UNIVERSITE DES SCIENCES ET TECHNIQUES
DE LILLE FLANDRES ARTOIS

N° d'ordre : 210



# THÈSE

**Nouveau Régime**

**présentée à**

L'UNIVERSITE DES SCIENCES ET TECHNIQUES DE LILLE FLANDRES ARTOIS

**pour obtenir le titre de**

DOCTEUR EN Mathématiques appliquées

**par**

Hassane SADOK

# ACCELERATION DE LA CONVERGENCE DE SUITES VECTORIELLES ET METHODES DE POINT FIXE.

Soutenue le 3 1 mars 1988 devant la Commission d'Examen

| | | |
|---|---|---|
| Président : | C. | BREZINSKI |
| Rapporteurs : | A. | CUYT |
| | B. | GERMAIN BONNE |
| Examinateur : | P. | SABLONNIERE |

# Chapitre I

# Etude de la transformation d'Henrici

# Introduction

La transformation d'Henrici est une méthode d'accélération de la convergence de suites de vecteurs. Cette transformation a été définie comme une généralisation du procédé $\Delta^2$ d'Aitken qui est une méthode bien connue d'accélération de la convergence de suites scalaires.

Nous commencerons par rappeler les propriétés que possède le procédé $\Delta^2$ d'Aitken, qui seront généralisées pour la transformation d'Henrici. Nous l'étudierons en caractérisant son noyau et en donnant un théorème de convergence ainsi qu'un théorème d'accélération de la convergence.
Enfin nous donnerons deux méthodes pour sa mise en oeuvre, en utilisant pour la première le H-algorithme et pour la seconde une méthode d'orthonormalisation ainsi que sa mise à jour. Pour illustrer le comportement de ces deux méthodes nous donnerons quelques exemples numériques.

## 1. Principales notations et définitions

### a) notations

| | |
|---|---|
| $\mathbb{R}$ | corps des nombres réels |
| $\mathbb{C}$ | corps des nombres complexes |
| $\mathbb{C}^*$ | $\mathbb{C} - \{0\}$ |
| | si $z \in \mathbb{C}$ $\quad z = u + iv$ où $u \in \mathbb{R}$ et $v \in \mathbb{R}$ |
| | $\bar{z} = u - iv$ et $\lvert z \rvert = \sqrt{u^2 + v^2}$ |
| | $\bar{z}$ : le nombre complexe conjugué du nombre $z$. |
| | $\lvert z \rvert$ : le module du nombre complexe $z$. |
| $\mathbb{R}^p$ | espace vectoriel à composantes réelles, de dimension $p$. |
| $\mathbb{C}^p$ | espace vectoriel à composantes complexes, de dimension $p$. |
| | Pour éviter de considérer successivement $\mathbb{R}$ ou $\mathbb{C}$, on notera $K$ l'un des ces corps. |
| $L(K^m, K^n)$ | l'ensemble des matrices rectangulaires $(m, n)$, dont les éléments appartiennent au corps $K$. ($m$ : nombres de lignes) |
| $L(K^p)$ | l'ensemble des matrices carrées d'ordre $p$, dont les éléments appartiennent au corps $K$. |
| $O$ | matrice nulle. |
| $I_p$ | matrice unité d'ordre $p$. |
| $A^T$ | matrice transposée de $A$, $A^T = (b_{ij})$ et $A = (a_{ij})$ |
| | $b_{ij} = a_{ji} \quad \forall i, \forall j$ |

| | |
|---|---|
| $A^*$ | matrice adjointe de la matrice $A$, $A \in L(\mathbb{C}^p)$ , $A = (a_{ij})$ <br> $a_{ij} \in \mathbb{C}$ $\quad A^* = (a^*_{ij}) = (\overline{a_{ji}})$ |
| $A^{-1}$ | matrice inverse de la matrice $A$. <br> $A A^{-1} = A^{-1} A = I_p$ si $A \in L(K^p)$. |
| $\mathrm{diag}(a_1, \dots, a_p)$ | matrice diagonale dont les éléments diagonaux sont $a_1, a_2, \dots, a_p$. |
| $\det(A)$ | déterminant de $A$. |
| $\lambda_i(A), \lambda_i$ | valeur propre de $A$. |
| $\rho(A) = \max_i \lvert \lambda_i(A) \rvert$ | rayon spectral de $A$. |
| $(x)_i$ | i-ème composante de $x$ |
| $\lVert x \rVert_\infty = \max_i \lvert x_i \rvert$ | norme vectorielle $l_\infty$ de $x$ |
| $\lVert x \rVert$ | une norme vectorielle du vecteur $x$ |
| $\lVert x \rVert_2 = \left( \sum_{i=1}^{p} \lvert (x)_i \rvert^2 \right)^{1/2}$ | norme euclidienne du vecteur $x$ |
| $(x, y) = {}^t\bar{y}\, x = \sum_{i=1}^{p} x_i \bar{y}_i$ | produit scalaire de deux vecteurs de $K^p$. |
| $\lVert A \rVert_2 = \max_{x \neq 0} \dfrac{\lVert Ax \rVert_2}{\lVert x \rVert_2} = \sqrt{\rho(A^* A)}$ | norme matricielle induite par la norme vectorielle $\lVert\ \rVert_2$. |
| $\lVert A \rVert_F = \left( \sum_{ij} \lvert a_{ij} \rvert^2 \right)^{1/2}$ | norme matricielle de Frobenius. |

| | |
|---|---|
| $A = (a^1, a^2, \ldots, a^p)$ | la matrice A dont la i-ème colonne est le vecteur $a^i$. |
| $V[a^1, a^2, \ldots, a^k]$ | sous espace vectoriel engendré par les vecteurs $a^1, a^2, \ldots, a^k$. |

---

Si $\{s_n\}$ est une suite de vecteurs complexes : $s_n \in \mathbb{C}^p$

On note :

$$\Delta s_n = s_{n+1} - s_n \quad \text{et} \quad \Delta^2 s_n = \Delta(\Delta s_n) = \Delta s_{n+1} - \Delta s_n.$$

et

$$S_n = (s_n, s_{n+1}, \ldots, s_{n+p-1})$$

de plus $\Delta s_k^{(n)} = s_k^{(n+1)} - s_k^{(n)}$. si $s_k^{(n)} \in \mathbb{C}^p$

## b) Définitions

Soient $\{u_n\}$ et $\{v_n\}$ deux suites de nombres complexes qui tendent vers zéro lorsque $n$ tend :

1. Si $\exists N$ et $C > 0$ : $\forall n > N$ on a $|v_n| < C|u_n|$,
   alors on écrit $v_n = O(u_n)$

2. si $\forall \varepsilon > 0$ $\exists N$ : $\forall n > N$ on a $|v_n| < \varepsilon |u_n|$,
   alors on écrit $v_n = o(u_n)$.

## 2 - Rappels

Commençons par énoncer les principales propriétés du procédé $\Delta^2$- d'Aitkem.

Le procédé $\Delta^2$ d'Aitken consiste à transformer une suite $\{s_n\}$ en une autre suite $\{s'_n\}$ donnée par :

$$s'_n = s_n - \frac{(\Delta s_n)^2}{\Delta^2 s_n} \qquad s_n \in \mathbb{C}.$$

En ce qui concerne le noyau, nous avons le résultat suivant :

> **théorème 1 [9]** Supposons que $\forall n \quad \Delta s_n \neq 0$ et $\Delta^2 s_n \neq 0$
> Une condition nécessaire et suffisante pour que $s'_n = s \ \forall n > N$ est que la suite $\{s_n\}$ vérifie :
>
> $$a_0(s_n - s) + a_1(s_{n+1} - s) = 0 \qquad \forall n > N$$
>
> avec $\quad a_0 + a_1 \neq 0$

La démonstration de ce théorème est basée sur le fait que $s'_n$ peut s'écrire sous la forme d'un rapport de deux déterminants :

$$s'_n = \begin{vmatrix} s_n & s_{n+1} \\ \Delta s_n & \Delta s_{n+1} \end{vmatrix} \Big/ \begin{vmatrix} 1 & 1 \\ \Delta s_n & \Delta s_{n+1} \end{vmatrix}.$$

Comme résultat d'accélération de la convergence, nous allons donner un théorème qui est dû à Henrici [42].

théorème 2 [42]

Si on applique le procédé $\Delta^2$ d'Aitken à une suite $\{s_n\}$ qui converge vers $s$ et si :

$$s_{n+1} - s = (b + b_n)(s_n - s) \quad \text{où } b \in \mathbb{C}, \text{ est une constante telle que}$$

$$b \neq 1 \text{ et } \lim_n b_n = 0$$

alors 1) la suite $\{s'_n\}$ est définie pour $n$ assez grand

2) $s'_n - s = o(s_n - s)$

## 3) Définition de la transformation d'Henrici.

Considérons maintenant une suite de vecteurs complexes $\{s_n\}$.
Pour accélérer la convergence de telle suite Henrici a proposé la transformation suivante ( [42] p116 formule 5-35)

$$h : \{s_n\} \longrightarrow \{h_n\}$$

avec

$$h_n = s_n - \Delta S_n \cdot \Delta^2 S_n^{-1} \Delta s_n$$

où

$\Delta S_n \in L(\mathbb{C}^p)$ et $\Delta S_n$ est la matrice qui a pour colonnes les vecteurs $\Delta s_n, \ldots, \Delta s_{n+p-1}$. $\Delta^2 S_n$ est la matrice dont les colonnes sont $\Delta^2 s_n, \ldots, \Delta^2 s_{n+p-1}$.

Quand $p = 1$, on a $h_n = s'_n$. Cette transformation apparait donc comme une généralisation du procédé $\Delta^2$ d'Aitken au cas vectoriel.

4) Etude de la transformation d'Henrici :

Montrons que $h_n$ peut aussi s'écrire sous forme d'un rapport de deux déterminants.

Au préalable, rappelons l'identité de Magnus[17] (qui est en fait une généralisation de celle de Schur [20]).

Soit $\mathring{x} \in \mathbb{C}^p$, $u \in \mathbb{C}^p$ et $X \in L(\mathbb{C}^p)$ où $X = (x^1, \ldots, x^p)$

$$\frac{\det \begin{bmatrix} \mathring{x} & X \\ u & A \end{bmatrix}}{\det(A)} = \mathring{x} - X A^{-1} u$$

Le déterminant du numérateur est développable suivant la première ligne.

Prenons $\mathring{x} = s_m$, $X = \Delta S_m$, $A = \Delta^2 S_m$ et $u = \Delta s_m$, alors:

$$h_m = \begin{vmatrix} s_m & \Delta s_m & \ldots & \Delta s_{n+p-1} \\ (\Delta s_m)_1 & (\Delta^2 s_m)_1 & \ldots & (\Delta^2 s_{m+p-1})_1 \\ \ldots & \ldots & \ldots & \ldots \\ (\Delta s_m)_p & (\Delta^2 s_m)_p & \ldots & (\Delta^2 s_{m+p-1})_p \end{vmatrix} \Big/ \det(\Delta^2 S_n).$$

qui peut aussi s'écrire, en remplacant chaque colonne par sa somme avec la précédente :

$$h_n = \begin{vmatrix} s_n & s_{n+1} & \cdots & s_{n+p} \\ (\Delta s_n)_1 & (\Delta s_{n+1})_1 & \cdots & (\Delta s_{n+p})_1 \\ \cdots & \cdots & \cdots & \cdots \\ (\Delta s_n)_p & (\Delta^2 s_n)_p & \cdots & (\Delta s_{n+p})_p \end{vmatrix} \Bigg/ \begin{vmatrix} 1 & 1 & \cdots & 1 \\ (\Delta s_n)_1 & (\Delta s_{n+1})_1 & \cdots & (\Delta s_{n+p})_1 \\ \cdots & \cdots & \cdots & \cdots \\ (\Delta s_n)_p & (\Delta s_{n+1})_p & \cdots & (\Delta s_{n+p})_p \end{vmatrix}.$$

Cette relation nous permet d'énoncer le théorème suivant :

<u>théorème 3</u> supposons que $\det(\Delta S_n) \neq 0$ et $\det(\Delta^2 S_n) \neq 0$ $\forall n$

Une condition nécessaire et suffisante pour que $h_n = s$ $\forall n > N$ est que la suite $\{s_n\}$ vérifie :

$$\sum_{i=0}^{p} a_i (s_{n+i} - s) = 0 \qquad \forall n > N$$

avec

$$a_i \in \mathbb{C} \text{ et } \sum_{i=0}^{p} a_i = 1$$

preuve : condition suffisante :

On a le système d'equations linéaires

$$\begin{cases} a_0 + a_1 + \cdots + a_p = 1 \\ a_0 s_n + \cdots + a_p s_{n+p} = s \\ a_0 s_{n+1} + \cdots + a_p s_{n+p+1} = s \end{cases}$$

Ce système peut s'écrire :

$$\begin{cases} a_0 + a_1 + \cdots + a_p = 1 & (1) \\ a_0 \Delta s_n + \cdots + a_p \Delta s_{n+p} = 0 & (2) \\ a_0 s_n + \cdots + a_p s_{n+p} = s & (3) \end{cases}$$

le déterminant principal du système d'equations linéaires (1), (2) est le déterminant de la matrice $\Delta^2 S_n$ qui est non nul par hypothèse, donc on peut déterminer $a_0, a_1, \ldots, a_p$, que l'on reporte dans (3) et on obtient : $s = t_n \quad \forall n > N$

## Condition nécessaire :

On a vu précédemment que $t_n$ pouvait être mis sous la forme d'un rapport de deux déterminants,

donc $t_n = s \quad \forall n > N$ devient :

$$(4) \qquad \begin{vmatrix} s_n - s & s_{n+1} - s & \dots & s_{n+p} - s \\ (\Delta s_n)_1 & (\Delta s_{n+1})_1 & \dots & (\Delta s_{n+p})_1 \\ \dots & \dots & \dots & \dots \\ (\Delta s_n)_p & (\Delta s_{n+1})_p & \dots & (\Delta s_{n+p})_p \end{vmatrix} = 0 \qquad \forall n > N.$$

Développons le déterminant, on a alors :

$$(5) \qquad x_0^{(n)} (s_n - s) + x_1^{(n)} (s_{n+1} - s) + \dots + x_p^{(n)} (s_{n+p} - s) = 0 .$$

d'un autre côté :

$$\sum_{i=0}^{p} x_i^{(n)} = \det(\Delta^2 S_n) \neq 0 .$$

Posons alors $\quad a_i^{(n)} = \dfrac{x_i^{(n)}}{\sum_{i=0}^{p} x_i^{(n)}}$ .

La relation (5) devient :
$$\begin{cases} \sum_{i=0}^{p} a_i^{(n)} (s_{n+i} - s) = 0 \\ \text{et } \sum_{i=0}^{p} a_i^{(n)} = 1 \end{cases} \qquad (6) .$$

Comme par ailleurs, la relation (4) peut s'écrire

$$\begin{vmatrix} s_{n+1} - s & s_{n+2} - s & \dots & s_{n+p+1} - s \\ (\Delta s_n)_1 & (\Delta s_{n+1})_1 & \dots & (\Delta s_{n+p})_1 \\ \dots & \dots & \dots & \dots \\ (\Delta s_n)_p & (\Delta s_{n+1})_p & \dots & (\Delta s_{n+p})_p \end{vmatrix} = 0 \qquad \forall n > N.$$

En utilisant le même raisonnement que précédemment on obtient :

$$\begin{cases} \sum_{i=0}^{p} a_i^{(n)} \left( s_{n+i+1} - s \right) = 0 \\ \text{et} \quad \sum_{i=0}^{p} a_i^{(n)} = 1 \end{cases} \qquad (7)$$

En combinant les relations (5) et (6) on obtient :

$$\begin{cases} \sum_{i=0}^{p} \Delta a_i^{(n)} = 0 \\ \text{et} \\ \sum_{i=0}^{p} \Delta a_i^{(n)} \left( s_{n+i+1} - s \right) = 0 \end{cases}$$

Le déterminant de ce système d'équations homogènes n'est autre que le déterminant de la matrice $\Delta S_{n+1}$ qui est supposé par hypothèse non nul, et par conséquent :

$$\Delta a_i^{(n)} = 0 \qquad i = 0, \ldots, p$$

$$\text{d'où} \qquad a_i^{(n)} = a_i \qquad \blacksquare$$

Une conséquence immédiate de ce théorème est le :

**Théorème 4** Supposons que $\det(\Delta S_n) \neq 0$ et $\det(\Delta^2 S_n) \neq 0$ $\forall n$.
Une condition nécessaire et suffisante pour que $t_n = s$ $\forall n > N$ est que :

$$s_n = s + \sum_{i=1}^{j} \tilde{a}_i(n)\, r_i^n + \sum_{i=j+1}^{q} \left[ \tilde{b}_i(n) \cos b_i n + \tilde{c}_i(n) \sin b_i n \right] e^{w_i n} + \sum_{i=0}^{m} x_i\, \delta_{in} \qquad \forall n > N$$

$r_i$, $w_i$ et $b_i$ appartiennent à $\mathbb{C}$ et l'on a $r_i \neq 1$ pour $i = 1, \ldots, j$.
$\tilde{a}_i$, $\tilde{b}_i$ et $\tilde{c}_i$ sont des polynomes en $n$ dont les coefficients appartiennent à $\mathbb{C}^p$. les $x_i$ appartiennent à $\mathbb{C}^p$ et $\delta_{in}$ est le symbole de Kronecker.
Si $d_i$ désigne le degré de $\tilde{a}_i$ plus un pour $i = 1, \ldots, j$ et le plus grand des degrés des $\tilde{b}_i$ et de $\tilde{c}_i$ pour $i = j+1, \ldots, q$, on doit avoir,

$$m + \sum_{i=1}^{j} d_i + 2 \sum_{i=j+1}^{q} d_i = p - 1,$$ avec la convention que $m = -1$ s'il n'y a aucun terme en $\delta_{in}$

preuve:

L'equation aux différences peut être résolue dans $\mathbb{C}^p$ de la même façon que lorsque $s_n \in \mathbb{C}$, voir [9]. ■

Propriété

> Si l'application de la transformation d'Henrici aux suites $\{s_n\}$ et $\{A s_n + b\}$ avec $A \in L(\mathbb{C}^p)$, A inversible et $b \in \mathbb{C}^p$, fournit respectivement les éléments $h_n$ et $\tilde{h}_n$ alors on a:
>
> $$\tilde{h}_n = A h_n + b$$

preuve:

$$\begin{aligned} \tilde{h}_n &= A s_n + b - [A \Delta S_n] [A \Delta^2 S_n]^{-1} A \Delta s_n \\ &= A \{ s_n - \Delta S_n \Delta^2 S_n^{-1} \Delta s_n \} + b \\ &= A h_n + b \end{aligned}$$

■

Donnons un théorème d'accélération de la convergence, qui généralise le théorème 2, établi pour le $\Delta^2$ d'Aitken.

Nous avons auparavant besoin des résultats suivants:

Lemme 1

> Soit A une matrice, $A \in L(\mathbb{C}^p)$ $p \geq 2$
> alors:
>
> $$\|A^{-1}\|_2 \leq \frac{\|A\|_F^{p-1}}{|\det(A)| \, (p-1)^{(p-1)/2}}$$

preuve: C'est une conséquence immédiate du résultat (23.16) p158 de [61]:

si $\lambda(A)$ est la plus petite valeur propre de $A^*A$.

alors $\quad \lambda(A) > (p-1)^{(p-1)/2} \, |\det(A)| \, / \, \|A\|_F^{(p-1)}$ $\quad *$

or $\quad \|A^{-1}\|_2 = \dfrac{1}{\lambda(A)}$ voir par exemple [51] p170.

d'où le résultat cherché. ■

Remarques :

a) Le résultat $*$ est dû à U. Wegner [89].

b) Il existe d'autres majorations de la norme de $A^{-1}$, en particulier un résultat qui est dû à Kato [48] :

$$\|A^{-1}\| \leq \frac{\|A\|^{p-1}}{|\det(A)|}$$

où $\| \ \|$ est une norme induite par une norme vectorielle.

Si l'on prend la norme euclidienne dans $\mathbb{C}^p$, cette égalité devient :

$$\|A^{-1}\|_2 \leq \frac{\|A\|_2^{p-1}}{|\det(A)|} \leq \frac{\|A\|_F^{p-1}}{|\det(A)|} \qquad **.$$

La majoration donnée dans le lemme 1 est plus précise que la majoration $**$.

Rappelons aussi le lemme de perturbation [60] p45.

Lemme 2

Soit $A$ et $C$ deux matrices carrées d'ordre $p$, supposons que $A$ soit inversible et que $\|A^{-1}\| \leq \alpha$.

Si $\|A-C\| \leq \beta$ et que $\alpha\beta < 1$ alors :

1) $C$ est inversible

2) $\|C^{-1}\| \leq \dfrac{\alpha}{1-\alpha\beta}$

Donnons maintenant la définition d'une matrice uniformement inversible [30] p369.

Définition 1

Soit : $c \in \mathbb{R}$ et $c > 0$

Notons $K(c)$ l'ensemble des matrices, défini par :

$$K(c) = \{ A = (a^1, \ldots, a^p) \in L(\mathbb{C}^p) \mid a^i \neq 0,\ i = 1, \ldots, p \text{ et } \left| \det \left( \frac{a^1}{\|a^1\|_2}, \ldots, \frac{a^p}{\|a^p\|_2} \right) \right| \geq c \}.$$

les matrices $B_n$ de $L(\mathbb{C}^p)$ sont dites uniformément inversibles si et seulement si :

$\exists\, d > 0, \exists N$ ; tel que $B_n \in K(d)$ $\forall n > N$

faisons quelques remarques concernant cette définition.

Remarques :

1) si $B_n$ est uniformément inversible alors $B_n$ est inversible.

2) $K(c)$ est vide si $c > 1$ en effet :

D'après l'identité de Hadamard [31] p 256.

$$|\det(a^1, a^2, \ldots, a^p)|^2 \leq \prod_{i=1}^{p} \|a^i\|_2^2$$

donc

$$\left| \det \left( \frac{a^1}{\|a^1\|_2}, \ldots, \frac{a^p}{\|a^p\|_2} \right) \right| \leq 1 .$$

3) si $D$ est une matrice diagonale inversible alors $D \in K(1)$.

Lemme 3

Soit $\{B_n\}$ une suite de matrices carrées, telles que $\lim_n B_n = 0$

et

soit $\{U_n\}$ une autre suite de matrices telles que $U_n \in L(\mathbb{C}^p)$ et

$$\exists d > 0 \;, \exists N, \quad \forall n > N \qquad U_n \in K(d) \text{ et}$$
$$U_n = (u_n, \dots, u_{n+p-1}) .$$

Si on pose $\quad \tilde{B}_n = (B_n u_n, \dots, B_{n+p-1} u_{n+p-1}) \, U_n^{-1}$

alors :

$$\lim_n \tilde{B}_n = 0$$

preuve: Par définition

$$\tilde{B}_n = (B_n u_n, \dots, B_{n+p-1} u_{n+p-1}) \, U_n^{-1}$$

ce qui peut encore s'écrire

$$\tilde{B}_n = \left(B_n \frac{u_n}{\|u_n\|_2}, \dots, B_{n+p-1} \frac{u_{n+p-1}}{\|u_{n+p-1}\|_2}\right) V_n^{-1}$$

avec

$$V_n = \left(\frac{u_n}{\|u_n\|_2}, \dots, \frac{u_{n+p-1}}{\|u_{n+p-1}\|_2}\right) .$$

Posons $\quad x_n = \left\| \left(B_n \frac{u_n}{\|u_n\|_2}, \dots, B_{n+p-1} \frac{u_{n+p-1}}{\|u_{n+p-1}\|_2}\right) \right\|_2$

et remarquons que si $A \in L(\mathbb{C}^p)$ et $A = (a^1, a^2, \dots, a^p)$ on a :

$\|A\|_2 \leq \|A\|_F$ et $\|A_F\|^2 = \sum_{i=1}^{p} \|a^i\|_2^2$,

il vient donc que :

$$x_n \leq \left(\sum_{i=0}^{p-1} \left\|B_{n+i} \frac{u_{n+i}}{\|u_{n+i}\|_2}\right\|_2\right)^{1/2}$$

d'où

$$x_n \leq \left(\sum_{i=0}^{p-1} \|B_{n+i}\|_2^2\right)^{1/2}$$

et donc $\quad \lim_n x_n = 0 .$

D'autre part d'après le lemme 1,

$$\|V_n^{-1}\|_2 \leq \frac{\|V_n\|_F^{p-1}}{|\det(V_n)|\ (p-1)^{(p-1)/2}} .$$

L'inégalité précédente s'écrit :

$$\|V_n^{-1}\|_2 \leq \frac{\left\|\left(\frac{u_n}{\|u_n\|_2}, \dots, \frac{u_{n+p-1}}{\|u_{n+p-1}\|_2}\right)\right\|_F^{p-1}}{|\det(V_n)| \times (p-1)^{(p-1)/2}} ,$$

or par hypothèse $|\det(V_n)| > d \quad , \forall n > N$

et de plus $\|V_n\|_F = \sqrt{p}$

donc

$$\|V_n^{-1}\| \leq \frac{p^{(p-1)/2}}{d \times (p-1)^{(p-1)/2}}$$

et comme

$$\|\mathcal{B}_n\|_2 \leq \alpha_n \times \|V_n^{-1}\|_2$$

on a $\lim_n \|\mathcal{B}_n\| = 0$

d'où le résultat cherché. ■

Donnons un théorème qui permet de déduire le comportement de la suite $\{\Delta s_n\}$ à partir du comportement de la suite $\{s_n - s\}$

théorème 5

Soit $\{s_n\}$ une suite de vecteurs complexes tels que

$s_{n+1} - s = (B + B_n)(s_n - s)$ où $B \in L(\mathbb{C}^p)$, $B_n \in L(\mathbb{C}^p)$

$\lim_n B_n = 0$ et la matrice $(B - I)$ inversible.

alors :

$\exists N, \forall n > N \qquad \Delta s_{n+1} = B\,\Delta s_n + C_n\,\Delta s_n$

où $\lim_n C_n = 0$

preuve : Posons $e_n = s_n - s$.

Par hypothèse $e_{n+1} = (B+B_n)\, e_n$.

Donc

$$\Delta e_n = \Delta s_n = \{(B-I)+B_n\}\, e_n$$
$$= (I+F_n)\,(B-I)\, e_n$$

avec $F_n = B_n\,(B-I)^{-1}$

On a $\lim_n F_n = 0$, d'après le lemme 2 avec $A = I$ et $C = I+F_n$.

la matrice $C$ est inversible, $\forall\, n > N$

d'où $e_n = (B-I)^{-1}\,(I+F_n)^{-1}\,\Delta s_n$.

D'autre part

$$\Delta e_{n+1} = \{(B-I)+B_{n+1}\}\, e_{n+1}$$
$$= \{(B-I)+B_{n+1})\}\,(B+B_n)\,(B-I)^{-1}(I+F_n)^{-1}\,\Delta s_n.$$

Posons $G_n = I-(I+F_n)^{-1}$

D'après le lemme 2 $\|G_n\|_2 \leq \dfrac{\|F_n\|_2}{1-\|F_n\|_2}$

et donc $\lim_n G_n = 0$.

En utilisant ce qui précède on a :

$$\Delta s_{n+1} = \{(B-I)+B_{n+1}\}\,(B+B_n)\,(B-I)^{-1}\,(I-G_n)\,\Delta s_n$$
$$= \{(B-I)\,B\,(B-I)^{-1} + C_n\}\,\Delta s_n \qquad \text{où } \lim C_n = 0$$
$$= (B+C_n)\,\Delta s_n$$

car $(B-I)B = B(B-I)$

■

Nous sommes maintenant en mesure de démontrer le théorème fondamental suivant

<u>théorème 6</u> :

> Soit $\{s_n\}$ une suite de vecteurs complexes tels que $s_{n+1}-s=(B+B_n)(s_n-s)$, où $B$ et $B_n$ sont des matrices carrées telles que le rayon spectral de $B$ soit strictement inférieur à un et que $\lim_n B_n = 0$.
>
> si de plus $\exists\ d>0,\ \exists N,\ \forall n>N \quad \Delta S_n \in K(d)$
>
> alors a) la suite $\{t_n\}$ est définie pour n assez grand.
>
> b) $\|t_n - s\| = o\left(\|s_n - s\|\right)$

preuve: D'après le théorème 5, on sait que

$\exists N',\ \forall n>N' \quad \Delta s_{n+1} = B\,\Delta s_n + C_n\,\Delta s_n \qquad$ où $\lim_n C_n = 0$.

Donc $\quad \Delta^2 s_n = (B-I)\,\Delta s_n + C_n\,\Delta s_n$

D'où :

$$\Delta^2 s_{n+i} = (B-I)\,\Delta s_{n+i} + C_{n+i}\,\Delta s_{n+i} \quad \text{pour } i=0,\dots,p-1.$$

Ecrivons ces relations sous une forme matricielle ; on a:

$$\Delta^2 S_n = (B-I)\,\Delta S_n + (C_n\,\Delta s_n, \dots\dots, C_{n+p-1}\,\Delta s_{n+p-1}).$$

Cette égalité peut s'écrire :

$$\Delta^2 S_n = (B-I)(I+X_n)\,\Delta S_n$$

avec

$$X_n = (B-I)^{-1}\,\mathcal{C}_n = (B-I)^{-1}(C_n\,\Delta s_n, \dots, C_{n+p-1}\,\Delta s_{n+p-1}),$$

d'où en utilisant le lemme 3, $\lim_n X_n = 0$.

D'après le lemme 2 $\quad I+X_n$ est inversible $\forall n>N''$

donc $\Delta^2 S_m$ est inversible $\forall m > \max(N', N'', N) = N_1$

et par conséquent la suite $\{h_m\}$ est bien définie $\forall n > N_1$.

Démontrons maintenant la seconde partie du théorème.

Posons $(I + X_m)^{-1} = I - D_m$ , on a donc $\lim_n D_n = 0$.

Donc

$$\Delta^2 S_m^{-1} = \Delta S_m^{-1} (I - D_m) (B - I)^{-1} \qquad \forall n > N_1 .$$

Comme par définition

$$h_m = s_m - \Delta S_m \Delta^2 S_m^{-1} \Delta s_m ,$$

en utilisant ces deux égalités on a:

$$h_m - s = s_m - s - (I - D_m)(B - I)^{-1} \Delta s_m \quad . \quad **$$

Comme par ailleurs $e_{n+1} = (B + B_m) e_m$

donc $\Delta e_n = \{(B - I) + B_m\} e_n$ .

En reportant cette relation dans ** on obtient

$$\begin{aligned} h_m - s &= s_m - s - (I - D_n)(B - I)^{-1}\{(B - I) + B_m\} e_n \\ &= \left[I - (I - D_m)(B - I)^{-1}\{(B - I) + B_m\}\right] e_n \\ &= G_m e_m \qquad \text{où} \quad \lim_n G_m = 0 \end{aligned}$$

$$\| h_n - s \| \leq \| G_m \| \; \| e_m \|$$

On a donc $\| h_m - s \| = o(\| s_m - s \|)$

■

Nous allons maintenant examiner l'implémentation de la transformation d'Henrici.

## 5- Implémentation de la transformation d'Henrici :

Nous allons voir deux mises en oeuvres de la transformation d'Henrici, l'une basée sur le H-algorithme l'autre basée sur l'orthonormalisation de Gram-Schmidt.

### a) Le H-algorithme

Cet algorithme a été introduit dans [15].

C'est un algorithme récursif qui permet le calcul des termes $H_k^{(n)}$ définis par des rapports de deux déterminants :

$$H_k^{(n)} = \begin{vmatrix} s_n & s_{n+1} & \cdots & s_{n+k} \\ g_1(n) & \cdots & \cdots & g_1(n+k) \\ \cdots & \cdots & \cdots & \cdots \\ g_k(n) & \cdots & \cdots & g_k(n+k) \end{vmatrix} \Bigg/ \begin{vmatrix} 1 & \cdots & \cdots & 1 \\ g_1(n) & \cdots & \cdots & g_1(n+k) \\ \cdots & \cdots & \cdots & \cdots \\ g_k(n) & \cdots & \cdots & g_k(n+k) \end{vmatrix}$$

avec $g_i(n)$ des nombres complexes
et $s_n$ des vecteurs de $\mathbb{C}^p$.

Posons $H_k^{(n)} = \dfrac{N_k^{(n)}}{D_k^{(n)}}$ et $g_{k,i}^{(n)} = \dfrac{N_{k,i}^{(n)}}{D_k^{(n)}}$ où

$$N_{k,i}^{(n)} = \begin{vmatrix} g_i(n) & \cdots & \cdots & g_i(n+k) \\ g_1(n) & \cdots & \cdots & g_1(n+k) \\ \cdots & \cdots & \cdots & \cdots \\ g_k(n) & \cdots & \cdots & g_k(n+k) \end{vmatrix} \qquad \text{pour } i > k$$

Si on applique pour $N_{k,i}^{(n)}$ et $D_k^{(n)}$ l'identité de Sylvester, et pour $N_k^{(n)}$ l'identité de Sylvester généralisée [17], on obtient l'algorithme suivant :

$$\left[\begin{array}{lll}
H_0^{(n)} = s_n \quad \text{et} \quad g_{0,i}^{(n)} = g_i(n) & & i=1,\ldots \\[2ex]
H_k^{(n)} = \dfrac{g_{k-1,k}^{(n+1)}\, H_{k-1}^{(n)} - g_{k-1,k}^{(n)}\, H_{k-1}^{(n+1)}}{g_{k-1,k}^{(n+1)} - g_{k-1,k}^{(n)}} & & \begin{array}{l} k=1,\ldots \\ n=0,\ldots \end{array} \\[3ex]
g_{k,i}^{(n)} = \dfrac{g_{k-1,k}^{(n)}\, g_{k-1,i}^{(n)} - g_{k-1,k}^{(n)}\, g_{k-1,i}^{(n+1)}}{g_{k-1,k}^{(n+1)} - g_{k-1,k}^{(n)}} & & \text{pour } i > k
\end{array}\right.$$

On a le résultat suivant :

$$\boxed{\begin{array}{l} \text{si} \quad g_i(n) = (\Delta s_n)_i \qquad i=1,\ldots p \\ \text{alors} \qquad t_n = H_p^{(n)} \end{array}}$$

<u>Remarques</u>: ① En cas de singularité, on peut utiliser la relation suivante [20] :

$$H_{k+m}^{(n)} = \begin{vmatrix} H_k^{(n)} & H_k^{(n+1)} & \cdots & H_k^{(n+m)} \\ g_{k,k+1}^{(n)} & g_{k,k+1}^{(n+1)} & \cdots & g_{k,k+1}^{(n+m)} \\ \cdots & \cdots & \cdots & \cdots \\ g_{k,k+m}^{(n)} & g_{k,k+m}^{(n+1)} & \cdots & g_{k,k+m}^{(n+m)} \end{vmatrix} \Bigg/ \begin{vmatrix} 1 & \cdots 1 & \cdots & 1 \\ g_{k,k+1}^{(n)} & & \cdots & g_{k,k+1}^{(n+m)} \\ \cdots & \cdots & \cdots & \cdots \\ g_{k,k+m}^{(n)} & & \cdots & g_{k,k+m}^{(n+m)} \end{vmatrix}$$

② Cet algorithme peut aussi être utilisé pour le calcul de $\varepsilon_{2k}^{(n)}$ qui est le rapport de deux déterminants dans l'ε-algorithme topologique, ainsi que les approximants de Padé vectoriels récemment définis dans [86].

③ dans le cas réel, avec le H-algorithme le calcul de $h_0$ nécessite de l'ordre de $\frac{5}{2}p^3$ opérations arithmétiques ; puis connaissant $h_n$ le calcul de $h_{n+1}$ se fait en $O(p^2)$ opérations arithmétiques.

Nous allons dans ce qui suit proposer une autre mise en oeuvre basée sur l'orthonormalisation de Gram-Schmidt [25].

b) L'orthonormalisation de Gram-Schmidt

Plaçons nous dans le cas réel, et posons $A_m = \Delta^2 S_m$. Nous allons montrer comment obtenir la factorisation de $A_{m+1}$ en fonction de celle de $A_m$ et pour cela nous allons utiliser les techniques et les méthodes élaborées dans [25] et [36].

Supposons que la factorisation de $A_0$ est connue.

- Mise à jour de la factorisation de $A_m = Q_n \times R_n$

Supposons que l'on dispose de la factorisation de $A_m$ c'est à dire :

$$A_m = Q_n \times R_n \qquad \text{avec } Q_m \in L(\mathbb{R}^p)$$
$$R_m \in L(\mathbb{R}^p)$$

$Q_n$ : matrice orthogonale $({}^TQ_m \cdot Q_n = I)$

$R_n$ : matrice triangulaire supérieure.

Ecrivons $A_n$ sous la forme : $A_n = (a_n^1, a_n^2, \dots, a_n^p)$
et donc

$$A_{n+1} = (a_{n+1}^1, \dots, a_{n+1}^p) = (a_n^2, \dots, a_n^{p-1}, v_n)$$

où

$$v_n = \Delta^2 s_{n+p} .$$

La matrice $A_{n+1}$ peut être obtenue à partir de $A_n$ de la manière suivante :

On supprime la première colonne de $A_n$, puis on translate ses colonnes vers la gauche (on met $a_n^i$ dans la $(i-1)$ ème colonne, pour $i=2, \dots, p$) et enfin $v_n$ est inseré dans la dernière colonne.

A partir de cette remarque montrons comment $Q_{n+1}$ et $R_{n+1}$ seront obtenus.

On a $\quad A_{n+1} = Q_{n+1} \cdot R_{n+1}$

posons

$$A_{n+1} = (\tilde{A}_n, v_n) \quad \text{avec } \tilde{A}_n \in L(\mathbb{R}^p, \mathbb{R}^{p-1})$$

$$\text{et } A_n = (a_n^1, \tilde{A}_n)$$

<u>Suppression de la première colonne.</u>

Ecrivons la matrice $R_n$ sous la forme suivante :

$$R_n = \begin{pmatrix} r_{11}^{(n)} & r_{12}^{(n)} & \dots & r_{1p}^{(n)} \\ & r_{22}^{(n)} & & \vdots \\ & & \ddots & \vdots \\ & 0 & & r_{pp}^{(n)} \end{pmatrix} = (P_n^1, P_n^2, \dots, P_n^p) .$$

On a $\quad P_n^1 = r_{11}^{(n)} e^1$

On sait que $\quad A_n = Q_n R_n = Q_n (\rho_n^1, \rho_n^2, \dots, \rho_n^p)$

$$= (r_{11}^{(n)} Q_n e^1, Q_n \tilde{R}_n) .$$

Posons $\quad B_n = Q_n \tilde{R}_n$ .

$\tilde{R}_n$ a la structure suivante :

$$\tilde{R}_n = \begin{pmatrix} r_{12}^{(n)} & \cdots & \cdots & r_{1p}^{(n)} \\ r_{22}^{(n)} & \cdots & \cdots & r_{2p}^{(n)} \\ & \ddots & & \vdots \\ 0 & & \ddots & \vdots \\ & & & r_{pp}^{(n)} \end{pmatrix}$$

Pour transformer $\tilde{R}_n$ en une matrice triangulaire supérieure, nous allons la multiplier par $(p-1)$ matrices de Givens [25] et [36].

Notons

$$G_{i,j} = \begin{array}{c} \\ i \\ \\ j \\ \\ \end{array} \begin{pmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & c & 1 & \cdots & s \\ & & & & \ddots & & \\ & s & & & 1 & -c & 1 \\ & & & & & \ddots & \\ & & & & & & 1 \end{pmatrix} .$$

$$\qquad\qquad\qquad\qquad i \qquad\qquad\qquad j$$

Remarquons que $G_{i,j}$ est symétrique .

Les nombres $c$ et $s$ sont choisis pour que :

$$\begin{pmatrix} c & s \\ s & -c \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \begin{pmatrix} +\rho \\ 0 \end{pmatrix}$$

avec $\rho^2 = z_1^2 + z_2^2$

$\rho = \text{signe}(z_1)\ |\rho|$

$c = \frac{z_1}{\rho}$ et $s = \frac{z_2}{\rho}$ .

Or $G_{p-1,p} \ldots G_{1,2} \tilde{R}_n = \begin{pmatrix} \hat{R}_n \\ {}^t 0 \end{pmatrix}$

$\hat{R}_n$ est une matrice triangulaire supérieure, et de la même façon

$$Q_n G_{1,2} \ldots G_{p-1,p} = (\hat{Q}_n , \tilde{q}_n) \quad \text{ou} \quad {}^T\hat{Q}_n \hat{Q}_n = I_{(p-1)}$$

et donc

$$\tilde{A}_n = \hat{Q}_n \hat{R}_n$$

<u>Adjonction de la colonne $p$.</u>

On a :

$$A_{n+1} = (\tilde{A}_n , v_n) = (\hat{Q}_n \hat{R}_n , v_n)$$

Commençons par chercher un vecteur $\tilde{q}_n$ tel que :

$$\begin{cases} - q_n \text{ orthogonal à } \hat{Q}_n & \text{i.e } {}^T\hat{Q}_n \tilde{q}_n = 0 \\ \|q_n\|_2 = 1 \end{cases}$$

$$(\hat{Q}_n \hat{R}_n , v_n) = (\hat{Q}_n , \tilde{q}_n) \left(\begin{array}{c|c} \hat{R}_n & {}^T\hat{Q}_n v_n \\ \hline 0 & d_n \end{array}\right)$$

$$v_n = \hat{Q}_n {}^T\hat{Q}_n v_n + d_n \tilde{q}_n \qquad , v_n \in \mathbb{R}^{p-1} \text{ et } d_n \in \mathbb{R} .$$

En multipliant cette égalité par ${}^T\hat{Q}_n$ on obtient

$$d_n \tilde{q}_n = (I_{p-1} - \hat{Q}_n {}^T\hat{Q}_n) v_n = v'_n .$$

Posons :

$$d_n = \|v'_n\|_2 \qquad \text{et} \quad \tilde{q}_n = \frac{v'_n}{d_n}$$

$$Q_{n+1} = (\hat{Q}_n , \tilde{q}_n)$$

et

$$R_{n+1} = \left(\begin{array}{c|c} \hat{R}_n & {}^t\hat{Q}_n v_n \\ \hline 0 & \|v'_n\|_2 \end{array}\right)$$

Ainsi la factorisation de $A_{n+1}$ est obtenue.

Montrons comment utiliser cette factorisation pour calculer $h_n$ :

$$h_n = s_n - \Delta S_n \, \Delta^2 S_n^{-1} \, \Delta s_n .$$

a) On réalise la factorisation de $\Delta^2 S_n$.

$$\Delta^2 S_n = Q_n \times R_n$$

b) On résoud le système d'équations linéaires :

$$R_n \, x = {}^T Q_n \, \Delta s_n$$

c) puis on calcule $h_n$ par la formule :

$$h_n = s_n - \Delta S_n \, x .$$

Remarques

① Le nombre d'opérations arithmétiques pour calculer $h_0$ est un $O(p^3)$

② connaissant $h_n$, le calcul de $h_{n+1}$ se fait en $O(p^2)$ opérations arithmétiques

Le coût est donc comparable à celui obtenu avec le H-algorithme.

Donnons quelques exemples numériques :

Commençons par deux exemples où les deux techniques de programmation ont donné les mêmes résultats.

**Exemple n°1** Soit la suite $\{s_n\}$ définie par :

$$s_{n+1} = G(s_n) \qquad s_0 = \begin{pmatrix} 0.6 \\ 3 \end{pmatrix}$$

Notons $s_n = \begin{pmatrix} x_n \\ y_n \end{pmatrix}$ et $G = \begin{pmatrix} G_1 \\ G_2 \end{pmatrix}$, $h_n = \begin{pmatrix} u_n \\ v_n \end{pmatrix}$

$$\begin{cases} G_1(x,y) = \sin(xy) - \dfrac{y}{2\pi} \\ G_2(x,y) = 2\pi x - \left(\pi - \dfrac{1}{4}\right)\left(e^{2x-1} - 1\right) \end{cases}$$

$\lim_n s_n = (0.5, \pi)^T$.

| $n$ | $x_{n+3}$ | $y_{n+3}$ | $u_n$ | $v_n$ |
|---|---|---|---|---|
| 0 | 0.500289598729D+00 | 0.314244656856D+01 | 0.500274538405D+00 | 0.314284030518D+01 |
| 1 | 0.499863201420D+00 | 0.314173696784D+01 | 0.499990373064D+00 | 0.314158228248D+01 |
| 2 | 0.499976967725D+00 | 0.314152414608D+01 | 0.499994595440D+00 | 0.314159101125D+01 |
| 3 | 0.500010897625D+00 | 0.314158113438D+01 | 0.499999395870D+00 | 0.314159258456D+01 |
| 4 | 0.500001832933D+00 | 0.314159810172D+01 | 0.499999965530D+00 | 0.314159264324D+01 |
| 5 | 0.499999132868D+00 | 0.314159357004D+01 | 0.499999996170D+00 | 0.314159265315D+01 |
| 6 | 0.499999854140D+00 | 0.314159222002D+01 | 0.499999999978D+00 | 0.314159265352D+01 |
| 7 | 0.500000069005D+00 | 0.314159258066D+01 | 0.499999999998D+00 | 0.314159265359D+01 |

**Exemple n°2**

$$s_{n+1} = G(s_n) \qquad s_0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$\begin{cases} G_1(x,y) = -(2x+y)^3 + 0.127 \\ G_2(x,y) = -0.5\,(x+3y)^5 + 0.10512 \end{cases}$$

$\lim_n s_n = (0.1, 0.1)^T$

| $n$ | $x_{n+3}$ | $y_{n+3}$ | $u_n$ | $v_n$ |
|---|---|---|---|---|
| 0 | 0.109822237638D+00 | 0.101622838715D+00 | 0.985160632209D-01 | 0.996949519038D-01 |
| 1 | 0.938411372651D-01 | 0.989881465730D-01 | 0.993018830298D-01 | 0.998744502174D-01 |
| 2 | 0.103441444764D+00 | 0.100562005805D+00 | 0.998099730734D-01 | 0.999671128278D-01 |
| 3 | 0.979395817958D-01 | 0.996633208053D-01 | 0.999151587937D-01 | 0.999854279246D-01 |
| 4 | 0.101185735280D+00 | 0.100193515376D+00 | 0.999807487194D-01 | 0.999967050736D-01 |
| 5 | 0.993015156841D-01 | 0.998859552518D-01 | 0.999832471833D-01 | 0.999971330038D-01 |
| 6 | 0.100405922217D+00 | 0.100066253964D+00 | 0.999985117421D-01 | 0.999997454076D-01 |
| 7 | 0.997622188042D-01 | 0.999611830342D-01 | 0.100001413339D+00 | 0.100000241843D+00 |
| 8 | 0.100138644435D+00 | 0.100022630736D+00 | 0.999999175311D-01 | 0.999999858884D-01 |
| 9 | 0.999189407228D-01 | 0.999867679975D-01 | 0.100000025593D+00 | 0.100000004380D+00 |
| 10 | 0.100047316983D+00 | 0.100007723673D+00 | 0.999999963412D-01 | 0.999999937738D-01 |
| 11 | 0.999723540070D-01 | 0.999954871776D-01 | 0.100000000833D+00 | 0.100000000143D+00 |
| 12 | 0.100016144080D+00 | 0.100002635263D+00 | 0.999999998510D-01 | 0.999999999745D-01 |
| 13 | 0.999905695784D-01 | 0.999984606234D-01 | 0.100000000030D+00 | 0.100000000005D+00 |

<u>Exemple n°3</u> $\qquad s_{n+1} = G(s_n) \qquad s_0 = \begin{pmatrix} 6 \\ 4 \end{pmatrix}$

$$\begin{cases} G_1(x,y) = \dfrac{x_1}{3} - \dfrac{x_2}{6} + \dfrac{1}{12}(x_1-x_2)^2 \\ G_2(x,y) = -\dfrac{x_1}{8} + \dfrac{x_2}{3} + \dfrac{1}{12}(x_1-x_2)^2 \end{cases} \qquad \lim s_n = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

La suite $\{s_n\}$ ne vérifie pas l'une des hypothèses du théorème 8 : les matrices $\Delta S_n$ ne sont pas uniformement inversibles comme nous allons le voir.

Posons $\qquad c_n = |\det(\Delta S_n)| \qquad$ et $\qquad d_n = c_n / (\|\Delta s_n\|_2 \|\Delta s_{n+1}\|_2)$

$$e_n = -\operatorname{Log}_{10}(\|s_n - s\|_\infty)$$

$$f_n^{(i)} = -\operatorname{Log}_{10}(\|h_n^{(i)} - s\|_\infty) \qquad i = 1,2$$

$h_n^{(1)}$ : La transformation d'Henrici avec le H-algorithme

$h_n^{(2)}$ : La transformation d'Henrici avec la technique d'orthonormalisation.

Et à titre de comparaison, nous allons donner les résultats obtenus avec l'$\varepsilon$-algorithme vectoriel de Wynn [94], ainsi qu'avec l'$\varepsilon$-algorithme topologique de Brezinski [8].

Posons

$$k_n = -\operatorname{Log}_{10}(\|\varepsilon_{2p}^{(n)} - s\|_\infty) \qquad \text{pour l'}\varepsilon\text{-algorithme vectoriel}$$

$$l_n = -\operatorname{Log}_{10}(\|\varepsilon_{2p}^{(n)} - s\|_\infty) \qquad \text{pour l'}\varepsilon\text{-algorithme topologique}$$

avec le vecteur $y$ tel : $\qquad y_i = i \qquad \forall i = 1, \ldots, p$

| $n$ | $c_n$ | $d_n$ | $e_{n+3}$ | $f_n^{(1)}$ | $f_n^{(2)}$ | $h_{n-1}$ | $l_{n-1}$ |
|---|---|---|---|---|---|---|---|
| 0 | 0.103D+01 | 0.144D+00 | 0.716 | 2.234 | 2.234 | | |
| 1 | 0.117D+00 | 0.258D+00 | 1.103 | 2.968 | 2.968 | 2.389 | 1.242 |
| 2 | 0.136D-01 | 0.346D+00 | 1.452 | 3.714 | 3.714 | 3.145 | 2.250 |
| 3 | 0.163D-02 | 0.300D+00 | 1.779 | 4.469 | 4.469 | 3.900 | 3.061 |
| 4 | 0.197D-03 | 0.184D+00 | 2.094 | 5.231 | 5.231 | 4.656 | 3.835 |
| 5 | 0.240D-04 | 0.961D-01 | 2.401 | 5.998 | 5.998 | 5.416 | 4.601 |
| 6 | 0.295D-05 | 0.481D-01 | 2.706 | 6.769 | 6.769 | 6.181 | 5.367 |
| 7 | 0.365D-06 | 0.239D-01 | 3.009 | 7.542 | 7.541 | 6.950 | 6.137 |
| 8 | 0.453D-07 | 0.119D-01 | 3.310 | 8.317 | 8.313 | 7.721 | 6.908 |
| 9 | 0.564D-08 | 0.592D-02 | 3.612 | 9.093 | 9.084 | 8.495 | 7.682 |
| 10 | 0.703D-09 | 0.295D-02 | 3.913 | 9.869 | 9.847 | 9.270 | 8.457 |
| 11 | 0.877D-10 | 0.147D-02 | 4.214 | 10.646 | 10.581 | 10.046 | 9.233 |
| 12 | 0.109D-10 | 0.735D-03 | 4.515 | 11.424 | 11.167 | 10.823 | 10.010 |
| 13 | 0.137D-11 | 0.367D-03 | 4.816 | 12.201 | 11.584 | 11.600 | 10.787 |
| 14 | 0.171D-12 | 0.183D-03 | 5.117 | 12.979 | 12.016 | 12.378 | 11.565 |
| 15 | 0.213D-13 | 0.916D-04 | 5.419 | 13.757 | 12.268 | 13.156 | 12.343 |
| 16 | 0.267D-14 | 0.458D-04 | 5.720 | 14.535 | 12.654 | 13.933 | 13.121 |
| 17 | 0.333D-15 | 0.229D-04 | 6.021 | 15.313 | 13.108 | 14.711 | 13.899 |
| 18 | 0.416D-16 | 0.114D-04 | 6.322 | 16.091 | 13.458 | 15.489 | 14.677 |
| 19 | 0.521D-17 | 0.572D-05 | 6.623 | 16.870 | 13.772 | 16.266 | 15.455 |
| 20 | 0.651D-18 | 0.286D-05 | 6.924 | 17.648 | 14.075 | 17.035 | 16.233 |

On peut montrer que $\lim_{n \to +\infty} c_n = \lim_n d_n = 0$, ce qui est confirmé par les résultats numériques.

**Remarques**

① Bien que l'une des hypothèses du théorème 6 ne soit pas vérifiée, nous avons quand même accélération de la convergence de la suite $\{s_n\}$.

② Pour cet exemple la programmation avec le H-algorithme est meilleure.

③ l'ε-algorithme vectoriel ainsi que l'ε-algorithme topologique accélèrent aussi la convergence de la suite $\{s_n\}$.

**Exemple n°4** $\qquad s_{n+1} = G(s_n)$

$$G(x) = b + Ax + Q(x)$$

où

$$A = \begin{bmatrix} 2.25 & 0.01 & 0.05 & 0.5 \\ 0.01 & 1.75 & 0. & 0.05 \\ 0.05 & 0. & 1.75 & 0.01 \\ 0.5 & 0.05 & 0.01 & 2.25 \end{bmatrix}$$

$$b = (-0.81, -0.31, -0.31, -0.81)^T$$

$$Q(x) = -0{,}5(x_1^2 + x_1 x_4 \,,\, x_2^2 \,,\, x_3^2 \,,\, x_1 x_4 + x_4^2)^T$$

avec $\quad x = (x_1, x_2, x_3, x_4)^T$

Si $\quad s_0 = (1.5, 1.6, 1.7, 1.8)^T \qquad \lim_n s_n = (1,1,1,1)^T$

Utilisons les mêmes notations que pour l'exemple n°3

| $n$ | $e_{n+5}$ | $f_n^{(1)}$ | $f_n^{(2)}$ | $k_{n-3}$ | $\ell_{n-3}$ |
|---|---|---|---|---|---|
| 0 | 1.156 | 2.621 | 2.621 | | |
| 1 | 1.278 | 2.393 | 2.393 | | |
| 2 | 1.395 | 2.797 | 2.797 | | |
| 3 | 1.506 | 2.863 | 2.863 | 2.878 | 2.775 |
| 4 | 1.615 | 2.819 | 2.819 | 3.332 | 3.112 |
| 5 | 1.720 | 2.621 | 2.621 | 3.754 | 3.400 |
| 6 | 1.823 | 1.785 | 1.783 | 4.161 | 3.655 |
| 7 | 1.925 | 2.932 | 2.932 | 4.550 | 3.884 |
| 8 | 2.025 | 3.298 | 3.297 | 4.921 | 4.090 |
| 9 | 2.123 | 3.518 | 3.518 | 5.272 | 4.279 |
| 10 | 2.221 | 3.629 | 3.629 | 5.617 | 4.456 |
| 11 | 2.318 | 3.541 | 3.541 | 5.956 | 4.625 |
| 12 | 2.414 | 3.351 | 3.351 | 6.277 | 4.788 |
| 13 | 2.509 | 4.303 | 4.303 | 6.475 | 4.950 |
| 14 | 2.604 | 4.821 | 4.821 | 6.684 | 5.114 |
| 15 | 2.698 | 5.252 | 5.253 | 6.914 | 5.286 |
| 16 | 2.792 | 5.657 | 5.657 | 7.142 | 5.477 |
| 17 | 2.886 | 6.068 | 6.078 | 7.320 | 5.706 |
| 18 | 2.979 | 6.532 | 6.537 | 7.529 | 6.007 |
| 19 | 3.072 | 6.776 | 6.783 | 7.767 | 6.461 |
| 20 | 3.165 | 6.801 | 6.793 | 7.810 | 6.455 |
| 21 | 3.258 | 6.862 | 6.859 | 7.859 | 6.515 |
| 22 | 3.351 | 6.942 | 6.935 | 7.901 | 6.626 |
| 23 | 3.443 | 7.030 | 7.040 | 7.933 | 6.760 |
| 24 | 3.535 | 7.118 | 7.111 | 7.956 | 6.903 |
| 25 | 3.628 | 7.202 | 7.206 | 7.972 | 7.050 |
| 26 | 3.720 | 7.275 | 7.312 | 7.983 | 7.199 |
| 27 | 3.812 | 7.329 | 7.447 | 7.991 | 7.349 |

## Exemple n°5

$$s_{n+1} = G(s_n)$$

$$x = (x_1, x_2, x_3, x_4, x_5, x_6)^t$$

$$\begin{cases} G_1(x) = -0.75 - \dfrac{x_2^2 x_4 x_6}{4} \\ G_2(x) = -0.405\, e^{1+x_1 x_2} + 1.405 \\ G_3(x) = \dfrac{x_4 x_6}{2} - 1.5 \\ G_4(x) = 0.605\, e^{1-x_3^2} + 0.395 \\ G_5(x) = \dfrac{x_2 x_6}{2} - 1.5 \\ G_6(x) = x_1 x_5 \end{cases}$$

$s_0 = (0.1, 0.1, 0.1, 0.1, 0.1, 0.1)^T$ et $\lim_m s_m = (-1, 1, -1, 1, -1, 1)^T = s$

Avec les mêmes notations que pour l'exemple n°3 on a :

| $n$ | $e_{n+7}$ | $f_n^{(1)}$ | $f_n^{(2)}$ | $k_{n-5}$ | $\ell_{n-5}$ |
|---|---|---|---|---|---|
| 0 | 0.845 | 1.033 | 1.033 | | |
| 1 | 1.050 | 1.460 | 1.460 | | |
| 2 | 1.035 | 0.462 | 0.462 | | |
| 3 | 1.207 | 1.681 | 1.681 | | |
| 4 | 1.171 | 1.097 | 1.097 | | |
| 5 | 1.351 | 0.550 | 0.550 | 1.704 | 1.644 |
| 6 | 1.318 | 1.956 | 1.956 | 1.704 | 1.820 |
| 7 | 1.484 | 2.443 | 2.443 | 2.068 | 1.793 |
| 8 | 1.450 | 2.498 | 2.498 | 2.004 | 1.255 |
| 9 | 1.616 | 2.781 | 2.781 | 2.362 | 2.960 |
| 10 | 1.583 | 3.159 | 3.159 | 2.484 | 2.521 |
| 11 | 1.744 | 3.546 | 3.546 | 2.730 | 2.706 |
| 12 | 1.711 | 3.659 | 3.659 | 2.795 | 3.707 |
| 13 | 1.871 | 3.987 | 3.987 | 3.387 | 3.835 |
| 14 | 1.838 | 4.337 | 4.337 | 3.485 | 3.376 |
| 15 | 1.997 | 4.688 | 4.688 | 3.707 | 3.634 |
| 16 | 1.963 | 4.768 | 4.768 | 3.730 | 4.215 |
| 17 | 2.121 | 4.815 | 4.815 | 4.282 | 4.376 |
| 18 | 2.088 | 4.803 | 4.803 | 4.434 | 4.227 |
| 19 | 2.245 | 4.875 | 4.875 | 4.571 | 4.476 |
| 20 | 2.212 | 4.982 | 4.982 | 4.601 | 4.801 |
| 21 | 2.368 | 5.095 | 5.095 | 4.974 | 5.002 |
| 22 | 2.335 | 5.189 | 5.189 | 5.123 | 5.006 |
| 23 | 2.491 | 5.307 | 5.307 | 5.285 | 5.234 |
| 24 | 2.458 | 5.428 | 5.428 | 5.389 | 5.448 |
| 25 | 2.614 | 5.557 | 5.556 | 5.574 | 5.649 |
| 26 | 2.581 | 5.676 | 5.676 | 5.727 | 5.725 |
| 27 | 2.736 | 5.808 | 5.808 | 5.864 | 5.935 |
| 28 | 2.704 | 5.939 | 5.939 | 6.010 | 6.102 |
| 29 | 2.859 | 6.075 | 6.075 | 6.150 | 6.298 |

# Chapitre II

## Transformations composites de deux transformations vectorielles

## Introduction

Pour obtenir de nouvelles méthodes d'accélération, il est souvent utile de combiner deux (ou plusieurs) transformations de suites. Cette notion a été définie et étudiée par C. Brezinski dans le cas scalaire. Dans ce chapitre, nous examinerons son extension au cas vectoriel.

En se basant sur deux généralisations de la procédure $\theta$, nous proposerons deux transformations composites vectorielles que nous étudierons ensuite.

A partir de ces transformations, nous donnerons deux algorithmes généraux pour la résolution des systèmes d''equations non linéaires, nous montrerons qu'en particulier, le premier généralise la méthode de la sécante et le second généralise la méthode d'Henrici. Nous utiliserons aussi ces deux algorithmes pour donner deux méthodes de type Regula-Falsi.

## I Rappel (cas scalaire)

Une généralisation possible du procédé $\Delta^2$ d'Aitken est la transformation composité de rang deux qui a été définie et étudiée par C. Brezinski dans [18]. Donnons quelques propriétés de cette transformation.

Soit $\{s_n\}$ une suite complexe que l'on désire accélérer et soit $s$ sa limite. Considérons deux transformations $t_1$ et $t_2$ définies par :

$$t_i : \{s_n\} \longrightarrow \{t_i^{(n)}\} \qquad i=1,2 .$$

On appelle transformation composite de rang deux la transformation $T$ définie par:

$$T : \{s_n\} \longrightarrow \{T_n\}$$

où

$$T_n = a_n t_1^{(n)} + b_n t_2^{(n)} .$$

Si l'on veut que le noyau de la transformation $T$ contienne ceux des transformations $t_1$ et $t_2$ il faut prendre : $b_n = 1 - a_n$, mais cela ne suffit pas. Pour avoir cette propriété, un choix possible est celui proposé par C. Brezinski dans [18]:

$$a_n = \frac{\Delta t_1^{(n)}}{\Delta t_1^{(n)} - \Delta t_2^{(n)}}$$

il en résulte que

$$T_n = t_1^{(n)} - \frac{\Delta t_1^{(n)}}{\Delta t_2^{(n)} - \Delta t_1^{(n)}} \left( t_2^{(n)} - t_1^{(n)} \right) .$$

D'autres choix sont proposés dans [70].

Si l'on pose $t_1^{(n)} = s_n$ et $t_2^{(n)} = s_{n+1}$ alors $T_n = s'_n$.
C'est pourquoi cette transformation peut être considérée comme une généralisation du procédé $\Delta^2$ d'Aitken.

La suite $\{T_n\}$ possède des propriétés analogues a celles de la suite $\{s'_n\}$.
Tout d'abord $T_n$ peut se mettre sous la forme d'un rapport de deux déterminants.

$$T_n = \begin{vmatrix} t_1^{(n)} & t_1^{(n+1)} \\ t_2^{(n)} - t_1^{(n)} & t_2^{(n+1)} - t_1^{(n+1)} \end{vmatrix} \Big/ \begin{vmatrix} 1 & 1 \\ t_2^{(n)} - t_1^{(n)} & t_2^{(n+1)} - t_1^{(n+1)} \end{vmatrix}$$

où encore

$$T_n = \begin{vmatrix} t_1^{(n)} & t_2^{(n)} \\ \Delta t_1^{(n)} & \Delta t_2^{(n)} \end{vmatrix} \Big/ \begin{vmatrix} 1 & 1 \\ \Delta t_1^{(n)} & \Delta t_2^{(n)} \end{vmatrix} .$$

Le théorème suivant nous renseigne sur les conditions que doivent vérifier $\{t_1^{(n)}\}$ et $\{t_2^{(n)}\}$ pour que $T_n = s \quad \forall n > N$

<u>théorème 1</u> Supposons que $\forall n \quad t_1^{(n)} \neq t_2^{(n)}$

Une condition nécessaire et suffisante pour qu'il existe N tel que:
$\forall n \geq N \quad T_n = s$ est que :
$\exists a_0, a_1$ avec $a_0 + a_1 \neq 0$ tels que :

$$\forall n \geq N \qquad a_0 (t_1^{(n)} - s) + a_1 (t_2^{(n)} - s) = 0$$

Nous allons maintenant donner un résultat d'accélération de la convergence.

<u>théorème 2</u>

Si $t_2^{(n)} - s = (a+a_n)(t_1^{(n)} - s)$ avec $a \neq 1$ et $\lim_n a_n = 0$

et $t_1^{(n+1)} - s = (b+b_n)(t_1^{(n)} - s)$ avec $b \neq 1$ et $\lim_n b_n = 0$

alors $T_n - s = o(t_1^{(n)} - s)$

si de plus $a \neq 0$ alors $T_n - s = o(t_2^{(n)} - s)$

C'est une conséquence du théorème 3 de [18].

Rappelons aussi quelques propriétés de la procédure $\Theta$ [14]. Ces résultats seront généralisés par la suite.

Soit la suite $\{b_n\}$ telle que : $\forall n \quad \Delta b_n \neq 0$, et $f$ la fonction définie par: $f(\{a_n\},\{b_n\}) = f(a_n, b_n) = a_n - \frac{\Delta a_n}{\Delta b_n} b_n$.

Considérons un algorithme de la forme $c_n = a_n + b_n$, la procédure $\Theta$ consiste a remplacer $c_n$ par $\Theta(c_n)$ où :

$$\Theta(c_n) = f(a_n, b_n)$$

qui peut encore s'écrire :

$$\Theta(c_n) = \begin{vmatrix} a_n & a_{n+1} \\ b_n & b_{n+1} \end{vmatrix} \Big/ \begin{vmatrix} 1 & 1 \\ b_n & b_{n+1} \end{vmatrix}.$$

Si on applique la procédure $\Theta$ à $t_2^{(n)} = t_1^{(n)} + (t_2^{(n)} - t_1^{(n)})$ on obtient $T_n$.

Nous allons maintenant examiner, la généralisation de tout ce qui vient d'être rappelé au cas vectoriel.

## II) Généralisations de la procédure $\theta$.

Soit $S$ l'ensemble des suites de vecteurs complexes. Soit $S'$ le sous ensemble de $S$ des suites de vecteurs $\{b_n\}$ vérifiant :

$$b_n \in \mathbb{C}^p$$

et

la matrice $\Delta B_n = (\Delta b_n, \ldots, \Delta b_{n+p-1})$ est inversible $\forall n$.

Soient $f_1$ et $f_2$ deux applications définies par :

$$f_i : S \times S' \longrightarrow S$$

$$(\{a_n\}, \{b_n\}) \longrightarrow f_i(\{a_n\}, \{b_n\}) = \{f_i(a_n, b_n)\}_n \qquad i = 1, 2$$

où

$$f_1(a_n, b_n) = a_n - \Delta A_n (\Delta B_n)^{-1} b_n$$

et

$$f_2(a_n, b_n) = a_n - B_n (\Delta B_n)^{-1} \Delta a_n .$$

<u>Définition</u>

| Soit un algorithme de la forme<br>(I) $\quad c_n = a_n + b_n \qquad \{a_n\} \in S$ et $\{b_n\} \in S'$<br><br>Nous dirons que la procédure $\theta^{(i)}$ a été appliqué à cet algorithme si la relation (I) a été remplacée par la relation (II) :<br>(II) $\quad \theta^{(i)}(c_n) = f_i(a_n, b_n) \qquad i = 1$ ou $2$ |
|---|

Remarques :

① $\theta^{(1)}(c_m)$ nécessite la connaissance de $a_m, a_{m+1}, \dots, a_{m+p}$ et $b_m, b_{m+1}, \dots, b_{m+p}$,

par contre

$\theta^{(2)}(c_m)$ nécessite la connaissance seulement de $a_m, a_{m+1}$ et de $b_m, b_{m+1}, \dots, b_{m+p}$.

② si on écrit $s_{m+1} = s_m + \Delta s_m$, et on applique l'une des procédures $\theta^{(1)}$ ou $\theta^{(2)}$, on obtient la transformation d'Henrici.

En effet :

$$\Delta A_m = \Delta S_m \quad , \quad \Delta B_m = \Delta^2 S_m \ , \ B_m = \Delta S_n \text{ et } b_n = \Delta a_n = \Delta s_m$$

et donc $\theta^{(i)}(s_{m+1}) = h_m \quad i = 1,2 \quad \forall n$.

③ Ces deux procédures peuvent se mettre sous la forme d'un rapport de deux déterminants. Par utilisation de l'identité de Magnus rappelée dans le chapitre I nous avons :

$$\theta^{(1)}(c_n) = \begin{vmatrix} a_m & a_{m+1} & \dots & a_{m+p} \\ (b_m)_1 & (b_{m+1})_1 & \dots & (b_{m+p})_1 \\ \dots & \dots & \dots & \dots \\ (b_m)_p & (b_{m+1})_p & \dots & (b_{m+p})_p \end{vmatrix} \Bigg/ \begin{vmatrix} 1 & 1 & \dots & 1 \\ (b_m)_1 & (b_{m+1})_1 & \dots & (b_{m+p})_1 \\ \dots & \dots & \dots & \dots \\ (b_m)_p & (b_{m+1})_p & \dots & (b_{m+p})_p \end{vmatrix}$$

$$\theta^{(2)}(c_n) = \begin{vmatrix} a_m & b_m & \dots & b_{m+p-1} \\ (\Delta a_m)_1 & (\Delta b_m)_1 & \dots & (\Delta b_{m+p-1})_1 \\ \dots & \dots & \dots & \dots \\ (\Delta a_m)_p & (\Delta b_m)_p & \dots & (\Delta b_{m+p-1})_p \end{vmatrix} \Bigg/ \begin{vmatrix} 1 & 1 & \dots & 1 \\ (b_m)_1 & (b_{m+1})_1 & \dots & (b_{m+p})_1 \\ \dots & \dots & \dots & \dots \\ (b_m)_p & (b_{m+1})_p & \dots & (b_{m+p})_p \end{vmatrix}$$

Donnons maintenant quelques résultats pour ces deux procédures.

**Propriété 1**

Supposons que $\Delta B_m$ soit inversible $\forall n$

ⓐ 1- si $A \in L(\mathbb{C}^p)$, $b \in \mathbb{C}^p$, $C \in L(\mathbb{C}^p)$ et $C$ inversible

alors :

$$f_1(Aa_m + b, Cb_m) = A f_1(a_m, b_m) + b$$

2- $f_1(b_m, b_m) = 0$

ⓑ si $b \in \mathbb{C}^p$, $A \in L(\mathbb{C}^p)$ et $A$ inversible

alors

$$f_2(Aa_m + b, Ab_m) = A f_2(a_m, b_m) + b$$

ⓒ Si $x \in \mathbb{C}$, $y \in \mathbb{C}^*$ et $b \in \mathbb{C}^p$

$$f_i(x a_m + b, y b_m) = x f_i(a_m, b_m) + b_m \qquad i=1,2$$

preuve:

ⓐ 1- $f_1(Aa_m + b, Cb_m) = Aa_n + b - (A\Delta A_m)(C\Delta B_n)^{-1} C b_m$
$= A f_1(a_m, b) + b$.

2- $f_1(b_m, b_m) = b_n - \Delta B_n (\Delta B_n)^{-1} b_m = 0$

ⓑ $f_2(A a_n + b, Ab_m) = Aa_m + b - (AB_m)(A \Delta B_m)^{-1} A \Delta a_n$
$= A f_2(a_m, b_n) + b$.

ⓒ $f_1(x a_m + b, y b_m) = x a_n + b - (x \Delta A_n)(y \Delta B_n)^{-1} y b_n$
$= x f_2(a_n, b_n) + b$

$f_2(x a_n + b, y b_n) = x a_m + b - (y B_m)(y \Delta B_n)^{-1} x \Delta a_n$
$= x f_2(a_n, b_m) + b$

■

Pour la procédure $\theta^{(1)}$, nous avons le résultat suivant :

**théorème 3** Supposons que $\Delta B_n$ soit inversible $\forall n$.
Si $a_n = s + C b_n$ où $C \in L(\mathbb{C}^p)$
alors $f_1(a_n, b_n) = s$

preuve:

$$f_1(s + Cb_n, b_n) = C f_1(b_n, b_n) + s$$
$$= s$$

d'après la propriété 1. ■

Pour la procédure $\theta^{(2)}$, nous avons une condition nécessaire et suffisante pour que $f_2(a_n, b_n) = s$. Nous allons donner un théorème qui généralise le théorème 1 de [14 p 139]:

**théorème 4** Supposons que les matrices $B_n$ et $\Delta B_n$ soient inversibles pour tout $n$.
Une condition nécessaire et suffisante pour que $f_2(a_n, b_n) = s$ $\forall$ est qu'il existe $p$ nombres complexes $x_0, x_1, \ldots, x_{p-1}$ tels que:
$$a_n = s + \sum_{i=0}^{p-1} x_i \, b_{n+i} \qquad \forall n$$

preuve: <u>Condition suffisante</u>

On a le système d'equations linéaires :

$$\begin{cases} a_n = s + \sum_{i=0}^{p-1} x_i \, b_{n+i} \\ a_{n+1} = s + \sum_{i=0}^{p-1} x_i \, b_{n+i+1} \end{cases}$$

Ce système peut s'écrire :

$$\begin{cases} s = a_n - \sum_{i=0}^{p-1} x_i \, b_{n+i} & (1) \\ \text{et} \\ \Delta a_n = \sum_{i=0}^{p-1} x_i \Delta b_{n+i} & (2) \end{cases}$$

de système (2) est un système d'equations linéaires de $p$ équations à $p$ inconnues, dont le déterminant principal est $\det(\Delta B_n)$ qui est non nul par hypothèse ; donc en résolvant ce système on obtient les coefficients $x_i$ pour $i = 0, \dots, p-1$ et en reportant dans (4) on obtient $f_2(a_n, b_n) = s \quad \forall n$.

<u>Condition nécessaire</u> :

$$f_2(a_n, b_n) = s \iff \begin{vmatrix} a_n - s & b_n & \dots\dots & b_{n+p-1} \\ (\Delta a_n)_1 & (\Delta b_n)_1 & & (\Delta b_{n+p-1})_1 \\ & & & \\ (\Delta a_n)_p & (\Delta b_n)_p & & (\Delta b_{n+p-1})_p \end{vmatrix} = 0$$

Développons ce déterminant par rapport à sa première ligne :

$$y_0^{(n)} (a_n - s) + \sum_{i=1}^{p} y_i^{(n)} b_{n+i-1} = 0 .$$

Or

$$y_0^{(n)} = \det(\Delta B_n) \neq 0 \qquad \text{par hypothèse}$$

et donc

$$* \qquad a_n - s = \sum_{i=0}^{p-1} x_i^{(n)} b_{n+i} \qquad \text{avec} \quad x_i^{(n)} = - \frac{y_{i+1}^{(n)}}{y_0^{(n)}} .$$

L'équivalence précédente peut aussi s'écrire :

$$f_2(a_n, b_n) = s \iff \begin{vmatrix} a_{n+1} - s & b_{n+1} \dots & \dots\dots & b_{n+p} \\ (\Delta a_n)_1 & (\Delta b_n)_1 & \dots & (\Delta b_{n+p-1})_1 \\ \dots & \dots & \dots & \dots \\ (\Delta a_n)_p & (\Delta b_n)_p & \dots & (\Delta b_{n+p-1})_p \end{vmatrix} = 0 ,$$

En développant ce déterminant, on obtient :

$$** \qquad a_{n+1} - s = \sum_{i=0}^{p-1} x_i^{(n)} b_{n+i+1} .$$

Ecrivons la relation $*$ en remplaçant $n$ par $n+1$ et faisons la différence avec $**$ on obtient :

$$0 = \sum_{i=0}^{k-1} \Delta x_i^{(n)} \, b_{n+i+1} \quad .$$

Le déterminant de ce système homogène est $\det(B_{n+1})$, qui est non nul par hypothèse donc $\Delta x_i^{(n)} = 0$

d'où $x_i^{(n)} = x_i$, ce qui termine la démonstration.

■

Remarquons que l'hypothèse $B_n$ inversible a été utilisée seulement pour la démonstration de la condition nécessaire.

Nous avons vu dans les rappels, comment la procedure $\theta$ peut être utilisée pour obtenir la transformation composite de rang deux dans le cas scalaire. De la même façon nous allons utiliser les deux procédures $\theta^{(1)}$ et $\theta^{(2)}$ pour obtenir deux transformations composites vectorielles.

## III – Première transformation composite vectorielle.

### 1) Définition

Soient $t_1$ et $t_2$ deux transformations de suites vectorielles :

$$t_i : \{s_n\} \longrightarrow \{t_i^{(n)}\} \qquad \text{pour } i=1,2$$
$$t_i^{(n)} \in \mathbb{C}^p$$

Appliquons la procédure $\theta^{(1)}$ à $c_n = t_1^{(n)} + (t_2^{(n)} - t_1^{(n)})$,

nous obtenons :

$$\boxed{t_{p,n} = \theta^{(1)}(c_n) = t_1^{(n)} - \Delta T_1^{(n)} \left(\Delta T_2^{(n)} - \Delta T_1^{(n)}\right)^{-1} (t_2^{(n)} - t_1^{(n)})}$$

si $t_1^{(n)} = s_n$ et $t_2^{(n)} = s_{n+1}$ alors $t_{p,n} = h_n$.

Remarque : Cette transformation peut être obtenue d'une autre façon [21],[70] :
Plaçons nous en premier lieu dans le cas scalaire :

le point fixe de la fonction $f$ définie par :

$$\begin{cases} f : \mathbb{C} \longrightarrow \mathbb{C} \\ f(x) = ax+b \quad \text{et } a \neq 1 \quad , a \in \mathbb{C} \text{ et } b \in \mathbb{C} \end{cases}$$

et telle que $f(t_1^{(n+i)}) = t_2^{(n+i)}$ $i=0,1$

est $T_n$ , On retrouve ainsi la transformation composite de rang deux ( cas scalaire ).

Dans le cas vectoriel

$t_{p,n}$ est le point fixe de la fonction $F$ définie par :

$$F : \mathbb{C}^p \longrightarrow \mathbb{C}^p$$

$$\begin{cases} F(x) = Ax+b \quad , A \in L(\mathbb{C}^p) \text{ et } b \in \mathbb{C}^p \\ \text{et} \\ F(t_1^{(n+i)}) = t_2^{(n+i)} \quad \text{pour } i=0,1,\ldots,p. \end{cases}$$

Avant d'étudier cette transformation, montrons que $t_{p,n}$ peut être calculé par le H-algorithme.

Nous avons déjà vu comment écrire $\theta^{(n)}(c_n)$ comme rapport de deux déterminants, et donc en particulier :

$$t_{p,n} = \begin{vmatrix} t_1^{(n)} & t_1^{(n+1)} & \cdots & t_1^{(n+p)} \\ (t_2^{(n)}-t_1^{(n)})_1 & (t_2^{(n+1)}-t_1^{(n+1)})_1 & \cdots & (t_2^{(n+p)}-t_1^{(n+p)})_1 \\ \cdots & \cdots & & \cdots \\ (t_2^{(n)}-t_1^{(n)})_p & (t_2^{(n+1)}-t_1^{(n+1)})_p & \cdots & (t_2^{(n+p)}-t_1^{(n+p)})_p \end{vmatrix} \Bigg/ \begin{vmatrix} 1 & \cdots & 1 \\ (t_2^{(n)}-t_1^{(n)})_1 & \cdots & (t_2^{(n+p)}-t_1^{(n+p)})_1 \\ \cdots & & \cdots \\ (t_2^{(n)}-t_1^{(n)})_1 & \cdots & (t_2^{(n+p)}-t_1^{(n+p)})_p \end{vmatrix}$$

Avec cette écriture, il apparait clairement que le calcul de $t_{p,n}$ peut se faire avec le H-algorithme. On a donc le résultat suivant.

Si $\quad H_0^{(n)} = t_1^{(n)}$ et $\quad g_i(n) = (t_2^{(n)} - t_1^{(n)})_i \quad i = 1, \ldots p$

alors

$$t_{p,n} = H_p^{(n)}$$

* Comme pour la transformation d'Henrici, on peut utiliser la méthode d'orthogonalisation de Gram-Schmidt étudiée dans le chapitre I, pour calculer $t_{p,n}$. Il suffit de poser :

$$A_n = \Delta T_2^{(n)} - \Delta T_1^{(n)} .$$

## 2- étude de la premiere transformation composite.

Commençons par donner un résultat concernant le noyau de cette transformation.

### théorème 5

Supposons que la matrice $\{\Delta T_2^{(n)} - \Delta T_1^{(n)}\}$ soit inversible $\forall n$.

Si $\quad t_1^{(n)} - s = C\,(t_2^{(n)} - t_1^{(n)}) \quad \forall n$

où $C$ est une matrice carrée de $L(\mathbb{C}^p)$,

alors

$$t_{p,n} = s \quad \forall n$$

preuve : Posons $\quad a_n = t_1^{(n)}$ et $b_n = t_2^{(n)} - t_1^{(n)}$

et on applique le théorème 3 ∎

Donnons maintenant un théorème de convergence pour $t_{p,n}$. Ce théorème est analogue au théorème 2 de [18].

Etant donné une norme $\| \ \|$ de $\mathbb{C}^p$. On définit une norme matricielle associée à la norme $\| \ \|$ par :

$A \in L(\mathbb{C}^p)$

$$\|A\| = \sup_{\substack{x \in \mathbb{C}^p \\ x \neq 0}} \frac{\|Ax\|}{\|x\|} .$$

<u>théorème 6</u>

> . Si $\lim t_1^{(n)} = \lim t_2^{(n)} = s$,
>
> et pour tout $n$ la matrice $\{\Delta T_1^{(n)}\}$ est inversible.
>
> . Si de plus $B_n = \Delta T_2^{(n)} \Delta T_1^{(n)^{-1}}$
>
> et $\exists \gamma > 0 \quad \exists N \quad \forall n > N \quad \|B_n\| \leq \gamma < 1$
>
> alors
>
> 1°) La matrice $\{\Delta T_2^{(n)} - \Delta T_1^{(n)}\}$ est inversible pour $n > N$
>
> 2°) $\lim_n t_{p,n} = s$

preuve :

1°) Prenons $A = I$, $\|A\| = 1 = \alpha$ et $C = I - B_n$

$A - C = B_n$ on a $\|A - C\| = \|B_n\| < \gamma$, $\forall n > N$.

Posons $\beta = \gamma$, et comme $\gamma = \alpha\beta < 1$, d'après le lemme 2 du chapitre I, la matrice $C = I - B_n$ est inversible et $\|C^{-1}\| \leq \frac{1}{1-\gamma}$

De plus

$$\Delta T_2^{(n)} - \Delta T_1^{(n)} = \left[\Delta T_2^{(n)} \Delta T_1^{(n)^{-1}} - I\right] \Delta T_1^{(n)} \qquad \forall n > N$$

Donc la matrice $\Delta T_2^{(n)} - \Delta T_1^{(n)}$ est inversible $\forall n > N$

et par conséquent :

$t_{p,n}$ est définie pour tout $n > N$.

2°) $$t_{p,n} - s = t_1^{(n)} - s - (B_n - I)^{-1} \left(t_2^{(n)} - s - (t_1^{(n)} - s)\right)$$

donc

$$\|t_{p,n} - s\| \leqslant \|t_1^{(n)} - s\| + \frac{1}{1-\gamma} \left\{ \|t_2^{(n)} - s\| + \|t_1^{(n)} - s\| \right\}$$

On passe à la limite quand $n$ tend vers l'infini on a donc :

$$\lim_{n \to +\infty} t_{p,n} = s \qquad \blacksquare$$

On peut également démontrer un résultat analogue à celui du théorème 2 pour la transformation composite de rang deux scalaire. Auparavant nous avons besoin du résultat suivant :

théorème 7

> Si $t_2^{(n)} - s = (A + A_n)(t_1^{(n)} - s)$ avec $\lim_n A_n = 0$ et $(A, A_n) \in L(\mathbb{C}^p)^2$
>
> et si
>
> $$t_1^{(n+1)} - s = (B + B_n)(t_1^{(n)} - s) \quad \text{avec} \quad \lim_n B_n = 0 \ \text{ et } (B, B_n) \in L(\mathbb{C}^p)^2$$
>
> si de plus $(B - I)$ est inversible, alors :
>
> $$\exists N, \quad \forall n > N \qquad \Delta t_2^{(n)} = (A + C_n)\, \Delta t_1^{(n)} \quad \text{avec} \quad \lim_n C_n = 0$$

preuve :

Par définition $\quad t_2^{(n+1)} - s = (A + A_{n+1})(t_1^{(n+1)} - s)$

$$= (A + A_{n+1})(B + B_n)(t_1^{(n)} - s)$$

$$* \qquad t_2^{(n+1)} - s = (AB + D_n) \qquad \text{avec} \qquad \lim_n D_n = 0,$$

d'autre part

$$\Delta t_1^{(n)} = (t_1^{(n+1)} - s) - (t_1^{(n)} - s) = (B - I + B_n)(t_1^{(n)} - s)$$

$$= (B - I)(I + E_n)(t_1^{(n)} - s)$$

et $\lim_n E_n = 0$ .

D'après le lemme 2 (chapitre I) :

$\exists N', \forall n > N'$ , $I + E_n$ est inversible.

Posons $F_n = I - (I+E_n)^{-1} \quad \forall n > N'$

$$\|F_n\| \leq \frac{\|E_n\|}{1-\|E_n\|} \quad \text{et donc } \lim_n F_n = 0$$

d'où

$$** \qquad t_1^{(m)} - s = (I - F_n)(B-I)^{-1} \Delta t_1^{(m)}.$$

En utilisant l'égalité $*$, on voit que :

$$\Delta t_2^{(m)} = (AB - A + G_n)(t_1^{(m)} - s) \quad \text{où l'on a posé } G_n = D_n - A_n.$$

En remplaçant $t_1^{(m)} - s$ par sa valeur dans $**$, on obtient:

$$\Delta t_2^{(m)} = (AB - A + D_n)(I - F_n)(B-I)^{-1} \Delta t_1^{(m)}$$
$$= [A + C_n] \Delta t_1^{(m)}$$
$$\text{où} \quad \lim_n C_n = 0$$

■

Ce théorème généralise le théorème 5 du chapitre I, pour le retrouver il suffit de poser $t_1^{(m)} = s_n$ et $t_2^{(m)} = s_{n+1}$ et donc $A = B$ et $A_n = B_n$.

Nous pouvons maintenant démontrer le théorème fondamental suivant

**théorème 8**

si $t_2^{(m)} - s = (A + A_n)(t_1^{(m)} - s)$ où $\lim_m A_m = 0$ et $(A, A_n) \in L(\mathbb{C}^p)^2$

et $t_1^{(m+1)} - s = (B + B_n)(t_1^{(m)} - s)$ où $\lim_n B_n = 0$ et $(B, B_n) \in L(\mathbb{C}^p)^2$.

Si de plus $\rho(A) < 1$ et $\rho(B) < 1$,

et si les matrices $\Delta T_1^{(m)}$ sont uniformément inversibles

alors

1) $t_{p,n}$ est définie pour n assez grand.

2) $\|t_{p,n} - s\| = o\,(\|t_1^{(n)} - s\|)$ .

. Si de plus A est inversible alors :

$$\|t_{p,n} - s\| = o\,(\|t_2^{(n)} - s\|) .$$

preuve:

par définition $t_{p,n}$ s'écrit :

$$t_{p,n} - s = t_1^{(n)} - s - \Delta T_1^{(n)} \left( \Delta T_2^{(n)} - \Delta T_1^{(n)} \right)^{-1} (t_2^{(n)} - t_1^{(n)}) \qquad * .$$

Par hypothèse et par utilisation du théorème précédent on a:

$$\exists N', \quad \forall n > N' \qquad \Delta t_2^{(n)} = (A + C_n)\, \Delta t_1^{(n)} ,$$

d'où encore

$$\Delta t_2^{(n)} - \Delta t_1^{(n)} = (A - I + C_n)\, \Delta t_1^{(n)}$$

et par conséquent

$$\Delta t_2^{(n+i)} - \Delta t_1^{(n+i)} = (A - I)\, \Delta t_1^{(n+i)} + C_{n+i}\, \Delta t_1^{(n+i)} \quad \text{pour } i = 0, \dots, p-1,$$

Ecrivons ces relations sous forme matricielle on a :

$$\Delta T_2^{(n)} - \Delta T_1^{(n)} = (A - I)\,(I + D_n)\, \Delta T_1^{(n)}$$

où l'on a posé

$$D_n = (A - I)^{-1} \left( C_n \Delta t_1^{(n)}, \dots, C_{n+p-1} \Delta t_1^{(n+p-1)} \right) \Delta T_1^{(n)^{-1}} .$$

D'aprés le lemme 3 (chapitre I) , $\lim_n D_n = 0$ .

1) $(I + D_n)$ est inversible $\quad \forall n > N''$

donc $\Delta T_2^{(n)} - \Delta T_1^{(n)}$ est inversible $\quad \forall n > N = \max(N', N'')$

et par conséquent, $t_{p,n}$ est bien définie $\quad \forall n > N$ .

Montrons la seconde partie du théorème .

2) Posons $(I+D_n)^{-1} = I - F_n \qquad \forall n > N''$

Par suite :

$$\left(\Delta T_2^{(n)} - \Delta T_1^{(n)}\right)^{-1} = \Delta T_1^{(n)^{-1}} (I - F_n)(A - I)^{-1}.$$

En reportant cette égalité dans * on obtient :

$$t_{p,n} - s = t_1^{(n)} - s - (I - F_n)(A - I)^{-1}(t_2^{(n)} - t_1^{(n)}).$$

Comme par ailleurs :

$$t_2^{(n)} - t_1^{(n)} = (A - I + A_n)(t_1^{(n)} - s)$$

il s'ensuit que

$$t_{p,n} - s = t_1^{(n)} - s - (I - F_n)(A - I)^{-1}(A - I + A_n)(t_1^{(n)} - s)$$

$$= t_1^{(n)} - s - (I - E_n)(t_1^{(n)} - s)$$

avec $\lim_n E_n = 0$

où encore

$$t_{p,n} - s = E_n (t_1^{(n)} - s) \qquad ** .$$

Par conséquent

$$\| t_{p,n} - s \| = o(\| t_1^{(n)} - s \|) .$$

Si de plus $A$ est inversible, alors :

$$t_2^{(n)} - s = A(I + G_n)(t_1^{(n)} - s)$$

où l'on a posé $\quad G_n = A^{-1} A_n \quad$ et $\lim_n G_n = 0$.

$\exists N''', \forall n > N'''$ $\quad (I + G_n)$ est inversible et donc

$$t_1^{(n)} - s = (I + G_n)^{-1} A^{-1} (t_2^{(n)} - s) \qquad \forall n > N''' .$$

En reportant cette égalité dans ** on a :

$$t_{p,n} - s = E_n (I + G_n)^{-1} A^{-1} (t_2^{(n)} - s) = H_n (t_2^{(n)} - s) .$$

comme $\lim_n H_n = 0$ on a donc $\| t_{p,n} - s \| = o(\| t_2^{(n)} - s \|)$ □

Ce théorème généralise le théorème 2 ainsi que le théorème 6 du premier chapitre.

## 3) Application à la résolution des systèmes d'équations non linéaires.

Nous allons examiner dans ce qui suit, la connexion qui existe entre des méthodes de résolution des systèmes d'équations non linéaires et les transformations composites vectorielles. Nous allons nous intéresser aux méthodes qui ne nécessitent pas de calcul des dérivées, et qui sont d'ordre plus grand que un.

Soit à resoudre le système d'équations

$$\begin{cases} F(x) = 0 \\ \text{où} \quad F : \mathbb{C}^p \longrightarrow \mathbb{C}^p \end{cases}$$

ce système peut aussi s'écrire sous la forme suivante:

$$\begin{cases} x = G(x) \\ \text{où} \quad G : \mathbb{C}^p \longrightarrow \mathbb{C}^p \end{cases} .$$

Nous supposerons qu'il existe $\overset{*}{x}$ tel que $F(\overset{*}{x}) = 0$ et $G(\overset{*}{x}) = \overset{*}{x}$.

On se donne un entier naturel $l$ et des fonctions arbitraires $A_{i,l}$ et $B_{i,l}$ :

$$A_{i,l} : (\mathbb{C}^p)^{l+1} \longrightarrow \mathbb{C}^p \qquad \text{pour } i = 0, \ldots, p$$

$$B_{i,l} : (\mathbb{C}^p)^{l+1} \longrightarrow \mathbb{C}^p \qquad \text{pour } i = 0, \ldots, p$$

$$\text{avec} \begin{cases} A_{i,l}(\overset{*}{x}, \overset{*}{x}, \ldots, \overset{*}{x}) = \overset{*}{x} \\ \text{et} \quad B_{i,l}(\overset{*}{x}, \overset{*}{x}, \ldots, \overset{*}{x}) = \overset{*}{x} \end{cases}$$

$$\text{et} \quad A_{i,l} \not\equiv B_{i,l} \qquad \forall i = 0, \ldots, p .$$

On propose l'algorithme général suivant:

$C_I$

**Initialisation** : Soit $l$ un entier fixé et $x^{-l}, \dots, x^{0}$ donnés, $k=0$

**itération $k+1$**

$$t_1^{(i)} = A_{i,l}(x^{k-l}, \dots, x^{k})$$

et

$$t_2^{(i)} = B_{i,l}(x^{k-l}, \dots, x^{k}) \qquad i=0,\dots,p$$

et

$$x^{k+1} = t_{p,0}$$

Notons que $x^{k+1}$ peut être calculé avec le H-algorithme.

Nous allons montrer que cet algorithme généralise la méthode de la sécante [60]

a) La méthode de la sécante

La méthode de la sécante est une méthode générale pour la résolution des systèmes d''equations. Elle dépend de P suites auxiliaires $\{x^{k,i}\}$ pour $i=1,\dots,p$. Chaque choix particulier des $\{x^{k,i}\}$ donne une méthode particulière. Nous verrons que cette méthode englobe beaucoup de méthodes connues notamment la méthode d'Henrici, celle d'Ul'm, la méthode de la sécante a $p+1$ points et bien d'autres encore.

Commençons par rappeler la définition de cette méthode:

Posons $\quad H=(h^1, \dots, h^p) \quad , H \in L(\mathbb{C}^p)$

et $h^i \in \mathbb{C}^p \quad i=1,\dots p$

Soit D une partie de $\mathbb{C}^p$ ; On définit une application J par :

$$J : D \times L(\mathbb{C}^p) \to L(\mathbb{C}^p)$$

$$J(x,H) = \left(F(x+h^1)-F(x), \ldots\ldots, F(x+h^p)-F(x+h^{p-1})\right) \hat{H}^{-1}$$

et

$$\hat{H} = (h^1, h^2-h^1, \ldots, h^p-h^{p-1}).$$

La méthode de la sécante est définie par :

$$\left[\begin{array}{l} x^{k+1} = x^k - J(x^k, H_k)^{-1} F(x^k) \\ \text{et} \quad H_k = (x^{k,1}-x^k, \ldots\ldots, x^{k,p}-x^k) \\ \text{où} \quad x^{k,i} \in \mathbb{C}^p \quad \text{pour } i=1,\ldots,p \end{array}\right.$$

Montrons que cette méthode est un cas particulier de l'algorithme $C_I$.

Posons $F(x) = x - G(x)$,

$F(x^{k,i}) = x^{k,i} - G(x^{k,i})$

et

$$J(x^k,H_k) = \left(x^{k,1}-x^k-(G(x^{k,1})-G(x^k)), \ldots\ldots, x^{k,p}-x^{k,p-1}-(G(x^{k,p})-G(x^{k,p-1}))\right)\hat{H}_k^{-1}$$

avec

$$\hat{H}_k = (x^{k,1}, \ldots\ldots, x^{k,p}-x^{k,p-1}).$$

Posons aussi

$$x^{k,0} = x^k = A_{0,\ell}(x^{k-\ell}, \ldots, x^k)$$

$$x^{k,i} = A_{i,\ell}(x^{k-\ell}, \ldots, x^k) \quad \text{pour } i=1,\ldots,p$$

et

$$B_{i,\ell} = G \circ A_{i,\ell} \quad \text{pour } i=0,\ldots,p.$$

Comme
$$\begin{cases} t_1^{(i)} = A_{i,\ell}(x^{k-\ell}, \dots, x^k) \\ t_2^{(i)} = B_{i,\ell}(x^{k-\ell}, \dots, x^k) \end{cases} \quad \text{pour } i=0,\dots,p.$$

on a donc
$$\hat{H}_k = \Delta T_1^{(0)}$$
et
$$J(x^k, H_k) = (\Delta T_1^{(0)} - \Delta T_2^{(0)})\ \Delta T_1^{(0)^{-1}}$$

Par conséquent :
$$x^{k+1} = t_1^{(0)} - \Delta T_1^{(0)}(\Delta T_2^{(0)} - \Delta T_1^{(0)})^{-1}(t_2^{(0)} - t_1^{(0)})$$

Cette méthode est donc bien un cas particulier de l'algorithme $(\mathcal{E}_1)$.

Comme on l'a déja dit, la méthode de la sécante dépend d choix des $x^{k,i}$ ; certaines choix possibles soit examinés dans le chapitre 7 de [60]. Nous allons donner quelques exemples :

1°) Prenons $\ell = 1$ et $x^{k,i} = x^k + (x_i^{k-1} - x_i^k)\, e^i \qquad i = 1, .. p$

(rappelons que $e^i$ est un vecteur de la base canonique : $e_j^i = \begin{cases} 1 & i=j \\ 0 & i \neq j \end{cases}$
$x_i^k$ c'est la i-ème composante du vecteur $x^k$.)

Cette méthode est appelée méthode de la sécante à deux points dans [ ], méthode de Newton modifiée dans [50], méthode de Régula-Falsi dans [2:] et méthode discrète de Newton dans d'autres articles.

Une autre méthode du même type est celle défini par le choix suivant :
$$x^{k,i} = x^k + \sum_{j=1}^{i} (x_j^{k-1} - x_j^k)\, e^j \qquad , i = 1, - p$$

Ces deux méthodes sont d'ordre $\frac{1+\sqrt{5}}{2}$, cependant la dernière est meilleure que la première puisqu'elle nécessite seulement $p$ évaluations de fonctions alors que l'autre nécessite $p+1$ évaluations de fonctions à chaque itération.

2°) $\ell = 0$, et $x^{k,i} = x^k + (F(x^k))_i\, e^i \quad i = 1, \ldots p$

C'est une méthode de type Steffensen. Elle est d'ordre 2 et nécessite $p+1$ évaluations de fonctions à chaque itération.

On obtient une méthode très similaire en prenant :

$$x^{k,i} = x^k + \sum_{j=1}^{i} (F(x^k))_j\, e^j .$$

Ces deux méthodes possèdent les mêmes propriétés.

3°) $\ell = 0$

$$\begin{cases} x^{k,i} = x^{k,i-1} + (G(x^k) - x^k)_i\, e^i & i = 1, \ldots p \\ x^{k,0} = x^k \end{cases}$$

avec $G(x) = x - F(x)$

Cette méthode a été proposé par Ulm dans [85].
C'est une méthode de type Steffensen qui est d'ordre 2 et nécessite $(p+1)$ évaluations de fonctions.

Une variante de cette méthode est définie par [85] :

$\ell = 0$

$$\begin{cases} x^{k,0} = G(x^k) \\ x^{k,i} = x^{k,i-1} + (x^k - G(x^k))_i\, e^i & i = 1, \ldots p \end{cases}$$

Elle possède les mêmes propriétés que la première méthode.

4°) <u>La méthode de la sécante a $p+1$ points</u>

Cette méthode est très ancienne, puisqu'elle a été utilisé par Gauss [32], [33] dans le cas $p=2$ pour résoudre des problèmes astronomiques.
La généralisation à $p$ quelconque a été faite indépendamment par Wolfe [92] et par Bittner [5].
Posons $l=p$ et $x^{k,i}=x^{k-i}$ pour $i=0,\dots,p$, la méthode de la sécante devient donc :

$$\begin{cases} x^{-p},\dots,\overset{\circ}{x} \text{ donnés, et} \\ x^{k+1}=x^k-\left(x^k-x^{k-1},\dots\dots,x^{k-p+1}-x^{k-p}\right)\left(F(x^k)-F(x^{k-1}),\dots\dots,F(x^{k-p+1})-F(x^{k-p})\right)^{-1}F(x^k) \end{cases}$$

Remarques :

1) ~~La méthode à (p+1) points nécessite~~ une évaluation de fonctions à chaque itération autre que la première, qui elle nécessite $p+1$ évaluations.

2) Posons
$$H_k=\left(x^k-x^{k-1},\dots\dots\dots,x^{k-p+1}-x^{k-p}\right)$$
et
$$\Gamma_k=\left(F(x^k)-F(x^{k-1}),\dots\dots,F(x^{k-p+1})-F(x^{k-p})\right)$$
et par conséquent:
$$x^{k+1}=x^k-H_k\Gamma_k^{-1}F(x^k).$$
Notons $v^1,\dots,v^p$ les lignes de $\Gamma_k^{-1}$ et $B$ la matrice qui a pour lignes les vecteurs $v^p,v^1,\dots,v^{p-1}$, alors:
$$\Gamma_{k+1}^{-1}=B-\frac{B(q^k-q^{k-p})v^p}{1+v^p(q^k-q^{k-p})}$$
où $\quad q^i=F(x^{i+1})-F(x^i)$.

Avec ces résultats, nous remarquons que le calcul de $x^1$ se fait en $O(p^3)$ opérations arithmétiques alors que le calcul de $x^{k+1}$

connaissant $x^k$, se fait en $O(p^2)$ opérations arithmétiques. Malheureusement le calcul de $x^k$ avec cette technique est très instable numériquement.

③ Si $F : D \subset \mathbb{R}^p \longrightarrow \mathbb{R}^p$ où $D$ est un convexe ouvert et $F$ continûment differentiable dans $D$.

et si $\forall (x,y) \subset D \times D$ $\qquad \|F'(x) - F'(y)\| \leq M\|x-y\|$

$\overset{*}{x} \in D$ et $F(\overset{*}{x}) = 0$ et $F'(\overset{*}{x})$ inversible.

Si de plus $\forall k$ la matrice $H_k$ est uniformément inversible. alors l'ordre de la méthode de la sécante a $(p+1)$ points est la racine positive de l'équation $t^{p+1} - t^p - 1 = 0$. pour la démonstration voir [5], [45], [60], [73].

de problème de cette méthode est l'hypothèse sur la matrice $H_k$. Gragg et stewart [?] ont introduit un algorithme stable pour l'implémentation de cette méthode ; cette implémentation utilise la factorisation $QR$ ainsi que sa mise à jour ( le coût est en $O(p^2)$ pour calculer $x^{k+1}$ connaissant $x^k$); de plus ils ont proposé une technique de correction dans le cas où la matrice $H_k$ n'est pas uniformement inversible.

On trouvera dans [56] et [57] d'autres techniques d'implémentation de cette méthode.

Si l'on pose $A_k = I_k$ on peut utiliser la méthode de Gram-Schmidt du chapitre I. Nous avons montré que cette méthode pouvait être implémentée par le H-algorithme et par la factorisation $Q_\times R$

Donnons quelques exemples numériques pour illustrer le comportement de ces deux techniques.

**Exemple n°1** $F(x) = x - G(x)$

avec $G(x) = b + Ax + Q(x)$ où

$$A = \begin{bmatrix} 3.9 & -3.7 & 2.4 & -0.6 \\ 2.4 & -2.0 & 2.2 & -0.6 \\ 2.4 & -3.6 & 4.1 & -0.9 \\ 2.8 & -5.2 & 4.8 & -0.4 \end{bmatrix}$$

$$b = (-0{,}75, -0{,}75, -0{,}75, -0{,}75)$$

$$x = (x_1, x_2, x_3, x_4)^T$$

$$Q(x) = -0{,}25(x_1^2, x_2^2, x_3^2, x_4^2)^T$$

$$x^{-4} = \begin{pmatrix} 1.5 \\ 1.6 \\ 1.7 \\ 1.8 \end{pmatrix} \qquad x^i = G(x^{i-1}) \text{ pour } i = -3, -2, -1, 0 \; ; \; \overset{*}{x} = (3,3,3,3)^T$$

Posons
$$\left.\begin{aligned} f_k^{(i)} &= -\log_{10}(\|F(x^k)\|_\infty) \\ e_k^{(i)} &= -\log_{10}(\|x^k - \overset{*}{x}\|_\infty) \end{aligned}\right\}$$

$i=1$ H-algorithme

$i=2$ orthonormalisation de Gram-Schmidt

| $k$ | $f_k^{(1)}$ | $e_k^{(1)}$ |
|---|---|---|
| 1 | 0.702 | 0.538 |
| 2 | 0.677 | 0.440 |
| 3 | 1.346 | 1.057 |
| 4 | 1.547 | 1.266 |
| 5 | 1.751 | 1.454 |
| 6 | 2.384 | 2.079 |
| 7 | 3.283 | 2.974 |
| 8 | 5.655 | 5.333 |
| 9 | 7.611 | 7.291 |
| 10 | 8.926 | 8.619 |
| 11 | 10.606 | 10.300 |
| 12 | 13.993 | 13.685 |
| 13 | 17.761 | 16.983 |

| $f_k^{(2)}$ | $e_k^{(2)}$ |
|---|---|
| 0.702 | 0.538 |
| 0.677 | 0.440 |
| 1.346 | 1.057 |
| 1.547 | 1.266 |
| 1.751 | 1.454 |
| 2.384 | 2.079 |
| 3.283 | 2.974 |
| 5.655 | 5.333 |
| 7.611 | 7.292 |
| 8.926 | 8.620 |
| 10.618 | 10.313 |
| 13.998 | 13.690 |
| 17.761 | 16.900 |

<u>Exemple n°2</u> $F(x) = x - G(x)$

où $G$ est la fonction définie dans l'exemple n°5 du Chapitre I.

$x^{-6} = (0.1, 0.1, 0.1, 0.1, 0.1, 0.1)^T$ $\qquad x^i = G(x^{i-1}) \quad i = -5, -4, \dots, 0$

$\overset{*}{x} = (-1, 1, -1, 1, -1, 1)$

Nous avons obtenu les résultats suivants :

| $k$ | $f_k^{(1)}$ | $e_k^{(1)}$ |
|---|---|---|
| 1 | 1.548 | 1.033 |
| 2 | 0.498 | 0.491 |
| 3 | 1.730 | 1.168 |
| 4 | 1.951 | 1.950 |
| 5 | 2.816 | 2.563 |
| 6 | 3.524 | 2.973 |
| 7 | 4.407 | 3.635 |
| 8 | 5.076 | 4.398 |
| 9 | 6.518 | 5.903 |
| 10 | 7.149 | 6.464 |
| 11 | 10.735 | 9.839 |
| 12 | 13.394 | 12.644 |
| 13 | 15.004 | 14.253 |

| $f_k^{(2)}$ | $e_k^{(2)}$ |
|---|---|
| 1.548 | 1.033 |
| 0.498 | 0.491 |
| 1.730 | 1.168 |
| 1.951 | 1.950 |
| 2.816 | 2.563 |
| 3.524 | 2.973 |
| 4.407 | 3.635 |
| 5.076 | 4.398 |
| 6.518 | 5.903 |
| 7.149 | 6.464 |
| 10.735 | 9.839 |
| 13.394 | 12.644 |
| 15.003 | 14.252 |

**Exemple n°3 :** $F(x) = x - G(x)$

$$G_i(x) = x_i - h^{i}\left(i - \sum_{j=1}^{i} x_j + 0.3\sum_{j=i}^{p} (1-x_j)^2\right) \qquad i=1,\dots,p$$

$p = 10$ et $h = 0.1$

$$x_j^{10} = \begin{cases} 0.9 & \text{si } j \text{ est impair} \\ 1.1 & j \text{ est pair} \end{cases} \qquad \text{et } x^i = G(x^{i-1}) \quad i=-9,\dots,0$$

$$\overset{*}{x}_i = 1 \quad \forall i = 1,\dots,p$$

| $k$ | $f_k^{(1)}$ | $e_k^{(1)}$ |
|---|---|---|
| 1 | 1.288 | 0.717 |
| 2 | 1.663 | 1.186 |
| 3 | 1.407 | 0.948 |
| 4 | 1.732 | 1.322 |
| 5 | 0.894 | 0.417 |
| 6 | 2.004 | 1.083 |
| 7 | 1.390 | 0.894 |
| 8 | 2.368 | 1.643 |
| 9 | 1.884 | 1.336 |
| 10 | 2.662 | 1.824 |
| 11 | 2.893 | 1.891 |
| 12 | 3.115 | 2.114 |
| 13 | 3.138 | 2.137 |
| 14 | 3.079 | 2.077 |
| 15 | 3.080 | 2.079 |
| 16 | 2.764 | 1.762 |
| 17 | 2.835 | 1.833 |
| 18 | 5.005 | 4.005 |
| 19 | 3.860 | 2.860 |
| 20 | 6.573 | 5.573 |
| 21 | 7.148 | 6.148 |
| 22 | 8.842 | 7.842 |
| 23 | 11.517 | 10.517 |
| 24 | 14.481 | 13.481 |

| $f_k^{(2)}$ | $e_k^{(2)}$ |
|---|---|
| 1.288 | 0.717 |
| 1.662 | 1.185 |
| 1.406 | 0.947 |
| 1.731 | 1.323 |
| 0.891 | 0.414 |
| 2.015 | 1.089 |
| 1.383 | 0.887 |
| 2.365 | 1.641 |
| 1.890 | 1.343 |
| 2.661 | 1.824 |
| 2.893 | 1.891 |
| 3.113 | 2.112 |
| 3.135 | 2.134 |
| 3.078 | 2.077 |
| 3.078 | 2.077 |
| 2.902 | 1.900 |
| 4.169 | 3.168 |
| 5.159 | 4.159 |
| 5.771 | 4.771 |
| 7.346 | 6.346 |
| 7.965 | 6.965 |
| 11.655 | 10.655 |
| 14.802 | 13.802 |
| 17.217 | 16.215 |

## Exemple n°4

On reprend la fonction $G$ de l'exemple précédent avec $h = 10^{-2}$ et $p = 15$

$$x_j^{-15} = \begin{cases} 0.8 & j \text{ est impair} \\ 1.2 & j \text{ est pair} \end{cases} \quad \text{et} \quad x^i = G(x^{i-1}) \quad \text{pour } i = -14, \dots, 0$$

$$\overset{*}{x}_j = 1 \quad \text{pour } j = 1, \dots, 15$$

| $k$ | $f_k^{(1)}$ | $e_k^{(1)}$ |
|---|---|---|
| 20 | 2.997 | 1.064 |
| 21 | 2.970 | 1.049 |
| 22 | 3.000 | 1.116 |
| 23 | 2.493 | 0.555 |
| 24 | 3.579 | 1.814 |
| 25 | 3.399 | 1.393 |
| 26 | 3.707 | 1.704 |
| 27 | 4.349 | 2.348 |
| 28 | 5.975 | 4.221 |
| 29 | 5.669 | 3.697 |
| 30 | 6.064 | 4.388 |
| 31 | 6.446 | 4.509 |
| 32 | 6.522 | 4.522 |
| 33 | 6.500 | 4.500 |
| 34 | 6.508 | 4.508 |
| 35 | 6.433 | 4.483 |
| 36 | 6.938 | 4.938 |
| 37 | 9.121 | 7.121 |
| 38 | 13.308 | 11.308 |
| 39 | 16.557 | 14.558 |

| $f_k^{(2)}$ | $e_k^{(2)}$ |
|---|---|
| 3.611 | 1.678 |
| 3.337 | 1.584 |
| 3.924 | 2.130 |
| 3.388 | 1.520 |
| 4.363 | 2.535 |
| 4.352 | 2.545 |
| 3.874 | 1.954 |
| 3.357 | 1.443 |
| 3.402 | 1.622 |
| 3.392 | 1.611 |
| 3.592 | 1.681 |
| 3.243 | 1.340 |
| 4.325 | 2.659 |
| 4.313 | 2.313 |
| 4.624 | 2.624 |
| 4.584 | 2.585 |
| 4.634 | 2.634 |
| 4.094 | 2.185 |
| 5.354 | 3.434 |
| 5.497 | 3.268 |

## 5) La méthode d'Henrici

Cette méthode est aussi très ancienne, puisqu'elle a aussi été utilisée par Gauss [32], [53] pour résoudre des systèmes d'équations à deux inconnues. Pour des systèmes de taille quelconque, cette méthode a été obtenue indépendamment par Ludwig [54] et par Henrici [42]. Ces deux auteurs ont utilisé deux approches différentes et ont donné deux formulations distinctes. L'équivalence de ces deux approches a été signalée dans [60].

Posons $\ell = 0$ , $x^{k,0} = x^k$ et $x^{k,i} = G(x^{k,i-1})$ $i = 1, \dots, p$

la méthode de la sécante s'écrit:

$$x^{k+1} = x^k - H_k S_k^{-1} (G(x^k) - x^k)$$

où $H_k = (G(x^k) - x^k, \dots\dots, G^{(p)}(x^k) - G^{(p-1)}(x^k))$

et $S_k = (G^{(2)}(x^k) - 2G(x^k) + x^k, \dots\dots, G^{(p+1)}(x^k) - 2\,G^{(p)}(x^k) + G^{(p-1)}(x^k))$

avec $G^{(i)} = \underbrace{G \circ \dots \circ G}_{i \text{ fois}}$

### Remarque

Cette méthode nécessite $(p+1)$ évaluations de fonctions et est d'ordre 2 si l'on suppose que les matrices $H_k$ sont uniformément inversibles voir [60] p 373 (Resultat 11.3.5 pour la suite (10) qui définit la méthode d'Henrici)

Pour la mise en oeuvre de cette méthode, nous avons utilisé le H-algorithme ainsi que la méthode LU (avec pivot partiel) pour inverser les matrices $S_k$. Donnons quelques exemples numériques.

Soit à résoudre l''equation $f(x)=0$

avec

$$\begin{cases} f_1(x) = (3-x_1)x_1 + 1 - 2x_2 \\ f_p(x) = (3-x_p)x_p + 1 - 2x_{p-1} \\ f_i(x) = (3-x_i)x_i + 1 - x_{i-1} - 2x_{i+1} \qquad i=2,..,p-1 \end{cases}$$

Posons $G(x) = x - h f(x)$

et

$e_k^{(i)} = \| f(x_k)\|_2$ $\begin{cases} i=1 & \text{programmation avec le H-algorithme} \\ i=2 & \text{inversion de la matrice par la méthode de Gauss.} \end{cases}$

$\overset{\circ}{x}_j = -1$ pour $j=1,..,p$

pour $i=2$, nous avons calculer le conditionnement des matrices $S_k$.

$\chi_k = \mathrm{Cond}_\infty(S_k) = \|S_k\|_\infty \cdot \|S_k^{-1}\|_\infty$

Exemple n°1

$p=10$ , $h=10^{-1}$

| $k$ | $e_k^{(1)}$ | $e_k^{(2)}$ | $\chi_k$ |
|---|---|---|---|
| 1 | 0.122D-02 | 0.122D-02 | 0.389D+03 |
| 2 | 0.135D-09 | 0.103D-09 | 0.998D+09 |
| 3 | 0.823D-12 | 0.508D-12 | 0.407I+06 |
| 4 | 0.130D-13 | 0.182D-12 | 0.252D+05 |

Exemple n°2 $p = 20$ , $h = 0,1$

| $k$ | $e_k^{(1)}$ | $e_k^{(2)}$ | $\chi_k$ |
|---|---|---|---|
| 1 | 0.204D-05 | 0.224D-04 | 0.261D+16 |
| 2 | 0.120D-09 | 0.894D-10 | 0.432D+11 |
| 3 | 0.156D-12 | 0.682D-13 | 0.102D+07 |

Exemple n°3 $p=30$ , $h = 0.3$

| $k$ | $e_k^{(1)}$ | $e_k^{(2)}$ | $\chi_k$ |
|---|---|---|---|
| 1 | 0.189D+00 | 0.189D+00 | 0.127D+10 |
| 2 | 0.136D-03 | 0.136D-03 | 0.251D+10 |
| 3 | 0.389D-09 | 0.207D-05 | 0.253D+15 |
| 4 | 0.155D-12 | 0.258D-09 | 0.167D+16 |

Exemple n°4 $p=40$ , $h=0.3$

| $k$ | $e_k^{(1)}$ | $e_k^{(2)}$ | $\chi_k$ |
|---|---|---|---|
| 1 | 0.143D+00 | 0.143D+00 | 0.141D+10 |
| 2 | 0.270D-02 | 0.271D-02 | 0.549D+12 |
| 3 | 0.303D-05 | 0.686D-05 | 0.465D+17 |
| 4 | 0.181D-12 | 0.643D-08 | 0.319D+18 |
| 5 | 0.826D-13 | 0.254D-11 | 0.178D+17 |

## b. Première généralisation de la méthode Regula-Falsi:

Prenons $\ell = 1$ dans l'algorithme $C_I$, nous avons donc:

$$A_{i,1} : \mathbb{C}^p \times \mathbb{C}^p \longrightarrow \mathbb{C}^p \qquad i=0,\dots p.$$
$$B_{i,1} : \mathbb{C}^p \times \mathbb{C}^p \longrightarrow \mathbb{C}^p$$

Soit $(y,z) \in \mathbb{C}^p \times \mathbb{C}^p$, notons $\begin{cases} G^{(0)} : & G^{(0)}(x) = x \\ G^{(i)}(x) = G(G^{(i-1)}(x)) \end{cases}$

et posons

$$\begin{cases} A_{i,1}(y,z) = G^{(i)}(y) \\ \text{et } B_{i,1}(y,z) = G^{(i)}(z) \end{cases} \qquad i=0,\dots,p.$$

Avec ces choix l'algorithme $C_I$ devient:

$$\begin{cases} x^{-1}, \overset{\circ}{x} \text{ donnés} \\ \underline{\text{itération } k+1}: \quad t_1^{(i)} = G^{(i)}(x^{k-1}) \\ \qquad\qquad \text{et } t_2^{(i)} = G^{(i)}(x^k) \qquad i=0,\dots p \\ x^{k+1} = x^{k-1} - \Delta T_1^{(0)} (\Delta T_2^{(0)} - \Delta T_1^{(0)})^{-1} (x^k - x^{k-1}) \end{cases}$$

or

$$\Delta T_1^{(0)} = (G(x^{k-1}) - x^{k-1}, \dots\dots, G^{(p)}(x^{k-1}) - G^{(p-1)}(x^{k-1}))$$

et

$$\Delta T_2^{(0)} = (G(x^k) - x^k, \dots\dots, G^{(p)}(x^k) - G^{(p-1)}(x^{k-1})).$$

Notons $E(x) = (x - G(x), \dots\dots, G^{(p-1)}(x) - G^{(p)}(x))$

et comme $F(x) = x - G(x)$

$$\underline{E(x) = (F(x), \dots\dots, F(G^{(p-1)}(x))}$$

La première généralisation de la méthode Regula-Falsi est définie par :

$$\left[\begin{array}{l} x^{-1},\ \overset{\circ}{x} \text{ donnés} \\ \text{pour } k \geqslant 0 \\ \quad x^{k+1} = x^{k-1} - E(x^{k-1})\left(E(x^k) - E(x^{k-1})\right)^{-1}(x^k - x^{k-1}) \end{array}\right.$$

Quand $p=1$ $E(x) = F(x)$ et on retrouve la méthode Regula-Falsi dans le cas scalaire.

- La première itération nécessite $2p$ évaluations de fonctions, les autres n'en nécessitent que $p$.
- L'ordre de cette méthode semble être $\frac{1+\sqrt{5}}{2}$.

Exemple numérique :

On veut résoudre le système d'équations non linéaires suivant :

$$\begin{cases} x_1 = 2x_1x_2 - x_1^2 \\ x_2 = x_1^2 - 2{,}9(x_2 - 1) \end{cases}$$

$\overset{*}{x} = (1,1)^T$ Posons $e^k = \|x^k - \overset{*}{x}\|_2$

| $k$ | $x_1^k$ | $x_2^k$ | $e^k$ | $\frac{e^k}{e^{k-1}}$ |
|---|---|---|---|---|
| -1 | 0.105000000000000D+01 | 0.950000000000000D+00 | | |
| 0 | 0.160000000000000D+01 | 0.940000000000000D+00 | 0.60E+00 | 0.85E+01 |
| 1 | 0.102435408463708D+01 | 0.970599269573722D+00 | 0.38E-01 | 0.63E-01 |
| 2 | 0.101329941741878D+01 | 0.983416623555611D+00 | 0.21E-01 | 0.56E+00 |
| 3 | 0.100309989906124D+01 | 0.999629241755710D+00 | 0.31E-02 | 0.15E+00 |
| 4 | 0.100240483821892D+01 | 0.100113357411628D+01 | 0.27E-02 | 0.85E+00 |
| 5 | 0.100207030054171D+01 | 0.100105894798874D+01 | 0.23E-02 | 0.87E+00 |
| 6 | 0.100008972030230D+01 | 0.100004499475457D+01 | 0.10E-03 | 0.43E-01 |
| 7 | 0.100000384849449D+01 | 0.100000161210390D+01 | 0.42E-05 | 0.42E-01 |
| 8 | 0.100000000839374D+01 | 0.100000000064220D+01 | 0.84E-08 | 0.20E-02 |
| 9 | 0.100000000000095D+01 | 0.999999999999378D+00 | 0.11E-11 | 0.14E-03 |
| 10 | 0.100000000000003D+01 | 0.100000000000000D+01 | 0.28E-13 | |

# IV Seconde transformation composite vectorielle

## 1- Définition

Appliquons la procédure $\theta^{(2)}$ à $c_n = t_1^{(n)} + (t_2^{(n)} - t_1^{(n)})$ nous obtenons :

$$\boxed{s_{p,n} = \theta^{(2)}(c_n) = t_1^{(n)} - (T_2^{(n)} - T_1^{(n)})(\Delta T_2^{(n)} - \Delta T_1^{(n)})^{-1}\,\Delta t_1^{(n)}}$$

Si $t_1^{(n)} = s_n$ et $t_2^{(n)} = s_{n+1}$ alors $s_{p,n} = h_n$.

Remarque : Cette transformation peut être obtenue d'une autre façon [20] :

Dans le cas scalaire : le point fixe de la fonction $f$ définie par :

$$\begin{cases} f : \mathbb{C} \longrightarrow \mathbb{C} \\ f(x) = ax + b \end{cases} \quad \text{et } a \neq 1$$

et qui vérifie

$$\begin{cases} f(t_1^{(n)}) = t_1^{(n+1)} \\ f(t_2^{(n)}) = t_2^{(n+1)} \end{cases}$$

est $T_n = t_1^{(n)} - (t_2^{(n)} - t_1^{(n)})(\Delta t_2^{(n)} - \Delta t_1^{(n)})^{-1}\Delta t_1^{(n)}$. Tétant la transformation composite de rang deux (cas scalaire)

Dans le cas vectoriel :

$s_{p,n}$ est le point fixe de la fonction $F$ définie par :

$$F : \mathbb{C}^p \longrightarrow \mathbb{C}^p$$

$$F(x) = Ax + b \text{ où } A \in L(\mathbb{C}^p) \text{ et } b \in \mathbb{C}^p$$

$$A - I \text{ inversible}$$

et vérifiant

$$\begin{cases} F(t_1^{(n+i)}) = t_1^{(n+i+1)} \\ F(t_2^{(n+i)}) = t_2^{(n+i+1)} \end{cases} \quad \text{pour } i = 0, \ldots, p-1.$$

Comme $s_{p,m}$ a été obtenu par utilisation de la procédure $\theta^{(2)}$, on peut donc l'écrire comme un rapport de deux déterminants :

$$s_{p,m} = \begin{vmatrix} t_1^{(m)} & t_2^{(m)}-t_1^{(m)} & \cdots & t_2^{(m+p-1)}-t_1^{(m+p-1)} \\ (\Delta t_1^{(m)})_1 & (\Delta t_2^{(m)}-\Delta t_1^{(m)})_1 & \cdots & (\Delta t_2^{(m+p-1)}-\Delta t_1^{(m+p-1)})_1 \\ \cdots & \cdots & \cdots & \cdots \\ (\Delta t_1^{(m)})_p & (\Delta t_2^{(m)}-\Delta t_1^{(m)})_1 & \cdots & (\Delta t_2^{(m+p-1)}-\Delta t_1^{(m+p-1)})_p \end{vmatrix} \Big/ \det(\Delta T_2^{(m)}-\Delta T_1^{(m)})$$

Nous allons maintenant donner deux algorithmes récursifs pour calculer $s_{p,m}$ quand $m$ et $p$ sont fixés.

Posons $y = \begin{pmatrix} t_1^{(m)} \\ t_1^{(m+1)} \end{pmatrix}$ et $x^i = \begin{pmatrix} t_2^{(m+i-1)} - t_1^{(m+i-1)} \\ t_2^{(m+i)} - t_1^{(m+i)} \end{pmatrix}$ $i=1,\ldots,p$

$y \in \mathbb{C}^{2p}$, $x^i \in \mathbb{C}^{2p}$.

Soient $z^i$ des vecteurs de $\mathbb{C}^{2p}$ tels que :

$$(z^i, v) = (v_2 - v_1)_i \qquad \forall v \in \mathbb{C}^{2p} \quad v=(v_1,v_2)^T$$
$$v_1 \in \mathbb{C}^p \text{ et } v_2 \in \mathbb{C}^p$$

Donc
$$\begin{cases} z_j^i = 0 & \text{si } j\neq i \text{ ou } j\neq i+p \\ z_i^i = \text{ et } z_{i+p} = 1. \end{cases}$$

Avec ces notations $s_{p,n}$ peut être calculé avec le R.P.A [16] qui est dans ce cas:

$$\boxed{\begin{aligned} & E_0 = y \qquad \text{et} \qquad g_{0,i} = x^i \qquad \text{pour } i=1,\ldots,p \\ & E_k = E_{k-1} - \frac{(E_{k-1})_{p+k} - (E_{k-1})_k}{(g_{k-1,k})_{p+k} - (g_{k-1,k})_k}\, g_{k-1,k} \qquad k=1,\ldots,p \\ & g_{k,i} = g_{k-1,i} - \frac{(g_{k-1,i})_{p+k} - (g_{k-1,i})_k}{(g_{k-1,k})_{p+k} - (g_{k-1,k})_k}\, g_{k-1,k} \qquad i>k \end{aligned}}$$

$E_k \in \mathbb{C}^{2p}$ et $s_{n,p} \in \mathbb{C}^p$. On a le résultat suivant :

$$(s_{n,p})_i = (E_p)_i \qquad \text{pour } i=1,\dots p.$$

On peut aussi utiliser le C.R.P.A [16]

$$\begin{cases} e_0^{(0)} = y & e_0^{(i)} = x^i \quad i=1,\dots p \\ e_k^{(i)} = e_{k-1}^{(i)} - \dfrac{(e_{k-1}^{(i)})_{p+k} - (e_{k-1}^{(i)})_k}{(e_{k-1}^{(i+1)})_{p+k} - (e_{k-1}^{(i+1)})_k} \, e_{k-1}^{(i+1)} & \text{pour } \begin{cases} k=1,\dots p \\ \text{et} \\ i=0,\dots p-k \end{cases} \end{cases}$$

Comme pour le R.P.A nous avons :

$$(s_{n,p})_i = (e_p^{(0)})_i \qquad i=1,\dots p$$

2 – Etude de la seconde transformation composite.

Commençons par caractériser son noyau :

théorème 9

> Supposons que les matrices $T_2^{(n)} - T_1^{(n)}$ et $\Delta T_2^{(n)} - \Delta T_1^{(n)}$ soient inversibles pour tout $n$.
>
> Une condition nécessaire et suffisante pour que $s_{p,n} = s$ $\forall n$ est qu'il existe $p$ nombres complexes $x_0, x_1, \dots, x_{p-1}$ tels que :
>
> $$t_1^{(n)} - s = \sum_{i=0}^{p-1} x_i \left( t_2^{(n+i)} - t_1^{(n+i)} \right) \qquad \forall n$$

preuve : On pose $a_n = t_1^{(n)}$ et $b_n = t_2^{(n)} - t_1^{(n)}$

et on applique le théorème 4. ■

Ce théorème généralise le théorème 3 du chapitre I, ainsi que le théorème 1 (du cas scalaire au cas vectoriel).

Propriété 2 : Soit $\{s_{p,n}\}$ la suite obtenue à partir de $\{t_1^{(n)}\}$ et $\{t_2^{(n)}\}$ et soit $\{\tilde{s}_{p,n}\}$ la suite obtenue à partir de $\{A t_1^{(n)}+b\}$ et $\{A t_2^{(n)}+b\}$ où $A$ est une matrice carrée complexe d'ordre $p$ inversible et $b$ un vecteur de $\mathbb{C}^p$ alors :

$$\tilde{s}_{p,n} = A\, s_{p,n} + b$$

preuve :

$$\tilde{s}_{p,n} = f_2\,(A t_1^{(n)} + b,\ A(t_2^{(n)} - t_1^{(n)}))$$

$$= A\, f_2(t_1^{(n)}, t_2^{(n)} - t_1^{(n)}) + b = A\, s_{p,n} + b$$

d'après la propriété 1. ■

Comme théorème de convergence nous avons le résultat suivant :

théorème 10

si $\lim_n t_1^{(n)} = s$ et pour tout $n$ la matrice $T_2^{(n)} - T_1^{(n)}$ est inversible et si

$$B_n = (T_2^{(n+1)} - T_1^{(n+1)})\,(T_2^{(n)} - T_1^{(n)})^{-1}$$

et $\exists\, \gamma > 0$, $\exists\, N > 0$, $\forall n > N$ $\|B_n\| < \gamma < 1$

alors 1) $\Delta T_2^{(n)} - \Delta T_1^{(n)}$ est inversible pour $n > N$

2) $\lim_n s_{p,n} = s$

preuve :

On applique le lemme 2 du chapitre I avec $A = I$ et $C = I - B_n$

$\|I\| = 1 = \alpha$.

Donc $I - B_n$ est inversible $\forall n > N$

et par conséquent $\Delta T_2^{(n)} - \Delta T_1^{(n)}$ est inversible $\forall n > N$,

et de plus $\|(I - B_n)^{-1}\| \leq \dfrac{1}{1-\gamma}$

Comme $\quad s_{p,n} - s = t_1^{(n)} - s + (I - B_n)^{-1} \Delta t_1^{(n)}$

on a

$$\| s_{p,n} - s \| \leq \| t_1^{(n)} - s \| + \frac{1}{1-\gamma} \| \Delta t_1^{(n)} \|$$

et donc $\lim_n s_{p,n} = s$. ■

Pour démontrer un résultat d'accélération de la convergence, nous avons besoin du théorème suivant :

théorème 11

> Si $\quad t_2^{(n)} - s = (A + A_n)(t_1^{(n)} - s)$ avec $\lim_m A_n = 0$ et $(A, A_n) \in [L(\mathbb{C}^p)]^2$
>
> et si
>
> $t_1^{(n+1)} - s = (B + B_n)(t_1^{(n)} - s)$ avec $\lim_m B_n = 0$ et $(B, B_n) \in [L(\mathbb{C}^p)]^2$
>
> si de plus $(A - I)$ est inversible alors :
>
> $\exists N, \forall n > N \quad t_2^{(n+1)} - t_1^{(n+1)} = [(A-I)B(A-I)^{-1} + C_n](t_2^{(n)} - t_1^{(n)})$
>
> avec $\lim_m C_n = 0$

preuve : Par hypothèse on a :

$$t_2^{(n+1)} - s = (A + A_{n+1})(B + B_n)(t_1^{(n)} - s)$$

et $t_1^{(n+1)} - s = (B + B_n)(t_1^{(n)} - s)$,

En faisant la différence de ces deux égalités on obtient :

$$t_2^{(n+1)} - t_1^{(n+1)} = (A - I + A_{n+1})(B + B_n)(t_1^{(n)} - s) \quad *$$

D'autre par $\quad t_2^{(n)} - t_1^{(n)} = (A - I + A_n)(t_1^{(n)} - s)$

où encore

$$t_2^{(n)} - t_1^{(n)} = (A - I)(I + F_n)(t_1^{(n)} - s)$$

avec $F_n = (A - I)^{-1} A_m$ et donc $\lim_n F_n = 0$.

$\exists N'$, $\forall n > N'$ $(I+F_n)$ est inversible,

Posons $G_n = I-(I+F_n)^{-1}$ ; on a aussi $\lim\limits_n G_n = 0$.

Il en résulte que

$$t_1^{(n)} - s = (I-G_n)(A-I)^{-1}(t_2^{(n)} - t_1^{(n)}),$$

d'où en reportant dans $*$ on obtient:

$$t_2^{(n+1)} - t_1^{(n+1)} = (A-I+A_{n+1})(B+B_n)(I-G_n)(A-I)^{-1}(t_2^{(n)} - t_1^{(n)})$$

$$= [(A-I)B(A-I)^{-1} + C_n](t_2^{(n)} - t_1^{(n)})$$

et $\lim\limits_n C_n = 0$ ∎

Ce théorème généralise aussi le théorème 5 du chapitre I; pour le retrouver il suffit de poser:

$$t_1^{(n)} = s_n \text{ et } t_2^{(n)} = s_{n+1} \quad \text{donc } B = A \text{ et } A_n = B_n$$

$$\text{et } (A-I)B(A-I)^{-1} = B = A.$$

En ce qui concerne l'accélération de la convergence on a :

**théorème 19**

Si $t_2^{(n)} - s = (A+A_n)(t_1^{(n)} - s)$ où $\lim\limits_n A_n = 0$ et $(A, A_n) \in (L(\mathbb{C}^p))^2$

et $t_1^{(n+1)} - s = (B+B_n)(t_1^{(n)} - s)$ où $\lim\limits_n B_n = 0$ et $(B, B_n) \in (L(\mathbb{C}^p))^2$.

Si de plus $\rho(A) < 1$, $\rho(B) < 1$ et A et B commutent,

et si les matrices $T_2^{(n)} - T_1^{(n)}$ sont uniformément inversibles, alors

1) $s_{p,n}$ est définie pour $n$ assez grand.

2) $\| s_{p,n} - s \| = o(\| t_1^{(n)} - s \|)$

Si de plus A est inversible alors $\| s_{p,n} - s \| = o(\| t_2^{(n)} - s \|)$.

preuve:

par hypothèse et d'après le théorème précédent on a:

$\exists N', \forall n > N'$ . $$t_2^{(n+1)} - t_1^{(n+1)} = ((A-I)B(A-I)^{-1} + C_n)\,(t_2^{(n)} - t_1^{(n)})$$

et donc:

$$\Delta t_2^{(n)} - \Delta t_1^{(n)} = ((A-I)(B-I)(A-I)^{-1} + C_n)\,(t_2^{(n)} - t_1^{(n)}).$$

Ecrivons cette relation pour $n+1, n+2, \ldots, n+p-1$, on obtient:

$$\Delta T_2^{(n)} - \Delta T_1^{(n)} = (A-I)(B-I)(A-I)^{-1}(I+D_n)\,(T_2^{(n)} - T_1^{(n)}) \qquad *$$

avec

$$D_n = (A-I)(B-I)^{-1}(A-I)^{-1}\left(C_n(t_2^{(n)} - t_1^{(n)}), \ldots, C_{n+p-1}(t_2^{(n+p-1)} - t_1^{(n+p-1)})\right)(T_2^{(n)} - T_1^{(n)})^{-1}$$

D'après les lemmes 2 et 3 du chapitre I:

$\lim_n D_n = 0$ et $\exists N'', \forall n > N''$ $(I+D_n)$ est inversible.

Donc $\Delta T_2^{(n)} - \Delta T_1^{(n)}$ est inversible $\forall n > \max(N', N'') = N$

et par conséquent:

$s_{p,n}$ est bien définie $\forall n > N$ .

Par définition

$$s_{p,n} - s = t_1^{(n)} - s - (T_2^{(n)} - T_1^{(n)})\,(\Delta T_2^{(n)} - \Delta T_1^{(n)})^{-1}\,\Delta t_1^{(n)}.$$

En reportant $*$ dans cette égalité on obtient:

$$s_{p,n} - s = t_1^{(n)} - s - (I - F_n)(A-I)(B-I)^{-1}(A-I)^{-1} \qquad \forall n > N$$

avec
$F_n = I - (I+D_n)^{-1}$ et $\lim_n F_n = 0$

Comme par ailleurs

$t_1^{(n+1)} - s = (B + B_n)(t_1^{(n)} - s)$, d'où immédiatement.

$$\Delta t_1^{(n)} = (B - I + B_n)(t_1^{(n)} - s),$$

et par conséquent

$$s_{p,n} - s = t_1^{(n)} - s - (I - F_n)(A - I)(B - I)^{-1}(A - I)^{-1}(B - I + B_n)(t_1^{(n)} - s)$$

Comme $AB = BA$

donc $(A - I)(B - I)^{-1} = (B - I)^{-1}(A - I)$,

ce qui donne

$$s_{p,n} - s = t_1^{(n)} - s - (I - G_n)(t_1^{(n)} - s) \quad \text{où } \lim_n G_n = 0$$
$$= G_n (t_1^{(n)} - s).$$

D'où $\|s_{p,n} - s\| = o(\|t_1^{(n)} - s\|)$

si de plus $A$ est inversible, on a

$$t_1^{(n)} - s = (I + \tilde{A}_n) A^{-1} (t_2^{(n)} - s) \quad \text{et } \tilde{A}_n = A^{-1} A_n .$$

Alors

$$s_{p,n} - s = G_n (I + \tilde{A}_n) A^{-1} (t_2^{(n)} - s)$$

ce qui donne

$$\|s_{p,n} - s\| = o(\|t_2^{(n)} - s\|)$$

■

<u>Remarque</u>:

Comme pour $t_{p,n}$ le calcul de $s_{p,n}$ peut se faire avec la méthode d'orthonormalisation de "Gram-Schmidt et sa mise à jour" étudiée au chapitre I. Il suffit de poser $A_n = \Delta T_2^{(n)} - \Delta T_1^{(n)}$ et donc le calcul de $s_{0,p}$ se fait en $O(p^3)$ opérations arithmétiques. Connaissant $s_{p,n}$ le calcul de $s_{p,n+1}$ se fait en $O(p^2)$ opérations arithmétiques.

Nous allons maintenant examiner l'utilisation de cette transformation pour résoudre les systèmes d'équations non linéaires.

## 3) Application à la résolution des systèmes d'équations non linéaires.

On veut résoudre le système d'équations non linéaires :

$$F(x)=0$$

ou

$$G(x)=x$$

On suppose qu'il existe $\overset{*}{x}$ tel que $F(\overset{*}{x})=0$, $G(\overset{*}{x})=\overset{*}{x}$

$$F : \mathbb{C}^p \longrightarrow \mathbb{C}^p \quad \text{et} \quad G : \mathbb{C}^p \longrightarrow \mathbb{C}^p$$

On se donne un entier $\ell$ et des fonctions arbitraires $A_{i,\ell}$ et $B_{i,\ell}$

avec

$$A_{i,\ell} : (\mathbb{C}^p)^{\ell+1} \longrightarrow \mathbb{C}^p$$

et

$$B_{i,\ell} : (\mathbb{C}^p)^{\ell+1} \longrightarrow \mathbb{C}^p$$

et on propose l'algorithme général suivant :

$C_{II}$

Initialisation $\ell$ un entier fixé

$x^{-\ell}, \dots, \overset{\circ}{x}$ donnés

Itération $k+1$

$$t_1^{(i)} = A_{i,\ell}(x^{k-\ell}, \dots, x^k)$$

$$t_2^{(i)} = B_{i,\ell}(x^{k-\ell}, \dots, x^k) \qquad \text{pour } i=0,\dots,p$$

et on pose

$$x^{k+1} = s_{p,0}$$

Supposons que $A_{i,\ell} \not\equiv B_{i,\ell}$ pour $i=0,\dots,p$

Montrons que cet algorithme généralise la méthode d'Henrici.

### a - Méthode d'Henrici

On note $G^{(0)} = Id$ . $Id : \mathbb{C}^p \longrightarrow \mathbb{C}^p$ $\quad Id(x)=x$

$$\text{et} \quad G^{(i)} = \underbrace{G \circ G \circ \dots \circ G}_{i \text{ fois}} \qquad i \geq 1 .$$

et on pose $\ell = 0$,

$$A_{i,0} : \mathbb{C}^p \longrightarrow \mathbb{C}^p \quad \text{et} \quad B_{i,0} : \mathbb{C}^p \longrightarrow \mathbb{C}^p$$

et

$$A_{i,0} = G^{(i)} \quad \text{et} \quad B_{i,0} = G^{(i+1)} \quad \text{pour } i=0,\dots,p$$

La ieme colonne de la matrice $T_2^{(0)} - T_1^{(0)}$ est le vecteur :

$$G^{(i)}(x^k) - G^{(i-1)}(x^k) \quad \text{pour } i=1,\dots p.$$

De même la ième colonne de la matrice $\Delta T_2^{(0)} - \Delta T_1^{(0)}$ est le vecteur :

$$G^{(i+1)}(x^k) - 2\, G^{(i)}(x^k) + G^{(i-1)}(x^k) \quad i=1,\dots p\ .$$

Posons $y^{(0)} = x^k$ et $y^{(i)} = G^{(i)}(x^k)$.

Avec ces notations

$$T_2^{(0)} - T_1^{(0)} = (y^{(1)} - y^{(0)}, \dots, y^{(p)} - y^{(p-1)}).$$

Notons $H_k$ cette matrice et $S_k$ la matrice $\Delta T_2^{(0)} - \Delta T_1^{(0)}$.

$x^{k+1}$ s'ecrit comme suit

$$x^{k+1} = y^{(0)} - H_k S^{-1} (G(y) - y)$$

On obtient la méthode d'Henrici voir [60] p200, NR 7-2-3.

Comme on l'a vu, cette méthode peut être mise en oeuvre par le H-algorithme et par le C.R.P.A.

Si on utilise le H-algorithme, le coût est $\frac{5}{2}p^3 + O(p^2)$ opérations arithmétiques alors qu'avec le C.R.P.A le coût est $2p^3 + O(p^2)$ opérations arithmétiques.

Du point ~~de vue~~ de la stabilité numérique, il semble que le H-algorithme soit meilleur que le C.R.P.A.

## b- Seconde généralisation de la méthode Regula-Falsi

Prenons dans l'algorithme $\mathcal{G}_{II}$, $\ell = 1$ on a donc :

$$A_{i,1} : \mathbb{C}^p \times \mathbb{C}^p \longrightarrow \mathbb{C}^p$$

et

$$B_{i,1} : \mathbb{C}^p \times \mathbb{C}^p \longrightarrow \mathbb{C}^p$$

$\forall (y,z) \in \mathbb{C}^p \times \mathbb{C}^p$ posons $\begin{cases} A_{i,1}(y,z) = G^{(i)}(z) \\ B_{i,1}(y,z) = G^{(i)}(y) \end{cases}$ pour $i = 0, \ldots, p$

Avec ces définitions l'algorithme $\mathcal{G}_{II}$ devient :

$$\left[\begin{array}{l} x^{-1} \text{ et } x^0 \text{ deux vecteurs donnés} \\ \text{itération } k+1. \quad t_1^{(0)} = x^k \text{ et } t_2^{(0)} = x^{k-1} \\ \qquad t_1^{(i+1)} = G(t_1^{(i)}) \\ \qquad \text{et } t_2^{(i+1)} = G(t_2^{(i)}) \qquad i = 0, \ldots, p-1 \\ \qquad \text{et } x^{k+1} = s_{p,0} \end{array}\right.$$

L'écriture de cet algorithme se simplifie si l'on pose :

$$H(x,y) = \left(F(y) - F(x), F(G(y)) - F(G(x)), \ldots\ldots, F(G^{(p-1)}(y)) - F(G^{(p-1)}(x))\right)$$

et

$$C(x,y) = \left(y - x, G(y) - G(x), \ldots, G^{(p-1)}(y) - G^{(p-1)}(x)\right)$$

$$F(x) = x - G(x) .$$

En effet avec ces notations, on a :

$$x^{k+1} = x^k - C(x^k, x^{k-1})\, H(x^k, x^{k-1})^{-1} F(x^k) \qquad P_I$$

Quand $p = 1$ on a $C(x^k, x^{k-1}) = x^k - x^{k-1}$ et $H(x^k, x^{k-1}) = F(x^k) - F(x^{k-1})$

On retrouve la méthode Regula-Falsi dans le cas scalaire.

Nous allons montrer que sous certaines hypothèses cette méthode est d'ordre $\frac{1+\sqrt{5}}{2}$ et pour cela nous avons besoin des résultats suivant :

$$\text{Soit} \quad F : D \subset \mathbb{C}^p \longrightarrow \mathbb{C}^p .$$

$D$ un ouvert de $\mathbb{C}^p$.

Lemme 1

Si $F$ est continûment différentiable dans un convexe $D_0$ de $D$ et si $\exists\, \alpha$, $\forall u \in D_0$ et $\forall v \in D_0$ on a :

$$\| F'(u) - F'(v) \| \leq \alpha \| u - v \|$$

alors

$$\forall x, y \in D_0 \quad \| F(y) - F(x) - F'(x)(y-x) \| \leq \frac{\alpha}{2} \| y - x \|^2$$

voir [60] Résultat 3.2.12 p.73.

Posons

$$C_1(x,y) = \left( \frac{y-x}{\|y-x\|_2}, \quad \cdots \cdots , \frac{G^{(p-1)}(y) - G^{(p-1)}(x)}{\| G^{(p-1)}(y) - G^{(p-1)}(x) \|_2} \right)$$

Lemme 2

Si $F$ est continûment différentiable dans un convexe $D_0$ de $D$ et si $\exists\, \alpha$, $\forall u \in D_0$ et $\forall v \in D_0$ $\quad \| F'(x) - F'(y) \|_2 \leq \alpha \| x - y \|_2$.

Si de plus $\forall x \in D_0$ alors $G^{(i)}(x) \in D_0$ $\forall i = 1, \ldots p-1$

et $\exists\, \gamma > 0$ $\quad \forall (x,y) \in D_0 \times D_0$ $\quad \| C_1(x,y) \|_2 \leq \gamma$

alors $\forall (x,y) \in D_0 \times D_0$ on a :

$$\| H(x,y)\, C(x,y)^{-1} - F'(x) \|_2 \leq \frac{1}{2} \alpha \gamma \left( \sum_{i=0}^{p-1} \| G^{(i)}(x) - G^{(i)}(y) \|_2^2 \right)^{1/2}$$

preuve : Posons $\Gamma(h) = F(x+h) - F(x) - F'(x) h \quad , \forall h \in \mathbb{C}^p$

D'après le lemme 1 $\|\Gamma(h)\|_2 \leq \frac{1}{2}\alpha\|h\|^2$ si $x \in D_0$ et $x+h \in D_0$.

$$H(x,y)\,C(x,y)^{-1} - F'(x) = \left[H(x,y) - F'(x)\,C(x,y)\right] C(x,y)^{-1}$$

$$= \left(\Gamma(y-x), \dots, \Gamma\left(G^{(p-1)}(y) - G^{(p-1)}(x)\right)\right) C(x,y)^{-1}$$

$$= \left(\frac{\Gamma(y-x)}{\|y-x\|_2}, \dots, \frac{\Gamma\left(G^{(p-1)}(y) - G^{(p-1)}(x)\right)}{\|G^{(p-1)}(y) - G^{(p-1)}(x)\|_2}\right) C_1(x,y)^{-1}$$

et donc $\|H(x,y)\,C(x,y)^{-1} - F'(x)\|_2 \leq \gamma \left\| \left(\frac{\Gamma(y-x)}{\|y-x\|_2}, \dots, \frac{\Gamma\left(G^{(p-1)}(y) - G^{(p-1)}(x)\right)}{\|G^{(p-1)}(y) - G^{(p-1)}(x)\|_2}\right)\right\|_2$

si $A$ est une matrice de $L(\mathbb{C}^p)$ on a vu que si

$A = (a^1, \dots, a^p)$ $\quad \|A\|_2 \leq \|A\|_F = \left(\sum_{i=1}^{p} \|a^i\|^2\right)^{1/2}$

d'où le résultat cherché ▨

<u>Remarque</u>:

Il y'a équivalence entre les deux propositions suivantes

1) $\exists\, \gamma > 0 : \quad \|C_1(x,y)^{-1}\|_2 \leq \gamma \quad \forall (x,y) \in D_0 \times D_0$

2) la matrice $C(x,y)$ est uniformément inversible $\forall (x,y) \in D_0 \times D_0$.

## <u>théorème 13</u>

Soit $F : D \subset \mathbb{C}^p \longrightarrow \mathbb{C}^p$. $F$ est continûment différentiable dans un ouvert $D$. On pose $G(x) = x - F(x)$ et on suppose que:

a) $\exists\, \overset{*}{x} : F(\overset{*}{x}) = 0$

b) $F'(\overset{*}{x})$ est inversible

c) $\forall (x,y) \in D \times D \quad \|F'(x) - F'(y)\|_2 \leq \alpha \|x-y\|_2$.

On note $S(\overset{*}{x}, \beta) = \{ x \in \mathbb{C}^p \mid \|x - \overset{*}{x}\|_2 \leq \beta \}$ et l'on suppose de plus que

d) $\forall (x,y) \in S(\overset{*}{x}, \beta) \quad \|G(x) - G(y)\|_2 \leq \|x-y\|_2$

e) $\exists\, \gamma > 0 \quad \forall x,y \in S(\overset{*}{x}, \beta) \quad \|C_1(x,y)\|_2 \leq \gamma$

4) $\|(F'(\overset{*}{x}))^{-1}\|_2\, \alpha\, \beta\, (\gamma\sqrt{p}+\frac{3}{2}) < \frac{1}{2}$

Si $x^{-1} \in S(\overset{*}{x},\beta)$ et $x \in S(\overset{*}{x},\beta)$ alors :

1°) . La suite $\{x^k\}$ définie par $(P_1)$ est bien définie

. $x^k \in S(\overset{*}{x},\beta)$ et $\lim_k x^k = \overset{*}{x}$

2°) La suite $\{x^k\}$ est d'ordre $(1+\sqrt{5})/2$

Preuve : On pose $D_0 = S(\overset{*}{x},\beta)$

si $x \in D_0$ alors $G^{(i)}(x) \in D_0$ pour $i=1,..p$

en effet : $\| G(x) - \overset{*}{x} \|_2 = \| G(x) - G(\overset{*}{x})\|_2 \leq \|x-\overset{*}{x}\|_2 \leq \beta$

et par récurrence $\| G^{(i)}(x) - \overset{*}{x}\|_2 \leq \beta$. $\forall x \in D_0$

D'après le lemme 2, $\forall x \in D_0$ et $\forall y \in D_0$

$$\|H(x,y)\ C(x,y)^{-1} - F'(x)\|_2 \leqslant \alpha\frac{\gamma}{2}\Big(\sum_{i=0}^{p-1} \| G^{(i)}(x) - G^{(i)}(y)\|_2^2\Big)^{1/2}$$

Comme $\| G^{(i)}(x) - G^{(i)}(y)\|_2 \leqslant \|x-\overset{*}{x}\|_2 + \|y-\overset{*}{x}\|_2$

on a

$$\|H(x,y)\ C(x,y)^{-1} - F'(x)\|_2 \leqslant \alpha\frac{\gamma}{2}\sqrt{p}\ \left(\|x-\overset{*}{x}\|_2 + \|y-\overset{*}{x}\|_2\right).$$

De plus

$$\|H(x,y)\ C(x,y)^{-1} - F'(\overset{*}{x})\|_2 \leq \|H(x,y)\ C(x,y)^{-1} - F'(x)\|_2 + \|F'(x) - F'(\overset{*}{x})\|_2$$

et par conséquent

$$\|H(x,y)\ C(x,y)^{-1} - F'(\overset{*}{x})\|_2 \leq \beta\, \alpha\, (\gamma\sqrt{p}+1) = \theta$$

Posons $a = \|F'(\overset{*}{x})^{-1}\|_2 = \|\{I - G'(\overset{*}{x})\}^{-1}\|_2$

et comme par hypothèse $a\, \alpha\, \beta\, (\gamma\sqrt{p}+\frac{3}{2}) < \frac{1}{2}$.

d'après le lemme 2 du chapitre I, la matrice

$\mathcal{T} = H(x,y)\ C(x,y)^{-1}$ est inversible ; de plus

$$\| \mathcal{T}^{-1}\|_2 \leq \frac{a}{1-a\theta} \leqslant \frac{a}{1-h} \qquad \forall (x,y) \in D_0 \times D_0$$

avec $h = a(\theta + \frac{\alpha\beta}{2})$ et $h < \frac{1}{2}$.

Donc si $\begin{cases} x \in D_0 \\ y \in D_0 \end{cases}$ alors $z = x - C(x,y)\ H(x,y)^{-1} F(x)$ existe $(P_{II})$

Par ailleurs:

$$z - \overset{*}{x} = \mathcal{C}^{-1}\left(H(x,y)\ C(x,y)^{-1}\ F'(x) + F'(x) - F'(\overset{*}{x}) + F'(\overset{*}{x})\right)(x - \overset{*}{x}) - F(x))$$

d'où

$$\|z - \overset{*}{x}\|_2 \leq \frac{a}{1-h}\left[\frac{\alpha}{2}\gamma\sqrt{p}\ (\|x - \overset{*}{x}\|_2 + \|y - \overset{*}{y}\|_2 + \alpha\ \|x - \overset{*}{x}\|_2\right]\|x - \overset{*}{x}\|_2 + \frac{\alpha}{2}\|x - \overset{*}{x}\|_2^2 \quad (P_{III})$$

(1°) . Montrons que $\{x^k\}$ est bien définie et que $x^k \in S(\overset{*}{x}, \beta)$

$x^{-1} \in S(\overset{*}{x}, \beta)$ $\quad \overset{\circ}{x} \in S(\overset{*}{x}, \beta)$

$H(\overset{\circ}{x}, x^{-1})\ C(\overset{\circ}{x}, x^{-1})^{-1}$ est inversible donc $x^1$ existe $(P_{II})$

D'après $(P_{III})$

$$\|x^1 - \overset{*}{x}\|_2 \leq \frac{a}{1-h}\left[\alpha\gamma\beta\sqrt{p} + \alpha\beta + \frac{\alpha}{2}\beta\right]\beta = \frac{h}{1-h}\beta \ ;$$

comme $h < \frac{1}{2}$ donc $\|x^1 - \overset{*}{x}\|_2 \leq \beta$.

par récurrence la suite $\{x^k\}$ est bien définie.

. Montrons que $\lim_k x^k = \overset{*}{x}$

D'après $(P_{III})$ $\qquad \|x^{k+1} - \overset{*}{x}\|_2 \leq \frac{h}{1-h}\ \|x^k - \overset{*}{x}\|_2$

et comme $\frac{h}{1-h} < 1$

alors $\|x^k - \overset{*}{x}\|_2 \leq \left(\frac{h}{1-h}\right)^k \|\overset{\circ}{x} - \overset{*}{x}\|_2$ et donc $\lim_k x^k = \overset{*}{x}$

(2°) La suite $\{x_k\}$ est d'ordre $(1+\sqrt{5})/2$.

D'après $(P_{III})$

$$\|x^{k+1} - \overset{*}{x}\|_2 \leq \frac{a}{1-h}\left\{\left[\frac{\alpha}{2}\gamma\sqrt{p} + \frac{3}{2}\alpha\right]\|x^k - \overset{*}{x}\|_2 + \frac{\alpha}{2}\gamma\sqrt{p}\ \|x^{k-1} - \overset{*}{x}\|_2\right\}\|x^k - \overset{*}{x}\|_2$$

et donc

$$\|x^{k+1}-\overset{*}{x}\|_2 \leq (\alpha_1 \|x^k-\overset{*}{x}\|_2+\beta_1 \|x^{k-1}-\overset{*}{x}\|_2)\, \|x^k-\overset{*}{x}\|_2$$

et d'après [72] p 147, théorème. la suite $\{x^k\}$ est d'ordre $(1+\sqrt{5})/2$ ■

<u>Remarques:</u>

1) Chaque itération nécessite p évaluations de fonctions sauf la première.

2) D'après les essais numériques que l'on fait, la première généralisation semble meilleure que la seconde.

3) les deux généralisations sont différentes, comme le confirme l'exemple que l'on donné pour la première généralisation. En effet, en prenant la même initialisation, nous avons obtenu deux solutions distinctes.

Reprenons l'exemple donné pour la première généralisation.

<u>Exemple numérique</u>

$$\begin{cases} x_1 = 2\,x_1 x_2 - x_1^2 \\ x_2 = x_1^2 - 2.9\,(x_2-1) \end{cases}$$

$$\overset{*}{x} = (0.95,\ 0.975)^T \quad \text{et} \quad e^k = \|x^k - \overset{*}{x}\|_2$$

| k | $x_1^k$ | $x_2^k$ | $e^k$ | $\frac{e^k}{e^{k-1}}$ |
|---|---|---|---|---|
| -1 | 0.105000000000000D+01 | 0.950000000000000D+00 | — | |
| 0 | 0.160000000000000D+01 | 0.940000000000000D+00 | 0.65D+00 | 0.63D+01 |
| 1 | 0.966318114302724D+00 | 0.971431211109540D+00 | 0.17D-01 | 0.26D-01 |
| 2 | 0.954597448676674D+00 | 0.975624076094147D+00 | 0.46D-02 | 0.28D+00 |
| 3 | 0.952224691360161D+00 | 0.976075879208661D+00 | 0.25D-02 | 0.53D+00 |
| 4 | 0.950213932429835D+00 | 0.975101824384335D+00 | 0.24D-03 | 0.96D-01 |
| 5 | 0.949990952504829D+00 | 0.974995463371114D+00 | 0.10D-04 | 0.43D-01 |
| 6 | 0.950000038444684D+00 | 0.975000019219006D+00 | 0.43D-07 | 0.42D-02 |
| 7 | 0.950000000006961D+00 | 0.975000000003481D+00 | 0.78D-11 | 0.18D-03 |
| 8 | 0.949999999999996D+00 | 0.974999999999998D+00 | 0.45D-14 | 0.57D-03 |

# Chapitre III

## Synthèse des méthodes d'accélération de la convergence des suites de vecteurs et applications.

# Introduction

Le but de ce chapitre est de présenter les méthodes d'accélération de la convergence des suites de vecteurs.
Il est divisé en trois paragraphes. Dans le premier, nous passons en revue les principales méthodes d'accélération de la convergence ainsi que les principaux résultats les concernant. Dans le second paragraphe, nous donnons de nouveaux résultats d'accélération de la convergence pour certaines méthodes, quand les suites sont générées par des itération non linéaires. Enfin, le troisième paragraphe est consacré à la connexion existante entre les méthodes d'accélération de la convergence et les méthodes de résolution de systèmes d'equations non linéaires. Nous montrons par une étude unifiée, qu'en particulier toute une classe de méthodes est à convergence quadratique.

## A – Présentation des méthodes d'accélération de la convergence.

Dans ce paragraphe, nous donnons les définitions des diverses méthodes d'accélération de la convergence. Pour chacune des méthodes traitées nous donnerons, quand c'est possible, des propriétés algébriques ainsi que des résultats de convergence. des méthodes que l'on va voir sont essentiellement de deux types: ① des méthodes linéaires, c'est à dire les méthodes dont les coefficients ne dépendent pas directement des termes de la suite. Ces méthodes sont surtout utilisées pour des suites générées linéairement. Par exemple celles considérées dans les methodes itératives de résolution des systèmes d'équations linéaires.

② Pour les suites quelconques, c'est à dire celles pour lesquelles on dispose seulement d'une information numérique, on utilisera des méthodes non linéaires.

### I – Méthodes linéaires

Dans cette section, nous allons nous intéresser principalement aux suites générées linéairement :

Soit $\{s_n\}$ une suite de vecteurs , cette suite sera dite générée linéairement si l'on a :

$$s_{n+1} = B s_n + c$$

où $B$ est une matrice carrée, $c$ est un vecteur. Pour simplifier la présentation, nous allons, dans cette partie, nous limiter au cas de suites de vecteurs réels.

Pour pouvoir comparer la vitesse de convergence de deux suites générées linéairement, nous avons besoin de la définition suivante [87].

> **Définition :** Soit $\{s_n\}$ une suite générée linéairement telle que :
>
> $$s_{n+1} = B s_n + c .$$
>
> On appelle taux asymptotique de convergence le nombre (positif si $\rho(B) < 1$)
>
> $$R_\infty(B) = -\text{Log}\, \rho(B)$$

Ce taux asymptotique permet d'apprécier la vitesse de convergence d'une suite. Une suite convergente est d'autant plus rapide que son taux asymptotique de convergence est plus grand (i.e que $\rho(B)$ sera petit).

Remarques

1. $R_\infty(B)$ représente le nombre de digits décimaux exacts ajoutés à chaque itération.

2. Soient $\{s_n\}$ et $\{t_n\}$ deux suites générées linéairement :

$$s_{n+1} = B s_n + c \quad , \; B \in L(\mathbb{R}^p) \quad \text{et } c \in \mathbb{R}^p$$

$$\text{et} \quad t_{n+1} = B' t_n + c' \quad , \; B' \in L(\mathbb{R}^p) \quad \text{et } c' \in \mathbb{R}^p$$

si $\quad \rho(B) < \rho(B') < 1 \quad$ alors $\quad R_\infty(B) > R_\infty(B')$

et donc la suite $\{s_n\}$ converge plus vite que la suite $\{t_n\}$ [35]

Commençons notre revue par la méthode de Richardson.

## 1) Méthode de Richardson [37]

Pour résoudre un système linéaire $Ax = b$
où A est une matrice carrée inversible, b est un vecteur, on peut utiliser une méthode itérative. Pour cela effectuons une décomposition de la matrice A. Nous avons :

$$A = M - N \quad , \quad M \in L(\mathbb{R}^p) \text{ et } N \in L(\mathbb{R}^p)$$

M étant supposée régulière.
Le système à résoudre peut alors s'écrire :

$$Mx = Nx + b$$

Définissons la suite de vecteurs $\{s_n\}$ par :

$$\begin{cases} s_0 \quad \text{donné} \\ s_{n+1} = B s_n + c \end{cases}$$

où $B = M^{-1}N$ et $c = M^{-1}b$.

Pour améliorer la convergence de la suite $\{s_n\}$, on propose la transformation suivante :

$$t : \{s_n\} \longrightarrow \{t_n\}$$

avec

$$t_n = s_n + \alpha \Delta s_n = s_n + \alpha (c - T s_n)$$

où $\alpha$ est un réel strictement positif et $T = I - B$

Supposons que la suite $\{s_n\}$ converge vers $s$ ; on a

$$s = Bs + s$$

et en posant $e_n = s_n - s$, on obtient :

$$e_{n+1} = B e_n = (I-T) e_n .$$

$\left\{ \begin{array}{l} \text{la suite } \{s_n\} \text{ converge} \\ \text{quelque soit } s_0 \end{array} \right\}$ si et seulement si $\{ \rho(B) = \rho(I-T) < 1 \}$

De plus on a :

$$t_n - s = (I - \alpha T)(s_n - s)$$

On voit que l'amélioration de la convergence dépend du rayon spectral de la matrice $I - \alpha T$.

Calculons le taux asymptotique de convergence de la suite $\{t_n\}$. On a

$$t_{n+1} - s = (I - \alpha T)(s_{n+1} - s) = (I - \alpha T)(I - T)(s_n - s)$$
$$= (I-T)(I - \alpha T)(s_n - s)$$

On obtient :

$$t_{n+1} - s = (I - T)(t_n - s) .$$

Donc

1) $\left\{ \begin{array}{l} \text{la suite } \{t_n\} \text{ converge} \\ \text{quelque soit } t_0 \end{array} \right\}$ si et seulement si $\{\rho(I-T) < 1\}$

2) La suite $\{t_n\}$ converge mieux que la suite $\{s_n\}$ si la constante $\alpha$ est telle que : $\rho(I - \alpha T) < 1$.

La suite $\{t_n\}$ possède le même taux asymptotique de convergence que la suite $\{s_n\}$ ; on va donc itérer ce procédé, et pour cela posons :

$$t_{n,0} = s_n ,$$

et définissons une suite $\{t_{n,k}\}$ qui dépend de deux indices par :

$$\begin{cases} t_{n,0} = s_n \\ t_{n,k} = t_{n,k-1} + \alpha ( c - T t_{n,k-1}) \end{cases}$$

Proposition 1

> La suite double $\{t_{n,k}\}_{n,k}$ possède les deux propriétés suivantes :
> $$t_{n,k} - s = (I - \alpha T) ( t_{n,k-1} - s) \qquad (1)$$
> et
> $$t_{n+1,k} - s = (I - T) ( t_{n,k} - s) \qquad (2)$$

preuve :

- La première égalité est évidente, puisqu'il suffit de voir que $c = T s$

- La seconde égalité se démontre par récurrence sur $k$

pour $k=0$ ; on a par construction $t_{n,0} = s_n$, et par hypothèse $s_{n+1} - s = (I - T)(s_n - s)$.

Pour $k=1$ $\quad t_{n,1} = t_n$ et on a vu précédemment que :

$$t_{n+1} - s = (I - T) ( t_n - s)$$

Supposons que l'égalité (2) soit vraie pour $k-1$ et montrons qu'elle est vraie pour $k$.

Par hypothèse $\quad t_{n+1,k-1} = (I - T) ( t_{n,k-1} - s)$

Comme

$$t_{n+1,k} - s = (I - \alpha T) ( t_{n+1,k-1} - s)$$

On a donc

$$t_{n+1,k} - s = (I - \alpha T)(I - T) ( t_{n,k-1} - s)$$

De plus

$$= (I - T) (I - \alpha T) ( t_{n,k-1} - s)$$
$$= (I - T) ( t_{n,k} - s),$$

ce qui termine la démonstration. ■

A partir de la proposition précédente, on peut faire les deux remarques suivantes :

① pour k fixé, la suite $\{t_{n,k}\}_n$ converge si et seulement si $\rho(I-T) < 1$.

Le taux asymptotique de convergence est : $R_\infty(I-T) = R_\infty(B)$, et donc en général la suite $\{t_{n,k}\}_n$ ne converge pas plus vite que la suite $\{s_n\}$, on a seulement amélioration de la convergence ($k \geq 1$).

② pour n fixé, la suite $\{t_{n,k}\}_k$ converge si et seulement si $\rho(I-\alpha T) < 1$. Dans ce cas, le taux asymptotique de convergence est : $R_\infty(I-\alpha T) = -\log(\rho(I-\alpha T))$. On voit donc que si $\alpha$ est tel que $\rho(I-\alpha T) < \rho(I-T)$ alors la suite $\{t_{n,k}\}_k$ converge plus vite que la suite $\{s_n\}$ puisque l'a $R_\infty(I-\alpha T) > R_\infty(I-T)$.

C'est donc ce dernier cas qui est intéressant dans la pratique, et surtout lorsque l'on peut trouver la valeur de $\alpha$ optimale, c'est à dire celle qui minimise $\rho(I-\alpha T)$. La détermination de la valeur de $\alpha$ optimale est possible dans certains cas, c'est ce que l'on va voir maintenant.

Faisons l'hypothèse suivante :

T est symétrique, définie positive.

Notons $\lambda_1, \lambda_2, -, \lambda_p$ ses valeurs propres avec :

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p > 0$$

Donnons une condition nécessaire et suffisante de convergence.

**théorème [37]**
Si la matrice $T$ est symétrique définie positive, alors :
la suite $\{t_{m,k}\}_k$ converge si et seulement si $\alpha < \frac{2}{\lambda_1}$

Pour ce type de matrices, on peut même donner la valeur optimale de $\alpha$ [ ] :

$$\rho(I - \alpha_{opt} T) \leq \rho(I - \alpha T) \quad \forall \alpha \in ]0, \frac{2}{\lambda_1}[$$

et

$$\alpha_{opt} = \frac{2}{\lambda_1 + \lambda_p}.$$

D'autres résultats concernant ce genre de méthodes se trouvent dans [37], [44], [62], ainsi que dans [40] où il y a une revue de toutes les méthodes d'extrapolation de ce type ainsi que des généralisations.

Remarques

① Avec les hypothèses du théorème précédent pour connaître $\alpha_{opt}$, il faut connaître $\lambda_1$ et $\lambda_p$.

② $$\rho(I - \alpha_{opt}) = 1 - \frac{2}{\lambda_1 + \lambda_p} \lambda_p = \frac{K(T) - 1}{K(T) + 1}$$

avec $K(T) = \frac{\lambda_1}{\lambda_p}$ qui est le conditionnement de $T$.

Cette méthode peut se généraliser facilement ; on peut, par exemple, prendre $\alpha$ dépendant de $k$, On obtient alors l'algorithme suivant :

$$\begin{cases} t_{n,0} = s_n \\ \text{et} \\ t_{n,k} = t_{n,k-1} + \alpha_{k-1}\,(c - T\,t_{n,k-1}) \end{cases}$$

On vient de définir une suite double $\{t_{n,k}\}_{n,k}$ dont le comportement est donné par la :

<u>Proposition 2 :</u>

La suite double $\{t_{n,k}\}_{n,k}$ possède les propriétés suivantes :

$$t_{n,k} - s = (I - \alpha_{k-1} T)\,(t_{n,k-1} - s)$$

et

$$t_{n+1,k} - s = (I - T)\,(t_{n,k} - s).$$

preuve analogue à celle donnée pour la proposition 1.

Comme pour la méthode précédente, la suite $\{t_{n,k}\}_n$ ne converge pas plus vite que la suite $\{s_n\}$ en général.

On a $$t_{n,k} - s = \prod_{i=0}^{k-1} (I - \alpha_i T)\;(t_{n,0} - s)$$

$$t_{n,k} - s = P_k(T)\,(t_{n,0} - s)$$

$$P_k(x) = \prod_{i=0}^{k-1} (1 - \alpha_i t)$$

$P_k$ est un polynome de degré $k$.

Faisons la même hypothèse que précédemment c'est à dire $T$ symétrique, définie positive et notons $Q$ son polynome caractéristique.

On a $\quad Q(x) = \prod_{i=1}^{p} (\lambda_i - x)$ , où les $\lambda_i$ sont les valeurs propres de $T$ avec $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_p > 0$

D'après le théorème de Cayley-Hamilton : $Q(T) = 0$.

En prenant $\quad d_{i-1} = \lambda_i \quad$ pour $i = 1, \dots, p-1$

on obtient $\quad t_{m,p} = s \quad \forall m$.

Avec ce choix la méthode devient :

$$\begin{cases} t_{m,0} = s_n \\ t_{m,k} = t_{m,k-1} + \frac{1}{\lambda_k}(c - T\, t_{m,k-1}) \quad \text{pour } k \leq p \end{cases}$$

de problème avec ce type de méthode est la connaissance des valeurs propres de $T$.

Cette méthode est souvent présentée d'une autre façon.
Elle s'appelera la méthode de Lyusternik-Pugashev.

2) <u>Méthode de Lyusternik-Pugashev</u> [4], [63]

a- <u>Méthode de Lyusternik</u>

Soit $\{s_n\}$ une suite generée linéairement :

$$\begin{cases} s_0 \text{ donné} \\ s_{n+1} = B s_n + c \end{cases}$$

Supposons que les valeurs propres $\mu_i$ de $B$ soient toutes distinctes et notons $v_1, \dots, v_p$ les vecteurs propres associés réspectivement à $\mu_1, \mu_2 \dots, \mu_p$. nous pouvons décomposer le vecteur $s_0 - s$ suivant la base $\{v_i\}_{i=1,\dots p}$.

On a alors :

$$e_0 = s_0 - s = \sum_{i=1}^{p} a_i v_i$$

où $\quad s = Bs + c \quad .$

Supposons de plus que les valeurs propres de T soient telles que :

$$|\mu_1| > |\mu_2| > |\mu_3| > \cdots > |\mu_p| \quad \text{et} \quad \mu_i \neq 1 \, , \, i=1,-p$$

On a déja vu que :

$$e_n = s_n - s = B(s_{n-1} - s) = B^n (s_0 - s) = B^n e_0$$

et donc

$$e_n = \sum_{i=1}^{p} a_i \mu_i^n v_i$$

et

$$\Delta s_n = \Delta e_n = \sum_{i=1}^{p} a_i (\mu_i - 1) \mu_i^n v_i .$$

En supposant que la valeur propre $\mu_1$ est connue, Lusternik a montré que l'on pouvait accélérer la convergence de la suite $\{s_n\}$ en utilisant la transformation t définie par :

$$\begin{cases} t : \{s_n\} \longrightarrow \{t_n\} \\ t_n = s_n + \dfrac{1}{1-\mu_1} \Delta s_n . \end{cases}$$

On a

$$t_n - s = s_n - s + \frac{1}{1-\mu_1} \Delta s_n$$

et par conséquent

$$t_n - s = \sum_{i=2}^{p} a_i \frac{(\mu_i - \mu_1)}{1-\mu_1} \mu_i^n v_i = \sum_{i=2} b_i \mu_i^n v_i$$

On voit que si $\quad a_1 \neq 0 \quad$ alors $\quad s_n - s = a_1 \mu_1^n v_1 + \varepsilon_n \mu_1^n$

$$\text{et} \quad t_n - s = b_{i_0} \mu_{i_0}^n v_{i_0} + \varepsilon'_n \mu_{i_0}^n$$

où $\quad \lim_n \varepsilon_n = \lim_n \varepsilon'_n = 0$ , le nombre $i_0$ est tel que :

$$a_i = 0 \quad \text{pour} \quad i = 2, -, i_0 - 1 \text{ et } a_{i_0} \neq 0 .$$

Avec ces relations, on voit que la suite $\{t_n\}$ converge plus vite que la suite $\{s_n\}$.

On peut recommencer le même raisonnement pour éliminer $a_2$ et ainsi de suite.

Posons $s_{n,0} = s_n$ et $s_{n,1} = t_n$.

Pour éliminer $a_2$, il suffit de poser :

$$s_{n,2} = s_{n,1} + \frac{1}{1-\mu_2}(s_{n+1,1} - s_{n,1})$$

$$= \frac{s_{n+2} - (\mu_1+\mu_2)\, s_{n+1} + \mu_1\mu_2}{(1-\mu_1) \times (1-\mu_2)}$$

$$\left\{\begin{array}{ll} \text{si} & a_2 \neq 0 \\ \text{et} & a_3 \neq 0 \end{array}\right. \quad \text{alors} \quad \begin{array}{l} s_{n,1} - s = b_2\, \mu_2^n\, v_2 + \varepsilon'_n\, \mu_2^n \\ s_{n,2} - s = c_2\, \mu_3^n\, v_3 + \varepsilon''_n\, \mu_3^n \end{array}$$

avec $c_2 \neq 0$ et $\lim\limits_n \varepsilon''_n = 0$

$\{s_{n,2}\}$ converge plus vite que $\{s_{n,1}\}$

Cette technique a été généralisée par Pugachev, et c'est l'objet de ce qui suit :

## b- Méthode de Pugachev.

Montrons que $s_{n,1}$ et $s_{n,2}$ peuvent se mettre sous la forme d'un rapport de deux déterminants :

$$s_{n,1} = \begin{vmatrix} s_n & s_{n+1} \\ 1 & \mu_1 \end{vmatrix} \Big/ \begin{vmatrix} 1 & 1 \\ 1 & \mu_1 \end{vmatrix} = \begin{vmatrix} s_n & \Delta s_n \\ 1 & \mu_1 - 1 \end{vmatrix} \Big/ (\mu_1 - 1)$$

et

$$s_{n,2} = \begin{vmatrix} s_n & s_{n+1} & s_{n+2} \\ 1 & \mu_1 & \mu_1^2 \\ 1 & \mu_2 & \mu_2^2 \end{vmatrix} \Big/ \begin{vmatrix} 1 & 1 & 1 \\ 1 & \mu_1 & \mu_1^2 \\ 1 & \mu_2 & \mu_2^2 \end{vmatrix}$$

A partir de ces relations, on peut facilement généraliser la méthode de Lyasternik ; c'est ce qui a été fait par Pugachev pour obtenir la suite $\{s_{n,k}\}$ :

$$s_{n,k} = \begin{vmatrix} s_n & s_{n+1} & \cdots & s_{n+k} \\ 1 & \mu_1 & \cdots & \mu_1^k \\ \cdots & \cdots & \cdots & \cdots \\ 1 & \mu_k & \cdots & \mu_k^k \end{vmatrix} \Big/ \begin{vmatrix} 1 & 1 & \cdots & 1 \\ 1 & \mu_1 & \cdots & \mu_1^k \\ \cdots & \cdots & \cdots & \cdots \\ 1 & \mu_k & \cdots & \mu_k^k \end{vmatrix}$$

Posons
$$\begin{cases} a_i^{(0)} = a_i & i = 1, - p \\ \text{et} \\ a_i^{(k)} = a_i^{(k-1)} \dfrac{(\mu_i - \mu_k)}{1 - \mu_k} & k = 1, - p-1 \text{ et } i = k+1, - p \end{cases}$$

Alors
$$\begin{cases} s_{n,k} - s = \displaystyle\sum_{i=k+1}^{p} a_i^{(k)} \mu_i^n v_i & k < p \\ \text{et } s_{n,p} = s \end{cases}$$

De plus, les $s_{n,k}$ peuvent être calculés d'une façon recursive. En effet :

$$\begin{cases} s_{n,0} = s_n \\ s_{n,k} = s_{n,k-1} + \dfrac{1}{1-\mu_k} \Delta s_{n,k-1} & k = 1, - p. \end{cases}$$

Posons $\alpha_{k-1} = \dfrac{1}{1-\mu_k}$ pour $k = 1, -, p$.

Nous allons montrer que $s_{n,k} = t_{n,k}$ où

$$s_{n,0} = t_{n,0}$$

$$s_{n,k} = s_{n,k-1} + d_{k-1}\, \Delta s_{n,k-1}$$

$$t_{n,k} = t_{n,k-1} + d_{k-1}\,(c - T t_{n,k-1}) .$$

Pour cela, il suffit de prouver que $\Delta t_{n,k-1} = c - T t_{n,k-1}$.

Nous avons déja montré que :

$$t_{n,k} - s = \prod_{i=0}^{k-1} (I - d_i T)(s_n - s)$$

et donc $\quad t_{n+1,k} - s = \prod_{i=0}^{k-1} (I - d_i T)\, B(s_n - s).$

Faisons la soustraction de ces deux égalités ; nous obtenons :

$$\Delta t_{n,k} = \prod_{i=0}^{k-1} (I - d_i T)\, T(s - s_n) .$$

D'autre part $\quad c - T t_{n,k-1} = T(s - t_{n,k-1})$

$$= T \prod_{i=0}^{k-1} (I - d_i T)(s - s_n) = \Delta t_{n,k} .$$

Les valeurs propres de $T$ ont été notées $\lambda_i$, et celles de $B$ $\mu_i$ ;

comme $B = I - T$, alors $\mu_i = 1 - \lambda_i$ .

Nous avons donc obtenu deux interprétations différentes de la méthode Lyusternik - Pugachev .

Comme on l'a déja remarqué, le calcul de $t_{n,k}$ nécessite la connaissance des valeurs propres de $T$ ; cette information est évidemment utopique car il est équivalent de connaître les valeurs propres ou de connaître $s$.

L'intérêt essentiel de cette méthode, réside dans le fait qu'elle permet la compréhension du fonctionnement de méthodes plus compliquées comme les méthodes non linéaires que l'on verra par la suite.

Lorsque l'on remplace l'information précédente, c'est à dire la connaissance des valeurs propres, par une information plus restreinte, comme par exemple le fait que les valeurs propres sont dans un domaine du plan complexe, on obtient les méthodes dites universelles au sens de Faddeev et Faddeeva [28].

## 3) Méthodes universelles

### a) Méthode universelle au sens du premier critère.

Pour résoudre le système d'equations $Ax=b$, on a vu que l'on pouvait utiliser des méthodes itératives qui peuvent s'ecrire sous la forme :

$$s_{m+1} = (I - \alpha A)\, s_n + \alpha b$$
$$= B\, s_m + c$$

où $B = I - \alpha A$ et $c = \alpha b$

Notons $\lambda_1, \dots, \lambda_p$ les valeurs propres de A et supposons que ces valeurs propres sont dans l'intervalle $[m, M]$ avec $0 < m < M$.

des valeurs propres de $B$ sont $1-\alpha\lambda_i$. Prenons $\alpha = \frac{2}{m+M}$ et posons $\gamma = \frac{M+m}{M-m}$ ; alors $1-\alpha\lambda_i \in [-\frac{1}{\gamma}, \frac{1}{\gamma}]$.

Comme $\gamma > 1$, la suite $\{s_n\}$ converge, notons $s$ sa limite.

Pour accélérer la convergence de cette suite, nous allons construire une nouvelle suite $\{t_{n,k}\}$ de la manière suivante :

$$t_{n,k} = s_n + \sum_{i=1}^{k-1} \beta_i^{(n,k)} \Delta s_{n+i}$$

Nous allons montrer comment choisir les coefficients $\beta_i^{(n,k)}$, mais auparavant exprimons $t_{n,k}-s$ en fonction de $s_n-s$.
Par définition on a :

$$t_{n,k} - s = s_n - s + \sum_{i=1}^{k-1} \beta_i^{(n,k)} \Delta s_{n+i}$$

Or $s$ vérifie la relation : $s = Bs + c$

et donc $\Delta s_{n+i} = B^i (B-I)(s_n - s)$

D'où

$$t_{n,k} - s = s_n - s + \sum_{i=1}^{k-1} \beta_i^{(n,k)} B^i (B-I)(s_n - s)$$
$$= e_k^{(n)}(B)\,(s_n - s)$$

avec

$$e_k^{(n)}(x) = 1-(1-x)\, f_{k-1}^{(n)}(x) \quad \text{et} \quad f_{k-1}^{(n)}(x) = \sum_{i=0}^{k-1} \beta_i^{(n,k)} x^i .$$

L'accélération de la convergence dépend de $R_\infty(e_k^{(n)}(B))$. Pour que ce nombre soit le plus grand possible, nous allons choisir $e_k^{(n)}$ comme étant la meilleure approximation uniforme de zéro sur $\mathcal{P}_k$ espace vectoriel des polynômes de degré $\leq k$.

Soit $C_\infty[u,v]$ l'espace vectoriel des fonctions continues sur $[u,v]$ muni de la norme:

$$\|f\| = \max_{x\in[u,v]} |f(x)| \qquad \forall f \in C_\infty[u,v].$$

$e_k^{(n)}$ est donc choisi tel que:

$$\|0 - e_k^{(n)}\| = \min_{g\in\mathcal{P}_k} \max_{x\in[-\frac{1}{\gamma},\frac{1}{\gamma}]} |0 - g(x)|,$$

Il est bien connu que:

$$e_k^{(n)}(x) = \frac{T_k(\gamma x)}{T_k(\gamma)}$$

(Ce polynôme est indépendant de $n$, nous allons le noter $e_k$).

avec $\quad T_k(x) = \cos(k \arccos x) \qquad x \in [-1,+1].$

$T_k$ est le polynôme de Tchebysheff de 1e espèce.

On sait que ces polynômes peuvent se calculer par la relation de récurrence:

$$\begin{cases} T_0(x) = 1 \quad ; \; T_1(x) = x \\ T_i(x) = 2x\,T_{i-1}(x) - T_{i-2}(x) \end{cases}.$$

A partir de ces relations, on peut donner un algorithme récursif pour le calcul des $t_{n,k}$:

$$\begin{cases} t_{n,0} = s_n \\ t_{n,1} = s_{n+1} = B s_n + c \\ t_{n,i} = \left[1 + \frac{T_{i-2}(\gamma)}{T_i(\gamma)}\right](B t_{n,i-1} + c) - \frac{T_{i-2}(\gamma)}{T_i(\gamma)}\, t_{n,i-2} \qquad i \geq 2 \end{cases}$$

Nous allons écrire ces relations sous une autre forme.

Posons $\quad w_i = \frac{T_{i-2}(\gamma)}{T_i(\gamma)} \quad$ et $\quad w_1 = 1.$

D'après la relation qui relie les polynômes de Tchebycheff on a :

$$T_i(\gamma) = 2\gamma\, T_{i-1}(\gamma) - T_{i-2}(\gamma)$$

et donc $\quad w_i = 2\gamma \dfrac{T_{i-1}(\gamma)}{T_i(\gamma)}$ ; de plus

$$1 = 2\gamma \frac{T_{i-1}(\gamma)}{T_i(\gamma)} - \frac{T_{i-2}(\gamma)}{T_i(\gamma)} = w_i - \frac{w_i w_{i-1}}{4\gamma^2}$$

d'où

$$w_i = \frac{1}{1 - \dfrac{w_{i-1}}{4\gamma^2}} .$$

En définitive, nous obtenons l'algorithme suivant :

$$\left[\begin{array}{l} t_{n,0} = s_n \quad , \quad t_{n,1} = B s_n + c \\ t_{n,i} = w_i \left(B\, t_{n,i-1} + c - t_{n,i-2}\right) + t_{n,i-2} \\ \\ w_i = \dfrac{1}{1 - \dfrac{w_{i-1}}{4\gamma^2}} \quad \text{et} \quad w_1 = 1 \end{array}\right.$$

Sous cette forme et avec $n=0$, cette méthode est généralement appelée méthode semi-itérative de Tchebycheff [87].

Etudions maintenant les suites $\{t_{n,k}\}_n$ et $\{t_{n,k}\}_k$.

par construction

$$t_{n+1,k} - s = e_k(B)\,(s_{n+1} - s)$$
$$= B\, e_k(B)\,(s_n - s) = B\,(t_{n,k} - s)$$

et donc :

$$t_{n+1,k} = B\, t_{n,k} + c .$$



La vitesse de convergence de la suite $\{t_{n,k}\}_n$ est caractérisée par le taux asymptotique de convergence qui est $R_\infty(B)$.

Pour $k$ fixé, on n'a pas accélération de la convergence, mais seulement amélioration de la convergence puisque :

$$t_{n,k} - s = e_k(B)\,(s_n - s).$$

A partir de cette relation, on peut démontrer que :

$$\|t_{n,k} - s\| \leq \frac{1}{T_k(\gamma)}\,\|s_n - s\|.$$

Pour $n$ fixé, la suite $\{t_{n,k}\}_k$ converge plus vite que la suite $\{s_n\}$.

Il existe des généralisations de cette méthode quand les valeurs propres sont incluses dans un domaine $D$ du plan complexe, par exemple une ellipse ayant des axes parallèles aux axes de coordonnées et ne contenant pas l'origine, voir [54], [71].

Nous allons étudier une autre méthode de type universel.

b) <u>Méthode universelle au sens du second critère.</u>

On utilise les mêmes notations et hypothèses que dans le paragraphe précédent.

On a vu que $t_{n,k} - s = e_k^{(n)}(B)\,(s_n - s)$

et $\Delta s_n = (B - I)\,(s_n - s)$

Supposons $(B - I)$ inversible, alors

$$t_{n,k} - s = e_k^{(n)}(B)\,(B - I)^{-1}\,\Delta s_n$$

$$= g_k^{(n)}(B)\,\Delta s_n\,.$$

$$g_k^{(n)}(x) = \left\{1-(1-x)\, f_{k-1}^{(n)}(x)\right\}(x-1)^{-1}$$

$$= f_{k-1}^{(n)}(x) - \frac{1}{1-x}.$$

Le polynôme $f_{k-1}^{(n)}$ va être choisi comme étant la meilleure approximation uniforme de la fonction $h$ sur $\mathcal{P}_{k-1}$ où :

$$h(x) = \frac{1}{1-x} \qquad x \in \left[-\frac{1}{\gamma}, \frac{1}{\gamma}\right]$$

$$h \in C_\infty\left[-\frac{1}{\gamma}, \frac{1}{\gamma}\right] \quad , \quad \gamma > 1$$

On a :

$$\| h - f_{k-1}^{(n)} \| = \min_{g \in \mathcal{P}_{k-1}} \max_{x \in [-\frac{1}{\gamma}, \frac{1}{\gamma}]} \left| \frac{1}{1-x} - g(x) \right|$$

La solution de ce problème est :

$$f_{k-1}^{(n)}(x) = f_{k-1}(x)$$

$$= \frac{1}{1-x} - 2\frac{\theta^{k/2}}{(1-\theta)^2}\frac{1}{1-x}\left[T_k(\gamma x) - 2\sqrt{\theta}\, T_{k-1}(\gamma x) + \theta\, T_{k-2}(\gamma x)\right]$$

avec

$$\theta = (\gamma - \sqrt{\gamma^2 - 1})^2.$$

On a vu que $\quad e_k(x) = 1 - (1-x)\, f_{k-1}(x)$

En utilisant le fait que les polynômes de Tchebycheff vérifient une relation de récurrence à trois termes, on montre que :

$$\begin{cases} e_i(x) = (1+\theta)x\, e_{i-1}(x) - \theta\, e_{i-2}(x) & i \geq 2 \\ e_1(x) = \left(\frac{1+\theta}{1-\theta}\right)^2 (x-1) + 1 & \\ e_2(x) = x\, e_1(x) + \frac{2\theta}{1-\theta}(x-1) & \end{cases}$$

et comme $\quad t_{n,k} - s = e_k(B)\,(s_n - s)$

On en déduit un algorithme de calcul des $t_{n,k}$ :

$$t_{n,0} = s_n$$

$$\begin{cases} t_{n,k} = (1+\theta)\,(B\,t_{n,k-1} + c) - \theta\; t_{n,k-2} & k \geq 2 \\ t_{n,1} = t_{n,0} + \left(\frac{1+\theta}{1-\theta}\right)(B\,t_{n,0} + c - t_{n,0}) \\ t_{n,2} = B\,t_{n,1} + c + \frac{2\theta}{1-\theta}\left(B\,t_{n,0} + c - t_{n,0}\right) \end{cases}$$

et $\quad t_{n+1,k} = B\,t_{n,k} + c$.

Les remarques que l'on a faites pour la méthode universelle au sens du premier critère restent valables pour cette méthode. Il existe d'autres méthodes de ce type ; pour plus de détails voir [28].

Les méthodes universelles étudiées font partie d'une classe appelée méthode d'accélération polynomiale dans [41]. Cette classe contient également la méthode du gradient conjugué.

4) Méthode du gradient conjugué.

Soit $\{s_n\}$ une suite définie par :

$$s_{n+1} = B\,s_n + c\,,$$

et qui converge vers la solution de l'équation $Ax = b$.

Posons $A = M - N$ où $M$ est une matrice inversible,

$B = M^{-1}N$ et $c = M^{-1}b$. Supposons de plus que $A$ et $M$ sont symétriques définies positives.

Pour accélérer la convergence de la suite $\{s_n\}$, on propose de choisir une nouvelle suite $\{t_{n,k}\}$ telle que $t_{n,k}$ soit de la forme :

$$t_{m,k} = s_n + \sum_{i=0}^{k-1} \beta_i^{(m,k)} \Delta^{i+1} s_m .$$

Exprimons $\Delta^{i+1}s_n$ en fonction de $\Delta s_n$ et de $B$:

$$\Delta s_{n+1} = s_{n+2} - s_{n+1} = (B s_{n+1} + c) - (B s_n + c) = B \Delta s_n$$

et $\Delta^2 s_n = (B-I)\Delta s_n$.

Par recurrence on a : $\Delta^{i+1} s_n = (B-I)^i \Delta s_n$.

Posons $T = (B-I)$ et $r(x) = c + Tx$ alors $r(s_n) = \Delta s_n$.

$t_{m,k}$ et $r(t_{m,k})$ peuvent s'écrire sous la forme

$$t_{m,k} = s_n + \sum_{i=0}^{k-1} \beta_i^{(m,k)} T^i \Delta s_n \qquad (I)$$

et

$$r(t_{m,k}) = r(s_n) + \sum_{i=0}^{k-1} \beta_i^{(m,k)} T^{i+1} r(s_n) \qquad (II) .$$

Nous allons déterminer les paramètres $\beta_0^{(m,k)}, \ldots, \beta_{k-1}^{(m,k)}$ de sorte que l'on ait

$$(r(t_{m,i}), M r(t_{m,j})) = 0 \qquad i \neq j \quad \begin{matrix} i=0, \ldots k \\ j=0, \ldots k \end{matrix}$$

D'après (I) on a :

$$\begin{cases} t_{m,1} = s_n + \beta_0^{(m,1)} r(s_n) \\ t_{m,2} = s_m + \beta_0^{(m,2)} r(s_n) + \beta_1^{(m,2)} T r(s_n) \\ \cdots \\ t_{m,k} = s_n + \beta_0^{(m,k)} r(s_n) + \cdots + \beta_{k-1}^{(m,k)} T^{k-1} r(s_n) \end{cases}$$

A partir de ces relations, on peut exprimer $T^{j-1} r(s_n)$ en fonction de $t_{m,i} - s_n$ pour $i = 1, \ldots j$. En effet :

$$T^{j-1} r(s_m) = \sum_{i=1}^{j} d_i^{(m,j)} (t_{m,j} - s_m) \qquad j=1,\dots k$$

Reportons ces égalités dans II, on obtient

$$r(t_{m,k}) = \sum_{i=1}^{k+1} \gamma_i^{(m,k)} (t_{m,k} - s_m) .$$

Multiplions cette égalité par A et utilisons la relation $A = -MT$, on a:

$$\left\{ \begin{array}{l} A\, r(t_{m,k}) = \sum_{i=0}^{k+1} \alpha_i^{(m,k)} M\, r(t_{m,i}) \\ \text{et } \sum_{i=0}^{k+1} \alpha_i^{(m,k)} = 0 . \end{array} \right. \qquad \text{III}$$

Nous allons maintenant établir que :

$$\alpha_j^{(m,k)} = 0 \qquad \text{pour } j=0,\dots, k-2.$$

Remarquons tout d'abord que $(r(t_{m,k}), M T^j \Delta s_n) = 0 \quad j=0.\dots k-1$

car

$$T^j r(s_n) = \sum_{i=1}^{j} d_i^{(m,j)} (r(t_{n,i}) - r(t_{n,0})) .$$

De plus pour $j=0,\dots k-1$.

$$(r(t_{m,k}), M T^j \Delta s_m) = (r(t_{m,k}), MT\, T^{j-1} \Delta s_n) = 0$$

$$\Rightarrow (A\, r(t_{m,k}), T^{j-1} \Delta s_m) = 0 .$$

D'après II

$$(A\, r(t_{m,k}) , r(t_{m,j})) = 0 \qquad j=0,\dots, k-2 .$$

Multiplions scalairement la relation III par $r(t_{m,j})$ pour $j=0,\dots k$; on a alors :

$$0 = \alpha_j^{(m,k)} (M\, r(t_{m,j}), r(t_{m,j})) \qquad j=0,\dots k-2 .$$

D'où $\alpha_j^{(m,k)} = 0 \quad j=0,\dots k-2$, et $$\sigma_j^{(m,k)} = \frac{(A\, r(t_{m,k}), r(t_{m,j}))}{(M\, r(t_{m,j}), r(t_{m,j}))} \qquad j = k-1, k.$$

D'après III
$$r(t_{m,k}) = A^{-1} \sum_{i=k-1}^{k+1} \sigma_i^{(m,k)} M\, r(t_{m,i})$$
$$= -\sigma_{k-1}^{(m,k)} t_{m,k-1} - \sigma_k^{(m,k)} t_{m,k} - \sigma_{k+1}^{(m,k)} t_{m,k+1}$$

et $\sigma_{k+1}^{(m,k)} = -\sigma_{k-1}^{(m,k)} - \sigma_k^{(m,k)}$.

Posons $\alpha_k^{(m)} = \dfrac{1}{\sigma_k^{(m,k)}}$ et $w_{k+1}^{(m)} = -\dfrac{\sigma_k^{(m,k)}}{\sigma_{k+1}^{(m,k)}}$. Nous sommes en mesure de donner un algorithme récursif qui permet le calcul des termes $t_{m,k}$ pour $m$ fixé :

$$\left|\begin{array}{l} t_{m,-1} = t_{m,0} = s_m \\ \text{et} \\ t_{m,k+1} = t_{m,k-1} + w_{k+1}^{(m)} (t_{m,k} - t_{m,k-1}) + w_{k+1}^{(m)} \alpha_k^{(m)}\, r(t_{m,k}) \end{array}\right.$$

Examinons le lien qui existe entre cette méthode et la méthode du gradient conjugué.

Si on note $\{y^k\}$ la suite $\{t_{m,k}\}$ pour $m$ fixé, l'algorithme décrit plus haut s'écrit :

$$\left|\begin{array}{l} y^0 = y^{-1} = s_m \\ 1)\quad M\,.\, z^k = b - A y^k \\ 2)\quad \alpha_k = \dfrac{(z^k, M z^k)}{(z^k, A z^k)} \\ 3)\quad w_{k+1} = \left(1 - \dfrac{\alpha_k}{w_k \alpha_{k-1}} \dfrac{(z^k, M z^k)}{(z^{k-1}, M z^{k-1})}\right)^{-1}, \quad w_1 = 1 \end{array}\right.$$

4) $y^{k+1} = y^{k-1} + \omega_{k+1}\left(\alpha_k z^k + y^k - y^{k-1}\right)$ .

Dans cet algorithme nous avons posé : $z^k = r(y^k) = M^{-1}b - M^{-1}Ay^k$ .

La méthode définie par les relations 1), 2), 3) et 4) est la méthode du gradient conjugué généralisé [37], elle est souvent appelée méthode du gradient conjugué préconditionné [47] ("Preconditioned conjugate Gradient" ou PCG dans la littérature anglo-saxonne).

Cette méthode peut s'écrire sous une autre forme :

Soit $\overset{o}{x}$ quelconque

$$M\overset{o}{z} = \overset{o}{r} = b - A\overset{o}{x}$$

$$\overset{o}{p} = \overset{o}{z} \quad , \quad a_0 = \frac{(\overset{o}{z}, M\overset{o}{z})}{(\overset{o}{p}, A\overset{o}{p})} \quad , \quad \overset{1}{x} = \overset{o}{x} + a_0 \overset{o}{p}$$

$\forall\, k = 1, 2, \ldots$

$$M z^k = r^k$$

$$c_k = \frac{(z^k, Mz^k)}{(z^{k-1}, Mz^{k-1})}$$

$$p^k = z^k + c_k\, p^{k-1}$$

$$a_k = \frac{(z^k, Mz^k)}{(p^k, Ap^k)}$$

$$x^{k+1} = x^k + a_k p^k$$

Lorsque $M = I$ , on retrouve la méthode originelle (Hestenes et Stiefel [43]) .

Revenons à l'analyse de la suite $\{t_{n,k}\}$ ; nous allons montrer que $t_{m,k}$ peut s'écrire comme un rapport de deux déterminants.

Par définition $\quad t_{n,k} = s_n + \sum_{i=0}^{k-1} \beta_i^{(m,k)} T^i \Delta s_m$ , et comme

$$(r(t_{m,k}), M T^j \Delta s_n) = 0 \qquad j = 0, \ldots k-1 \text{ , où}$$

$$r(t_{n,k}) = \Delta s_m + \sum_{i=0}^{k-1} \beta_i^{(m,k)} T^{i+1} \Delta s_m \text{ , on a :}$$

$$t_{m,k} = \frac{\begin{vmatrix} s_m & \Delta s_m & \cdots & \cdots & T^{k-1} \Delta s_m \\ (\Delta s_n, \Delta s_m)_M & (T\Delta s_n, \Delta s_m)_M & \cdots & \cdots & (T^k \Delta s_m, \Delta s_m)_M \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ (\Delta s_m, T^{k-1}\Delta s_m)_M & (T\Delta s_n, T^{k-1}\Delta s_m)_M & \cdots & \cdots & (T^k \Delta s_m, T^{k-1}\Delta s_m)_M \end{vmatrix}}{\begin{vmatrix} (T\Delta s_m, \Delta s_n)_M & \cdots & \cdots & (T^k \Delta s_n, \Delta s_m)_M \\ \cdots & \cdots & \cdots & \cdots \\ (T\Delta s_n, T^{k-1}\Delta s_m)_M & \cdots & \cdots & (T^k \Delta s_m, T^{k-1}\Delta s_m)_M \end{vmatrix}}$$

où

$$(x,y)_M = (x, My) .$$

Le déterminant du numérateur est développable suivant la première ligne.

Ce résultat nous servira pour montrer l'équivalence de la méthode du gradient conjugué avec des méthodes non linéaires qui seront developpées par la suite.

Il existe de nombreuses généralisations de la méthode du gradient conjugué ; c'est pourquoi nous allons nous contenter de décrire les généralisations qui utilisent les méthodes de projections.

## 5) Méthodes de projections :

Les méthodes que nous allons examiner sont basées sur la méthode de projection oblique.

### a) Méthode de projection oblique [68]

Soit à résoudre le système d'equations linéaires

$$Ax = b \quad \text{où} \quad x, b \in \mathbb{R}^p$$
$$A \in L(\mathbb{R}^p).$$

Soient $K_m$ et $L_m$ deux sous espaces vectoriels de dimension $m$ et engendrés respectivement par les vecteurs $v_1, —, v_m$ et $w_1, —, w_m$.
Posons $V_m = (v_1, —, v_m)$ et $W_m = (w_1, —, w_m)$. Pour résoudre le système d'equations, on utilise la méthode suivante :

$$\begin{cases} \overset{\circ}{x} \quad \text{donné}, \ r^0 = b - A\overset{\circ}{x} \\ x^m = \overset{\circ}{x} + z^m \quad \text{où} \quad z^m \in K_m \ \text{et} \\ \overset{\circ}{r} - Az^m \perp L_m \end{cases}$$

si $\det(W_m^T V_m) \neq 0$ et $\det(W_m^T A V_m) \neq 0$ alors :

$$x^m = \overset{\circ}{x} + V_m (W_m^T A V_m)^{-1} W_m^T \overset{\circ}{r}.$$

Nous sommes maintenant en mesure de présenter quelques généralisations de la méthode du gradient conjugué.

b) Méthode du gradient bi-conjugué [29].

Soit $\{s_n\}$ une suite défini par : $s_{n+1} = B s_n + c$, posons $T = B - I$.

Pour accélérer la convergence d'une telle suite, nous allons utiliser la méthode de projection oblique.

Posons $V_k^{(n)} = (r(s_n), Tr(s_n), \ldots, T^{k-1} r(s_n))$ avec $r(x) = c + Tx$

et

$W_k^{(n)} = (r(s_n), T^T r(s_n), \ldots, (T^T)^{k-1} r(s_n))$ et définissons la suite $\{t_{n,k}\}$ par :

$$t_{n,k} = s_n - V_k^{(n)} \left( W_k^{(n)T} T V_k^{(n)} \right)^{-1} W_k^{(n)} r(s_n)$$
$$= s_n + z_k^{(n)} .$$

$z_k^{(n)}$ appartient au sous espace vectoriel engendré par les colonnes de la matrice $V_k^{(n)}$, donc on a :

$$t_{n,k} = s_n + \sum_{i=0}^{k-1} \beta_i^{(n,k)} T^i r(s_n).$$

De plus on a :

$$r(s_n) + T z_k^{(n)} \perp V[r(s_n), T^T r(s_n), \ldots, (T^T)^{k-1} r(s_n)]$$

qui s'écrit :

$$\left( r(s_n) + T z_k^{(n)}, (T^T)^i r(s_n) \right) = 0 \qquad i = 0, \ldots, k-1 .$$

A partir de ces relations, on peut écrire $t_{n,k}$ sous forme d'un rapport de deux déterminants :

$$t_{n,k} = \frac{\begin{vmatrix} s_n & T\,r(s_n) & \cdots & T^{k-1} r(s_n) \\ (r(s_n), r(s_n)) & (T r(s_n), r(s_n)) & \cdots & (T^k r(s_n), r(s_n)) \\ \cdots & \cdots & \cdots & \cdots \\ (r(s_n), (T^T)^{k-1} r(s_n)) & (T r(s_n), (T^T)^{k-1} r(s_n)) & \cdots & (T^k r(s_n), (T^T)^{k-1} r(s_n)) \end{vmatrix}}{\begin{vmatrix} (T r(s_n), r(s_n)) & \cdots & (T^k r(s_n), r(s_n)) \\ \cdots & \cdots & \cdots \\ (T r(s_n), (T^T)^{k-1} r(s_n)) & \cdots & (T^k r(s_n), (T^T)^{k-1} r(s_n)) \end{vmatrix}}.$$

On note également qu'il existe un algorithme récursif qui permet le calcul des termes $t_{n,k}$ pour $n$ fixé [29].

c) <u>La méthode du résidu conjugué</u> : La méthode GCR (generalized conjugate residual method) [27]

Pour cette méthode, on prend $W_k^{(n)} = T\,V_k^{(n)}$ où
$V_k^{(n)} = (r(s_n), T r(s_n), \ldots, T^{k-1} r(s_n))$
La relation d'orthogonalité devient :

$$(r(s_n) + T z_k^{(n)}, T^i r(s_n)) = 0 \qquad i = 0, \ldots, k-1.$$

Et donc comme pour la méthode précédente, on peut exprimer $t_{n,k}$ sous forme d'un rapport de deux determinants, ~~et~~ on peut donner un algorithme récursif pour calculer les termes $t_{n,k}$ avec $n$ fixé.

d) méthode d'Arnoldi [68]

C'est une méthode de projection orthogonale, puisque $W_k^{(m)} = V_k^{(m)} = (r(s_m), Tr(s_m), \ldots, T^{k-1}r(s_m))$. $t_{m,k}$ s'écrit ici aussi comme un rapport de deux déterminants.

Remarques :

① Il existe d'autres généralisations de la méthode du gradient conjugué [37], [69], [82] [90]. Nous avons rappelé seulement celles qui ont un lien avec les méthodes d'accélération non linéaires.

② On trouvera dans [1] et [2] un formalisme général regroupant la plupart des méthodes de projection ainsi que des théorèmes de convergence pour les méthodes incomplètes.

③ tout au long de cette partie, nous avons transformé des méthodes qui sont habituellement utilisées pour la résolution des systèmes d'équations linéaires, en des méthodes d'accélération ou d'amélioration de la convergence.

La base de toutes les méthodes que l'on a vu est la relation :

(*) $\quad t_{m,k} = s_m + \sum_{i=0}^{k-1} \beta_i^{(m,k)} T^i \Delta s_n$. Le choix des nombres $\beta_i^{(m,k)}$ caractérise chaque méthode. Une relation analogue à l'égalité (*) sera aussi la base des méthodes non linéaires.

## II Méthodes non linéaires :

Dans cette partie, nous allons examiner des méthodes pour accélérer la convergence de suites générées linéairement, c'est à dire des suites qui ont la forme suivante :

$$s_{n+1} = B_n s_n + c_n \quad \text{où} \quad B_n \in L(\mathbb{C}^p)$$
$$s_n, c_n \in \mathbb{C}^p \quad .$$

Pour accélérer la convergence de telles suites, on peut évidemment appliquer des algorithmes scalaires ([9], [81]) aux suites de chaque composante des vecteurs. Ces méthodes ne seront pas étudiées ici. Nous allons nous intéresser seulement aux méthodes vectorielles qui sont, pour des raisons théoriques meilleures.

Nous avons décomposé les méthodes vectorielles en trois familles. Dans la première, nous avons regroupé les méthodes qui généralisent la transformation de Shanks [74] où l'$\varepsilon$-algorithme [93] du cas scalaire au cas vectoriel et qui seront appelées méthodes de type $\varepsilon$-algorithme. La seconde famille contient les méthodes qui sont basées sur le calcul des coefficients d'un certain polynôme minimal. Nous les appelerons méthodes polynomiales. Enfin, dans la dernière famille, nous avons rassemblé les généralisations du E-algorithme scalaire ; les méthodes de cette famille, seront appelés méthodes générales d'extrapolation.

# 1- Méthodes de type $\varepsilon$-algorithme.

## a) l'$\varepsilon$-algorithme vectoriel

### a.1) Définition

Soit $\{s_n\}$ une suite de vecteurs de $\mathbb{C}^p$. Les règles de l'$\varepsilon$-algorithme vectoriel de Wynn sont les suivantes [94] :

$$\left[\begin{array}{lll} \varepsilon_{-1}^{(n)} = 0 & \varepsilon_0^{(n)} = s_n & n = 0, 1, \ldots \\ \text{et} & & \\ \varepsilon_{k+1}^{(n)} = \varepsilon_{k-1}^{(n+1)} + \left[\Delta \varepsilon_k^{(n)}\right]^{-1} & & n, k = 0, 1, \ldots \end{array}\right.$$

où l'inverse $y^{-1}$ d'un vecteur $y$ de $\mathbb{C}^p$ est défini par :

$$y^{-1} = \frac{\overline{y}}{(y, y)} .$$

Remarques :

① On ne possède pas pour l'$\varepsilon$-algorithme vectoriel de définition des termes $\varepsilon_k^{(n)}$ à l'aide de déterminants.
Donc pour $k$ et $n$ fixés, on ne sait pas quel système d'equations linéaires on résoud contrairement aux autres algorithmes. Ceci explique peut être le fait que cet algorithme est très pauvre en résultats théoriques.

② Pour cet algorithme, il existe des règles particulières afin d'éviter la propagation des erreurs ainsi qu'une règle singulière à utiliser lorsque deux termes consécutifs sont très voisins [24].

③ Malgré le manque de support théorique, cet algorithme admet beaucoup d'applications, pour plus de détails voir [9], [10] a.2) Propriétés

Le résultat fondamental sur lequel repose la théorie algébrique de l'$\varepsilon$-algorithme vectoriel a été conjecturé par Wynn [95] et démontré dans le cas de coefficients réels par Mc Leod [58], et par Graves-Morris dans le cas complexe [39]. Ce résultat est le suivant :

théorème 2

Si on applique l'$\varepsilon$-algorithme vectoriel à une suite $\{s_m\}$ qui vérifie

$$\sum_{i=0}^{k} a_i \,( s_{m+i} - s) = 0 \qquad \forall n > N$$

où les $a_i$ sont des nombres complexes avec $a_k \neq 0$ alors :

$$\varepsilon_{2k}^{(n)} = s \qquad \forall n > N \qquad \text{si } \sum_{i=0}^{k} a_i \neq 0$$

$$\text{et} \quad \varepsilon_{2k}^{(n)} = 0 \qquad \forall n > N \qquad \text{sinon.}$$

Ce résultat admet beaucoup d'applications. Il est en particulier utilisé pour montrer que l'$\varepsilon$-algorithme peut être utilisé comme méthode directe de résolution des systèmes d'équations linéaires. Avant d'énoncer ce résultat, nous avons besoin de la définition suivante :

définition :

Soit A une matrice carrée d'ordre $p$ et $x$ un vecteur donné de $\mathbb{C}^p$. On appelle polynôme minimal de A pour le vecteur $x$ le polynôme de plus petit degré tel que :

$$P(A)\,x = 0$$

Théorème 3

Si on applique l'$\varepsilon$-algorithme vectoriel à la suite $\{s_m\}$ produite par $s_{m+1} = A s_m + b$ avec $s_0$ donné où $A$ est une matrice carrée complexe, $b$ et $s_0$ deux vecteurs complexes et si la matrice $I - A$ est inversible alors :

$$\varepsilon_{2(m-r)}^{(n+r)} = (I-A)^{-1} b = s \qquad n = 0, 1, \ldots$$

où $m$ est le degré du polynome minimal de $A$ pour le vecteur $s_0 - s$ et $r$ la multiplicité de la racine nulle pour ~~ce~~ polynôme minimal.

ce théorème a été démontré par Gekeler [33] dans le cas où $r = 0$, et généralisé par Brezinski [6].

Remarques :

① On voit que l'$\varepsilon$-algorithme fournit une méthode directe de résolution d'équations linéaires. Cette méthode n'est pas intéressante dans la pratique car elle nécessite $2p^2$ mémoires dans l'ordinateur (si $m = p$ et $r = 0$) et un coût de l'ordre de $8p^3$ opérations.

② Nous pensons qu'une utilisation efficace pour remédier à ces problèmes serait d'employer cet l'algorithme d'une manière cyclique comme pour la résolution des systèmes non linéaires et de prendre $k$ très petit devant $p$ comme on le verra plus loin.

③ Cet algorithme n'est connecté à aucune méthode de gradient conjugué comme l'a montré Wynn [95].

④ En appliquant l'ε-algorithme vectoriel aux suites définies par $s_{n+1} = A s_n$, Brezinski a donné une généralisation de la méthode de la puissance pour calculer toutes les valeurs propres de la matrice A (et ceci sous certaines hypothèses) [7]. Malheureusement cette méthode semble très instable numériquement [67].

Donnons un théorème d'accélération de la convergence pour certaines suites générées linéairement

théorème 4 [9]

Soit $\{s_m\}$ une suite de vecteurs réels générés par :

$$s_0 \quad \text{donné}$$

$$s_{m+1} = A s_m + b$$

où A est une matrice carrée telle que $I - A$ soit inversible et b un vecteur. Soient $\lambda_1, \lambda_2, \dots, \lambda_p$ les valeurs propres de A. Supposons que $|\lambda_1| > |\lambda_2| > \dots > |\lambda_p|$ et posons $s = (I - A)^{-1} b$. Si on applique l'ε-algorithme vectoriel à la suite $\{s_n\}$ alors :

$$\| \varepsilon_{2k}^{(m)} - s \| = O(\lambda_{k+1}^m) \qquad k = 0, \dots, p-1$$

On voit que même si la suite $\{s_n\}$ diverge ($|\lambda_1| < 1$), la suite $\{\varepsilon_{2k}^{(m)}\}_n$ peut converger.

Du fait que $\varepsilon_{2k}^{(m)}$ n'est pas défini comme un rapport de deux déterminants, on ne sait même pas quand ce terme est défini. Pour pallier à cet inconvénient et au manque de résultats théoriques, Brezinski a proposé l'$\varepsilon$-algorithme topologique.

b) l'$\varepsilon$-algorithme topologique

b.1) <u>premiere généralisation de la transformation de Shanks</u> [9]

soit $\{s_m\}$ une suite de vecteurs de $\mathbb{C}^p$ et $s \in \mathbb{C}^p$. Supposons que la suite $\{s_m\}$ vérifie la relation suivante :

$$(1) \quad \sum_{i=0}^{k} a_i \, (s_{n+i} - s) = 0 \qquad n = 0, 1, \ldots$$

où les $a_i$ sont des nombres complexes tels que $a_k \neq 0$ et $\sum_{i=0}^{k} a_i \neq 0$.

Comme par hypothèse $\sum_{i=0}^{k} a_i \neq 0$, posons $b_i = \frac{a_i}{\sum_{i=1}^{k} a_i}$ on a

donc $\sum_{i=0}^{k} b_i = 1$, et alors $s$ est la solution du système suivant :

$$\begin{cases} b_0 + b_1 + \ldots + b_k = 1 \\ b_0 s_{n+i} + \ldots + b_k s_{n+k+i} = s \qquad i = 0, \ldots, k \end{cases}$$

Ce système peut encore s'écrire :

$$* \begin{cases} b_0 + b_1 + \ldots + b_k = 1 \\ b_0 \Delta s_{n+i} + \ldots + b_k \Delta s_{n+k+i} = 0 \qquad i = 0, \ldots, k-1 \quad (2) \\ b_0 s_n + b_1 s_{n+1} + \ldots + b_k s_{n+k} = s \qquad (3) \end{cases}$$

Multiplions scalairement les relations (2) par un vecteur $y$, on obtient un système d'équations linéaires (en ne considérant que les $k+1$ premières lignes) que l'on résoud pour obtenir les coefficients $b_i$ qui seront reportés dans la relation (3). On obtient donc $s$, qui peut être écrit comme un rapport de deux déterminants, ce qui va nous permettre de définir une transformation qui généralise celle de Shanks.

$$e_k(s_n) = \frac{\begin{vmatrix} s_n & \cdots & s_{n+k} \\ (y, \Delta s_n) & \cdots & (y, \Delta s_{n+k}) \\ \cdots & \cdots & \cdots \\ (y, \Delta s_{n+k-1}) & \cdots & (y, \Delta s_{n+2k-1}) \end{vmatrix}}{\begin{vmatrix} 1 & \cdots & 1 \\ (y, \Delta s_n) & \cdots & (y, \Delta s_{n+k}) \\ \cdots & \cdots & \cdots \\ (y, \Delta s_{n+k-1}) & \cdots & (y, \Delta s_{n+2k-1}) \end{vmatrix}}.$$

Définissons maintenant l'$\varepsilon$-algorithme topologique par ses règles :

$$\begin{cases} \varepsilon_{-1}^{(n)} = 0 \quad , \quad \varepsilon_0^{(n)} = s_n & n = 0, 1, \ldots \\ \varepsilon_{2k+1}^{(n)} = \varepsilon_{2k-1}^{(n)} + \dfrac{y}{(y, \Delta \varepsilon_{2k}^{(n)})} & n, k = 0, 1, \ldots \\ \varepsilon_{2k+2}^{(n)} = \varepsilon_{2k}^{(n+1)} + \dfrac{\Delta \varepsilon_{2k}^{(n)}}{(\Delta \varepsilon_{2k+1}^{(n)}, \Delta \varepsilon_{2k}^{(n)})} & \end{cases}$$

l'opérateur $\Delta$ porte sur les indices supérieurs.

Le lien entre l'$\varepsilon$-algorithme et la généralisation de la transformation de Shanks est donnée par le :

<u>théorème 5</u>

$$\boxed{\begin{aligned} & \varepsilon_{2k}^{(n)} = e_k(s_n) \\ \text{et} & \qquad\qquad\qquad\qquad n, k = 0, 1, \ldots \\ & \varepsilon_{2k+1}^{(n)} = \frac{y}{(y, e_k(\Delta s_n))} \end{aligned}}$$

Avant de donner des résultats analogues à ceux donnés pour l'$\varepsilon$-algorithme vectoriel, faisons les remarques suivantes:

Remarques:

① La transformation $e_k(s_n)$, dans le cas où $y$ est un vecteur quelconque de la base canonique, a été aussi définie par Pugachev [63], pour accélérer la convergence de suites générées linéairement.

② Comme pour l'$\varepsilon$-algorithme vectoriel le calcul de $\varepsilon_{2k}^{(n)}$ s'effectue à partir de $s_n, s_{n+1}, \dots, s_{n+2k}$.

③ l'$\varepsilon$-algorithme topologique, possède presque les mêmes propriétés que l'$\varepsilon$-algorithme scalaire, et ceci est dû au fait que l'on peut généraliser le déterminant de Hankel classique de la manière suivante [8]:

Soit $\{u_n\}$ une suite de vecteurs complexes de $\mathbb{C}^p$.

$$\tilde{H}_{k+1}^{(n)}(u_n) = \begin{vmatrix} u_n & \dots & u_{n+k} \\ (y, \Delta u_n) & \dots & (y, \Delta u_{n+k}) \\ \dots & \dots & \dots \\ (y, \Delta u_{n+k-1}) & \dots & (y, \Delta u_{n+2k-1}) \end{vmatrix}$$

On a:

$$\varepsilon_{2k}^{(n)} = \frac{\tilde{H}_{k+1}^{(n)}(s_n)}{H_k^{(n)}(\Delta s_n)} \quad \text{où} \quad H_k^{(n)}(u_n) = (y, \tilde{H}_k^{(n)}(u_n))$$

$H_k^{(n)}$ est le déterminant de Hankel classique de la suite $(y, u_n)$.

Par construction, nous avons le résultat suivant :

Théorème 6 [8]

Une condition suffisante pour que $e_k(s_n) = s$ $\forall n > N$ est que la suite $\{s_n\}$ vérifie la relation,

$$\sum_{i=0}^{k} a_i (s_{n+i} - s) = 0 \qquad \forall n > N \text{ , avec } a_i \in \mathbb{C} \text{ et } \sum_{i=0}^{k} a_i \neq 0$$

Grace à ce théorème, on peut démontrer un résultat analogue au théorème 3 et donc l' $\varepsilon$-algorithme topologique peut aussi être utilisé comme méthode directe pour résoudre des systèmes d'équations linéaires. On peut aussi l'utiliser pour calculer les valeurs propres comme nous l'avons vu auparavant.

En ce qui conserne l'accélération de la convergence, nous allons donner un théorème qui est plus général que le théorème 4.

Examinons l'accélération de la convergence des suites $\{s_n\}$ telles que le terme général admet le développement suivant [78]

$$* \qquad s_n \sim s + \sum_{i=1}^{\infty} \lambda_i^n v_i$$ où, $s$, $s_n$ et $v_i$ sont des vecteurs $\lambda_i$ est un nombre complexe.

La signification de cette relation est la suivante :

$\forall N > 0$, $\exists K > 0$, $\exists n_0(N)$ tels que :

$$\forall n \geq n_0(N) \qquad \| s_n - s - \sum_{i=1}^{N-1} v_i \lambda_i^n \| \leq K |\lambda_N|^n \text{ .}$$

Donnons tout d'abord la définition suivante

**Définition :** la suite $\{s_n\}$ est un élément de $\mathcal{C}l$ si et seulement si :

1) $s_n \in \mathbb{C}^p$

2) $s_n \sim s + \sum_{i=1}^{\infty} \lambda_i^n v_i \quad \lambda_i \in \mathbb{C}$ et $v_i \in \mathbb{C}^p$

   a) $\lambda_i \neq \lambda_j \quad \forall i \neq j$

   b) $\lambda_i \neq 1 \quad \forall i$

   c) $|\lambda_1| \geq |\lambda_2| \geq \cdots$

   d) si $A_i = \{j \,/\, |\lambda_j| = |\lambda_i|\}$, alors l'ensemble $A_i$ admet un nombre fini d'éléments

Remarque

- si $s_{n+1} = B s_n + c$ $\quad$ $I - B$ inversible et $B$ diagonalisable
  et si $s = Bs + c$ et
  alors $\{s_n\} \in \mathcal{C}l$.

théorème 7 [78]

Soit $\{s_n\}$ une suite de $\mathcal{C}l$, Supposons que

a) $|\lambda_1| \geq \cdots \geq |\lambda_k| > |\lambda_{k+1}| \geq \cdots$

b) les vecteurs $\{v_i\}$ sont linéairements indépendants

c) $\forall\, i = 1, \ldots, k \qquad (y, v_i) \neq 0$

alors

$$\| \varepsilon_{2k}^{(n)} - s \| = O\left((\lambda_{k+1})^n\right) \qquad (n \to \infty)$$

- si $|\lambda_1| \leq 1$ alors la suite $\{s_n\}$ converge vers $s$ sinon la suite diverge et dans ce cas $s$ sera appelé l'anti-limite de la suite $\{s_n\}$. On voit que même si la suite initiale diverge ; la suite $\{\varepsilon_{2k}^{(n)}\}_n$ peut converger.

Le théorème 7 est bien un théorème d'accélération de la convergence puisque :

$$\frac{\| \varepsilon_{2k}^{(n)} - s \|}{\| s_{n+2k} - s \|} = O\left[ \left( \frac{d_{k+1}}{d_1} \right)^n \right] \qquad (n \longrightarrow +\infty)$$

Pour les suites appartenant à l'ensemble GL, on peut démontrer la stabilité asymptotique de l'ε-algorithme topologique. Cette notion est donnée par la :

**Définition** [78] : Soit $\{s_n\}$ une suite de vecteurs complexes. Soit $k$ un entier naturel fixé et $T_k$ une transformation de suites définie par : $T_k : \{s_n\} \longrightarrow \{T_k^{(n)}\}$

où

$$T_k^{(n)} = \sum_{j=0}^{k} \gamma_j^{(n,k)} s_{n+j+q}$$

avec

$$\sum_{j=0}^{k} \gamma_j^{(n,k)} = 1 \quad \text{et } q \text{ un entier naturel fixé.}$$

$T_k$ est dite asymptotiquement stable si et seulement si $\sup_n \sum_{j=0}^{k} |\gamma_j^{(n,k)}| < \infty$

Pour les suites de GL, on peut montrer le résultat suivant :

**Théorème 8** [78]

Soit $\{s_n\}$ une suite de l'ensemble GL. Si $\{s_n\}$ vérifie les hypothèses du théorème 7 alors l'ε-algorithme topologique est asymptotiquement stable.

Les théorèmes 7 et 8 ont été étendus récemment à une classe.

## Lien entre la méthode du gradient conjugué et l'$\varepsilon$-algorithme topologique

Etudions maintenant le lien existant entre l'$\varepsilon$-algorithme topologique et la méthode du gradient conjugué quand la suite $\{s_n\}$ est générée linéairement :

$$s_{n+1} = B s_n + c \quad \text{où } B \in \mathcal{L}(\mathbb{R}^p) \text{ et } s_n \in \mathbb{R}^p.$$

Commencons tout d'abord par le cas symétrique.

1) Posons $B = M^{-1}N$

$$A = M - N \quad , \quad c = M^{-1}b \quad \text{et} \quad T = B - I.$$

et supposons que la matrice $I-B$ est inversible, la solution $s$ du système $x = Bx + c$ est soit la limite où l'anti-limite de la suite $\{s_n\}$.

Supposons de plus que les matrices $M$ et $A$ soient symétriques définies positives.

Prenons $s_0 = 0$ et notons $\varepsilon_{2k}^{(0)} = x_k$. Soit $\{y_k\}$ la suite fournie par la méthode du gradient conjugué préconditionné.

Nous avons vu précedemment que $y_k$ peut être mis sous la forme d'un rapport de deux déterminants :

$$y_k = \begin{vmatrix} s_0 & \Delta s_0 & \cdots & T^{k-1}\Delta s_0 \\ (\Delta s_0, \Delta s_0)_M & (T\Delta s_0, \Delta s_0)_M & \cdots & (T^k\Delta s_0, \Delta s_0)_M \\ \cdots & \cdots & \cdots & \cdots \\ (\Delta s_0, T^{k-1}\Delta s_0)_M & (T\Delta s_0, T^{k-1}\Delta s_0)_M & \cdots & (T^k\Delta s_0, T^{k-1}\Delta s_0)_M \end{vmatrix} \Bigg/ \begin{vmatrix} (T\Delta s_0, \Delta s_0)_M & \cdots & (T^k\Delta s_0, \Delta s_0)_M \\ \cdots & \cdots & \cdots \\ (T\Delta s_0, T^{k-1}\Delta s_0)_M & \cdots & (T^k\Delta s_0, T^{k-1}\Delta s_0)_M \end{vmatrix}$$

de même

$$x_k = \begin{vmatrix} s_0 & \Delta s_0 & \cdots & \Delta^k s_0 \\ (y,\Delta s_0) & (y,\Delta^2 s_0) & \cdots & (y,\Delta^{k+1} s_0) \\ \vdots & & & \\ (y,\Delta^k s_0) & (y,\Delta^{k+1} s_0) & \cdots & (y,\Delta^{2k} s_0) \end{vmatrix} \Big/ \begin{vmatrix} (y,\Delta^2 s_0) & \cdots & (y,\Delta^{k+1} s_0) \\ \cdots & \cdots & \cdots \\ (y,\Delta^{k+1} s_0) & \cdots & (y,\Delta^{2k} s_0) \end{vmatrix}.$$

et comme $\Delta s_0 = c$ et $\Delta^{i+1} s_0 = T^i \Delta s_0$

on a $(T^i \Delta s_0, T^j \Delta s_0)_M = (Mc, T^{i+j} c) = (b, T^{i+j} c)$

et $(y, \Delta^{i+1} s_0) = (y, T^i c)$ .

<u>théorème 9</u>

Soient $A$ et $M$ deux matrices symétriques définies positives et soit $\{y_k\}$ la suite obtenue en appliquant la méthode du gradient conjugué préconditionnée au système $Ax=b$ avec $y_0=0$.
Soit $B = I - M^{-1}A$ , $s_0 = 0$ et $s_{n+1} = B s_n + M^{-1} b$ et soit $\{\varepsilon_{2k}^{(0)}\}_k$ la suite obtenue en appliquant l'$\varepsilon$-algorithme topologique à la suite $\{s_n\}$ avec $y=b$ alors

$$y_k = \varepsilon_{2k}^{(0)} \qquad \forall\ k = 0, 1, \ldots$$

Ce théorème généralise le théorème 4.1 p 186 de [13], que l'on retrouve si $M=I$.

2) Lorsque les matrices $A$ et $M$ sont quelconques, on peut montrer qu'avec les mêmes notations que le théorème 9, la méthode du gradient bi-conjugué et la méthode basée sur l'$\varepsilon$-algorithme topologique génèrent des suites identiques [13].

b.2) seconde généralisation [8]

Si l'on remplace dans le système *(p17) la relation (3) par (3)' :

$$b_0\, s_{n+k} + \cdots + b_k\, s_{n+2k} = s \qquad (3)'$$

et si on utilise la même technique pour exprimer $s$, on obtient une seconde transformation définie par :

$$e_k(s_n) = \frac{\begin{vmatrix} s_{n+k} & \cdots & s_{n+2k} \\ (y, \Delta s_n) & \cdots & (y, \Delta s_{n+k}) \\ \cdots & \cdots & \cdots \\ (y, \Delta s_{n+k-1}) & \cdots & (y, \Delta s_{n+2k-1}) \end{vmatrix}}{\begin{vmatrix} 1 & \cdots & 1 \\ (y, \Delta s_n) & \cdots & (y, \Delta s_{n+k}) \\ \cdots & \cdots & \cdots \\ (y, \Delta s_{n+k-1}) & \cdots & (y, \Delta s_{n+2k-1}) \end{vmatrix}}$$

Pour calculer cette quantités, on utilise un second ε-algorithme topologique dont les règles sont les suivantes :

$$\begin{cases} \varepsilon_{-1}^{(n)} = 0 \quad , \quad \varepsilon_0^{(n)} = s_n \\ \varepsilon_{2k+1}^{(n)} = \varepsilon_{2k-1}^{(n+1)} + \dfrac{y}{(y, \Delta\varepsilon_{2k}^{(n)})} \\ \varepsilon_{2k+2}^{(n)} = \varepsilon_{2k}^{(n+1)} + \dfrac{\Delta\varepsilon_{2k}^{(n+1)}}{(\Delta\varepsilon_{2k+1}^{(n)}, \Delta\varepsilon_{2k}^{(n+1)})} \end{cases}$$

On a le résultat fondamental suivant :

<u>théorème 10[ ]</u> $\boxed{\varepsilon_{2k}^{(n)} = \tilde{e}_k(s_n) \quad \text{et} \quad \varepsilon_{2k+1}^{(n)} = \frac{y}{(y, \tilde{e}_k(s_n))}}$.

Pour ce second $\varepsilon$-algorithme, les théorèmes 6, 7 et 8 sont encore valables.

D'après les essais numériques donnés par Tan dans [84] il semble que le second $\varepsilon$-algorithme topologique soit meilleur que le premier.

c) <u>Transformation de Germain-Bonne</u>

Une autre généralisation de la transformation de Shanks scalaire a été définie par Germain-Bonne [35, propriété 12, p71].

Si au lieu de multiplier scalairement les relations (2) du système * par le vecteur y fixe, nous les multiplions par $\Delta s_n$ alors nous obtenons une nouvelle transformation définie par :

$$\bar{e}_k(s_n) = \frac{\begin{vmatrix} s_n & \dots & s_{n+k} \\ (\Delta s_n, \Delta s_n) & \dots & (\Delta s_n, \Delta s_{n+k}) \\ \dots & \dots & \dots \\ (\Delta s_n, \Delta s_{n+k-1}) & \dots & (\Delta s_n, \Delta s_{n+2k-1}) \end{vmatrix}}{\begin{vmatrix} 1 & \dots & 1 \\ (\Delta s_n, \Delta s_n) & \dots & (\Delta s_n, \Delta s_{n+k}) \\ \dots & \dots & \dots \\ (\Delta s_m, \Delta s_{m+k-1}) & \dots & (\Delta s_m, \Delta s_{m+2k-1}) \end{vmatrix}}$$

Pour cette transformation, le théorème 6 est encore valable ainsi que les théorèmes 7 et 8 (sans l'hypothèse c)).

Lorsque n et k sont fixés, on peut utiliser l'algorithme de la transformation G [66] pour calculer $\bar{e}_k(s_n)$ [13] :

Posons
$$x_0 = s_n \qquad c_0 = (\Delta s_n, \Delta s_n)$$
$$x_1 = s_{n+1} \qquad c_1 = (\Delta s_n, \Delta s_{n+1})$$
$$\cdots \qquad \cdots$$
$$x_k = s_{n+k} \qquad c_{2k-1} = (\Delta s_n, \Delta s_{n+2k-1})$$

$$\text{alors} \quad \bar{e}_k(s_n) = G_k^{(0)} .$$

où l'algorithme de la transformation G est définit par :

$$\begin{cases} s_0^{(n)} = 1 \quad , \quad r_1^{(n)} = c_n \quad , \quad G_0^{(n)} = x_n \\ \dfrac{s_k^{(n+1)}}{s_k^{(n)}} = 1 + \dfrac{r_{k+1}^{(n)}}{r_k^{(n+1)}} \quad \text{et} \quad \dfrac{r_{k+1}^{(n+1)}}{r_{k+1}^{(n)}} = 1 + \dfrac{s_{k+1}^{(n)}}{s_k^{(n+1)}} \qquad \begin{array}{l} \text{pour } n=0,1,\ldots \\ k=1,\ldots \end{array} \\ G_{k+1}^{(n)} = -\dfrac{s_k^{(n+1)}}{s_{k+1}^{(n)}} \left( G_k^{(n+1)} - \dfrac{r_{k+1}^{(n+1)}}{r_{k+1}^{(n)}} G_k^{(n)} \right) \end{cases}$$

L'inconvénient a cette méthode, est que, si l'on change de valeur de k ou de n, tous les calculs sont à recommencer.

Notons que l'on peut utiliser l'algorithme de la transformation G pour mettre en oeuvre les deux généralisations de la transformation de Shanks pour k et n fixés. Le calcul de $e_k(s_n)$ où de $\bar{e}_k(s_n)$ est plus économique en utilisant l'algorithme G qu'en utilisant l'$\varepsilon$-algorithme. Lorsque l'on veut calculer une diagonale, c'est à dire $\varepsilon_{2k}^{(n)}$ pour n fixé et $k=0,1,\ldots$, il existe des méthodes plus économiques [13].

## 2) Méthodes polynomiales

Soit $\{s_n\}$ une suite de vecteurs complexes telque :

$$\begin{cases} s_0 \text{ donné} \\ s_{n+1} = B s_n + c \end{cases}$$

B et c ne sont pas nécessairement connus.
Supposons que $I - B$ soit inversible ; il existe donc un unique point $s$ tel que : $\quad s = Bs + c$ .

Si la suite $\{s_n\}$ converge alors $s$ est sa limite.
Si la suite $\{s_n\}$ diverge alors $s$ est appelé anti-limite [23].

Soit $P$ le polynôme minimal de B pour le vecteur $\Delta s_0$ et soit $k$ son degré :

$$P(\lambda) = \sum_{j=0}^{k} d_j \lambda^j \qquad d_k \neq 0$$

Soit $r$ l'entier naturel telque :

$$P(\lambda) = \lambda^r Q(\lambda) = \lambda^r \sum_{j=r}^{k} d_j \lambda^{j-r} \qquad \begin{cases} d_r \neq 0 \\ \text{et } d_k \neq 0 \end{cases}$$

**Lemme 1** [ ] : Le polynome minimal $P_j$ pour le vecteur $\Delta s_j$ est défini par :

$$\begin{cases} P_j(\lambda) = \lambda^{r-j} Q(\lambda) & j = 0, - r . \\ P_j(\lambda) = Q(\lambda) & j > r \end{cases}$$

De plus le polynôme minimal de B pour le vecteur $\Delta s_j$ est le même que celui de B pour le vecteur $s_j - s$.

D'après ce lemme

$$\sum_{j=0}^{k} d_j \; B^j \; (s_n - s) = 0$$

et comme $s_{n+j} = B^j(s_n - s) + s$

alors :

$$\sum_{j=0}^{k} d_j \, (s_{n+j} - s) = 0$$

Par conséquent :

$$\left(\sum_{j=0}^{k} d_j\right) s = \sum_{j=0}^{k} d_j \, s_{n+j} \, .$$

Donc pour avoir $s$, il faut déterminer les coefficients $d_j$.

D'après le lemme précédent

$$\sum_{j=0}^{k} d_j \, B^j \, \Delta s_n = 0$$

et comme $\Delta s_{n+j} = B^j \, \Delta s_n$.

on a donc $\sum_{j=0}^{k} d_j \, \Delta s_{n+j} = 0$

$s$ est donc déterminé de la manière suivante :

1) Trouver $d_0, \ldots, d_k$ vérifiant :

$$\sum_{j=0}^{k} d_j \, \Delta s_{n+j} = 0 \quad \text{et} \quad d_k \neq 0 \qquad (*)$$

2) Prendre $s = \left(\sum_{j=0}^{k} d_j \, s_{n+j}\right) \times \left(\sum_{j=0}^{k} d_j\right)^{-1}$

Le système * contient en général plus d'équations que d'inconnues. Les méthodes que l'on va maintenant examiner sont caractérisées par la méthode de résolution du système * choisi.

## a) M.P.E (Minimal polynomial extrapolation)

Cette méthode a été définie par Vorobyev [88], Germain Bonne [34] Cabay et Jackson [23], chacun a donné une approche différente.

Posons dans ⊛ $c_j = \frac{d_j}{d_k}$ , On obtient :

$$\sum_{j=0}^{k} c_j \, \Delta s_{n+j} = 0 \quad \text{et} \quad c_k = 1 ,$$

Ce système peut s'écrire :

$$\sum_{j=0}^{k-1} c_j \, \Delta s_{n+j} = - \Delta s_{n+k} .$$

La méthode MPE consiste à résoudre ce système par la méthode des moindres carrés, et donc :

$$c = -\Delta S_{m,k}^{+} \, \Delta s_{m+k}$$ avec les notations

suivantes :

$$\Delta S_{m,k} = (\Delta s_m, \text{—}, \Delta s_{n+k-1}) \;,\; c = (c_0, -, c_{k-1})^T$$

et $A^+$ étant le pseudo-inverse de la matrice A [52]

<u>La méthode M.P.E consiste à :</u>

1) se donner $s_m, s_{n+1}, \text{—}, s_{m+k+1}$

2) calculer a) $\Delta S_{m,k} = (\Delta s_m, \ldots, \Delta s_{n+k-1})$

b) $\Delta s_{m+k}$

c) $c = -\Delta S_{m,k}^{+} \, \Delta s_{m+k}$

2) Poser $c_k = 1$

et calculer $t_{m,k} = \left(\sum_{j=0}^{k} c_j \, s_{m+j}\right)\left(\sum_{j=0}^{k} c_j\right)^{-1}$.

La présentation que l'on vient de donner a été faite dans [81].

Donnons maintenant une présentation similaire à celle faite pour les généralisations de la transformations de Shanks.

Posons dans * $\beta_j = \frac{\alpha_j}{\sum_{i=0}^{k} \alpha_i}$. On obtient:

$$s = \sum_{j=0}^{k} \beta_j \, s_{n+j} \quad (1)$$

et

$$\begin{cases} \sum_{j=0}^{k} \beta_j = 1 & (2) \\ \sum_{j=0}^{k} \beta_j \, \Delta s_{n+j} = 0 & (3) \end{cases}$$

Multiplions scalairement la relation (3) par $\Delta s_{n+\ell}$ pour $\ell = 0, \dots, k-1$; On obtient le système d'équations linéaires suivant:

$$\begin{cases} \beta_0 + \beta_1 + \dots + \beta_k = 1 \\ \beta_0 (\Delta s_n, \Delta s_n) + \dots + \beta_k (\Delta s_n, \Delta s_{n+k}) = 0 \\ \dots \\ \beta_0 (\Delta s_{n+k-1}, \Delta s_n) + \dots + \beta_k (\Delta s_{n+k-1}, \Delta s_{n+k}) = 0 \end{cases}$$

On détermine les $\beta_i$ que l'on reporte dans (1), pour obtenir $s$.

Posons $t_{n,k} = s$,

$$t_{n,k} = \frac{\begin{vmatrix} s_n & \dots & s_{n+k} \\ (\Delta s_n, \Delta s_n) & \dots & (\Delta s_n, \Delta s_{n+k}) \\ \dots & \dots & \dots \\ (\Delta s_{n+k-1}, \Delta s_n) & \dots & (\Delta s_{n+k-1}, \Delta s_{n+k}) \end{vmatrix}}{\begin{vmatrix} 1 & \dots & 1 \\ (\Delta s_n, \Delta s_n) & \dots & (\Delta s_n, \Delta s_{n+k}) \\ \dots & \dots & \dots \\ (\Delta s_{n+k-1}, \Delta s_n) & \dots & (\Delta s_{n+k-1}, \Delta s_{n+k}) \end{vmatrix}}$$

ce qui peut aussi s'écrire :

$$t_{n,k} = s_n - \Delta S_{n,k} \left( \Delta S_{n,k}^* \, \Delta^2 S_{n,k} \right)^{-1} \Delta S_{n,k}^* \, \Delta s_n .$$

Nous donnerons des résultats théoriques à la fin de cette section. Rappelons les autres méthodes polynomiales.

b) R.R.E (Reduced rank extrapolation)

Cette méthode est due a Eddy [26] et Mešina [59]. Elle a été aussi implicitement suggérée par Peguchev dans [65].

Si nous multiplions scalairement la relation (3) par $\Delta^2 s_{n+l}$ pour $l = 0, \ldots, k-1$, nous obtenons le système d'équations linéaires suivant :

$$\left\{ \begin{array}{l} \beta_0 + \beta_1 + \cdots + \beta_k = 1 \\ \beta_0 (\Delta^2 s_n, \Delta s_n) + \cdots + \beta_k (\Delta^2 s_n, \Delta s_{n+k}) = 0 \\ \cdots\cdots\cdots\cdots\cdots \\ \beta_0 (\Delta^2 s_{n+k-1}, \Delta s_n) + \cdots + \beta_k (\Delta^2 s_{n+k-1}, \Delta s_{n+k}) = 0 \end{array} \right. \qquad (4)$$

et en posant $t_{n,k} = s$ nous avons :

$$t_{n,k} = \frac{\begin{vmatrix} s_n & s_{n+1} & \cdots & s_{n+k} \\ (\Delta^2 s_n, \Delta s_n) & (\Delta^2 s_n, \Delta s_{n+1}) & \cdots & (\Delta^2 s_n, \Delta s_{n+k}) \\ \cdots & \cdots & \cdots & \cdots \\ (\Delta^2 s_{n+k-1}, \Delta s_n) & (\Delta^2 s_{n+k-1}, \Delta s_{n+1}) & \cdots & (\Delta^2 s_{n+k-1}, \Delta s_{n+k}) \end{vmatrix}}{\begin{vmatrix} 1 & 1 & \cdots & 1 \\ (\Delta^2 s_n, \Delta s_n) & \cdots & \cdots & (\Delta^2 s_n, \Delta s_{n+k}) \\ \cdots & \cdots & \cdots & \cdots \\ (\Delta^2 s_{n+k-1}, \Delta s_n) & \cdots & \cdots & (\Delta^2 s_{n+k-1}, \Delta s_{n+k}) \end{vmatrix}}$$

ce qui peut aussi s'ecrire :

$$t_{n,k} = s_n - \Delta S_{n,k} \left(\Delta^2 S^*_{n,k} \Delta^2 S_{n,k}\right)^{-1} \Delta^2 S^*_{n,k} \Delta s_n .$$
$$= s_n - \Delta S_{n,k} \left(\Delta^2 S_{n,k}\right)^{+} \Delta s_n .$$

Une méthode très similaire a la méthode R.R.E a été obtenue par Kaniel et Stein [48].

Dans cette méthode les $\beta_i$ sont les mêmes que ceux de la méthode R.R.E c'est a dire que les $\beta_i$ $i=0,\ldots k$ sont solutions du système d'equations linéaires (4), mais la relation (1) a été remplacée par la relation :

$$s = \sum_{j=0}^{k} \beta_j \, s_{n+j+1} \text{ , on obtient :}$$

$$t_{n,k} = \frac{\begin{vmatrix} s_{n+1} & s_{n+2} & \cdots & s_{n+k+1} \\ (\Delta^2 s_n, \Delta s_n) & \cdots & \cdots & (\Delta^2 s_n, \Delta s_{n+k}) \\ \cdots & \cdots & \cdots & \cdots \\ (\Delta^2 s_{n+k-1}, \Delta s_n) & \cdots & \cdots & (\Delta^2 s_{n+k-1}, \Delta s_{n+k}) \end{vmatrix}}{\begin{vmatrix} 1 & \cdots & \cdots & 1 \\ (\Delta^2 s_n, \Delta s_n) & \cdots & \cdots & (\Delta^2 s_n, \Delta s_{n+k}) \\ \cdots & \cdots & \cdots & \cdots \\ (\Delta^2 s_{n+k-1}, \Delta s_n) & \cdots & \cdots & (\Delta^2 s_{n+k-1}, \Delta s_{n+k}) \end{vmatrix}}$$

$$= s_{n+1} - \Delta S_{n+1,k} \left(\Delta^2 S_{n,k}\right)^{+} \Delta s_n .$$

## c) MMPE (Modified Minimal Polynomial Extrapolation)

Cette méthode a été définie par Brezinski [8] et par Pugachev [65].

Nous allons donner la présentation faite par Brezinski.

Soient $y_1, \ldots, y_k$ $k$ vecteurs linéairement indépendants.
Multiplions scalairement la relation (3) par $y_{\ell+1}$ pour $\ell = 0, \ldots, k-1$, alors on a :

$$t_{n,k} = \frac{\begin{vmatrix} s_n & \cdots & s_{n+k} \\ (y_1, \Delta s_n) & \cdots & (y_1, \Delta s_{n+k}) \\ \vdots & & \vdots \\ (y_k, \Delta s_n) & \cdots & (y_k, \Delta s_{n+k}) \end{vmatrix}}{\begin{vmatrix} 1 & \cdots & 1 \\ (y_1, \Delta s_n) & \cdots & (y_1, \Delta s_{n+k}) \\ \vdots & & \vdots \\ (y_k, \Delta s_n) & \cdots & (y_k, \Delta s_{n+k}) \end{vmatrix}}$$

pour $y_i = e^i$ $i = 1, \ldots, k$, cette méthode a été utilisée par Pugachev [64], pour accélérer la convergence de suites générées linéairement.

Posons $Y_k = (y_1, \ldots, y_k) \in L(\mathbb{C}^p, \mathbb{C}^k)$. $t_{n,k}$ peut s'écrire sous la forme matricielle suivante :

$$t_{n,k} = s_n - \Delta S_n \left( {}^*Y_k \, \Delta^2 S_n \right)^{-1} Y_k^* \, \Delta s_n .$$

Germain-Bonne a défini et étudié une classe de méthodes qui contient comme cas particulier la méthode M.P.E, c'est ce que nous allons voir maintenant :

d) G.M.P.E (Generalized Minimal polynomial extrapolation)

Soit $\{s_n\}$ une suite de vecteurs complexes.

Supposons que $s_n \in \mathbb{C}^p$ et que les vecteurs $\Delta^q s_n, \ldots, \Delta^q s_{n+p-1}$ soient linéairement indépendants ($\forall n \geq 0$, $\forall q \geq 1$).

Posons :

$$t_{n,k}^{(q)} = s_n + \alpha_1 \Delta^q s_n + \alpha_2 \Delta^q s_{n+1} + \ldots + \alpha_k \Delta^q s_{n+k-1},$$

les $\alpha_i$ étant solution du système linéaire :

$$\begin{pmatrix} (\Delta s_n, \Delta s_n) \\ \\ \\ (\Delta s_{n+k-1}, \Delta s_n) \end{pmatrix} + \begin{pmatrix} (\Delta^2 s_n, \Delta^q s_n) & \ldots & (\Delta^2 s_n, \Delta^q s_{n+k-1}) \\ \vdots & \ldots & \vdots \\ (\Delta^2 s_{n+k-1}, \Delta^q s_n) & \ldots & (\Delta^2 s_{n+k-1}, \Delta^q s_{n+k-1}) \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_k \end{pmatrix} = 0.$$

d'écriture de $t_{n,k}^{(q)}$ sous forme d'un rapport de deux déterminants est la suivante :

$$t_{n,k}^{(q)} = \frac{\begin{vmatrix} s_n & \Delta^q s_n & \ldots & \Delta^q s_{n+k-1} \\ (\Delta s_n, \Delta s_n) & (\Delta^2 s_n, \Delta^q s_n) & \ldots & (\Delta^2 s_n, \Delta^q s_{n+k-1}) \\ \ldots & \ldots & \ldots & \ldots \\ (\Delta s_{n+k-1}, \Delta s_n) & (\Delta^2 s_{n+k-1}, \Delta^q s_n) & \ldots & (\Delta^2 s_{n+k-1}, \Delta^q s_{n+k-1}) \end{vmatrix}}{\begin{vmatrix} (\Delta^2 s_n, \Delta^q s_n) & \ldots & (\Delta^2 s_n, \Delta^q s_{n+k-1}) \\ \ldots & \ldots & \ldots \\ (\Delta^2 s_{n+k-1}, \Delta^q s_n) & \ldots & (\Delta^2 s_{n+k-1}, \Delta^q s_{n+k-1}) \end{vmatrix}}$$

pour $q = 1$ ; $t_{n,k}^{(1)}$ nécessite la connaissance de $s_n, \ldots, s_{n+k+1}$

$q = 2$ ; $t_{n,k}^{(2)}$ nécessite la connaissance de $s_n, \ldots, s_{n+k+1}$.

et pour $q > 2$ il faut utiliser $p+q$ termes de la suite $\{s_n\}$.

Posons $\Delta^q S_{m,k} = (\Delta^q s_m, \ldots, \Delta^q s_{m+k-1}) \qquad \forall q \geq 1$

alors

$$t_{m,k}^{(q)} = s_m - \Delta^q S_{m,k} \left(\overset{*}{\Delta^2} S_{m,k} \; \Delta^q S_{m,k}\right)^{-1} \overset{*}{\Delta} S_{m,k} \; \Delta s_m$$

Germain-Bonne a démontré le résultat d'accélération de la convergence suivant :

théorème 11 [34], [35]

Soit A une matrice symétrique, telle que $\|A\| < 1$.
soient $d_i$ ces valeurs propres (avec $|d_1| > |d_2| > \ldots > |d_i| > |d_{i+1}| > \ldots$) et $v_i$ les vecteurs propres associés ($(v_i, v_j) = 0$, si $i = j$).
Soit $\{s_n\}$ une suite d'elements de $\mathbb{R}^p$ engendrée par :

$$s_{n+1} = A s_n + b$$

avec $s_0$ tel que $(e_0, v_1) = a_1 \neq 0$
($e_0$ désigne l'erreur associée à $s_0$ : $s_0 - s$, $s$ étant solution de $s = As + b$)
alors

$$\|t_{m,k}^{(q)} - s\| = o(\|s_m - s\|) \qquad \forall k = 1, \ldots, p \quad \forall q \geq 1$$

Dans le cas réel :

$$t_{m,k}^{(q)} = s_n - \Delta^q S_{m,k} \left({}^T\Delta^q S_{m,k} \; \Delta^2 S_{m,k}\right)^{-1} {}^T\Delta S_{m,k} \; \Delta s_m$$

et si $q = 1$, on obtient la méthode M.P.E.

C'est pour les trois premières méthodes (MPE, RRE, MMPE) que l'on dispose du plus grand nombre de résultats théoriques.

c) Résultats théoriques.

Dans ce qui suit $\{t_{m,k}\}$ désigne la suite obtenue par l'une des 3 méthodes (MPE, RRE, MMPE).

théorème 12.

Une condition suffisante pour que $t_{m,k} = s$ $\forall n > N$ est que la suite $\{s_m\}$ vérifie la relation $\sum_{i=0}^{k} a_i (s_{m+i} - s) = 0$ $\forall n > N$ avec $a_i \in \mathbb{C}$ et $\sum_{i=0}^{k} a_i \neq 0$

Comme résultat d'accélération de la convergence nous avons :

théorème 13 [75], [78]

Soit $\{s_m\}$ une suite de GP. Supposons que

a) $|\lambda_1| \geq \cdots \geq |\lambda_k| > |\lambda_{k+1}| \geq \cdots$

b) les vecteurs sont linéairement indépendants

alors 1) $\| t_{m,k} - s \| = O\left( (\lambda_{k+1})^n \right)$ $(n \to +\infty)$

pour les deux méthodes MPE et RRE.

2) si on suppose de plus que le déterminant F est différent de zéro alors

$\| t_{m,k} - s \| = O\left( (\lambda_{k+1})^n \right)$ $(n \to +\infty)$

pour la méthode MMPE où,

$$F = \begin{vmatrix} (y_1, v_1) & \cdots & (y_1, v_k) \\ \cdots & & \\ (y_k, v_1) & \cdots & (y_k, v_k) \end{vmatrix}$$

Ce théorème a été récemment généralisé par Sidi et Bridger [77]. Ces deux auteurs se sont intéressés aux suites dont le terme général est tel que :

$$s_n \sim s + \sum_{i=1}^{\infty} P_i(n) \lambda_i^n \qquad (n \to +\infty)$$

où $P_i(n) \in \mathbb{C}^p$ et $\lambda_i \in \mathbb{C}$. $P_i(n)$ sont des polynomes en $n$. Cette classe de suites contient les suites générées linéairement :

$s_{n+1} = A s_n + b$ ou $A$ est defective (non diagonalisable).

En ce qui concerne la stabilité asymptotique nous avons le résultat suivant :

<u>théorème 14</u> [ ], [ ]

Soit $\{s_n\}$ une suite de l'ensemble Gl. Si $\{s_n\}$ vérifie les hypothèses du théorème 13. Alors les méthodes MPE, RRE et MMPE sont asymptotiquement stables.

Ce théorème a lui aussi été généralisé par Sidi et Bridger [77].

La connexion entre ces méthodes et les méthodes du gradient conjugué est la suivante [76] :

a) La méthode MPE coincide avec la méthode d'Arnoldi.

b) La méthode RRE coincide avec la méthode G.C.R.

et ceci quand la suite $\{s_n\}$ est généré linéairement :

$$s_{n+1} = B s_n + c$$

et les méthodes du gradient conjugué sont appliquées pour résoudre le système d'equations linéaires :

$$(I - B) x = c \quad .$$

## 3 - Méthodes générales d'extrapolation :

Comme généralisation du E-algorithme scalaire [13]. Wimp a proposé la méthode suivante :

### a) Méthode de Wimp [91]

Soit $\{s_n\}$ une suite de vecteurs complexes. Soient $\{\phi_n\}$ et $\{g_i(n)\}$ des suites arbitraires avec $\phi_n \in \mathbb{C}^p$ et $g_i(n) \in \mathbb{C}^p$ pour $i = 1, \dots k$.

Wimp a défini la transformation de suites suivante :

$$E_k(s_n) = s_k^{(n)} = \frac{\begin{vmatrix} s_n & 0 & g_1(n) & \cdots & g_k(n) \\ \langle \phi_n, s_n \rangle & 1 & \langle \phi_n, g_1(n) \rangle & \cdots & \langle \phi_n, g_k(n) \rangle \\ \vdots & \vdots & \cdots & & \cdots \\ \langle \phi_{n+k}, s_{n+k} \rangle & 1 & \langle \phi_{n+k}, g_1(n+k) \rangle & \cdots & \langle \phi_{n+k}, g_k(n+k) \rangle \end{vmatrix}}{\begin{vmatrix} 1 & \langle \phi_n, g_1(n) \rangle & \cdots & \langle \phi_n, g_k(n) \rangle \\ \vdots & & \cdots & \\ 1 & \langle \phi_{n+k}, g_1(n+k) \rangle & \cdots & \langle \phi_{n+k}, g_k(n+k) \rangle \end{vmatrix}}$$

En général il n'y a pas d'algorithme récursif pour calculer $s_k^{(n)}$, par contre $\langle \phi_n, s_n^{(k)} \rangle$ peut être calculé par le E-algorithme scalaire.

Posons $\qquad \tilde{s}_k^{(n)} = \langle \phi_n, s_k^{(n)} \rangle \quad ; \quad \tilde{s}_k^{(n)} \in \mathbb{C}.$

et

$$\tilde{g}_{k,i}^{(n)} = \langle \phi_n, E_k(g_i(n)) \rangle \quad ; \quad \tilde{g}_{k,i}^{(n)} \in \mathbb{C}.$$

pour $i > k$

on a donc :

$$\tilde{g}_{k,i}^{(n)} = \frac{\begin{vmatrix} \langle \phi_n , g_i(n) \rangle & \langle \phi_n , g_1(n) \rangle & \cdots & \langle \phi_n , g_k(n) \rangle \\ \cdots & \cdots & \cdots & \cdots \\ \langle \phi_{n+k} , g_i(n+k) \rangle & \langle \phi_{n+k} , g_1(n+k) \rangle & \cdots & \langle \phi_{n+k} , g_k(n+k) \rangle \end{vmatrix}}{\begin{vmatrix} 1 & \langle \phi_n , g_1(n) \rangle & \cdots & \langle \phi_n , g_k(n) \rangle \\ \cdots & \cdots & \cdots & \cdots \\ 1 & \langle \phi_{n+k} , g_1(n+k) \rangle & \cdots & \langle \phi_{n+k} , g_k(n+k) \rangle \end{vmatrix}}$$

et d'après les règles du E-algorithme scalaire nous avons :

$$\left[\begin{array}{l} \tilde{S}_0^{(n)} = \langle \phi_n , S_n \rangle \qquad \tilde{g}_{0,i}^{(n)} = \langle \phi_n , g_i(n) \rangle \\[2ex] \tilde{S}_k^{(n)} = \dfrac{\tilde{g}_{k-1,k}^{(n+1)} \, \tilde{S}_{k-1}^{(n)} - \tilde{g}_{k-1,k}^{(n)} \, \tilde{S}_{k-1}^{(n+1)}}{\tilde{g}_{k-1,k}^{(n+1)} - \tilde{g}_{k-1,k}^{(n)}} \\[2ex] \text{et} \\[1ex] \tilde{g}_{k,i}^{(n)} = \dfrac{\tilde{g}_{k-1,k}^{(n+1)} \, \tilde{g}_{k-1,i}^{(n)} - \tilde{g}_{k-1,k}^{(n)} \, g_{k-1,i}^{(n+1)}}{\tilde{g}_{k-1,k}^{(n+1)} - \tilde{g}_{k-1,k}^{(n)}} \end{array}\right.$$

De plus si on pose $r_k^{(n)} = \langle \phi_n , S_k^{(n)} - S \rangle$

alors :

$$r_0^{(n)} = \langle \phi_n , S_n - S \rangle$$

$$r_k^{(n)} = \frac{\tilde{g}_{k-1,k}^{(n)} \, \tilde{r}_{k-1}^{(n)} - g_{k-1,k}^{(n)} \, r_{k-1}^{(n+1)} - \Delta \langle \phi_n , S \rangle \, g_{k-1,k}^{(n)}}{g_{k-1,k}^{(n+1)} - \tilde{g}_{k-1,k}^{(n)}}$$

des résultats qui sont vérifiés par le E-algorithme scalaire se généralisent facilement dans le cas vectoriel.

Examinons maintenant le cas où la suite $\{\phi_n\}$ est constante.

b) <u>E-algorithme vectoriel</u> [12].

Dans le cas où $\phi_m = y$, la transformation précédente devient :

$$E_k^{(m)} = E_k(s_m) = \frac{\begin{vmatrix} s_m & g_1(m) & \cdots & g_k(m) \\ \langle y, \Delta s_m\rangle & \langle y, \Delta g_1(m)\rangle & \cdots & \langle y, \Delta g_k(m)\rangle \\ \cdots & & & \cdots \\ \langle y, \Delta s_{n+k-1}\rangle & \langle y, \Delta g_1(m+k-1)\rangle & \cdots & \langle y, \Delta g_k(m+k-1)\rangle \end{vmatrix}}{\begin{vmatrix} \langle y, \Delta g_1(m)\rangle & \cdots & \langle y, \Delta g_k(m)\rangle \\ \cdots & & \cdots \\ \langle y, \Delta g_1(m+k-1)\rangle & \cdots & \langle y, \Delta g_k(m+k-1)\rangle \end{vmatrix}}$$

pour calculer les termes $E_k(s_m)$ on peut utiliser le E-algorithme vectoriel dont les règles sont :

$$\left[\begin{array}{ll} E_0^{(m)} = s_m & m = 0, 1, \ldots \\ g_{0,i}^{(m)} = g_i(m) & i = 1, 2, \ldots \text{ et } m = 0, 1, \ldots \\ \text{pour } k = 1, 2, \ldots \text{ et } m = 0, 1, \ldots & \\ E_k^{(m)} = E_{k-1}^{(m)} - \dfrac{\langle y, \Delta E_{k-1}^{(m)}\rangle}{\langle y, \Delta g_{k-1}^{(m)}\rangle}\, g_{k-1,k}^{(m)} & \\ g_{k,i}^{(m)} = g_{k-1,i}^{(m)} - \dfrac{\langle y, \Delta g_{k-1,i}^{(m)}\rangle}{\langle y, \Delta g_{k-1,k}^{(m)}\rangle}\, g_{k-1,k}^{(m)} & \end{array}\right.$$

La différence entre cet algorithme et le H-algorithme réside dans le fait que dans le H-algorithme les $g_{k,i}^{(m)}$ sont des scalaires alors que dans le E-algorithme vectoriel ce sont des vecteurs.
Notons aussi que si l'on pose $g_i(m) = \Delta s_{m+i-1}$ la transformation $E_k$ devient la première généralisation de la transformation de Shanks.
Des résultats de convergence et d'accélération sont donnés dans [91].

Pour calculer les termes $E_k(s_n)$ lorsque $n$ est fixé, on peut utiliser le R.P.A, dont on déja parlé au chapitre II, dans un cas particulier. La formulation générale de cet algorithme est la suivante :

<u>R.P.A (Recursive Projection Algorithme)</u> [16]

Soit $E$ un espace vectoriel sur $K$ ($\mathbb{R}$ ou $\mathbb{C}$) et $E^*$ son dual.
Notons $<\ ,\ >$ la forme bilineaire de dualité entre $E$ et $E^*$.
Soit $z \in E$, $x_i \in E$ et $z_i \in E^*$.

On pose

$$N_k = \begin{vmatrix} z & x_1 & \cdots & x_k \\ \langle z_1, z\rangle & \langle z_1, x_1\rangle & \cdots & \langle z_1, x_k\rangle \\ \cdots & \cdots & \cdots & \cdots \\ \langle z_k, z\rangle & \langle z_k, x_1\rangle & \cdots & \langle z_k, x_k\rangle \end{vmatrix}$$

$$D_k = \begin{vmatrix} \langle z_1, x_1\rangle & \cdots & \langle z_1, x_k\rangle \\ \cdots & \cdots & \cdots \\ \langle z_k, x_1\rangle & \cdots & \langle z_k, x_k\rangle \end{vmatrix}$$

$$N_{k,i} = \begin{vmatrix} x_i & x_1 & \cdots & x_k \\ \langle z_1, x_i\rangle & \langle z_1, x_1\rangle & \cdots & \langle z_1, x_k\rangle \\ \cdots & \cdots & \cdots & \cdots \\ \langle z_k, x_i\rangle & \langle z_k, x_1\rangle & \cdots & \langle z_k, x_k\rangle \end{vmatrix}$$

$$E_k = \frac{N_k}{D_k} \quad \text{et} \quad g_{k,i}^{(n)} = \frac{N_{k,i}}{D_k}$$

. Et on suppose que $\forall k$, $D_k \neq 0$.

$E_k$ peut se calculer récursivement par

$$\begin{cases} E_0 = z \quad ; \quad g_{0,i} = x_i \quad i \geq 1 \\ E_k = E_{k-1} - \dfrac{\langle z_k , E_{k-1} \rangle}{\langle z_k , g_{k-1,k} \rangle} g_{k-1,k} \quad , \quad k > 0 \\ g_{k,i} = g_{k-1,i} - \dfrac{\langle z_k , g_{k-1,i} \rangle}{\langle z_k , g_{k-1,k} \rangle} g_{k-1,k} \quad , \quad i > k > 0 . \end{cases}$$

On montre facilement que cet algorithme permet le calcul de $E_k(s_n)$ pour $n$ fixé voir [16].

Remarques :

① pour $n$ fixé, on peut utiliser le R.P.A pour mettre en oeuvre toutes les méthodes polynomiales que l'on a vues. Le cas du MPE a été fait par Brezinski dans [46].

② La méthode MMPE peut être mise en oeuvre par le H-algorithme où par le $(S-\beta)$ algorithme [46] qui est défini par :

$$\begin{cases} S_0^{(n)} = s_n \quad ; \quad \beta_0^{(n)} = \Delta s_n \; ; \; n \geq 0 \\ S_k^{(n)} = \dfrac{S_{k-1}^{(n)} - a_k^{(n)} S_{k-1}^{(n+1)}}{1 - a_k^{(n)}} \\ \beta_k^{(n)} = \dfrac{\beta_{k-1}^{(n)} - a_k^{(n)} S_{k-1}^{(n+1)}}{1 - a_k^{(n)}} \end{cases} \quad \text{où } a_k^{(n)} = \dfrac{\langle y_k , \beta_{k-1}^{(n)} \rangle}{\langle y_k , \beta_{k-1}^{(n+1)} \rangle}$$

③ Pour la mise en oeuvre des méthodes MPE et RRE, on peut utiliser un algorithme qui a été défini par Ford et Sidi [30], quand $n$ est fixé (où quand $k$ est fixé)

④ Lorsque $n$ est fixé et qu'on ne désire calculer que les termes $t_{n,k}$; $t_{n,k+1}$; $t_{n,k+2}$; ..., On peut utiliser la méthode de Bordage [19].

⑤ Le R.P.A peut être utilisé pour obtenir des méthodes directes de résolutions d'un système d'équations linéaires, c'est ce qui a été fait par Jbilou [46]. L'une des méthodes avait déja été obtenue par Sloboda [80].

## ④ Application numérique :

Nous allons donner quelques exemples numériques pour illustrer le comportement des cinq méthodes suivantes :

ⓐ- MMPE, pour sa mise en oeuvre nous avons utilisé le H-algorithme ($y_i = e^i$)

ⓑ MPE : Nous avons utilisé la formulation donné dans la page 130, c'est celle qui utilise le calcul de $\Delta S^+_{m,k}$

ⓒ RRE : $t_{m,k} = s_m - \Delta S_{m,k} \Delta^2 S^+_{m,k} \Delta s_m$.

les méthodes MPE et RRE nécessitent le calcul de $A_n^+ u_n$, ceci a été fait de la manière suivante :

1) On cherche une factorisation de $A_m$ :

$$A_m = Q_m R_m \qquad ; {}^T Q_n Q_n = I.$$

2) Posons $V_m = A_m^+ u_m$, comme $A_m^+ = ({}^T A_n A_n)^{-1} {}^T A_n$

on a : $V_m = R_n^{-1} \, {}^T Q_n u_n$.

- pour déterminer $v_n$, on résoud le système

$$R_m v_m = {}^T Q_m u_m$$

$R_m$ est une matrice triangulaire superieure.

De plus la factorisation de $A_{n+1}$ est obtenue a partir de celle de $A_m$ en utilisant une technique similaire a celle donnée dans le chapitre I.

d) VEA , on a utilisé le sous programme EPSV [10] qui utilise les règles particulières de F. Cordellier [24].

e) TEA , on a utilisé le sous programme EPSTO2 [10].
de vecteur $y$ : $y_i = i$ $i=1, \ldots p$.

Pour les résultats numériques nous avons utilisé les notations suivantes :

$$f_m = -\mathrm{Log}_{10} \| s_{n+k+1} - s \|_\infty$$

et

$$f_m^{(i)} = -\mathrm{Log}_{10} \| t_{m,k} - s \| \quad \begin{cases} i=1 & \text{MMPE} \\ i=2 & \text{MPE} \\ i=3 & \text{RRE} \end{cases}$$

$$f_m^{(i)} = -\mathrm{Log}_{10} \| \varepsilon_{2k}^{(m)} - s \|_\infty \quad \begin{cases} i=4 & \text{VEA} \\ i=5 & \text{TEA} \end{cases}$$

Exemple n°1 $\quad s_{m+1} = A s_m + b \quad , \quad s_0 = 0$

A est obtenue en appliquant la méthode de Gauss-Siedel au système d'equations : $Cx = d$

$$\text{où} \quad C = \begin{bmatrix} 2 & 1 & 3 & 4 \\ 1 & -3 & 1 & 5 \\ 3 & 1 & 6 & -2 \\ 4 & 5 & -2 & -1 \end{bmatrix}.$$

le vecteur b est choisi pour que le vecteur dont toutes les composantes sont égales à 1 soit solution du système $x = Ax + b$.

la suite $\{s_n\}$ diverge car les valeurs propres de $A$ sont :

$$\begin{cases} \lambda_1 = \bar{\lambda}_2 \simeq -2.35 \pm 2.05\, i \\ \lambda_3 \simeq -0.03 \\ \lambda_4 = 0 \end{cases}$$

$k=2$

| $n$ | $f_n^{(1)}$ | $f_n^{(2)}$ | $f_n^{(3)}$ | $f_n^{(4)}$ | $f_n^{(5)}$ |
|---|---|---|---|---|---|
| 0 | -0.643 | -0.092 | -0.069 | | |
| 1 | 1.885 | 2.143 | 2.143 | 0.057 | 0.481 |
| 2 | 3.526 | 3.784 | 3.785 | 2.196 | 2.955 |
| 3 | 5.167 | 5.379 | 5.510 | 3.837 | 4.596 |
| 4 | 6.808 | 6.347 | 5.513 | 5.478 | 6.238 |

Exemple n°2. $\qquad s_{n+1} = A s_n + b \quad , \quad s_0 = 0$

$$A = 0.06 \begin{bmatrix} 5 & 2 & 1 & 1 & & & & & & & \\ 2 & 6 & 3 & 1 & 1 & & & & & & \\ 1 & 3 & 6 & 3 & 1 & 1 & & & & & \\ 1 & 1 & 3 & 6 & 3 & 1 & 1 & & & & \\ & 1 & 1 & 3 & 6 & 3 & 1 & 1 & & & \\ & & 1 & 1 & 3 & 6 & 3 & 1 & 1 & & \\ & & & 1 & 1 & 3 & 6 & 3 & 1 & 1 & \\ & & & & 1 & 1 & 3 & 6 & 3 & 1 & 1 \\ & & & & & 1 & 1 & 3 & 6 & 3 & 1 \\ & & & & & & 1 & 1 & 3 & 6 & 2 \\ & & & & & & & 1 & 1 & 2 & 5 \end{bmatrix}$$

$$b = s - As \quad \text{où} \quad s = (1, 1, \dots, 1)^T.$$

Les valeurs propres de $A$ sont toutes réelles et sont dans l'intervalle $]0,1[$, la plus grande est $\lambda_1 \simeq 0.8965$

$$\lambda_2 \simeq 0.7318$$
$$\lambda_3 \simeq 0.5297$$

$k = 1$ :

| $n$ | $f_n$ | $f_n^{(1)}$ | $f_n^{(2)}$ | $f_n^{(3)}$ | $f_n^{(4)}$ | $f_n^{(5)}$ |
|---|---|---|---|---|---|---|
| 0 | 0.039 | 0.052 | 0.035 | 0.029 | 0.068 | 0.090 |
| 1 | 0.068 | 0.097 | 0.092 | 0.064 | 0.176 | 0.193 |
| 2 | 0.104 | 0.162 | 0.228 | 0.137 | 0.321 | 0.351 |
| 3 | 0.145 | 0.250 | 0.460 | 0.273 | 0.349 | 0.550 |
| 4 | 0.188 | 0.363 | 0.760 | 0.500 | 0.464 | 0.775 |
| 5 | 0.233 | 0.503 | 1.084 | 0.817 | 0.668 | 1.019 |
| 6 | 0.279 | 0.671 | 1.405 | 1.180 | 0.926 | 1.274 |
| 7 | 0.326 | 0.866 | 1.716 | 1.547 | 1.204 | 1.537 |
| 8 | 0.373 | 1.086 | 2.015 | 1.899 | 1.487 | 1.805 |
| 9 | 0.420 | 1.324 | 2.306 | 2.230 | 1.769 | 2.076 |
| 10 | 0.467 | 1.576 | 2.591 | 2.544 | 2.049 | 2.349 |
| 11 | 0.515 | 1.838 | 2.873 | 2.844 | 2.328 | 2.623 |
| 12 | 0.562 | 2.104 | 3.152 | 3.135 | 2.605 | 2.898 |
| 13 | 0.609 | 2.375 | 3.430 | 3.420 | 2.882 | 3.173 |
| 14 | 0.657 | 2.647 | 3.708 | 3.701 | 3.159 | 3.449 |
| 15 | 0.704 | 2.921 | 3.984 | 3.981 | 3.435 | 3.724 |
| 16 | 0.752 | 3.196 | 4.261 | 4.258 | 3.711 | 4.000 |
| 17 | 0.799 | 3.472 | 4.537 | 4.536 | 3.988 | 4.276 |
| 18 | 0.846 | 3.747 | 4.813 | 4.812 | 4.264 | 4.552 |
| 19 | 0.894 | 4.023 | 5.089 | 5.089 | 4.540 | 4.828 |
| 20 | 0.941 | 4.299 | 5.365 | 5.365 | 4.816 | 5.104 |
| 21 | 0.989 | 4.575 | 5.641 | 5.641 | 5.092 | 5.380 |
| 22 | 1.036 | 4.850 | 5.917 | 5.917 | 5.368 | 5.656 |
| 23 | 1.084 | 5.126 | 6.193 | 6.193 | 5.644 | 5.932 |
| 24 | 1.131 | 5.402 | 6.469 | 6.469 | 5.919 | 6.208 |
| 25 | 1.179 | 5.678 | 6.745 | 6.745 | 6.195 | 6.484 |
| 26 | 1.226 | 5.954 | 7.021 | 7.021 | 6.471 | 6.760 |
| 27 | 1.273 | 6.230 | 7.297 | 7.297 | 6.747 | 7.036 |
| 28 | 1.321 | 6.506 | 7.573 | 7.573 | 7.023 | 7.312 |
| 29 | 1.368 | 6.782 | 7.849 | 7.849 | 7.299 | 7.588 |

$k = 2$

| $n$ | $f_n$ | $f_n^{(1)}$ | $f_n^{(2)}$ | $f_n^{(3)}$ | $f_n^{(4)}$ | $f_n^{(5)}$ |
|---|---|---|---|---|---|---|
| | 0.068 | 0.146 | 0.154 | 0.093 | | |
| 1 | 0.104 | 0.249 | 0.457 | 0.243 | 0.335 | 0.595 |
| 2 | 0.145 | 0.384 | 1.065 | 0.685 | 0.474 | 0.924 |
| 3 | 0.188 | 0.577 | 1.745 | 1.478 | 0.681 | 1.324 |
| 4 | 0.233 | 0.861 | 2.291 | 2.289 | 1.103 | 1.794 |
| 5 | 0.279 | 1.252 | 2.864 | 2.867 | 1.666 | 2.314 |
| 6 | 0.326 | 1.736 | 3.446 | 3.449 | 2.267 | 2.866 |
| 7 | 0.373 | 2.276 | 4.031 | 4.033 | 2.869 | 3.434 |
| 8 | 0.420 | 2.845 | 4.618 | 4.618 | 3.467 | 4.011 |
| 9 | 0.467 | 3.426 | 5.204 | 5.204 | 4.060 | 4.591 |
| 10 | 0.515 | 4.014 | 5.790 | 5.789 | 4.651 | 5.172 |
| 11 | 0.562 | 4.603 | 6.383 | 6.374 | 5.239 | 5.755 |
| 12 | 0.609 | 5.192 | 6.952 | 6.957 | 5.826 | 6.337 |
| 13 | 0.657 | 5.781 | 7.241 | 7.519 | 6.412 | 6.919 |

Exemple n°3 $\qquad s_{n+1} = A s_n + b \qquad s_0 = 0$

$$A = 0.04\begin{bmatrix} 12 & 11 & & & & & \\ 11 & 11 & 10 & & & & \\ 10 & 10 & 10 & 9 & & & \\ \vdots & \vdots & \vdots & \vdots & & & \\ 2 & 2 & 2 & 2 & \cdots & 2 & 1 \\ 1 & 1 & 1 & 1 & \cdots & 1 & 1 \end{bmatrix}$$

$b = s - As \qquad$ et $\qquad s = (1, 1, \ldots, 1)$

Les valeurs propres de A sont toutes réelles.

$\lambda_1 \simeq 1.2892 \; ; \; \lambda_2 \simeq 0.808 \; ; \; \lambda_3 = 0.4924$.

$k=3$

| $n$ | $f_n^{(1)}$ | $f_n^{(2)}$ | $f_n^{(3)}$ | $f_n^{(4)}$ | $f_n^{(5)}$ |
|---|---|---|---|---|---|
| | -0.076 | -0.480 | -0.471 | | |
| 1 | 0.121 | -0.070 | -0.384 | | |
| 2 | 0.393 | 0.894 | 0.466 | -0.378 | -0.481 |
| 3 | 0.739 | 1.570 | 1.785 | -0.267 | -0.662 |
| 4 | 1.176 | 2.242 | 2.265 | 0.221 | 0.552 |
| 5 | 1.740 | 2.828 | 2.824 | 1.034 | 1.331 |
| 6 | 2.515 | 3.398 | 3.362 | 1.756 | 2.024 |
| 7 | 4.463 | 3.963 | 3.902 | 2.414 | 2.665 |
| 8 | 3.654 | 4.796 | 4.516 | 3.033 | 3.267 |
| 9 | 4.005 | 4.519 | 4.823 | 3.625 | 3.846 |
| 10 | 4.479 | 5.024 | 5.595 | 4.199 | 4.411 |

Exemple n°4 $\qquad s_{n+1} = G(s_n) = b + A s_n + Q(s_n)$

G est la fonction définie dans l'exemple n°4 du chapitre 1.

$s_0 = (1.5; 1.6; 1.7; 1.8)^T \quad , \quad s = (1, 1, 1, 1)^T$

Les valeurs propres de $G'(s^*)$ sont:

0.9 ; 0.8 ; 0.7 ; 0.6

k=2.

| $n$ | $f_n$ | $f_n^{(1)}$ | $f_n^{(2)}$ | $f_n^{(3)}$ | $f_n^{(4)}$ | $f_n^{(5)}$ |
|---|---|---|---|---|---|---|
| | 0.880 | 1.151 | 1.134 | 1.133 | | |
| 1 | 1.024 | 1.928 | 1.840 | 1.837 | 1.406 | 1.544 |
| 2 | 1.156 | 2.192 | 2.108 | 2.106 | 1.800 | 1.868 |
| 3 | 1.278 | 2.507 | 2.415 | 2.412 | 2.107 | 2.133 |
| 4 | 1.395 | 2.750 | 2.769 | 2.766 | 2.387 | 2.375 |
| 5 | 1.506 | 2.841 | 2.890 | 2.891 | 2.651 | 2.601 |
| 6 | 1.615 | 2.972 | 3.002 | 3.003 | 2.905 | 2.816 |
| 7 | 1.720 | 3.123 | 3.139 | 3.140 | 3.154 | 3.023 |
| 8 | 1.823 | 3.293 | 3.300 | 3.300 | 3.399 | 3.223 |
| 9 | 1.925 | 3.474 | 3.475 | 3.475 | 3.641 | 3.418 |
| 10 | 2.025 | 3.664 | 3.659 | 3.659 | 3.880 | 3.608 |
| 11 | 2.123 | 3.857 | 3.849 | 3.849 | 4.117 | 3.792 |
| 12 | 2.221 | 4.054 | 4.042 | 4.042 | 4.350 | 3.973 |
| 13 | 2.318 | 4.251 | 4.235 | 4.235 | 4.580 | 4.148 |
| 14 | 2.414 | 4.448 | 4.428 | 4.428 | 4.806 | 4.320 |
| 15 | 2.509 | 4.643 | 4.620 | 4.620 | 5.029 | 4.487 |
| 16 | 2.604 | 4.804 | 4.809 | 4.809 | 5.248 | 4.651 |
| 17 | 2.698 | 4.965 | 4.993 | 4.994 | 5.460 | 4.811 |
| 18 | 2.792 | 5.124 | 5.156 | 5.157 | 5.669 | 4.968 |
| 19 | 2.886 | 5.281 | 5.316 | 5.318 | 5.877 | 5.123 |
| 20 | 2.979 | 5.435 | 5.474 | 5.476 | 6.085 | 5.274 |
| 21 | 3.072 | 5.587 | 5.628 | 5.631 | 6.293 | 5.423 |
| 22 | 3.165 | 5.735 | 5.777 | 5.781 | 6.501 | 5.570 |
| 23 | 3.258 | 5.880 | 5.923 | 5.928 | 6.709 | 5.715 |
| 24 | 3.351 | 6.022 | 6.063 | 6.069 | 6.916 | 5.858 |
| 25 | 3.443 | 6.161 | 6.200 | 6.207 | 7.122 | 5.999 |
| 26 | 3.535 | 6.297 | 6.331 | 6.339 | 7.292 | 6.138 |
| 27 | 3.628 | 6.431 | 6.458 | 6.468 | 7.430 | 6.277 |
| 28 | 3.720 | 6.562 | 6.582 | 6.592 | 7.574 | 6.414 |
| 29 | 3.812 | 6.691 | 6.701 | 6.713 | 7.721 | 6.550 |

## Exemple n°5

$$s_{n+1} = G(s_n) = b + As_n + Q(s_n)$$

G est la fonction définie dans l'exemple n°1 chapitre II.

$s_0 = (1.5\,;\,1.6\,;\,1.7\,;\,1.8)^T$ , $s = (3,3,3,3)$

les valeurs propres de $G'(s^*)$ sont:

0.5 ; −0.4 ; −0.3 ; −0.2.

$k=3$

| $n$ | $f_n$ | $f_n^{(1)}$ | $f_n^{(2)}$ | $f_n^{(3)}$ | $f_n^{(4)}$ | $f_n^{(5)}$ |
|---|---|---|---|---|---|---|
| | 0.126 | -0.172 | -0.321 | -0.313 | | |
| 1 | 0.286 | 0.555 | 0.570 | 0.570 | | |
| 2 | 0.487 | 0.743 | 0.748 | 0.748 | 0.090 | 0.156 |
| 3 | 0.722 | 0.881 | 0.203 | 0.734 | -0.127 | 0.041 |
| 4 | 0.984 | 1.565 | 1.583 | 1.583 | 0.261 | 0.303 |
| 5 | 1.263 | 1.882 | 1.883 | 1.883 | 1.550 | 1.563 |
| 6 | 1.552 | 2.342 | 2.348 | 2.348 | 2.610 | 2.704 |
| 7 | 1.847 | 2.886 | 2.932 | 2.939 | 3.634 | 3.719 |
| 8 | 2.145 | 5.436 | 3.615 | 3.402 | 4.519 | 4.568 |
| 9 | 2.445 | 3.742 | 3.793 | 3.824 | 5.958 | 6.149 |
| 10 | 2.745 | 4.396 | 4.415 | 4.421 | 6.690 | 6.762 |
| 11 | 3.046 | 5.007 | 5.033 | 5.027 | 7.964 | 8.130 |
| 12 | 3.347 | 5.609 | 5.614 | 5.604 | 8.592 | 8.654 |
| 13 | 3.647 | 6.210 | 6.313 | 6.215 | 9.198 | 9.587 |
| 14 | 3.948 | 6.809 | 6.824 | 7.799 | 9.293 | 9.564 |
| 15 | 4.249 | 7.409 | 7.396 | 7.925 | 9.537 | 8.871 |
| 16 | 4.550 | 8.009 | 8.025 | 7.123 | 10.759 | 10.806 |
| 17 | 4.852 | 8.609 | 8.593 | 8.628 | 11.519 | 11.474 |
| 18 | 5.153 | 9.209 | 9.228 | 9.170 | 12.583 | 12.773 |
| 19 | 5.454 | 9.810 | 9.791 | 9.880 | 13.072 | 13.111 |
| 20 | 5.755 | 10.411 | 10.440 | 10.303 | 14.017 | 14.190 |
| 21 | 6.056 | 11.015 | 10.975 | 11.325 | 14.350 | 14.437 |
| 22 | 6.357 | 11.630 | 11.724 | 11.450 | 14.968 | 15.081 |
| 23 | 6.658 | 12.209 | 12.171 | 12.514 | 15.488 | 15.592 |
| 24 | 6.959 | 12.721 | 12.928 | 12.645 | 16.003 | 16.103 |

## B- Etude des problèmes non linéaires.

Nous allons commencer par donner un formalisme général pour les méthodes polynomiales.

Dans la définition des transformations composites étudiées au chapitre II, nous avions supposé que la suite $\{\Delta T_2^{(n)} - \Delta T_1^{(n)}\}$ était une suite de matrices inversibles, or ceci n'est pas toujours vérifié c'est pourquoi nous allons proposer des extensions de ces transformations composites.

### 1) Définition de la transformation $t_{k,\cdot}$.

Soit $\{s_n\}$ une suite de vecteurs complexes et soient $t_1$ et $t_2$ deux transformations de suites définies par :

$$t_i : \{s_n\} \longrightarrow \{t_i^{(n)}\} \qquad i = 1, 2$$

. Soit $k$ un entier fixé $(k \leq p)$. La transformtion $t_{k,\cdot}$ est définie par :

$$t_{k,\cdot} : \{s_n\} \longrightarrow \{t_{k,n}\}$$

où

$$t_{k,n} = t_1^{(n)} - \Delta T_{1,k}^{(n)} \left( \overset{*}{Y}_k^{(n)} (\Delta T_{2,k}^{(n)} - \Delta T_{1,k}^{(n)}) \right)^{-1} \overset{*}{Y}_k^{(n)} (t_2^{(n)} - t_1^{(n)}) .$$

Nous avons utilisé la notation suivante :

$$A_k^{(n)} = (a_1^{(n)}, \ldots, a_k^{(n)}) \quad \text{où} \quad a_i^{(n)} \in \mathbb{C}^p \text{ pour } i = 1, \ldots, k$$

$$\text{et} \quad A_k^{(n)} \in L(\mathbb{C}^p, \mathbb{C}^k) ,$$

et la suite de matrices auxiliaires :

$$Y_k^{(n)} = (y_1^{(n)}, \ldots, y_k^{(n)}) .$$

Remarques:

1- Si $k=p$ et $Y_k^{(n)}$ est inversible alors on retrouve la première transformation composite étudiée au chapitre II.

2- Cette transformation généralise plusieurs méthodes d'accélération de la convergence :

$t_1^{(n)} = s_n$ et $t_2^{(n)} = s_{n+1}$

a) $y_i^{(n)} = \Delta s_{n+i-1}$ , On obtient la méthode MPE

b) $y_i^{(n)} = \Delta^2 s_{n+i-1}$ , On obtient la méthode RRE

c) $y_i^{(n)} = y_i$ , On obtient la méthode MMPE.

Lorsque les suites $\{t_1^{(n)}\}$ et $\{t_2^{(n)}\}$ sont générées linéairement, on peut avoir accélération de la convergence sous certaines hypothèses (les résultats obtenus pour les méthodes polynomiales se généralisent facilement).

Quand les matrices $\Delta T_2^{(n)} - \Delta T_1^{(n)}$ sont de rang maximal nous avons vu que pour des suites générées non linéairement, on peut avoir accélération de la convergence.

Nous allons nous interesser maintenant au cas ou les matrices $\Delta T_2^{(n)} - \Delta T_1^{(n)}$ ne sont pas inversibles et dans ce cas $k$ ne doit plus être fixé, mais doit dépendre de $n$ comme nous allons le voir.

② <u>Choix de $k$ pour avoir accélération de la convergence.</u>

Posons :

$$\boxed{k_n = \max \{ k \;/\; \mathrm{rang}\,(T_{2,k}^{(n)} - T_{1,k}^{(n)}) = k \}}$$

Nous avons les propriétés suivantes:

ⓐ- $k_n \leq p$

ⓑ- $k_{n+1} \geq k_n - 1$

par définition de $k_{n+1}$ :

$$k_{n+1} = \max \{ \ell \,/\, \mathrm{rang}\,(T_{2,\ell}^{(n+1)} - T_{1,\ell}^{(n+1)}) = \ell \}$$

et comme $\mathrm{rang}\left(T_{2,k_n-1}^{(n+1)} - T_{1,k_n-1}^{(n+1)}\right) = k_n - 1$

alors $k_{n+1} \geq k_n - 1$ .

Notons $u$ la transformation suivante :

$$u : \{s_n\} \longrightarrow \{u_n\}$$

où

$$u_n = t_1^{(n)} - \Delta T_{1,k_n}^{(n)} \left( Y_{k_n}^{*(n)} \left( \Delta T_{2,k_n}^{(n)} - \Delta T_{1,k_n}^{(n,k)} \right) \right)^{-1} Y_{k_n}^{*(n)} (t_2^{(n)} - t_1^{(n)})$$

Nous supposerons dans ce qui suit que:

$\forall n \quad Y_{k_n}^{*} \left( \Delta T_{2,k_n}^{(n)} - \Delta T_{1,k_n}^{(n)} \right)$ est inversible.

Remarque :

si on prend $Y_{k_n} = \Delta T_{2,k_n}^{(n)} - \Delta T_{1,k_n}^{(n)}$

On obtient :

$$u_n = t_1^{(n)} - \Delta T_{1,k_n}^{(n)} \left( \Delta T_{2,k_n}^{(n)} - \Delta T_{1,k_n}^{(n)} \right)^{+} (t_2^{(n)} - t_1^{(n)}) .$$

Donnons une extension de la notion de matrice uniformement inversible dans le cas de matrices rectangulaires.

**Définition :** Soit $c \in \mathbb{R}_+^*$.

Notons $G_k(c)$ l'ensemble des matrices défini par :

$$G_k(c) = \{ A = (a^1, a^2, \ldots, a^k) \in L(\mathbb{C}^p, \mathbb{C}^k) \,/\, k \leq p \text{ et } a^i \neq 0 \text{ pour } i = 1, \ldots k$$
$$\text{et } \det(\tilde{A}^* \tilde{A}) \geq c^2 \quad \text{avec } \tilde{A} = \left(\frac{a^1}{\|a^1\|_2}, \ldots, \frac{a^k}{\|a^k\|_2}\right)$$

des vecteurs $a_n^1, a_n^2, \ldots, a_n^k$ seront dits uniformement linéairement indépendants si et seulement si : $\exists c > 0$, $\exists N$ tel que $(a_n^1, a_n^2, \ldots, a_n^k) \in G_k(c)$ $\forall n > N$

$G_k(c)$ généralise bien $K(c)$, puisque $G_p(c) = K(c)$ (voir chapitre I).

si $A \in G_k(c)$ alors $\mathrm{rang}(A) = k$. Une définition similaire a été donnée dans [22].

Nous allons montrer que le choix que l'on a fait pour $k_n$, permet de donner le théorème d'accélération de la convergence suivant :

**théorème 15 :** si $t_2^{(n)} - s = (A + A_n)(t_1^{(n)} - s)$ où $\lim_n A_n = 0$ et $(A, A_n) \in L(\mathbb{C}^p)^2$

et si $t_1^{(n+1)} - s = (B + B_n)(t_1^{(n)} - s)$ où $\lim_n B_n = 0$ et $(B, B_n) \in L(\mathbb{C}^p)^2$

si de plus $\rho(A) < 1$ et $\rho(B) < 1$

et si $\exists d > 0$ $\exists N$ $\forall n > N$ $T_{2,k_n}^{(n)} - T_{1,k_n}^{(n)} \in G_{k_n}(d)$

alors

$$\|u_n - s\| = o(\|t_1^{(n)} - s\|)$$

si de plus $A$ est inversible alors $\|u_n - s\| = o(\|t_2^{(n)} - s\|)$

Pour démontrer ce théorème nous avons besoin des lemmes suivants :

**Lemme 2**

> Soit $A$ une matrice de l'ensemble $L(\mathbb{C}^p, \mathbb{C}^k)$ où $1 \leq k \leq p$
> et $rang(A) = k$.
> Posons $A^+ = (A^* A)^{-1} A^*$, alors :
> $$\|A^+\|_2 \leq \frac{\|A\|_F^{k-1}}{\sqrt{\det(A^* A)}}$$

preuve :

$$\|A^+\|_2^2 = \|A^+ A^{+*}\|_2 \qquad (\text{D'après } [82])$$
$$= \|(A^* A)^{-1} A^* A \{(A^* A)^{-1}\}^* \|_2$$
$$= \|(A^* A)^{-1}\|_2$$

$A^* A$ est une matrice carrée elle appartient à l'ensemble $L(\mathbb{C}^k)$.
appliquons le lemme 1 du chapitre I.

$$\|(A^* A)^{-1}\|_2 \leq \frac{\|A^* A\|_F^{k-1}}{|\det(A^* A)|}$$

et donc
$$\|(A^* A)^{-1}\|_2 \leq \frac{\|A\|_F^{2k-2}}{\det(A^* A)}$$
on obtient donc le résultat cherché.

■

Utilisons ce lemme pour démontrer le résultat suivant :

> **Lemme 3** Soient $\{x_n\}$ et $\{y_n\}$ deux suites de vecteurs complexes telsque :
> $x_n = (C + C_n) y_n$ où $C$ est une matrice inversible,
> $\lim_n C_n = 0$ et $(C, C_n) \in L(\mathbb{C}^p)^2$. Soit $\{f_n\}$ une suite d'entiers telsque : $f_n \in \{1, \dots, p\}$.
> Si $\exists d > 0$, $\exists N$, $\forall n > N$ $\quad Y_{n,f_n} \in G_{f_n}(d)$
> alors :
> $$X_{n,f_n} = (C + B_n) Y_{n,f_n} \quad \text{où } \lim B_n = 0$$
> et $B_n \in L(\mathbb{C}^p)$

preuve :

Par définition $z_{n+i} = (C + C_{n+i})\, y_{n+i}$ $\quad i = 0, \dots, f_n - 1$

Ecrivons ces relations sous une forme matricielle

$$X_{n,f_n} = \left( (C + C_n) y_n, \ \dots, (C + C_{n+f_n-1})\, y_{n+f_n-1} \right)$$

où encore

$$X_{n,f_n} = C\, Y_{n,f_n} + \left( C_n y_n, \dots, C_{n+f_n-1}\, y_{n+f_n-1} \right)$$

par hypothèse le rang $(Y_{n,f_n}) = f_n$

et donc $\quad Y^+_{n,f_n} = \left( \overset{*}{Y}_{n,f_n} Y_{n,f_n} \right)^{-1} \overset{*}{Y}_{n,f_n}$

or

$$\left( C_n y_n, \dots, C_{n+f_n-1}\, y_{n+f_n-1} \right) = \left( C_n y_n, \dots, C_{n+f_n-1}\, y_{n+f_n-1} \right) Y^+_{n,f_n} Y_{n,f_n}$$
$$= B_n Y_{n,f_n}$$

$$B_n = \left( C_n \frac{y_n}{\|y_n\|_2}, \dots, C_{n+f_n-1} \frac{y_{n+f_n-1}}{\|y_{n+f_n-1}\|_2} \right) \times \left( \frac{y_n}{\|y_n\|_2}, \dots, \frac{y_{n+f_n-1}}{\|y_{n+f_n-1}\|_2} \right)^+$$

$$\|B_n\|_2 \leq \left( \sum_{i=0}^{f_n-1} \left\| C_{n+i} \frac{y_{n+i}}{\|y_{n+i}\|_2} \right\|_2^2 \right)^{1/2} \left\| \left( \frac{y_n}{\|y_n\|_2}, \dots, \frac{y_{n+f_n-1}}{\|y_{n+f_n-1}\|_2} \right)^+ \right\|$$

d'après le lemme 2, en en posant $\tilde{Y}_{n,f_n} = \left( \frac{y_n}{\|y_n\|_2}, \dots, \frac{y_{n+f_n-1}}{\|y_{n+f_n-1}\|_2} \right)$,

$$\|B_n\|_2 \leq \left( \sum_{i=0}^{f_n-1} \|C_{n+i}\|_2^2 \right)^{1/2} \frac{\|\tilde{Y}_{n,f_n}\|_F^{f_n-1}}{\det\left( \overset{*}{\tilde{Y}}_{n,f_n} Y_{n,f_n} \right)^{1/2}}$$

et par conséquent

$$\|B_n\|_2 \leq \left( \sum_{i=0}^{p-1} \|C_{n+i}\|_2^2 \right)^{1/2} \frac{p^{(p-1)/2}}{d} \qquad \text{si } f_n \neq 1.$$

et

$$\|B_n\|_2 \leq \frac{\|C_n\|}{d} \qquad \text{si } f_n = 1.$$

et donc $\quad \lim_n B_n = 0$ ■

Lemme 4 | Soit $A$ une matrice appartenant à l'ensemble $L(\mathbb{C}^p, \mathbb{C}^k)$. Notons $a^i$ sa $i$-ème colonne. Si $z \in V[a^1, \dots, a^k]$ alors $A(B^*A)^{-1}B^* z = z$ où $B \in L(\mathbb{C}^p, \mathbb{C}^k)$ et telle que $B^*A$ est inversible

preuve:

$$z \in V(a^1, \dots, a^k) \quad \text{i.e} \quad z = \sum_{i=1}^{k} d_i a^i$$

$$A(B^*A)^{-1}B^* z = A(B^*A)^{-1}B^*\left(\sum_{i=1}^{k} d_i a^i\right)$$

$$= \sum_{i=1}^{k} d_i \left[A(B^*A)^{-1}B^* a^i\right]$$

or $A(B^*A)^{-1}B^*A = A$ et donc $A(B^*A)^{-1}B^* a^i = a^i$

d'où $A(B^*A)^{-1}B^* z = z$ ■

Nous sommes maintenant en mesure de démontrer le théorème 15.

**démonstration du théorème 15:**

D'après le théorème 11 du chapitre I

$$\Delta t_2^{(n)} - \Delta t_1^{(n)} = \left[(A-I)(B-I)(A-I)^{-1} + C_n\right](t_2^{(n)} - t_1^{(n)}) \quad \text{où } \lim_n C_n = 0$$

en utilisant le lemme 2 nous obtenons:

$$\Delta T_{2,k_n}^{(n)} - \Delta T_{1,k_n}^{(n)} = \left[(A-I)(B-I)(A-I)^{-1} + D_n\right](T_{2,k_n}^{(n)} - T_{1,k_n}^{(n)}) \qquad (1)$$

et $\lim_n D_n = 0$.

De plus $\Delta t_1^{(n)} = \left[(B-I)(A-I)^{-1} + E_n\right](t_2^{(n)} - t_1^{(n)})$

avec $\lim_n E_n = 0$ et d'après le lemme 3

$$\Delta T_{1,k_n}^{(n)} = \left[(B-I)(A-I)^{-1} + F_n\right](T_{2,k_n}^{(n)} - T_{1,k_n}^{(n)}) \qquad (2)$$

et $\lim_n F_n = 0$.

par définition de $u_n$ nous avons :

$$u_n = t_1^{(n)} - \Delta T_{1,k_n}^{(n)} \left( \overset{*}{Y}_k^{(n)} \left(\Delta T_{2,k_n}^{(n)} - \Delta T_{1,k_n}^{(n)}\right)\right)^{-1} \overset{*}{Y}_{k_n} \left(t_2^{(n)} - t_1^{(n)}\right)$$

(1) et (2) impliquent :

$$\exists N_1, \forall n > N_1, \qquad \Delta T_{1,k_n}^{(n)} = \left[(A-I)^{-1} + H_n\right] \left(\Delta T_{2,k_n}^{(n)} - \Delta T_{1,k_n}^{(n)}\right)$$

où

$$\lim_n H_n = 0 .$$

et par conséquent :

$$u_n = t_1^{(n)} - \left[(A-I)^{-1} + H_n\right] \left(\Delta T_{2,k_n}^{(n)} - \Delta T_{1,k_n}^{(n)}\right) \left[ Y_k^{(n)} \left(\Delta T_{2,k_n}^{(n)} - \Delta T_{1,k_n}^{(n)}\right)\right]^{-1} Y_k^{(n)} \left(t_2^{(n)} - t_1^{(n)}\right).$$

Montrons que :

$$\exists N_2, \forall n > N_2 \qquad t_2^{(n)} - t_1^{(n)} \in V\left(\Delta t_2^{(n)} - \Delta t_1^{(n)}, \Delta t_2^{(n+1)} - \Delta t_1^{(n+1)}, \dots, \Delta t_2^{(n+k_n-1)} - \Delta t_1^{(n+k_n-1)}\right)$$

sinon

$$\operatorname{rang}\left(t_2^{(n)} - t_1^{(n)}, t_2^{(n+1)} - t_1^{(n+1)}, \dots, t_2^{(n+k_n)} - t_1^{(n+k_n)}\right) = k_n + 1$$

impossible puisque par définition de $k_n$ nous avons :

$$k_n = \max_k \left\{ \operatorname{rang}\left(T_{2,k}^{(n)} - T_{1,k}^{(n)}\right) = k \right\} .$$

Et donc d'après le lemme 4,

$$u_n = t_1^{(n)} - \left[(A-I)^{-1} + H_n\right]^{-1} \left(t_2^{(n)} - t_1^{(n)}\right)$$

et comme

$$t_2^{(n)} - t_1^{(n)} = \left[(A-I) + A_n\right] \left(t_1^{(n)} - s\right)$$

alors

$$u_n - s = F_n \left(t_1^{(n)} - s\right)$$

d'où $\qquad \|u_n - s\| = o\left(\|t_1^{(n)} - s\|\right) .$

Si de plus A est inversible alors $\left\{ t_1^{(n)} - s = (A+A_n)^{-1}(t_2^{(n)} - s) \right. \quad \forall n > N_3$

$u_n - s = F_n (A+A_n)^{-1} (t_2^{(n)} - s)$

et donc $\|u_n - s\| = o(\|t_2^{(n)} - s\|)$.

■

③ Cas où $t_1^{(n)} = s_n$ et $t_2^{(n)} = s_{n+1}$.

Avec ces choix, les hypothèses du théorème 15 deviennent:

$$s_{n+1} - s = (A + A_n)(s_n - s) \quad \text{avec } \rho(A) < 1 \text{ et } \lim A_n = 0,$$

$$k_n = \max_k \{ \text{rang}(\Delta S_{n,k}) = k \}$$

et $\exists d > 0$ et $\exists N$, $\forall n > N$ $\quad \Delta S_{n,k_n} \in G_{k_n}(d)$.

Et comme conclusion nous avons:

$$\|u_n - s\| = o(\|s_n - s\|) \quad \text{où}$$

avec $$u_n = s_n - \Delta S_{n,k_n} \left( \overset{*}{Y}{}_{k_n}^{(n)} \Delta^2 S_{n,k_n} \right)^{-1} \overset{*}{Y}{}_{k_n}^{(n)} \Delta s_n .$$

On voit donc qu'avec le choix de $k_n$ que l'on a donné les trois transformations polynomiales (MMPE, MPE et RRE) accélèrent la convergence de la suite $\{s_n\}$.

Posons $p_n$ le degré du polynome minimal de la matrice A pour le vecteur $s_n - s$.

Nous allons maintenant examiner le lien existant entre $k_n$ et $p_n$.

théorème 16

> Soit $\{s_n\}$ une suite de vecteurs complexes tels que
> $$s_{n+1} - s = (A + A_n)(s_n - s) \quad \text{ou} \quad A \in L(\mathbb{C}^p),\ A_n \in L(\mathbb{C}^p)$$
> avec $\lim A_n = 0$ et $\rho(A) < 1$.
> Si $\exists\ d > 0 \quad \exists N \quad \forall n > N \quad \Delta S_{n,p_n} \in G_{p_n}(d)$,
> alors $\exists N'$, $\quad k_n = p_n \quad \forall n > N'$

preuve: par hypothèse nous avons :
$$\text{rang}(\Delta S_{n,p_n}) = p_n, \quad \forall n > N$$

il nous reste à montrer que le rang de la matrice $\Delta S_{n,p_n+1}$ est aussi égal à $p_n$.

D'après le théorème 5 du chapitre I, nous avons :
$$\Delta^2 s_n = (A - I)\Delta s_n + C_n \Delta s_n \quad \text{avec} \quad \lim_n C_n = 0.$$

Par application du lemme 3, nous obtenons :
$$\Delta^2 S_{n,p_n} = ((A - I) + B_n)\,\Delta S_{n,p_n}.$$
$$\text{avec} \quad \lim_n B_n = 0$$

Posons $\quad X_n = ((A - I) + B_n)$

donc $\quad \Delta^2 s_{n+i-1} = X_n\, \Delta s_{n+i-1} \quad$ pour $i = 1, \ldots, p_n$

et
$$\begin{aligned} \Delta s_{n+i} &= (I + X_n)\,\Delta s_{n+i-1} && i = 1, \ldots p_n. \\ &= Y_n\, \Delta s_{n+i-1} && i = 1, \ldots p_n \\ &= Y_n^i\, \Delta s_n. \end{aligned}$$

Montrons que $p_n$ est le degré du polynome minimal de $Y_n$ pour $\Delta s_n$ à partir d'un certain rang.

Soit $d_i^{(n)}$ les coefficients du polynome minimal $P_n$ de $A$ pour $e_n$:

$$\sum_{i=0}^{p_n} \alpha_i^{(n)} A^i e_n = 0 \qquad \text{avec} \qquad \alpha_{p_n}^{(n)} = 1 .$$

Comme

$$\Delta e_n = \Delta s_n = (A - I + A_n)\, e_n = (A-I)e_n + A_n e_n$$

et

$$\Delta s_{n+1} = (A - I + A_{n+1})\, e_{n+1} = (A - I + A_{n+1})(A + A_n)\, e_n$$
$$= (A-I) A e_n + A_{n,1} e_n$$

donc par récurrence

$$\Delta s_{n+i} = (A-I) A^i e_n + A_{n,i}\, e_n \qquad i = 0, \ldots, p_n$$

avec $\quad A_{n,0} = A_n \quad$ et $\quad \lim_n A_{n,i} = 0$ .

Multiplions $\Delta s_{n+i}$ par $\alpha_i^{(n)}$ et sommons, nous obtenons

$$\sum_{i=0}^{p_n} \alpha_i^{(n)} \Delta s_{n+i} = (A-I) \sum_{i=0}^{p_n} \alpha_i^{(n)} A^i e_n + \left(\sum_{i=0}^{p_n} \alpha_i^{(n)} A_{n,i}\right) e_n$$

d'où

$$\sum_{i=0}^{p_n} \alpha_i^{(n)} Y_n^i\, \Delta s_n = \left(\sum_{i=0}^{p_n} \alpha_i^{(n)} A_{n,i}\right) e_n .$$

des $\alpha_i^{(n)}$ sont bornés, donc $\displaystyle\sum_{i=0}^{p_n} \alpha_i^{(n)} A_{n,i} = D_n$ où $\lim_n D_n = 0$.

$$\sum_{i=0}^{p_n} \alpha_i^{(n)} Y_n^i\, \Delta s_n = D_n e_n$$

Comme les vecteurs $\Delta s_n, \ldots, \Delta s_{n+p_n-1}$ sont linéairement indépendants le dégré du polynome minimal $Q_n$ de $Y_n$ pour $\Delta s_n$ est donc superieur où egal à $p_n$. Soit $p'_n$ ce degré, supposons que $p'_n > p_n$.

Posons $$Q_n(t) = \sum_{i=0}^{p'_n} \beta_i^{(n)} t^i \qquad \beta_{p'_n}^{(n)} = 1 .$$

$F_n(t)$ ne peut pas tendre vers $Q_n(t)$ quand n tend vers l'infini puisque $\beta_{p'_n}^{(n)} = 1$ et $p'_n > p_n$.

Par ailleurs,

$$\sum_{i=0}^{p_n} \alpha_i^{(n)} Y_n^i \frac{\Delta s_n}{\|\Delta s_n\|} = D_n \frac{e_n}{\|e_n\|} \frac{\|e_n\|}{\|\Delta s_n\|}$$

et $\|\Delta D_n\| \leq \|(A - I + A_n)\| \; \|e_n\|$

en passant a la limite quand $n$ tend vers l'infini nous obtenons

$$d_i^{(n)} = o(1) \quad \text{pour} \quad i = 0, \ldots p_n$$

et comme $d_{p_n}^{(n)} = 1$, nous avons une contradiction et

donc $p'_n = p_n$,

d'où
$$\sum_{i=0}^{p_n} \beta_i^{(n)} Y_n^{\,i} \, \Delta D_n = 0 \qquad \text{avec } \beta_{p_n}^{(n)} = 1 .$$

et donc
$$\sum_{i=0}^{p_n} \beta_i^{(n)} \Delta D_{n+i} = 0 .$$

On vient donc de démontrer que les vecteurs $\Delta D_n, \ldots, \Delta D_{n+p_n}$ sont dépendants, et par définition de $k_n$ nous avons :

$$\exists N' \quad , \quad k_n = p_n \qquad \forall n > N'$$

■

La démonstration de ce théorème a été inspirée de la preuve de la proposition 2-3-1 de [3].

## ④ Résolution des systèmes d'équations non linéaires.

Soit à résoudre le système d'equations non linéaires :

$$x = G(x),$$

où $G$ est une application de $\mathbb{C}^p$ dans lui même et soit $\overset{*}{x}$ un point fixe de $G$.

Nous allons étudier la méthode suivante

(M)

> $\overset{\circ}{x}$ un vecteur donné
>
> iteration $k+1$
>
> On pose $s_0 = x^k$.
>
> puis et $s_{i+1} = G(s_i)$ pour $i = 0, -, d_k$,
>
> avec $d_k$ le degré du polynome minimal de $G'(\overset{*}{x})$ pour $x^k - \overset{*}{x}$.
>
> Puis prendre
>
> $$x^{k+1} = s_0 - \Delta S_{0,d_k} \left( \overset{*}{Y}{}_{d_k}^{(0)} \Delta^2 S_{0,d_k} \right)^{-1} \overset{*}{Y}{}_{d_k}^{(0)} \Delta s_0$$

Le cas où $Y_{d_k}^{(0)} = \Delta S_{0,d_k}$, c'est à dire l'application de la méthode MPE a été étudiée par Skelboe [79] qui a montré que cette méthode est à "convergence quadratique". Malheureusement dans sa démonstration il y'a une faille, voir [81]. Il a en fait montré que $\|x^{k+1} - \overset{*}{x}\| \leq \alpha_k \|x^k - \overset{*}{x}\|^2$, mais il n'a pas prouvé que $\alpha_k$ était borné.

Beuneu dans [3] a montré que toutes les méthodes de "type (M)" sont a "convergence quadratique". Dans sa démonstration il y'a deux failles, la première est du même type que celle faite par Skelboe, la seconde est la suivante,

pour démontrer le lemme 2-2 [3] qui est à la base de sa démonstration, Beuneu utilise la relation :

$$* \qquad \sum_{i=1}^{d_k} a_i^{(k)} \varphi_i (x^k - \overset{*}{x}) = O(\|x^k - \overset{*}{x}\|)$$

avec $\varphi_i (x^k - \overset{*}{x}) = O(\|x^k - \overset{*}{x}\|) \quad i = 1, \dots, d_k$.

Or cette relation n'est pas toujours vérifiée comme le montre l'exemple suivant : $\varphi_i(k) = \frac{1}{(k+i-1)^2} \quad i = 1, 2$ ;

nous avons : $\varphi_i(k) = O(\frac{1}{k^2})$ , $d_k = 2$ , $a_1^{(k)} = k^2$ et $a_2^{(k)} = -k^2+1$

$$a_1^{(k)} \varphi_1(k) + a_2^{(k)} \varphi_2(k) = \frac{2}{(k+1)} \neq O(\frac{1}{k^2}) .$$

La relation * serait vérifiée si par exemple les nombres $a_i^{(k)}$ étaient bornés.

---

Nous allons dans ce qui suit faire les hypothèses suivantes :

$$G : D \subset \mathbb{C}^p \longrightarrow \mathbb{C}^p$$

Posons $J = G'(\overset{*}{x})$

(H) :

1) $J - I$ est inversible , $M = \|(J-I)^{-1}\|_2$

2) $\|G'(x) - G'(y)\|_2 \leq L \|x-y\|_2 \quad \forall x \in D, \forall y \in D$ .

3) $d_k(x) = \det(\overset{*}{A}_k(x) A_k(x))^{1/2}$

où $A_k(x) = \left( \frac{G(x) - x}{\|G(x) - x\|_2}, \dots, \frac{G^{(d_k)}(x) - G^{(d_k - 1)}(x)}{\|G^{(d_k)}(x) - G^{(d_k-1)}(x)\|_2} \right)$

4) $\exists D_0 \subset D$, $D_0$ convexe, $\quad \forall x \in D_0 / \{\overset{*}{x}\} \; : \; d_k(x) \neq 0$

et

$$Z_k(x) = \frac{(d_k)^{\frac{d_k}{2}} \, ML}{\alpha_k(x)} \max_{0 \leq i \leq d_k} \| G^{(i)}(x) - \overset{*}{x} \|_2 < \frac{2}{5} .$$

théorème 17

> · Faisons l'hypothèse (H),
> alors $\forall x_0 \in D_0$, la suite $\{x^k\}$ qui définie la méthode (M) converge vers $\overset{*}{x}$.
> si de plus
> $\exists \alpha > 0 \; : \; \forall x \in D_0 / \{\overset{*}{x}\} \cdot d_k(x) > \alpha$
> alors la suite $\{x^k\}$ converge quadratiquement

preuve: Posons $F(x) = G(x) - x$

et $g(x) = F(x) - F'(\overset{*}{x})(x - \overset{*}{x})$

D'après [60], nous avons:

$$\| g(x) \| \leq \tfrac{1}{2} L \| x - \overset{*}{x} \|^2 \qquad (a)$$

et

$$\| g(x) - g(y) \| \leq L \| x - y \| \max \left( \| x - \overset{*}{x} \|, \| y - \overset{*}{x} \| \right) \quad (b) .$$

. $\Delta s_0 = G(x^k) - x^k$

$\Delta s_1 = G(G(x^k)) - G(x^k)$ et $\Delta^2 s_0 = F(G(x^k)) - F(x^k)$

et donc

$\Delta s_i = G^{(i+1)}(x^k) - G^{(i)}(x^k)$ et $\Delta^2 s_i = F(G^{(i+1)}(x)) - F(G^{(i)}(x))$

pour $i = 0, \ldots, d_k$ .

Posons en plus $\quad G_k = \left\{ \Delta^2 S_{0,d_k} - F'(\overset{*}{x}) \, \Delta S_{0,d_k} \right\}$

donc

$$C_k = \left( g(G(x^k)) - g(x^k) \; , \; \ldots\ldots\ldots , \; g\left(G^{(d_k)}(x^k)\right) - g\left(G^{(d_k-1)}(x^k)\right) \right).$$

Posons

$$\tilde{C}_k = \left( \frac{g(G(x^k)) - g(x^k)}{\| G(x^k) - x^k \|_2} \; , \; \ldots\ldots\ldots , \; \frac{g\left(G^{(d_k)}(x^k)\right) - g\left(G^{(d_k-1)}(x^k)\right)}{\| G^{(d_k)}(x^k) - G^{(d_k-1)}(x^k) \|} \right)$$

et $\quad B_k = F'(\overset{*}{x})^{-1} \; C_k \quad \Delta S^{+}_{0,d_k}$

$$= F'(\overset{*}{x})^{-1} \; \tilde{C}_k \; \overset{+}{A}_k(x^k) \; ,$$

par conséquent

$$\|B_k\|_2 \leq M \; \| \tilde{C}_k \|_2 \; \| A^{+}_k(x^k) \|_2$$

$$\leq M \; \| \tilde{C}_k \|_F \; \| A^{+}_k(x^k) \|_2 .$$

D'après le lemme 2, $\| A^{+}_k(x^k) \|_2 \leq \dfrac{\| A_k(x^k) \|_F^{d_k-1}}{d_k(x^k)} = \dfrac{d_k^{(d_k-1)/2}}{\alpha_k(x^k)}$.

$$\| \tilde{C}_k \|_F^2 = \sum_{i=1}^{d_k} \left\| \frac{g(G^{(i)}(x^k)) - g(G^{(i-1)}(x^k))}{\| G^{(i)}(x^k) - G^{(i-1)}(x^k) \|_2} \right\|_2^2$$

et d'après ( b)

$$\| \tilde{C}_k \|_F \leq \sqrt{d_k} \; L \max_{0 \leq i \leq d_k} \| G^{(i)}(x^k) - \overset{*}{x} \|_2 .$$

En regroupant les deux dernières inégalités, on obtient:

$$\|B_k\|_2 \leq M L \; d_k^{(d_k/2)} \; \frac{\max_{0 \leq i \leq d_k} \| G^{(i)}(x^k) - \overset{*}{x} \|_2}{\alpha_k(x^k)} .$$

Si $\quad x^k \in D_0 \qquad \|B_k\|_2 < 1$ et donc $I + B_k$ est inversible.

et comme $\qquad \Delta^2 S_{0,d_k} = F'(\overset{*}{x}) \; (I + B_k) \; \Delta S_{0,d_k}$

alors $\qquad \Delta S_{0,d_k} = (I + B_k)^{-1} \; F'(\overset{*}{x})^{-1} \; \Delta^2 S_{0,d_k} .$

Posons $D_k = (I - (I+B_k)^{-1}$, nous avons d'après le lemme de pertubation ( lemme 2, Chapitre I)

$$\| D_k \|_2 \leqslant \frac{\|B_k\|_2}{1 - \|B_k\|_2} .$$

Par définition de la suite $\{x^k\}$ :

$$x^{k+1} = x^k - \Delta S_{0,d_k} ( \overset{*}{Y}{}^{(0)}_{d_k} \Delta^2 S_{0,d_k})^{-1} \overset{*}{Y}_k \Delta s_0 ,$$

$$= x^k - (I - D_k) F'(\overset{*}{x})^{-1} \Delta^2 S_{0,d_k} (\overset{*}{Y}_{d_k} \Delta^2 S_{0,d_k})^{-1} \overset{*}{Y}_{d_k} .$$

Par des arguments similaires a ceux utilisés lors de la démonstration du théorème 17, on montre que les vecteurs $\Delta s_0, —, \Delta s d_k$ sont linéairement indépendants et donc $\Delta s_0 \in V[\Delta^2 s_0, \dots, \Delta^2 s d_{k-1}]$, et en utilisant le lemme 4, nous obtenons :

$$x^{k+1} = x^k - (I - D_k) F'(\overset{*}{x})^{-1} F(x^k)$$

et

$$x^{k+1} - \overset{*}{x} = (D_k - I) F'(x^k)^{-1} g(x^k) + D_k (x^k - \overset{*}{x}) .$$

$$\|x^{k+1} - \overset{*}{x}\|_2 \leqslant \|D_k - I\|_2 M \|g(x^k)\|_2 + \|D_k\|_2 \|x^k - \overset{*}{x}\|_2 ,$$

et comme d'une part

$$\|D_k - I\|_2 \leqslant \frac{1}{1 - \|B_k\|_2}$$

et d'autre part $\|g(x^k)\|_2 \leq \frac{1}{2} L \|x^k - \overset{*}{x}\|_2$ c'est la relation (a);

alors

$$\|x^{k+1} - \overset{*}{x}\|_2 \leq \frac{3}{2} \left( \frac{z_k(x^k)}{1 - z_k(x^k)} \right) \|x^k - \overset{*}{x}\|_2 . \quad (c)$$

Posons $z = \max\limits_{\substack{x \in D_0 \\ x \neq \overset{*}{x}}} z_k(x)$ et donc :

$$\|x^{k+1} - \overset{*}{x}\| \leqslant \frac{3z}{2(1-z)} \|x^k - \overset{*}{x}\|$$

et comme $\frac{3z}{2(1-z)} = \rho < 1$ et $x^k \in D_0$ alors $x^{k+1} \in D_0$

Par recurrence $x^k \in D_0 \quad \forall k$,

de plus $\lim_k x^k = \overset{*}{x}$, ce qui termine la démonstration de la première partie du théorème.

$$z_k(x^k) = \frac{(d_k)^{d_k/2} \, ML}{d_k(x)} \max_{0 \leq i \leq d_k} \| G^{(i)}(x^k) - \overset{*}{x} \| \leq \frac{ML \, p^{p/2}}{\alpha} K \|x^k - \overset{*}{x}\|_2$$

$\lim_k z_k(x^k) = 0$ et donc $\exists A$, $\left| \frac{1}{1 - z_k(x)} \right| \leq A \quad \forall k$.

En reportant ces majorations dans (c), on obtient

$$\|x^{k+1} - \overset{*}{x}\|_2 \leq \frac{3}{2} \frac{ML \, p^{p/2}}{\alpha} AK \|x^k - \overset{*}{x}\|^2$$

et donc la suite $\{x^k\}$ est d'ordre 2. ■

Pour clore ce chapitre, nous allons faire quelques observations concernant les trois derniers exemples numériques traités par Smith, Ford et Sidi dans [81].

<u>Exemple n°6</u> (Non linear iteration)

Pour cet exemple nous avons obtenu le point fixe suivant :

$$\overset{*}{x} = \begin{bmatrix} 1 \\ 1.398\ 773\ 006\ 134\ 969 \\ 1.582\ 282\ 208\ 588\ 957 \\ 1.644\ 417\ 177\ 914\ 110 \end{bmatrix}, \quad \boxed{d_k = 3}, \quad \forall k$$

et ceci avec les 4 méthodes (VEA, TEA, MPE, RRE).

Dans [81], toutes les méthodes ont donné un autre point fixe qui est le vecteur $(1, 1, 1, 1)^T$

<u>L'Exemple n°7</u> (Quadratic iteration).

$$\overset{0}{x} = 2\overset{*}{x} \quad , \quad \overset{*}{x} = (1,1,1,1)^T$$

$$G'(\overset{*}{x}) = \begin{bmatrix} 0.75 & 0.01 & 0.05 & 0 \\ 0.01 & 0.75 & 0 & 0.05 \\ 0.05 & 0 & 0.75 & 0.01 \\ 0 & 0.05 & 0.01 & 0.75 \end{bmatrix} = B$$

$$e^0 = \overset{0}{x} - \overset{*}{x} = \overset{*}{x}$$

$$B e^0 = 1.81\, \overset{*}{x} ,$$

et donc $d_0 = 1$. De plus on peut facilement montrer que $\overset{1}{x} = a_1 \overset{*}{x} + b_1 f$ où $f = (-1, 0, 0, -1)$

$$B\overset{*}{x} = 1.81\,\overset{*}{x} \qquad \text{et} \qquad Bf = 0.08\,\overset{*}{x} + 0.69\, f$$

$$B e^1 = B(x^1 - \overset{*}{x}) = (a_1 - 1)\, B\overset{*}{x} + b_1 Bf$$

$$= \{(a_1-1)1.81 + 0.06\, b_1\}\overset{*}{x} + \{0.69\, b_1\} f$$

et donc $V[e^1, Be^1, Be^2] = V[\overset{*}{x}, f]$,

d'où $d_1 = 2$, et par recurrence $d_k = 2, \ \forall k \geqslant 1$.

Dans [81] il est écrit que $d_k = 3, \forall k$ ; ce qui est faux comme on vient de le démontrer.

De plus pour $k \geqslant 1$ le rang $(\Delta S_{0,3}) =$ rang$(\Delta^2 S_{0,3}) = 2$, donc on ne peut plus utiliser la formulation donnée dans [81], puisque celle-ci suppose que la matrice $\Delta S_{0,d_k}$ soit de rang maximal.

Exemple n°8 (Quadratic iteration with two solutions)

$$\tilde{x} = (1.5, 1.5, 1.5, 1.5)^T \qquad \overset{*}{x} = (1, 1, 1, 1)$$

$$G'(\overset{*}{x}) = \begin{bmatrix} 3.4 & -3.7 & 2.4 & -0.6 \\ 2.4 & -2.5 & 2.2 & -0.6 \\ 2.4 & -3.6 & 3.6 & -0.6 \\ 2.8 & -5.2 & 4.8 & -0.9 \end{bmatrix} = B$$

$$e^0 = a_0 \overset{*}{x}$$

$$Be^0 = 1.5\, a_0 \overset{*}{x}$$

et donc $d_0 = 1$, et comme $x^0 = (a_0 + 1)\overset{*}{x}$

On montre facilement que $e^1 = a_1 \overset{*}{x}$, et par récurrence $d_k = 1, \forall k$. Ce qui explique pourquoi la méthode RRE n'a pas convergé quand on a pris deux comme degré du polynome minimal de $G'(\overset{*}{x})$ pour $e^k$, voir [81]. Si on prend $d_k = 1$, toutes les méthodes coincident. De plus pour cet exemple toutes les méthodes de type $\varepsilon$-algorithme ont donné le même résultat. Pour $k = 1, 2$ où $3$, ces méthodes ont toutes convergé vers $\overset{*}{x}$. Dans le cas où $k = 4$, elles ont convergé vers $3\overset{*}{x}$. l'$\varepsilon$-topologique donne le même résultat quelque soit le vecteur $y$.

Les résultats que l'on a obtenu sont similaires a ceux donnés par TAN dans [84] en ce qui concerne l'$\varepsilon$-topologique.

# REFERENCES


[1] J. BEUNEU

Méthodes de projection-minimisation pour les problèmes linéaires, RAIRO Analyse Numérique, 17 (1983) 221-248.

[2] J. BEUNEU

Méthodes de projection à convergence finie; remarques sur leur forme incomplète, ANO 88, Université de Lille, 1982.

[3] J. BEUNEU

Some projection méthods for non linear problems, ANO 134, Université de Lille, 1984.

[4] I.S BEREZIN, N.P. ZHIDKOV

Computing methods, Pergamon Press, 1965.

[5] L. BITTNER

Mehrpunktverfahren zur Auflözung von Gleichungssystemen, Z. Angew. Math. Mech., 43(1963) 111-126.

[6] C. BREZINSKI

Some results in the theory of the vector ε-algorithm, Linear Alg. App., 8(1974) 77-86.

[7] C. BREZINSKI
Computation of the eigenelements of matrix by the $\varepsilon$-algorithm, Linear Alg. App., 11(1975) 7-20.

[8] C. BREZINSKI
Généralisations de la transformation de Shanks, de la table de Padé et de l'$\varepsilon$-algorithme, Calcolo, 12(1975) 317-360.

[9] C. BREZINSKI
Accélération de la convergence en analyse numérique, LNM 584, Springer-Verlag, Heidelberg, 1977.

[10] C. BREZINSKI
Algorithmes d'accélération de la convergence. Etude numérique, Editions Technip, Paris, 1978.

[11] C. BREZINSKI
Sur le calcul de certains rapports de déterminants, dans "Padé approximation and its application", L. Wuytack ed., LNM 765, Springer-Verlag, Heidelberg, 1978.

[12] C. BREZINSKI
A general extrapolation algorithm, Numer. Math. 35(1980) 175-187

[13] C. BREZINSKI
Padé type approximation and general orthogonal polynomials. ISNM vol. 50, Birkhaüser-Verlag, Basel, 1980.

[14] C. BREZINSKI
Some new convergence acceleration methods, Math. Comp., 39 (1982) 133-145.

[15] C. BREZINSKI
About Henrici's method for non linear equations, Symposium on numerical analysis and computational complex analysis, Zürich, Août 15-17, 1983, non publié.

[16] C. BREZINSKI
Recursive interpolation, extrapolation and projection, J. Comp. App. Math., 9 (1983) 369-379.

[17] C. BREZINSKI
Some determinantal identities in a vector space with applications, dans "Padé approximation and its applications", H. Werner and H.J. Bünger eds., LNM 1071, Springer-Verlag, Heidelberg, 1984.

[18] C. BREZINSKI
Composite sequence transformations, Numer. Math., 46(1985) 311-321.

[19] C. BREZINSKI
Bordering methods and progressive forms for sequence transformations. Zastos. Math., à paraître.

[20] C. BREZINSKI
Other manifestations of the Schur complement, à paraître.

[21] C. BREZINSKI, H. SADOK
Vector sequence transformations and fixed point methods.
Dans "Numerical Methods in laminar and turbulent flows",
C. Taylor and al. eds, Primeridge Press, Swansea, 1987.

[22] O.P. BURDAKOV
On superlinear Convergence of Some stable Variants of the Secant Method, Z. Angew. Math. Mec., 66(1986)615-622.

[23] S. CABAY, L.W. JACKSON
"A polynomial extrapolation method for finding limits and antilimits of vector sequences", SIAM J. Numer. Anal., 13(1976),734-752.

[24] F. CORDELLIER
Particular rules for the vector $\epsilon$-algorithm, Numer. Math., 27(1977) 203-207.

[25] J.W. DANIEL, W.B. GRAGG, L. KAUFMAN, G.W. STEWART
Reorthogonalisation and stable algorithms for updating the Gram-Schmidt QR factorisation, Math. Comp., 30(1976) 772-795.

[26] R.P. EDDY
Extrapolating to the limit of a vector sequence, Dans "Information linkage between Applied Mathematics and Industry", (P.C.C. Wang, ed.), Academic Press, New-York, 1979, pp 387-396.

[27] S.C EISENSTAT, H.C. ELMANN, M. SCHULTZ
"Variational iterative methods for nonsymmetric systems of linear equations", SIAM J. Numer. Anal., 20 (1983) 345-357.

[28] D.K. FADDEEV, V.N. FADDEEVA
Computational Methods of linear Algebra, W.H. Freeman, San Francisco, 1963.

[29] R. FLETCHER
Conjugate gradients methods for undefinite systems. Dans "Numerical analysis, Dundee 1975", Lecture Notes in Mathematics 506, Springer-Verlag, Heidelberg, 1976.

[30] W.F. FORD, A. SIDI
Recursive algorithms for vector extrapolation methods, à paraître.

[31] F.R. GANTMACHER
théorie des matrices, Vol. 1, Dunod, Paris, 1966.

[32] GAUSS

[33] E. GEKELER
On the solution of systems of equations by the epsilon algorithm of Wynn, Math. Comp., 26 (1972) 427-436.

[34] B. GERMAIN-BONNE
Accelération de la convergence par projection, Seminaires d'analyse numérique, Grenoble, 1975.

[35] B. GERMAIN-BONNE
Estimation de la limite de suites et formalisation de procédés d'accélération de la convergence, thèse d'Etat, Univ. de Lille I, 1978.

[36] P.E. GILL, G.H. GOLUB, W. MURRAY, M.A. SAUNDERS,
Methods for modifying matrix factorisations, Math. Comp. 28(1974) 505-535.

[37] G.H. GOLUB, G.A. MEURANT
Résolution numérique des grands systèmes linéaires, Eurolles, Paris, 1983.

[38] W.B. GRAGG, G.W. STEWART
A stable variant of the secant method for solving nonlinear equations, SIAM J. Numer. Analys. 13 (1976) 889-903.

[39] P.R. GRAVES-MORRIS
Vector valued rational interpolants I, Numer. Math. 42 (1983) 331-348.

[40] A. HADJIDIMOS
A survey of the iterative methods for the solution of linear systems by extrapolation, relaxation and other techniques, J. Comput. Appl. Math. 20 (1987) 37-51.

[41] L.A. HAGEMAN, F.T. LUK, D.M. YOUNG
On the equivalence of certain iterative acceleration methods, SIAM J. Numer. Anal., 17(1980)852-873.

[42] P. HENRICI
Elements of numerical analysis, Wiley, New-York, 1964.

[43] M.R. HESTENES, E. STIEFEL
Method of conjugate gradients for solving linear systems, J. Res. NBS, 49(1952) 409-436.

[44] E. ISAACSON, H.B. KELLER
Analysis of Numerical Methods, Wiley, New-York, 1966.

[45] J. JANKOWSKA
Theory of multivariate of the secant method, SIAM J. Numer. Anal., 16(1979) 547-562.

[46] KH. JBILOU
Methodes d'extrapolation et de projection. Application aux suites de vecteurs. thèse de 3$^{eme}$ cycle, Univ. de Lille I, 1988.

[47] O.G. JOHNSON, C.A. MICHELLI, G. PAUL
Polynomial preconditioners for conjugate gradients calculations, SIAM J. Numer. Anal., 20(1983) 362-376.

[48] S. KANIEL, J. STEIN
Least-Square acceleration of iterative methods for linear equations, Journal of Optimisation theory and applications, 14(1974) 431-437.

[49] T. KATO
Estimation of iterated matrices with application to the von Neumann condition, Numer. Math., 2(1960) 22-29.

[50] A. KORGANOFF
Méthodes de calcul numérique, tome 1, Dunod, Paris, 1961.

[51] P. LASCAUX, R. THEODOR
Analyse numérique matricielle appliquée à l'art de l'ingenieur, tome 1, Masson, Paris, 1986.

[52] C.L. LAWSON, R.J. HANSON
Solving least Squares Problems, Prentice-Hall, Englewood cliffs, NJ, 1974.

[53] G. LEMAITRE
L'itération rationnelle, Bull. Acad. Roy. Belg. 5e ser. cl. Sa., 28 (1942) 347–354.

[54] R. LUDWIG
Verbesserung einer Iterationsfolge bei gleichungssystemen, Z. Angew. Math. Mech., 32 (1952) 232-234.

[55] T. MANTEUFFEL
The tchebychev iteration for non symmetric linear systems, Numer. Math., 28 (1977) 307-327

[56] J.M. MARTINEZ
Three new algorithms based on the sequential secant method, BIT, 19 (1979) 236-243

[57] J.M. MARTINEZ, T.L. LOPES
Combination of the Sequential Secant Method and Broyden's Method with Projected Updates, Computing, 25(1980) 379-386.

[58] J.B. McLEOD
A note on the $\varepsilon$-algorithm, Computing, 7(1971) 17-24.

[59] M. MEŠINA
Convergence acceleration for the iterative solution of the equation $X=AX+f$, Comput. Methods Appl. Mech. Engrg, 10(1977) 165-173.

[60] J.M. ORTEGA, W.C. RHEINBOLDT
Iterative solution of non - equations in several variables, Academic Press, New-York, 1970.

[61] A.M. OSTROWSKI
Solution of equations in euclidean and Banach spaces. Academic Press, New-York and London, 1973.

[62] J. DE PILLIS, M. NEUMANN
Iterative methods with k-part, splitting, IMA J. Numer. Anal., 1(1981) 65-79.

[63] B.P. PUGACHEV
Application of a generalisation of Vandermonde's determinant, U.S.S.R. Comput. Maths. Maths Phys., 8 (1968) 207-215.

[64] B.P PUGACHEV
The use of badly converging iterative processes for the solution of sets of linear equations, U.S.S.R. Comput. Maths. Maths Phys., 8 (1968) 172-176.

[65] B.P. PUGACHEV
Acceleration of the convergence of iterative processes and a method of solving systems of non-linear equations, U.S.S.R. Comput. Maths. Maths Phys., 17 (1978) 199-207

[66] W.C. PYE, T.A. ATCHISON
An algorithm for the computation of higher order G-transformation. SIAM J. Numer. Anal., 10 (1973) 1-7

[67] M.F. RACHDI
Sur quelques algorithmes utilisant l'ε-convergence dans la recherche des éléments propres d'une matrice, thèse de 3ème cycle, Univ. de Tunis, 1980

[68] Y. SAAD
Krylov subspace methods for solving large unsymmetric linear systems, Math. Comp., 37 (1981) 105-126.

[69] Y. SAAD
The Lanczos biorthogonalisation algorithm and other oblique methods for solving large unsymmetric linear systems, SIAM J. Numer. Anal., 19(1982) 485-506.

[70] H. SADOK
Accéleration de la convergence de suites par utilisation des transformations composites, thèse de 3ème cycle, Univ. de Lille I, 1986.

[71] P. A. SAYLOR
Use of the singular value decomposition with the Manteuffel algorithm for non symmetric linear systems, SIAM J. on Sci. Stat. Comput., 1(1980) 210-222.

[72] J. SCHMIDT
Rate of convergence of the Regula-Falsi and Steffensen processes in a Banach space, Z. Angew. Math. Mech. 46(1966) 146-148.

[73] H. SCHWETLICK
Über die Realisierung und Konvergenz Von Mehrschrittverfahren zur iterativen Lösung nichtlinearer Gleichungen, Z. Angew. Math. Mech., 51 (1974) 479-493.

[74] D. SHANKS
Non-linear transformations of divergent and slowly convergent sequences, J. Math. Phys., 34 (1955) 1-42.

[75] A. SIDI
Convergence and stability propriety of minimal polynomial and reduced rank extrapolation algorithms, SIAM J. Numer. Anal., 23(1986)197-209.

[76] A. SIDI
Extrapolation vs Projection methods for linear systems of equations, à paraître.

[77] A. SIDI, J. BRIDGER
Convergence and stability analysis for some vector extrapolation methods in the presence of defective iteration matrices, à paraître.

[78] A. SIDI, W. F. FORD, D.A. SMITH
Acceleration of convergence of vector sequences, SIAM J. Numer. Anal., 23(1986) 178-196.

[79] S. SKELBOE
Computation of the periodic steady-state reponse of nonlinear networks by extrapolation methods, IEEE Trans. Circuits and Systems, 27(1980)161-175.

[80] F. SLOBODA
A parallel projection method for linear algebric systems, Aplikace Matematiky, 23(1978)185-198

[81] D.A. SMITH, W.F. FORD, A. SIDI
Extrapolation methods for vector sequences, SIAM Review, 29(1987) 199-233

[82] G.W. STEWART
Conjugate Direction Methods for solving systems of linear Equations, Numer. Math., 21 (1973) 285-297.

[83] G.W. STEWART
Introduction to Matrix Computations, Academic Press, New-York, 1973.

[84] Roger C.E. TAN
Implementation of the topological ε-algorithm, à paraître dans SIAM J. Scientific and Statistical computing.

[85] S. YU. ULM
Extension of steffensen's method for solving non linear operator equations, USSR Comput. Maths. Math. Phys., 4 (1964) 159-165.

[86] J. VAN ISEGHEM
Approximants de Padé vectoriels, thèse d'Etat, Univ. de Lille I, 1987.

[87] R.S. VARGA
Matrix iterative Analysis, Prentice-Hall, Englewood Cliffs, NJ, 1962.

[88] Y.V. VOROBYEV
Method of Moments in Applied Mathematics, Gordon and Breach, New-York, 1965.

[89] U. WEGNER
Contributti alla teoria dei procedimenti iterativi per la risoluzione numerica dei sistemi di equazioni lineari algebriche, Mem. Accad. Naz. Lincei., 4 (1953) 1-49.

[90] O. WIDLUND
A Lanczos method for a class of nonsymmetric systems of linear equations, SIAM J. Numer. Anal., 15 (1978) 801-812.

[91] J. WIMP
Sequences transformations and their applications.
Academic Press, New-York, 1981.

[92] PH. WOLFE
The secant method for solving nonlinear equations,
Comm. ACM, 2 (1959) 12-13.

[93] P. WYNN
On a device for computing the $e_m(S_n)$ transformation, MTAC,
10 (1956) 91-96

[94] P. WYNN
Acceleration techniques for iterated vector and matrix problems,
Math. Comp., 16 (1962) 301-322.

[95] P. WYNN
Upon a conjecture concerning a method for solving linear equations, and certain other matters. MRC technical summary report 626, Madison, 1966.

## RESUME

Cette thèse est consacrée à l'étude de l'accélération de la convergence de suites de vecteurs complexes. Dans le premier chapitre, nous étudions la transformation d'Henrici qui est une extension naturelle du procédé $\Delta^2$ d'Aitken au cas vectoriel. Le second chapitre est consacré aux transformations composites vectorielles. Ces transformations ont été définies et étudiées par C. BREZINSKI dans le cas scalaire.

Nous proposons deux extensions au cas vectoriel, que nous utilisons pour montrer la connexion existant entre les méthodes d'accélérations de la convergence et les méthodes de point fixe. Nous donnons aussi deux généralisations de la méthode Regula-Falsi et montrons que l'une des deux méthodes est d'ordre $(1+\sqrt{5})/2$.

Dans le troisième chapitre, nous passons en revue les principales méthodes d'accélération de la convergence et donnons les résultats les plus significatifs. Nous terminons ce chapitre par l'étude des problèmes non linéaires, nous montrons qu'en particulier toute une classe de méthodes est à convergence quadratique, lorsque ces méthodes sont utilisées pour résoudre des systèmes d'équations non linéaires.