Metoda bisekcije za simetrični svojstveni problem

Beljan, Tomislav

Master's thesis / Diplomski rad

2024

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Science / Sveučilište u Zagrebu, Prirodoslovno-matematički fakultet**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:217:346411

Rights / Prava: In copyright/Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2025-02-19**

Repository / Repozitorij:

Repository of the Faculty of Science - University of Zagreb

SVEUČILIŠTE U ZAGREBU

PRIRODOSLOVNO–MATEMATIČKI FAKULTET

MATEMATIČKI ODSJEK

Tomislav Beljan

METODA BISEKCIJE ZA SIMETRIčNI SVOJSTVENI PROBLEM

Diplomski rad

Voditelj rada:
doc. dr. sc. Nela Bosner

Zagreb, Srpanj, 2024.

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. ____________________________

Mojoj obitelji koja je bila uz mene kroz uspone i padove sve ove godine. Mentorici doc. dr. sc. Nela Bosner koja je svojim trudom i strpljenjem bila velika pomoć u pisanju ovog rada.

Sadržaj

Sadržaj **iv**

Uvod **1**

1 Simetrični svojstveni problem **2**
1.1 Svojstveni problem 2
1.2 Simetrična matrica 2

2 Algoritam **7**
2.1 Pronalaženje nultočke neprekidne funkcije 7
2.2 Svojstvene vrijednosti u intervalu 8
2.3 *k*-ta najveća svojstvena vrijednost 13

3 Pronalazak svojstvenog vektora **17**
3.1 Metoda potencija 17
3.2 Rayleighev kvocijent 19
3.3 Metoda inverznih iteracija 20

4 Rezultati **21**

Bibliografija **31**

Uvod

U matematici, metoda bisekcije je tehnika pronalaženja nultočki koja se primjenjuje na neprekidne funkcije gdje su poznate dvije vrijednosti s različitim predznacima. Metoda dijeli interval u kojem postoji rješenje na manje intervale. Glavna ideja iza ove metode je da ako neprekidna funkcija mijenja predznak unutar intervala, onda mora postojati barem jedna nultočka unutar tog intervala. Iako je vrlo jednostavna i pouzdana, metoda bisekcije je relativno spora. Stoga se često koristi za dobivanje grube aproksimacije rješenja, koja se zatim može koristiti kao početna točka za brže konvergentne metode. Ova metoda također je poznata kao metoda raspolavljanja intervala, metoda binarne pretrage ili metoda dihotomije.

Procjenom funkcije na krajevima svakog intervala i određivanjem novog intervala na temelju predznaka tih vrijednosti funkcije, metoda bisekcije jamči konvergenciju barem jednom korijenu unutar početnog intervala. To je jednostavan i pouzdan pristup koji može rješavati različite vrste jednadžbi, čineći ga vrijednim alatom u računalnoj matematici.

U ovom radu ćemo primijeniti metodu bisekcije na simetrični svojstveni problem, odnosno pomoću ove metode ćemo pronaći određene dijelove spektra realne simetrične matrice A. U prvom poglavlju rada ćemo se upoznati s osnovnim pojmovima vezanim uz svojstveni problem i simetrične matrice. U drugom poglavlju razvijamo dva algoritma rješavanja simetričnog svojstvenog problema bazirana na metodi bisekcije; prvi algoritam pronalazi svojstvene vrijednosti realne simetrične matrice A u zadanom intervalu, dok drugi algoritam traži k-tu najveću svojstvenu vrijednost matrice A. U trećem poglavlju se bavimo pronalaskom svojstvenog vektora pridruženog određenoj svojstvenoj vrijednosti simetrične matrice, za potrebe čega uvodimo metodu potencija i pojam Rayleighevog koeficijenta. U četvrtom poglavlju predstavljamo implementaciju navedenih algoritama u programskom jeziku Matlab.

Poglavlje 1

Simetrični svojstveni problem

1.1 Svojstveni problem

Definicija 1.1.1. *Neka je A $n \times n$ realna matrica te neka postoji broj $\lambda \in \mathbb{C}$ te $n \times 1$ vektor x različit od nule tako da vrijedi*

$$Ax = \lambda x.$$

Vrijednost λ nazivamo ***svojstvena vrijednost matrice*** *A, a vektor x* ***(desni) svojstveni vektor matrice*** *A* ***pridružen svojstvenoj vrijednosti*** *λ. Skup svih svojstvenih vrijednosti matrice A se naziva* ***spektar matrice*** *A te se označava sa $\sigma(A)$.*

1.2 Simetrična matrica

Definicija 1.2.1. *Realna kvadratna matrica A je simetrična ukoliko je jednaka svojoj transponiranoj matrici A^T, odnosno ako vrijedi $a_{ij} = a_{ji}$ za sve i, j.*

Neka je A simetrična i $\lambda \in \sigma(A)$, tj. postoji $x \neq 0$ tako da vrijedi

$$Ax = \lambda x \tag{1.1}$$

Ako pomnožimo obje strane jednadžbe (1.1) s x^* slijeva, imamo

$$x^* Ax = \lambda x^* x \tag{1.2}$$

Jer je $x \neq 0$ vrijedi

$$x^* x = \sum_{i=1}^{n} \overline{x_i} x_i = \sum_{i=1}^{n} |x_i|^2 > 0.$$

Sada imamo
$$\lambda = \frac{x^* A x}{x^* x}.$$
Ukoliko kompleksno adjungiramo obje strane u (1.2) dobivamo:

$$x^* A^* x = \overline{\lambda} x^* x$$

A je simetrična i realna pa je $A^* = A$:

$$\overline{\lambda} = \frac{x^* A x}{x^* x}.$$

Sada je $\lambda = \overline{\lambda}$ pa vrijedi
$$\lambda \in \mathbb{R},$$

odnosno sve svojstvene vrijednosti realne simetrične matrice su realne. Realne vrijednosti možemo poredati po veličini, pa ćemo u ostatku rada koristiti sljedeću konvenciju:

$$\lambda_1(A) \geq \lambda_2(A) \geq \cdots \geq \lambda_{n-1}(A) \geq \lambda_n(A).$$

Za razvijanje metoda će nam od velike koristi biti sljedeći teorem koji nam daje interval u kojem je sadržan cijeli spektar simetrične realne matrice A:

Teorem 1.2.2. ***(Geršgorin)*** *Neka je $A \in \mathbb{R}^{n \times n}$ simetrična matrica. Definiramo R_i kao sumu apsolutnih vrijednosti nedijagonalnih elemenata matrice A u i-tom retku:*

$$R_i = \sum_{\substack{i=1 \\ i \neq j}}^{n} |a_{ij}|$$

Tada za spektar $\sigma(A)$ matrice A vrijedi:

$$\sigma(A) \subseteq \bigcup_{i=1}^{n} [a_{ii} - R_i, a_{ii} + R_i]$$

Dokaz. [3] □

Tridijagonalne matrice

U radu ćemo pokazati da se iterativne metode za rješavanje simetričnog svojstvenog problema mogu osjetno ubrzati ukoliko se matrica prethodno svede na tridijagonalni oblik, pa uvodimo pojam tridijagonalne matrice:

Definicija 1.2.3. *Matrica $T \in \mathbb{R}^{n\times n}$ je tridijagonalna ako vrijedi*

$$T_{ij} = 0 \quad \text{za } |i-j| > 1.$$

$$\begin{bmatrix} * & * & 0 & \dots & 0 \\ * & * & * & & \vdots \\ 0 & * & * & \ddots & \vdots \\ \vdots & & \ddots & \ddots & * \\ 0 & \dots & \dots & * & * \end{bmatrix}$$

Definicija 1.2.4. *Neka je zadan vektor $a \in \mathbb{R}^n$, $a \neq 0$. Neka je $H \in \mathbb{R}^{n\times n}$ ortogonalna matrica oblika*

$$H = I - \frac{1}{\gamma}vv^T, \qquad \text{gdje je } \gamma > 0,\ v \neq 0$$

za koju dodatno vrijedi

$$Ha = -\alpha e, \qquad \text{gdje je } e \in \mathbb{R}^n,\ ||e||_2 = 1 \text{ zadani vektor.}$$

Matrica H se zove ***Householderov reflektor****.*

Napomena 1.2.5.

- *Lako se može pokazati da je matrica H simetrična, odnosno da vrijedi $H^T = H$.*
- *Ukoliko stavimo $\gamma = \frac{||v||_2^2}{2}$, imamo*

$$\begin{aligned} H^TH = H^2 &= (I - \frac{2}{||v||_2^2}vv^T)(I - \frac{2}{||v||_2^2}vv^T) = \\ &= I - \frac{2}{||v||_2^2}vv^T - \frac{2}{||v||_2^2}vv^T + \frac{4}{||v||_2^4}vv^Tvv^T = \\ &= I - \frac{4}{||v||_2^2}vv^T + \frac{4}{||v||_2^2}vv^T = I \end{aligned}$$

  odnosno matrica H je ortogonalna.

Householderove reflektore možemo primijeniti na traženje tridijagonalne forme simetrične matrice A. Zapišimo A kao

$$A = \left[\begin{array}{c|c} a_{11} & a_1^T \\ \hline a_1 & A_1 \end{array}\right]$$

Ako je $a_1 \neq 0$, možemo naći Householderov reflektor H_1 takav da je

$$H_1 a_1 = -\alpha_1 \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

Ako označimo

$$P_1 = \left[\begin{array}{c|c} 1 & 0^T \\ \hline 0 & H_1 \end{array}\right],$$

imamo

$$P_1 A P_1^T = \left[\begin{array}{c|c} a_{11} & (H_1 a_1)^T \\ \hline H_1 a_1 & H_1 A_1 H_1^T \end{array}\right] = \left[\begin{array}{cc|c} a_{11} & -\alpha_1 & 0^T \\ -\alpha_1 & a_{22}^{(1)} & (a_2^{(1)})^T \\ \hline 0 & a_2^{(1)} & A_2 \end{array}\right]$$

Ako je $a_1^{(1)} = 0$, stavimo

$$H_1 = I.$$

Ako je $a_2^{(2)} \neq 0$, možemo naći Householderov reflektor H_2 takav da je

$$H_2 a_2^{(1)} = -\alpha_2 \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

Označimo

$$P_2 = \left[\begin{array}{cc|c} 1 & 0 & 0^T \\ 0 & 1 & 0^T \\ \hline 0 & 0 & H_2 \end{array}\right]$$

Sada imamo

$$P_2 P_1 A P_1^T P_2^T = \left[\begin{array}{cc|c} a_{11} & -\alpha_1 & 0^T \\ -\alpha_1 & a_{22}^{(1)} & (H_2 a_2^{(1)})^T \\ \hline 0 & H_2 a_2^{(1)} & H_2 A_2 H_2^T \end{array}\right]$$

$$
= \left[\begin{array}{ccc|c}
a_{11} & -\alpha_1 & 0 & 0^T \\
-\alpha_1 & a_{22}^{(1)} & -\alpha_2 & 0^T \\
0 & -\alpha_2 & a_{33}^{(2)} & (a_3^{(2)})^T \\
\hline
0 & 0 & a_3^{(2)} & A_3
\end{array}\right]
$$

Ponavljanjem istog postupka n-2 puta dobiva se simetrična tridijagonalna matrica. Stavimo

$$P \coloneqq P_{n-2}P_{n-3}\dots P_2P_1$$

Kako su matrice $P_1, \dots, P_{n-2}$ ortogonalne, matrica P je također ortogonalna. Slijedi da je tridijagonalna matrica PAP^T simetrična te da ima isti spektar kao i A.

Poglavlje 2

Algoritam

2.1 Pronalaženje nultočke neprekidne funkcije

Neka je zadana neprekidna funkcija $f : I \to \mathbb{R}$ te neka je $[a, b] \subseteq I$. Neka također vrijedi $f(a) \cdot f(b) < 0$. U tom slučaju postoji $c \in (a, b)$ takva da je $f(c) = 0$. Jednu od tih nultočaka možemo pronaći koristeći metodu bisekcije:

Algorithm 1 Bisekcija za nultočku funkcije

```
function [ x ] = BisectionFunctionZeros(a, b, f, tol)
    while b − a > tol do
        c = 0.5 · (a + b)
        if f(a) · f(c) > 0 then
            a = c
        else if f(a) · f(c) < 0 then
            b = c
        else
            x = c
            return
        end if
    end while
    x = 0.5 · (a + b)
end function
```

U svakoj od iteracija se interval koji sadrži nultočku prepolovi, sve dok ne bude manji od unaprijed određene tolerancije. Na kraju se kao nultočka vraća polovište zadnjeg od intervala. U slučaju da je nultočka funkcije upravo polovište jednog od intervala iz iteracija, ta vrijednost se odmah vrati i algoritam prekida izvršavanje.

2.2 Pronalaženje sv. vrijednosti simetrične matrice u zadanom intervalu

Sylvesterov teorem o inerciji

U ovom odjeljku ćemo razviti algoritam zasnivan na metodi bisekcije koji će pronalaziti svojstvene vrijednosti simetrične realne matrice A u zadanom intervalu. Bitna tvrdnja pri razvijanju algoritma je Sylvesterov teorem o inerciji, za potrebe kojeg uvodimo pojam inercije:

Definicija 2.2.1. *Inercija simetrične matrice A je trojka cijelih brojeva $Inertia(A) = (\nu, \zeta, \pi)$ takva da je ν broj negativnih svojstvenih vrijednosti od A, ζ broj svojstvenih vrijednosti od A jednakih nula, te π broj pozitivnih svojstvenih vrijednosti od A.*

Ukoliko je zadana ortogonalna matrica Q, tada su matrice Q^TAQ i A slične te imaju jednaki skup svojstvenih vrijednosti. Ukoliko Q nije ortogonalna, nego samo regularna, kažemo da su Q^TAQ i A **kongruentne**. U tom slučaju Q^TAQ i A nemaju nužno jednak skup svojstvenih vrijednosti, ali nam idući teorem govori da će njihove odgovarajuće svojstvene vrijednosti imati jednake predznake.

Teorem 2.2.2. ***(Sylvesterov teorem o inerciji)*** *Neka je A simetrična matrica i X regularna matrica. Tada A i X^TAX imaju jednaku inerciju, odnosno $Inertia(A) = Inertia(X^TAX)$.*

Dokaz. Neka je n dimenzija matrice A. Pretpostavimo da A ima ν negativnih svojstvenih vrijednosti ali da X^TAX ima $\nu' < \nu$ negativnih svojstvenih vrijednosti. Neka je $\mathbf{N}$ ν-dimenzionalni potprostor razapet svojstvenim vektorima pridruženim negativnim svojstvenim vrijednostima od A. Tada je svaki $x \in \mathbf{N}$ oblika $x = \sum \xi_i x_i$, gdje su x_i ortonormirani svojstveni vektori pridruženi negativnim svojstvenim vrijednostima od A. Vrijedi $Ax = \sum \xi_i A x_i = \sum \xi_i \lambda_i x_i$, pa iz ortonormiranosti vektora x_i slijedi $x^TAx = \left(\sum \xi_i x_i^T\right)(\sum \xi_i \lambda_i x_i) = \sum \xi_i^2 \lambda_i \|x_i\|_2^2 = \sum \xi_i^2 \lambda_i < 0$, jer je $\lambda_i < 0$ za svaki i. Neka je $\mathbf{P}$ $(n - \nu')$-dimenzionalni potprostor razapet svojstvenim vektorima pridruženim nenegativnim svojstvenim vrijednostima od X^TAX, na sličan način za svaki $x \in \mathbf{P}$ vrijedi $x^TX^TAXx \geq 0$. Jer je X regularna matrica, prostor $X\mathbf{P}$ je također $(n - \nu')$-dimenzionalan. Pošto je $dim(\mathbf{N}) + dim(X\mathbf{P}) = \nu + n - \nu' > n$, mora postojati vektor y različit od nule koji je sadržan u presjeku prostora $\mathbf{N}$ i $X\mathbf{P}$. Tada vrijedi $y^TAy < 0$ jer je $y \in \mathbf{N}$. Kako je $y \in X\mathbf{P}$, postoji $x \in \mathbf{P}$ tako da je $y = Xx$. Sada je $0 \leq x^TX^TAXx = (Xx)^TAXx = y^TAy$, što je kontradikcija. Slično se pokaže da ne može vrijediti ni $\nu < \nu'$. Slijedi da je $\nu = \nu'$, odnosno matrice A i X^TAX imaju jednak broj negativnih svojstvenih vrijednosti. Na analogan način se može pokazati da vrijedi ista tvrdnja i za brojeve pozitivnih svojstvenih vrijednosti promatranih matrica. Dalje slijedi da promatrane matrice imaju i jednak broj svojstvenih vrijednosti jednakih nuli. □

Neka je A realna simetrična matrica. Pretpostavimo da pomoću Gaussovih eliminacija možemo faktorizirati matricu $A - zI = LDL^T$, gdje je L regularna a D dijagonalna. Po prethodnom teoremu imamo $Inertia(A - zI) = Inertia(D)$. Pošto je D dijagonalna matrica, njena inercija se lako izračuna. Broj negativnih svojstvenih vrijednosti od D će biti jednostavno broj negativnih vrijednosti na dijagonali matrice D. Pošto D ima jednaku inerciju kao i $A - zI$, to će ujedno biti broj negativnih svojstvenih vrijednosti od $A - zI$, odnosno broj svojstvenih vrijednosti od A manjih od z. Analogan zaključak se dobije za broj svojstvenih vrijednosti od D jednak nuli te broj pozitivnih svojstvenih vrijednosti od D.

Pretpostavimo da želimo pronaći broj svojstvenih vrijednosti od A u intervalu $[z_1, z_2)$. To će biti jednako broju svojstvenih vrijednosti od A manjih od z_2 umanjenom za broj svojstvenih vrijednosti od A manjih od z_1. Za potrebe algoritma definiramo

$$Negcount(A, z) = \text{broj sv. vrijednosti od } A \text{ manjih od } z.$$

Slijedi da je broj svojstvenih vrijednosti od A u intervalu $[z_1, z_2)$ jednak

$$Negcount(A, z_2) - Negcount(A, z_1).$$

Algorithm 2 Metoda Bisekcije za svojstvene vrijednosti u intervalu $[a, b)$

```
n_a = Negcount(A, a)
n_b = Negcount(A, b)
if n_a = n_b then
    break   /* Nema sv. vrijednosti u zadanom intervalu */
end if
Stavi [a, n_a, b, n_b] u radnu listu   /* Radna lista se sastoji od intervala [x, y) koji
                                          sadržavaju barem jednu svojstvenu vrijednost od A */
while radna lista nije prazna do
    Makni [low, n_low, up, n_up] iz radne liste
    if up − low < tol then
        Print „U intervalu [low, up) se nalazi n_up − n_low svojstvenih vrijednosti”
    else
        mid = (low + up)/2
        n_mid = Negcount(A, mid)
        if n_mid > n_low then
            Stavi [low, n_low, mid, n_mid] u radnu listu
        end if
        if n_up > n_mid then
            Stavi [mid, n_mid, up, n_up] u radnu listu
        end if
    end if
end while
```

Kompleksnost Gaussovih transformacija na nekoj gustoj matrici A bi bila $O(n^3)$, dok u slučaju tridijagonalne simetrične matrice A situacija je puno jednostavnija:

$$A - zI = \begin{bmatrix} a_1 - z & b_1 & & \\ b_1 & a_2 - z & \ddots & \\ & \ddots & \ddots & b_{n-1} \\ & & b_{n-1} & a_n \end{bmatrix} = LDL^T$$

$$= \begin{bmatrix} 1 & & & \\ l_1 & \ddots & & \\ & \ddots & \ddots & \\ & & l_{n-1} & 1 \end{bmatrix} \cdot \begin{bmatrix} d_1 & & & \\ & \ddots & & \\ & & \ddots & \\ & & & d_n \end{bmatrix} \cdot \begin{bmatrix} 1 & l_1 & & \\ & \ddots & \ddots & \\ & & \ddots & l_{n-1} \\ & & & 1 \end{bmatrix}$$

Sada imamo

$$a_1 - z = d_1, \quad d_1 l_1 = b_1$$

Dok za $i = 2, \ldots, n$ imamo

$$l_{i-1}^2 d_{i-1} + d_i = a_i - z, \quad d_i l_i = b_i$$

Uvrstimo li $l_i = b_i / d_i$ u gornju jednadžbu, dobivamo

$$d_i = (a_i - z) - \frac{b_{i-1}^2}{d_{i-1}} \tag{2.1}$$

čime složenost određivanja dijagonalnih elemenata d_i iznosi $O(n)$.

Ukoliko dodatno definiramo $b_0 \equiv 0$ možemo zapisati algoritam za funkciju $Negcount(A, a)$:

Algorithm 3 Negcount za tridijagonalnu matricu

```
function [ k ] = Negcount(A, a)
    k = 0
    for i = 1 to n do
        d = a_i − a − b_{i−1}^2 / d
        if d < 0 then
            k = k + 1
        end if
    end for
end function
```

Pokazuje se da je metoda stabilna čak i za jako male vrijednosti d_{i-1}. [2][4][5]

Uvodimo sljedeću definiciju:

Definicija 2.2.3. *Najveća relativna greška koja se može dogoditi prilikom zaokruživanja u konačnoj aritmetici se zove* ***jedinična greška zaokruživanja****, a označava se sa* ϵ.

Lema 2.2.4. *Neka je* $|\delta_i| \leq \epsilon$, *gdje je* ϵ *jedinična greška zaokruživanja te neka je* $\rho_i = \pm 1$ *za* $i = 1 : n$ *i* $nu < 1$. *Tada vrijedi:*

$$\prod_{i=1}^{n}(1+\delta_i)^{\rho_i} = 1 + \theta_n,$$

gdje je

$$|\theta_n| \leq \frac{n\epsilon}{1-n\epsilon} =: \gamma_n$$

Dokaz. Tvrdnju dokazujemo indukcijom.

Za $n = 1$ i $\rho_1 = +1$ imamo $\theta_1 = \delta_1$, pa iz $|\delta_i| \leq \epsilon$ trivijalno slijedi

$$|\theta_1| \leq \frac{\epsilon}{1-\epsilon}.$$

Za $n = 1$ i $\rho_1 = -1$ imamo

$$\frac{1}{1+\delta_1} = 1 + \theta_1$$

$$\theta_1 = \frac{1}{1+\delta_1} - 1 = \frac{-\delta_1}{1+\delta_1}$$

$$|\theta_1| \leq \frac{\epsilon}{1-\epsilon},$$

odnosno tvrdnja vrijedi za $n = 1$. Pretpostavimo sada da tvrdnja vrijedi za neki $n - 1 \in \mathbb{N}$.

Za $\rho_n = +1$ imamo

$$\prod_{i=1}^{n}(1+\delta_i)^{\rho_i} = (1+\delta_n)(1+\theta_{n-1}) = 1 + \theta_n$$

$$\begin{aligned}
\theta_n &= \delta_n + (1+\delta_n)\theta_{n-1} \\
|\theta_n| &\leq \epsilon + (1+\epsilon)\frac{(n-1)\epsilon}{1-(n-1)\epsilon} \\
&= \frac{\epsilon\,[1-(n-1)\epsilon] + (1+\epsilon)(n-1)\epsilon}{1-(n-1)\epsilon} \\
&= \frac{n\epsilon}{1-(n-1)\epsilon} \leq \gamma_n
\end{aligned}$$

U slučaju $\rho_n = -1$ imamo

$$\frac{1+\theta_{n-1}}{1+\delta_n} = 1+\theta_n$$

$$\theta_n = \frac{1+\theta_{n-1}}{1+\delta_n} - 1 = \frac{\theta_{n-1}-\delta_n}{1+\delta_n}$$

$$|\theta_n| \leq \frac{\frac{(n-1)\epsilon}{1-(n-1)\epsilon}+\epsilon}{1-\epsilon}$$

$$= \frac{n\epsilon-\epsilon+\epsilon-(n-1)\epsilon^2}{1-\epsilon-n\epsilon+\epsilon+(n-1)\epsilon^2}$$

$$= \frac{n\epsilon-(n-1)\epsilon^2}{1-n\epsilon+(n-1)\epsilon^2} \leq \gamma_n$$

□

Lema 2.2.5. *Brojevi d_i izračunati u konačnoj aritmetici koristeći jednadžbu* (2.1) *imaju jednake predznake kao $\hat{d}_i$ izračunati egzaktno iz matrice $\hat{A}$ koja je „jako slična" matrici A:*

$$(\hat{A})_{ii} \equiv \hat{a}_i = a_i \quad i \quad (\hat{A})_{i,i+1} \equiv \hat{b}_i = b_i(1+\epsilon_i), \quad gdje\ je \quad |\epsilon_i| \leq 2.5\epsilon + O(\epsilon^2)$$

Dokaz. Neka $\tilde{d}_i$ označava vrijednosti izračunate koristeći (2.1) uključujući i greške zaokruživanja:

$$\tilde{d}_i = \left[(a_i - z)(1+\epsilon_{1,i}) - \frac{b_{i-1}^2(1+\epsilon_{2,i})}{\tilde{d}_{i-1}} \cdot (1+\epsilon_{3,i})\right](1+\epsilon_{4,i}), \tag{2.2}$$

gdje su svi epsiloni manji od jedinične greške zaokruživanja. $\epsilon_{1,i}$ označava grešku zaokruživanja kod prvog oduzimanja, $\epsilon_{2,i}$ grešku kod množenja, $\epsilon_{3,i}$ grešku kod dijeljenja i $\epsilon_{4,i}$ grešku kod drugog oduzimanja za i-ti član u (2.1). Definiramo varijable

$$\begin{aligned} \hat{d}_i &= \frac{\tilde{d}_i}{(1+\epsilon_{1,i})(1+\epsilon_{4,i})} \\ \hat{b}_{i-1} &= b_{i-1}\left[\frac{(1+\epsilon_{2,i})(1+\epsilon_{3,i})}{(1+\epsilon_{1,i})(1+\epsilon_{1,i-1})(1+\epsilon_{4,i-1})}\right]^{1/2} \equiv b_{i-1}(1+\epsilon_i) \end{aligned} \tag{2.3}$$

Očito je da $\hat{d}_i$ i $\tilde{d}_i$ imaju jednak predznak. Koristeći 2.3 i lemu 2.2.4 imamo

$$(1+\epsilon_i)^2 = \frac{(1+\epsilon_{2,i})(1+\epsilon_{3,i})}{(1+\epsilon_{1,i})(1+\epsilon_{1,i-1})(1+\epsilon_{4,i-1})}$$

$$(1+\epsilon_i)^2 = 1+\theta, \qquad \text{gdje je } |\theta| \leq \frac{5\epsilon}{1-5\epsilon}$$

$$1 + 2\epsilon_i + \epsilon_i^2 \leq 1 + \frac{5\epsilon}{1 - 5\epsilon}$$
$$2\epsilon_i \leq \frac{5\epsilon}{1 - 5\epsilon}$$
$$\epsilon_i \leq \frac{5\epsilon}{2}\left[1 + 5\epsilon + (5\epsilon)^2 + ...\right]$$
$$\epsilon_i \leq 2.5\epsilon + O(\epsilon^2).$$

Uvrstimo li (2.3) u (2.2) dobijemo

$$\hat{d}_i(1 + \epsilon_{1,i}) = (a_i - z)(1 + \epsilon_{1,i}) - \frac{\hat{b}_{i-1}^2(1 + \epsilon_{1,i})(1 + \epsilon_{1,i-1})(1 + \epsilon_{4,i-1})}{\tilde{d}_{i-1}},$$

odnosno ako uvrstimo $\tilde{d}_{i-1} = \hat{d}_{i-1}(1 + \epsilon_{1,i-1})(1 + \epsilon_{4,i-1})$ imamo

$$\hat{d}_i = (a_i - z) - \frac{\hat{b}_{i-1}^2}{\hat{d}_{i-1}}$$

čime je tvrdnja dokazana. □

2.3 Pronalaženje k-te najveće sv. vrijednosti simetrične matrice

U ovom odjeljku ćemo razviti još jednu varijantu metode bisekcije koja se također zasniva na tridijagonalnoj simetričnoj matrici. Zadana je matrica T:

$$T = \begin{bmatrix} a_1 & b_1 & 0 & \dots & 0 \\ b_1 & a_2 & b_2 & & \vdots \\ 0 & b_2 & a_3 & \ddots & \vdots \\ \vdots & & \ddots & \ddots & b_{n-1} \\ 0 & \dots & \dots & b_{n-1} & a_n \end{bmatrix} \tag{2.4}$$

Neka T_k predstavlja $k\times k$ vodeću glavnu podmatricu od T. Definiramo polinome $p_k(x) = \det(T_k - xI), \quad r = 1, \dots, n$. Razvojem po zadnjem retku matrice T imamo

$$\det(T_k - xI) = \begin{vmatrix} a_1 - x & b_1 & \dots & 0 \\ b_1 & a_2 - x & \ddots & \vdots \\ \vdots & \ddots & \ddots & b_{k-1} \\ 0 & \dots & b_{k-1} & a_k - x \end{vmatrix}$$

$$= (a_k - x) \cdot \begin{vmatrix} a_1 - x & b_1 & \dots & 0 \\ b_1 & a_2 - x & \ddots & \vdots \\ \vdots & \ddots & \ddots & b_{k-2} \\ 0 & \dots & b_{k-2} & a_{k-1} - x \end{vmatrix} - b_{k-1} \cdot \begin{vmatrix} a_1 - x & b_1 & \dots & 0 & 0 \\ b_1 & a_2 - x & \ddots & \vdots & \vdots \\ \vdots & \ddots & \ddots & b_{k-3} & 0 \\ 0 & \dots & b_{k-3} & a_{k-2} - x & 0 \\ 0 & \dots & 0 & b_{k-2} & b_{k-1} \end{vmatrix}$$

$$= (a_k - x) \cdot \begin{vmatrix} a_1 - x & b_1 & \dots & 0 \\ b_1 & a_2 - x & \ddots & \vdots \\ \vdots & \ddots & \ddots & b_{k-2} \\ 0 & \dots & b_{k-2} & a_{k-1} - x \end{vmatrix} - b_{k-1}^2 \cdot \begin{vmatrix} a_1 - x & b_1 & \dots & 0 \\ b_1 & a_2 - x & \ddots & \vdots \\ \vdots & \ddots & \ddots & b_{k-3} \\ 0 & \dots & b_{k-3} & a_{k-2} - x \end{vmatrix},$$

Odnosno dobivamo

$$p_k(x) = (a_k - x)p_{k-1}(x) - b_{k-1}^2 p_{k-2} \tag{2.5}$$

Ukoliko stavimo $p_0(x) = 1$, gornja jednadžba vrijedi za $k = 2, \dots, n$.

Budući da se $p_n(x)$ može izračunati u $O(n)$ koraka, smisleno je pronaći njene nultočke koristeći metodu bisekcije. Neka je $p_n(y)p_n(z) < 0$ te $y < z$, tada će algoritam

Algorithm 4 Metoda bisekcije za nultočke od p_n

```
function [ x ] = ZerosBisection(p_n, y, z)
    while |z - y| > tol do
        x = (y + z)/2
        if p_n(y)p_n(z) < 0 then
            z = x
        else
            y=x
        end if
    end while
    x = (y + z)/2
end function
```

pronalaziti intervale koji sadrže neku svojstvenu vrijednost $\lambda_k(T)$, a svakom iteracijom su upola manji po duljini od prethodnog.

Svojstvo Sturmovog niza

Uvodimo sljedeći teorem:

Teorem 2.3.1. ***(Svojstvo ispreplitanja)*** *Neka je* $A \in \mathbb{R}^{n \times n}$ *simetrična matrica te neka je* $A_k = A(1:k, 1:k)$*. Vrijedi:*

$$\lambda_{k+1}(A_{k+1}) \leq \lambda_k(A_k) \leq \lambda_k(A_{k+1}) \leq \cdots \leq \lambda_2(A_{k+1}) \leq \lambda_1(A_k) \leq \lambda_1(A_{k+1})$$

$$za\ k = 1 : n - 1$$

Dokaz. [6] □

Teorem 2.3.2. ***(Svojstvo Sturmovog niza)*** *Ako tridijagonalna matrica iz* (2.4) *nema nijedan element na sporednoj dijagonali jednak nuli, tada svojstvene vrijednosti od* T_{k-1} *strogo „dijele" svojstvene vrijednosti matrice* T_k*:*

$$\lambda_k(T_k) < \lambda_{k-1}(T_{k-1}) < \lambda_{k-1}(T_k) < \cdots < \lambda_2(T_k) < \lambda_1(T_{k-1}) < \lambda_1(T_k)$$

Dokaz. Iz teorema 2.3.1 slijedi da svojstvene vrijednosti od T_{k-1} „razdvajaju" svojstvene vrijednosti matrice T_r. Treba pokazati da su sve nejednakosti stroge. Pretpostavimo da je $p_k(\mu) = p_{k-1}(\mu) = 0$ za neke k i μ. Iz (2.5) i iz $b_k \neq 0$, $k = 1 : n - 1$ slijedi da je $p_{k-2}(\mu) = p_{k-3}(\mu) = \cdots = p_1(\mu) = p_0(\mu) = 0$, što je kontradikcija jer je $p_0(\mu) = 1$. Slijedi da nejednakosti moraju biti stroge. □

Označimo sada s $a(\lambda)$ broj promjena predznaka u skupu

$$\{p_0(\lambda),\ p_1(\lambda), \ldots,\ p_n(\lambda)\}.$$

Uvodimo konvenciju da $p_r(\lambda)$ ima različit predznak od $p_{r-1}(\lambda)$ ukoliko je $p_r(\lambda) = 0$. Pokaže se da je $a(\lambda)$ jednak broju svojstvenih vrijednosti od T koje su manje od λ. Dokaz tvrdnje se može pronaći u [6].

Pretpostavimo da T ima r elemenata na sporednoj dijagonali jednakih nuli. Tada je T jednaka direktnoj sumi $(r + 1)$ tridijagonalnih matrica nižeg reda:

$$C = \begin{bmatrix} T^{(1)} & & & \\ & T^{(2)} & & \\ & & \ddots & \\ & & & T^{(r+1)} \end{bmatrix}$$

Označimo red matrice $T^{(k)}$ sa m_i, tada vrijedi $\sum_i m_i = n$. Dodatno, unija spektara matrica $T^{(k)}$ čini spektar od T, te ako je x svojstveni vektor matrice $T^{(k)}$, pripadajući svojstveni vektor y matrice T je dan sa

$$y^T = (0, \dots, 0, x^T, 0, \dots, 0).$$

Stoga ćemo bez smanjenja općenitosti pretpostaviti da nijedan od elemenata na sporednoj dijagonali matrice T nije jednak nuli jer u suprotnom polazni problem razbijamo na manje sa tim svojstvom.

Pretpostavimo da želimo izračunati $\lambda_k(T)$. Definiramo $b_0 = b_n = 0$. Iz Geršgorinovog teorema 1.2.2 imamo da je $\lambda_k(T) \in [y, z]$, gdje je

$$y = \min_{1 \le i \le n} a_i - |b_i| - |b_{i-1}| \qquad z = \max_{1 \le i \le n} a_i + |b_i| + |b_{i-1}|$$

Uzimajući $[y, z]$ kao početni interval i koristeći rezultate iz Svojstva Sturmovog niza 2.3.2, dolazimo do sljedećeg algoritma:

Algorithm 5 Metoda bisekcije za $\lambda_k(T)$

function [x] = $EigenvalueBisection(a, y, z)$
  while $|z - y| > tol$ **do**
    $x = (y + z)/2$
    if $a(x) > n - k$ **then**
      $z = x$
    else
      y=x
    end if
  end while
  $x = (y + z)/2$
end function

Poglavlje 3

Pronalazak svojstvenog vektora

3.1 Metoda potencija

Za računanje svojstvenih vektora pridruženih pripadajućim svojstvenim vrijednostima koristit ćemo metodu inverznih iteracija, no u svrhu opisivanja te metode prvo uvodimo metodu potencija. Kao ulazni argument funkciji se osim matrice A predaje i neki inicijalni vektor $x_0 \neq 0$:

Algorithm 6 Metoda Potencija

---

function [x, k] = $PowerMethod(A, x_0)$
  $x = \frac{x_0}{\|x_0\|_2}$
  $k = 0$
  while nije zadovoljen kriterij zaustavljanja **do**
    $y = Ax$
    $x = \frac{y}{\|y\|_2}$
    $k = k + 1$
  end while
end function

Teorem 3.1.1. *Neka je $A \in \mathbb{C}^{n \times n}$ dijagonalizabilna matrica, za čije svojstvene vrijednosti λ_i, $i = 1 : n$ vrijedi*

$$|\lambda_1| > |\lambda_2| \geq \cdots \geq |\lambda_n|.$$

Neka su svojstveni vektori definirani kao

$$Av_i = \lambda_i v_i, \qquad \|v_i\|_2 = 1, \; i = 1 : n.$$

Pretpostavimo da zapis od x_0 u bazi svojstvenih vektora ima netrivijalnu komponentu u smjeru v_1, tada za niz x_k vrijedi

$$\lim_{k\to\infty} x_k = v_1,$$

a konvergencija ovisi o izrazu

$$\left(\frac{|\lambda_2|}{|\lambda_1|}\right)^k.$$

Dokaz. Jer je A dijagonalizabilna, može se napisati kao $A = V\Lambda V^{-1}$ gdje je $\Lambda = diag(\lambda_1, \lambda_2, \ldots, \lambda_n)$ i $V = [v_1 \ \ldots \ v_n]$ regularna matrica svojstvenih vektora. Prema tome skup $v_1, \ldots, v_n$ čini bazu prostora. Izrazimo vektor x_0 u toj bazi:

$$x_0 = c_1 v_1 + \sum_{i=2}^{n} c_i v_i$$

Jer je $c_1 \neq 0$ po pretpostavci, imamo

$$\begin{aligned} A^k x_0 &= c_1 \lambda_1^k v_1 + \sum_{i=2}^{n} c_i \lambda_i^k v_i \\ &= c_1 \lambda_1^k \left[v_1 + \sum_{i=2}^{n} \frac{c_i}{c_1} \left(\frac{\lambda_i}{\lambda_1}\right)^k v_i \right] \end{aligned}$$

Označimo x iz k-te iteracije gornjeg algoritma sa x_k. Vrijedi $x_k \in \text{span}\{A^k x_0\}$. Zato imamo da je kut između potprostora razapetih sa v_1 i x_k jednak kutu između potprostora razapetih sa v_1 i $A^k x_0$. Slijedi da za x_k vrijedi

$$\begin{aligned} \cos(\angle(x_k, v_1)) = \cos(\angle(A^k x_0, v_1)) &= \frac{|v_1^* A^k x_0|}{\|A^k x_0\|_2} \\ &= \frac{\left|c_1 \lambda_1^k\right| \left|1 + \sum_{i=2}^{n} \frac{c_i}{c_1} \left(\frac{\lambda_i}{\lambda_1}\right)^k v_1^* v_i\right|}{\left|c_1 \lambda_1^k\right| \left\|v_1 + \sum_{i=2}^{n} \frac{c_i}{c_1} \left(\frac{\lambda_i}{\lambda_1}\right)^k v_i\right\|_2} \end{aligned}$$

Jer je λ_1 dominantna i izolirana od ostatka spektra, vrijedi $\frac{\lambda_i}{\lambda_1} < 1$ za $i = 2 : n$, tj. $\lim_{k\to\infty} (\frac{\lambda_i}{\lambda_1})^k = 0$. Jer je $\|v_1\|_2 = 1$ imamo

$$\lim_{k\to\infty} \cos(\angle(x_k, v_1)) = 1 \implies \lim_{k\to\infty} \angle(x_k, v_1) = 0$$

Čime možemo zaključiti da x_k konvergira prema jediničnom svojstvenom vektoru od λ_1. □

Po teoremu 3.1.1 imamo rezultat da ako je dominantna svojstvena vrijednost dobro izolirana od ostatka spektra, metoda potencija brzo konvergira. U suprotnom, konvergencija će biti spora.

3.2 Rayleighev kvocijent

Pretpostavimo da smo izračunali aproksimaciju svojstvenog vektora w i aproksimaciju svojstvene vrijednosti μ matrice A. Razumno je za ocjenu aproksimacije uzeti normu reziduala

$$r = Aw - \mu w, \tag{3.1}$$

no kako je μ samo aproksimacija svojstvene vrijednosti, možemo i bolje. Uvodimo pojam Rayleighevog kvocijenta.

Definicija 3.2.1. *Neka je $A \in \mathbb{C}^{n\times n}$ hermitska te neka je zadan vektor $w \neq 0$. Broj*

$$R(A, w) = \frac{w^*Aw}{w^*w}$$

se naziva ***Rayleighev kvocijent****.*

Promatramo rezidual dobiven Rayleighevim kvocijentom:

$$r_R(A, w) = Aw - R(A, w)w = Aw - \frac{w^*Aw}{w^*w}w.$$

Primijetimo da ako je $x = v_i$ svojstveni vektor imamo

$$R(A, x) = \frac{v_i^*Av_i}{v_i^*v_i} = \frac{\lambda_i v_i^* v_i}{v_i^*v_i} = \lambda_i,$$

tj. Rayleighev koeficijent je tada jednak upravo svojstvenoj vrijednosti kojoj je pridružen svojstveni vektor v_i. Vijedi sljedeće:

$$w^*r_R = w^*Aw - \frac{w^*Aw}{w^*w}w^*w = 0,$$

odnosno r_R je okomit na w. Također vrijedi:

$$\begin{aligned}\|r\|_2^2 = \|Aw - \mu w\|_2^2 &= \|Aw - R(A, w)w + R(A, w)w - \mu w\|_2^2 \\ &= \|r_R + (R(A, w) - \mu)w\|_2^2 \\ &= \|r_R\|_2^2 + |R(A, w) - \mu|^2\|w\|_2^2 \geq \|r_R\|_2^2,\end{aligned}$$

gdje je posljednja jednakost dobivena iz okomitosti r_R na w. Slijedi da se u 3.1 najmanja norma reziduala za zadani w postiže upravo za $\mu = R(A, w)$. U [1] je pokazano da je pogodan kriterij zaustavljanja uvjet

$$\|r_R(A, w)\|_2 < tol$$

3.3 Metoda inverznih iteracija

Pretpostavimo sada da je matrica A dijagonalizabilna i da su joj svojstvene vrijednosti uređene na sljedeći način:

$$|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_{n-1}| > |\lambda_n| > 0,$$

odnosno najmanja svojstvena vrijednosti je po apsolutnoj vrijednosti izolirana. Tada matrica A^{-1} ima svojstvene vrijednosti:

$$\left|\frac{1}{\lambda_n}\right| > \left|\frac{1}{\lambda_{n-1}}\right| \geq \cdots \geq \left|\frac{1}{\lambda_2}\right| \geq \left|\frac{1}{\lambda_1}\right| > 0,$$

Ako metodu potencija primijenimo na A^{-1} ona će konvergirati ka svojstvenom vektoru koji pripada $\left|\frac{1}{\lambda_n}\right|$, a to je upravo v_n. Tako smo dobili svojstveni vektor najmanje po modulu svojstvene vrijednosti od A. Zbog korištenja inverza matrice A^{-1} ova metoda se naziva **metoda inverznih iteracija**.

Brzina konvergencije u ovom slučaju je određena sa

$$\frac{|\lambda_{n-1}^{-1}|}{|\lambda_n^{-1}|} = \frac{|\lambda_n|}{|\lambda_{n-1}|} < 1$$

Želimo sličan postupak primijeniti na računanje svojstvenih vektora koji ne pripadaju λ_1 ni λ_n. Ukoliko metodu inverznih iteracija primijenimo na matricu $A - \sigma I$, gdje je σ puno bliži nekoj svojstvenoj vrijednosti λ_i od bilo koje druge svojstvene vrijednosti, onda je $\lambda_i - \sigma$ najmanja svojstvena vrijednost od $A - \sigma I$ pa će metoda inverznih iteracija konvergirati prema svojstvenom vektoru v_i koji pripada λ_i. Drugim riječima, za vrlo točnu aproksimaciju svojstvene vrijednosti imamo vrlo brzu konvergenciju.

Sada možemo napisati algoritam koji će primijeniti metodu inverznih iteracija za pronalazak svojstvenog vektora pridruženog svojstvenoj vrijednosti koja je najbliža ulaznom parametru σ.

Algorithm 7 Inverzne iteracije

function [x, k] = $InverseIteration(A, \sigma, x_0, tol)$
  $x = \frac{x_0}{||x_0||_2}$
  $q = x^T A x$
  while $||Ax - qx|| > tol$ **do**
    $y = (A - \sigma I)^{-1} x$
    $x = \frac{y}{||y||_2}$
    $q = x^T A x$
  end while
end function

Poglavlje 4

Rezultati

U ovom poglavlju ćemo prezentirati rezultate dobivene ranije opisanim metodama. Opisane algoritme ćemo implementirati u programu Matlab. Algoritme ćemo izvršavati sa zadanom tolerancijom $tol = 10^{-6}$.

Za pojedini algoritam ćemo konstruirati tridijagonalne matrice sa unaprijed poznatim svojstvenim vrijednostima. Za konstrukciju takve matrice, definiramo dijagonalnu matricu D koja na dijagonali sadrži odabrani spektar. Matricu D zatim pomnožimo s nekom ortogonalnom matricom Q zdesna te sa Q^T slijeva. Na kraju funkcijom *hess* dobijemo tridijagonalnu matricu odabranog spektra.

Algoritam za pronalaženje svojstvenih vrijednosti u intervalu

Za prvi opisani algoritam ćemo konstruirati kvadratne tridijagonalne matrice reda 10 i 100. U oba slučaja ćemo namjestiti spektar da u relativno malom intervalu sadrži više svojstvenih vrijednosti, dok će ostale svojstvene vrijednosti biti „razdvojene".

Mjerit ćemo broj potrebnih iteracija za dobivanje svakog od dijelova spektra. Analizirat ćemo vrijeme izvršavanja algoritma te ga usporediti s vremenom izvršavanja QR metode sa pomakom, koja će također biti implementirana u programu Matlab, a o kojoj se više može pročitati u [3].

Generiramo matricu reda 10 sa sljedećim svojstvenim vrijednostima:

Slika 4.1: Sortirane svojstvene vrijednosti matrice *A* reda 10

```
>> sort(eig(A),'descend')
ans =

   14.0000
   12.0000
   10.5000
    8.0000
    6.0000
    3.4916
    3.4696
    3.4615
    3.4592
    3.4544
```

Prvo tražimo svojstvene vrijednosti u intervalu $[10, 11]$, gdje očekujemo jednu svojstvenu vrijednost, a zatim tražimo svojstvene vrijednosti u intervalu $[3, 4]$, u kojem očekujemo pet svojstvenih vrijednosti.

Slika 4.2: Izračunate svojstvene vrijednosti matrice A reda 10 i vremena izvršavanja, interval $[3, 4]$

```
Bisection: n=10, interval=[3,4]:
interval_eigenvalues =

   3.5296   3.4955   3.4924   3.4881   3.4685

Number of iterations: 83
Elapsed time is 0.0327208 seconds.


QR: n=10, interval=[3,4]:
interval_eigenvalues =

   3.5296   3.4685   3.4955   3.4881   3.4924

Number of iterations: 9
Elapsed time is 0.0119641 seconds.
```

Slika 4.3: Izračunate svojstvene vrijednosti matrice A reda 10 i vremena izvršavanja, interval $[10, 11]$

```
Bisection: n=10, interval=[10,11]:
interval_eigenvalues = 10.500
Number of iterations: 21
Elapsed time is 0.00811505 seconds.


QR: n=10, interval=[10,11]:
interval_eigenvalues = 10.500
Number of iterations: 9
Elapsed time is 0.0114141 seconds.
```

Vidimo da je u slučaju intervala $[3, 4]$ koji sadrži pet svojstvenih vrijednosti metoda bisekcije gotovo tri puta sporija od implementirane QR metode. Razlog tome je relativno mali red matrice A te činjenica da se u intervalu nalazi pet svojstvenih vrijednosti, pa radna lista u metodi bisekcije sadrži pet različitih intervala koje mora „prepolavljati" do zadane preciznosti, što utječe na brzinu izvršavanja. Također možemo uočiti da je, zbog istih razloga, broj iteracija u metodi bisekcije 83 što je znatno više nego broj iteracija u QR metodi sa pomakom, koji iznosi 9.

U slučaju intervala $[10, 11]$ koji sadrži samo jednu svojstvenu vrijednost možemo vidjeti da metoda bisekcije ima manje vrijeme izvršavanja. Smanjeno vrijeme izvršavanja metode bisekcije u odnosu na prethodni slučaj je očekivano jer u ovom slučaju radna lista sadrži samo jedan interval koji se u svakoj iteraciji smanjuje do zadane preciznosti. Također je očekivan manji broj iteracija u metodi bisekcije koji u ovom slučaju iznosi 21, što je znatno manje od prethodnog broja iteracija 83.

QR metoda u oba slučaja ima slična vremena izvršavanja te isti broj iteracija. To je očekivan rezultat jer QR metoda za razliku od metode bisekcije pronalazi cijeli spektar matrice A te potom ispisuje samo one vrijednosti koje se nalaze u zadanom intervalu.

Sada generiramo matricu reda 50 sa sljedećih 10 najvećih svojstvenih vrijednosti:

Slika 4.4: 10 najvećih svojstvenih vrijednosti matrice A reda 50

```
>> sort(eig(A),'descend')(1:10)
ans =

   14.0000
   12.0000
   10.5000
    8.0000
    3.5337
    3.5271
    3.5183
    3.4920
    3.4723
    2.5462
```

Slično kao i prije, tražit ćemo svojstvene vrijednosti u intervalu [10, 11] gdje očekujemo jednu svojstvenu vrijednost, a zatim u intervalu [3, 4], u kojem očekujemo pet svojstvenih vrijednosti.

Slika 4.5: Izračunate svojstvene vrijednosti matrice A reda 50 i vremena izvršavanja, interval $[3, 4]$

```
Bisection: n=50, interval=[3,4]:
interval_eigenvalues =

   3.5488   3.5306   3.5086   3.4679   3.4670

Number of iterations: 83
Elapsed time is 0.0945082 seconds.


QR: n=50, interval=[3,4]:
interval_eigenvalues =

   3.5488   3.5306   3.5086   3.4670   3.4679

Number of iterations: 49
Elapsed time is 0.907597 seconds.
```

Slika 4.6: Izračunate svojstvene vrijednosti matrice A reda 50 i vremena izvršavanja, interval $[10, 11]$

```
Bisection: n=50, interval=[10,11]:
interval_eigenvalues = 10.500
Number of iterations: 21
Elapsed time is 0.028939 seconds.


QR: n=50, interval=[10,11]:
interval_eigenvalues = 10.500
Number of iterations: 49
Elapsed time is 0.883777 seconds.
```

Za matricu reda 50 možemo uočiti dulja vremena izvršavanja za obje metode. To je očekivan rezultat zbog povećane dimenzije matrice A.

U slučaju intervala $[3,4]$ možemo uočiti znatno brže vrijeme izvršavanja metode bisekcije u odnosu na QR metodu, dok je razlika u brzini izraženija u slučaju intervala $[10,11]$.

Razlog smanjenog vremena izvršavanja za metodu bisekcije u intervalu $[10,11]$ u odnosu na interval $[3,4]$ je isti kao i u prethodnom slučaju, metoda se brže izvršava u slučaju manjeg broja svojstvenih vrijednosti u zadanom intervalu. Broj iteracija metode bisekcije za pojedini interval je isti kao u slučaju matrice reda 10.

QR metoda računa cijeli spektar matrice A čiji je red sada veći, što je razlog povećanog broja iteracija pa i samog vremena izvršavanja.

Algoritam za pronalaženje k-te najveće svojstvene vrijednosti

Za drugi algoritam ćemo konstruirati tridijagonalnu matrice reda 100. U ovom slučaju će matrica imati jednu izdvojenu svojstvenu vrijednost, koja će ujedno biti i najveća po apsolutnoj vrijednosti. Također će imati više svojstvenih vrijednosti sadržanih u relativno malom intervalu:

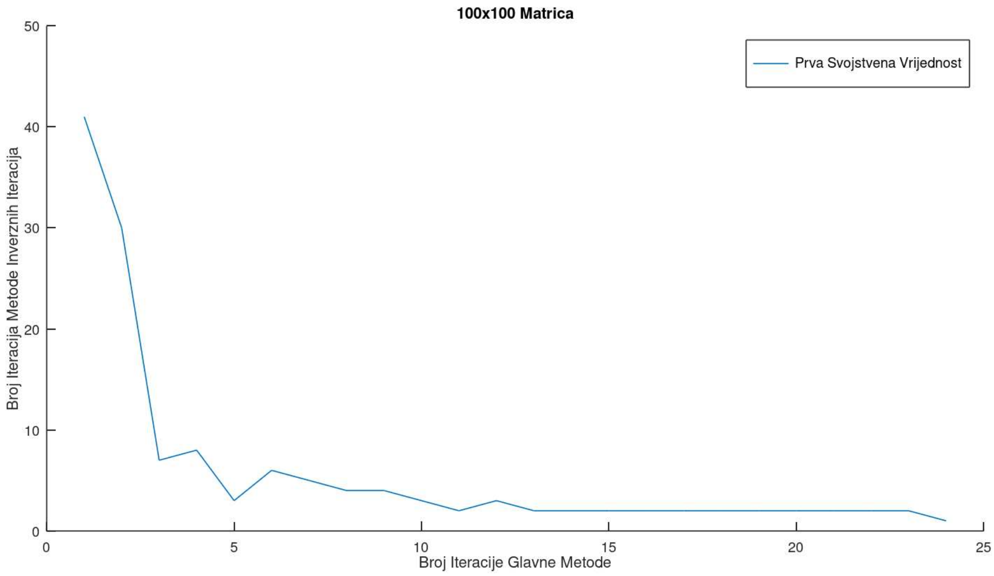
Slika 4.7: 10 najvećih svojstvenih vrijednosti matrice A

```
>> sort(eig(A),'descend')(1:10)
ans =

   7.0000
   3.9272
   3.9253
   3.9080
   3.9000
   3.8520
   2.8912
   2.7940
   2.7636
   2.6656
```

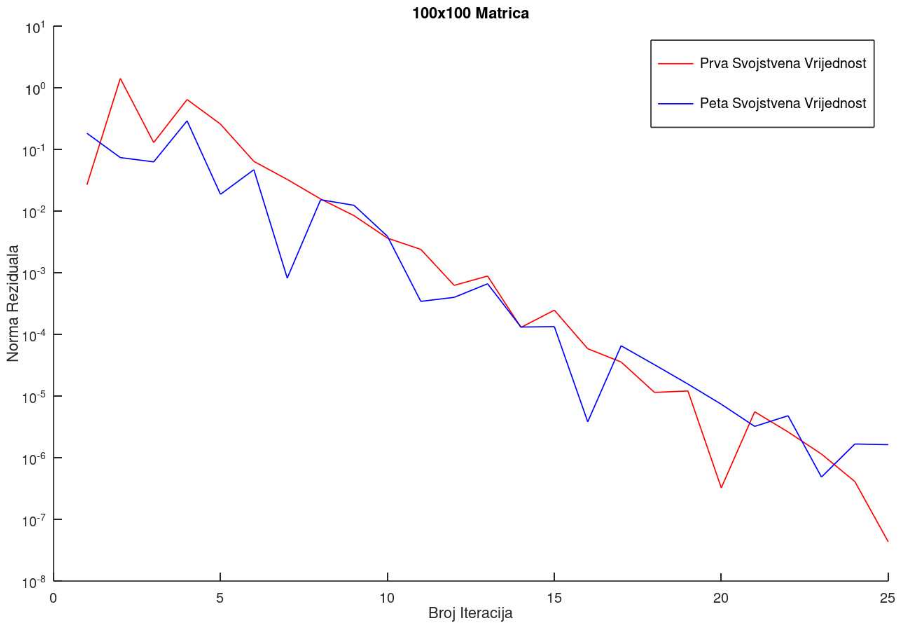
Algoritmom ćemo pokušati pronaći prvu i petu najveću svojstvenu vrijednost matrice A. Mjerimo broj iteracija te norme reziduala pojedine iteracije u oba slučaja. Normu reziduala računamo kao $\|Av-\sigma v\|_2$, gdje je σ trenutna aproksimacija svojstvene vrijednosti, a v aproksimacija svojstvenog vektora izračunata metodom inverznih iteracija.

Slika 4.8: Broj iteracija metode inverznih iteracija



Povećanjem broja iteracija glavne metode dobiva se sve bolja aproksimacija svojstvene vrijednosti, pa se smanjuje i broj iteracija metode inverznih iteracija.

Slika 4.9: Norme reziduala za prvu i petu najveću svojstvenu vrijednost matrice A



Broj iteracija je u oba slučaja jednak 25, što je i očekivano jer da bi se postigla određena preciznost određena zadanom tolerancijom, potrebno je isti broj puta „prepoloviti" početni interval. Također možemo uočiti da nema značajne razlike u normama reziduala u pojedinoj iteraciji.

Bibliografija

[1] Nela Bosner i Sanja Singer, *Numerička analiza 10. predavanje*, 2011, https://web.math.pmf.unizg.hr/~nela/nadpredavanja/10nb.pdf.

[2] James W Demmel, *Applied numerical linear algebra*, SIAM, 1997.

[3] Gene H Golub i Charles F Van Loan, *Matrix computations*, Johns Hopkins University Press, 1996.

[4] William Gragg, *On computing accurate singular values and eigenvalues of acyclic matrices*, (1992).

[5] William Kahan, *Accurate eigenvalues of a symmetric tri-diagonal matrix*, Stanford University, 1966.

[6] James Hardy Wilkinson, *The Algebraic Eigenvalue Problem*, Clarendon Press - Oxford, 1965.

Sažetak

Tema ovog rada je metoda bisekcije primijenjena na simetrični svojstveni problem.

Iako se u prvotnom obliku koristi za pronalaženje nultočki neprekidne funkcije, u ovom radu smo primijenili metodu na pronalaženje svojstvenih vrijednosti realne simetrične matrice. Simetričnu matricu smo prvo sveli na tridijagonalni oblik, a zatim razvili metode koje koriste tridijagonalnu strukturu za brzo rješavanje problema.

Rad je podijeljen u četiri poglavlja. U prvom poglavlju uvodimo neke osnovne pojmove i rezultate vezane uz simetrični svojstveni problem. U drugom poglavlju razvijamo dva algoritma za pronalaženje svojstvenih vrijednosti realne tridijagonalne matrice. Prvi algoritam pronalazi sve svojstvene vrijednosti u nekom unaprijed zadanom intervalu, dok drugi algoritam pronalazi k-tu najveću svojstvenu vrijednost matrice. U trećem poglavlju se bavimo računanjem svojstvenog vektora pridruženog ranije izračunatoj svojstvenoj vrijednosti. Za računanje svojstvenog vektora se koristi metoda inverznih iteracija. U zadnjem poglavlju iznosimo neke rezultate koji dobivamo primjenom navedenih algoritama.

Summary

The topic of this thesis is the bisection method applied to the symmetric eigenvalue problem.

Although the bisection method is originally used to find the roots of a continuous function, in this paper we use the bisection method to find eigenvalues of a real symmetric matrix. We first reduce the symmetric matrix to tridiagonal form, and then develop methods that utilize the tridiagonal structure for efficient problem solving.

The thesis is divided into four chapters. In the first chapter, we introduce some basic concepts and results related to the symmetric eigenvalue problem. In the second chapter, we develop two algorithms for finding eigenvalues of a real tridiagonal matrix. The first algorithm identifies all eigenvalues within a specified interval, while the second algorithm finds the k-th largest eigenvalue of the matrix. In the third chapter, we address the computation of the corresponding eigenvector for previously computed eigenvalues. The method of inverse iteration is used to compute this eigenvector. In the final chapter, we present some results obtained by applying the aforementioned algorithms.

Životopis

Rođen sam 28. svibnja 1993. godine u Zagrebu. Svoje školovanje počinjem u OŠ Rugvica, tijekom kojeg sam redovno nastupao na općinskim i županijskim natjecanjima iz matematike. Školovanje nastavljam u prirodoslovno-matematičkoj XV. gimnaziji u Zagrebu. 2016. godine upisujem preddiplomski sveučilišni studij Matematika na Prirodoslovno-matematičkom fakultetu u Zagrebu. Nakon završenog preddiplmskog studija, 2020. godine upisujem diplomski studij Primijenjena matematika.

Za vrijeme studiranja sam radio u Hrvatskim tvrtkama kao što su Photomath i AI technologies, a od ljeta 2023. godine radim u tvrtci Vestigo.