{-t})2^{k^+}$. A legkisebb szám ennek a negatívja.

Például legyen a gépi számok halmaza $M(5,-4,3)$. Ekkor a legnagyobb mantissza: $.11111 = 1-2^{-5}$, a legkisebb mantissza ½. Az első pozitív gépi szám: $\varepsilon_0 = 1/2 \cdot 2^{-4} = 2^{-5}$. Az 1 után következő első gépi szám távolsága 1-től: $\varepsilon_1 = 2^{-t+1} = 2^{-4}$. A legnagyobb ábrázolható szám: $M_\infty = (1-2^{-t}) \cdot 2^{k^+} =$ $= (1-2^{-5})2^3 = 8-1/4$.

1.3. Valós számok konverziója gépi számmá

A következő kérdés: a valós számokat hogyan alakítsuk át gépi számokká. Az ezt megvalósító input függvényt fl-lel jelöljük (a *floating point number* kifejezés kezdőbetűi), $\mathrm{fl}: \mathbb{R} \to M$. Megadása a következő:

$$\mathrm{fl}(x) = \begin{cases} \infty, & \text{ha} \quad |x| > M_\infty \\ 0, & \text{ha} \quad |x| < \varepsilon_0 \\ x\text{-hez legközelebbi gépi szám, ha } \varepsilon_0 \le |x| \le M_\infty \end{cases}, \tag{1.2}$$

ahol az x-hez legközelebbi gépi szám a kerekítés szabályai szerint értendő.

Például alakítsuk át 10.87-et 8-jegyű bináris számmá. Ezt célszerűen úgy tesszük, hogy az egész részt 2-vel osztjuk, és jegyezzük a maradékokat. A sorrendet megfordítva kapjuk a bináris jegyeket. A tört részt 2-vel szorozzuk. A kijövő egész részt nem szorozzuk tovább, hanem bináris jegyként megőrízzük. Az utolsó jegyet már abból meg tudjuk állapítani, hogy a tört rész kisebb-e 0.5-nél. Ha kisebb, az adódó jegy 0, egyébként 1.

$$\begin{array}{r|l} 10 & 0 \\ 5 & 1 \\ 2 & 0 \\ 1 & 1 \end{array} \to 10_2 = 1010 \qquad \begin{array}{r|l} . & 87 \\ 1 & 74 \\ 1 & 48 \\ 0 & 96 \end{array} \to 0.87_2 = .1101\ldots$$

Kaptuk: $10.87_2 = 1010.1101\ldots$. Ez nem kerekítéssel, hanem csonkítással kapott eredmény. A kerekített szám megállapításához még egy jegyet meg kell határozni. Ha a következő jegy 1, akkor az utolsó bináris jegyhez 1-et adunk, egyébként változatlanul hagyjuk. Esetünkben a következő (kilencedik) jegy 1, így a kerekített érték: 1010.1110. Ha 10.87-et az előbbi példában szereplő $M(5,-4,3)$ halmazra kívánjuk leképezni az fl függvénnyel, akkor $\mathrm{fl}(11.87) = \infty$, mert $M_\infty < 10.87$.

1.1 Gyakorlat. Legyen a gépi számok halmaza $M(5,-4,4)$. Határozzuk meg a nevezetes számait! Mi lesz a következő számok leképezése a halmazba: 1/50, 0.37, 3.67, 7.2, 21.78?

1.2 Gyakorlat. Hogyan konvertálnánk 10.87-et 3-as alapú számrendszerbe?

Feltesszük, hogy x-et pontosan ismerjük. Ekkor $\mathrm{fl}(x)$ hibája a következőképp becsülhető:

$$|x - \mathrm{fl}(x)| \le \begin{cases} \infty, & \text{ha} \quad |x| > M_\infty \\ \varepsilon_0, & \text{ha} \quad |x| < \varepsilon_0 \\ \varepsilon_M |x|, & \text{ha} \quad \varepsilon_0 \le |x| \le M_\infty \end{cases}, \tag{1.3}$$

ahol $\varepsilon_M = \varepsilon_1 / 2 = 2^{-t}$ a *gépi epszilon*, ez adja az ε_0 és M_∞ közé eső szám ábrázolásának relatív hibáját. Itt az első sornak csak jelzés értéke van. A második sor önmagáért beszél, egyedül a harmadik sor kíván némi magyarázatot. Azt fejezi ki, hogy az ábrázolt szám hibája nem nagyobb, mint a t-edik bináris jegyben elkövetett hiba. A harmadik sor átrendezése a relatív hiba korlátját adja:

$$\frac{|x - \mathrm{fl}(x)|}{|x|} \le \varepsilon_M . \tag{1.4}$$

A relatív hiba megállapításakor elég a mantissza hibáját tekinteni, mert a kitevő osztáskor kiesik. A kerekítéskor a mantisszában elkövetett hiba legfeljebb 2^{-t-1}. A relatív hibájának felső korlátját úgy kapjuk, hogy a lehetséges legkisebb pozitív mantissza-értékkel osztunk: ½-vel. Így kapjuk eredményül $\varepsilon_M = 2^{-t}$ -t.

1.3 Gyakorlat. Hogyan módosulna a gépi epszilon, ha a kerekítés helyett csonkítást alkalmaznánk?

1.4. A gépi aritmetika

Vannak gépi számaink, a következő kérdés, hogy milyen tulajdonságú lesz a lebegőpontos számokkal megvalósított gépi aritmetika. A következő számpéldákban a tízes alapú számrendszert fogjuk használni, ahol van négy decimális jegyünk és a kitevő előjeles kétjegyű szám lehet. Ezen gépi számok halmazát egyszerűen M -mel fogjuk jelölni. Jelölés: $0.2543 \cdot 10^2 = 0.2543\ +02$

A gépi artimetikában nem lesz igaz minden, amit a valós számtestben megszoktunk. Az alábbiakban felsorolunk ilyen eltéréseket:

- Létezhet nemzérus $a,b \in M$, amelyre $a+b=a$. Ez a számok eltérő nagyságrendje miatt lehetséges. Például adjuk össze a következő számokat: 0.3460 +02 és 0.4524 –03:

$$\begin{array}{r} 0.3460\ +02 \\ \underline{0.000004524\ +02} \\ 0.3460\ +02 \end{array}$$

- Létezhet $a,b,c \in M$, amelyre $(a+b)+c \neq a+(b+c)$. Például

$$\begin{array}{r} 0.3460\ +02 \\ \underline{0.00004524\ +02} \\ 0.3460\ +02 \end{array} \qquad \begin{array}{r} 0.3460\ +02 \\ \underline{0.00003872\ +02} \\ 0.3460\ +02 \end{array}$$

de először a két kicsi számot összeadva

$$\begin{array}{r} 0.3872\ -02 \\ \underline{0.4524\ -02} \\ 0.8386\ -02 \end{array} \qquad \begin{array}{r} 0.3460\ +02 \\ \underline{0.00008386\ +02} \\ 0.3461\ +02 \end{array}$$

Ez arra int, hogyha sok számot összegzünk, akkor az abszolút érték szerinti kicsikkel érdemes kezdeni.

- Létezhet $a,b,c \in M$, amelyre $(ab)c \neq a(bc)$. Például

$$(0.1245\ +62 \cdot 0.4314\ -58) \cdot 0.4362\ -54 = .5371\ +03 \cdot 0.4362\ -54 = .2343\ -51,$$

míg a másik zárójelezés szerint a második és harmadik szám szorzata kisebb, mint a legkisebb ábrázolható gépi szám, így ez a szorzat zérus, ami a teljes szorzatra zérus eredményt ad. Így, ha sok számot kell összeszoroznunk, még nagyobb gondossággal kell eljárnunk, mert könnyen kerülhetünk abba a helyzetbe, hogy az eredmény, vagy valamely rész-szorzata kívül esik a számábrázolás tartományán. Ha az eredmény túl nagy, vagy túl kicsi, akkor egy lehetőség a gondok csökkentésére az eredmény logaritmusát számolni.

- Összevonás után az eredmény relatív hibája jelentősen megnőhet. Például

$$\begin{array}{r} 0.4693\ +02 \\ \underline{-0.4682\ +02} \\ 0.0011+02 \end{array}$$

ami egyenlő 0.1100 +00-val. Látjuk, itt már csak az első két jegy pontos. Ezt jelenséget *kivonási jegyveszteségnek* nevezzük. Néha adhatók fogások a kivonási jegyveszteség elkerülésére vagy

csökkentésére, pl. ha $\sqrt{3472}-\sqrt{3471}$-et így számítjuk, kihasználva, hogy a gyök alatt egész számok vannak:

$$\frac{(\sqrt{3472}-\sqrt{3471})(\sqrt{3472}+\sqrt{3471})}{\sqrt{3472}+\sqrt{3471}}=\frac{1}{\sqrt{3472}+\sqrt{3471}}.$$

A *másodfokú egyenlet gyökeit* pedig az alábbi módon célszerű számítani:

$$x^2-2px+q=0 \text{ gyökei: } x_1=p+\text{sign}(p)\sqrt{p^2-q},\quad x_2=q/x_1.$$

- Előfordulhat olyan eset, amikor a közbülső eredmény túlcsordul (nagyobb mint M_∞), emiatt rossz a program futása, pedig a végeredmény az ábrázolható számok közt van. Például legyen $a=0.3265+60$, $b=0.5671+02$ és számítandó $\sqrt{a^2+b^2}$. Az első szám kitevője négyzetre emeléskor 120, így túlcsordult számot kapunk. Ha viszont $s\sqrt{(a/s)^2+(b/s)^2}$-et számítjuk, ahol $s=\max(|a|,|b|)$, akkor ez nem következik be.

- Néha arra is számítani kell, hogy egy függvény nem adja olyan pontossággal vissza a helyettesítési értéket, mint amilyen pontossággal indultunk. Például tekintsük a sin függvényt. Ha az argumentum kicsi, akkor nincs semmi baj. Ha azonban x értéke nagy, például $x=2356$, akkor $\sin(2356)$ számításakor 2356 π-vel vett osztási maradékát kell vennünk. A maradékban már csak 1 jegy lesz pontos ha a fenti aritmetikát használjuk, így az eredménynél sem remélhetünk nagyobb pontosságot.

A mutatott példák alapján megállapíthatjuk, hogy a gépi aritmetika nemkívánatos jelenségei elsősorban akkor következnek be, ha a számok között túl nagy a nagyságrendi különbség, vagy egymáshoz nagyon közeli számokat vonunk ki egymásból.

1.5. Hibák

Az igényes számításoknál arra is kíváncsiak vagyunk, hogy az eredményt milyen pontosan tudtuk előállítani. Ehhez számba kell venni a lehetséges hibafajtákat. Az első a kiindulásul használt adatok *öröklött hibája*, nevezhetjük ezt *adathibának* is. Lehet, hogy a számítás során magunk is *tévedünk*, ezt gondos ellenőrzéssel magunknak kell felfedeznünk és kijavítanunk. A *képlethiba* az alkalmazott módszerhez tartozik. A *kerekítési hibák* részben bekövetkezhetnek a kézi számítás, adatelőkészítés során, de a gépi aritmetikának is mindig van ilyen hibája. A hibaelemzés során fel kell ismernünk, melyik az a hibafajta, ami az adott feladat szempontjából lényeges. Sok olyan számítás van, amikor az adathiba, vagy a képlethiba jelenti a fő hibaforrást. Az adathibát sokszor csak tudomásul vehetjük, de a képlethibát esetleg csökkenthetjük pontosabb módszer alkalmazásával.

A hibaszámítás alapmodellje szerint a közelítő értékekkel kapott pontos számítás eredményét közelítésnek tekintjük és azt vizsgáljuk, mekkora a hibája.

Jelölések. Az x mennyiség *pontos értéke* x^*, hibája: $\Delta x=x-x^*$, ahol Δx előjeles szám. A relatív hiba $\delta x=\Delta x/x\approx\Delta x/x^*$. Itt megjegyezzük, hogy egyes szerzők a relatív hibát a pontos értékkel definiálják, tehát az itt látható második formulát használják. A mi választásunk tudomásul veszi, hogy a pontos értéket nem ismerjük. A *hibakorlát* Δ_x egy nemnegatív szám, amellyel felülről becsüljük a hiba abszolút értékét: $|\Delta x|\le\Delta_x$. Hasonlóképp δ_x a *relatív hibakorlát*, amelyre $|\delta x|\le\delta_x$.

1.4 Gyakorlat. Mutassuk meg, hogy a relatív hiba kétféle megadása között a különbség másodrendű: $\Delta x/x^*-\delta x=\delta x/(1-\delta x)-\delta x=(\delta x)^2/(1-\delta x)$.

A valóságban a Δx hibát nem ismerjük, csak annak felső korlátját. Emiatt kiindulásul annyit tudunk, hogy x^* az x érték valamely Δ_x-sugarú környezetében van.

A hibanalízis szempontjából fontosak az alapműveletek, $+,-,*,/$ hibái. Alább a baloldali összefüggések a hibákra, a jobboldaliak pedig a hibakorlátokra vonatkoznak:

$$\begin{aligned} &\Delta(x \pm y) = \Delta x \pm \Delta y, && \Delta_{x\pm y} = \Delta_x + \Delta_y, \\ &\Delta(xy) = x\Delta y + y\Delta x, && \Delta_{xy} = |x|\Delta_y + |y|\Delta_x, \\ &\Delta(x/y) = \frac{y\Delta x - x\Delta y}{y^2}, && \Delta_{x/y} = \frac{|y|\Delta_x + |x|\Delta_y}{|y^2|}. \end{aligned} \tag{1.5}$$

A hibaformulák hasonló módon származtathatók, mint az összeg-, szorzat-, és hányadosfüggvények differenciálási szabályai. Innen az is látható, hogy a formulák csak akkor tekinthetők jóknak, ha a hibák valóban kicsik, és a másodrendű hibatagok elhanyagolhatók. A jobboldali formulák a baloldaliakból következnek, akárcsak az alábbi, relatív hibákra vonatkozó kifejezések:

$$\begin{aligned} &\delta(x \pm y) = \frac{x\delta x \pm y\delta y}{x \pm y}, && \delta_{x\pm y} = \frac{|x|\delta_x + |y|\delta_y}{|x \pm y|}, \\ &\delta(xy) = \delta y + \delta x, && \delta_{xy} = \delta_y + \delta_x, \\ &\delta(x/y) = \delta x - \delta y, && \delta_{x/y} = \delta_x + \delta_y. \end{aligned} \tag{1.6}$$

A függvényértékek hibája. Legyen $f : \mathbb{R} \to \mathbb{R}$ legalább kétszer folytonosan differenciálható függvény. Ekkor a Lagrange középérték-tétel szerint létezik $\xi \in [x, x^*]$, amelyre

$$f(x) = f(x^*) + f'(x^*)\Delta x + f''(\xi)(\Delta x)^2/2 .$$

Innen a másodrendű kicsiny utolsó tag elhagyásával a *függvényérték hibája*:

$$f(x) - f(x^*) = \Delta f \approx f'(x^*)\Delta x . \tag{1.7}$$

Legyen $\max_{x\in[x-\Delta_x, x+\Delta_x]} |f'(x)| = M_1$, ezzel $\Delta_f = M_1\Delta_x$, ahol vegyük tekintetbe, hogy a becslés x egy Δ_x sugarú környezetére vonatkozik. A relatív hibára kapjuk:

$$\delta f = \frac{\Delta f}{f(x)} \approx \frac{xf'(x)}{f(x)}\frac{\Delta x}{x} = \frac{xf'(x)}{f(x)}\delta x.$$

Az abszolút értékekre áttérve:

$$|\delta f| \approx c(f, x)|\delta x|, \tag{1.8}$$

ahol a $c(f, x) = |xf'(x)/f(x)|$ számot az f függvény x *pontbeli kondiciószámának* nevezzük. Ha ez a szám nagy, akkor a függvényt *instabilnak*, vagy *gyengén meghatározottnak* nevezzük, mert az argumentum kicsiny megváltozása nagy függvényérték-megváltozást eredményez. Túl nagy kondició-szám mellett a gépi számok kerekítési hibái is elviselhetetlenül nagy végső hibához vezetnek.

Az (1.7) és (1.8) összefüggések sugallják a következő stabilitás fogalmat: egy algoritmus *stabil,* ha két bemenő érték: x_1, x_2 és a hozzájuk tartozó kimenő értékek, f_1, f_2 között fennáll egy

$$|f_2 - f_1| \le C|x_1 - x_2|, \;\; x_1, x_2 \in X \tag{1.9}$$

típusú összefüggés, ahol C az algoritmus adataitól független nem túlságosan nagy állandó. Vegyük észre, itt x_i, f_i gépi számok, egy véges halmaz elemei.

Fontos még az *inverz stabilitás* fogalma. Egy leképezés inverz stabil, ha az eredmény egy kissé perturbált kezdetiértékből pontos számítással megkapható.

2. Normák, egyenlőtlenségek

Ebben a szakaszban vektorok és mátrixok között távolságfüggvényeket fogunk bevezetni.

1.1. Metrikus tér

Legyen $\mathcal{X}$ egy halmaz, amelynek elemei közt bevezetünk egy távolságfüggvényt $\delta:(\mathcal{X}\times\mathcal{X})\to\mathbb{R}$. Azt kívánjuk, $a,b\in\mathcal{X}$ -re rendelkezzen a következő tulajdonságokkal:

i) $\delta(a,b)=\delta(b,a)$, azaz a legyen olyan távolságra b-től, mint b a-tól (szimmetria).

ii) $\delta(a,b)=0 \Leftrightarrow a=b$, a távolság csak akkor legyen zérus, ha a két elem azonos.

iii) $\delta(a,c)\le\delta(a,b)+\delta(b,c)$, a háromszög-egyenlőtlenség. Azt fejezi ki, hogy két pont között legrövidebb út az egyenes.

Ekkor a $(\delta,\mathcal{X})$ párt *metrikus térnek* nevezzük. A következőkben $\mathcal{X}$ gyanánt az $\mathbb{R}^n$ és $\mathbb{R}^{m\times n}$ halmazok kerülnek szóba, azaz vektorok és mátrixok között fogunk távolságfüggvényeket készíteni. Ez a δ nem lehet negatív értékű, mert $0=\delta(a,a)\le\delta(a,b)+\delta(b,a)=2\delta(a,b)$ következmény.

2.1. A vektorok hatványnormája

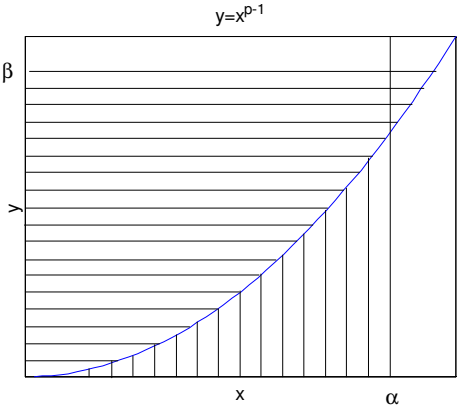
A vektor normája $\|x\|:\ \mathbb{R}^n\to\mathbb{R}$ a következő tulajdonságokkal rendelkezik:

$$\begin{aligned} &i) && \|x\|=0 \ \Leftrightarrow\ x=0,\\ &ii) && \|\lambda x\|=|\lambda|\|x\|,\\ &iii) && \|x+y\|\le\|x\|+\|y\|. \end{aligned} \tag{2.1}$$

Ekkor a $\delta(x,y)=\|x-y\|$ választás metrikát ad, mert a kívánt tulajdonságok teljesülnek. Az első két feltételt triviálisan kielégíti a hatványnorma:

$$\|x\|_p=\left(\sum_{i=1}^{n}|x_i|^p\right)^{1/p},\qquad 1\le p\le\infty, \tag{2.2}$$

a harmadikat később fogjuk belátni.



2.2. A Hölder-egyenlőtlenség

A hatványnormákra fennáll a Hölder-egyenlőtlenség:

$$\left|y^Tx\right|\le\sum_{i=1}^{n}|x_i||y_i|\le\|x\|_p\,\|y\|_q,\qquad \frac{1}{p}+\frac{1}{q}=1, \tag{2.3}$$

ami $p=q=2$ -re a jólismert Cauchy-Bunyakovszkij egyenlőtlenségbe megy át. A p és q közötti összefüggés átrendezhető a $p-1=1/(q-1)$ alakba, amit szem előtt tartva könnyen belátható az alábbi egyenlőtlenség. Az alkalmazott függvény $y=x^{p-1}$, az első integrál a függőlegesen, a második a vízszintesen satírozott területet jelenti:

$$\alpha\beta \le \int_0^{\alpha} x^{p-1}dx + \int_0^{\beta} y^{q-1}dy = \frac{\alpha^p}{p} + \frac{\beta^q}{q}$$

Ezután az

$$\alpha_i = \frac{|x_i|}{\|x\|_p}, \qquad \beta_i = \frac{|y_i|}{\|y\|_q}$$

helyettesítéssel és az i szerinti összegzés elvégézésével kapjuk (2.3) jobb oldali összefüggését.

(2.1) harmadik összefüggése, a háromszög-egyenlőtlenség úgy látható be, hogy $p/q = p-1$ szem előtt tartása mellett a

$$\|x+y\|_p^p = \sum_{i=1}^{n} |x_i + y_i|^p \le \sum_{i=1}^{n} \{|x_i| + |y_i|\} |x_i + y_i|^{p-1}$$

egyenlőtlenség jobb oldalának mindkét tagjára alkalmazzuk a Hölder-egyenlőtlenséget. Ekkor az első tagra a következő eredmény adódik:

$$\sum_{i=1}^{n} |x_i| |x_i + y_i|^{p-1} \le \|x\|_p \left\{ \sum_{i=1}^{n} |x_i + y_i|^{(p-1)q} \right\}^{1/q} = \|x\|_p \, \|x+y\|_p^{p/q},$$

és a másik taggal is hasonló eredményre jutunk, a kettőt együtt rendezve kapjuk a kívánt egyenlőtlenséget, amit általánosan a p index mellett a *Minkowski-egyenlőtlenségnek* nevezünk.

2.3. A hatványnormák néhány tulajdonsága

A hatványnormákra teljesül:

$$\|x\|_{p+s} \le \|x\|_p, \quad 1 \le p, \quad 0 \le s, \tag{2.4}$$

hiszen ez az összefüggés átrendezhető a

$$\sum_{i=1}^{n} \left| \frac{x_i}{x_k} \right|^{p+s} \le \left\{ \sum_{i=1}^{n} \left| \frac{x_i}{x_k} \right|^{p} \right\} \left\{ \sum_{i=1}^{n} \left| \frac{x_i}{x_k} \right|^{p} \right\}^{s/p}, \quad x_k \neq 0$$

alakba. Ha itt $|x_k| = \max_i |x_i|$ akkor a jobb oldal első tényezője tagról tagra nagyobb vagy egyenlő a bal oldalnál, a második tényező viszont biztosan nem kisebb 1-nél.

A fontosabb hatványnormák a következők:

$$\|x\|_1 = \sum_{i=1}^{n} |x_i| .$$

Ez az 1-es vagy oktaéder norma, mivel a 3-dimenziós térben az azonos normájú vektorok egy olyan oktaéderen helyezkednek el, amelynek csúcspontjai az $\|x\|_1 \{(\pm 1,0,0),(0,\pm 1,0),(0,0,\pm 1)\}$ vektorok.

$$\|x\|_2 = \left\{ \sum_{i=1}^{n} |x_i|^2 \right\}^{1/2}$$

az x vektor euklidészi, kettes vagy gömbnormája.

A $p \to \infty$ határesetben adódik

$$\|x\|_\infty = \max_j |x_j| \cdot \lim_{p\to\infty} \left\{ \sum_{i=1}^{n} \left| \frac{x_i}{\max_j |x_j|} \right|^p \right\}^{1/p} = \max_j |x_j|$$

a Csebisev-, ∞ -, vagy kocka-norma. Mint látjuk, (2.4) alapján itt a legnagyobb és legkisebb hatvány-normák szerepelnek, továbbá az ortogonális transzformációkkal szemben invariáns 2-es norma. Ezekre a normákra a definíciók alapján levezethetők a következő egyenlőtlenségek:

$$\begin{aligned} &\|x\|_\infty \le \|x\|_1 \le n\|x\|_\infty, \\ &\|x\|_\infty \le \|x\|_2 \le \sqrt{n}\|x\|_\infty, \\ &\frac{1}{\sqrt{n}}\|x\|_1 \le \|x\|_2 \le \|x\|_1. \end{aligned} \tag{2.5}$$

2.4. Konvergencia normában. A normák ekvivalenciája

A norma alkalmas arra, hogy segítségével egy vektorsorozat konvergenciáját értelmezzük. Ezek alapján $x^{(k)} \to x$ alatt azt értjük, hogy $\exists x \in \mathbb{R}^n$, $\lim_{k\to\infty} \|x^{(k)} - x\| = 0$.

Az $\|x\|_{(1)}$ és $\|x\|_{(2)}$ normákat *ekvivalensnek* nevezzük, ha $\exists c_1, c_2 > 0$ úgy, hogy

$$c_1\|x\|_{(1)} \le \|x\|_{(2)} \le c_2\|x\|_{(1)}.$$

6.5.1 Tétel (bizonyítás nélkül): Végesdimenziós vektortérben bármely két norma ekvivalens. Ez azt jelenti, hogy a normák akármennyire nem különbözhetnek egymástól. Így mindegy, milyen normában vizsgáljuk a konvergenciát.

2.5. Mátrixnormák

A mátrix normája $\|A\|$: $\mathbb{R}^{m\times n} \to \mathbb{R}$ a következő tulajdonságokkal rendelkezik:

$$\begin{aligned} &i) && \|A\| = 0 \iff A = 0, \\ &ii) && \|\lambda A\| = |\lambda|\|A\|, \\ &iii) && \|A + B\| \le \|A\| + \|B\|, \\ &iv) && \|AB\| \le \|A\|\|B\|. \end{aligned} \tag{2.6}$$

Az utolsó két tulajdonságot akkor követeljük meg, ha a két mátrix összeadható vagy összeszorozható. Mivel a vektorok speciális mátrixoknak tekinthetők, minden mátrixnorma meghatároz egy vektor-normát, amelyet a mátrixnormával *kompatibilis* vektornormának nevezünk. Ez az út fordítva is bejárható, ugyanis minden vektornorma *indukál egy mátrixnormát* a következőképpen:

$$\|A\| = \sup_{x\neq 0} \frac{\|Ax\|}{\|x\|} = \sup_{\|x\|=1} \|Ax\|, \tag{2.7}$$

ahol $\|.\|$ vektornorma. Csak megjegyezzük, az általánosabb definicióban megengedhető, hogy más normák szerepeljenek a számlálóban és a nevezőben. A (2.7) definíció egyenes következménye

$$\|Ax\| \le \|A\|\|x\|. \tag{2.8}$$

2.6. Tétel

Az indukált mátrixnorma eleget tesz a (2.6) feltételeknek.

Bizonyítás. Ad 1. $A = 0 \to \|A\| = 0$. $\|A\| = 0 \to Ax = 0 \ \forall x - \text{re} \to A = 0$.

Ad 2. $\|\lambda A\| = \sup_{\|x\|=1} \|\lambda A x\| = |\lambda| \sup_{\|x\|=1} \|Ax\| = |\lambda| \|A\|$.

Ad 3. $\|A+B\| = \sup_{\|x\|=1} \|(A+B)x\| \le \sup_{\|x\|=1} \{\|Ax\| + \|Bx\|\} \le \|A\| + \|B\|$.

Ad 4. $\exists x_0 \in \mathbb{R}^n$, $\|x_0\| = 1$: $\|AB\| = \|ABx_0\| \le \|A\| \|Bx_0\| \le \|A\| \|B\|$. ■

2.7. Az indukált mátrixnormák meghatározása

$p = 1$, *oszlopnorma:*

$$\|A\|_1 = \max_{(j)} \|Ae_j\|_1 = \max_{(j)} \sum_{i=1}^{m} |a_{ij}|. \tag{2.9}$$

Legyen $\|x\|_1 = 1$, ekkor $\|Ax\|_1 = \sum_{i=1}^{m} \left| \sum_{j=1}^{n} a_{ij} x_j \right| \le \sum_{i=1}^{m} \sum_{j=1}^{n} |a_{ij}| |x_j| = \sum_{j=1}^{n} |x_j| \sum_{i=1}^{m} |a_{ij}| \le$

$\le \left(\sum_{j=1}^{n} |x_j| \right) \max_{(j)} \sum_{i=1}^{m} |a_{ij}| = \max_{(j)} \|Ae_j\|_1$. Ezt a felső korlátot valamely e_k -ra el is éri, így a maximumot találtuk meg.

$p = \infty$, *sornorma*:

$$\|A\|_\infty = \max_{(i)} \|e_i^T A\|_\infty = \max_{(i)} \|A^T e_i\|_1 = \max_{(i)} \sum_{j=1}^{n} |a_{ij}|. \tag{2.10}$$

Legyen $\|x\|_\infty = 1$, ekkor $\|Ax\|_\infty = \max_{(i)} \left| \sum_{j=1}^{n} a_{ij} x_j \right| \le \max_{(i)} \sum_{j=1}^{n} |a_{ij}| |x_j| \le \max_{(i)} \sum_{j=1}^{n} |a_{ij}|$. Tegyük fel, a maximum a k -adik sorra következett be. Ekkor $\|x\|_\infty = 1$ és $\|Ax\|_\infty$ éppen a megállapított felső korlát az $x = \left[x_j \right] = \left[\overline{a}_{kj} / |a_{kj}| \right]$ vektorral, ahol a felülvonás komplex konjugáltat jelöl.

$p = 2$, *spektrál norma:*

$$\|A\|_2 = \max_{(k)} \left(\lambda_k (A^T A) \right)^{1/2}. \tag{2.11}$$

Ekkor a következő maximumot keressük:

$$\|A\|_2^2 = \max \frac{\|Ax\|_2^2}{\|x\|_2^2} = \max \frac{x^T A^T A x}{x^T x}.$$

Az itt látható hányados az $A^T A$ mátrixra vonatkozó *Rayleigh-hányados*. Ha $A^T A$ egy sajátvektora u_k λ_k sajátértékkel, akkor $x = u_k$ választással a Rayleigh-hányados értéke éppen λ_k lesz. Innen világos, a Rayleigh hányados legnagyobb értéke legalább $\lambda_{\max} = \max_k \lambda_k$. Megmutatjuk, nagyobb értéke nem lehet. Tudjuk, a szimmetrikus mátrix sajátvektorai teljes ortonormált rendszert alkotnak, így bármely x vektor kifejthető $x = \sum_{j=1}^{n} \alpha_j u_j$ alakban. Ezt helyettesítve a Rayleigh-hányadosba, a különbségre kapjuk:

$$\lambda_{\max} - \frac{x^T A^T A x}{x^T x} = \lambda_{\max} - \frac{\sum_{j=1}^{n} \lambda_j \alpha_j^2}{\sum_{j=1}^{n} \alpha_j^2} = \frac{\sum_{j=1}^{n} (\lambda_{\max} - \lambda_j) \alpha_j^2}{\sum_{j=1}^{n} \alpha_j^2} \ge 0,$$

ami mutatja, hogy a maximumot találtuk meg.

2.8. A spektrálsugár és az indukált normák összefüggése

Egy mátrix *spektrál sugara* alatt a következőt értjük:

$$\rho(A) = \max_k |\lambda_k(A)|, \tag{2.12}$$

ahol $\lambda_k(A)$ az A mátrix sajátértéke. Az $\|A\|$ mátrixnorma és az $\|x\|$ vektornorma *illeszkedő,* ha bármely x-re eleget tesznek a (2.8) összefüggésnek. Ez utóbbi definíció arra az esetre szól, amikor a vektornorma nem kompatibilis, vagy a mátrixnorma nem a vektornormából indukált, mert különben az illeszkedés triviális. Igaz az összefüggés:

$$\rho(A) \le \|A\|, \tag{2.13}$$

ahol $\|A\|$ tetszőleges norma, mert $Au_k = \lambda_k u_k$, $\| u_k \| = 1$ mellett a vektorokra is ugyanazt a mátrixnormát alkalmazva

$$|\lambda_k| \|u_k\| = |\lambda_k| = \|Au_k\| \le \|A\| \|u_k\| = \|A\|, \quad \forall k\text{-ra.}$$

A (2.13) reláció akkor is igaz, ha olyan mátrixnormánk van, ami csak négyzetes mátrixokra van definiálva. Ekkor az $Au_k u_k^T = \lambda_k u_k u_k^T$ kifejezést képezve lehet a bizonyítást megismételni.

2.9. A lineáris egyenletrendszer megoldásának perturbációi

Két esetet fogunk megvizsgálni. Az egyik, amikor az egyenletrendszer b jobboldalát perturbáljuk egy kis δb vektorral, a másik, amikor az együtthatómátrix perturbációját vizsgáljuk.

Az első esetben $A(x + \delta x) = b + \delta b$-ből következik $A\delta x = \delta b$ és illeszkedő normák esetén kapjuk a becslést:

$$\frac{1}{\|A\|\|A^{-1}\|} \frac{\|\delta b\|}{\|b\|} \le \frac{\|\delta x\|}{\|x\|} \le \|A\|\|A^{-1}\| \frac{\|\delta b\|}{\|b\|}. \tag{2.14}$$

Az eredeti és a perturbált értékekre vonatkozó egyenletekből

$$\begin{array}{ll} b = Ax, & \delta x = A^{-1}\delta b, \\ \downarrow & \downarrow \\ \|b\| \le \|A\|\|x\|, & \|\delta x\| \le \|A^{-1}\|\|\delta b\|. \end{array}$$

A kapott egyenlőtlenségek azonos oldalait összeszorozva kapjuk (2.14) jobboldali összefüggését. A baloldalit ugyanígy kapjuk, csak a mátrixot a másik oldalra rendezzük az induló egyenletekben:

$$\begin{array}{ll} x = A^{-1}b, & \delta b = A\delta x, \\ \downarrow & \downarrow \\ \|x\| \le \|A^{-1}\|\|b\|, & \|\delta b\| \le \|A\|\|\delta x\|. \end{array}$$

2.9.1 Lemma.

Ha $\|B\| < 1$, *akkor* $I + B$ *invertálható és indukált normára érvényes*

$$\|(I + B)^{-1}\| \le \frac{1}{1 - \|B\|}. \tag{2.15}$$

Az előző szakaszban látott norma és spektrál sugár összefüggése szerint most B spektrál sugara kisebb 1-nél, így minden sajátértéke is kisebb, azaz nem lehet $I + B$ egyik sajátértéke sem 0.

$$(I + B)^{-1} = (I + B - B)(I + B)^{-1} = I - B(I + B)^{-1},$$

ahonnan $\|(I + B)^{-1}\| \le 1 + \|B\|\|(I + B)^{-1}\|$, és átrendezéssel kapjuk az állítást. ■

Ha az együtthatómátrixot perturbáljuk δA-val: $(A+\delta A)(x+\delta x)=b \;\rightarrow (A+\delta A)\delta x=-\delta A x \;\rightarrow$ $\rightarrow \;\; \delta x=-(I+A^{-1}\delta A)^{-1}A^{-1}\delta A x$, innen kapjuk a becslést:

$$0\le\frac{\|\delta x\|}{\|x\|}\le\left\|\left(I+A^{-1}\delta A\right)^{-1}\right\|\left\|A^{-1}\right\|\|A\|\frac{\|\delta A\|}{\|A\|}\le\|A\|\left\|A^{-1}\right\|\frac{\|\delta A\|}{\|A\|}\frac{1}{1-\left\|A^{-1}\delta A\right\|}, \tag{2.16}$$

az utolsó lépésben felhasználtuk az előbbi lemmát.

2.10. A mátrix kondiciószáma

Az előbbi becslések azt mutatják, hogy a megoldás relatív megváltozása arányos a $\mathrm{cond}(A)=\|A\|\left\|A^{-1}\right\|$ számmal, ezért ezt a számot a mátrix kondiciószámának nevezzük. Szokás még a $\kappa(A)$ jelölés használata is. Ha az egyenletrendszer együtthatómátrixának kondiciószáma nagy, akkor az egyenletrendszert *gyengén meghatározottnak* nevezzük.

2.11. A relatív maradék

A $\|\delta x\|/\|x\|$ szám nem jellemzi a megoldó módszer stabilitását, mert a megoldó módszertől függetlenül nagy lehet, ha $\mathrm{cond}(A)$ nagy. Erre a célra alkalmasabb a maradékvektor. Tegyük fel, az $\tilde{x}$ közelítő megoldást kaptuk, ekkor a maradékvektor: $r=b-A\tilde{x}$, amit szokás még reziduum vektornak is nevezni. A relatív maradékot a következő formulával készítjük:

$$\eta=\frac{\|r\|}{\|A\|\|\tilde{x}\|}. \tag{2.17}$$

A stabilitás inverz megfogalmazása szerint a megoldó módszer stabil, ha a kapott eredmény egy kissé perturbált kiinduló eredményhez tartozik: $(A+\delta A)\tilde{x}=b$, ahol $\|\delta A\|/\|A\|$ kicsi.

Meg lehet mutatni: ha η nagy, $\|\delta A\|/\|A\|$ is nagy. Ugyanis $0=b-(A+\delta A)\tilde{x}=r-\delta A\tilde{x}$, ahonnan $\|r\|\le\|\delta A\|\|\tilde{x}\|$. Ezt η kifejezésébe írva

$$\eta=\frac{\|r\|}{\|A\|\|\tilde{x}\|}\le\frac{\|\delta A\|}{\|A\|}.$$

Másrészt, ha η kicsi, akkor 2-es normában a mátrix relatív megváltozása is kicsi. Ugyanis δA-ra megoldás

$$\delta A=\frac{r\tilde{x}^T}{\tilde{x}^T\tilde{x}},\;\text{ mert }\; b-\left(A+\frac{r\tilde{x}^T}{\tilde{x}^T\tilde{x}}\right)\tilde{x}=b-A\tilde{x}-r=0. \tag{2.18}$$

Ekkor 2-es normában $\left\|r\tilde{x}^T\right\|_2=\|r\|_2\left\|\tilde{x}^T\right\|_2$ (l. 2.5 gyakorlat), s ezzel $\dfrac{\|\delta A\|_2}{\|A\|_2}=\dfrac{\|r\|_2}{\|A\|_2\|\tilde{x}\|_2}$.

2.12. Gyakorlatok

2.1. Mutassuk meg: indukált normára $\|I\|=1$.

2.2. Ha A invertálható, akkor $\|x\|_A=\|Ax\|$ is vektornorma.

2.3. A mátrix kondiciószáma indukált normánál nem lehet kisebb 1-nél.

2.4. 2-es normánál az ortogonális vagy unitér mátrixok kondiciószáma 1.

2.5. $\left\|ab^T\right\|_2 = \|a\|_2 \|b\|_2$. $\left\|ab^T\right\|_1 = \|a\|_1 \|b\|_\infty$. $\left\|ab^T\right\|_\infty = \|a\|_\infty \|b\|_1$.

2.6. $U^T U = I$ (ortogonális) $\rightarrow \|AU\|_2 = \|A\|_2$.

2.7. $\big|\|A\| - \|B\|\big| \le \|A \pm B\|$.

2.8. $A = \begin{bmatrix} 2 & -3 & 1 \\ -4 & -2 & 1 \end{bmatrix}$, $\|A\|_1 = ?$ $\|A\|_\infty = ?$ $\|A\|_2 = ?$

2.9. $\|A\|_2 \le \sqrt{\|A\|_1 \|A\|_\infty}$.

2.10. Frobenius-norma: $\|A\|_F = \left(\sum_{i,j} |a_{ij}|^2 \right)^{1/2} = \sqrt{\mathrm{tr}(A^T A)} = \sqrt{\mathrm{tr}(AA^T)}$. Igazoljuk, ez is mátrixnorma, de nem indukált norma, $\|I\|_F = ?$ $\|Ax\|_2 \le \|A\|_F \|x\|_2$. (A 2-es normával illeszkedő mátrixnorma.)

2.11. $A = A^T$, akkor $\|A\|_2 = \rho(A)$ = spektrál sugár, azaz szimmetrikus mátrixokra a spektrálnorma a minimális norma. ($\|\,.\,\|_2 =$ spektrál norma).

2.12. $U^T U = I$ (ortogonális) $\rightarrow \|AU\|_F = \|A\|_F$.

2.13. $\|AB\|_2 = \|BA\|_2$, ha $A = A^T$ és $B = B^T$.

2.14. $\mathrm{cond}_2(A^T A) = \mathrm{cond}_2^2(A)$.

3. A numerikus lineáris algebra egyszerű transzformációi

3.1. Jelölések

A mátrixokat latin nagybetűkkel: $A,B,C,\ldots$ a vektorokat latin kisbetűkkel: $a,b,c,\ldots$ jelöljük, kivéve az i,j,k,l,m,n betűket, amelyeket indexekben fogunk használni. A skalárokra görög kisbetűket alkalmazunk. Ha az A mátrixot az $a_1,a_2,\ldots$ oszlopvektorokból állítjuk össze, akkor ezt így jelöljük: $A=[a_1a_2\ldots a_n]$. A mátrix egy másik megadási formája $A=[a_{ij}]$, ekkor az ij-edik elemet adjuk meg általánosan. Az n-edrendű egységmátrix $I_n=[e_1\,e_2\ldots e_n]$, amely az $e_1,e_2,\ldots,e_n$ Descartes-egységvektorokat tartalmazza az oszlopaiban. A transzponált jelölése: a^T, komplex esetben a transzponált konjugált jelölése a^H.

3.2. A mátrixok szorzása

Az $A=[a_{ij}]\in\mathbb{R}^{m\times n}, B=[b_{jk}]\in\mathbb{R}^{n\times l}$ mátrixok összeszorzásának eredménye a $C=AB=\left[c_{ik}\right]=$ $=\left[\sum_{j=1}^{n}a_{ij}b_{jk}\right]\in\mathbb{R}^{m\times l}$ mátrix. A vektorok egy sorból vagy oszlopból álló speciális mátrixoknak tekinthetők, szorzásuk nem jelent újat. Az alkalmazásokban megkülönböztetjük a vektorok kétféle szorzási módját. Az egyik a *skaláris* szorzat, például a^Tb, amelynek eredménye egy skalár. A másik a *diadikus* szorzat, például ab^T, az eredmény egy 1-rangú mátrix. Vegyük észre, az első esetben szükséges, hogy a vektorok hossza azonos legyen, a második esetben nem.

3.1 Gyakorlat. Írjunk fel egy diádot. Indokoljuk meg, hogy a rangja tényleg 1. Hogyan egyszerűbb egy vektort diáddal szorozni? *a)* Képezzük $A=ab^T$-t, majd Ax-et. *b)* Először kiszámítjuk b^Tx-et és ezzel a skalárral szorozzuk az a vektort.

A továbbiakban rátérünk speciális mátrixok ismertetésére.

3.3. Permutáció-mátrix

Úgy kapjuk, ha az egységmátrix sorait vagy oszlopait permutáljuk, emiatt minden sor és oszlopban csak egy 1-es fordulhat elő, a többi elem 0. Az ábrázolásukhoz nem szükséges a mátrixot kitölteni, elég egy egész (számokból álló) vektor.

Tegyük fel, egy mátrix sorait cserélgetjük és ezt szeretnénk egy vektorban feljegyezni, ami a permutációmátrixot reprezentálja. Kezdetben a vektor k-adik eleme legyen egyenlő k-val. A cserék során ennek a vektornak az elemeit cserélgessük ugyanúgy, mint a mátrix sorait (mintha oszlopvektorként a mátrixhoz csatoltuk volna). Így a végén mindegyik sorról meg tudjuk állapítani, hogy hova került. Ha például az első elem 5-ös, akkor ez azt jelenti, hogy az ötödik sor az elsőbe került.

3.2 Gyakorlat. Tekintsük a $\Pi=[e_2,e_4,e_3,e_1]$ permutáció-mátrixot és ellenőrizzük, hogy az inverze a transzponáltja! Ezt a tényt általánosan bizonyítsuk be! Hogyan tároljuk a fenti mátrixot egy 4-elemű vektorban?

3.4. Diáddal módosított egységmátrix

A numerikus lineáris algebrában különösen fontos szerepet játszanak az olyan egyszerű mátrixok, amelyek az egységmátrixtól csak egy diádban különböznek:

$$F=I+ab^T \tag{3.1}$$

Segítségükkel a különféle lineáris algebrai transzformációk egyszerűen végezhetők, a bennük szereplő a és b vektorok megválasztása mindig az elérendő céltól függ.

Ennek a mátrixnak az inverze könnyen meghatározható. Feltételezve, hogy $F^{-1} = I + \alpha ab^T$, az $FF^{-1} = I$ összefüggésből adódik: $\alpha = -1/(1 + b^T a)$, így

$$F^{-1} = I - \frac{ab^T}{1 + b^T a}. \tag{3.2}$$

Az inverz nem létezik, ha $1 + b^T a = 0$, ebből már sejthetjük, hogy a nevező nem más, mint F determinánsa.

3.5. Példa

Ha az egységmátrixból kivesszük az i-edik oszlopot és a helyére betesszük az a vektort:

$$F = I + (a - e_i)e_i^T.$$

Az inverze:

$$F^{-1} = I - \frac{(a - e_i)e_i^T}{1 + e_i^T(a - e_i)} = I - \frac{(a - e_i)e_i^T}{e_i^T a}.$$

Az ilyen típusú mátrixok fontosak a lineáris egyenletrendszer-megoldó algoritmusoknál.

3.3 Gyakorlat. Ellenőrízzük: $F^{-1}a = e_i$.

3.6. Példa

A következő műveletet végezzük: az A mátrix i-edik oszlopát α-val szorozzuk és hozzáadjuk a k-adik oszlopához. Írjuk fel azt a mátrixot, amellyel szorozva A-t, pont ez történik!

Megoldás. $A + \alpha Ae_i e_k^T = A(I + \alpha e_i e_k^T)$.

3.4 Gyakorlat. Az előbb kapott összefüggés segítségével bizonyítsuk be, hogy a mátrix determinánsa nem változik, ha egy oszlopának számszorosát egy másik oszlopához hozzáadjuk. Használjuk fel a szorzatmátrix determinánsára tanultakat!

3.7. Példa

Igazoljuk, hogy az $\left|I + ab^T\right|$ determináns egyenlő $1 + b^T a$-val!

Megoldás. Feltesszük, az a és b vektorok egyike sem zérus, mert különben a feladat triviális volna. Legyen az a vektor i-edik eleme $e_i^T a = a_i \neq 0$, és tekintsük az $I - (a/a_i - e_i)e_i^T$ mátrixot. Ennek minden átlóeleme 1 és az i-edik oszlopában vannak még nemzérus elemek. De ezeket a nemzérus elemeket az i-edik sor valamely számszorosának hozzáadásával ki lehet nullázni, ebből következik, hogy a determinánsa 1. Most szorozzuk az $I + ab^T$ mátrixot balról $I - (a/a_i - e_i)e_i^T$-vel. Ez az a vektort az $a_i e_i$ vektorba viszi, így az eredmény: $I - (a/a_i - e_i)e_i^T + a_i e_i b^T$, amely már csak az i-edik sorában és oszlopában különbözik az egységmátrixtól. Most szorozzuk a kapott mátrix k-adik oszlopát a_k/a_i-vel és adjuk hozzá a i-edik $(i \neq k)$ oszlophoz (ld. 3.6 Példa):

$$\left(I - (\frac{a}{a_i} - e_i)e_i^T + a_i e_i b^T\right)\left(I + \frac{a_k}{a_i}e_k e_i^T\right) = I - \left(\frac{a - a_k e_k}{a_i} - e_i\right)e_i^T + a_i e_i b^T + a_k b_k e_i e_i^T.$$

Mint látjuk, az a vektor k-adik eleme kinullázódott, és az i-edik átlóelem $1 + a_i b_i + a_k b_k$ lett. Ezt a műveletet minden $k \neq i$-re végrehajtva az a/a_i vektor minden átlón kívüli eleme kinullázódik, az i-

edik átlóelem $1+b^T a$, a többi pedig 1-gyel egyenlő. A következő fázisban az e_k^T, $k \neq i$ sorvektorokkal az $a_i e_i b^T$ sorvektor nemdiagonális elemeit a determináns megváltozása nélkül kinullázhatjuk.

3.8. Diádösszegek, kifejtések

Az n-edrendű egységmátrix felírható diádösszegként: $I_n = \sum_{i=1}^n e_i e_i^T$. Ha ezt beírjuk két mátrix közé, akkor a szorzatmátrix diádösszeg-előállítását kapjuk:

$$AB = \sum_{i=1}^n A e_i e_i^T B \ ,$$

A oszlopai és B sorai képezik a diádokat, i-edik oszlop és i-edik sor.

3.5 Gyakorlat: Írjuk ki ADB diádösszeg előállítását, ahol $D = [d_i \delta_{ij}]$ diagonálmátrix, (csak a főátló elemei nemzérusok).

Tudjuk, az n-edrendű x vektor előállítása az egységvektorok segítségével $x = \sum_{i=1}^n e_i (e_i^T x)$. Az előállítás hasonló a $\{q_i\}_{i=1}^n$ ortonormált vektorrendszerrel. Ugyanis vezessük be a $Q = [q_1 q_2 \ldots q_n]$ mátrixot. Ekkor $Q^T Q = I = QQ^T$ az ortonormáltság miatt, tehát írható $x = QQ^T x = \sum_{i=1}^n q_i (q_i^T x)$. Az ilyen tulajdonságú Q mátrixokat *ortogonális* (komplex megfelelője: *unitér*) mátrixoknak nevezzük.

3.9. Definíció

Az $\{a_i\}_{i=1}^n$ és $\{b_i\}_{i=1}^n$ rendszerek *biortogonális vektorrendszert* alkotnak, ha $a_i^T b_j = \alpha_i \delta_{ij}$, $\alpha_i \neq 0$ teljesül bármely indexre. Ha n a vektorok dimenziója, akkor a rendszer *teljes.*

3.6 Gyakorlat. Az előbbi vektorokat gyűjtsük az $A = [a_1, a_2, \ldots, a_n]$ és $B = [b_1, b_2, \ldots, b_n]$ mátrixba. Ellenőrízzük, hogy $A^T B$ diagonálmátrix! Ekkor az x vektor hogyan állítható elő az a_i vektorok lineáris kombinációjaként? És hogyan fejthető ki a b_i vektorok segítségével?

3.10. Tétel, mátrix egyszerű szorzatokra bontása

Minden nemszinguláris $A \in \mathbb{R}^{n \times n}$ mátrix felírható n egyszerű mátrix szorzataként, ahol egy tényező egy permutációból és egy $I + (a_i - e_i) e_i^T$ típusú tagból áll. A permutációra nincs mindig szükség.

Bizonyítás. Megadjuk az eljárást. Az első lépésben vizsgáljuk meg az A mátrix első oszlopát. Ha az első elem $a_{11} = e_1^T A e_1 \neq 0$, akkor sorcserére nincs szükség. Ha az első elem zérus, akkor az oszlopban keresünk egy nemzérus elemet, majd ennek a sorát felcseréljük az első sorral. Ha az oszlop minden eleme zérus volna, akkor nem lenne invertálható a mátrix. Az első permutáció mátrixot jelüljük Π_1-gyel és legyen $A_1 = \Pi_1 A$.

Most szorozzuk A_1-et a $T_1 = I - (A_1 e_1 - e_1) e_1^T / e_1^T A_1 e_1$ mátrixszal. Tudjuk, ennek eredményeként az első oszlop e_1-be megy át és $T_1^{-1} = I + (A_1 e_1 - e_1) e_1^T$. A második lépésben hasonlóan járunk el $T_1 A_1$ második oszlopával: $A_2 = \Pi_2 T_1 A_1$ olyan mátrix lesz, ahol a 22-es pozición nemzérus elem van. Így a $T_2 = I - (A_2 e_2 - e_2) e_2^T / e_2^T A_2 e_2$ mátrixszal szorozva a második oszlopot az e_2 vektorba visszük. Vegyük észre, T_2 az e_1 vektort helyben hagyja.

Hasonlóan folytatva, az i-edik lépésben $A_i = \Pi_i T_{i-1} A_{i-1}$ olyan mátrix, ahol az ii pozicióban nemzérus áll. (Ha az i-edik oszlop zérus volna, ismét ellentmondásba kerülnénk azzal a feltevéssel, hogy a mátrix nemszinguláris.) Ekkor a $T_i = I - (A_i e_i - e_i) e_i^T / e_i^T A_i e_i$ mátrixszal szorozva kapunk e_i-t az i-

edik oszlopban és az eddig elkészült kisebb indexű egységvektorok sem romlottak el. A n-edik lépés után egységmátrixot kapunk, tehát végeredményben a mátrix inverzével szoroztunk. A szorzatokat összegyűjtve:

$$\Pi_1^T T_1^{-1} \Pi_2^T T_2^{-1} \ldots T_n^{-1} = A\,.$$

Figyeljük meg, T_i^{-1} megadásához elég, ha az i indexet és a benne szereplő $a_i = A_i e_i$ vektort ismerjük.

3.11. Háromszögmátrixok szorzatokra bontása

Az L mátrixot alsó háromszögmátrixnak nevezzük, ha a főátló feletti elemei mind zérust tartalmaznak. Hasonlóan az U mátrix felső háromszög mátrix, ha a főátló alatti elemek zérusok. A háromszögmátrixok szorzatokra bontása különösen egyszerű. Az előbbi tételt alkalmazva azonnal adódik az n-edrendű L alsó háromszögmátrix szorzat-előállítása:

$$L = \left(I + (L-I)e_1 e_1^T\right)\left(I + (L-I)e_2 e_2^T\right)\ldots\left(I + (L-I)e_n e_n^T\right),$$

ami tömören így is írható

$$L = \prod_{i=1}^{n}\left(I + (L-I)e_i e_i^T\right),$$

ha megjegyezzük, hogy a tényezők növekvő indexek szerint mindig balról jobbra haladva írandók. A kifejezést közvetlenül is igazolhatjuk a j-edik oszlop meghatározásával. Jobbról az e_j vektorral szorozva az első e_j vektortól különböző eredményű szorzat $e_j + Le_j - e_j = Le_j$ az L mátrix j-edik oszlopa. A többi szorzatban lévő e_k, $k < j$ vektorral ennek a skaláris szorzata zérus, emiatt a végeredmény Le_j. Felírható a sorvektorokkal is a szorzatokra bontás:

$$L = \prod_{i=1}^{n}\left(I + e_i e_i^T (L-I)\right).$$

Ellenőrízzük, hogy ennek a j-edik sora visszaadja L j-edik sorát!

A U felső háromszögmátrixra vonatkozó hasonló összefüggések:

$$U = \prod_{i=n\ (-1)}^{1}\left(I + (U-I)e_i e_i^T\right) = \prod_{i=n\ (-1)}^{1}\left(I + e_i e_i^T (U-I)\right),$$

ahol a tényezők balról jobbra az indexek szerint csökkenő sorrendben írandók.

3.12. Vetítőmátrixok

Tekintsük a

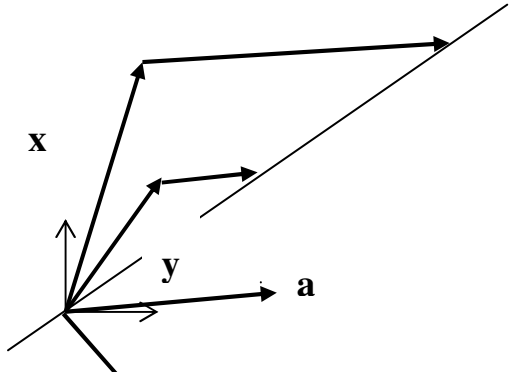
$$P = I - ab^T \tag{3.3}$$

mátrixot, ahol $b^T a = 1$. Ennek determinánsa 0, így az inverze nem létezik. Van azonban egy érdekes tulajdonsága: önmagával szorozva visszaadja saját magát:

$$\left(I - ab^T\right)\left(I - ab^T\right) = I - 2ab^T + ab^T ab^T = I - ab^T.$$

Az $P^2 = P$ feltételt kielégítő mátrixokat *vetítő-mátrixoknak* vagy *projektoroknak* nevezzük.

Ha $a = b$, akkor a mátrix szimmetrikus. A szimmetrikus vetítőmátrixok *ortogonális* vetítők,



mert egy altérre merőleges vetítést valósítanak meg. Ha a és b nem egyirányú, akkor *ferde* vetítésről beszélünk. Szokás még a projektorokat *idempotens* mátrixoknak nevezni arra a tulajdonságukra utalva, hogy a mátrix minden hatványa önmaga. Vegyük észre, (3.3)-ból: $Pa = 0$ és $b^T P = 0$.

Az 1. ábra azt szemlélteti, a (3.3) projektor hogy vetíti az x és y vektort az a irány mentén a b normálisú síkba, amely áthalad az origón. Ha a iránya megegyezne b irányával, akkor a síkba vetítés merőlegesen történne.

3.7 Gyakorlat. Ellenőrízzük: ha P projektor, akkor $I - P$ is az.

3.8 Gyakorlat. Egy sík normálvektora s, egyenlete $s^T x = \sigma$. Legyen a vetítőmátrix $P = I - ss^T / s^T s$. Mutassuk meg, a tér bármely y vektorára a $Py + \sigma s / s^T s$ művelet egy síkbeli vektort állít elő.

3.9 Gyakorlat. Mutassuk meg, az előbbi P mátrixszal $Py \perp s$. Adjuk meg a $Py + \sigma s / s^T s$ vektort és az y vektort összekötő vektort!

3.13. Involutórius mátrixok

Az A mátrixot *involutóriusnak* nevezzük, ha eleget tesz az $A^2 = I$ összefüggésnek. Minden projektor $A = I - 2P$ alakban meghatároz egy involutórius mátrixot:

$$(I - 2P)(I - 2P) = I - 4P + 4P = I,$$

és minden involutórius mátrix $(I - A)/2$ alakban meghatároz egy projektort:

$$(I - A)(I - A)/4 = (2I - 2A)/4 = (I - A)/2.$$

Innen látható, 1-nél nagyobb méretű egységmátrixból végtelen sokféleképp lehet gyököt vonni.

3.10 Gyakorlat. Igazoljuk, hogy a $J = [e_n e_{n-1} \ldots e_1]$ mátrix, ahol az egységmátrix oszlopai fordított sorrendben vannak felsorolva, involutórius mátrix. Milyen projektort határoz meg ez a mátrix, ha $n = 2,3$?

Az $ab^T / b^T a$, $b^T a \neq 0$ projektorral a következő involutórius mátrixot készíthetjük: $I - 2ab^T / b^T a$. Az 1.ábrából megállapíthatjuk, hogy ez a mátrix a b normálisú síkra való „ferde” tükrözést végzi, ami annyit jelent, hogy az a irány mentén eljutunk a síkig, majd azt keresztezve ugyanakkora távolságot teszünk meg a túloldalon. A tükrözés akkor merőleges a síkra, ha $a = b$.

3.11 Gyakorlat. Mutassuk meg, hogy az $I - 2(x - y)(x - y)^T / (x - y)^T (x - y)$ mátrix az x és y vektorokat egymásba tükrözi, ha azok különbözőek és $x^T x = y^T y$.

3.12 Gyakorlat. Az előbbi tükröző mátrixszal lehetőségünk van arra, hogy az x vektort az $y = \pm \sigma e_1$ vektorba tükrözzük, ahol $\sigma^2 = x^T x$. Hogyan válasszuk meg σ előjelét ahhoz, hogy a kivonási jegyveszteséget biztosan elkerüljük?

3.14. Blokk mátrixok

A mátrixokat nemcsak skalár elemekből rakhatjuk össze, hanem kisebb méretű mátrixokból is. Az ilyen mátrix elemeit *blokkoknak* nevezzük, ha pedig egy mátrixot kisebb mátrixokra bontunk, akkor a mátrixot *blokkosítjuk*. A blokkosítás történhet a következőképp: Az egységmátrixot az oszlopok

mentén felszeleteljük k részre: $I=[E_1,E_2,\dots,E_k]$. Ha a sorokat ugyanilyen módon osztjuk fel blokkokra, akkor az ij-edik blokk $A_{ij}=E_i^TAE_j$ és a mátrix:

$$A=\begin{bmatrix} A_{11} & A_{12} & \dots & A_{1k} \\ A_{21} & A_{22} & \dots & A_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ A_{k1} & A_{k2} & \dots & A_{kk} \end{bmatrix}.$$

3.13 Gyakorlat. Legyen $F=I+UV^T$, U és V $n\times l$-es mátrixok, azaz $l<n$ oszlopból állnak. Ha a kijelölt inverz létezik, ellenőrízzük: $F^{-1}=I-U(I_l+V^TU)^{-1}V^T$, ahol I_l $l\times l$-es egységmátrix.

4. Mátrixok LU-felbontása, Gauss-Jordan algoritmus

Az *LU*-felbontás nem más, mint a Gauss-elimináció olyan átrendezése, ahol a részleteredményeket is elrakjuk. Ez úgy történik, hogy az A mátrixot felbontjuk egy L alsó és egy U felső háromszögmátrix szorzatára.

4.1. Tétel, LU-felbontás.

Ha $A\in\mathbb{R}^{n\times n}$ nemszinguláris mátrix, akkor a sorai mindig átrendezhetők egy P permutáció-mátrixszal PA-ba úgy, hogy az felbontható egy L alsó és U felső háromszögmátrix szorzatára. PA felbontása egyértelmű, ha L átlóelemeit 1-nek választjuk.

Bizonyítás. Tekintsük A első oszlopát. Ha a_{11} zérus, akkor keressünk az oszlopban egy nemzérus elemet és sorcserével mozgassuk az első sorba. A továbbiakban feltesszük, hogy $a_{11}\neq 0$. Ekkor szorozzuk A-t az $L_1^{-1}=I-(Ae_1/a_{11}-e_1)e_1^T$ mátrixszal! A 3.7 példában láttuk, ennek a mátrixnak determinánsa és minden átlóeleme 1, következik, hogy az inverzét úgy kapjuk, ha a benne szereplő diádot pozitív előjellel vesszük. A szorzás eredményeként az Ae_1 oszlopvektor

$$\left(I-(Ae_1/a_{11}-e_1)e_1^T\right)Ae_1=Ae_1-Ae_1+a_{11}e_1=a_{11}e_1 \tag{4.1}$$

-be megy át, tehát

$$L_1^{-1}A=\begin{bmatrix} a_{11} & * & \dots & * \\ 0 & * & \dots & * \\ \vdots & \vdots & \ddots & \vdots \\ 0 & * & \dots & * \end{bmatrix}, \tag{4.2}$$

ahol a * egységesen nemzérus mátrixelemeket jelöl. Látjuk, a felső háromszögmátrix első oszlopa megjelent. $L_1=I+(Ae_1/a_{11}-e_1)e_1^T$ pedig a *LU*-felbontásban szereplő *L* mátrix első szozója, ahonnan kiolvashatjuk *L* első oszlopvektorát: Ae_1/a_{11}-et.

A második lépésben ugyanezt ismételjük meg a kapott mátrix jobb alsó $(n-1)\times(n-1)$-es blokkjára, ahol az első lépés valamely nemzérus elemnek a 2,2 pozicióba mozgatása, ha szükséges:

$$A_2=\begin{pmatrix} a_{11} & * & \dots & * \\ 0 & \boxed{*} & \dots & * \\ \vdots & \vdots & \ddots & \vdots \\ 0 & * & \dots & * \end{pmatrix},$$

így L második oszlopában az első elem 0, a második elem 1. Az eljárást hasonlóan folytatva végül

$$L = L_1 L_2 \dots L_{n-1}, \quad U = L_{n-1}^{-1} L_{n-2}^{-1} \dots L_1^{-1} PA = \begin{pmatrix} * & * & \dots & * \\ & * & \dots & * \\ & & \ddots & \vdots \\ & & & * \end{pmatrix}. \tag{4.3}$$

■

Ha az $Ax = b$ egyenletrendszert oldjuk meg, akkor a b vektort célszerű az A mátrix mellé venni: $[A,b]$, mert b-re is ugyanazok a transzformációk hatnak. Például legyen az egyenletrendszer:

$$\begin{bmatrix} 2 & 0 & 3 \\ -4 & 5 & -2 \\ 6 & -5 & 4 \end{bmatrix} x = \begin{bmatrix} -1 \\ 3 \\ -3 \end{bmatrix}.$$

Vegyük észre, az L_1^{-1}-gyel való szorzás a mátrix első sorát nem változtatja meg: $e_1^T L_1^{-1} = e_1^T$. A jobb alsó $(n-1)$-edrendű blokkban pedig a következőket kell számítani, $k,i > 1$:

$$e_i^T \left(I - \left(\frac{Ae_1}{a_{11}} - e_1 \right) e_1^T \right) Ae_k = a_{ik} - \frac{a_{i1}a_{1k}}{a_{11}} = a_{ik} - \left(\frac{a_{i1}}{a_{11}} \right) a_{1k}.$$

Ez mutatja, hogy az $A - \dfrac{Ae_1}{e_1^T Ae_1} e_1^T A$ diádot kell számítanunk a jobb alsó $(n-1)$-edrendű blokkra. Az ebben szereplő oszlopvektor éppen L_1 első oszlopa, így célszerűen a következőképpen járhatunk el: kijelöljük a főelemet, vele leosztjuk az alatta lévő oszlopelemeket, a saját sorát pedig változatlanul átmásoljuk. A mátrix többi részében ebből a sorból és oszlopból készített diádot vonjuk le:

$$\begin{bmatrix} 2 & 0 & 3 & -1 \\ -4 & 5 & -2 & 3 \\ 6 & -5 & 4 & -3 \end{bmatrix} \to \begin{bmatrix} \boxed{2} & 0 & 3 & -1 \\ -2 & 5 & 4 & 1 \\ 3 & -5 & -5 & 0 \end{bmatrix} \to \begin{bmatrix} 2 & 0 & 3 & -1 \\ -2 & \boxed{5} & 4 & 1 \\ 3 & -1 & -1 & 1 \end{bmatrix}$$

$$L = \begin{bmatrix} 1 & & \\ -2 & 1 & \\ 3 & -1 & 1 \end{bmatrix}, \quad U = \begin{bmatrix} 2 & 0 & 3 \\ & 5 & 4 \\ & & -1 \end{bmatrix}.$$

A végén még megoldandó $Ux = [-1 \quad 1 \quad 1]^T$, alulról felfelé megoldva $x = [1 \quad 1 \quad -1]^T$.

4.1 Gyakorlat. Oldjuk meg *LU*-felbontással a következő egyenletrendszert:

$$\begin{bmatrix} 2 & 2 & 3 \\ 4 & 3 & 7 \\ 6 & 7 & 5 \end{bmatrix} x = \begin{bmatrix} 1 \\ 5 \\ -3 \end{bmatrix}.$$

4.2. Az LU-felbontás műveletigénye.

Az első lépésben az oszlopvektor leosztása $n-1$ osztás, a diád levonása $(n-1)^2$ szorzást és összeadást igényel. Az aritmetikai műveletek mindegyike ugyanannyi idejűnek számít, emiatt az első lépés műveletigénye: $(n-1)(2n-1)$ flop (= *fl*oating point *op*eration, magyarul: lebegőpontos művelet). A következő lépés igénye $(n-2)(2n-3)$ flop, így a teljes műveletigény $\sum_{k=1}^{n-1}(k-1)(2k-1)$ flop. Ezt úgy közelítjük, hogy a legmagasabb fokú tagot integráljuk 0-tól n-ig: $2n^3/3$. A korrekciós tagok n kisebb hatványai, nem határozzuk meg őket, mert a legmagasabbfokú tag a domináns.

4.2 Gyakorlat. Mennyi Ax, LUx, $U^{-1}L^{-1}x$ műveletigénye? Az utolsó példánál alkalmazzuk a 2.11 szakaszban megismert faktorizációs összefüggéseket!

4.3. Blokk LU-felbontás

Néha célszerű a felbontást – vagy a mátrix invertálását – blokkosított formában végezni. Tipikusan ez a helyzet akkor, amikor az egyik elkülönített blokk egyszerűen invertálható, például azért mert egységmátrix, vagy alsó ill. felső háromszögmátrix. Mi most a blokk LU -felbontást a 2×2 -es esetben fogjuk megnézni. A főelem ilyenkor blokk, amelyről fel kell tételeznünk, hogy létezik az inverze. Legyen az egységmátrix egy felosztása $I=[E_1,E_2]$, $A_{ij}=E_i^T AE_j$, ekkor az L mátrix a (4.1)-ben látható L_1 mátrix blokkos megfelelője (ld. még *3.13 Gyakorlat*)

$$L=I-\left(AE_1A_{11}^{-1}-E_1\right)E_1^T \tag{4.4}$$

és a mátrix blokkos felbontása a következő:

$$\begin{bmatrix} \boxed{A_{11}} & A_{12} \\ A_{21}A_{11}^{-1} & A_{22}-A_{21}A_{11}^{-1}A_{12} \end{bmatrix},\ \text{ahol}\ L=\begin{bmatrix} I_1 & 0 \\ A_{21}A_{11}^{-1} & I_2 \end{bmatrix},\ U=\begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22}-A_{21}A_{11}^{-1}A_{12} \end{bmatrix}. \tag{4.5}$$

4.4. Schur-komplemens.

A felbontás jobb alsó sarkában megjelent mátrixot az A mátrix A_{11}-re vonatkozó Schur-komplemensének nevezzük és jelölése: $\left(A\middle|A_{11}\right)=A_{22}-A_{21}A_{11}^{-1}A_{12}$. Természetesen létezik az A_{22}-re vonatkozó Schur-komplemens is. Ez az előbbiből úgy jön létre, hogy az $1\leftrightarrow 2$ indexcserét elvégezzük.

4.5. Particionált inverz

A (4.5) felbontás alapján írhatjuk:

$$A=\begin{bmatrix} I_1 & 0 \\ A_{21}A_{11}^{-1} & I_2 \end{bmatrix}\begin{bmatrix} A_{11} & A_{12} \\ 0 & \left(A\middle|A_{11}\right) \end{bmatrix}=\begin{bmatrix} I_1 & 0 \\ A_{21}A_{11}^{-1} & I_2 \end{bmatrix}\begin{bmatrix} A_{11} & \\ & \left(A\middle|A_{11}\right) \end{bmatrix}\begin{bmatrix} I_1 & A_{11}^{-1}A_{12} \\ 0 & I_2 \end{bmatrix},$$

ahonnan

$$A^{-1}=\begin{bmatrix} I_1 & -A_{11}^{-1}A_{12} \\ 0 & I_2 \end{bmatrix}\begin{bmatrix} A_{11}^{-1} & \\ & \left(A\middle|A_{11}\right)^{-1} \end{bmatrix}\begin{bmatrix} I_1 & 0 \\ -A_{21}A_{11}^{-1} & I_2 \end{bmatrix}. \tag{4.6}$$

A diádösszeg kifejtés blokkos alakját felhasználva (ld. *3.5 Gyakorlat*) ez még a két blokk-oszlop és blokk-sor alapján kifejthető az

$$A^{-1}=\begin{bmatrix} A_{11}^{-1} & 0 \\ 0 & 0 \end{bmatrix}+\begin{bmatrix} -A_{11}^{-1}A_{12} \\ I_2 \end{bmatrix}\left(A\middle|A_{11}\right)^{-1}\begin{bmatrix} -A_{21}A_{11}^{-1} & I_2 \end{bmatrix} \tag{4.7}$$

alakban.

4.5.1 Feladatok

1. A *3.13 Gyakorlat* alapján mutassuk meg, hogy a (4.4) mátrix inverze úgy készíthető, hogy a 21-es blokk negatívját vesszük. A felső háromszögmátrixra vonatkozó eredmény innen transzponálással származtatható.

2. L_{11} alsó háromszögmátrix, amelyet alul kiegészítünk egy blokk-sorral $\begin{bmatrix} L_{21} & L_{22} \end{bmatrix}$ nagyobb méretű alsó háromszögmátrixra. Mutasssuk meg, hogyha a diagonálblokkok invertálhatók, akkor particionált inverz formulával a kiegészített mátrix inverze

$$\begin{bmatrix} L_{11} & \\ L_{21} & L_{22} \end{bmatrix}^{-1} = \begin{bmatrix} L_{11}^{-1} & \\ -L_{22}^{-1} L_{21} L_{11}^{-1} & L_{22}^{-1} \end{bmatrix}$$

4.6. A Gauss-Jordan módszer az inverz mátrix kiszámítására

Láttuk, minden mátrix, amelynek van inverze, egyszerű mátrixok szorzatára bontható, ahol az n művelet mindegyike tartalmaz egy sorcserét – ha szükséges, és egy diáddal módosított egységmátrixszal való szorzást. Egy ilyen műveletsorozattal a mátrix az egységmátrixba transzformálható. Kézenfekvő az ötlet: a mátrixhoz hozzáírjuk az egységmátrixot: $A \to [A, I]$ és a kibővített mátrixra alkalmazzuk a transzformáció-sorozatot: $[TA, T] = [I, T]$. Világos, $T = A^{-1}$.

Ez a módszer alkalmas lineáris egyenletrendszer megoldására is, de a műveletszámlálás azt mutatja, hogy az LU-felbontás előnyösebb. Ha azonban a mátrix inverzét akarjuk előállítani, akkor a műveletigény ugyanakkora, sőt lehetőség van arra, hogy a mátrixot helyben invertáljuk.

Tegyük fel, az i-edik lépésben A_i már olyan, hogy a sorcserét végrehajtottuk, ha kellett. Az i-edik szorzás:

$$\left(I - \frac{A_i e_i - e_i}{e_i^T A_i e_i} e_i^T \right) A_i = A_i - \frac{A_i e_i e_i^T A_i}{e_i^T A_i e_i} + \frac{e_i e_i^T A_i}{e_i^T A_i e_i}.$$

Itt jobb oldalon az első két tag azt a diád-levonást jelenti, amit már megismertünk. Az LU-felbontáshoz képest azonban eltérés, hogy az i-edik sor és oszlop kivételével minden területre kell alkalmaznunk. Az harmadik tag azt mutatja, hogy az i-edik sort a főelemmel kell osztani, az első két tagból származó i-edik sor ugyanis zérus.

Az elmondottakat egy példán szemléltetjük. Invertálandó a $\begin{bmatrix} 0 & 1 & 1 \\ 1 & 2 & 3 \\ 1 & 3 & 6 \end{bmatrix}$ mátrix. A kibővített mátrixban az első lépés egy sorcsere:

$$\begin{bmatrix} 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 2 & 3 & 0 & 1 & 0 \\ 1 & 3 & 6 & 0 & 0 & 1 \end{bmatrix} \underset{\text{sorcsere}}{1 \leftrightarrow 2} \begin{bmatrix} \boxed{1} & 2 & 3 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 3 & 6 & 0 & 0 & 1 \end{bmatrix} \overset{Tr1}{\to}$$

Az első transzformációs lépésben az első oszlop átmegy e_1-be, az első sort végigosztjuk a főelemmel, a többi helyen pedig végrehajtjuk az első diád levonását:

$$\to \begin{bmatrix} 1 & 2 & 3 & 0 & 1 & 0 \\ 0 & \boxed{1} & 1 & 1 & 0 & 0 \\ 0 & 1 & 3 & 0 & -1 & 1 \end{bmatrix} \overset{Tr2}{\to} \begin{bmatrix} 1 & 0 & 1 & -2 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & \boxed{2} & -1 & -1 & 1 \end{bmatrix} \overset{Tr3}{\to} \begin{bmatrix} 1 & 0 & 0 & -3/2 & 3/2 & -1/2 \\ 0 & 1 & 0 & 3/2 & 1/2 & -1/2 \\ 0 & 0 & 1 & -1/2 & -1/2 & 1/2 \end{bmatrix}.$$

Az utolsó lépésben az induló egységmátrix helyén megjelent az inverz.

A „helyben” invertáláshoz azt kell észrevennünk, hogy minden lépésben összegyűjthető egy egységmátrix a kibővített mátrixból. Ezt szükségtelen tárolni. A jobboldali 3×3-as területen minden lépésben egy új vektor jelenik meg, a bal oldali 3×3-as területen pedig a távozó vektor helyére egy egységvektor lép be. A „tömör” algoritmusban a jobb oldalon belépő új vektort beírjuk a bal oldalon belépő egységvektor helyére. Az i-edik egységvektor helyén a jobb oldalról származó új vektor

$$\left(I - \frac{A_i e_i - e_i}{e_i^T A_i e_i} e_i^T \right) e_i = e_i - \frac{A_i e_i - e_i}{e_i^T A_i e_i} = \begin{cases} 1/\, e_i^T A_i e_i, & j = i, \\ -a_{ji}^{(i)} / a_{ii}^{(i)}, & j \neq i \end{cases}$$

Ez fog átkerülni a bal oldalon az i-edik oszlopba. Így a tömör algoritmusban a főelem helyére annak reciproka kerül, az oszlop többi eleme pedig negatív előjelet kap és osztódik a főelemmel. A levonandó diád kezelése ugyanaz, mint korábban. A bekeretezett elem jelöli ki azt a diádot (sor, oszlop), amelyből a levonandó diádot képezzük. Tehát a tömör algoritmus:

$$\begin{bmatrix} 0 & 1 & 1 \\ 1 & 2 & 3 \\ 1 & 3 & 6 \end{bmatrix} \overset{\substack{1\leftrightarrow 2 \\ \text{sorcsere}}}{\rightarrow} \begin{bmatrix} \boxed{1} & 2 & 3 \\ 0 & 1 & 1 \\ 1 & 3 & 6 \end{bmatrix} \overset{Tr1}{\rightarrow} \begin{bmatrix} 1 & 2 & 3 \\ 0 & \boxed{1} & 1 \\ -1 & 1 & 3 \end{bmatrix} \overset{Tr2}{\rightarrow} \begin{bmatrix} 1 & -2 & 1 \\ 0 & 1 & 1 \\ -1 & -1 & \boxed{2} \end{bmatrix} \overset{Tr3}{\rightarrow}$$

$$\rightarrow \begin{bmatrix} 3/2 & -3/2 & -1/2 \\ 1/2 & 3/2 & -1/2 \\ -1/2 & -1/2 & 1/2 \end{bmatrix} \overset{\substack{1\leftrightarrow 2 \\ \text{oszlopcsere}}}{\rightarrow} \begin{bmatrix} -3/2 & 3/2 & -1/2 \\ 3/2 & 1/2 & -1/2 \\ -1/2 & -1/2 & 1/2 \end{bmatrix}.$$

A kezdeti sorcsere miatt nem az eredeti, hanem a ΠA mátrixot invertáltuk, ahol Π permutációmátrix. Ennek az inverze $A^{-1}\Pi^T$, mert $\Pi^{-1} = \Pi^T$. Így a kapott eredményt még szoroznunk kellett jobbról Π^T-vel, ami itt az $1 \leftrightarrow 2$ oszlopcserét jelenti.

4.4 Gyakorlat. Mi a Gauss-Jordan invertáló módszer műveletigényében a domináns tag?

5. Az LU-felbontás tulajdonságai, speciális inverzek

5.1. Szimetrikus pozitív definit mátrixok

Egy valós szimmetrikus A mátrixot *pozitív definitnek* nevezünk, ha $x^T Ax > 0$ teljesül minden $x \neq 0$ vektorra. *Pozitív szemidefinit* a mátrix, ha csak $x^T Ax \geq 0$ teljesül. A *negatív definit* és *negatív szemidefinit* tulajdonságot hasonlóképp értelmezzük, ha $x^T Ax < 0$ vagy $x^T Ax \leq 0$ valamelyike teljesül. *Indefinit* esetben a belső szorzat negatív és pozitív értékeket egyaránt felvehet.

A pozitív definit tulajdonságnak adható még két másik ekvivalens definiciója. Az egyik szerint ekkor a mátrix minden sajátértéke pozitív, a másik szerint pedig a bal felső sarok aldeterminánsok (főminorok) mind pozitívak. Szemidefinit mátrixnak van zérus sajátértéke és zérus értékű sarok aldeterminánsa.

A nemszimmetrikus mátrixot pozitív definitnek mondjuk, ha a szimmetrikus része pozitív definit. A mátrix szimmetrikus része $A_+ = (A + A^T)/2$ és az antiszimmetrikus része $A_- = (A - A^T)/2$, $A = A_+ + A_-$. Vegyük észre, az antiszimmetrikus részhez tartozó belső szorzat $x^T A_- x$ mindig zérus.

Ha x-et e_i-nek választjuk, akkor a definicióból következik, hogy valós szimmetrikus pozitív definit mátrixra $a_{ii} > 0$ minden i-re, $x = e_i \pm e_j$ esetén pedig $a_{ii} + a_{jj} \pm 2a_{ij} > 0$-nak kell teljesülnie. Ezek az egyszerű feltételek néha hasznosak annak gyors eldöntésében, hogy a mátrix egyáltalán lehet-e pozitív definit. Például, ha a mátrix főátló-beli elemei mind 0-k és a főátlón kívüli elemek között vannak nemzérus elemek, akkor rögtön állítható, hogy a mátrix indefinit.

5.1.1 Tétel, elegendő feltétel pozitív definitségre.

Ha $A = V^T V$ alakban előállítható, ahol V oszlopai lineárisan függetlenek, akkor A pozitív definit.

Bizonyítás. A definició alapján minden nemzérus x-re $x^T Ax = x^T V^T Vx = \|Vx\|_2^2 > 0$ mert $Vx \neq 0$, ha V oszlopai lineárisan függetlenek. ■

5.1.2 Tétel, a pozitív definitség megőrződik az LU-felbontásban.

Pozitív definit A mátrix *LU*-felbontása megőrzi a pozitv definitséget, más szóval: minden lépés után a visszamaradó jobb alsó blokk pozitív definit marad. Az állítás blokk *LU*-felbontáskor is igaz.

Bizonyítás. Legyen A blokkos alakja

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \quad (A \mid A_{11}) = A_{22} - A_{21} A_{11}^{-1} A_{12},$$

ahol a blokk *LU*-felbontás egy lépése után visszamaradó blokk az $(A \mid A_{11})$ Schur-komplemens. Azt kell igazolni, hogy tetszőleges nemzérus x_2 vektorra $x_2^T (A \mid A_{11}) x_2 > 0$. Az állítást azzal bizonyítjuk, hogy megmutatjuk: létezik egy kiegészített $x^T = (x_1^T, x_2^T)$ vektor, amelyre $x^T Ax = x_2^T (A \mid A_{11}) x_2$. Ehhez válasszuk x_1-et úgy, hogy szorzáskor az első blokk-sor zérust adjon: $A_{11} x_1 + A_{12} x_2 = 0$. Ezzel $x_1 = -A_{11}^{-1} A_{12} x_2$ és $0 < x^T Ax = \begin{bmatrix} x_1^T & x_2^T \end{bmatrix} \begin{bmatrix} 0 \\ A_{21} x_1 + A_{22} x_2 \end{bmatrix} = x_2^T (A_{22} - A_{21} A_{11}^{-1} A_{12}) x_2$. ■

Megjegyzés. Ugyanígy látható be, hogy a felbontás során a pozitív szemidefinitség is megőrződik.

5.1.3 Tétel, pozitív szemidefinit mátrix felbonthatósága.

Ha A pozitív szemidefinit, akkor $A = LL^T$ alakban előállítható.

Bizonyítás. Láttuk, A főátlójában csak nemnegatív elemek lehetnek. Ha $a_{11} > 0$, akkor készítsük el a következő

$$A_2 = A - \frac{Ae_1 e_1^T A}{e_1^T A e_1}, \tag{5.1}$$

mátrixot, amelyről tudjuk, hogy az első sora és oszlopa zérus. Válasszuk L első oszlopának $Le_1 = Ae_1 / \sqrt{a_{11}}$ -et, ezzel $A_2 = A - Le_1 e_1^T L$.

Ha az első diagonálelem zérus, akkor ugyanazon sor és oszlop cseréjével mozgassunk egy nemzérus diagonálelemet az 1,1 pozicióba és ugyanígy járjunk el.

Folytassuk az eljárárást a megmaradó 1-gyel kisebb méretű jobb alsó blokkal mindaddig, ameddig találunk pozitív diagonálelemet. Minden lépésben az L mátrix egy újabb oszlopát nyerjük. Ha olyan helyzethez értünk, ahol a megmaradt jobb alsó blokkban minden diagonálelem zérus, akkor a teljes blokknak zérusnak kell lennie, mert ha nem így volna, akkor a megmaradó blokk indefnit volna az 5.1.1 Tétel előtt tett megjegyzés szerint és ez ellentmondana annak, hogy a szemidefinitség megmarad.

Vegyük észre, az alkalmazott sor-oszlop cserék a felbontást csak annyiban befolyásolják, hogy $P^T AP = LL^T$ -et kellett volna írnunk, - P permutáció mátrix -, de ez átrendezhető az $A = \tilde{L}\tilde{L}^T$ alakba, ahol $\tilde{L} = PL$. ■

Szimmetrikus, pozitív definit mátrixra az $A = LL^T$ felbontást *Cholesky-felbontásnak* nevezzük. Itt most L főátlójában nem 1-esek állnak, mert például $Le_1 = Ae_1 / \sqrt{a_{11}}$ első eleme $\sqrt{a_{11}}$. A Cholesky-felbontás hasonlóképp készíthető, mint az *LU*-felbontás, csak most a főelemből gyököt kell vonni, és azzal végig kell osztani a saját sort és oszlopot. A számítógépes algoritmusban kihasználható, hogy a felső háromszög részt nem kell számítani, ezzel a műveletigény nagyjából megfeleződik.

5.1.4 Példa Choleski-felbontásra

$$\begin{bmatrix} 4 & -2 & 2 \\ -2 & 10 & -7 \\ 2 & -7 & 21 \end{bmatrix} \to \begin{bmatrix} \boxed{2} & -1 & 1 \\ -1 & 9 & -6 \\ 1 & -6 & 20 \end{bmatrix} \to \begin{bmatrix} 2 & & \\ -1 & \boxed{3} & -2 \\ 1 & -2 & 16 \end{bmatrix} \to \begin{bmatrix} 2 & & \\ -1 & 3 & \\ 1 & -2 & \boxed{4} \end{bmatrix},$$

$$L = \begin{bmatrix} 2 & & \\ -1 & 3 & \\ 1 & -2 & 4 \end{bmatrix}, \quad L^T = \begin{bmatrix} 2 & -1 & 1 \\ & 3 & -2 \\ & & 4 \end{bmatrix}.$$

Látható, a diádok levonása ugyanolyan módon történik, mint az *LU*-felbontásban.

5.1.5 Feladatok.

- Legyen $A = LL^T$ egy Cholesky felbontás. Mennyi a műveletigénye $x^T Ax$ számításának, ha az eredeti mátrixot használjuk? Hogyan csökkenthető a műveletigény, ha az $x^T LL^T x$ alakot használjuk?
- A gyökvonást elkerülhetjük, ha az $A = LDL^T$ alakot használjuk, ahol L egységátlójú mátrix és D diagonálmátrix. Dolgozzuk ki ennek a felbontásnak a lépéseit! Ezt a módszert indefinit esetben is alkalmazhatjuk, ha nem adódik túlságosan kicsiny elem D-be.

5.2. Főátló-dominancia

Sorok szerint főátló-domináns vagy diagonál-dominánsnak nevezzük a mátrixot, ha minden sorban a nemdiagonális sorelemek abszolút összege kisebb, mint a főátlóbeli elem abszolút értéke:

$$|a_{ii}| > \left\| e_i^T (A - \mathrm{diag}(A)) \right\|_\infty .$$

Lényegében főátló-domináns a mátrix, ha *nem minden sorban* a $\geq$ jel is megengedett és ezek a sorok nemzérus sorok. Az oszlopok szerint *főátló-domináns* mátrixok értelmezése hasonló. Itt $\mathrm{diag}(A) = D$ a mátrix főátlójából készített diagonálmátrixot jelöli.

5.2.1 Tétel, a főátló-dominancia megmaradása.

Amennyiben az A mátrix főátló-domináns, az LU -felbontás végrehatása során a még fel nem bontott jobb alsó részben a főátló-dominancia megmarad. Másképpen: a Schur-komplemens megőrzi a főátló-dominanciát.

Bizonyítás. Az *LU*-felbontás első lépése után a mátrix első oszlopa az $a_{11}e_1$ vektorba megy át és a k -adik sorvektor:

$$e_k^T (I - (a_1 / a_{11} - e_1)e_1^T)A = (e_k^T A - \frac{a_{k1}}{a_{11}} e_1^T A)(I - e_1 e_1^T), \quad k > 1 ,$$

ahol a hozzáírt $I - e_1 e_1^T$ vetítőmátrix az amúgy is zérus első sorelemet nullázza, így változást nem okoz. Az $e_k^T A(I - e_1 e_1^T)$ sorvektor rendelkezik a főátló-dominancia tulajdonsággal, mert csak az első a_{k1} elemet hagytuk el. A levont vektor sornormája pedig

$$\left\| a_{k1} e_1^T A(I - e_1 e_1^T)/a_{11} \right\|_\infty = |a_{k1}| \left\| e_1^T A(I - e_1 e_1^T)/a_{11} \right\|_\infty < |a_{k1}| ,$$

ha $a_{k1} \neq 0$. Itt az átlóelemmel osztott első sor normája kisebb 1-nél (főátló-dominancia!) és ez szorozza a_{k1}-et. Tehát a kivett a_{k1} helyébe egy kisebb abszolút értékű elem kerül az abszolút sorösszeg számításakor, így a *k*-adik sor főátló-dominanciája nem romolhat. A további lépésekben a helyzet hasonló. ■

A tétel következménye, hogy főátló-domináns mátrixoknál az átlóelem mindig alkalmas főelemnek.

5.2.2 Feladatok. Mutassuk meg:

- A főátló-dominancia megmarad, ha a mátrixot balról nemszinguláris diagonálmátrixszal szorozzuk, vagy ha ugyanazt a két sort és oszlopot felcseréljük.
- Lényegében főátló-domináns mátrixok *LU*-felbontásakor a j-edik lépésben szigorú főátló-dominancia következik be a *k*-adik sorban, ha a j-edik sorban megvolt a szigorú főátló-dominancia és volt nemzérus $a_{jk}^{(j)}$, $j < k$ elem.
- Az oszlopok szerinti főátló-dominancia is öröklődik.

5.3. Két- és háromátlójú mátrixok

5.3.1 Speciális mátrixok

A kétátlójú vagy bidiagonális mátrixoknál csak a főátló és valamelyik mellette lévő átlóban vannak nemzérus elemek: $a_{ij} \neq 0$, $j - i \in \{0,1\}$, vagy $j - i \in \{0,-1\}$. Nevezetes képviselőjük a különbségképzés mátrixa:

$$K=\begin{bmatrix} 1 & & & \\ -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{bmatrix}, \quad K^{-1}=S=\begin{bmatrix} 1 & & & \\ 1 & 1 & & \\ \vdots & \vdots & \ddots & \\ 1 & 1 & \cdots & 1 \end{bmatrix}.$$

Inverze éppen az összegzésmátrixot adja. E két mátrix segítségével egyszerűen megadható a gyakran előforduló

$$T=\begin{bmatrix} 2 & -1 & & \\ -1 & 2 & \ddots & \\ & \ddots & \ddots & -1 \\ & & -1 & 2 \end{bmatrix} \tag{5.2}$$

mátrix inverze:

$$T^{-1}=\left(K+K^T\right)^{-1}=\left[K\left(S+S^T\right)K^T\right]^{-1}=K^{-T}\left(I+ee^T\right)^{-1}K^{-1}=K^{-T}\left(I-\frac{ee^T}{1+n}\right)K^{-1}, \tag{5.3}$$

ahol e a csupa 1-esből álló vektor. A $T^{-1}x$ vektor előállítása így $4n$ flop műveletet igényel.

5.3.2 Főátló-domináns háromátlós mátrix

Láttuk, ebben az esetben nem kell a főelemválasztással foglalkozni az *LU*-felbontás során. Ha a felbontást a mutatott módon hajtjuk végre, akkor a lineáris egyenletrendszer megoldásának műveletigénye lényegében $9n$ flop. Háromátlós esetben van azonban két módszer is, amellyel $8n$ flop művelettel célba érünk. A következőkben ezeket ismertetjük. Az első módszert hívhatjuk gyors *LU*-felbontásnak. Vegyük fel a háromátlós mátrixú egyenletrendszert a következő alakban:

$$Hx=\begin{bmatrix} d_1 & c_1 & & \\ a_1 & d_2 & \ddots & \\ & \ddots & \ddots & c_{n-1} \\ & & a_{n-1} & d_n \end{bmatrix}x=\begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}. \tag{5.4}$$

Az LU -felbontás első lépése csak a második sort változtatja meg:

$$\begin{bmatrix} a_1/d_1 & d_2-a_1c_1/d_1 & c_2 & \dots & 0 \end{bmatrix}x=b_2-b_1a_1/d_1.$$

Eredményül kaptunk egy 1-gyel kisebb méretű háromátlójú mátrixot, amire az eljárást megismételhetjük. Tovább folytatva végül a főelemek és jobboldalak a következők lesznek:

$$\begin{aligned} &d_1'=d_1;\ d_i'=d_i-a_{i-1}c_{i-1}/d_{i-1}', \quad i=2,3,\dots,n, \\ &b_1'=b_1;\ b_i'=b_i-a_{i-1}b_{i-1}'/d_{i-1}', \quad i=2,3,\dots,n. \end{aligned} \tag{5.5}$$

Most a felbontás U mátrixa felső bidiagonális - kétátlójú mátrix - és a megoldandó egyenletrendszer:

$$\begin{bmatrix} d_1 & c_1 & & \\ 0 & d_2' & \ddots & \\ & \ddots & \ddots & c_{n-1} \\ & & 0 & d_n' \end{bmatrix}x=\begin{bmatrix} b_1 \\ b_2' \\ \vdots \\ b_n' \end{bmatrix}, \quad x_n=b_n'/d_n';\ x_i=(b_i'-c_ix_{i+1})/d_i',\ i=n-1,n-2,\dots,1.$$

Látjuk: az *L* mátrix nem is kell a megoldáshoz, másrészt (5.5) mindkét sorában szerepel a_{i-1}/d_{i-1}', amit elegendő egyszer előállítani. Ezzel a megoldási algoritmus:

Kezdés: $d'_1 = d_1, \quad b'_1 = b_1.$
$i = 2,3,\ldots,n$-re
$s := a_{i-1} / d'_{i-1}; \quad d'_i := d_i - c_{i-1} * s; \qquad b'_i := b_i - b'_{i-1} * s.$
$x_n := b'_n / d'_n;$
$i = n-1, n-2,\ldots,1$-re
$x_i := (b'_i - c_i * x_{i+1}) / d'_i.$

A másik módszer a megoldás második fázisában érvényes rekurziót veszi alapul:

$$x_i = f_i - g_i x_{i+1}.$$

Az egyenletrendszer első sorából $x_1 = (b_1 - c_1 x_2) / d_1$, ezzel $f_1 = b_1 / d_1$ és $g_1 = c_1 / d_1$. Ezután az i-edik sorba helyettesítve x_{i-1} kifejezését

$$a_{i-1}(f_{i-1} - g_{i-1} x_i) + d_i x_i + c_i x_{i+1} = b_i,$$

innen

$$x_i = \frac{b_i - a_{i-1} f_{i-1}}{d_i - a_{i-1} g_{i-1}} - \frac{c_i}{d_i - a_{i-1} g_{i-1}} x_{i+1} = f_i - g_i x_{i+1},$$

ahonnan f_i és g_i előállítása kiolvasható. Ezzel az „üldözéses” vagy „passzázs” algoritmus:

Kezdés: $f_1 = b_1 / d_1, \quad g_1 := c_1 / d_1.$
$i = 2,3,\ldots,n$-re
$s := d_i - a_{i-1} g_{i-1}; \quad f_i := (b_i - a_{i-1} f_{i-1}) / s; \qquad g_i := c_i / s.$
$x_n := f_n;$
$i = n-1, n-2,\ldots,1$-re
$x_i := f_i - g_i * x_{i+1}.$

5.3.3 Feladat

- Ha új jobboldal vektort kapunk, milyen részletszámításokat őrizzünk meg és mit számítsunk újra mindkét algoritmusban?
- Igazoljuk, hogy az (5.2)-ben szereplő háromátlós mátrix pozitív definit, mert van LL^T-felbontása.