Direktne metode za rješavanje linearnih sustava

Milobara, Matej

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

Matej Milobara

Matični broj: 42 811/14–R

Studij: Informacijski sustavi

DIREKTNE METODE ZA RJEŠAVANJE LINEARNIH SUSTAVA

ZAVRŠNI RAD

Mentor:

Dr. sc. Bojan Žugec

Varaždin, rujan 2018.

Matej Milobara

Izjava o izvornosti

Izjavljujem da je moj završni rad izvorni rezultat mojeg rada te da se u izradi istoga nisam koristio drugim izvorima osim onima koji su u njemu navedeni. Za izradu rada su korištene etički prikladne i prihvatljive metode i tehnike rada.

Autor/ica potvrdio/la prihvaćanjem odredbi u sustavu FOI-radovi

Sažetak

U ovom radu će se navesti primjeri direktnih metoda za rješavanje linearnih sustava. Na početku se opisuju linearni sustavi u općenitom smislu gdje će biti objašnjeni linearni sustavi, njihovi problemi i sl. Nakon toga prelazi se na objašnjavanje pojma direktnih metoda kako bi se shvatila razlika između direktnih te iterativnih metoda. Na tom dijelu će se naznačiti koje su mane i prednosti tih metoda što će ujedno dati uvod u sljedeće poglavlje, a to je usporedba LU faktorizacije s različitim metodama. Kada je objašnjena LU faktorizacija te su napravljene sve usporedbe različitih metoda LU faktorizacije bit će objašnjeno kakve su to pozitivno definitne matrice te kako ih dobiti. Nakon tog dijela bit će objašnjene loše uvjetovane matrice te navedeni razlozi zašto one nisu stabilne za numeričke algoritme gdje će rad dati i konkretan primjer. Također, kraj rada označit će primjer složenosti već gotovog algoritma koji će biti objašnjen te prezentiran na način da se on objasni u potpunosti.

Ključne riječi: linearni, sustav, faktorizacija, matrice, algoritmi, metode

Sadržaj

1. Uvod 1

2. Linearni sustavi 2

2.1. Linearne jednadžbe 2

2.2. Sustavi linearnih jednadžbi 2

2.3. Oblici sustava linearnih jednadžbi 3

2.4. Direktne i iterativne metode 6

2.4.1. Gaussove eliminacije 8

2.4.2. LU faktorizacija 11

2.4.3. Pozitivno definitne matrice 19

3. Opis algoritma 21

4. Zaključak 26

Popis literature 27

Popis slika 28

1. Uvod

U uvodu ovog završnog rada ukratko će se objasniti kakve su to direktne metode. Za početak potrebno je prvo ponoviti ili se prvi puta upoznati s pojmom linearnih sustava odnosno linearnih jednadžbi, od onih osnovnih pa sve do onih kompliciranijih. Nakon toga će se napraviti kratka usporedba te podjela između direktnih te iterativnih metoda koja će biti jako bitna za daljnji nastavak odnosno razvoj teme ovog rada.

Nakon raspodjele na direktne i iterativne metode rad se okreće sve više praktičnom prikazivanju metoda koje se nalaze u centru pažnje ovog rada pa će tako biti detaljno objašnjena metoda Gaussove eliminacije te LU faktorizacija i sl. Te metode će naravno biti prikazane kroz neke već izrađene primjere kako bi se bolje shvatila njihova svrha, također sve metode će biti prikazane i kroz svoje pripadajuće algoritme kako bi se mogle shvatiti u računalnom smislu. Nakon toga kratko će se razmotriti pozitivna definitnost, loše uvjetovane matrice te će se sve skupa zaključiti.

2. Linearni sustavi

Ovo poglavlje odgovorit će na nekoliko bitnih pitanja koja će biti ključna za cjelokupni razvoj priče rada kao cjeline. Pitanja na koja će se odgovoriti u ovome poglavlju su: Što su to zapravo sustavi linearnih jednadžbi? Koje su metode za rješavanje takvih sustava te koje su prednosti i nedostaci tih sustava? No prije toga će se u podnaslovima napraviti kratki uvod u linearne jednadžbe kao sastavni dio teme ovoga rada.

2.1. Linearne jednadžbe

Za jednadžbu $ax = b$ se kaže da je ona linearna jednadžba s jednom nepoznanicom te da se može riješiti na sljedeći način:

$$ax = b / : a$$

$$x = \frac{b}{a}$$

U prvom koraku može se primijetiti da se cijela jednadžba dijeli s a kako bi nepoznanica x ostala sama na lijevoj strani. Na desnoj strani se zatim dobije da je rješenje ove jednadžbe zapravo razlomak $\frac{b}{a}$ gdje mora postojati uvjet $a \neq 0$ jer x ne može biti nula nego mora biti realno rješenje. Postoji i slučaj kada je $a = 0, b \neq 0$ pa onda ta jednadžba neće imati rješenje, odnosno ako su $a = b = 0$ onda ta jednadžba ima beskonačno mnogo rješenja. Nadalje, linearna jednadžba s dvije nepoznanice ima oblik $ax + by = c$ gdje su $a, b, c \in \mathbb{R}$, a x, y nepoznanice. Rješenje te jednadžbe je sadržano u paru realnih brojeva koji zadovoljavaju tu jednadžbu. Ako se dogodi slučaj $(a^2 + b^2) \neq 0$ onda ta jednadžba ima beskonačno mnogo rješenja te sva ta rješenja leže na pravcu $ax + by - c = 0$. Posljednji sustav koji će ovaj rad razmotriti je sustav jednadžbi s tri nepoznanice koji je oblika $ax + by + cz = d$ gdje su $a, b, c, d \in \mathbb{R}$, a x, y, z nepoznanice. Za razliku od slučaja kod sustava jednadžbi s dvije nepoznanice, rješenje je svaka uređena trojka realnih brojeva koja zadovoljava tu jednadžbu. Ako se dogodi slučaj da je $a^2 + b^2 + c^2 \neq 0$ onda ta jednadžba ima beskonačno mnogo rješenja.

2.2. Sustavi linearnih jednadžbi

Sustavi linearnih jednadžbi su jedan od osnovnih problema numeričke matematike, točnije samo njihovo rješavanje je jedan od osnovnih problema. Jedan od najbitnijih elemenata o kojemu će se govoriti su matrice sustava koje se sastoje od elemenata koji se nazivaju koeficijentima. Takve matrice mogu se označiti $A = [a_{ij}] \in \mathbb{R}^{n \times n};\ i, j = 1, \ldots, n$, one se nazivaju matricama sustava, a svaki element te matrice naziva se koeficijentom matrice.

Prvi i osnovni matrični zapis kada govorimo o matricama sustava je $Ax = b$, gdje je A matrica koja je objašnjena u nekoliko rečenica prije, a b označava slobodni koeficijent te se prikazuje na sljedeći način $b = [b_i] \in R^n;\ i = 1, \ldots, n$. Treći element je nepoznanica x koja se označava kao $x = [x_i] \in \mathbb{R}^n;\ i = 1, \ldots, n$.

Rješenje ovog sustava je nepoznanica x koju dobijemo tako da cjelokupni sustav pomnožimo s lijeve strane inverzom matrice A.

$$A^{-1}/Ax = b$$

$$A^{-1}Ax = A^{-1}b$$

$$Ix = A^{-1}b$$

$$x = A^{-1}b$$

U iznad prikazanom redoslijedu se jasno može vidjeti na koji način se dođe do rješenja ovog sustava, sve se množi s inverzom matrice A. S lijeve strane dobijemo jediničnu matricu I, a s desne strane inverz matrice A ostaje pomnožen vektorom slobodnih koeficijenata b. Također, moguće je zaključiti da rješenje jednadžbe $x = A^{-1}b$ postoji te da je jedinstveno. To sve znamo jer postoje realni brojevi koji se množe između inverzne matrice te slobodnog koeficijenta b.

2.3. Oblici sustava linearnih jednadžbi

U ovom podnaslovu bit će kroz tri slučaja objašnjena tematika naslova te detaljno razloženo objašnjenje svih detalja koji se vežu uz te tvrdnje. Tri slučaja odnose se na sljedeći zadatak: pomoću elementarnih aritmetičkih operacija potrebno je pronaći sve trojke realnih brojeva x_1, x_2, x_3 istovremeno zadovoljavajući sljedeće jednadžbe:

Prvi slučaj:

$$x_1 + x_3 = 0$$

$$x_2 + x_3 = 1$$

$$x_1 + 2x_2 + x_3 = 1$$

Drugi slučaj:

$$x_1 + x_3 = 0$$

$$x_2 - x_3 = 1$$

$$x_1 + 2x_2 - x_3 = 1$$

Treći slučaj:

$$x_1 + x_3 = 0$$

$$x_2 - x_3 = 1$$

$$x_1 + 2x_2 - x_3 = 2$$

Prije nego se ovaj sustav problema krene rješavati moguće je uočiti da se ova tri slučaja razlikuju u samo nekoliko detalja jednadžbi pa se tako prvi slučaj razlikuje u dva predznaka u drugoj i trećoj jednadžbi, a treći se slučaj razlikuje od drugog u samo jednoj brojci s desne strane treće jednadžbe. Kako bi ovi problemi mogli biti riješeni potrebno je koristiti osnovne aritmetičke operacije koje moraju imati neki smisleni redoslijed pri svojemu izvršavanju. Također prije nego se gornja tri slučaja krenu rješavati potrebno je uvesti tri elementarne transformacije na koje će se rješavanje gornjih problema oslanjati u cjelokupnom procesu. Te tri transformacije su sljedeće:

1. Zamjena bilo koje dvije jednadžbe sustava
2. Množenje neke jednadžbe sustava brojem koji nije nula
3. Dodavanje jednoj jednadžbi druge jednadžbe koja je pomnožena nekim brojem

Moguće je primjetiti da su ove transformacije koje se uvijek standardno koriste kod računanja sustava jednadžbi, npr. sustav dvije jednadžbe s dvije nepoznanice. Sada će biti prikazana njihova primjena na konkretno rješavanje prethodno navedena tri slučaja. Kod prvog radi se zamjena treće jednadžbe razlikom treće i prve jednadžbe.

$$x_1 + x_3 = 0$$

$$x_2 - x_3 = 1$$

$$2x_2 = 1$$

Moguće je primijetiti da se sada u trećoj jednadžbi lako dobije rješenje za x_2 koje će biti $x_2 = \frac{1}{2}$. Ako se x_2 uvrsti u drugu jednadžbu dobije se:

$$\frac{1}{2} - x_3 = 1$$

$$-x_3 = 1 - \frac{1}{2} / : (-1)$$

$$x_3 = \frac{-1}{2}$$

Te na kraju x_3 se uvrsti u prvu jednadžbu pa se dobije:

$$x_1 - \frac{-1}{2} = 1$$

$$x_1 + \frac{1}{2} = 1$$

$$x_1 = 1 - \frac{1}{2}$$

$$x_1 = \frac{1}{2}$$

Rješenja ovog problema odnosno sustava jednadžbi su: $x_1 = \frac{1}{2}, x_2 = \frac{1}{2}, x_3 = -\frac{1}{2}$. Iz ovog rješenja moguće je izvesti zaključak da postoji jedinstvena trojka brojeva koja je rješenje ovog problema.

Drugi slučaj moguće je započeti zamjenom treće jednadžbe s razlikom te iste treće jednadžbe te prve jednadžbe:

$$x_1 + x_3 = 0$$

$$x_2 - x_3 = 1$$

$$2x_2 - 2x_3 = 1$$

Nadalje moguće je zamijeniti treću jednadžbu sustava iznad s jednadžbom koja se dobije tako da se od treće jednadžbe oduzme druga pomnožena s -2. Ovdje je moguće primijetiti primjenu treće transformacije:

$$x_1 + x_3 = 0$$

$$x_2 - x_3 = 1$$

$$0 = -1$$

Prema sustavu iznad može se zaključiti da rješenje sustava ne postoji jer smo dobili kontradikciju $0 = -1$. Treći slučaj može se odmah krenuti raditi tako da se dva puta primjeni treća transformacija prema kojoj će se prvo od treće jednadžbe oduzeti prva, a onda će se oduzeti i druga jednadžba pomnožena s 2.

$$x_1 + x_3 = 0$$

$$x_2 - x_3 = 1$$

$$0 = 0$$

Iznad je moguće uvidjeti da se dobije sustav od dvije jednadžbe s tri nepoznanice:

$$x_1 + x_3 = 0$$

$$x_2 - x_3 = 1$$

Ovdje je moguće zaključiti da ovaj sustav nema jedinstveno rješenje odnosno da ih ima beskonačno mnogo jer je:

$$x_1 = -x_3$$

$$x_2 = 1 + x_3$$

Točnije:

$$x_1 = -\beta$$

$$x_2 = 1 + \beta$$

$$x_3 = \beta$$

Na kraju je moguće zaključiti da u sva tri slučaja imamo različita rješenja. U prvom slučaju dobiveno je jedno jedinstveno rješenje, u drugom ne postoji niti jedno , a u trećem ih ima beskonačno mnogo. Sustavi su rješavani prema trima elementarnim transformacijama koje su navedene netom prije nego se krenulo u njihovo rješavanje te nam ta pravila daju algoritam za rješavanje sustava triju linearnih jednadžbi s tri nepoznanice. Takvo rješavanje sustava naziva se Gaussova metoda eliminacija.

2.4. Direktne i iterativne metode

U računanju odnosno rješavanju sustava linearnih jednadžbi razlikuju se dvije vrste metoda za njihovo rješavanje, a to su direktne i iterativne metode. Pod direktne metode se svrstavaju:

- Cramerovo pravilo
- Gaussove eliminacija
- LR faktorizacija
- Gauss-Jordanova metoda
- Thomasov algoritam itd.

Pod iterativne metode rješavanja sustava linearnih jednadžbi ubrajaju se:

- Jacobijeva metoda
- Gauss-Seidelova metoda

U nastavku teksta će biti prikazan po jedan primjer iz svake vrste metoda te će se u kratkim crtama objasniti razlika i navesti glavne prednosti i nedostaci između direktnih i iterativnih metoda. Prva vrsta za koju će se prikazati primjer su direktne metode, a izabrani primjer će biti Gaussova eliminacija. Osnova metode Gaussove eliminacije se svodi na pretvaranje početne matrice sustava u trokutastu formu pomoću elementarnih transformacija. Elementarne transformacije su:

- Množenje jednadžbe brojem različitim od nula
- Zbrajanje ili oduzimanje jednadžbe drugim jednadžbama
- Zamjena poretka jednadžbi

Samo rješenje sustava prema metodi Gaussovih eliminacija dobiva se supstitucijom prema unatrag.

$$a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1$$

$$a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2$$

$$\ldots$$

$$a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nm}x_n = b_m$$

Gdje se matrica sustava na prethodnoj stranici svodi na trokutastu formu koja izgleda ovako:

$$a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1$$

$$a_{22}x_2 + \cdots + a_{2n}x_n = b_2$$

$$\ldots$$

$$a_{mn}x_n = b_m$$

Naposljetku je još potrebno navesti kako izgleda supstitucija prema unatrag.

$$x_n = \frac{b_m}{a_{mn}}$$

$$x_i = \frac{1}{a_{ii}}(b_i - \sum_{j=i+1}^{n} a_{ij}x_j) \rightarrow i = (n-1), \ldots, 1$$

Ovo je bio primjer metode koja se može predstaviti ispred direktnih metoda. U nastavku će biti objašnjene iterativne metode te prednosti i nedostaci odnosno razlike između direktnih i iterativnih metoda.

Budući da je broj potrebnih računskih operacija kod metoda dekompozicije reda veličine n^3, za velike matrice s puno elemenata koji iščezavaju direktne metode ne mogu se preporučiti. Osim toga, treba ekonomizirati i s brojem elemenata matrice za koje treba rezervirati mjesto u memoriji računala. Takve situacije javljaju se primjerice kod proučavanja električnih mreža, kod velikih ekonometrijskih modela nacionalne privrede, kod spline-interpolacija, kod rješavanja rubnih problema za obične i parcijalne diferencijalne jednadžbe, itd. Upravo se iterativne metode koriste u takvim situacijama. **[5]**

Iterativne metode same po sebi ne traže puno računalne memorije jer izvršavaju dosta manji broj matematičkih operacija u odnosu na direktne metode. Jedna od glavnih karakteristika iterativnih metoda je da one počinju od nekog pretpostavljenog rješenja te se onda sukcesivnim rješavanjem jednadžbi dobiva sve bolje i bolje rješenje do unaprijed zadane točnosti.

2.4.1. Gaussove eliminacije

U prethodnom dijelu kroz kratke crte objašnjena je metoda Gaussove eliminacija no sada će ta metoda biti prikazana na primjeru te detaljnije, odnosno preciznije objašnjena jer ovaj dio će biti krucijalan za prikaz kasnijih usporedbi između različitih metoda. Primjer jednog sustava s dvije jednadžbe i dvije nepoznanice je sljedeći:

$$3x_1 + 3x_2 = 3$$

$$2x_1 - x_2 = 2$$

Prvi korak koji se može povući je dijeljenje cijele prve jednadžbe s 3 jer sve nepoznanice imaju 3 kao zajedničku vrijednost.

$$3x_1 + 3x_2 = 3/:3$$

$$2x_1 - x_2 = 2$$

Dobije se jednadžba iz koje se lako može izlučiti jedna od nepoznanica kako bi se s desne strane dobio njen izračun.

$$x_1 + x_2 = 1 \quad \rightarrow \quad x_1 = 1 - x_2$$

$$2x_1 - x_2 = 2$$

Nakon što je jedna nepoznanica izlučena ta nepoznanica se može uvrstiti u drugu jednadžbu kako bi se konačno dobila vrijednost druge nepoznanice.

$$x_1 + x_2 = 1 \quad \rightarrow \quad x_1 = 1 - x_2$$

$$2(1 - x_2) - x_2 = 2$$

Izračunava se druga jednadžba iz koje će se množenjem dobiti da je njezina vrijednost nula.

$$x_1 + x_2 = 1 \quad \rightarrow \quad x_1 = 1 - x_2$$

$$2 - 2x_2 - x_2 = 2 \quad \rightarrow \quad -3x_2 = 0/:(-3) \quad \rightarrow \quad x_2 = 0$$

Nakon što se dobilo da je vrijednost druge nepoznanice nula ta vrijednost se uvrštava natrag u prvu jednadžbu kako bi se dobila vrijednost prve nepoznanice.

$$x_1 = 1 - x_2 \quad \rightarrow \quad x_1 = 1 - 0 \quad \rightarrow \quad x_1 = 1$$

$$x_2 = 0$$

Vrijednost prve nepoznanice je jedan. Sada se može reći da je ovaj sustav riješen Gaussovom metodom eliminacije jer eliminacijom jedne nepoznanice iz druge jednadžbe uz pomoć prve, dobije se rješenje. Nakon kratkog ponavljanja Gaussovog sustava koji se koristi eliminacijom nepoznanica sada će se ukratko prikazati matrični zapis metode eliminacija kako bi se sljedeći dio gradiva, koji uključuje razmatranje trokutastih sustava, LU faktorizacija i sl., mogao lakše shvatiti. Za primjer, sustav koji je napravljen u prethodnom bit će prikazan te matrično izračunat:

$$3x_1 + 3x_2 = 3$$

$$2x_1 - x_2 = 2$$

Matrični zapis linearnog sustava od tri jednadžbe s tri nepoznanice koji je dan za primjer iznad izgleda:

$$\begin{bmatrix} 3 & 3 \\ 2 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 3 \\ 2 \end{bmatrix}$$

Također sustav je moguće zapisati i uz pomoć proširene matrice koja izgleda ovako:

$$\left[\begin{array}{cc:c} 3 & 3 & 3 \\ 2 & -1 & 2 \end{array}\right]$$

Kako bi se dobilo rješenje ovakvog sustava potrebno je prema već definiranim pravilima svesti ovaj oblik na ešalon formu. Prvi korak će biti da se cijeli prvi redak podijeli s brojem tri.

$$\left[\begin{array}{cc:c} 3 & 3 & 3 \\ 2 & -1 & 2 \end{array}\right] \begin{array}{l} \mid : 3 \\ \end{array} \rightarrow \left[\begin{array}{cc:c} 1 & 1 & 1 \\ 2 & -1 & 2 \end{array}\right]$$

Nakon toga odabire se prva jedinica u prvom retku te se sve vrijednosti ispod nje eliminiraju. Kako bi se eliminirale vrijednosti prvo se cijeli prvi redak pomnoži s minus dva te se doda na drugi redak nakon čega su sve vrijednosti ispod odabrane jedinice poništene. Zatim je moguće primijetiti da se može podijeliti i broj minus tri u drugome retku jer je to jedina vrijednost koja je u njemu preostala.

$$\left[\begin{array}{cc:c} 1 & 1 & 1 \\ 2 & -1 & 2 \end{array}\right] \begin{array}{l} \mid \cdot (-2) \\ \longleftarrow + \end{array} \rightarrow \left[\begin{array}{cc:c} 1 & 1 & 1 \\ 0 & -3 & 0 \end{array}\right] \begin{array}{l} \\ \mid : (-3) \end{array} \rightarrow \left[\begin{array}{cc:c} 1 & 1 & 1 \\ 0 & 1 & 0 \end{array}\right]$$

Sada se odabire vrijednost koja je ostala u drugome retku te se uz pomoć nje eliminiraju sve vrijednosti iznad.

$$\left[\begin{array}{cc:c} 1 & 1 & 1 \\ 0 & 1 & 0 \end{array}\right] \begin{array}{l} \longleftarrow + \\ \mid \cdot (-1) \end{array} \rightarrow \left[\begin{array}{cc:c} 1 & 0 & 1 \\ 0 & 1 & 0 \end{array}\right]$$

Moguće je uočiti da je sustav sveden na ešalon formu te da su dobivena sljedeća rješenja sustava:

$$x_1 = 1$$

$$x_2 = 0$$

Zapis sustava uz pomoću matrica je prirodan način zapisivanja sustava jer se sustavi pohranjeni u računalu zapisuju upravo u matričnom formatu. Kada se rješava nekoliko linearnih sustava istom matricom, ali više različitih desnih strana, puno je praktičnije koristiti se matričnim zapisom iz mnogo različitih razloga. Sljedeći dio koji će biti objašnjen su trokutasti sustavi koji se još mogu nazvati i trokutaste matrice. Postoje dva oblika trokutastih matrica. To su gornje trokutaste matrice te donje trokutaste matrice.

Kvadratna je matrica gornja trokutasta ako su svi njeni elementi ispod dijagonale jednaki nuli. **[9]**

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ 0 & a_{22} & a_{23} & \dots & a_{2n} \\ 0 & 0 & a_{33} & \dots & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & a_{nn} \end{bmatrix}$$

Kvadratna je matrica donja trokutasta ako su svi njeni elementi iznad dijagonale jednaki nuli. **[9]**

$$\begin{bmatrix} a_{11} & 0 & 0 & \dots & 0 \\ a_{21} & a_{22} & 0 & \dots & 0 \\ a_{31} & a_{32} & a_{33} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{13} & a_{23} & a_{n3} & \dots & a_{nn} \end{bmatrix}$$

Ovi sustavi prilično olakšavaju posao kada imamo više jednadžbi s više nepoznanica jer dopuštaju lakše uočavanje puta kojim se može brže doći do konačnog rješenja. Oba oblika trokutastih matrica se mogu rješavati kroz algoritme. Algoritam za rješavanje linearnog sustava jednadžbi $Lx = b$ s regularnom donjetrokutastom matricom $L \in \mathbb{R}^{n \times n}$.

$$x_1 = \frac{b_1}{l_{11}};$$

$$\text{for } i = 2, \cdots, n \left\{ x_i = \frac{1}{l_{ii}}(b_i - \sum_{j=1}^{i-1} l_{ij}x_j) \right\}$$

Prebrojimo li operacije u gornjem algoritmu vidimo da imamo n dijeljenja, $1+2+\cdots+n = n(n+1)/2$ množenjem i isto toliko zbrajanja (i oduzimanja). Dakle, ukupna složenost je $O(n^2)$, što je bolje od predviđene za n-dimenzionalni sustav linearnih jednadžbi. [8]

Algoritam za rješavanje linearnog sustava jednadžbi $Ux = b$ s regularnom gornjetrokutastom matricom $U \in \mathbb{R}^{n \times n}$.

$$x_n = \frac{b_n}{u_{nn}};$$

$$\text{for } i = n-1, \cdots, 1 \left\{ x_i = \frac{1}{u_{ii}}(b_i - \sum_{j=i+1}^{n} u_{ij}x_j) \right\}$$

I u ovom slučaju je ukupna složenost $O(n^2)$, što je bolje od predviđene za n-dimenzionalni sustav linearnih jednadžbi. **[8]**

2.4.2. LU faktorizacija

Do sada je već prikazana te objašnjena priča o formuli $Ax = b$ gdje je A kvadratna matrica koja ima svoju inverznu matricu A^{-1} te koja množenjem sustava $Ax = b$ s tom inverznom matricom kreira jedinstveno rješenje $x = A^{-1}b$. Prema određenim izvorima ovakvo rješavanje problema nije uobičajeno u praksi jer zahtjeva traženje inverza matrice te slobodnog koeficijenta i matrice. Upravo u toj istoj praksi nije uobičajeno vidjeti primjenu metode koja je objašnjena u prethodnom dijelu, a to je metoda Gaussove eliminacije. Metoda koja se zapravo najčešće sreće u praksi je metoda LU faktorizacije iz nekoliko razloga. Prvi razlog je jer metoda ima manji broj operacija nego metoda Gaussove eliminacije te ima veliku prednost jer ne transformira slobodni koeficijent b kad i matrica, tako da se može mijenjati više slobodnih koeficijenata b. Kako bi LU faktorizacija postojala za neku matricu $A \in \mathbb{R}^{n \times n}$ za tu matricu mora postojati donja trokutasta matrica L i gornja trokutasta matrica U tako da vrijedi $A = LU$. Bitno je napomenuti da LU faktorizacija ne postoji u svim slučajevima odnosno za svaku matricu te da u slučaju ako postoji sustav se piše na sljedeći način:

$$Ax = b \iff LUx = b$$

Točnije može se napisati i u obliku koji se svodi na rješavanje dva sustava s trokutastim matricama:

$$Ly = b$$

$$Ux = y$$

Nadalje kako bi se bolje razumio ovaj dio o LU faktorizaciji potrebno je naglasiti da imamo dvije vrste LU faktorizacije, a to je s pivotiranjem i bez pivotiranja. Sada će obje metode LU faktorizacije biti prikazane na primjeru jednog linearnog sustava. Zadan je sljedeći sustav:

$$x_1 - 2x_2 - x_3 = 1$$

$$-6x_1 + x_3 + x_4 = 1$$

$$x_1 + 6x_2 - 4x_4 = 0$$

$$2x_1 - x_2 + x_3 + x_4 = 0$$

Prvi način na koji će ovaj slučaj biti riješen je LU faktorizacijom bez pivotiranja:

$$\begin{bmatrix} 1 & -2 & -1 & 0 \\ -6 & 0 & 1 & 1 \\ 1 & 6 & 0 & -4 \\ 2 & -1 & 1 & 1 \end{bmatrix}$$

Kao što se može primijetiti u prvom koraku iznad, linearni sustav je prikazan u matričnom obliku. S obzirom da se na prvoj poziciji nalazi jedinica, potrebno je sve ostale vrijednosti ispod te jedinice poništiti tako da se ona pomnoži s određenom vrijednošću te doda na jednadžbu ispod.

$$\begin{bmatrix}1&-2&-1&0\\-6&0&1&1\\1&6&0&-4\\2&-1&1&1\end{bmatrix}\begin{array}{l}\mid\cdot 6 \qquad \mid\cdot(-1) \qquad \mid\cdot(-2)\\ \longleftarrow + \\ \longleftarrow \qquad\qquad + \\ \longleftarrow \qquad\qquad\qquad\qquad + \end{array}$$

Iznad navedeni prikaz može se napisati i kao množenje matrica te se tako vidi kako se uopće dobije trokutasta matrica.

$$\begin{bmatrix}1&0&0&0\\6&1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix}\begin{bmatrix}1&-2&-1&0\\-6&0&1&1\\1&6&0&-4\\2&-1&1&1\end{bmatrix}=\begin{bmatrix}1&-2&-1&0\\0&-12&-5&1\\1&6&0&-4\\2&-1&1&1\end{bmatrix}$$

$$\begin{bmatrix}1&0&0&0\\0&1&0&0\\-1&0&1&0\\0&0&0&1\end{bmatrix}\begin{bmatrix}1&-2&-1&0\\0&-12&-5&1\\1&6&0&-4\\2&-1&1&1\end{bmatrix}=\begin{bmatrix}1&-2&-1&0\\0&-12&-5&1\\0&8&1&-4\\2&-1&1&1\end{bmatrix}$$

$$\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\-2&0&0&1\end{bmatrix}\begin{bmatrix}1&-2&-1&0\\0&-12&-5&1\\0&8&1&-4\\2&-1&1&1\end{bmatrix}=\begin{bmatrix}1&-2&-1&0\\0&-12&-5&1\\0&8&1&-4\\0&3&3&1\end{bmatrix}$$

Sljedeći korak će poništiti sve ispod pozicije 22:

$$\begin{bmatrix}1&-2&-1&0\\0&-12&-5&1\\0&8&1&-4\\0&3&3&1\end{bmatrix}\begin{array}{l}\\ \mid\cdot\frac{2}{3} \qquad \mid\cdot\frac{1}{4}\\ \longleftarrow + \\ \longleftarrow \qquad\qquad + \end{array}$$

$$\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&\frac{2}{3}&1&0\\0&0&0&1\end{bmatrix}\begin{bmatrix}1&-2&-1&0\\0&-12&-5&1\\0&8&1&-4\\0&3&3&1\end{bmatrix}=\begin{bmatrix}1&-2&-1&0\\0&-12&-5&1\\0&0&\frac{-7}{3}&\frac{-10}{3}\\0&3&3&1\end{bmatrix}$$

$$\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&\frac{1}{4}&0&1\end{bmatrix}\begin{bmatrix}1&-2&-1&0\\0&-12&-5&1\\0&0&\frac{-7}{3}&\frac{-10}{3}\\0&3&3&1\end{bmatrix}=\begin{bmatrix}1&-2&-1&0\\0&-12&-5&1\\0&0&\frac{-7}{3}&\frac{-10}{3}\\0&0&\frac{7}{4}&\frac{5}{4}\end{bmatrix}$$

Prirodno sljedeća pozicija ispod koje će biti poništene sve vrijednosti je pozicija 33:

$$\begin{bmatrix} 1 & -2 & -1 & 0 \\ 0 & -12 & -5 & 1 \\ 0 & 0 & \frac{-7}{3} & \frac{-10}{3} \\ 0 & 0 & \frac{7}{4} & \frac{5}{4} \end{bmatrix} \begin{array}{l} \\ \\ \mid \cdot \frac{3}{4} \\ \longleftarrow + \end{array}$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{3}{4} & 1 \end{bmatrix} \begin{bmatrix} 1 & -2 & -1 & 0 \\ 0 & -12 & -5 & 1 \\ 0 & 0 & \frac{-7}{3} & \frac{-10}{3} \\ 0 & 0 & \frac{7}{4} & \frac{5}{4} \end{bmatrix} = \begin{bmatrix} 1 & -2 & -1 & 0 \\ 0 & -12 & -5 & 1 \\ 0 & 0 & \frac{-7}{3} & \frac{-10}{3} \\ 0 & 0 & 0 & \frac{-5}{4} \end{bmatrix}$$

Nakon posljednjeg koraka dobije se matrica U, a matrica L umnoškom svih prethodnih matrica koje imaju jedinice na glavnoj dijagonali.

$$L = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -6 & 1 & 0 & 0 \\ 1 & \frac{-2}{3} & 1 & 0 \\ 2 & \frac{-1}{4} & \frac{-3}{4} & 1 \end{bmatrix}$$

$$U = \begin{bmatrix} 1 & -2 & -1 & 0 \\ 0 & -12 & -5 & 1 \\ 0 & 0 & \frac{-7}{3} & \frac{-10}{3} \\ 0 & 0 & 0 & \frac{-5}{4} \end{bmatrix}$$

Početna matrica dobije se umnoškom matrice L i U.

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ -6 & 1 & 0 & 0 \\ 1 & \frac{-2}{3} & 1 & 0 \\ 2 & \frac{-1}{4} & \frac{-3}{4} & 1 \end{bmatrix} \begin{bmatrix} 1 & -2 & -1 & 0 \\ 0 & -12 & -5 & 1 \\ 0 & 0 & \frac{-7}{3} & \frac{-10}{3} \\ 0 & 0 & 0 & \frac{-5}{4} \end{bmatrix} = \begin{bmatrix} 1 & -2 & -1 & 0 \\ -6 & 0 & 1 & 1 \\ 1 & 6 & 0 & -4 \\ 2 & -1 & 1 & 1 \end{bmatrix}$$

Sada se ide na rješavanje prvog sustava koji se spomenut na početku ovog dijela, a to je sustav $Ly = b$.

$$y_1 = 1 \implies y_1 = 1$$

$$y_2 = 6y_1 + 1 \implies y_2 = 7$$

$$y_3 = -y_1 + \frac{2}{3}y_2 + 0 \implies y_3 = \frac{11}{3}$$

$$y_4 = -2y_1 + \frac{1}{4}y_2 + \frac{3}{4}y_3 + 0 \implies y_4 = \frac{5}{2}$$

Nakon što je riješen prvi sustav ide se na rješavanje drugog sustava koji je također ranije spomenut, a izgleda ovako $Ux = y$.

$$-\frac{5}{4}x_4 = \frac{5}{2} \implies x_4 = -2$$

$$-\frac{7}{3}x_3 = \frac{10}{3}x_4 + \frac{11}{3} \implies x_3 = \frac{9}{7}$$

$$12x_2 = -x_4 + 5x_3 + 7 \implies x_2 = -\frac{9}{7}$$

$$x_1 = x_3 + 2x_2 + 1 \implies x_1 = -\frac{2}{7}$$

U slučajevima kada se na dijagonali matrice pojavljuju nule koristimo LU faktorizaciju s pivotiranjem. Idući primjer će prikazati korak po korak LU faktorizaciju s pivotiranjem koja će biti jako bitan faktor za objašnjavanje nadolazećih dijelova. Prva stvar koju je potrebno označiti je vektor p koji će označavati permutacije redaka, te koji će izgledati na sljedeći način:

$$p = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

S obzirom da će se raditi s istim sustavom kao i u prethodnom zadatku navest će se oblici tog sustava.

Sustav:

$$x_1 - 2x_2 - x_3 = 1$$

$$-6x_1 + x_3 + x_4 = 1$$

$$x_1 + 6x_2 - 4x_4 = 0$$

$$2x_1 - x_2 + x_3 + x_4 = 0$$

Matrični zapis:

$$\begin{bmatrix} 1 & -2 & -1 & 0 \\ -6 & 0 & 1 & 1 \\ 1 & 6 & 0 & -4 \\ 2 & -1 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

Prvi korak u procesu ovoga izračuna je zamjena prvog i drugog retka. Pivotiranje se radi tako da se matrica p pomnoži s matricom koju želimo permutirati.

$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & -2 & -1 & 0 \\ -6 & 0 & 1 & 1 \\ 1 & 6 & 0 & -4 \\ 2 & -1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} -6 & 0 & 1 & 1 \\ 0 & -2 & -1 & 0 \\ 1 & 6 & 0 & -4 \\ 2 & -1 & 1 & 1 \end{bmatrix}$$

$$p = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Prvi redak odnosno prva jednadžba će se pomnožiti s $\frac{1}{6}$ te dodati na treću jednadžbu. Zatim prva jednadžba će se pomnožiti s $\frac{1}{3}$ i dodati na četvrtu jednadžbu kako bi se vrijednosti ispod prvog elementa poništile.

$$\begin{bmatrix} -6 & 0 & 1 & 1 \\ 0 & -2 & -1 & 0 \\ 1 & 6 & 0 & -4 \\ 2 & -1 & 1 & 1 \end{bmatrix} \begin{matrix} \mid \cdot \frac{1}{6} \quad \mid \cdot \frac{1}{3} \\ \\ \longleftarrow + \\ \longleftarrow \quad + \end{matrix}$$

$$\begin{bmatrix} -6 & 0 & 1 & 1 \\ 0 & -2 & -1 & 0 \\ 0 & 6 & \frac{1}{6} & \frac{-23}{6} \\ 0 & -1 & \frac{4}{3} & \frac{4}{3} \end{bmatrix}$$

Drugi redak se množi s određenim vrijednostima kako bi se vrijednosti ispod njega poništile.

$$\begin{bmatrix} -6 & 0 & 1 & 1 \\ 0 & -2 & -1 & 0 \\ 0 & 6 & \frac{1}{6} & \frac{-23}{6} \\ 0 & -1 & \frac{4}{3} & \frac{4}{3} \end{bmatrix} \begin{matrix} \\ \mid \cdot 3 \mid \cdot -\frac{1}{2} \\ \\ \\ \end{matrix}$$

$$\begin{bmatrix} -6 & 0 & 1 & 1 \\ 0 & -2 & -1 & 0 \\ 0 & 0 & \frac{-17}{6} & \frac{-23}{6} \\ 0 & 0 & \frac{11}{6} & \frac{4}{3} \end{bmatrix}$$

U konačnici množi se treći redak s $\frac{11}{17}$ kako bi se vrijednosti ispod njega poništile.

$$\begin{bmatrix} -6 & 0 & 1 & 1 \\ 0 & -2 & -1 & 0 \\ 0 & 0 & \frac{-17}{6} & \frac{-23}{6} \\ 0 & 0 & \frac{11}{6} & \frac{4}{3} \end{bmatrix} \mid \cdot \frac{11}{17}$$

$$\begin{bmatrix} -6 & 0 & 1 & 1 \\ 0 & -2 & -1 & 0 \\ 0 & 0 & \frac{-17}{6} & \frac{-23}{6} \\ 0 & 0 & 0 & \frac{-39}{34} \end{bmatrix}$$

$$\begin{bmatrix} -6 & 0 & 1 & 1 \\ 0 & -2 & -1 & 0 \\ -\frac{1}{6} & -3 & \frac{-17}{6} & \frac{-23}{6} \\ -\frac{1}{3} & \frac{1}{2} & -\frac{11}{17} & \frac{-39}{34} \end{bmatrix}$$

Radi uštede memorije koristimo zapis iznad, ali iz njega je moguće pročitati te razvrstati sljedeće matrice.

$$L = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -\frac{1}{6} & -3 & 1 & 0 \\ -\frac{1}{3} & \frac{1}{2} & \frac{-11}{17} & 1 \end{bmatrix}$$

$$U = \begin{bmatrix} -6 & 0 & 1 & 1 \\ 0 & -2 & -1 & 0 \\ 0 & 0 & \frac{-17}{6} & \frac{-23}{6} \\ 0 & 0 & 0 & \frac{-39}{34} \end{bmatrix}$$

$$P = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

$$b' = Pb = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

Nakon izvučenih elemenata iz rezultata koji je dobiven oblikovanjem početne matrice, potrebno je riješiti dva sustava od kojih je prvi $Ly = b'$.

$$y_1 = 1 \implies y_1 = 1$$

$$y_2 = 0y_1 + 1 \implies y_2 = 1$$
$$y_3 = \frac{1}{6}y_1 + 3y_2 + 0 \implies y_3 = \frac{19}{6}$$
$$y_4 = \frac{1}{3}y_1 - \frac{1}{2}y_2 + \frac{11}{17}y_3 + 0 \implies y_4 = \frac{32}{17}$$

Te drugi sustav koji je oblika $Rx = y$

$$-\frac{39}{34}x_4 = \frac{32}{17} \implies x_4 = -\frac{64}{39}$$

$$-\frac{17}{6}x_3 = -\frac{23}{6}x_4 + \frac{19}{6} \implies x_3 = -\frac{2213}{663}$$
$$-2x_2 = 0x_4 - 1x_3 + 1 \implies x_2 = -\frac{1438}{663}$$
$$-6x_1 = 1x_4 + 1x_3 + 0x_2 + 1 \implies x_1 = -\frac{2638}{663}$$

Objašnjenje sustava koji je napravljen u prethodnom izračunu je prilično jednostavno. Matrica se faktorizira te nakon faktoriziranja u gornjem dijelu trokutaste matrice nalazi se matrica U, a u donjem dijelu gdje su smješteni multiplikatori nalazi se matrica L. Algoritam LU faktorizacije te algoritam rješavanja sustava nakon LU faktorizacije moguće je vidjeti ispod.

Ulaz: red matrice n i matrica $A = (a_{ij})$

for $k = 1, 2, \ldots, n-1$ do

for $i = k+1, k+2, \ldots n$ do

$z = a_{ik}/a_{kk}$

$a_{ik} = z$

end for

end for

end for

Izlaz: matrica $A = (a_{ij})$ **[5]**

Sljedeći algoritam će nakon napravljene faktorizacije transformirati desnu stranu i riješiti sustav s gornjom trokutastom matricom.

Ulaz: red matrice n i matrica $A = (a_{ij})$ nakon LU faktorizacije

i vektor desne strane $b = (b_i)$

for $k = 1, 2, \ldots, n-1$ do

for $i = k+1, k+2, \ldots n$ do

$b_i = b_i - a_{ik} b_k$

end for

end for

$b_n = b_n / a_{nn}$

for $i = n-1, n-2, \ldots, 1$ do

for $j = i+1, i+2, \ldots, n$ do

$b_i = b_i - a_{ij} b_j$

end for

$b_i = b_i / a_{ii}$

end for

Izlaz: vektor $b = (b_i)$ **[5]**

Nakon danih jasnih primjera sada je potrebno iskazati i egzistenciju te jedinstvenost. Kod ovoga dijela jako je bitno spomenuti kvadratne matrice, simetrične pozitivne definitne matrice, te generalne matrice odnosno dio koji će reći još nešto općenito o matricama. U nastavku će biti iskazan teorem koji jasno dokazuje egzistenciju i jedinstvenost LU faktorizacije.

Teorem 1. *Neka je $A \in \mathbb{R}^{n \times n}$ i neka su determinante glavnih podmatrica $A(1:k, 1:k)$ različite od nule za $k = 1, 2, \cdots, n-1$. Tada postoji donjetrokutasta matrica L s jedinicama na dijagonali i gornjetrokutasta matrica U, tako da vrijedi $A = LU$.Ako faktorizacija $A = LU$ postoji i ako je još matrica A regularna, onda je faktorizacija jedinstvena: postoji točno jedna matrica L i točno jedna matrica U s ovim svojstvima.* **[10]**

Dokaz. Dokažimo prvo jedinstvenost LU faktorizacije. Neka postoje dvije takve faktorizacije,

$$A = LU = L'U'$$

Ako je A regularna onda su i L, U, L', U' također regularne matrice pa vrijedi

$$L^{-1}L' = U(U')^{-1}$$

U gornjoj relaciji imamo jednakost donje trokutaste i gornje trokutaste matrice - znači da na obe strane jednakosti stoje dijagonalne matrice. Nadalje, L i L' po pretpostavci imaju jedinice na dijagonali, a zbog činjenice da se na dijagonali produkta donje trokutastih matrica nalaze produkti dijagonalnih elemenata matrica koje se množe, na dijagonali od $L^{-1}L'$ jedinice. Dakle, $L^{-1}L' = I$, tj. $L = L'$. Tada je i $U = U'$. Dokažimo sada egzistencije LU faktorizacije. Induktivni dokaz je zapravo već skiciran u opisu računanja faktorizacije. Pogledajmo kako uvjeti teorema omogućuju prelaz sa $A^{(k)}$ na $A^{(k+1)}$ gdje je:

$$A^{(k)} = L^{(k)} \cdots L^{(1)} A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1k} & a_{1,k+1} & \dots & a_{1n} \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & \dots & \vdots & a_{2,k+1}^{(1)} & \dots & a_{2n}^{(1)} \\ \vdots & \ddots & a_{33}^{(2)} & & \vdots & a_{3,k+1}^{(2)} & \dots & a_{3n}^{(2)} \\ \vdots & & \ddots & \ddots & \vdots & \vdots & & \vdots \\ 0 & \dots & \dots & 0 & a_{k,k}^{(k-1)} & a_{k,k+1}^{(k-1)} & \dots & a_{kn}^{k-1)} \\ 0 & \dots & & \dots & 0 & a_{k+1,k+1}^{(k)} & \dots & a_{k+1,n}^{k)} \\ \vdots & & & & \vdots & \vdots & & \vdots \\ 0 & \dots & & \dots & 0 & a_{n,k+1}^{(k)} & \dots & a_{nn}^{k)} \end{bmatrix}$$

Kako je produkt $(L^{(k)} \dots L^{(1)})^{-1}$ donje trokutasta matrica sa jedinicama na dijagonali, zaključujemo da je

$$\det A(1:k+1, 1:k+1) = a_{11} a_{22}^{(1)} a_{33}^{(2)} \dots a_{kk}^{(k-1)} a_{k+1,k+1}^{(k)} \neq 0$$

Odavde je i $a_{k+1,k+1}^{(k)} \neq 0$ pa možemo definirati matricu $L^{(k+1)}$ koja će poništiti elemente ispod dijagonale u $(k+1)$-om stupcu i dati $A^{(k+1)} = L^{(k+1)} A^{(k)}$. Jasno je da nakon konačnog koraka dobijemo matricu $A^{(n-1)}$ koja je gornjetrokutasta. **[10]** □

2.4.3. Pozitivno definitne matrice

Prema svim izvorima koji su navedeni u ovom radu može se zaključiti da je matrica koju nazivamo pozitivno definitnom matricom, odnosno pozitivno definitnim sustavom, zapravo kvadratna $n \times n$ matrica A za koju vrijedi:

$$x \in R^n$$

$$x \neq 0$$

$$Ax \cdot x > 0$$

Trebalo bi se naglasiti da je gore navedena operacija $Ax \cdot x > 0$ skalarni produkt. Iz ovih uvjeta slijedi sljedeća definicija:

Definicija 1. Kvadratna matrica $A \in R^{n \times n}$ je pozitivno definitna ako za svaki vektor $x \in R^n, x \neq 0$, vrijedi $Ax \cdot x > 0$. **[5]**

Teorem 2. *(Faktorizacija Choleskog) Neka je $A \in R^{n\times n}$ simetrična pozitivno definitna matrica. Tada postoji jedinstvena donja trokutasta matrica G s pozitivnom dijagonalom, takva da je $A = GG^T$).* **[5]**

Dokaz. Znamo da je $A = LDL^T$, gdje je $D = diag(d_1; \cdots, d_n)$. Budući da su elementi d_i pozitivni definiramo $G = Ldiag(\sqrt{d_1}, \cdots, \sqrt{d_n})$ Evidentno je $A = GG^T$).Jedinstvenost slijedi iz jedinstvenosti LDL^T faktorizacije. **[5]** □

Također možemo ovo prikazati uz pomoć algoritma faktorizacije Choleskog koji jasno kroz korake pokazuje kako se prema njemu vrši faktorizacija.

Ulaz:red matrice n i matrica $A = (a_{ij})$

for $j = 1, 2, \ldots, n$do

$g_{jj} = a_{jj}$

for $k = 1, \ldots, j-1$do

$g_{jj} = g_{jj} - g_{jk}^2$

$g_{jj} = \sqrt{g_{jj}}$

for $i = j+1, \ldots, n$do

$g_{ij} = a_{ij}$

for $k = 1, \ldots, j-1$do

$g_{ij} = g_{ij} - g_{ik}g_{jk}$

end for

$g_{ij} = g_{ij}/g_{jj}$

end for

end for

Izlaz:matrica $G = (g_{ij})$ **[5]**

3. Opis algoritma

Pod ovom cjelinom rad će napraviti usporedbu između algoritama koji su napravljeni prema literaturi te gotovih rješenja koja su izrađena za ovaj završni rad.

Ovaj kod se sastoji od nekoliko dijelova koji omogućuju rad programa, a ti dijelovi se odnose na različite izračune koje korisnik treba odabrati nakon unosa svih potrebnih podataka. Početak programa započinje s iscrtavanjem prozora u kojemu će se program prikazivati. Nadalje kako bi program mogao znati s kojom standardnom veličinom matrice radi, korisnik mora napraviti unos veličine kvadratne matrice.

Nakon unosa veličine u kodu se mogu primijetiti razne funkcije koje su definirane te koje će se međusobno pozivati kada korisnik odabere određenu faktorizaciju. U varijablu "faktorizacije" će se spremiti unos korisnika koji odabire faktorizaciju choleskog ili LU faktorizaciju s pivotiranjem. Kada je korisnik odabrao određenu faktorizaciju vrijednost te faktorizacije se prosljeđuje te se poziva funkcija cholesky(A) koja će izvesti dekompoziciju prema Choleskom matrice A koja mora biti simetrična i pozitivno definitna matrica, ta funkcija će kao rezultat vratiti donje trokutastu matricu.

Sustav koji ćemo koristiti za faktorizaciju Choleskog.

$$x_1 + x_2 + x_3 + x_4 + x_5 = 1$$

$$x_1 + 2x_2 + 3x_3 + 4x_4 + 5x_5 = 5$$

$$x_1 + 3x_2 + 6x_3 + 10x_4 + 15x_5 = 1$$

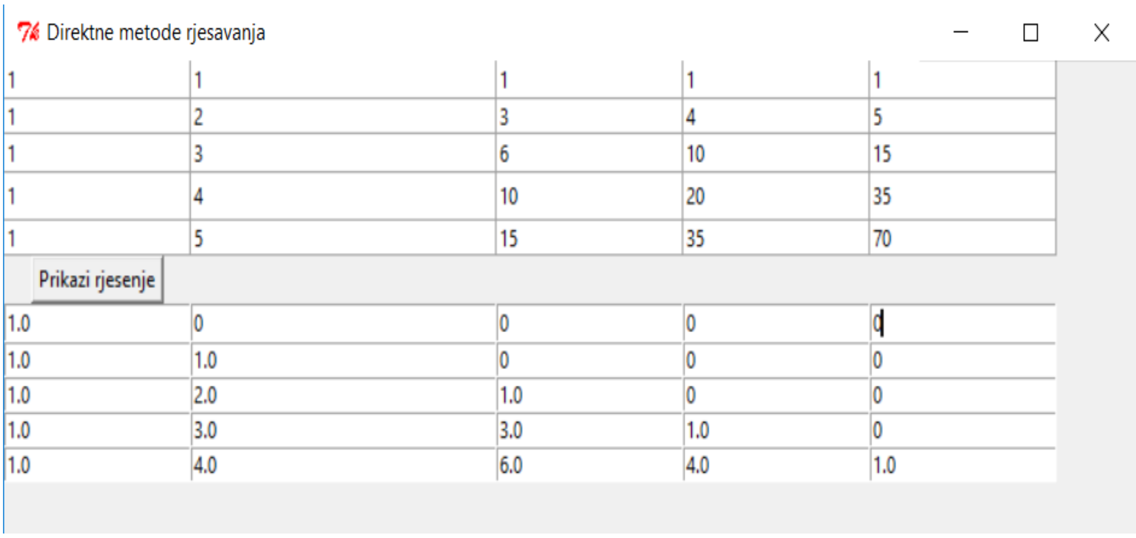
$$x_1 + 4x_2 + 10x_3 + 20x_4 + 35x_5 = 4$$

$$x_1 + 5x_2 + 15x_3 + 35x_4 + 70x_5 = 3$$

$$A = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 3 & 4 & 5 \\ 1 & 3 & 6 & 10 & 15 \\ 1 & 4 & 10 & 20 & 35 \\ 1 & 5 & 15 & 35 & 70 \end{bmatrix}, B = \begin{bmatrix} 1 \\ 5 \\ 1 \\ 4 \\ 3 \end{bmatrix}$$

```
def cholesky(A):
    n = len(A)
    L = [[0.0] * n for i in xrange(n)]
    for i in xrange(n):
        for k in xrange(i + 1):
            tmp_sum = sum(MatrixZaCho[i][j] * MatrixZaCho[k][j] for j in xrange(k))

            if i == k:
                MatrixZaCho[i][k] = round(sqrt(A[i][i] - tmp_sum), 2)
            else:
                MatrixZaCho[i][k] = round(1.0 / MatrixZaCho[k][k] * (A[i][k] -
                    tmp_sum), 2)
    return MatrixZaCho
```

Rezultat izvršavanja algoritma iznad izgleda ovako:



1	1	1	1	1
1	2	3	4	5
1	3	6	10	15
1	4	10	20	35
1	5	15	35	70

1.0	0	0	0	0
1.0	1.0	0	0	0
1.0	2.0	1.0	0	0
1.0	3.0	3.0	1.0	0
1.0	4.0	6.0	4.0	1.0

Slika 1: Cholesky dekompozicija

Pomoću sljedećeg algoritma smo dobili rješenje gore navedenog sustava.

```
from scipy.linalg import cho_factor, cho_solve
import numpy as np
A = np.array([[1, 1, 1, 1, 1],
 [1, 2, 3, 4, 5],
 [1, 3, 6, 10, 15],
 [1, 4, 10, 20, 35],
 [1, 5, 15, 35, 70]])
L, low = cho_factor(A)
B = np.array([1,5,1,4,3])
x = cho_solve((L,low), B)

print "Choleskog faktorizacija od matrice A:\n",L

print "Rjesenje sustava:"
for i in range(len(x)):
    print('x' + str(i + 1), x[i])
```

Iz gore navedenog sustava matrica A ubacuje se u funkciju cho_factor() koja se nalazi u python biblioteci pod nazivom scipy te dobivamo novu matricu L. Zatim, matricu L skupa s matricom slobodnih koeficijenata B ubacujemo u funkciju cho_solve() te dobivamo rješenje gore navedenog sustava.

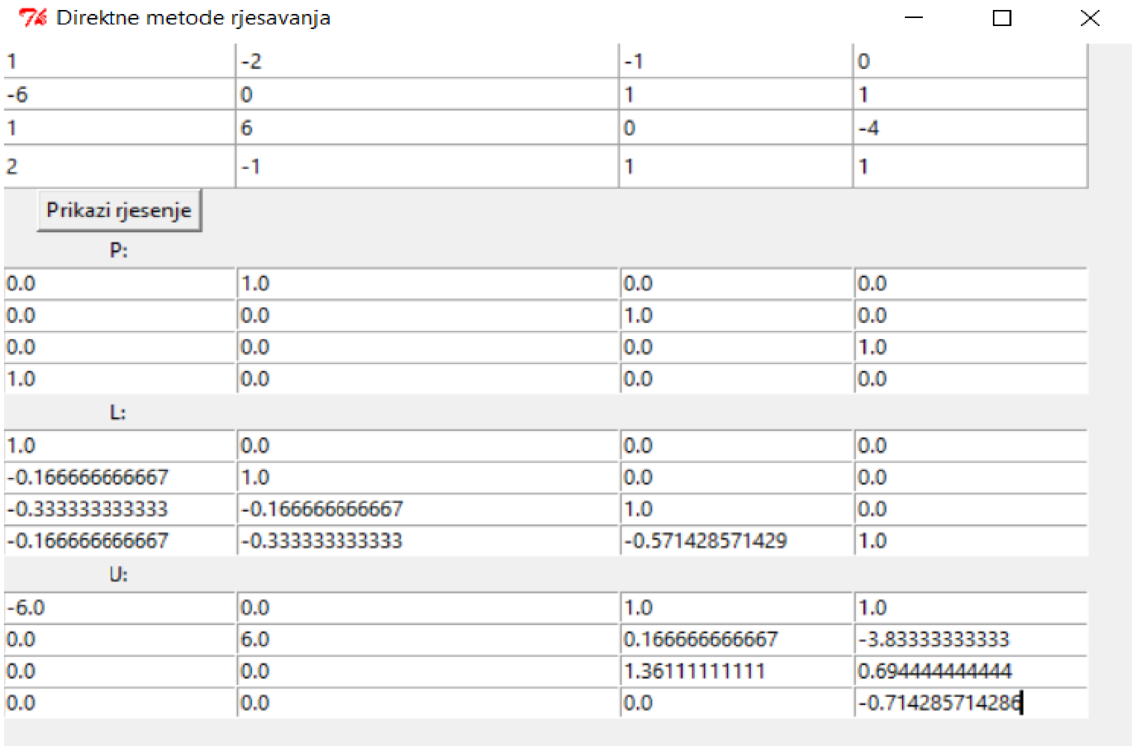
```
Choleskog faktorizacija od matrice A:
[[ 1.   1.   1.   1.   1.]
 [ 1.   1.   2.   3.   4.]
 [ 1.   3.   1.   3.   6.]
 [ 1.   4.  10.   1.   4.]
 [ 1.   5.  15.  35.   1.]]
Rjesenje sustava:
('x1', -52.0)
('x2', 169.0)
('x3', -209.0)
('x4', 119.0)
('x5', -26.0)
```

Slika 2: Rješenje linearnog sustava pomoću Cholesky faktorizacije

U slučaju odabira LU faktorizacije s pivotiranjem poziva se funkcija luDecomposition(A) koja izvodi dekompoziciju matrice A koja mora biti kvadratna matrica. Ova funkcija vraća tri vrijednosti označene sa slovima P, L i U.

```
def luDecomposition(A):
    n = len(A)
    global P, L, U
    L = [[0.0] * n for i in xrange(n)]
    U = [[0.0] * n for i in xrange(n)]

    P = pivotMatrix(A)
    PA = multMatrix(P, A)

    for j in xrange(n):
        L[j][j] = 1.0

        for i in xrange(j+1):
            s1 = sum(U[k][j] * L[i][k] for k in xrange(i))
            U[i][j] = PA[i][j] - s1

        for i in xrange(j, n):
            s2 = sum(U[k][j] * L[i][k] for k in xrange(j))
            L[i][j] = (PA[i][j] - s2) / U[j][j]
```

Ovaj algoritam ima tri fukcije, glavnu funkciju koja se zove luDekompozicija(A) koja poziva ostale dvije funkcije multMatrix(M,N) čiji je zadatak da pomnoži dvije kvadratne matrice istih dimenzija M i N i funkciju pivotMatrix(M) koja vraća pivotiranu matricu za M. A sami rezultat izvršavanja ovoga algoritma izgleda na sljedeći način:



Slika 3: LU dekompozicija

Pomoću sljedećeg algoritma smo dobili rješenje gore navedenog sustava.

```
import pprint
import scipy
import scipy.linalg

A = scipy.array([[1, -2, -1,0],
 [-6,0,1,1],
 [1,6,0,-4],
 [2, -1, 1, 1]])
P, L, U = scipy.linalg.lu(A)
#definiranje B
B = scipy.array([1, 1, 0, 0])
#lu faktorizacija
LU = scipy.linalg.lu_factor(A)
#dobivanje rjesenja
x = scipy.linalg.lu_solve(LU, B)
print "A:"
pprint.pprint(A)
print "P:"
pprint.pprint(P)
print "L:"
pprint.pprint(L)
print "U:"
pprint.pprint(U)

print "Rjesenje:"
for i in range(len(x)):
    print('x'+str(i+1),x[i])
```

Kao primjer matrice u gore navedenom izvođenju programa koristila se matrica iz poglavlja 2.4.2 LU faktorizacija. Matrica A iz tog sustava ubacuje se u funkciju lu() te dobivamo matrice P, L, U. Nakon toga definiramo matricu slobodnih koeficijenata B. Ubacivanjem matrice A u funkciju lu_factor() dobiva se LU matrica koja se zatim ubacuje u funkciju lu_solve() skupa s matricom B da bi se dobilo rješenje sustava. Dobivena rješenja ista su kao u teorijskom dijelu rada.

```
A:
array([[ 1, -2, -1,  0],
       [-6,  0,  1,  1],
       [ 1,  6,  0, -4],
       [ 2, -1,  1,  1]])
P:
array([[0., 0., 0., 1.],
       [1., 0., 0., 0.],
       [0., 1., 0., 0.],
       [0., 0., 1., 0.]])
L:
array([[ 1.        ,  0.        ,  0.        ,  0.        ],
       [-0.16666667,  1.        ,  0.        ,  0.        ],
       [-0.33333333, -0.16666667,  1.        ,  0.        ],
       [-0.16666667, -0.33333333, -0.57142857,  1.        ]])
U:
array([[-6.        ,  0.        ,  1.        ,  1.        ],
       [ 0.        ,  6.        ,  0.16666667, -3.83333333],
       [ 0.        ,  0.        ,  1.36111111,  0.69444444],
       [ 0.        ,  0.        ,  0.        , -0.71428571]])
Rjesenje:
('x1', -0.28571428571428575)
('x2', -1.285714285714286)
('x3', 1.285714285714286)
('x4', -2.0000000000000004)
```

Slika 4: Rješenje linearnog sustava pomoću LU faktorizacije

4. Zaključak

Tema ovog završnog rada bile su direktne metode za rješavanje linearnih sustava koje su detaljno objašnjene i potkrijepljene primjerima. Zaključili smo da se direktne metode ne mogu uvijek koristiti jer nekada nisu dovoljno efikasne, provode velik broj matematičkih operacija i koriste više računalne memorije u odnosu na iterativne metode. U programskom jeziku python napravljeni su i algoritmi metode Choleskog i LU faktorizacije koji su zatim uspoređeni i prikazani primjerima s rješenjima. Nakon rješavanja programskog dijela zadatka dobili smo upravo ono što je opisano prilikom razmatranja algoritama u teorijskom dijelu rada. Možemo zaključiti da sve spomenuto u ovom radu ima praktičnu primjenu te se u stvarnom životu često primjenjuje prilikom težih izračuna u određenim tehnološkim projektima.

Popis literature

[1] Miljenko Marušić, Mladen Rogina, Sanja Singer, Saša Singer, Vjeran Hari, Zlatko Drmač: Numerička analiza, predavanje I vježbe, 2003.

[2] Web materijali FSB: https://www.fsb.unizg.hr/mat-4/OldWeb/2.pdf, posjećeno 08.srpnja 2018.

[3] Web materijali FSB: www.fsb.unizg.hr/matematika/download/ZS/mat1/Matematika-1-beamer-CH-07.pdf, posjećeno 08.srpnja 2018.

[4] Web materijali PMF https://web.math.pmf.unizg.hr/~hari/la.pdf, posjećeno 07.srpnja 2018.

[5] Web materijali PMF: https://web.math.pmf.unizg.hr/nastava/ppm1/ppm_lin.pdf, posjećeno 08.srpnja 2018.

[6] Web materijali FSB: http://powerlab.fsb.hr/numerickemetodeuzii/rjesavanje_linearnih_sustava.htm, posjećeno 07.srpnja 2018.

[7] Web materijali Mathos: http://www.mathos.unios.hr/rp2/Iter_Method.pdf, posjećeno 06.rujna 2018.

[8] Web materijali PMFST: http://mapmf.pmfst.unist.hr/~milica/Numericka_matematika/Folije_za_predavanja/UNM_SustaviLJ.pdf, posjećeno 06.rujna 2018.

[9] Web materijali Element: https://element.hr/artikli/file/1406, posjećeno 06.rujna 2018.

[10] Web materijali PMF: https://web.math.pmf.unizg.hr/~drmac/na001.pdf, posjećeno 06.rujna 2018.

Popis slika

1. Cholesky dekompozicija 22
2. Rješenje linearnog sustava pomoću Cholesky faktorizacije 23
3. LU dekompozicija 24
4. Rješenje linearnog sustava pomoću LU faktorizacije 25