

implémenté dans Matlab, peut être
schematisé comme suit :

- (a) Calculs faire de Sides $A = U T \cdot U^*$
avec U unitaire et T triangulaire supérieure.
Il reste à calculer $f(T)$ car $f(A) = U f(T) U^*$.
- (b) Couper $T = \begin{bmatrix} T_{11} & \dots & T_{1e} \\ 0 & T_{22} & \vdots \\ \vdots & \dots & \vdots \\ 0 & \dots & 0 & T_{ee} \end{bmatrix}$ en blocs, avec
 T_{jj} carrés (triangulaires supérieurs).
- (c) Evaluer $f_{jj} = f(T_{jj})$ d'une manière directe
(Taylor, interpolations polynomiales/rationnelles)
- (d) Partitionner $f(T) = \begin{bmatrix} F_{11} & \dots & F_{1e} \\ \vdots & \ddots & \vdots \\ 0 & \dots & F_{ee} \end{bmatrix}$ cone T ,

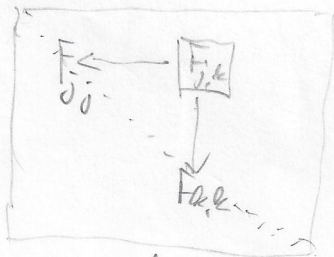
et trouver les blocs $F_{j,k}$, $k \geq j$ en utilisant
que $f(T) \cdot T = T \cdot f(T) \Rightarrow$ en comparant les blocs j,k

$$\sum_{i=j}^k F_{j,i} T_{i,k} = \sum_{i=j}^k T_{j,i} F_{i,k}$$

\Rightarrow en procédant par diagonale de j constant croissant :

$F_{j,k}$ est sol. de l'équation de Sylvester

$$(5) \quad T_{j,j} \cdot X - X \cdot T_{k,k} = \sum_{i=j}^{k-1} F_{j,i} T_{i,k} - \sum_{i=j+1}^k T_{j,i} F_{i,k}$$



Etudions chaque étape en détail :

§2.1. Calculez la forme de Schur

→ Golub, Van Loan, Matrix Computations

Existence/Unicité? → Lascaux, Théodor, A.N. matricielle appliquée à l'art de l'ingénieur

2.1.1. Lemme: Toute matrice $A \in \mathbb{C}^{n \times n}$ peut être factorisée $A = U \cdot T \cdot U^*$ avec $U, T \in \mathbb{C}^{n \times n}$, $U^* = U^*$ (unitaire) et T triangulaire supérieure.

N.B.1: $\sigma(A) = \sigma(T) = \{ \text{éléments diagonaux de } T \}$

N.B.2: si A est hermitienne alors aussi $T \Rightarrow T$ diag.

Même si A est normale alors aussi T ($T^*T - TT^* = U^*(A^*A - AA^*)U = 0$), et donc T doit être diagonal.

Preuve: par récurrence sur n , $n=1$ trivial.

Soit (λ, y) un élément propre de A , sans perte de la généralité $\|y\|=1 \Rightarrow$ en complétant par trouver une ONB: $\exists Q$ unitaire, $Q = (y | *) \Rightarrow$

$$Q^* A \cdot Q = Q^* \left[\begin{array}{c|c} \lambda & * \\ \hline 0 & \tilde{A} \\ \vdots & \\ 0 & \end{array} \right]$$

Par hypothèse de récurrence $\tilde{A} = \tilde{U} \tilde{T} \tilde{U}^{-1}$

et alors avec $U = Q \cdot \left[\begin{array}{c|c} 1 & 0 \dots 0 \\ \hline 0 & \tilde{U} \\ \vdots & \end{array} \right]$, U unitaire

$$U^* A U = \left[\begin{array}{c|c} \lambda & * \dots * \\ \hline 0 & \tilde{T} \\ \vdots & \end{array} \right] \text{ triangulaire supérieure. } \blacksquare$$

En général, une telle décomposition de Schur n'est pas unique. Aussi, la méthode de la preuve n'est pas viable numériquement car il faut connaître à priori les valeurs propres. M.g. la méthode QR (pas calculer numériquement les v.p.) calcule en effet directement une forme de Schur.

2.1.2. Méthode QR :

Partant de $A = A_0$ et une suite de $\mu_k, \dots \in \mathbb{C}$,
on calcule successivement

$$A_k - \mu_k I_n = Q_k R_k \quad Q_k \text{ unitaire, } R_k \text{ triang sup}$$

$$A_{k+1} = R_k Q_k + \mu_k I_n$$

Notons que $A_k = Q_{k-1}^* A_{k-1} Q_{k-1} = \dots = U_k^* A U_k$
avec $U_k = Q_{k-1} \dots Q_0$ unitaire. c'est à dire, A_k est semblable à A .
On montre que par A géométrique et par un large choix de μ_k que $A_k \rightarrow T$ triangulaire supérieure, voir TP.
 $U_k \rightarrow U$ unitaire, et alors $T = U^* A U$
décomposition de Schur (convergence exp. aussi pour A non sym)

Une implémentation naive de 2.1.2. nécessite $O(n^3)$ opérations par itérations, mais seulement $O(n^2)$ si la matrice A_0 est sous forme Hessenberg.
(supérieure)

x	x	x	x
x	x	x	x
x	x	x	x
x	x	x	x

2.1.3. Réduction à la forme Hessenberg (matlab: hess)

On adaptera dans la suite les notions Matlab pour les sous-matrices.

$\forall x \in \mathbb{C}^n \exists w \in \mathbb{C}^n$ de norme 1 t.q. $H_n(w) = I_n - 2ww^*$
est unitaire, et $H_n(w) \cdot x = \|x\| e_1$, trafo de Householder.

⊗ > Décrivons en détail la première transformation.
On détaille H_{n-1} de sorte que $H_{n-1} A(2:n, 1) = \alpha \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$
trafo de Householder

$\alpha > 0$. Alors

1	0	...	0
0	H_{n-1}		

 · A ·

1	0	...	0
0	H_{n-1}		

^{*} =

α	*	...	*
0	*		
0	*		
0	*		

 ·

1	0	...	0
0	H_{n-1}		

 =

α	*	...	*
0	*		
0	*		
0	*		

⊗ Implémenter $H \cdot A$ signifie que l'on ajoute des multiples de w à chaque colonne de A (de même $A \cdot H$ on ajoute des multiples de w^* à chaque ligne de A) \Rightarrow complexité $O(n^2)$.

En appliquant le même procédé à la sous-matrice $(2:n, 2:n)$ du résultat etc, on trouve en complexité $\mathcal{O}(n^2)$ Q unitaire de sorte que $Q^* A Q = A_0$ de forme Hessenberg sup., $Q = \begin{bmatrix} 1 & & \\ & H_{n-1} & \\ & & \ddots \end{bmatrix} \cdot \begin{bmatrix} 1 & & \\ & 1 & \\ & & \ddots \end{bmatrix} \cdot \dots \cdot \begin{bmatrix} 1 & & \\ & 1 & \\ & & H_2 \end{bmatrix}$

2.1.4. Détails d'implémentation de la méthode QR

M.q. que si A_0 est de forme de Hessenberg alors aussi A_1, A_2, \dots , ce qui permet de mettre en oeuvre une itération en complexité $\mathcal{O}(n^2)$.

$$A_k = \begin{bmatrix} \mu I & & & \\ & 1 & & * \\ & 0 & \boxed{2} & \\ & \vdots & & \ddots \\ & 0 & \dots & 0 & \boxed{n} & * \end{bmatrix}$$

à l'étape $j=1, \dots, n-1$, on construit une matrice de Givens $G_j = \begin{bmatrix} I_{j-1} & & \\ & H_j & \\ & & I_{n-j-1} \end{bmatrix}$ de sorte que $H_j *$

$(G_{j-1} \dots G_1 A_k)(j:j+1, j) = \begin{pmatrix} \alpha \\ 0 \end{pmatrix}$, $\alpha > 0 \Rightarrow$ par récurrence sur j : $G_j \dots G_1 A_k = \begin{bmatrix} \times & & \\ & \times & \\ & & \times & \\ & & & \times & \\ & & & & \times & \\ & & & & & \times & \\ & & & & & & \times & \\ & & & & & & & \times & \\ & & & & & & & & \times & \\ & & & & & & & & & \times \end{bmatrix}$

$\Rightarrow Q_k^* = G_{n-1} \dots G_1$, $R_k = Q_k^* (A_k - \mu I)$ éliminer

\Rightarrow mise à jour $U_{k+1} = Q_k U_k = Q_k \cdot G_1 \dots G_{n-1}$
 $A_{k+1} = \mu I + R_k Q_k = \mu I + R_k \cdot G_{n-1}^* \dots G_1^*$

comme toute multiplication avec G_j à gauche ou à droite coûte $\mathcal{O}(n)$ opérations, on arrive à $\mathcal{O}(n^2)$ opérations.

En pratique, on met à jour un indice $\delta = \delta_k$ de sorte que

$$A_k = \begin{bmatrix} \times & & & \\ & \times & & \\ & & \times & \\ & & & \times & \\ & & & & \times & \\ & & & & & \times & \\ & & & & & & \times & \\ & & & & & & & \times & \\ & & & & & & & & \times & \\ & & & & & & & & & \times \end{bmatrix} \leftarrow \delta_k$$

"petit"

de sorte que $A_k(j+1, j)$ "petit" par $j = \delta_k, \dots, n$ mais pas par $j = \delta_k - 1$ (c'est-à-dire, $|A_k(j+1, j)| \leq \beta \cdot (|A_k(j, j)| + |A_k(j+1, j+1)|)$ avec $\beta = 10 \times \text{eps}$?)

et on effectue l'élimination par R_k pour $j=1, 2, \dots, k-1$ (∞ est considéré comme 0).

Choix du paramètre ?

Shift de Rayleigh : $\mu_k = A_k(\delta_k, \delta_k)$

Shift de Wilkinson : μ_k v.p. de $A_k(\delta_{k-1} : \delta_k, \delta_{k-1} : \delta_k)$

le plus proche de $A_k(\delta_k, \delta_k)$.

double shift implicite dans deux itérations successives

on prend les deux v.p. de cette matrice (implémentation plus compliquée, point de vue dans $\mathbb{R}^{n \times n}$, mais T admet éventuellement

des blocs diagonaux 2×2 !

Cette approche est intriguement stable ! (Backward stability).

§ 2.2. Résolution des équations de Sylvester (S)

En changeant légèrement des notations, il reste à résoudre en X des équations de Sylvester $AX - XB = C$ avec A, B carrés triangulaires supérieurs d'ordre p et q , généralement p, q petit devant n , mais $p \neq q$ possible, et $C, X \in \mathbb{C}^{p \times q}$.

2.2.1. Lemme : Cette équation de Sylvester admet une sol. ~~est~~ unique par tout second membre C
 $\Leftrightarrow \sigma(A) \cap \sigma(B) = \emptyset$.

Preuve : \Rightarrow si $\alpha \in \sigma(A) \cap \sigma(B)$ avec $Ax = \alpha x, yB = \alpha y$ alors on peut ajouter à toute solution X le terme $\alpha \cdot x \cdot y$, et on garde une solution \Rightarrow non unicité.

\Leftarrow Pour $j=1, \dots, q$ on multiplie par e_j :

$$[A - B e_j, j] X(1:n, j) = C(1:n, j) + \sum_{i=1}^{j-1} B(i, j) X(1:n, i)$$

inversible par hypothèse $B(i, j) \neq \sigma(A)$.

Dans l'approche (S) de Parlett, la matrice C elle-même risque d'être entachée d'erreurs car résultat des calculs précédents.

Pour ne pas avoir une accumulation d'erreurs trop important, on souhaite qu'une petite perturbation ΔC n'induit pas une grande perturbation ΔX dans la solution: si on note \tilde{x} sol. de $A\tilde{x} + \tilde{x}B = C + \Delta C$ alors

$$\frac{\|x - \tilde{x}\|}{\|\Delta C\|} = \frac{\|y\|}{\|Ay - yB\|} \quad \text{avec } y = x - \tilde{x}$$

ce qui motive le vœux que $\text{spread}(A, B) := \min \frac{\|Ay - yB\|}{\|y\|}$ soit le plus grand possible (choix de A, B dans (S)).

Malheureusement, on dispose seulement des sous-estimations de $\text{spread}(A, B)$ que l'on pourrait chercher à maximiser:

avec $Ax = \lambda x$, $yB = \mu y$, en posant $y = x \cdot y \in \mathbb{C}^{p \times q}$, nous obtenons $\text{spread}(A, B) \leq \min_{\lambda \in \sigma(A), \mu \in \sigma(B)} |\lambda - \mu| =: \text{split}(\sigma(A), \sigma(B))$.

2.2.2. Exo pour A, B normales, montrer

que $\text{spread}(A, B) \leq \text{dist}(\sigma(A), \sigma(B)) \leq \sqrt{\min(p, q)} \text{spread}(A, B)$.

Idee: on montre d'abord qu'il suffit de considérer A, B diagonales. Ensuite on utilise les inégalités classiques entre la norme matricielle euclidienne et la norme de Frobenius.

§ 2.3. Comment partitionner

Comme vu en § 2.2., la stabilité de la méthode de Parlett dépend du fait si, $\forall j, k = 1, \dots, l, j \neq k$, $\text{spread}(T_{jj}, T_{kk})$ est suff. grand. On cherche donc à partitionner $\sigma(A)$, la diagonale de T , de sorte que $\text{spread}(\sigma(T_{jj}), \sigma(T_{kk}))$, la distance entre les éléments diagonaux de T_{jj} , et celles de T_{kk} soit le plus grand possible $\forall j \neq k$. En particulier, deux valeurs propres très proches (ou idéiques) devraient se trouver dans le même bloc.

La théorie dit que, généralement, $|t_{11}| \leq |t_{22}| \leq \dots$ donc, si jamais deux v.p. distincts sont de module différents, ^{2 v.p. - presque idéiques} ils devraient se trouver successivement sur la diagonale, ce qui facilite le partitionnement en blocs de T .

Néanmoins, il se peut que, après un premier calcul de $T = U^* A U$, on aurait souhaité de regrouper t_{jj} et t_{kk} par $j < k$.

ceci se fait en permutant successivement t_{ij} et $t_{i+1, i+1}$ par $i = j, j+1, \dots, k-1$

en multipliant avec $G_i = \begin{bmatrix} 1 & & & \\ & \ddots & & \\ & & \square_{n_i} & \\ & & & \ddots \\ & & & & 1 \end{bmatrix}$ et G_i^*

le bloc 2×2 unitaire choisi de sorte que $\begin{bmatrix} t(i, i) & t(i, i+1) \\ 0 & t(i+1, i+1) \end{bmatrix}$ devient $\begin{bmatrix} t(i+1, i+1) & * \\ 0 & t(i, i) \end{bmatrix}$

(voir preuve de 2.1.1.). Trouver composantes connexes dans graphe $\sigma(T)$ - sommets, arc entre deux t_{jj} si leur distance est $\leq \delta$. (42)

de la co-positivité de C_{ij}

↑