Jacobijeva metoda za simetrični problem svojstvenih vrijednosti

Marinović, Toni

SVEUČILIŠTE U ZAGREBU

PRIRODOSLOVNO–MATEMATIČKI FAKULTET

MATEMATIČKI ODSJEK

Toni Marinović

JACOBIJEVA METODA ZA SIMETRIČNI PROBLEM SVOJSTVENIH VRIJEDNOSTI

Diplomski rad

Voditelj rada:
prof. dr. sc. Nela Bosner

Zagreb, veljača, 2022.

Ovaj diplomski rad obranjen je dana ____________________ pred ispitnim povjerenstvom u sastavu:

1. ____________________________, predsjednik
2. ____________________________, član
3. ____________________________, član

Povjerenstvo je rad ocijenilo ocjenom _______________.

Potpisi članova povjerenstva:

1. ____________________________
2. ____________________________
3. ____________________________

Zahvaljujem svojoj obitelji koja mi je tijekom školovanja bili najveća podrška. Također se zahvaljujem prijateljima koji su mi uljepšali studentske dane kojih ću se rado prisjećati. Posebna zahvala mentorici na brojnim sugestijama, pomoći i uloženom vremenu za izradu ovog rada.

Sadržaj

Uvod

Simetričan problem svojstvenih vrijednosti je jedan dosta zastupljeni problem u raznim prirodnim znanostima poput fizike, kemije, ekonomije, ... Problem se sastoji od nalaženja ortogonalne matrice U takve da je za simetriču matricu A, $A = U\Lambda U^T$, a Λ je dijagonalna matrica svojstvenih vrijednosti. Razne varijante Jacobijeve metode predstavljaju važnu grupu metoda za računanje svih svojstvenih vrijednosti i vektora pune simetrične matrice. Jacobijeve metode se još i danas razvijaju, a glavna karakteristika im je da su točnije od ostalih metoda, i da su pogodne za paralelizaciju. Osnova Jacobijevog algoritma je primjena transformacija sličnosti pomoću elementarnih ortogonalnih matrica $A \leftarrow Q^T AQ$, tako da svaka nova inačica matrice A bude "više dijagonalna" od prethodne. U radnji se opisuje izvod metode, i pokazauje da pod određenim uvjetima ovaj proces konvergira prema dijagonalnoj matrici. Nakon određenog broja iteracija vandijagonalni elementi matrice A postaju dovoljno mali da ih možemo proglasiti nulama i metoda se zaustavlja sa dovoljno dobrim aproksimacijama matrice U i Λ. U radnji se obrađuje nekoliko varijanti Jacobijeve metode, koje uključuju blokiranu i paralelnu varijantu. Isto tako, navodimo rezultate o točnosti izračunatih svojstvenih vrijednosti i vektora. Na kraju obrađene algoritme ilustriramo programima izrađenim u MATLAB-u, te su primjenjeni na nekim konkretnim primjerima.

Poglavlje 1

Svojstva simetrične matrice

1.1 Definicije i osnovne činjenice

Matricu $A = [a_{ij}] \in \mathbb{R}^{n \times n}$ nazivamo simetrična matrica ako je jednaka svojoj transponiranoj matrici,

$$A = A^T \quad \text{ili} \quad a_{ij} = a_{ji}, \quad \text{za} \quad i, j = 1, ...n.$$

Matrica $A \in \mathbb{R}^{n \times n}$ je:

- ortogonalna ako vrijedi $A^T A = AA^T = I_n$,
- normalna ako vrijedi $A^T A = AA^T$.

Neka je $A \in \mathbb{R}^{n \times n}$, tada je Frobeniusova ili Euklidska norma definirana sa

$$\|A\|_F := \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{n} |a_{ij}|^2} = \sqrt{(tr(A^T A))},$$

gdje je $tr(\cdot)$ trag matrice - zbroj dijagonalnih elemenata.
Frobeniusova norma je unitarno invarijantna, tj. za unitarne matrice U i V vrijedi

$$\|UAV^*\|_F = \|A\|_F.$$

Posebno, to vrijedi i za svaku ortogonalnu matricu Q, tj.

$$\|QAQ^T\|_F = \|A\|_F.$$

1.2 Svojstvene vrijednosti i svojstveni vektori

Važno svojstvo realnih, simetričnih matrica je da imaju realne svojstvene vrijednosti. Odnosno, ako je A realna, simetrična matrica reda n, tada matrica A ima n ne nužno različitih realnih svojstvenih vrijednosti $\lambda_1, ..., \lambda_n$.

Definicija 1.2.1. *Neka je $A \in \mathbb{R}^{n\times n}$ simetrična matrica i $x \in \mathbb{R}^n$.Kažemo da je skalar λ svojstvena vrijednost simetrične matrice A, ako $\exists x \in \mathbb{R}^n, x \neq 0$ takav da je $Ax = \lambda x$. Tada vektor x nazivamo svojstveni vektor od A pridružen svojstvenoj vrijednosti λ.*

Napomena 1.2.2. *U MATLAB-ovskoj notaciji, sa $Q(:, k)$ pristupamo k-tom stupcu matrice Q. U narednom teoremu $Q(:, k)$ predstavlja k-ti svojstveni vektor.*

Teorem 1.2.3 (Simetrična Schurova dekompozicija)**.** *Ako je $A \in \mathbb{R}^{n\times n}$ simetrična, onda postoji realna ortogonalna matrica Q tako da vrijedi*

$$Q^T AQ = \Lambda = diag(\lambda_1, ..., \lambda_n).$$

Štoviše, za $k = 1, ..., n$, $AQ(:, k) = \lambda_k Q(:, k)$.

Dokaz. Dokaz možete pogledati u [2] ili [3]. □

Teorem 1.2.4 (Wielandt–Hoffman)**.** *Neka su D i E kvadratne matrice reda n i neka su matrice D i $D + E$ obje normalne. Neka su $\lambda_1, \ldots, \lambda_n$ svojstvene vrijednosti od D i neka su $\lambda'_1, \ldots, \lambda'_n$ svojstvene vrijednosti od $D + E$ u nekom poretku.*
Tada postoji permutacija indeksa π, takva da vrijedi

$$\left(\sum_{i=1}^{n} |\lambda'_{\pi(i)} - \lambda_i|\right)^{\frac{1}{2}} \leq \|E\|_F$$

Dokaz. Dokaz možete pogledati u [3]. □

Poglavlje 2

Jacobijeva metoda

2.1 Izvod i konvergencija Jacobijeve metode

Izvod

Neka je $A = A_0 \in \mathbb{R}^{n \times n}$ simetrična matrica. Jacobijeva metoda stvara slijed $A_1, A_2...$ ortogonalno sličnih matrica koje konvergiraju prema dijagonalnoj matrici sa svojstvenim vrijednostima na dijagonali. A_{i+1} je dobiven iz A_i prema formuli $A_{i+1} = J_i^T A_i J_i$ gdje je J_i ortogonalna matrica koju zovemo Jacobijeva rotacija.
Slijedi,

$$\begin{aligned} A_m &= J_{m-1}^T A_{m-1} J_{m-1} \\ &= J_{m-1}^T J_{m-2}^T A_{m-2} J_{m-2} J_{m-1} \\ &= J_{m-1}^T \dots J_0^T A_0 J_0 \dots J_{m-1} \\ &= J^T A J \end{aligned}$$

Ako svaki J_i prikladno izaberemo,vandijagonalni elementi matrice A_m, za dovoljno velike m, će postati dovoljno mali da ih možemo proglasiti nulama. Tada će se matrica A_m moći smatrati dijagonalnom matricom Λ. Dakle, možemo pisati $\Lambda \approx J^T A J$ ili $J \Lambda J^T \approx A$, gdje su stupci matrice J pripadni svojstveni vektori.
Matricu A ćemo učiniti dijagonalnom iterativnim biranjem ortogonalne matrice J_i tako da par vandijagonalnih elemenata (a_{jk}, a_{kj}) u sljedećem koraku $A_{i+1} = J_i^T A_i J$ učinimo nulama.

To ćemo postići birajući J_i kao ravninsku rotaciju

$$J_i = R(j,k,\phi) = \begin{bmatrix} 1 & & & & & & & & \\ & \ddots & & & & & & & \\ & & 1 & & & & & & \\ & & & \cos(\phi) & & \sin(\phi) & & & \\ & & & & \ddots & & & & \\ & & & -\sin(\phi) & & \cos(\phi) & & & \\ & & & & & & 1 & & \\ & & & & & & & \ddots & \\ & & & & & & & & 1 \end{bmatrix},$$

gdje je ϕ izabran tako da elementi $a_{jk}^{(i+1)}$ i $a_{kj}^{(i+1)}$ matrice A_{i+1} budu jednaki nula.
Rotaciju u čast autoru nazivamo Jacobijeva rotacija.
Da bi odredili ϕ (ili $\cos(\phi)$ i $\sin(\phi)$) raspišimo jedan korak

$$\begin{bmatrix} a_{jj}^{(i+1)} & a_{jk}^{(i+1)} \\ a_{kj}^{(i+1)} & a_{kk}^{(i+1)} \end{bmatrix} = \begin{bmatrix} \cos(\phi) & \sin(\phi) \\ -\sin(\phi) & \cos(\phi) \end{bmatrix}^T \begin{bmatrix} a_{jj}^{(i)} & a_{jk}^{(i)} \\ a_{kj}^{(i)} & a_{kk}^{(i)} \end{bmatrix} \begin{bmatrix} \cos(\phi) & \sin(\phi) \\ -\sin(\phi) & \cos(\phi) \end{bmatrix} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix},$$

gdje su λ_1 i λ_2 svojstvene vrijednosti od $\begin{bmatrix} a_{jj}^{(i)} & a_{jk}^{(i)} \\ a_{kj}^{(i)} & a_{kk}^{(i)} \end{bmatrix}$.
Radi jednostavnosti, uvodimo supstituciju $c = \cos(\phi)$ i $s = \sin(\phi)$.
Računamo,

$$\begin{aligned}
\begin{bmatrix} c & s \\ -s & c \end{bmatrix}^T \begin{bmatrix} a_{jj}^{(i)} & a_{jk}^{(i)} \\ a_{kj}^{(i)} & a_{kk}^{(i)} \end{bmatrix} \begin{bmatrix} c & s \\ -s & c \end{bmatrix} &= \begin{bmatrix} c & -s \\ s & c \end{bmatrix} \begin{bmatrix} a_{jj}^{(i)} & a_{jk}^{(i)} \\ a_{kj}^{(i)} & a_{kk}^{(i)} \end{bmatrix} \begin{bmatrix} c & s \\ -s & c \end{bmatrix} \\
&= \begin{bmatrix} ca_{jj}^{(i)} - sa_{kj}^{(i)} & ca_{jk}^{(i)} - sa_{kk}^{(i)} \\ sa_{jj}^{(i)} + ca_{kj}^{(i)} & sa_{jk}^{(i)} + ca_{kk}^{(i)} \end{bmatrix} \begin{bmatrix} c & s \\ -s & c \end{bmatrix} \\
&= \begin{bmatrix} c^2a_{jj}^{(i)} - sca_{kj}^{(i)} - sca_{jk}^{(i)} + s^2a_{kk}^{(i)} & csa_{jj}^{(i)} - s^2a_{kj}^{(i)} + c^2a_{jk}^{(i)} - sca_{kk}^{(i)} \\ sca_{jj}^{(i)} + c^2a_{kj}^{(i)} - s^2a_{jk}^{(i)} - sca_{kk}^{(i)} & s^2a_{jj}^{(i)} + sca_{kj}^{(i)} + sca_{jk}^{(i)} + c^2a_{kk}^{(i)} \end{bmatrix} \\
&= \text{simetričnost matrice, } a_{jk} = a_{kj} \\
&= \begin{bmatrix} c^2a_{jj}^{(i)} - 2sca_{jk}^{(i)} + s^2a_{kk}^{(i)} & sc(a_{jj}^{(i)} - a_{kk}^{(i)}) + a_{jk}^{(i)}(c^2 - s^2) \\ sc(a_{jj}^{(i)} - a_{kk}^{(i)}) + a_{jk}^{(i)}(c^2 - s^2) & s^2a_{jj}^{(i)} + c^2a_{kk}^{(i)} + 2sca_{jk}^{(i)} \end{bmatrix}
\end{aligned}$$

$$0 = sc(a_{jj}^{(i)} - a_{kk}^{(i)}) + a_{jk}^{(i)}(c^2 - s^2)$$

$$\frac{a_{kk}^{(i)} - a_{jj}^{(i)}}{a_{jk}^{(i)}} = \frac{c^2 - s^2}{sc} \tag{2.1}$$

Koristeći trigonometrijske formule dvostrukog kuta

$$\cos(2\phi) = c^2 - s^2$$
$$\sin(2\phi) = 2sc$$

iz (2.1) računamo

$$\frac{a_{kk}^{(i)} - a_{jj}^{(i)}}{2a_{jk}^{(i)}} = \frac{c^2 - s^2}{2sc} = \frac{\cos(2\phi)}{\sin(2\phi)} = \mathrm{ctg}(2\phi) \equiv \tau \tag{2.2}$$

Budući da τ znamo, iz prethodne jednadžbe ćemo iskoristiti jednakost

$$\tau = \frac{c^2 - s^s}{2sc} : \frac{c^2}{c^2} = \frac{1 - t^2}{2t},$$

gdje je $t = \frac{s}{c}$
Riješimo jednadžbu po t.

$$t^2 + 2t\tau - 1 = 0$$

$$t_{1,2} = \frac{-2\tau \pm \sqrt{4\tau^2 + 4}}{2} = -\tau \pm \sqrt{\tau^2 + 1}$$

Po apsolutnoj vrijednosti biramo manji od dva kuta

$$t = \begin{cases} -\tau + \sqrt{\tau^2 + 1}, & \tau \geq 0 \\ -\tau - \sqrt{\tau^2 + 1}, & \tau < 0 \end{cases},$$

jer je za konvergenciju bitno da je $|t| \leq 1$ (odnosno $|\phi| \leq \frac{\pi}{4}$). Zapišimo t ekvivalentno

$$t = sign(\tau)(-|\tau| + \sqrt{\tau^2 + 1}) = \frac{sign(\tau)}{|\tau| + \sqrt{\tau^2 + 1}},$$

odnosno t smo deracionalizirali kako se ne bi dogodilo katstrofalno kraćenje.
Sada računamo c i s.

$$c^2 + s^2 = 1$$
$$1 + t^2 = \frac{1}{c^2}$$
$$c^2 = \frac{1}{1 + t^2}$$

Kako smo za t birali manji kut, on se nalazi u desnoj poluravnini pa je njegov kosinus pozitivan te vrijedi

$$c = \frac{1}{\sqrt{1 + t^2}},$$

te konačno

$$s = ct = \frac{t}{\sqrt{1 + t^2}}.$$

S obzirom na način biranja matrica J_i vidimo da se mijenjaju samo elementi u j-tom i k-tom retku i stupcu.
Računamo dijagonalne elemente $a_{jj}^{(i+1)}$ i $a_{kk}^{(i+1)}$.

$$\begin{aligned} a_{jj}^{(i+1)} &= c^2 a_{jj}^{(i)} - 2sca_{jk}^{(i)} + s^2 a_{kk}^{(i)} \\ &= c^2 a_{jj}^{(i)} + s^2 a_{jj}^{(i)} - s^2 a_{jj}^{(i)} - 2sca_{jk}^{(i)} + s^2 a_{kk}^{(i)} \\ &= (c^2 + s^2) a_{jj}^{(i)} + s^2 (a_{kk}^{(i)} - a_{jj}^{(i)}) - 2sca_{jk}^{(i)} \end{aligned} \tag{2.3}$$

Koristeći (2.2) raspišimo izraz $s^2(a_{kk}^{(i)} - a_{jj}^{(i)})$,

$$\begin{aligned} s^2(a_{kk}^{(i)} - a_{jj}^{(i)}) &= 2a_{jk}^{(i)} s^2 \operatorname{ctg}(2\phi) \\ &= 2a_{jk}^{(i)} s^2 \frac{c^2 - s^2}{2sc} \\ &= a_{jk}^{(i)} t (c^2 - s^2) \end{aligned}$$

te izraz $2sca_{jk}^{(i)}$ zapišimo ekvivalento

$$\begin{aligned} 2sca_{jk}^{(i)} &= 2sca_{jk}^{(i)} \frac{c}{c} \\ &= 2tc^2 a_{jk}^{(i)} \end{aligned}$$

Uvrštavajući prethodna dva izraza u (2.3) dobivamo

$$\begin{aligned} a_{jj}^{(i+1)} &= a_{jj}^{(i)} + a_{jk}^{(i)} t (c^2 - s^2) - 2tc^2 a_{jk}^{(i)} \\ &= a_{jj}^{(i)} - a_{jk}^{(i)} t (s^2 - c^2 + 2c^2) \\ &= a_{jj}^{(i)} - a_{jk}^{(i)} t \end{aligned}$$

Drugu relaciju dobijemo analogno $a_{kk}^{(i+1)} = a_{kk}^{(i)} + a_{jk}^{(i)}t$.
Za vandijagonalne elemente za $p \neq j, k$ vrijedi

$$\begin{aligned}
a_{jp}^{(i+1)} &= ca_{jp}^{(i)} - sa_{kp}^{(i)} \\
a_{pj}^{(i+1)} &= ca_{pj}^{(i)} - sa_{pk}^{(i)} \\
a_{kp}^{(i+1)} &= sa_{jp}^{(i)} + ca_{kp}^{(i)} \\
a_{pk}^{(i+1)} &= sa_{pj}^{(i)} + ca_{pk}^{(i)}
\end{aligned}$$

Sada nije teško pokazati da prije i poslije transformacije vrijedi

$$(a_{jp}^{(i+1)})^2 + (a_{kp}^{(i+1)})^2 = (a_{jp}^{(i)})^2 + (a_{kp}^{(i)})^2,$$

što je rezultat korištenja ortogonalnih rotacija J_i koje čuvaju skalarni produkt. U nastavku ćemo opisati primjenu jedne Jacobijeve rotacije te osnovnu ideju Jacobijevog algoritma.

Algorithm 1 Jacobijeva rotacija

```
J = I_n
if |a_jk| ≠ 0 then
   τ = (a_kk − a_jj) / (2a_jk)
   t = sign(τ) / (τ + √(τ²+1))
   c = 1 / √(1+t²)
   s = ct
   A = R^T(j,k,φ) A R(j,k,φ)
   ukoliko želimo svojstvene vektore
   J = J R(j,k,φ)
end if
```

Uočimo da složenost izvršavanja $R^T(j,k,\phi)AR(j,k,\phi)$ (ili $JR(j,k,\phi)$) iznosi $O(n)$ jer se mijenjaju samo j-ti i k-ti redak i stupac matrice A (ili j-ti i k-ti stupac matrice J).

Algorithm 2 Jacobijev algoritam

```
while A nije dijagonalna do
   izaberi par j, k
   pozovi Jacobijevu rotaciju
end while
```

U nastavku ćemo pokazati kako treba odabrati element a_{jk} kako bi metoda konvegirala prema dijagonalnoj matrici.

Konvergencija

U ovom dijelu ćemo opisati konvergenciju. U tu svrhu ćemo promatrati kvadrat Frobeniusove norme kojeg ćemo podijeliti na dijagonalni i vandijagonalni dio. Vrijedi

$$||A||_F^2 = d(A) + 2\text{off}(A),$$

gdje je $d(A) = \sum_{i=1}^{n} a_{ii}^2$ dijagonalni, a $\text{off}(A) = \sum_{j=1}^{n-1} \sum_{k=j+1}^{n} a_{jk}^2$ vandijagonalni dio.
Kako bi matrica A bila dijagonalna, želimo njene vandijagonalne elemente svesti na nulu, odnosno $\text{off}(A) = 0$. Sljedeća lema nam pokazuje da se svakom primjenom Jacobijeve rotacije $\text{off}(A)$ monotono smanjuje.

Napomena 2.1.1. *Jacobijeva rotacija(A, j, k) označava primjenu Jacobijeve rotacije na pozicijama (j, k) i (k, j), odnosno poništava elemente a_{jk} i a_{kj}.*

Lema 2.1.2. *Neka je A' matrica dobivena nakon jedne Jacobijeve rotacije(A, j, k) za $j \neq k$. Tada vrijedi $\mathit{off}(A') = \mathit{off}(A) - a_{jk}^2$.*

Dokaz. Kako su Jacobijeve rotacije ortogonalne, vrijedi

$$a'^{\,2}_{jj} + a'^{\,2}_{kk} = \left\| \begin{bmatrix} a'_{jj} & a'_{jk} \\ a'_{kj} & a'_{kk} \end{bmatrix} \right\|_F = \left\| \begin{bmatrix} a_{jj} & a_{jk} \\ a_{kj} & a_{kk} \end{bmatrix} \right\|_F = a_{jj}^2 + a_{kk}^2 + 2a_{jk}^2, \tag{2.4}$$

Uočimo da se nakon jedne Jacobijeve rotacije mijenjaju samo dva dijagonalna elementa a_{jj} i a_{kk}, te da vrijedi

$$\begin{aligned}
2\text{off}(A') &= ||A'||_F - d(A') \\
&= ||A||_F - d(A') \\
&= ||A||_F - \sum_{p=1, p\neq j,k}^{n} a'^{\,2}_{pp} - (a'^{\,2}_{jj} + a'^{\,2}_{kk}) \\
&\overset{(2.4)}{=} ||A||_F - \sum_{p=1, p\neq j,k}^{n} a_{pp}^2 - (a_{jj}^2 + a_{kk}^2 + 2a_{jk}^2) \\
&= ||A||_F - \sum_{p=1}^{n} a_{pp}^2 - 2a_{jk}^2 \\
&= 2\text{off}(A) - 2a_{jk}^2,
\end{aligned}$$

odnosno $\text{off}(A') = \text{off}(A) - a_{jk}^2$. □

S ovim smo pokazali da vandijagonalna norma monotono pada, tj. sigurno konvergira, a na primjeru klasičnog Jacobijevog algoritma pokazat ćemo da konvergira baš u nulu.
U klasičnom Jacobijevom algoritmu element a_{jk} biramo kao najveći element po apsolutnoj vrijednosti.

Algorithm 3 Klasični Jacobijev algoritam

while $\text{off}(A)$ >tol (kriterij zaustavljanja postavljen od korinika) **do**
  izaberi par j, k takav da je $|a_{jk}|$ najveći
  pozovi Jacobijevu rotaciju(A, j, k)
end while

Teorem 2.1.3. *Poslije jedne Jacobijeve rotacije u klasičnom Jacobijevom algoritmu vrijedi* $\text{off}(A') \leq (1 - \frac{1}{N})\text{off}(A)$, *gdje je* $N = \frac{n(n-1)}{2}$ *broj vandijagonalnih elemenata gornjeg trokuta matrice A. Nakon k Jacobijevih rotacija vrijedi* $\text{off}(A^{(k)}) \leq (1 - \frac{1}{N})^k \text{off}(A)$.

Dokaz. Iz leme 2.1.2 znamo da vrijedi $\text{off}(A^{'}) = \text{off}(A) - a_{jk}^2$. Kako je a_{jk} najveći po apsolutnoj vrijednosti vandijagonalni element, vrijedi $\text{off}(A) \leq N a_{jk}^2$, odnosno
$a_{jk}^2 \geq \frac{1}{N}\text{off}(A)$. Sada imamo

$$\text{off}(A') = \text{off}(A) - a_{jk}^2 \leq (1 - \frac{1}{N})\text{off}(A).$$

Tvrdnja za k rotacija se lako pokaže indukcijom. □

S ovim smo pokazali da klasična Jacobijeva metoda konvergira linearno.
Pokazano je da klasična Jacobijeva metoda konvergira asimptotski, odnosno za dovoljno veliki i, postoji konstanta c takva da vrijedi

$$\text{off}(A^{(i+N)}) \leq c \cdot \text{off}^2(A^{(i)}) \tag{2.5}$$

tj. konvergira kvadratno. Za dokaz prethodnog svojstva je bitan izbor kuteva $|\phi| \leq \frac{\pi}{4}$, a dokaz možete pronaći u [4] ili [5].
Preostalo je još pokazati da su dijagonalni elementi dobivene matrice svojstvene vrijednosti od A. U tu svrhu ćemo primijeniti teorem 1.2.4. Ako za D uzmemo dijagonalne, a za E izvandijagonalne elemente matrice A, tada vrijedi

$$|\lambda'_{\pi(i)} - a_{ii}|^2 \leq \sum_{k=1}^{n} |\lambda'_{\pi(k)} - a_{kk}|^2 \leq 2\text{off}(A)$$

Dakle, kada $\text{off}(A) \to 0$ dijagonalni elementi konvergiraju prema nekoj permutaciji svojstvenih vrijednosti od A.

2.2 Ciklička Jacobijeva metoda

Mana klasičnog Jacobijevog algoritma je velik gubitak vremena za traženje pivotnog elementa (treba izvesti $n(n-1)/2$ usporedbu da bi utvrdili poziciju najvećeg elementa po apsolutnoj vrijednosti). Zato su razvijene cikličke metode čija je ideja sustavno poništavati sve vandijagonalne elemente po unaprijed zadanom poretku.

Ciklička strategija po retcima

Ideja cikličke strategije po retcima je sustavno poništavati pivotne elemente prolazeći gornjim trokutom redak po redak. Na primjeru 4×4 matrice ćemo pokazat redosljed biranja pivotnih elemenata:

$$\begin{bmatrix} * & \xrightarrow{1} & \xrightarrow{2} & \xrightarrow{3} \\ & * & \xrightarrow{4} & \xrightarrow{5} \\ & & * & \xrightarrow{6} \\ & & & * \end{bmatrix},$$

tj. ciklički po retcima biramo

$$(1,2),(1,3),(1,4);(2,3),(2,4);(3,4). \tag{2.6}$$

Kada jednom obiđemo sve vandijagonalne elemente, kažemo da smo napravili jedan ciklus. Cikluse ponavljamo dok svi vandijagonalni elementi ne budu dovoljno mali.
Ciklička pivotna strategija

$$\omega : N \to \{(j,k) : j \le 1 < k \le n\},\ \omega(p) = (j_p, k_p)$$

je periodična funkcija s periodom $n(n-1)/2$, gdje se osnovni period zadaje sa npr. (2.6). Ciklički Jacobijev algoritam konvergira kvadratno, odnosno vrijedi (2.5).

Slijedi prikaz cikličkog Jacobijevog algoritma po retcima.

Algorithm 4 Ciklički Jacobijev algoritam

```
J = I_n
ciklus=0
tol(kriterij zaustavljanja postavljen od korisnika)
while off(A) > tol do
   ciklus=ciklus+1
   for j = 1 : n-1 do
      for k = j+1 : n do
         if A(j,k) ≠ 0 then
            pozovi Jacobijevu rotaciju(A, j, k)
         end if
      end for
   end for
end while
```

2.3 Paralelna i blokirana Jacobijeva metoda

Paralelna Jacobijeva metoda

Kako se primjenom jedne Jacobijeve rotacije mijenjaju samo dva stupca i retka matrice A, algoritam se može izvršiti kroz nezavisne, nekonfliktne potprobleme.
Kažemo da je $(j_1, k_1), (j_2, k_2), \ldots, (j_N, k_N), N = n(n-1)/2$, paralelno raspoređivanje skupa $\{(j,k) : 1 \le j < k \le n\}$, ako za $s = 1 : n-1$ skup

$$rot.set(s) = \{(j_r, k_r) : r = 1 + n(s-1)/2 : ns/2\}$$

se sastoji od nekonflitnih rotacija.

Napomena 2.3.1. *Uočimo da iz definicije skupa $rot.set(s)$, ovo vrijedi za parne n.*

Potonje ćemo objasniti na primjeru šahovskog turnira od 8 igrača, gdje će u 7 rundi svatko sa svakim odigrati točno jednom. Izbor parova ćemo prikazati nizom sljedećih tablica. U prvoj rundi imamo 4 para, u kojem 1 igra protiv 2, 3 protiv 4 itd.

1	3	5	7
2	4	6	8

$$rot.set(1) = \{(1,2), (3,4), (5,6), (7,8)\}$$

U daljnjim rundama, igrač broj 1 će ostat na svom mjestu, dok ćemo sve ostale kružno pomaknuti za jedno mjesto.

1	2	3	5
4	6	8	7

$rot.set(2) = \{(1,4),(2,6),(3,8),(5,7)\}$

1	4	2	3
6	8	7	5

$rot.set(3) = \{(1,6),(4,8),(2,7),(3,5)\}$

1	6	4	2
8	7	5	3

$rot.set(4) = \{(1,8),(6,7),(4,5),(2,3)\}$

1	8	6	4
7	5	3	2

$rot.set(5) = \{(1,7),(5,8),(3,6),(2,4)\}$

1	7	8	6
5	3	2	4

$rot.set(6) = \{(1,5),(3,7),(2,8),(4,6)\}$

1	5	7	8
3	2	4	6

$rot.set(7) = \{(1,3),(2,5),(4,7),(6,8)\}$

Sada nije teško prethodni postupak opisati funkcijom koju ćemo iskoristiti za paralelizaciju Jacobijevog algoritma. Igrače ćemo podijeliti u dva niza $top(1:n/2)$ te $bot(1:n/2)$, gdje međusobno igraju igrači na istim pozicijama u nizu. Parove u narednim rundama ćemo dobiti pozivanjem sljedeće funkcije:

---

function: $[new.top, new.bot] = \textbf{music}(top, bot, n)$
$m = n/2$
for $j = 1 : m$ **do**
  if $j = 1$ **then**
    $new.top(j) = 1$
  else if $j = 2$ **then**
    $new.top(j) = bot(1)$
  else
    $new.top(j) = new.top(j-1)$
  end if
  if $j = m$ **then**
    $new.bot(j) = top(j)$
  else
    $new.bot(j) = new.bot(j+1)$
  end if
end for

Uočimo da se u paralelnom Jacobijevom algoritamu svaki element iz gornjeg trokuta matrice A obradi točno jednom nakon izvršavanja obje for petlje, pri čemu se l-petlja izvrši kroz $n/2$ nezavisnih potproblema.

Algorithm 5 Paralelni Jacobijev algoritam

$J = I_n$
$top = 1 : 2 : n$
$bot = 2 : 2 : n$
while off$(A) > tol$(kriterij zaustavljanja postavljen od korisnika) **do**
  for $s = 1 : n - 1$ **do**
    for $l = 1 : n/2$ **do**
      $j = min(top(l), bot(l))$
      $k = max(top(l), bot(l))$
      if $A(j, k) \neq 0$ **then**
        pozovi Jacobijevu rotaciju(A, j, k)
      end if
    end for
    $[top, bot] =$ music(top, bot, n);
  end for
end while

Blokirana Jacobijeva metoda

Pretpostavimo da je $n = rN$ te da matricu $A \in \mathbb{R}^{n\times n}$ zapišemo u sljdećem obliku

$$A = \begin{bmatrix} A_{11} & \dots & A_{1N} \\ \vdots & \ddots & \vdots \\ A_{N1} & \dots & A_{NN} \end{bmatrix},$$

gdje je svaki $A_{ij} \in \mathbb{R}^{r\times r}$.

Neka je $A_{jk} \in \mathbb{R}^{r\times r}$ podmatrica od A, te neka je $\bar{A} = \begin{bmatrix} A_{jj} & A_{jk} \\ A_{kj} & A_{kk} \end{bmatrix} \in \mathbb{R}^{2r\times 2r}$. Glavna ideja blokirane Jacobijeve metode je običnim Jacobijevim algoritmom riješiti mali $2r \times 2r$ problem, te nakon toga matričnim množenjem (sa kreiranom blok Jacobijevom rotacijom) ažurirati matricu A (odnosno samo j-te i k-te blok retke i blok stupce).

Neka je $\bar{J} \in \mathbb{R}^{2r\times 2r}$ ortogonalna matrica koja dijagonalizira $\bar{A}$, odnosno

$$\bar{J}^T \bar{A} \bar{J} = \bar{\Lambda}, \tag{2.7}$$

gdje je $\bar{\Lambda}$ dijagonalna matrica. Matricu $\bar{J}$, dobivenu običnim Jacobijevim algoritmom, podijelimo u četiri $r \times r$ podmatrice $\bar{J} = \begin{bmatrix} \bar{J}_{jj} & \bar{J}_{jk} \\ \bar{J}_{kj} & \bar{J}_{kk} \end{bmatrix}$, te svaku podmatricu ugradimo na odgovarajuće mjesto u jediničnoj matrici $I \in \mathbb{R}^{n\times n}$ kako bismo dobili blok Jacobijevu rotaciju $J \in \mathbb{R}^{n\times n}$.

Iz (2.7) znamo da su (j, k)-ti i (k, j)-ti blokovi matrice $J^T AJ$ jednaki nula. Množenjem s lijeva matricom J^T mijenjaju se samo j-ti i k-ti blok retci matrice A, dok se množenjem s desna matricom J mijenjaju j-ti i k-ti blok stupci matrice A.
Svojstveni vektori će biti stupci matrice dobivene kao produkt svih generirarnih blok rotacija J, dok će svojstvene vrijednosti biti dijagonalni elementi "dovoljno dijagonalne" matrice $J^T AJ$.
U nastavku ćemo prikazati dva algoritma:

- ciklički po retcima blokirani Jacobijev algoritam,
- paralelni blokirani Jacobijev algoritam,

čije su ideje obilaska blokova iste kao i u istoimenim inačicama običnog Jacobijevog algoritma.

Algorithm 6 Ciklički po retcima blokirani Jacobijev algoritam

---

$J = I_n$
ciklus=0
tol(kriterij zaustavljanja postavljen od korisnika)
while off$(A) > tol$ **do**
  ciklus=ciklus+1
  for j = 1 : N-1 **do**
    for k = j+1 : N **do**
      odredi matricu $\bar{R}$ koja dijagonalizira $\begin{bmatrix} A_{jj} & A_{jk} \\ A_{kj} & A_{kk} \end{bmatrix}$
      odredimo $R \in \mathbb{R}^{n \times n}$ tako da u I_n na odgovarajuća mjesta ugradimo blokove od $\bar{R}$
      $A = R^T AR$
      ukoliko želimo svojstvene vektore
      $J = JR$
    end for
  end for
end while

Algorithm 7 Paralelni blokirani Jacobijev algoritam

$J = I_n$
$top = 1 : 2 : N$
$bot = 2 : 2 : N$
while off(A) > tol(kriterij zaustavljanja postavljen od korisnika) **do**
  for s = 1 : N-1 **do**
    for l = 1 : N/2 **do**
      $j = min(top(l), bot(l))$
      $k = max(top(l), bot(l))$
      odredi matricu $\bar{R}$ koja dijagonalizira $\begin{bmatrix} A_{jj} & A_{jk} \\ A_{kj} & A_{kk} \end{bmatrix}$
      odredimo $R \in \mathbb{R}^{n\times n}$ tako da u I_n na odgovarajuća mjesta ugradimo blokove od $\bar{R}$
      $A = R^T AR$
      ukoliko želimo svojstvene vektore
      $J = JR$
    end for
    $[top, bot]$ = music(top, bot, n);
  end for
end while

Napomena 2.3.2. *U prethodnim algoritmima, prilikom izvršavanja matričnog množenja, matrice R^T i R množe i mijenjaju samo j-ti i k-ti blok redak i supac matrice A, odnosno j-ti i k-ti blok stupac od J.*

Nakon konvergencije, dijagonalni elementi matrice A su svostvene vrijednosti, dok su stupci matrice J pripadni svojstveni vektori.

Napomena 2.3.3. *Blokirani Jacobijev algoritam je brži od prethodnika jer je na računalu efikasnije izvoditi matrično množenje.*

2.4 Ocjena pogreške svojstvenih vrijednsti i svojstvenih vektora

U ovom dijelu ćemo navesti rezultate o točnosti izračunatih svojstvenih vrijednosti i svojstvenih vektora koji će pokazati da je Jacobijeva metoda točnija od drugih metoda (npr. QR metode).
Neka je $A \in \mathbb{R}^{n\times n}$ simetrična, pozitivno definitna matrica te neka je δA mala perturbacija od A u smislu $\frac{|\delta A_{ij}|}{|A_{ij}|} < \frac{\eta}{n}$ za $i, j = 1, \ldots, n$. Tada je $\frac{\|\delta A\|_F}{\|A\|_F} < \eta$. Neka su λ_i i λ_i' i-te svojstvene

vrijednosti od A i $A + \delta A$ (numerirane tako da je $\lambda_1 < \ldots < \lambda_n$). Tada standardna teorija perturbacije daje rezultat

$$\frac{|\lambda_i - \lambda_i'|}{|\lambda_i|} \leq \frac{\eta \|A\|_F}{\lambda_i} \leq \eta \|A\|_F \|A^{-1}\|_F = \eta \kappa(A),$$

gdje je $\kappa(A) \equiv \|A\|_F \|A^{-1}\|_F$ uvjetovanost matrice A.
Zapišimo $A = DHD$, gdje je $D = diag(\sqrt{A_{ii}})$ te $H_{ii} = 1$. Uglavnom je $\kappa(H) \leq \kappa(A)$ pa će nova ocjena biti barem jednaka ili bolja od gore navedene.
Sada ćemo navesti rezultate o točnosti izračunatih svojstvenih vrijednosti i svojstvenih vektora o čemu više možete pronaći u [1].

Korolar 2.4.1. *Pretpostavimo da Jacobijev algoritam konvergira te da je matrica A_M dobivena nakon M koraka (M primjena Jacobijeve rotacije) čije dijagonalne elemente uzimamo za svojstvene vrijednosti. Zapišimo $A_m = D_m H_m D_m$, gdje je D_m dijagonalna matrica, a H_m matrica sa jedinicama na dijagonali za $0 \leq m \leq M$. Neka je λ_j j-ta svojstvena vrijednost od $A = A_0$ te neka je λ_j' j-ti dijagonalni element od A_M. Tada za jediničnu grešku zaokruživanja aritmetike konačne preciznosti $\epsilon > 0$ vrijedi ocjena:*

$$\frac{|\lambda_j - \lambda_j'|}{|\lambda_j|} \leq (\epsilon \cdot M(182\sqrt{2n-4} + 104) + n \cdot tol) \max_{0 \leq m \leq M} \kappa(H_m)$$

Korolar 2.4.2. *Neka je $V = [v_1, \ldots, v_n]$ matrica jediničnih svojstvenih vektora dobivenih primjenom Jacobijevog algoritma u aritmetici konačne preciznosti (s preciznošću ϵ). Neka je $U = [u_1, \ldots, u_n]$ matrica egzaktnih svojstvenih vektora, te neka je $\bar{\kappa} \equiv \max_m \kappa(A_m)$ najveći $\kappa(A_m)$ iz bilo koje iteracije. Tada za pogrešku izračunatih svojstvenih vektora vrijedi ocjena*

$$\|v_i - u_i\|_2 \leq \frac{\sqrt{n-1}(n \cdot tol + M(182\sqrt{2n-4} + 104)\epsilon)\bar{\kappa}}{relgap\lambda_i} + 46M \cdot \epsilon,$$

gdje je $relgap\lambda_i \equiv min_{i \neq j} \frac{|\lambda_i - \lambda_j|}{\sqrt{|\lambda_i \cdot \lambda_j|}}$.

Napomena 2.4.3. *Navedeni rezultati kažu da Jacobijev algoritam daje malu relativnu pogrešku na svojstvene vrijednosti simetrične, pozitivno definitne matrice.*

2.5 Testiranje algoritama

U ovom dijelu, obrađene algoritme izrađene u MATLAB-u, primjenjujemo na konkretnim primjerima.

Napomena 2.5.1. *U svim primjerima testnu matricu generiramo formulom $A = UDU^T$, gdje je D dijagonalna matrica, a U je proizvoljno generirana ortogonalna matrica. Tada D sadrži egzakte svojstvene vrijednosti, a stupci od U su egzaktni svojstveni vektori od A. Definiramo kriterij zaustavljanja tol,a kod blokirane metode i dimenziju bloka r.*

Napomena 2.5.2. *U svim primjerima, za pogrešku izračunatih svojstvenih vrijednosti uzimamo relativnu pogrešku kao u 2.4.1 (najveću pogrešku označavat ćemo sa $\delta\lambda$). Za pogrešku svojstvenih vektora uzimamo $\|v_i - u_i\|_2$ kao u 2.4.2 (najveću pogrešku označavat ćemo sa δv.)*

U narednim primjera, na matricama malih dimenzija testiramo točnost obrađenih algoritama.

Primjer 2.5.3. $D = diag(1, 2, 13, 27)$, $tol = 1e - 10$, $r = 2$

Egzaktna vrijednost	*Klasični Jacobi*	*Ciklički po redovima Jacobi*	*Ciklički blokirani Jacobi*
1	0.999999999999999	1.000000000000003	0.999999999999999
2	2.000000000000000	2.000000000000000	1.999999999999998
13	13.00000000000000	13.00000000000000	12.99999999999999
27	27.00000000000000	27.00000000000000	26.99999999999999

	Klasični Jacobi	*Ciklički po redovima Jacobi*	*Ciklički blokirani Jacobi*
$\delta\lambda$	$5.5511e - 16$	$3.1086e - 15$	$8.1985e - 16$
δv	$5.9269e - 16$	$1.6690e - 15$	$1.6690e - 15$

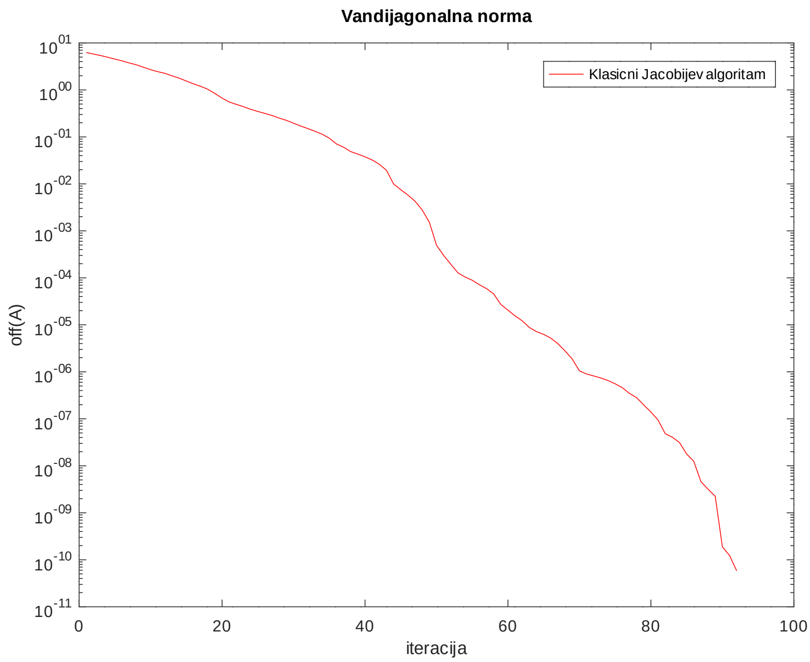
Primjer 2.5.4. $D = diag(1.00001,\ 1.00003,\ 1.00007,\ 4,\ 5,\ 6,\ 7,\ 7.00003)$, $tol = 1e-10$, $r = 2$

Egzaktna vrijednost	*Klasični Jacobi*	*Ciklički po redovima Jacobi*	*Ciklički blokirani Jacobi*
1.00001	1.000010000000000	1.000009999999999	1.000010000000004
1.00003	1.000029999999999	1.000029999999999	1.000030000000002
1.00007	1.000070000000001	1.000070000000000	1.000069999999999
4	4.000000000000000	3.999999999999999	4.000000000000006
5	5.000000000000002	5.000000000000002	5.000000000000006
6	6.000000000000008	6.000000000000006	6.000000000000017
7	7.000000000000002	7.000000000000003	7.000000000000026
7.00003	7.000030000000001	7.000030000000003	7.000030000000012

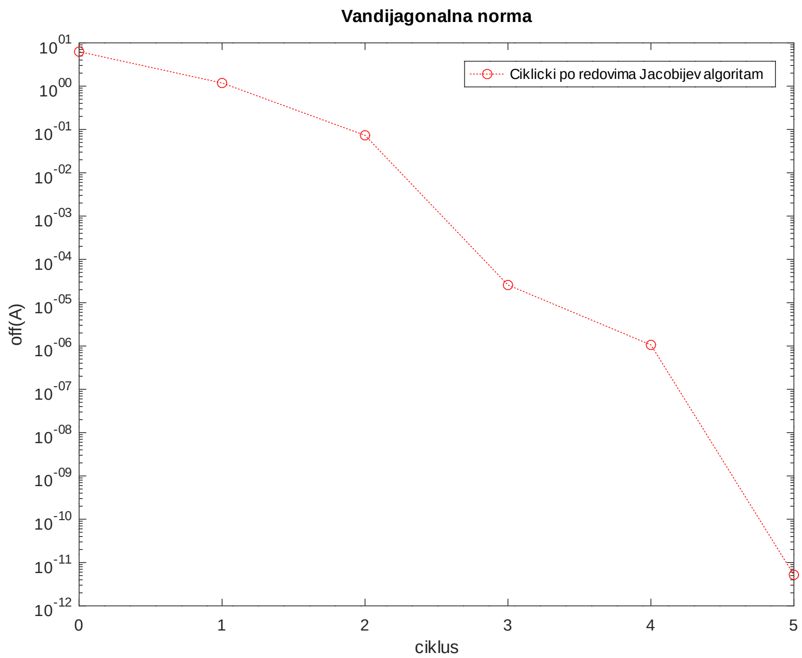
	Klasični Jacobi	*Ciklički po redovima Jacobi*	*Ciklički blokirani Jacobi*
$\delta\lambda$	$1.3323e-15$	$1.1102e-15$	$3.5527e-15$
δv	$1.3305e-09$	$6.1407e-08$	$5.6619e-11$

	Broj iteracija	*Vrijeme izvršavanja* (s)
Klasični Jacobi	91	0.27995
Ciklički po redovima Jacobi	140	0.202578
Ciklički blokirani Jacobi	24	0.395857

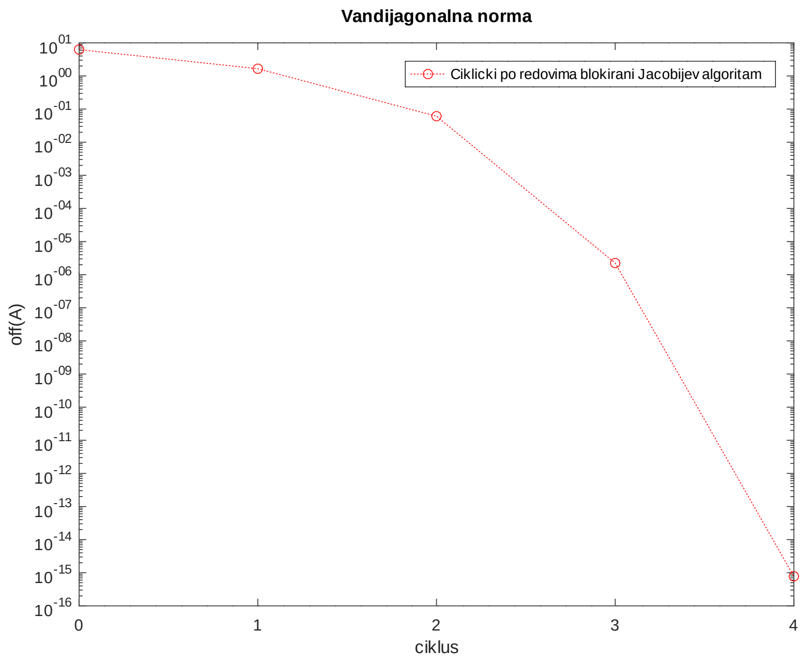
Slijede grafički prikazi vandijagonalne norme iz 2.5.4 , za koju smo u 2.1.2 pokazali da se svakom primjenom Jacobijeve rotacije monotono smanjuje. Svaki algoritam se zaustavlja kada je $\mathrm{off}(A) < tol$.



Slika 2.1: Slika 1



Slika 2.2: Slika 2



Slika 2.3: Slika 3

Iz dobivenih rezultata vidimo da je Jacobijev algoritam točan, pa čak i za jako bliske svojstvene vrijednosti. U nastavku smo na matricama većih dimenzija proučavali efikasnost algoritama. Kako se u klasičnom Jacobijevom algoritmu dosta vremena gubi na traženje pivotnog elementa, njega smo isključili iz daljnjih razmatranja.

Primjer 2.5.5. $D = diag(1.5 : 0.5 : 10,\ 11 : 1 : 50,\ 56.1 : 0.1 : 58,\ 71 : 1 : 100) \in \mathbb{R}^{100\times100}$, $tol = 1e-10$

	Broj ciklusa	*Vrijeme izvršavanja* (s)	*Prosječno vrijeme po ciklusu* (s)
Ciklički po redovima Jacobi	9	266.789	29.643
Ciklički blokirani Jacobi $r = 4$	8	283.025	35.378
Ciklički blokirani Jacobi $r = 10$	9	374.979	41.664
Ciklički blokirani Jacobi $r = 25$	6	565.202	94.200

Uočimo da odabirom velike dimenzije bloka r se vrijeme izvršavanja blokiranog Jacobijevog algoritma povećava. U potonjem slučaju, za $r = 25$ u jednom ciklusu trebamo riješiti šest $50x50$ "malih" problema običnom Jacobijevom metodom što u praksi i nije baš efikasno. Isti rezultat je dao i sljedeći primjer, kada se nakon optimalno izabrane dimenzije bloka ($r = 8$), vrijeme izvršavanja algoritma povećalo.

Primjer 2.5.6. $D = diag(1 : 1 : 256) \in \mathbb{R}^{256\times256}$, $tol = 1e-10$

	Broj ciklusa	*Vrijeme izvršavanja* (s)	*Prosječno vrijeme po ciklusu* (s)
Ciklički po redovima Jacobi	10	4876.13	487.613
Ciklički blokirani Jacobi $r = 4$	9	4047.45	449.72
Ciklički blokirani Jacobi $r = 8$	9	3361.24	373.47
Ciklički blokirani Jacobi $r = 16$	9	4463.01	495.89

Iz posljednjeg primjera vidimo kako je na matricama velikih dimenzija ciklički blokirani Jacobijev algoritam efikasniji od običnog što je rezultat efikasnijeg izvođenja matričnog množenja na računalu.

Bibliografija

[1] J. W. Demmel i K. Veselić, *Jacobi's method is more accurate than QR*, SIAM Journal on Matrix Analysis and Applications **13** (1992), 1204–1245.

[2] Zlatko Drmač, *Numerička matematika*, Skripta, 2010.

[3] G. H. Golub i C. F. Van Loan, *Matrix Computations*, The Johns Hopkins University Press, 1996.

[4] A. Schonhage, *On the Quadratic Convergence of the Jacobi Process*, Numer. Math. **6** (1964), 410–412.

[5] H.P.M. van Kempen, *On Quadratic Convergence of the Special Cyclic Jacobi Method*, Numer. Math. **9** (1966), 19–22.

Sažetak

U ovom diplomskom radu smo opisali ideju iterativne Jacobijeve metode za računanje svojstvenih vrijednosti pune simetrične matrice. U prvom poglavlju smo naveli osnovne tvrdnje o simetričnim matricama i svojstvenim vrijednostima. Drugo poglavlje započinjemo izvodom same metode i dokazujemo njenu konvergenciju. U nastavku smo obradili nekoliko različitih varijanti te ih ilustrirali algoritmima. Nakon toga smo naveli rezultate o točnosti izračunatih svojstvenih vrijednosti i vektora,a na samom kraju smo naveli rezultate dobivene testiranjem algoritama, programima izrađenim u MATLAB-u, na konkretnim primjerima iz kojih je vidljiva njihova točnost.

Summary

In this thesis we have described the idea of iterative Jacobi method for computing eigenvalues of a real symmetric matrix. In the first chapter we have named basic claims about symmetric matrices and eigenvalues. We start the second chapter with derivation of the method and we prove its convergence. We continue by describing several different variants and illustrate them with algorithms. After that, we proceed by presenting results on the accuracy of the calculated eigenvalues and eigenvectors, and at the very end we present the results obtained by testing the algorithms, by programs made using MATLAB, on specific examples that demonstrate their accuracy.

Životopis

Rođen sam 28.04.1994. u Zadru. Nakon završene gimnazije, opći smjer u Biogradu, 2013. godine upisujem preddiplomski sveučilišni studij Matematika na PMF-u kojeg uspješno završavam u rujnu 2018. Zatim upisujem diplomski studij Primijenjene matematike na matematičkom odsjeku PMF-a koji završavam s napisanim diplomskim radom o Jacobijevoj metodi za simetrični problem svojstvenih vrijednosti.