

## Calculul iterativ al unui vector propriu. Metoda puterii. Metoda puterii inverse

Fie matricea  $\mathbf{A} \in \mathbb{R}^{n \times n}$  având spectrul de valori proprii

$$\lambda(\mathbf{A}) = \{\lambda_1, \lambda_2, \dots, \lambda_n\}$$

și fie

$$\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$$

un set de vectori proprii, de normă euclidiană unitară, ai matricei  $\mathbf{A}$  (i.e.  $\mathbf{A}\mathbf{x}_i = \lambda_i \mathbf{x}_i$ ,  $\|\mathbf{x}_i\| = 1$ ,  $i = 1 : n$ ). Proprietățile din următoarea propoziție se vor dovedi utile în demersul calculativ al valorilor și vectorilor proprii.

**Propoziție.** Cu notațiile de mai sus avem

a) pentru orice  $\mu, \mu \in \mathbb{C}$

$$\lambda(\mathbf{A} - \mu \mathbf{I}) = \{\lambda_1 - \mu, \lambda_2 - \mu, \dots, \lambda_n - \mu\};$$

b) pentru orice  $k \in \mathbb{N}$

$$\lambda(\mathbf{A}^k) = \{\lambda_1^k, \lambda_2^k, \dots, \lambda_n^k\};$$

c) pentru orice  $\mu \in \mathbb{C} \setminus \lambda(\mathbf{A})$

$$\lambda((\mathbf{A} - \mu \mathbf{I})^{-1}) = \left\{ \frac{1}{\lambda_1 - \mu}, \frac{1}{\lambda_2 - \mu}, \dots, \frac{1}{\lambda_n - \mu} \right\}$$

și, în același timp, vectorii își păstrează semnificația, adică sunt vectori proprii pentru matricele  $\mathbf{A} - \mu \mathbf{I}$ ,  $\mathbf{A}^k$ ,  $(\mathbf{A} - \mu \mathbf{I})^{-1}$  respectiv

$$(\mathbf{A} - \mu \mathbf{I}) \mathbf{x}_i = (\lambda_i - \mu) \mathbf{x}_i$$

$$\mathbf{A}^k \mathbf{x}_i = \lambda_i^k \mathbf{x}_i$$

$$(\mathbf{A} - \mu \mathbf{I})^{-1} \mathbf{x}_i = \frac{1}{\lambda_i - \mu} \mathbf{x}_i$$

pentru toți  $i = 1 : n$ .

*Demonstrația* este imediată prima relație este evidentă, cea de-a doua se obține prin inducție, iar ultima se obține, pentru  $\mathbf{A} - \mu \mathbf{I}$  nesară.

Presupunem acum că mulțimea este ordonată în sensul descrescător al valorilor absolute și că matricea  $\mathbf{A}$  are o valoare proprie dominantă, i.e.

$$|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \dots \geq |\lambda_n|.$$

Dacă  $\mathbf{y} \in \mathbb{C}^n$  este un vector de normă euclidiană unitară având o componentă nenulă pe direcția vectorului propriu  $\mathbf{x}_1 \in \mathbf{X}$ , definim șirul vectorial  $(\mathbf{y}^{(k)})_{k \in \mathbb{N}}$  și șirul numeric  $(\lambda^{(k)})_{k \in \mathbb{N}}$  prin relațiile

1.  $\mathbf{y}^{(0)} = \mathbf{y}$     % inițializare
2. Pentru  $k = 1, 2, \dots$     % iterare
  1.  $\mathbf{z} \leftarrow \mathbf{A} \cdot \mathbf{y}^{(k-1)}$

$$\begin{aligned} 2. \mathbf{y}^{(k)} &\leftarrow \frac{\mathbf{z}}{\|\mathbf{z}_2\|_2} \\ 3. \lambda^{(k)} &\leftarrow (\mathbf{y}^{(k)})^T \mathbf{A} \mathbf{y}^{(k)} . \end{aligned}$$

care definesc metoda puterii.

$$\mathbf{y}^{(k)} = \rho_k \mathbf{A}^k \mathbf{y}^{(0)}, \quad k = 1, 2, \dots$$

unde  $\rho_k = \frac{1}{\|\mathbf{A}^k \mathbf{y}^{(0)}\|}$  este un factor scalar de normare. Dacă

$$\mathbf{y}^{(0)} = \sum_{i=1}^n \alpha_i \mathbf{x}_i$$

cu  $\alpha_1 \neq 0$ , atunci

$$\mathbf{y}^{(k)} = \rho_k \alpha_1 \lambda_1^k \left[ \mathbf{x}_1 + \sum_{i=2}^n \frac{\alpha_i}{\alpha_1} \left( \frac{\lambda_i}{\lambda_1} \right)^k \mathbf{x}_i \right] .$$

Datorită condiției de dominanță avem

$$\lim_{k \rightarrow \infty} \left( \frac{\lambda_i}{\lambda_1} \right)^k = 0, \quad i = 2 : n,$$

ceea ce asigură convergența șirului vectorial  $\mathbf{y}^{(k)}$  către vectorul propriu unitar asociat valorii proprii dominante

$$\mathbf{y}^{(\infty)} = \lim_{k \rightarrow \infty} \mathbf{y}^{(k)} = \mathbf{x}_1 .$$

Viteza de convergență este cu atât mai mare cu cât raportul  $\frac{|\lambda_2|}{|\lambda_1|}$  este mai mic, metoda puterii fiind performantă în cazul matricelor cu o valoare proprie net dominantă. Șirul numeric  $(\lambda^{(k)})_{k \in \mathbb{N}}$ , converge către valoarea proprie dominantă. În principiu, în cadrul metodei puterii, nu este necesară construcția explicită a șirului  $(\lambda^{(k)})_{k \in \mathbb{N}}$ , valoarea proprie  $\lambda_1$  putând fi calculată cu

$$\lambda_1 = (\mathbf{y}^{(\infty)})^H \mathbf{A} \mathbf{y}^{(\infty)} = \mathbf{x}_1^H \mathbf{A} \mathbf{x}_1 .$$

În schimb o astfel de evaluare devine foarte benefică în cadrul unei variante a metodei puterii inverse.

Metoda puterii inverse de determinare iterativă a unui vector propriu se bazează pe relația conform căreia matricea  $\mathbf{B} = (\mathbf{A} - \mu \mathbf{I})^{-1}$  va avea o valoare proprie net dominantă dacă, "deplasarea"  $\mu$  este o aproximație, chiar și grosieră, a unei valori proprii distincte a matricei  $\mathbf{A}$ . Metoda puterii inverse, este, în fapt metoda puterii aplicată matricei  $\mathbf{B}$  de mai sus și se definește

$$\begin{aligned} 1. \mathbf{y}^{(0)} &= \mathbf{y} \quad \% \text{ inițializare} \\ 2. \text{ pentru } k &\leftarrow 1, 2, \dots \% \text{ iterare} \end{aligned}$$

$$1. \text{ Se rezolvă sistemul liniar } (\mathbf{A} - \mu \mathbf{I})\mathbf{z} = \mathbf{y}^{(k-1)}$$

$$2. \mathbf{y}^{(k)} \leftarrow \frac{\mathbf{z}}{\|\mathbf{z}\|}$$

$$3. \lambda^{(k)} \leftarrow (\mathbf{y}^{(k)})^T \mathbf{A} \mathbf{y}^{(k)} .$$

Conform celor arătate la metoda puterii, șirul  $(\mathbf{y}^{(k)})_{k \in \mathbb{N}}$  este convergent către vectorul propriu al matricei  $\mathbf{A}$  asociat valorii proprii celei mai apropiate de deplasarea  $\mu$ . Proprietăți de convergență deosebite se obțin în varianta în care deplasarea  $\mu$  este modificată de la o iterație la alta utilizând aproximația curentă  $\mu_k = \lambda^{(k)}$  a unei valori proprii a matricei  $\mathbf{A}$ . Această variantă, cunoscută sub denumirea de *iterarea cântului Rayleigh*, este definită de vectorul inițial de normă unitară  $\mathbf{y}$ , deplasarea inițială

$$\mu = \mu_0 = \mathbf{y}^T \mathbf{A} \mathbf{y}$$

și procedura în care  $\mu$  se înlocuiește cu  $\mu_k = \lambda^{(k)}$ . Dacă inițializarea  $\mathbf{y}$  nu este ortogonală pe direcția vectorului propriu asociat valorii proprii celei mai apropiate de  $\mu_0$  (condiție generic satisfăcută de o alegere aleatoare a lui  $\mathbf{y}$ ), atunci se poate arăta că șirul are o convergență pătratică către vectorul propriu corespunzător, adică există o constantă  $\rho > 0$  astfel încât

$$\|\mathbf{y}^{(k)} - \mathbf{y}^{(\infty)}\| \leq \rho \cdot \|\mathbf{y}^{(k-1)} - \mathbf{y}^{(\infty)}\|^2 ,$$

ceea ce înseamnă o viteză de convergență foarte bună. Mai mult, pentru matrice hermitice (simetrice) se poate demonstra că se asigură chiar o convergență cubică.

Aceste excelente proprietăți de convergență sunt moderate de efortul de calcul sporit, în raport cu metoda puterii, necesare pentru rezolvarea unui sistem liniar la fiecare iterație. Totuși, dacă matricea  $\mathbf{A}$  are o structură particulară, cum ar fi de exemplu forma Hessenberg, metoda puterii inverse, cu alegerea deplasării egală cu cântul Rayleigh, este considerată ca fiind cea mai performantă. Deplasări de forma  $\mathbf{A} \rightarrow \mathbf{A} - \mu \mathbf{I}$ , destinate sporirii vitezei de convergență, pot fi utilizate, în principiu și în cazul metodei puterii, însă cu o eficiență mult mai redusă.

Formele concrete ale algoritmilor pentru implementarea metodelor puterii și puterii inverse se obțin prin completarea schemelor de calcul (76), (83) cu trunchierea șirurilor respective pe baza unor criterii de oprire a iterațiilor cum ar fi

$$\left| 1 - \left( \mathbf{y}^{(k-1)} \right)^T \mathbf{y}^{(k)} \right| < \varepsilon ,$$

$\varepsilon$  este o toleranță dată, care consideră aproximația drept satisfăcătoare atunci când doi vectori succesivi sunt “suficient de coliniari” (realizarea trunchierii când norma diferenței a două aproximații succesive scade sub o toleranță dată poate să nu dea rezultate: dacă sensul vectorului se schimbă la fiecare pas această normă tinde către 2). Dacă acest criteriu nu este atins după un număr maxim admis de iterații, se consideră că metoda nu asigură convergența dorită.

Cu aceste precizări putem să prezentăm algoritmii formali de implementare a metodei puterii și a metodei puterii inverse.

### Metoda puterii directe

```
[λ,x,OK]=MPD(A, y, maxit, ε)
% Intrări:
%   A = matricea dată;
%   y = aproximația inițială a vectorului propriu;
```

```

%      ε = toleranța impusă;
%      maxit = numărul maxim de iterații.
% Ieșiri:
%      λ = valoarea proprie dominantă;
%      x = vectorul propriu asociat.
%      OK = indicator succes / esec
OK=1;
k=0;
y=y/norm(y);
e=2*ε;
while e>=ε
    k=k+1;
    if k>maxit
        OK=0;
        error('nu s-a atins precizia impusa');
    end
    x=A*y;
    x=x/norm(x);
    e=abs(1-abs(y'*x));
    y=x;
end
λ=x'*A*x;

```

#### Metoda puterii inverse cu deplasare Rayleigh

```

[λ,x,OK]=MPI(A,y,maxit,ε)
% Intrări:
%      A = matricea dată;
%      y = aproximația inițială a vectorului propriu;
%      ε = toleranța impusă;
%      maxit = numărul maxim de iterații.
% Ieșiri:
%      λ = valoarea proprie dominantă;
%      x = vectorul propriu asociat.
%      OK = indicator succes / esec
k=0;
y=y/norm(y);
λ=y'*A*y;
E=2*ε;
while e>=ε
    k=k+1;
    if k>maxit
        OK=0;
        error('nu s-a atins precizia impusa');
    end
    x=(A-λ)\y;
    x=x/norm(x);
    e=abs(1-abs(y'*x));
    y=x;
    λ=x'*A*x;
end

```

Pentru a asigura eficiența necesară algoritmului, rezolvarea sistemului liniar trebuie să exploateze eventuale particularități structurale ale matricei sistemului.

Algoritmii prezentați sunt destinați calculului unui singur vector propriu și al valorii proprii asociate. Dacă se dorește calculul mai multor valori și vectori proprii pentru o matrice dată, acești algoritmi trebuie completați cu o procedură de deflație. Principiile unei astfel de proceduri sunt date în demonstrația teoremei 9.4 și sunt utilizate implicit în cadrul algoritmului **QR** prezentat ulterior.

În încheierea acestui paragraf, considerăm necesar să subliniem faptul că cea mai bună metodă de calcul a unui vector propriu, în absența oricărei informații inițiale despre valoarea proprie asociată, este metoda puterii inverse cu deplasarea dată de câțul Rayleigh.

### Deflația valorilor proprii.

Presupunem că am localizat o pereche dominantă  $(\lambda_1, \mathbf{x}_1)$  și dorim să continuăm procesul pentru celelalte perechi proprii, presupunând că acestea sunt distincte. Situația este aplicabilă matricelor simetrice cu elemente reale.

$$\lambda(\mathbf{A}) = \{\lambda_1, \lambda_2, \dots, \lambda_n\}$$

$$\mathbf{X}(\mathbf{A}) = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$$

Dacă formăm matricea:

$$\mathbf{B} = (\mathbf{I}_n - \mathbf{x}_1 \mathbf{y}^T) \mathbf{A}, \quad \text{cu} \quad \mathbf{y}^T \mathbf{x}_1 = 1$$

atunci:

$$\lambda(\mathbf{B}) = \{0, \lambda_2, \dots, \lambda_n\}$$

$$\mathbf{X}(\mathbf{B}) = \{\mathbf{x}_1, \mathbf{z}_2, \dots, \mathbf{z}_n\}$$

în care:

$$\mathbf{z}_i = \mathbf{x}_i - \mathbf{x}_1 \mathbf{y}^T \mathbf{x}_i = \mathbf{x}_i - \mathbf{y}^T \mathbf{x}_i \mathbf{x}_1, \quad i=2:n$$

*Demonstrație:* Mai întâi vom alege vectorul  $\mathbf{y}$  colinar cu  $\mathbf{x}$  (sunt și alte opțiuni posibile).

$$\mathbf{y} = \alpha \cdot \mathbf{x}, \quad \mathbf{y}^T \mathbf{x} = \alpha \cdot \mathbf{x}^T \mathbf{x} = \alpha \cdot \|\mathbf{x}\|_2^2 = 1 \Rightarrow \alpha = \frac{1}{\|\mathbf{x}\|_2^2} \Rightarrow \mathbf{y} = \frac{\mathbf{x}}{\|\mathbf{x}\|_2^2}$$

$$\mathbf{B} \cdot \mathbf{x}_1 = (\mathbf{I}_n - \mathbf{x}_1 \mathbf{y}^T) \mathbf{A} \cdot \mathbf{x}_1 = \lambda_1 \mathbf{x}_1 - \lambda_1 \mathbf{x}_1 \mathbf{y}^T \mathbf{x}_1 = 0$$

ceea ce arată că  $\mathbf{B}$  are valoarea proprie 0 și vectorul propriu corespunzător  $\mathbf{x}_1$ .

$$\mathbf{B} \cdot \mathbf{x}_i = \mathbf{A} \cdot \mathbf{x}_i - \mathbf{A} \cdot \mathbf{x}_1 \cdot \mathbf{y}^T \cdot \mathbf{x}_i = \lambda_i \mathbf{x}_i - \lambda_1 \mathbf{x}_1 \mathbf{y}^T \mathbf{x}_i$$

$$\mathbf{B} \cdot \mathbf{x}_i = \lambda_i \underbrace{(\mathbf{x}_i - \mathbf{x}_1 \mathbf{y}^T \mathbf{x}_i)}_{\mathbf{z}_i} = \lambda_i \mathbf{z}_i$$

$$\mathbf{B} \cdot \mathbf{z}_i = \mathbf{B} \cdot (\mathbf{x}_i - \mathbf{x}_1 \mathbf{y}^T \mathbf{x}_i) = \mathbf{B} \cdot \mathbf{x}_i - \underbrace{(\mathbf{B} \cdot \mathbf{x}_1)}_0 \mathbf{y}^T \mathbf{x}_i = \mathbf{B} \cdot \mathbf{x}_i$$

Din ultimele două relații rezultă că:  $\mathbf{B} \cdot \mathbf{z}_i = \lambda_i \mathbf{z}_i$ , ceea ce arată că matricea  $\mathbf{B}$  are valorile proprii  $\lambda_i$  și vectorii proprii  $\mathbf{z}_i$ , cu  $i=2:n$ .

Altă alternativă de deflație consideră:  $\mathbf{B} = \mathbf{A} \cdot (\mathbf{I}_n - \mathbf{x}_1 \mathbf{y}^T)$ , cu  $\mathbf{y}^T \mathbf{x}_1 = 1$

Vom arăta că:

$$\lambda(\mathbf{B}) = \{0, \lambda_2, \dots, \lambda_n\}$$

$$\mathbf{X}(\mathbf{B}) = \{\mathbf{x}_1, \mathbf{z}_2, \dots, \mathbf{z}_n\}$$

în care:

$$\mathbf{z}_i = \mathbf{x}_i - \frac{\lambda_1}{\lambda_i} (\mathbf{y}^T \mathbf{x}_i) \mathbf{x}_1, \quad i = 2 : n$$

În mod asemănător:

$$\mathbf{B} \cdot \mathbf{x}_1 = \mathbf{A} \cdot \mathbf{x}_1 - \underbrace{\mathbf{A} \cdot \mathbf{x}_1 \cdot (\mathbf{y}^T \cdot \mathbf{x}_1)}_1 = 0$$

$$\mathbf{B} \cdot \mathbf{x}_i = \mathbf{A} \cdot \mathbf{x}_i - \mathbf{A} \cdot \mathbf{x}_1 \mathbf{y}^T \mathbf{x}_i = \lambda_i \mathbf{x}_i - \lambda_1 \mathbf{x}_1 \mathbf{y}^T \mathbf{x}_i$$

$$\mathbf{B} \cdot \mathbf{x}_i = \lambda_i \left( \mathbf{x}_i - \frac{\lambda_1}{\lambda_i} (\mathbf{y}^T \mathbf{x}_i) \cdot \mathbf{x}_1 \right) = \lambda_i \mathbf{z}_i$$

$$\mathbf{B} \cdot \mathbf{z}_i = \mathbf{B} \cdot \left( \mathbf{x}_i - \frac{\lambda_1}{\lambda_i} (\mathbf{y}^T \mathbf{x}_i) \cdot \mathbf{x}_1 \right) = \mathbf{B} \cdot \mathbf{x}_i - \frac{\lambda_1}{\lambda_i} (\mathbf{y}^T \mathbf{x}_i) \underbrace{(\mathbf{B} \cdot \mathbf{x}_1)}_0 = \mathbf{B} \cdot \mathbf{x}_i$$

de unde rezultă  $\mathbf{B} \cdot \mathbf{z}_i = \lambda_i \mathbf{z}_i$ .

Metoda de deflație *Wielandt* consideră:

$$\mathbf{y}^T = \frac{1}{\lambda_1 \mathbf{x}_{1i}} \begin{bmatrix} \mathbf{A}_{i1} & \mathbf{A}_{i2} & \dots & \mathbf{A}_{in} \end{bmatrix}$$

Matricea  $\mathbf{B} = \mathbf{A} - \lambda_1 \mathbf{x}_1 \mathbf{y}^T$  are:

$$\lambda(\mathbf{B}) = \{0, \lambda_2, \dots, \lambda_n\}$$

$$\mathbf{X}(\mathbf{B}) = \{\mathbf{x}_1, \mathbf{z}_2, \dots, \mathbf{z}_n\}$$

în care:

$$\mathbf{z}_i = \mathbf{x}_i - \mathbf{x}_1 \cdot \frac{\mathbf{x}_{ii}}{\mathbf{x}_{1i}}, \quad i = 2 : n$$

$$(\mathbf{A} \cdot \mathbf{x}_1)_i = \sum_{j=1}^n \mathbf{A}_{ij} \mathbf{x}_{1j} = (\lambda_1 \mathbf{x}_1)_i = \lambda_1 \mathbf{x}_{1i}$$

$$\mathbf{y}^T \mathbf{x}_1 = \frac{1}{\lambda_1 \mathbf{x}_{1i}} \sum_{j=1}^n \mathbf{A}_{ij} \mathbf{x}_{1j} = 1$$

$$\mathbf{y}^T \mathbf{x}_i = \frac{1}{\lambda_1 \mathbf{x}_{1i}} \underbrace{\sum_{j=1}^n \mathbf{A}_{ij} \mathbf{x}_{1j}}_{(\mathbf{A} \mathbf{x}_1)_i} = \frac{\lambda_i \mathbf{x}_{ii}}{\lambda_1 \mathbf{x}_{1i}}$$

$$\mathbf{B} \cdot \mathbf{x}_1 = (\mathbf{A} - \lambda_1 \mathbf{x}_1 \mathbf{y}^T) \cdot \mathbf{x}_1 = \mathbf{A} \cdot \mathbf{x}_1 - \lambda_1 \mathbf{x}_1 \underbrace{(\mathbf{y}^T \mathbf{x}_1)}_1 = 0$$

$$\mathbf{B} \cdot \mathbf{x}_i = (\mathbf{A} - \lambda_1 \mathbf{x}_1 \mathbf{y}^T) \mathbf{x}_i = \mathbf{A} \cdot \mathbf{x}_i - \lambda_1 \mathbf{x}_1 (\mathbf{y}^T \mathbf{x}_i)$$

$$\mathbf{B} \cdot \mathbf{x}_i = \lambda_i \mathbf{x}_i - \lambda_1 \mathbf{x}_1 \frac{\lambda_i \mathbf{x}_{ii}}{\lambda_1 \mathbf{x}_{1i}} = \lambda_i \left( \mathbf{x}_i - \mathbf{x}_1 \frac{\mathbf{x}_{ii}}{\mathbf{x}_{1i}} \right) = \lambda_i \mathbf{z}_i$$

$$\mathbf{B} \cdot \mathbf{z}_i = \mathbf{B} \cdot \left( \mathbf{x}_i - \mathbf{x}_1 \frac{\mathbf{x}_{ii}}{\mathbf{x}_{1i}} \right) = \mathbf{B} \cdot \mathbf{x}_i - \underbrace{\mathbf{B} \cdot \mathbf{x}_1}_0 \cdot \frac{\mathbf{x}_{ii}}{\mathbf{x}_{1i}} = \mathbf{B} \cdot \mathbf{x}_i$$

$$\mathbf{B} \cdot \mathbf{z}_i = \lambda_i \cdot \mathbf{z}_i$$

Se știe că  $\mathbf{A}$  și  $\mathbf{A}^T$  au același spectru, iar vectorii proprii corespunzători a două valori proprii diferite sunt ortogonali.

Fie matricea  $\mathbf{B} = \mathbf{A} - \lambda_1 \frac{\mathbf{x}_1 \mathbf{y}_1^T}{\mathbf{y}_1^T \mathbf{x}_1}$  în care  $\mathbf{x}_1$  și  $\mathbf{y}_1$  sunt vectorii proprii dominanți din  $\mathbf{A}$  și  $\mathbf{A}^T$

$$\lambda(\mathbf{B}) = \{0, \lambda_2, \dots, \lambda_n\}$$

$$\mathbf{X}(\mathbf{B}) = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$$

$$\mathbf{B}\mathbf{x}_1 = \mathbf{A}\mathbf{x}_1 - \lambda_1 \mathbf{x}_1 \frac{\mathbf{y}_1^T \mathbf{x}_1}{\mathbf{y}_1^T \mathbf{x}_1} = 0$$

$$\mathbf{B}\mathbf{x}_i = \mathbf{A}\mathbf{x}_i - \lambda_1 \frac{\mathbf{x}_1 \left( \overbrace{\mathbf{y}_1^T \mathbf{x}_i}^0 \right)}{\mathbf{y}_1^T \mathbf{x}_1} = \mathbf{A}\mathbf{x}_i \text{ c.c.t.d.}$$

### Reducerea la forma Hessenberg

Am văzut că aducerea unei matrice  $\mathbf{A} \in \mathbb{R}^{n \times n}$  date la forma Schur reală prin transformări (ortogonale) de asemănare este, în absența unor informații apriorice privind valorile și vectorii proprii, un proces, în mod necesar, infinit. Se ridică întrebarea naturală privitoare la structura cea mai apropiată de forma Schur reală, care poate fi obținută printr-un proces de calcul finit.

**Teorema 1.** Oricare ar fi  $\mathbf{A} \in \mathbb{R}^{n \times n}$  există o matrice ortogonală  $\mathbf{U} \in \mathbb{R}^{n \times n}$ , calculabilă printr-o procedură directă (i.e. printr-un număr finit de operații aritmetice elementare, inclusiv extrageri de radical) astfel încât

$$\mathbf{H} = \mathbf{U}^T \mathbf{A} \mathbf{U}$$

este superior Hessenberg (adică  $\mathbf{H}(i, j) = 0$  pentru toți  $i > j + 1$ ).

*Demonstrație.* Vom da o demonstrație constructivă care va indica modul de obținere efectivă a matricei  $\mathbf{U}$  ca un produs de reflectori elementari (transformări Householder)

$$\mathbf{U} = \mathbf{U}_2 \mathbf{U}_3 \cdots \mathbf{U}_{n-1}$$

conform schemei de calcul

1. **pentru**  $k=1:n-2$

1. Se determină reflectorul  $\mathbf{U}_{k+1}$  astfel încât  $(\mathbf{U}_{k+1} \mathbf{A})(k+2:n, k) = 0$

2.  $\mathbf{A} \leftarrow \mathbf{U}_{k+1} \mathbf{A}$

3.  $\mathbf{A} \leftarrow \mathbf{A} \mathbf{U}_{k+1}$ .

care suprascrie matricea  $\mathbf{A}$  cu matricea  $\mathbf{H}$ . În primul rând să observăm că reflectorul elementar  $\mathbf{U}_{k+1}$  care anulează elementele  $k + 2 : n$  din coloana  $k$  a matricei  $\mathbf{A}$  există și are structura

$$\mathbf{U}_{k+1} = \text{diag}(\mathbf{I}_k, \bar{\mathbf{U}}_{k+1}),$$

unde  $\bar{\mathbf{U}}_{k+1}$  este un reflector de ordinul  $n - k$  și indice 1. În consecință, la primul pas ( $k = 1$ ) premultiplicarea matricei  $\mathbf{A}$  cu  $\mathbf{U}_2$  anulează elementele  $\mathbf{A}(3 : n, 1)$ , iar postmultiplicarea cu  $\mathbf{U}_2 = \mathbf{U}_2^T$ , datorită structurii nu afectează prima coloană a matricei  $\mathbf{U}_2 \mathbf{A}$ , respectiv conservă zerourile create. Prin urmare, am adus matricea  $\mathbf{A}$  la forma superior Hessenberg în prima coloană.

Presupunem acum că după  $k-1$  pași matricea

$$\mathbf{A} \leftarrow \mathbf{U}_k \dots \mathbf{U}_2 \mathbf{A} \mathbf{U}_2 \dots \mathbf{U}_k$$

este superior Hessenberg în primele  $k-1$  coloane. Acum, premultiplicarea cu reflectorul  $\mathbf{U}_{k+1}$ , adecvat calculat, nu modifică primele  $k$  linii și (în plus, datorită structurii din acest moment a lui  $\mathbf{A}$ ) primele  $k-1$  coloane (conservând astfel zerourile obținute în pașii precedenți) și anulează elementele  $\mathbf{A}(k+2 : n, k)$ . Postmultiplicarea cu  $\mathbf{U}_{k+1}$  nu modifică primele  $k$  coloane conservând zerourile create la pașii precedenți și cele create prin premultiplicarea de la pasul curent.

Deci procesul de reducere la forma Hessenberg prin transformări Householder de asemănare poate fi inițializat și apoi continuat, într-un număr finit de  $n-2$  pași, fiecare pas având un număr finit de operații, până la capăt, obținându-se în final matricea

$$\mathbf{A} \leftarrow \mathbf{H} = \mathbf{U}_{n-1} \dots \mathbf{U}_3 \mathbf{U}_2 \mathbf{A} \mathbf{U}_2 \mathbf{U}_3 \dots \mathbf{U}_{n-1}$$

în formă superior Hessenberg. Teorema este demonstrată.

Elementele defînitorii  $\mathbf{v}_k \in \mathbb{R}^n$ ,  $\beta_k = \frac{1}{2} \|\mathbf{v}_k\|^2$  ale reflectorilor elementari

$$\mathbf{U}_k = \mathbf{I}_n - \frac{\mathbf{v}_k \mathbf{v}_k^T}{\beta_k}, \quad k = 2 : n-1$$

pot fi memorate în matricea  $\mathbf{V} \in \mathbb{R}^{n \times (n-2)}$  ( $\mathbf{v}_k$  este coloana  $k-1$  a matricei  $\mathbf{V}$ ) și vectorul  $\mathbf{b} \in \mathbb{R}^{n-2}$  ( $\beta_k = b_{k-1}$ ,  $k = 2:n-2$ ).

Algoritmul de implementare a schemei de calcul de reducere la forma superior Hessenberg, la care adăugăm acumulația opțională a transformărilor (i.e. calculul matricei  $\mathbf{U}$ ) va utiliza intensiv rezultatele relative la transformările Householder prezentate în capitolul 2.

Concret, se presupune cunoscută procedura de calcul fiabil și economic al elementelor defînitorii ale unui reflector ce anulează anumite componente ale unui vector dat, procedură care va fi utilizată explicit în scrierea algoritmilor din acest capitol. Amintim în acest sens că fiind dat un vector nenul  $\mathbf{x} \in \mathbb{R}^n$ , elementele defînitorii  $\mathbf{v}_k \in \mathbb{R}^n$ ,  $\beta_k$  ale reflectorului (91) care asigură anularea componentelor  $k+1:n$  ale lui  $\mathbf{x}$ , i.e.  $(\mathbf{U}_k \mathbf{x})(k+1:n) = \mathbf{0}$ , se calculează economic cu relațiile

1.  $\mathbf{v}_k(1 : k-1) = \mathbf{0}$
2.  $\sigma = \text{sign}(\mathbf{x}(k)) \cdot \|\mathbf{x}(k:n)\|_2$
3.  $\mathbf{v}_k(k) = \mathbf{x}(k) + \sigma$
4.  $\beta_k = \mathbf{x}(k) \cdot \sigma$
5.  $\mathbf{v}_k(k+1:n) = \mathbf{x}(k+1:n)$ .

În ceea ce privește procedurile de premultiplicare și de postmultiplicare ale unei matrice cu un reflector, pentru comoditatea cititorului, preferăm să le prezentăm detaliat aici. Astfel este suficient să dispunem de procedurile de premultiplicare (eventual cu suprascriere)  $\mathbf{A} \leftarrow \mathbf{B} = \mathbf{U} \mathbf{A}$  a unei matrice  $\mathbf{A} \in \mathbb{R}^{n \times p}$  și de postmultiplicare (eventual cu suprascriere)  $\mathbf{A} \leftarrow \mathbf{B} = \mathbf{A} \mathbf{U}$  a unei matrice  $\mathbf{A} \in \mathbb{R}^{m \times n}$  cu un reflector  $\mathbf{U} = \mathbf{I}_n - \frac{\mathbf{u} \mathbf{u}^T}{\beta}$  de ordin  $n$  și indice 1. Precizăm că în variantele cu suprascriere aceasta se efectuează intern, element cu element, fără a se crea matricea intermediară  $\mathbf{B}$ .

Premultiplicarea economică  $\mathbf{A} \leftarrow \mathbf{B} = \mathbf{U}\mathbf{A}$  folosește partiția pe coloane a matricelor  $\mathbf{A}, \mathbf{B}$ . Notând  $\mathbf{a}_j = \mathbf{A}(:, j)$ ,  $\mathbf{b}_j = \mathbf{B}(:, j)$ ,  $j = 1 : p$  schema de calcul este

```
1. pentru j = 1 : p
    1.  $\mathbf{a}_j \leftarrow \mathbf{b}_j = \mathbf{U}\mathbf{a}_j$ 
```

în care produsul  $\mathbf{a}_j \leftarrow \mathbf{b}_j = \mathbf{U}\mathbf{a}_j = (\mathbf{I}_n - \frac{\mathbf{u}\mathbf{u}^T}{\beta})\mathbf{a}_j = \mathbf{a}_j - \tau\mathbf{u}$  se efectuează economic

calculând mai întâi scalarul  $\tau = \frac{\mathbf{u}^T \mathbf{a}_j}{\beta} = \frac{\sum_{i=1}^n u_i a_{ij}}{\beta}$ . Rezultă următorul algoritm.

```
[B]=refmat(u, beta, A)
% Premultiplicarea unei matrice cu un reflector elementar de indice 1.
% Intrări: u ∈ R^n, beta ∈ R = elementele defnitorii ale reflectorului;
%          A ∈ R^{n×p} = matricea dată.
% Ieșiri:  B = UA.
[n,p]=size(A);
for j=1:p
    tau = u(1:n)*A(1:n,j)/beta;
    B(1:n,j)=A(1:n,j) - tau*u(1:n);
end
```

Postmultiplicarea economică  $\mathbf{A} \leftarrow \mathbf{B} = \mathbf{A}\mathbf{U}$  folosește, absolut analog, partiția pe linii a matricelor  $\mathbf{A}, \mathbf{B}$ . Notând  $\mathbf{a}_i^T = \mathbf{A}(i, :)$ ,  $\mathbf{b}_i^T = \mathbf{B}(i, :)$ ,  $i = 1 : m$  schema de calcul este

```
1. pentru i = 1 : m
    1.  $\mathbf{a}_i^T \leftarrow \mathbf{b}_i^T = \mathbf{a}_i^T \mathbf{U}$ 
```

în care produsul  $\mathbf{a}_i^T \leftarrow \mathbf{b}_i^T = \mathbf{a}_i^T \mathbf{U} = \mathbf{a}_i^T (\mathbf{I}_n - \frac{\mathbf{u}\mathbf{u}^T}{\beta}) = \mathbf{a}_i^T - \tau\mathbf{u}^T$  se efectuează

economic calculând mai întâi scalarul  $\tau = \frac{\mathbf{a}_i^T \mathbf{u}}{\beta} = \frac{\sum_{j=1}^n a_{ij} u_j}{\beta}$ . Rezultă următorul algoritm.

```
[B]=matref(A,u, beta)
% Postmultiplicarea unei matrice cu un reflector elementar de indice 1.
% Intrări: A ∈ R^{m×n} = matricea dată;
%          u ∈ R^n, beta ∈ R = elementele defnitorii ale reflectorului.
% Ieșiri:  B = AU
[m,n]=size(A);
for i=1:m
```

```

     $\tau = A(i, 1:n) * u(1:n) / \beta;$ 
     $B(i, 1:n) = A(i, 1:n) - \tau * u(1:n);$ 
end

```

Utilizând relațiile de calcul ale unui reflector elementar ce anulează componentelele dorite ale unui vector dat precum și funcțiile **refmat** și **matref** algoritmul de reducere a unei matrice la forma superior Hessenberg prin transformări ortogonale de asemănare poate fi scris într-o formă mai concisă cum este cea prezentată în continuare.

```

% Reducerea la forma Hessenberg
[A,U,V,b]=HQ(A)
% Intrări:
%      A = matricea dată;
% Ieșiri:  A ← H = UTAU = matricea superior Hessenberg ortogonal asemenea cu A;
%      U ∈ Rn×n = matrice ortogonală de transformare
%      V ∈ Rn×(n-2) = matrice cu coloane vectorii definatori ai transformărilor;
%      b ∈ Rn-2 = vector conținând scalarii definatori ai transformărilor.
[n,n]=size(A);
U=eye(n,n);
for k=1:n-2
    % Determinarea lui Uk+1, i.e. vk+1 = V(:, k), ρk+1 = b(k)
    V(1 : k, k) = 0
    σ = sign(A(k + 1, k)) * ||A(k + 1 : n, k)||
    V(k + 1, k) = A(k + 1, k) + σ
    b(k) = V(k + 1, k) * σ
    V(k + 2 : n, k) = A(k + 2 : n, k);
    % Calculul A ← Uk+1A
    A(k + 1, k) = -σ;
    A(k + 2 : n, k) = 0;
    [A(k + 1 : n, k + 1 : n)] =
        refmat(V(k + 1 : n, k), b(k), A(k + 1 : n, k + 1 : n));
    % Calculul A ← AUk+1
    [A(:, k+1:n)] = matref(A(:, k+1:n), V(k+1:n, k), b(k));
    % Acumularea transformărilor (i.e. calculul A ← AUk+1)
    [U(:, k + 1 : n)] =
        matref(U(:, k + 1 : n), V(k + 1 : n, k), b(k));

```

Complexitatea algoritmului este  $O(n^3)$  fiind necesare  $N_{op} \cong \frac{10}{3} n^3$  operații elementare în format virgulă mobilă. Dacă se calculează și matricea de transformare  $U$ , atunci trebuie executate încă  $N'_{op} \cong \frac{4}{3} n^3$  operații.

Algoritmul 9.3 este un algoritm numeric stabil, existând demonstrații pentru faptul că matricea Hessenberg calculată într-o aritmetică de virgulă mobilă este un rezultat exact pentru o matrice inițială foarte puțin perturbată.

*Observație:* Este posibilă reducerea matricei  $\mathbf{A}$  la forma superior Hessenberg prin *transformări de asemănare neortogonale* cu un efort de calcul redus aproximativ la jumătate din cel cerut de algoritmul 9.3. O astfel de posibilitate o oferă construcția transformării sub forma unui produs de transformări gaussiene elementare stabilizate (deci cu permutări de linii)  $(\mathbf{P}_k \mathbf{M}_k)$ ,  $k = 2 : n - 1$  cu  $\mathbf{M}_k = \mathbf{I}_n - \mathbf{m}_k \mathbf{e}_k^T$  și  $\mathbf{P}_k$  o matrice de permutare a liniei  $k$  cu o linie  $i_k \geq k$  (vezi problema rezolvată R9.17). Motivele pentru care, totuși, sunt preferate transformările ortogonale se referă la faptul că acestea din urmă conservă condiționarea numerică a problemei de calcul a valorilor și vectorilor proprii și asigură o stabilitate numerică mult mai bună.

Matricea superior Hessenberg ortogonal asemenea cu matricea dată obținută cu algoritmul de mai sus nu este unica matrice Hessenberg care satisface (86). Într-adevăr dacă  $\mathbf{A}$  este matricea dată și  $\mathbf{Q}_0 \in \mathbb{R}^{n \times n}$  este o matrice ortogonală arbitrară atunci aplicând algoritmul matricei  $\mathbf{Q}_0^T \mathbf{A} \mathbf{Q}_0$  obținem

$$\mathbf{H}' = \mathbf{U}^T \mathbf{Q}_0^T \mathbf{A} \mathbf{Q}_0 \mathbf{U} = \mathbf{Q}^T \mathbf{A} \mathbf{Q}$$

cu  $\mathbf{Q} = \mathbf{Q}_0 \mathbf{U}$  și  $\mathbf{H}'$  superior Hessenberg, în general diferită de matricea  $\mathbf{H}$ . Se observă totuși că

$$\mathbf{U} \mathbf{e}_1 = \mathbf{U}_2 \mathbf{U}_3 \dots \mathbf{U}_{n-1} \mathbf{e}_1 = \mathbf{e}_1$$

și deci

$$\mathbf{Q} \mathbf{e}_1 = \mathbf{Q}_0 \mathbf{U} \mathbf{e}_1 = \mathbf{Q}_0 \mathbf{e}_1.$$

De aici se poate extrage ideea că determinanta pentru identitatea transformării este prima sa coloană. Această observație are o importanță capitală în dezvoltarea procedurală a algoritmului QR cu deplasare implicită prezentat într-o secțiune ulterioară și care reprezintă un rezultat de importanță recunoscută în calculul numeric matriceal. Contextul în care acționează algoritmul QR cu deplasare implicită este cel al matricelor superior Hessenberg *irreductibile* i.e. al matricelor superior Hessenberg cu toate elementele de pe prima subdiagonală nenule. Acum transpunem observația de mai sus într-un cadru formal dat de următoarea teoremă.

**Teorema 2.** Fie  $\mathbf{A} \in \mathbb{R}^{n \times n}$  și matricele ortogonale  $\mathbf{U}, \mathbf{V} \in \mathbb{R}^{n \times n}$  astfel încât matricele

$$\mathbf{H} = \mathbf{U}^T \mathbf{A} \mathbf{U}, \quad \mathbf{G} = \mathbf{V}^T \mathbf{A} \mathbf{V}$$

sunt ambele superior Hessenberg irreductibile, i.e.

$$\mathbf{h}_{j+1,j} \neq 0, \quad \mathbf{g}_{j+1,j} \neq 0, \quad j = 1 : n - 1.$$

Dacă matricele  $\mathbf{U}$  și  $\mathbf{V}$  au aceeași primă coloană

$$\mathbf{U}(:, 1) = \mathbf{V}(:, 1)$$

atunci între celelalte coloane există relațiile

$$\mathbf{U}(:, j) = \pm \mathbf{V}(:, j), \quad j = 2 : n.$$

Mai mult, avem cu necesitate

$$\mathbf{h}_{ij} = \pm \mathbf{g}_{ij}, \quad \forall i, j.$$

*Demonstrație.* Matricea ortogonală  $\mathbf{Q} = \mathbf{V}^T \mathbf{U}$  are ca primă coloană vectorul  $\mathbf{q}_1 = \mathbf{Q}(:, 1) = \mathbf{V}^T \mathbf{U}(:, 1) = \mathbf{V}^T \mathbf{V}(:, 1) = \mathbf{e}_1$ . Celelalte coloane ale matricei  $\mathbf{Q}$  pot fi exprimate recurent în raport de precedentele în modul următor. Rezultă  $\mathbf{A} = \mathbf{U} \mathbf{H} \mathbf{U}^T = \mathbf{V} \mathbf{G} \mathbf{V}^T$ , i.e.  $\mathbf{Q} \mathbf{H} = \mathbf{G} \mathbf{Q}$ , relație care, scrisă pe coloane, ne conduce la

$$GQ(:, j) = QH(:, j) = \sum_{k=1}^j Q(:, k)h_{k,j} + Q(:, j+1)h_{j+1,j} .$$

Acum, notând  $q_j = Q(:, j)$ , avem

$$q_{j+1} = \frac{Gq_j - \sum_{k=1}^j q_k h_{kj}}{h_{j+1,j}} .$$

Este ușor de observat că această recurență induce, în mod necesar, matricei  $Q$  o structură superior triunghiulară. Într-adevăr,  $G$  fiind superior Hessenberg iar  $q_1$  având numai primul element nenul rezultă că  $q_2$  are numai primele două elemnte nenule etc. Inversa unei matrice superior triunghiulare este superior triunghiulară. Dar  $Q$  fiind ortogonală rezultă că este simultan superior și inferior triunghiulară, i.e.  $Q$  este o matrice diagonală. Cum elementele diagonale ale unei matrice ortogonale diagonale nu pot fi decât egale cu  $\pm 1$ , rezultă

$$U(:, j) = VQ(:, j) = V \cdot (\pm e_j) = \pm V(:, j) .$$

De asemenea din  $QH = GQ$  avem  $H = QGQ$ , de unde rezultă că teorema este demonstrată.

Miezul teoremei constă în faptul că transformările ortogonale  $U$  și  $V$  de reducere a aceleiași matrice la o formă superior Hessenberg ireductibilă sunt esențial determinate (mai puțin un semn la fiecare coloană) de prima lor coloană. Dacă prima coloană din cele două matrice de transformare este identică, atunci și rezultatul reducerii este esențial același (mai puțin semnul unor elemente). Aici termenii “*esențial aceleași*” semnifică faptul că cele două matrice superior Hessenberg sunt “la fel de departe” de o formă Schur a matricei inițiale, al carei calcul este obiectivul principal al problemei valorilor și vectorilor proprii.