Projektionsmethoden zur Lösung der symmetrischen Eigenwertprobleme

Udo Schendel   Alke El-Fares−Ziegler

Forschungsprojekt *Parallele Numerik*
Fachbereich Mathematk und Informatik
Freie Universität Berlin 2001

Projektionsmethoden zur Lösung der symmetrischen Eigenwertprobleme

Herrn Prof. Dr. Manfred Feilmeier
zum 60. Geburtstag gewidmet

Udo Schendel und Alke El-Fares–Ziegler

Freie Universität Berlin
Fachbereich Mathematik und Informatik
Institut für Mathematik I
Arnimallee 2-6, D-14195 Berlin

Zusammenfassung

Die folgende Arbeit beschäftigt sich mit einer iterativen Projektionsmethode zur Lösung des symmetrischen Eigenwertproblems $Ax = \lambda x,\ A = A^T \in \mathbb{R}^{n\times n},\ x \in \mathbb{R}\backslash\{0\}$, $\lambda \in \mathbb{R}$. Das Verfahren berechnet sowohl alle Eigenwerte $\lambda_i,\ i = 1,...,n$, als auch alle Eigenvektoren $x_i,\ i = 1,...,n$. Diese Methode wird dann im zweiten Kapitel übertragen auf das verallgemeinerte symmetrische Eigenwertproblem $Ax = \lambda Bx$, $A = A^T \in \mathbb{R}^{n\times n},\ B = B^T \in \mathbb{R}^{n\times n},\ B$ positiv definit, $x \in \mathbb{R}\backslash\{0\},\ \lambda \in \mathbb{R}$.
Im dritten Abschnitt wird gezeigt, daß beide Verfahren sowohl für die Berechnung der Eigenwerte als auch der Eigenvektoren kubisch konvergieren.

Inhaltsverzeichnis

1 Symmetrisches Eigenwertproblem

Im folgenden Abschnitt wird die Projektionsmethode für das symmetrische Eigenwertproblem ausgearbeitet und der Algorithmus explizit angegeben.
Zunächst wird die Grundidee dargestellt, aus der sich dann die Projektionsmethode herleiten läßt:

1.1 Grundidee

Betrachtet wird das symmetrische Eigenwertproblem

$$Ax = \lambda x, \quad A = A^T \in \mathbb{R}^{n\times n},\ x \in \mathbb{R}\backslash\{0\},\ \lambda \in \mathbb{R}_{\geq 0} \tag{1.1}$$

Es seien die Eigenwerte $\lambda_1, ..., \lambda_{r-1}$ und die zugehörigen Eigenvektoren $x_1, ..., x_{r-1}$ bereits berechnet. Die Anfangswerte bzw. die erste Approximation für den neu zu bestimmen r-ten Eigenwert λ_r und r-ten Eigenvektor x_r werden definiert durch:

(i) $\lambda_1^0 := 0$ und $\lambda_r^0 := \lambda_{r-1}$ für $r = 2, ..., n$,

(ii) $x_r^0 := e_r = [0, ..., 0, 1, 0, ..., 0]^T$ für $r = 1, 2, \cdots, n$, ist der r-te Einheitsvektor.

Bestimmung der Eigenvektoren

Die Eigenvektoren einer symmetrischen Matrix $A \in \mathbb{R}^{n\times n}$ bilden eine orthonormierte Basis des $\mathbb{R}^n$. Für den Ansatz der Approximation x_r^{k+1} muß also die Orthogonalität zu den bereits berechneten Eigenvektoren x_i gelten:

$$x_i^T x_r^{k+1} = 0, \qquad i = 1, 2, ..., r-1, \tag{1.2}$$

$$(x_r^k)^T x_r^{k+1} = 1. \tag{1.3}$$

Damit ist gewährleistet, daß jede Näherung des Eigenwerts λ_r^k bzw. des Eigenvektors x_r^k die Eigenwertgleichung erfüllt. Für die Approximationen x_r^{k+1} gilt nun:

$$(A - \lambda_r^k I)x_r^{k+1} + \sum_{i=1}^{r-1} \alpha_i^{k+1} x_i + \alpha_r^{k+1} x_r^k = 0, \tag{1.4}$$

mit den Unbekannten $\alpha_i^{k+1} \in \mathbb{R},\ i = 1, \cdots, r$.

Die Gleichungen (1.2), (1.3) und (1.4) sind äquivalent zum folgenden linearen Gleichungssystem der Ordnung $(n + r)$:

$$\underbrace{\begin{bmatrix} A-\lambda_r^k I & x_1 & x_2 & \cdots & x_{r-1} & x_r^k \\ x_1^T & & & & & \\ x_2^T & & & & & \\ \vdots & & & 0 & & \\ x_{r-1}^T & & & & & \\ (x_r^k)^T & & & & & \end{bmatrix}}_{\in \mathbb{R}^{(n\cdot(r+1))\times(n+r)}} \underbrace{\begin{bmatrix} x_r^{k+1} \\ \alpha_1^{k+1} \\ \alpha_2^{k+1} \\ \vdots \\ \alpha_{r-1}^{k+1} \\ \alpha_r^{k+1} \end{bmatrix}}_{\in \mathbb{R}^{(n+r)}} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \in \mathbb{R}^{n\cdot(r+1)} \tag{1.5}$$

mit $A-\lambda_r^k I \in \mathbb{R}^{n\times n},\ x_i, x_r^k \in \mathbb{R}^n$ und $\alpha_i^{k+1} \in \mathbb{R},\ i = 1,...,r.$

Bestimmung der Eigenwerte

Definition 1.6 (Rayleigh-Quotient)
Für eine Matrix $A \in \mathbb{R}^{n\times n}$ und einen Vektor $x \in \mathbb{R}^n \backslash \{0\}$ ist der Rayleigh-Quotient definiert als

$$R_A[x] := \frac{\langle Ax, x\rangle}{\langle x, x\rangle}.$$

Satz 1.7
Ist $x \in \mathbb{R}^n \backslash \{0\}$ Eigenvektor zum Eigenwert λ von $A \in \mathbb{R}^{n\times n}$, so gilt $\lambda = R_A[x]$.

Eine Approximation λ_r^{k+1} des Eigenwertes λ_r kann also durch Einsetzen der Approximation x_r^{k+1}, Ergebnis des linearen Gleichungssystems (1.5), in den Rayleigh-Quotienten berechnet werden:

$$\lambda_r^{k+1} = \frac{(x_r^{k+1})^T\ Ax_r^{k+1}}{(x_r^{k+1})^T\ x_r^{k+1}} = R_A[x_r^{k+1}]. \tag{1.8}$$

Bemerkung 1.9
Die Ordnung des Systems (1.5) wächst mit jedem berechneten Eigenvektor x_r. (Am Ende hat die Systemmatrix $4n^2$ Elemente mit n^2 Nullen!) ∘

In einer neuen Version des Algorithmus hat das System (1.5) nur noch die Ordnung n für alle $r = 1,...,n$. Die Effizienz wird dadurch sehr verbessert:

1.2 Die Projektionsmethode

Im Folgenden seien

$$X_r = [x_1, ..., x_{r-1}] \in \mathbb{R}^{n\times(r-1)}$$

die Matrix der bereits bestimmten Eigenvektoren und

$$\nu_r^{k+1} = [\alpha_1^{k+1}, ..., \alpha_{r-1}^{k+1}]^T \in \mathbb{R}^{r-1}$$

die Vektoren der zugeordeten Parameter, die jeweils ermittelt werden müssen.
Aus der Gleichung (1.4) folgt dann

$$(A - \lambda_r^k I)x_r^{k+1} + X_r\nu_r^{k+1} + \alpha_r^{k+1}x_r^k = 0. \tag{1.10}$$

Es ergibt sich aus (1.10) durch Multiplikation von links mit $(x_r^{k+1})^T$

$$\alpha_r^{k+1} = -(x_r^{k+1})^T(A - \lambda_r^k I)x_r^{k+1} \tag{1.11}$$

und entsprechend durch Multiplikation mit X_r^T

$$X_r^T(A - \lambda_r^k I)x_r^{k+1} + \nu_r^{k+1} + \alpha_r^{k+1}X_r^T x_r^k = 0. \tag{1.12}$$

Aus der Gleichung

$$Ax_i = \lambda_i x_i,\ \ i = 1, 2, ..., r-1 \iff AX_r = X_r\mathrm{diag}(\lambda_i) \tag{1.13}$$

ergibt sich durch Subtraktion mit $\lambda_r^k x_r^k$

$$\begin{aligned} (A - \lambda_r^k I)x_i &= (\lambda_i - \lambda_r^k)x_i \\ (A - \lambda_r^k I)X_r &= X_r\mathrm{diag}(\lambda_i - \lambda_r^k) \end{aligned} \iff \qquad i = 1, 2, ..., r-1. \tag{1.14}$$

Mit A ist auch $A - \lambda_r^k I$ symmetrisch, also gilt für (1.14)

$$X_r^T(A - \lambda_r^k I) = \mathrm{diag}(\lambda_i - \lambda_r^k)X_r^T \tag{1.15}$$

und damit folgt aus (1.12) wegen $X_r^T x_r^{k+1} = 0$

$$\nu_r^{k+1} = -\alpha_r^{k+1}X_r^T x_r^k. \tag{1.16}$$

Damit folgt aus (1.10)

$$(A - \lambda_r^k I)x_r^{k+1} - \alpha_r^{k+1}X_rX_r^T x_r^k + \alpha_r^{k+1}x_r^k = 0 \iff$$

$$(A - \lambda_r^k I)x_r^{k+1} + \alpha_r^{k+1}(I - X_rX_r^T)x_r^k = 0 \iff \tag{1.17}$$

$$(A - \lambda_r^k I)x_r^{k+1} = -\alpha_r^{k+1}P_r x_r^k \qquad \text{mit} \quad P_r := I - X_rX_r^T. \tag{1.18}$$

Die Matrix P_r ist eine Projektion, und es gilt hierzu der Satz

Satz 1.19
Die Matrix $P_r := I - X_r X_r^T,\ r = 1, 2, ..., n,$ *ist eine orthogonale Projektion in den Unterraum* $\mathcal{X} := span\{x_r, x_{r+1}, ..., x_n\} \subseteq \mathbb{R}^n$ *der noch nicht berechneten Eigenvektoren* $x_i \in \mathbb{R}^n \backslash \{0\},\ i = r, ..., n.$

Beweis: P_r ist eine orthogonale Projektion, wenn gilt:

1. $P_r^T = P_r$, dies folgt aus der Symmetrie von $X_r X_r^T$.
2. $P_r^2 = (I - X_r X_r^T)(I - X_r X_r^T) = (I - 2X_r X_r^T + X_r I X_r^T) = P_r$.
3. $P_r z \in \mathcal{X}$ für alle $z \in \mathbb{R}^n$:
   Für alle $z \in \mathbb{R}^n$ gilt $z = \sum_{i=1}^{n} c_i x_i$ mit den Konstanten $c_i \in \mathbb{R}$ und den n orthonormierten Eigenvektoren $x_i \in \mathbb{R}^n \backslash \{0\}$. Mit $\langle x_i, x_j \rangle = \delta_{ij},\ i, j = 1, ..., n$, ist also $P_r z = (I - X_r X_r^T) z = z - X_r (\sum_{i=1}^{r-1} c_i \tilde{e}_i) = z - \sum_{i=1}^{r-1} c_i x_i = \sum_{i=r}^{n} c_i x_i \in \mathcal{X}$, $\tilde{e}_i \in \mathbb{R}^{r-1}$ ist der i-te Einheitsvektor. □

Die folgenden Aussagen werden zur algorithmischen Berechnung benötigt:

α_r^{k+1} berechnet sich durch Multipikation von links mit $(x_r^k)^T$ aus (1.18):

$$(x_r^k)^T x_r^{k+1} = 1 = -\alpha_r^{k+1} (x_r^k)^T (A - \lambda_r^k I)^{-1} P_r x_r^k \quad \Longrightarrow$$

$$(\alpha_r^{k+1})^{-1} = -(x_r^k)^T (A - \lambda_r^k I)^{-1} P_r x_r^k. \tag{1.20}$$

Mit (1.18) und (1.20) berechnet sich x_r^{k+1} wie folgt:

$$x_r^{k+1} = \frac{(A - \lambda_r^k I)^{-1} P_r x_r^k}{(x_r^k)^T (A - \lambda_r^k I)^{-1} P_r x_r^k}. \tag{1.21}$$

Der Rayleigh-Quotient $R_A[x_r^{k+1}]$ liefert aus (1.11) den Eigenwert

$$\lambda_r^{k+1} = \lambda_r^k - \frac{\alpha_r^{k+1}}{(x_r^{k+1})^T x_r^{k+1}} = \lambda_r^k - \frac{\alpha_r^{k+1}}{\|x_r^{k+1}\|_2^2}. \tag{1.22}$$

Nach der Berechnung des r-ten Eigenwerts wird x_r^{k+1} normiert: $\frac{x_r^{k+1}}{\|x_r^{k+1}\|_2}$.

Die Projektion P_r läßt sich rekursiv berechnen:

$$P_r = I - X_r X_r^T = I - \sum_{i=1}^{r-1} x_i x_i^T \quad \Longrightarrow \quad P_{r+1} = P_r - x_r x_r^T. \tag{1.23}$$

Mit diesen Beziehungen wurde der folgende Algorithmus entwickelt:

Algorithmus 1.24 (Eigenwertprblem $Ax = \lambda x$)

INPUT: $A \in I\!R^{n\times n}$; *symmetrisch*

$0 < eps < 1$; *vorgegebene Genauigkeitsschranke*

OUTPUT: $\lambda_i,\ i = 1, ..., n$; *alle Eigenwerte der symmetrischen Matrix* $A \in I\!R^{n\times n}$

$x_i,\ i = 1, ..., n$; *alle Eigenvektoren der symmetrischen Matrix* $A \in I\!R^{n\times n}$

FOR $r = 1, ..., n$ DO BEGIN

IF $r = 1$ THEN $\lambda_1^0 := 0$; $P_1 := I \in I\!R^{n\times n}$;

ELSE $\lambda_r^0 := \lambda_{r-1}$; $X_r := [x_1, ..., x_{r-1}] \in I\!R^{n\times(r-1)}$;

END IF

$x_r^0 := e_r = [0, ..., 0, 1, 0, ..., 0] \in I\!R^n$; ($r$-te Einheitsvektor)

$\delta_r^0 := 1$; $k = 0$;

WHILE $\delta_r^k > eps$ DO BEGIN

1. Berechnung von $\vartheta_r^k := (A - \lambda_r^k I)^{-1} P_r x_r^k$ als Lösung des linearen Gleichungssystems
   $(A - \lambda_r^k I)\vartheta_r^k = P_r x_r^k$;
2. Berechnung von $\alpha_r^{k+1} = \left((x_r^k)^T (A - \lambda_r^k I)^{-1} P_r x_r^k\right)^{-1} = \left((x_r^k)^T \vartheta_r^k\right)^{-1}$;
3. Berechnung der Approximationen des Eigenvektors x_r^{k+1}:
   $\hat{x}_r^{k+1} = \alpha_r^{k+1}(A - \lambda_r^k I)^{-1} P_r x_r^k = \alpha_r^{k+1}\vartheta_r^k$;
4. Berechnung der Approximationen des Eigenwerts $\lambda_r^{k+1} = \lambda_r^k + \frac{\alpha_r^{k+1}}{\|\hat{x}_r^{k+1}\|_2^2}$;
5. Normierung der Approximationen des Eigenvektors $x_r^{k+1} = \frac{\hat{x}_r^{k+1}}{\|\hat{x}_r^{k+1}\|_2}$;
6. Abbruchkriterium: $\delta_r^{k+1} := \frac{|\alpha_r^{k+1}|}{\|\hat{x}_r^{k+1}\|_2^2}$;

$k = k + 1$;

END WHILE

$x_r = x_r^k$; *Eigenvektoren*

$\lambda_r = \lambda_r^k$; *Eigenwerte*

$P_{r+1} = P_r - x_r x_r^T$;

END FOR

2 Allgemeines symmetrisches Eigenwertproblem

Betrachtet wird nun das allgemeine symmetrische Eigenwertproblem $Ax = \lambda Bx$ mit $A = A^T$ und $B = B^T$, B positiv definit.

Ist $B = B^{\frac{1}{2}}B^{\frac{1}{2}}$ hermitesch, so ist auch $B^{\frac{1}{2}}$ hermitesch [BuGe85] und es gilt

$$(A - \lambda B)x = 0 \Longleftrightarrow (A - \lambda B^{\frac{1}{2}}B^{\frac{1}{2}})x = 0 \Longrightarrow (B^{-\frac{1}{2}}AB^{-\frac{1}{2}} - \lambda I)B^{\frac{1}{2}}x = 0 \Longrightarrow$$

$$B^{-\frac{1}{2}}AB^{-\frac{1}{2}}z = \lambda z \qquad \text{mit } z := B^{\frac{1}{2}}x \text{ und } \hat{A} := B^{-\frac{1}{2}}AB^{-\frac{1}{2}}. \tag{2.1}$$

Es kann also der folgende Satz formuliert werden:

Satz 2.2
Das allgemeine Eigenwertproblem $Ax = \lambda Bx$, $A, B \in I\!K^{n\times n}$ hermitesch, B positiv definit, läßt sich auf das Standard-Eigenwertproblem $\hat{A}z = \lambda z$ zurückführen.
$\hat{A} \in I\!K^{n\times n}$ ist hermitesch, die Eigenwerte λ der Matrix $\hat{A}$ bzw. des Matrixpaares (A, B) sind reell und die Eigenvektoren $x \in I\!K^n\backslash\{0\}$ des Matrixpaares (A, B) sind B-orthogonal, d.h. $(x_i, Bx_j) = \delta_{ij}$ für alle $i, j = 1, ..., 2$.

Analog zu (1.2) und (1.3) gelten nun die Orthogonalitätsbedingungen:

$$x_i^T Bx_r^{k+1} = 0, \qquad i = 1, ..., r-1 \tag{2.3}$$

$$(x_r^k)^T Bx_r^{k+1} = 1. \tag{2.4}$$

$x_1, ..., x_{r-1}$ seien die bereits berechneten Eigenvektoren des Matrixpaares (A, B).
Mit (2.1) und Satz 2.2 ist

$$\begin{aligned} \hat{A} &= B^{-\frac{1}{2}}AB^{-\frac{1}{2}}, \\ z_i &= B^{\frac{1}{2}}x_i, \\ Z_r &:= [B^{\frac{1}{2}}x_1, ..., B^{\frac{1}{2}}x_{r-1}] = B^{\frac{1}{2}}[x_1, ..., x_{r-1}] = B^{\frac{1}{2}}X_r \end{aligned}$$

und daraus folgt analog zu (1.18) für die Approximationen z_r^{k+1}:

$$(\hat{A} - \lambda_r^k I)z_r^{k+1} + \alpha_r^{k+1}(I - Z_r Z_r^T)z_r^k = 0 \quad \Longleftrightarrow \tag{2.5}$$

$$(\hat{A} - \lambda_r^k I)z_r^{k+1} = -\alpha_r^{k+1}P_r z_r^k \qquad \text{mit } P_r := I - Z_r Z_r^T. \tag{2.6}$$

Satz 2.7
Die Matrix $P_r := I - Z_r Z_r^T$, $r = 1, 2, ..., n$, ist eine orthogonale Projektion in den Unterraum $\tilde{\mathcal{X}} := span\{B^{\frac{1}{2}}x_r, B^{\frac{1}{2}}x_{r+1}, ..., B^{\frac{1}{2}}x_n\} \subseteq I\!R^n$ der noch nicht berechneten Eigenvektoren $x_i \in I\!R^n\backslash\{0\}$, $i = r, ..., n$.

Beweis:

1. $Z_r Z_r^T$ ist symmetrisch $\Rightarrow \; P_r^T = P_r$.
2. $P_r^2 = (I - 2Z_r Z_r^T + Z_r X_r^T B^{\frac{1}{2}} B^{\frac{1}{2}} X_r Z_r^T) = B^{\frac{1}{2}}(I - 2Z_r Z_r^T + Z_r Z_r^T) = P_r$.
3. Für alle $y \in \mathbb{R}^n$ gilt $y = \sum\limits_{i=1}^{n} c_i B^{\frac{1}{2}} x_i$ mit den Konstanten $c_i \in \mathbb{R}$ und den B-orthonormierten Eigenvektoren $x_i \in \mathbb{R}^n \backslash \{0\}$. Mit $\langle x_i, Bx_j \rangle = \delta_{ij}, i, j = 1, ..., n$, ist also $P_r y = (I - Z_r Z_r^T) y = y - Z_r(\sum\limits_{i=1}^{r-1} c_i \tilde{e}_i) = y - \sum\limits_{i=1}^{r-1} c_i B^{\frac{1}{2}} x_i = \sum\limits_{i=r}^{n} c_i B^{\frac{1}{2}} x_i \in \tilde{\mathcal{X}}$, $\tilde{e}_i \in \mathbb{R}^{r-1}$ ist der i-te Einheitsvektor. $\square$

Für die numerischen Berechnung wird (2.5) umgeformt:

$$\begin{aligned}
(\hat{A} - \lambda_r^k I) z_r^{k+1} + \alpha_r^{k+1} (I - Z_r Z_r^T) z_r^k &= 0 \iff \\
(B^{-\frac{1}{2}} A B^{-\frac{1}{2}} - \lambda_r^k I) B^{\frac{1}{2}} x_r^{k+1} + \alpha_r^{k+1} (I - B^{\frac{1}{2}} X_r X_r^T B^{\frac{1}{2}}) B^{\frac{1}{2}} x_r^k &= 0 \iff \\
(A B^{-\frac{1}{2}} - \lambda_r^k B^{\frac{1}{2}}) B^{\frac{1}{2}} x_r^{k+1} + \alpha_r^{k+1} (B^{\frac{1}{2}} - B X_r X_r^T B^{\frac{1}{2}}) B^{\frac{1}{2}} x_r^k &= 0 \iff \\
(A - \lambda_r^k B) x_r^{k+1} + \alpha_r^{k+1} B (I - X_r X_r^T B) x_r^k &= 0.
\end{aligned} \tag{2.8}$$

Der Eigenwert λ_r^{k+1} berechnet sich wie in (1.22). Analog zu (1.10) gilt der Ansatz

$$(A - \lambda_r^k B) x_r^{k+1} + B X_r \nu_r^{k+1} + \alpha_r^{k+1} B x_r^k = 0$$

und durch Multiplikation mit $(x_r^{k+1})^T$ entsprechend zu (1.11) bekommt man

$$(x_r^{k+1})^T A x_r^{k+1} = \lambda_r^k (x_r^{k+1})^T B x_r^{k+1} - \alpha_r^{k+1},$$

und daraus folgt die Approximation $\lambda_r^{k+1} = \lambda_r^k - \frac{\alpha_r^{k+1}}{(x_r^{k+1})^T B x_r^{k+1}}$.

Die Projektionsmatrix wird mit $\mathcal{P}_1 := B$ wie folgt definiert:

$$\mathcal{P}_r := B(I - X_r X_r^T B) = \mathcal{P}_{r-1} - B x_{r-1} x_{r-1}^T B = \mathcal{P}_{r-1} - (B x_{r-1})(B x_{r-1})^T.$$

Bemerkung 2.9
$\mathcal{P}_r$ ist eine Projektion in den Unterraum $\hat{\mathcal{X}} := \text{span}\{Bx_r, Bx_{r+1}, ..., Bx_n\} \subseteq \mathbb{R}^n$ der noch nicht berechneten B-orthonormierten Eigenvektoren $x_i \in \mathbb{R}^n \backslash \{0\}, \; i = r, ..., n$. Analog zu Satz 2.7 gilt mit $x_r^k = \sum\limits_{i=1}^{n} c_i x_i$:

$$\begin{aligned}
\mathcal{P}_r x_r^k &= B(I - X_r X_r^T B) \sum_{i=1}^{n} c_i x_i = \sum_{i=1}^{n} c_i B x_i - B X_r (\sum_{i=1}^{r-1} c_i \tilde{e}_i) \\
&= \sum_{i=1}^{n} c_i B x_i - \sum_{i=1}^{r-1} c_i B x_i = \sum_{i=r}^{n} c_i B x_i \in \hat{\mathcal{X}}.
\end{aligned}$$

$\circ$

Algorithmus 2.10 (allgemeines Eigenwertproblem $Ax = \lambda Bx$)

INPUT: $A, B \in \mathbb{R}^{n\times n}$; *symmetrisch und B positiv definit*

$0 < eps < 1$; *vorgegebene Genauigkeitsschranke*

OUTPUT: $\lambda_i,\ i = 1, ..., n,$ *alle Eigenwerte von $Ax = \lambda Bx$*

$x_i,\ i = 1, ..., n,$ *alle Eigenvektoren von $Ax = \lambda Bx$*

FOR $r = 1, ..., n$ DO BEGIN

IF $r = 1$ THEN $\lambda_1^0 := 0;$ $\quad \mathcal{P}_1 := B \in \mathbb{R}^{n\times n};$

ELSE $\lambda_r^0 := \lambda_{r-1};$ $\quad X_r := [x_1, ..., x_{r-1}] \in \mathbb{R}^{n\times(r-1)};$

END IF

$x_r^0 := e_r = [0, ..., 0, 1, 0, ..., 0] \in \mathbb{R}^n;$ (r-te Einheitsvektor)

$\delta_r^0 := 1 \qquad k = 0;$

WHILE $\delta_r^k > eps$ DO BEGIN

1. Berechnung von $\vartheta_r^k := (A - \lambda_r^k B)^{-1}\mathcal{P}_r x_r^k$ als Lösung des linearen Gleichungssystems $(A - \lambda_r^k B)\vartheta_r^k = \mathcal{P}_r x_r^k$;
2. Berechnung von $\alpha_r^{k+1} = \left((x_r^k)^T (A - \lambda_r^k B)^{-1}\mathcal{P}_r x_r^k\right)^{-1} = \left((x_r^k)^T \vartheta_r^k\right)^{-1}$;
3. Berechnung der Approximationen des Eigenvektors x_r^{k+1}:
   $\hat{x}_r^{k+1} = \alpha_r^{k+1}(A - \lambda_r^k B)^{-1}\mathcal{P}_r x_r^k = \alpha_r^{k+1}\vartheta_r^k;$
4. Berechnung der Approximationen des Eigenwerts λ_r^{k+1}:
   $\lambda_r^{k+1} = \lambda_r^k + \frac{\alpha_r^{k+1}}{(\hat{x}_r^{k+1})^T B \hat{x}_r^{k+1}};$
5. Normierung der Approximationen des Eigenvektors x_r^{k+1}:
   $x_r^{k+1} = \frac{\hat{x}_r^{k+1}}{\|\hat{x}_r^{k+1}\|_B} = \frac{\hat{x}_r^{k+1}}{\sqrt{(\hat{x}_r^{k+1})^T B \hat{x}_r^{k+1}}};$
6. Abbruchkriterium: $\delta_r^{k+1} := \frac{|\alpha_r^{k+1}|}{\|\hat{x}_r^{k+1}\|_B};$

$k = k + 1;$

END WHILE

$x_r = x_r^k; \qquad \hat{x}_r = \hat{x}_r^k;$ *normierte und nichtnormierte Eigenvektoren*

$\lambda_r = \lambda_r^k;$ *Eigenwerte*

$\mathcal{P}_{r+1} = \mathcal{P}_r - (Bx_r)(Bx_r)^T = \mathcal{P}_r - \frac{1}{\|\hat{x}_r\|_B^2}(B\hat{x}_r)(B\hat{x}_r)^T;$

END FOR

3 Konvergenzbetrachtungen und Eigenschaften

Die Projektionsmethoden zur Berechnung der Eigenwertprobleme konvergieren, wenn $\lambda_r^{k+1} \longrightarrow \lambda_r$ und $x_r^{k+1} \longrightarrow x_r$ gilt, daraus folgt $\alpha_r^{k+1} \longrightarrow 0$ für $k \to \infty$ und mit den Gleichungen (1.18) bzw. (2.6) gilt dann:

$$\lim_{k\to\infty} (A - \lambda_r^k I)x_r^{k+1} = 0 \quad \text{und} \quad \lim_{k\to\infty} \alpha_r^{k+1}(I - X_r X_r^T)x_r^k = 0 \quad \text{bzw.}$$
$$\lim_{k\to\infty} (\hat{A} - \lambda_r^k I)z_r^{k+1} = 0 \quad \text{und} \quad \lim_{k\to\infty} \alpha_r^{k+1}(I - Z_r Z_r^T)z_r^k = 0.$$

Satz 3.1 (Konvergenz von Algorithmus 1.24 für das Eigenwertproblem)
Sei A symmetrisch mit den orthonormierten Eigenvektoren x_i, $i = 1, ..., n$, zu den Eigenwerten $\lambda_1 < ... < \lambda_n$.

λ_r^k ist die Näherung von λ_r mit $|\lambda_r - \lambda_r^k| = \epsilon_\lambda > 0$ und

x_r^k ist die Näherung von x_r mit $\|x_r - x_r^k\| = \epsilon_x > 0$.

Der Algorithmus 1.24 für das Eigenwertproblem $Ax = \lambda x$

$$\begin{aligned}
\hat{x}_r^{k+1} &= -\alpha_r^{k+1}(A - \lambda_r^k I)^{-1} P_r x_r^k, \qquad P_r = I - X_r X_r^T, \\
x_r^{k+1} &= \frac{\hat{x}_r^{k+1}}{\|\hat{x}_r^{k+1}\|_2}, \\
\lambda_r^{k+1} &= \lambda_r^k - \frac{\alpha_r^{k+1}}{\|\hat{x}_r^{k+1}\|_2^2}
\end{aligned}$$

konvergiert kubisch.

Beweis: Für genügend große k und $\epsilon := \max\{\epsilon_1, \epsilon_2, ..., \epsilon_n, \epsilon_x, \epsilon_\lambda\}$, $\epsilon_i > 0$ für alle $i = 1, ..., n$, gilt

$$\begin{aligned}
x_r^k &= (1 + \epsilon_r)x_r + \sum_{\substack{i=1\\i\neq r}}^n \epsilon_i x_i \implies \\
\|x_r^k\|_2^2 = 1 &= 1 + 2\epsilon_r + \sum_{\substack{i=1\\i\neq r}}^n \epsilon_i^2 \implies \epsilon_r = \mathcal{O}(\epsilon^2).
\end{aligned} \tag{3.2}$$

Durch Entwicklung in eine geometrische Reihe gilt:

$$\frac{1}{1 + \mathcal{O}(\epsilon)} = 1 + \mathcal{O}(\epsilon), \qquad 0 < \epsilon < 1. \tag{3.3}$$

Mit $\lambda_r^k = (x_r^k)^T A x_r^k$ (x_r^k normiert) ergibt sich:

$$\begin{aligned}
\lambda_r^k = (x_r^k)^T A x_r^k &= ((1 + \epsilon_r)x_r + \sum_{\substack{i=1\\i\neq r}}^n \epsilon_i x_i)^T ((1 + \epsilon_r)\lambda_r x_r + \sum_{\substack{i=1\\i\neq r}}^n \epsilon_i \lambda_i x_i) \\
&= (1 + \epsilon_r)^2 \lambda_r + \sum_{\substack{i=1\\i\neq r}}^n \epsilon_i^2 \lambda_i.
\end{aligned}$$

Daraus folgt mit $\epsilon_r = \mathcal{O}(\epsilon^2)$ (siehe (3.2))

$$\epsilon_\lambda = |\lambda_r^k - \lambda_r| = |(1+2\epsilon_r)\lambda_r + \sum_{i=1}^{n} \epsilon_i^2 \lambda_i| = \mathcal{O}(\epsilon^2). \tag{3.4}$$

Mit (3.2) gilt

$$\begin{aligned}\hat{x}_r^{k+1} &= -\alpha_r^{k+1}(A - \lambda_r^k I)^{-1} P_r x_r^k \\ &= -\alpha_r^{k+1}(A - \lambda_r^k I)^{-1}((1+\epsilon_r)x_r + \sum_{\substack{i=1\\ i\neq r}}^{n} \epsilon_i x_i - \sum_{i=1}^{r-1} \epsilon_i x_i) \\ &= -\alpha_r^{k+1}(A - \lambda_r^k I)^{-1}(x_r + \sum_{i=r}^{n} \epsilon_i x_i).\end{aligned}$$

Ist x Eigenvektor zum Eigenwert λ der symmetrischen Matrix $A \in \mathbb{R}^{n\times n}$ und $k \in \mathbb{C}$ kein Eigenwert von A, dann ist $(\lambda - k)^{-1}$ ein Eigenwert von $(A - kI)^{-1}$. Also gilt:

$$\hat{x}_r^{k+1} = -\alpha_r^{k+1}\left(\frac{1}{\lambda_r - \lambda_r^k} x_r + \sum_{i=r}^{n} \frac{\epsilon_i}{\lambda_i - \lambda_r^k} x_i\right) \qquad \Longrightarrow \tag{3.5}$$

$$(\hat{x}_r^{k+1})^T \hat{x}_r^{k+1} = 1 = -\alpha_r^{k+1}\left(\frac{1}{\lambda_r - \lambda_r^k} + \frac{\epsilon_r}{\lambda_r - \lambda_r^k}\right) = -\alpha_r^{k+1} \frac{1+\epsilon_r}{\lambda_r - \lambda_r^k} \quad \Longrightarrow$$

$$\alpha_r^{k+1} = -\frac{\lambda_r - \lambda_r^k}{1+\epsilon_r}. \tag{3.6}$$

Mit (3.6) gilt nun für (3.5)

$$\begin{aligned}x_r^{k+1} &= \tfrac{\lambda_r - \lambda_r^k}{1+\epsilon_r}\left(\tfrac{1}{\lambda_r - \lambda_r^k} x_r + \sum_{i=r}^{n} \tfrac{\epsilon_i}{\lambda_i - \lambda_r^k} x_i\right) = \tfrac{1}{1+\epsilon_r} x_r + \tfrac{\epsilon_r}{1+\epsilon_r} x_r + \sum_{i=r+1}^{n} \epsilon_i \tfrac{\lambda_r - \lambda_r^k}{\lambda_i - \lambda_r^k} x_i \\ &= x_r + \sum_{i=r+1}^{n} \epsilon_i \tfrac{\lambda_r - \lambda_r^k}{\lambda_i - \lambda_r^k} x_i\end{aligned} \tag{3.7}$$

und daraus folgt mit Gleichung (3.4) und wegen der Monotonie der Eigenwerte $|\lambda_i - \lambda_r^k| > |\lambda_r - \lambda_r^k|$

$$\|x_r^{k+1} - x_r\| = \mathcal{O}(\epsilon^3). \tag{3.8}$$

Mit (3.6), (3.4) und (3.3) gilt also

$$\begin{aligned}\lambda_r^{k+1} &= \lambda_r^k - \tfrac{\alpha_r^{k+1}}{\|\hat{x}_r^{k+1}\|_2} = \lambda_r^k + \tfrac{\lambda_r - \lambda_r^k}{(1+\epsilon_r)\|\hat{x}_r^{k+1}\|_2} = \lambda_r^k + \tfrac{\lambda_r - \lambda_r^k}{1+\mathcal{O}(\epsilon)} \\ &= \lambda_r^k + (\lambda_r - \lambda_r^k)(1+\mathcal{O}(\epsilon)) = \lambda_r^k + \lambda_r - \lambda_r^k + (\lambda_r - \lambda_r^k)\mathcal{O}(\epsilon) \\ &= \lambda_r^k + (\lambda_r - \lambda_r^k)\mathcal{O}(\epsilon) = \lambda_r^k + \mathcal{O}(\epsilon^3) \quad \Longrightarrow\end{aligned} \tag{3.9}$$

$$|\lambda_r^{k+1} - \lambda_r| = \mathcal{O}(\epsilon^3). \tag{3.10}$$

Mit (3.8) und (3.10) ist der Satz bewiesen. □

Bemerkung 3.11
Bei der Berechnung von λ_r^{k+1} mit dem Rayleigh-Quotienten, d.h. unabhängig vom Algorithmus, ergibt sich mit (3.4) nur eine quadratische Konvergenz:

$$\begin{aligned}\lambda_r^{k+1} &= \tfrac{(\hat{x}_r^{k+1})^T A \hat{x}_r^{k+1}}{\|\hat{x}_r^{k+1}\|_2} = (x_r^{k+1})^T A x_r^{k+1} \\ &= (1+\epsilon_r)^2\lambda_r + \sum_{\substack{i=1\\ i\neq r}}^{n} \epsilon_i^2 \lambda_i \implies |\lambda_r^{k+1} - \lambda_r| = \mathcal{O}(\epsilon^2).\end{aligned}$$

$\circ$

Bemerkung 3.12 (Konvergenz für mehrfache Eigenwerte)
Hat A mehrfache Eigenwerte $\lambda_1 < \lambda_2 < ... < \lambda_{r+1} = ... = \lambda_{r+k} < \lambda_{r+k+1} < ... < \lambda_n$, so ist $\frac{|\lambda_r - \lambda_r^k|}{|\lambda_i - \lambda_r^k|} = 1$ für $i = r+1, ..., r+k$, und es gilt also analog zu (3.7):

$$x_r^{k+1} = x_r + \sum_{i=r+1}^{n} \epsilon_i \frac{\lambda_r - \lambda_r^k}{\lambda_i - \lambda_r^k} x_i = x_r + \sum_{i=r+1}^{r+k} \epsilon_i x_i + \sum_{i=r+k+1}^{n} \epsilon_i \frac{\lambda_r - \lambda_r^k}{\lambda_i - \lambda_r^k} x_i.$$

Sind alle Eigenwerte gleich $\lambda_1 = ... = \lambda_{r+1} = ... = \lambda_{r+k} = \lambda_{r+k+1} = ... = \lambda_n$, so gilt

$$x_r^{k+1} = x_r + \sum_{i=r+1}^{n} \epsilon_i x_i \quad \implies \quad \|x_r^{k+1} - x_r\| = \mathcal{O}(\epsilon).$$

Algorithmus 1.24 konvergiert für mehrfache Eigenvektoren also mindestens linear. $\circ$

Satz 3.13 (Konvergenz des allgemeinen Eigenwertproblems)
Das Matrixpaar (A, B), A symmetrisch und B symmetrisch und positiv definit, habe die B-orthonormierten Eigenvektoren x_i zu den Eigenwerten $\lambda_1 < ... < \lambda_n$.

λ_r^m ist die Näherung von λ_r mit $|\lambda_r - \lambda_r^m| = \epsilon_\lambda > 0$ und

x_r^m ist die Näherung von x_r mit $\|x_r - x_r^m\| = \epsilon_x > 0$.

Der Algorithmus 2.10 für das allgemeine Eigenwertproblem $Ax = \lambda Bx$

$$\begin{aligned}\hat{x}_r^{k+1} &= -\alpha_r^{k+1}(A - \lambda_r^k B)^{-1} B P_r x_r^k, \qquad P_r = I - X_r X_r^T B \\ x_r^{k+1} &= \tfrac{\hat{x}_r^{k+1}}{\|\hat{x}_r^{k+1}\|_B} \\ \lambda_r^{k+1} &= \lambda_r^k - \tfrac{\alpha_r^{k+1}}{\|\hat{x}_r^{k+1}\|_B^2}\end{aligned}$$

konvergiert kubisch.

Beweis: Für genügend große k und $\epsilon := \max\{\epsilon_1, \epsilon_2, ..., \epsilon_n, \epsilon_x, \epsilon_\lambda\}$, $\epsilon_i > 0$ für alle $i = 1, ..., n$, gilt

$$x_r^k = (1+\epsilon_r)x_r + \sum_{\substack{i=1 \\ i\neq r}}^{n} \epsilon_i x_i \quad \Longrightarrow \tag{3.14}$$

$$\|x_r^k\|_B^2 \;=\; 1 = (x_r^k)^T B x_r^k = 1 + 2\epsilon_r + \sum_{\substack{i=1 \\ i\neq r}}^{n} \epsilon_i^2 \implies \epsilon_r = \mathcal{O}(\epsilon^2). \tag{3.15}$$

Mit $\lambda_r^k = (x_r^k)^T A x_r^k$ (x_r^k ist B-orthogonal) ergibt sich:

$$\begin{aligned} \lambda_r^k &= (x_r^k)^T A x_r^k = ((1+\epsilon_r)x_r + \sum_{\substack{i=1 \\ i\neq r}}^{n} \epsilon_i x_i)^T((1+\epsilon_r)\lambda_r B x_r + \sum_{\substack{i=1 \\ i\neq r}}^{n} \epsilon_i \lambda_i B x_i) \\ &= (1+\epsilon_r)^2 \lambda_r + \sum_{\substack{i=1 \\ i\neq r}}^{n} \epsilon_i^2 \lambda_i. \end{aligned}$$

Daraus folgt mit $\epsilon_r = \mathcal{O}(\epsilon^2)$ (siehe (3.15))

$$\epsilon_\lambda = |\lambda_r^k - \lambda_r| = |(1+2\epsilon_r)\lambda_r + \sum_{i=1}^{n} \epsilon_i^2 \lambda_i| = \mathcal{O}(\epsilon^2). \tag{3.16}$$

Mit (3.14) gilt

$$\begin{aligned} \hat{x}_r^{k+1} &= -\alpha_r^{k+1}(A - \lambda_r^k B)^{-1} B P_r x_r^k \\ &= -\alpha_r^{k+1}(A - \lambda_r^k B)^{-1} B((1+\epsilon_r)x_r + \sum_{\substack{i=1 \\ i\neq r}}^{n} \epsilon_i x_i - \sum_{i=1}^{r-1} \epsilon_i x_i) \\ &= -\alpha_r^{k+1}(A - \lambda_r^k B)^{-1} B(x_r + \sum_{i=r}^{n} \epsilon_i x_i) \end{aligned}$$

und analog zu (3.5) gilt

$$\hat{x}_r^{k+1} = -\alpha_r^{k+1}\left(\frac{1}{\lambda_r - \lambda_r^k} x_r + \sum_{i=r}^{n} \frac{\epsilon_i}{\lambda_i - \lambda_r^k} x_i\right) \qquad \Longrightarrow \tag{3.17}$$

$$(\hat{x}_r^{k+1})^T B \hat{x}_r^{k+1} = 1 = -\alpha_r^{k+1}\left(\frac{1}{\lambda_r - \lambda_r^k} + \frac{\epsilon_r}{\lambda_r - \lambda_r^k}\right) = -\alpha_r^{k+1}\frac{1+\epsilon_r}{\lambda_r - \lambda_r^k} \quad \Longrightarrow$$

$$\alpha_r^{k+1} = -\frac{\lambda_r - \lambda_r^k}{1+\epsilon_r}. \tag{3.18}$$

Mit (3.18) gilt nun für (3.17)

$$x_r^{k+1} = \frac{\lambda_r - \lambda_r^k}{1+\epsilon_r}\left(\frac{1}{\lambda_r - \lambda_r^k} x_r + \sum_{i=r}^{n} \frac{\epsilon_i}{\lambda_i - \lambda_r^k} x_i\right) = x_r + \sum_{i=r+1}^{n} \epsilon_i \frac{\lambda_r - \lambda_r^k}{\lambda_i - \lambda_r^k} x_i$$

und daraus folgt mit (3.16) und $|\lambda_i - \lambda_r^k| < |\lambda_r - \lambda_r^k|$

$$\|x_r^{k+1} - x_r\| = \mathcal{O}(\epsilon^3). \tag{3.19}$$

Mit (3.18), (3.16) und (3.3) ist dann

$$\begin{aligned} \lambda_r^{k+1} &= \lambda_r^k - \tfrac{\alpha_r^{k+1}}{\|\hat{x}_r^{k+1}\|_B} = \lambda_r^k + \tfrac{\lambda_r - \lambda_r^k}{(1+\epsilon_r)\|\hat{x}_r^{k+1}\|_B} \\ &= \lambda_r^k + \tfrac{\lambda_r - \lambda_r^k}{1+\mathcal{O}(\epsilon)} = \lambda_r^k + \mathcal{O}(\epsilon^3) \quad \Longrightarrow \end{aligned}$$

$$|\lambda_r^{k+1} - \lambda_r| = \mathcal{O}(\epsilon^3). \tag{3.20}$$

Mit (3.19) und (3.20) ist der Satz bewiesen. □

Einige Eigenschaften der Verfahren:

Satz 3.21
Für die in Algorithmus 1.24 und Algorithmus 2.10 beschriebenen Verfahren zur Lösung des symmetrischen Eigenwertproblems $Ax = \lambda x$ bzw. des symmetrischen allgemeinen Problems $Ax = \lambda Bx$ gilt ab einer gewissen Näherung l:
Wenn x_r^{l+1} eine bessere Approximation von x_r ist als x_r^l, dann ist auch λ_r^{l+1} eine bessere Approximation des Eigenwerts λ_r als λ_r^l, d.h. ist $(x_r^i)_{i=l,l+1,\ldots}$ eine monotone Folge, so ist auch $(\lambda_r^i)_{i=l,l+1,\ldots}$ eine monotone Folge.

Beweis: (für Algorithmus 1.24)
Für $k = l$ ist $\epsilon < 1$ und es ist zu zeigen, daß aus $\|x_r^{k+1} - x_r\|_2 < \|x_r^k - x_r\|_2$ folgt $|\lambda_r^{k+1} - \lambda_r| < |\lambda_r^k - \lambda_r|$.
Mit (3.2) ist

$$\|x_r^k - x_r\|_2 = \|\sum_{i=1}^{n} \epsilon_i x_i\|_2 \le \sum_{i=1}^{n} \epsilon_i < \epsilon_x < \epsilon \quad \Longrightarrow \quad \sum_{i=r}^{n} \epsilon_i < \epsilon. \tag{3.22}$$

Es gilt wegen (3.9)

$$\lambda_r^{k+1} - \lambda_r^k = -\frac{\alpha_r^{k+1}}{\|\hat{x}_r^{k+1}\|_2} \Longrightarrow -\alpha_r^{k+1} = (\lambda_r^{k+1} - \lambda_r^k)\|\hat{x}_r^{k+1}\|_2.$$

Aus (3.5) folgt dann

$$\hat{x}_r^{k+1} = (\lambda_r^{k+1} - \lambda_r^k)\|\hat{x}_r^{k+1}\|_2 \left(\frac{1}{\lambda_r - \lambda_r^k} x_r + \sum_{i=r}^{n} \frac{\epsilon_i}{\lambda_i - \lambda_r^k} x_i \right) \qquad \Longrightarrow$$

$$\frac{\hat{x}_r^{k+1}}{\|\hat{x}_r^{k+1}\|_2} = x_r^{k+1} = (\lambda_r^{k+1} - \lambda_r^k) \left(\frac{1}{\lambda_r - \lambda_r^k} x_r + \sum_{i=r}^{n} \frac{\epsilon_i}{\lambda_i - \lambda_r^k} x_i \right).$$

Nun kann gezeigt werden:

$$\begin{aligned} \|x_r^{k+1} - x_r\|_2 &= \left\| \tfrac{\lambda_r^{k+1} - \lambda_r^k}{\lambda_r - \lambda_r^k} x_r - x_r + \sum_{i=r}^{n} \tfrac{\epsilon_i(\lambda_r^{k+1} - \lambda_r^k)}{\lambda_r - \lambda_r^k} x_i \right\|_2 \\ &= \left\| \tfrac{\lambda_r^{k+1} - \lambda_r^k - \lambda_r + \lambda_r^k}{\lambda_r - \lambda_r^k} x_r + \sum_{i=r}^{n} \tfrac{\epsilon_i(\lambda_r^{k+1} - \lambda_r^k)}{\lambda_r - \lambda_r^k} x_i \right\|_2 \\ &\leq \tfrac{|\lambda_r^{k+1} - \lambda_r|}{|\lambda_r^k - \lambda_r|} + \tfrac{|\lambda_r^{k+1} - \lambda_r^k|}{|\lambda_r^k - \lambda_r|} \sum_{i=r}^{n} \epsilon_i \leq \tfrac{|\lambda_r^{k+1} - \lambda_r|}{|\lambda_r^k - \lambda_r|} + \tfrac{|\lambda_r^{k+1} - \lambda_r^k|}{|\lambda_r^k - \lambda_r|} \epsilon < \epsilon. \end{aligned}$$

Daraus folgt mit $\epsilon < 1$

$$|\lambda_r^{k+1} - \lambda_r| < \epsilon|\lambda_r^k - \lambda_r| - \epsilon|\lambda_r^{k+1} - \lambda_r^k| < \epsilon|\lambda_r^k - \lambda_r| < |\lambda_r^k - \lambda_r|.$$

Analog kann der Satz für Algorithmus 2.10 bewiesen werden. □

Weitere wichtige Eigenschaften der Algorithmen werden im Folgenden aufgezählt:

- Schon jeder Startvektor $x_r^0 = e_r$, $r = 1, ..., n$, wird auf den Unterraum span$\{x_r, ..., x_n\}$ bzw. span$\{Bx_r, ..., Bx_n\}$ abgebildet.
- Durch Rundungsfehler hat die Approximation x_r^{k+1} auch Komponenten in Richtung der bekannten Eigenvektoren. Durch die Projektion P_r bzw. $\mathcal{P}_r$ werden diese Komponenten unterdrückt.
- Auf Grund der Voraussetzungen liefern die Algorithmen am Ende direkt alle orthonormierten bzw. B-orthonormierten Eigenvektoren.
- $X_r = [x_1, ..., x_{r-1}]$ muß in den Algorithmen aber nur explizit gebildet werden, wenn am Ende des Algorithmus alle Eigenvektoren gespeichert sein sollen.
- Die Matrix X der berechneten Eigenvektoren $x_1, x_2, ..., x_n$ transformiert A in Diagonalgestalt mit den Eigenwerten in der Diagonalen und beim allgemeinen Eigenwertproblem B in die Einheitsmatrix

$$\begin{aligned} X^T A X &= D = \mathrm{diag}(\lambda_i) \qquad i = 1, ..., n \\ X^T B X &= I. \end{aligned}$$

Literatur

[BuGe85] W. Bunse, A. Bunse-Gerstner, *Numerische lineare Algebra,* Teubner Studienbücher Mathematik, Stuttgart 1985.

[Doll80] J.Doll, *Ein iteratives Verfahren zur Lösung des nichtsymmetrischen Eigenwertproblems,* Dissertation, FU Berlin, 1980.

[GoLo96] G.H. Golub, C.F. van Loan, *Matrix Computations,* The Johns Hopkins University Press, Baltimore 1996.

Argonne Nation laboratary, October 1992.

[Parl98] B.N. Parlett, *The symmetric Eigenvalue Problem,* Prentice-Hall, U.S.A. 1998.

[Schen92] U. Schendel, *Projection Methods for Solving Eigenvalue Problems,* Preprint No. A-92-1, Fachbereich Mathematik und Informatik Freie Universität Berlin, 1992.

[SchZi97] U. Schendel, A. Ziegler, *Numerische Algorithmen der Linearen Algebra,* Vorlesungsskript, Fachbereich Mathematik und Informatik, Freie Universität Berlin, 1997.

[Wilkin92] J.H. Wilkinson, *The Algebraic Eigenvalue Problem,* Clarendon Press, Oxford, 1992.