50376
1982
69

THÈSE

présentée à

L'UNIVERSITÉ DES SCIENCES ET TECHNIQUES DE LILLE

pour obtenir le titre de

DOCTEUR DE 3ème CYCLE

MATHEMATIQUES APPLIQUEES

par

Sliman ACHAKIR

CONNEXION ENTRE LES METHODES DE POINT FIXE ET D'ACCELERATION DE LA CONVERGENCE.

Soutenue le 23 juin 1982, devant la Commission d'Examen

Membres du Jury :	Président	P. POUZET
	Rapporteur	C. BREZINSKI
	Examinateurs	J. DENEL
		B. GERMAIN BONNE
		T. HÅVIE
		J. WIMP

PROFESSEURS 1ère CLASSE

M.	BACCHUS Pierre	Mathématiques
M.	BEAUFILS Jean-Pierre (dét.)	Chimie
M.	BIAYS Pierre	G.A.S.
M.	BILLARD Jean	Physique
M.	BONNOT Ernest	Biologie
M.	BOUGNON Pierre	Mathématiques
M.	BOURIQUET Robert	Biologie
M.	CELET Paul	Sciences de la Terre
M.	COEURE Gérard	Mathématiques
M.	CONSTANT Eugène	I.E.E.A.
M.	CORDONNIER Vincent	I.E.E.A.
M.	DEBOURSE Jean-Pierre	S.E.S.
M.	DELATTRE Charles	Sciences de la Terre
M.	DURCHON Maurice	Biologie
M.	ESCAIG Bertrand	Physique
M.	FAURE Robert	Mathématiques
M.	FOURET René	Physique
M.	GABILLARD Robert	I.E.E.A.
M.	GRANELLE Jean-Jacques	S.E.S.
M.	GRUSON Laurent	Mathématiques
M.	GUILLAUME Jean	Biologie
M.	HECTOR Joseph	Mathématiques
M.	HEUBEL Joseph	Chimie
M.	LABLACHE COMBIER Alain	Chimie
M.	LACOSTE Louis	Biologie
M.	LANSRAUX Guy	Physique
M.	LAVEINE Jean-Pierre	Sciences de la Terre
M.	LEBRUN André	C.U.E.E.P.
M.	LEHMANN Daniel	Mathématiques
Mme	LENOBLE Jacqueline	Physique
M.	LHOMME Jean	Chimie
M.	LOMBARD Jacques	S.E.S.
M.	LOUCHEUX Claude	Chimie
M.	LUCQUIN Michel	Chimie

M.	MAILLET Pierre	S.E.S.
M.	MONTREUIL Jean	Biologie
M.	PAQUET Jacques	Sciences de la Terre
M.	PARREAU Michel	Mathématiques
M.	PROUVOST Jean	Sciences de la Terre
M.	SALMER Georges	I.E.E.A.
Mme	SCHWARTZ Marie-Hélène	Mathématiques
M.	SEGUIER Guy	I.E.E.A.
M.	STANKIEWICZ François	Sciences Economiques
M.	TILLIEU Jacques	Physique
M.	TRIDOT Gabriel	Chimie
M.	VIDAL Pierre	I.E.E.A.
M.	VIVIER Emile	Biologie
M.	WERTHEIMER Raymond	Physique
M.	ZEYTOUNIAN Radyadour	Mathématiques

PROFESSEURS 2ème CLASSE

M.	AL FAKIR Sabah	Mathématiques
M.	ANTOINE Philippe	Mathématiques
M.	BART André	Biologie
Mme	BATTIAU Yvonne	Géographie
M.	BEGUIN Paul	Mathématiques
M.	BELLET Jean	Physique
M.	BKOUCHE Rudolphe	Mathématiques
M.	BOBE Bernard	S.E.S.
M.	BODART Marcel	Biologie
M.	BOILLY Bénoni	Biologie
M.	BONNELLE Jean-Pierre	Chimie
M.	BOSQ Denis	Mathématiques
M.	BREZINSKI Claude	I.E.E.A.
M.	BRUYELLE Pierre (Chargé d'enseignement)	Géographie
M.	CAPURON Alfred	Biologie
M.	CARREZ Christian	I.E.E.A.
M.	CHAMLEY Hervé	E.U.D.I.L.

M.	CHAPOTON Alain	C.U.E.E.P.
M.	COQUERY Jean-Marie	Biologie
Mme	CORSIN Paule	Sciences de la Terre
M.	CORTOIS Jean	Physique
M.	COUTURIER Daniel	Chimie
Mle	DACHARRY Monique	Géographie
M.	DEBRABANT Pierre	E.U.D.I.L.
M.	DEGAUQUE Pierre	I.E.E.A.
M.	DELORME Pierre	Biologie
M.	DEMUNTER Paul	C.U.E.E.P.
M.	DE PARIS Jean-Claude	Mathématiques
M.	DEVRAINNE Pierre	Chimie
M.	DHAINAUT André	Biologie
M.	DORMARD Serge	S.E.S.
M.	DOUKHAN Jean-Claude	E.U.D.I.L.
M.	DUBOIS Henri	Physique
M.	DUBRULLE Alain	Physique
M.	DUEE Gérard	Sciences de la Terre
M.	DYMENT Arthur	Mathématiques
M.	FLAMME Jean-Marie	E.U.D.I.L.
M.	FONTAINE Hubert	Physique
M.	GERVAIS Michel	S.E.S.
M.	GOBLOT Rémi	Mathématiques
M.	GOSSELIN Gabriel	S.E.S.
M.	GOUDMAND Pierre	Chimie
M.	GREVET Patrice	S.E.S.
M.	GUILBAULT Pierre	Biologie
M.	HANGAN Théodore	Mathématiques
M.	HERMAN Maurice	Physique
M.	JACOB Gérard	I.E.E.A.
M.	JACOB Pierre	Mathématiques
M.	JOURNEL Gérard	E.U.D.I.L.
M.	KREMBEL Jean	Biologie
M.	LAURENT François	I.E.E.A.
Mle	LEGRAND Denise	Mathématiques
Mle	LEGRAND Solange	Mathématiques (Calais)
Mme	LEHMANN Josiane	Mathématiques

M.	LEMAIRE Jean	Physique
M.	LENTACKER Firmin	G.A.S.
M.	LEVASSEUR Michel	I.P.A.
M.	LHENAFF René	G.A.S.
M.	LOCQUENEUX Robert	Physique
M.	LOSFELD Joseph	I.E.E.A.
M.	LOUAGE Francis	E.U.D.I.L.
M.	MACKE Bruno	Physique
M.	MAIZIERES Christian	I.E.E.A.
Mle	MARQUET Simone	Mathématiques
M.	MESSELYN Jean	Physique
M.	MIGEON Michel	E.U.D.I.L.
M.	MIGNOT Fulbert	Mathématiques
M.	MONTEL Marc	Physique
Mme	NGUYEN VAN CHI Régine	G.A.S.
M.	PARSY Fernand	Mathématiques
Mle	PAUPARDIN Colette	Biologie
M.	PERROT Pierre	Chimie
M.	PERTUZON Emile	Biologie
M.	PONSOLLE Louis	Chimie
M.	PORCHET Maurice	Biologie
M.	POVY Lucien	E.U.D.I.L.
M.	RACZY Ladislas	I.E.E.A.
M.	RICHARD Alain	Biologie
M.	RIETSCH François	E.U.D.I.L.
M.	ROGALSKI Marc	M.P.A.
M.	ROUSSEAU Jean-Paul	Biologie
M.	ROY Jean-Claude	Biologie
M.	SALAMA Pierre	S.E.S.
Mme	SCHWARZBACH Yvette (CCP)	M.P.A.
M.	SCHAMPS Joël	Physique
M.	SIMON Michel	S.E.S.
M.	SLIWA Henri	Chimie
M.	SOMME Jean	G.A.S.
Mle	SPIK Geneviève	Biologie
M.	STERBOUL François	E.U.D.I.L.
M.	TAILLIEZ Roger	Institut Agricole

M.	TOULOTTE Jean-Marc	I.E.E.A.
M.	VANDORPE Bernard	E.U.D.I.L.
M.	WALLART Francis	Chimie
M.	WATERLOT Michel	Sciences de la Terre
Mme	ZINN JUSTIN Nicole	M.P.A.

CHARGES DE COURS

M.	TOP Gérard	S.E.S.
M.	ADAM Michel	S.E.S.

CHARGES DE CONFERENCES

M.	DUVEAU Jacques	S.E.S.
M.	HOFLACK Jacques	I.P..A
M.	LATOUCHE Serge	S.E.S.
M.	MALAUSSENA DE PERNO Jean-Louis	S.E.S.
M.	OPIGEZ Philippe	S.E.S.

Je tiens à exprimer ma profonde gratitude à Monsieur P. POUZET, Professeur à l'Université de Lille I, qui me fait l'honneur de présider le jury de cette thèse.

Je suis très reconnaissant à Monsieur C. BREZINSKI, Professeur à l'Université de Lille I, qui m'a accueilli dans son équipe d'accélération de la convergence et qui m'a proposé ce sujet. Qu'il trouve ici l'expression de mes remerciements les plus sincères pour ses encouragements et conseils précieux.

Je suis très honoré de la présence de Messieurs J. DENEL, Professeur à l'Université de Lille I, B. GERMAIN-BONNE, Maître-assistant à l'Université de Lille I, T. HAVIE, Professeur à l'Université de Trondheim, J. WIMP, Professeur à l'Université Drexel de Philadelphie qui ont accepté de s'intéresser à ce travail et de le juger.

Mes remerciements vont à mon frère Ali, qui m'a beaucoup encouragé, à mon père, à ma mère et Youhanna pour leurs sacrifices.

Mes remerciements vont également à Madame F. TAILLY pour sa gentillesse ainsi qu'à Mademoiselle B. FIEVET qui a dactylographié cette thèse avec beaucoup de soin et de rapidité et à Monsieur et Madame DEBOCK qui l'ont imprimée.

CONNEXION ENTRE LES METHODES DE POINT FIXE ET D'ACCELERATION DE LA CONVERGENCE*

S. ACHAKIR

* *Ce travail a été réalisé dans le cadre de la subvention OTAN 027-81 : "Extrapolation methods".*

CHAPITRE I

CHAPITRE II

ETUDE DE METHODES DE POINTS FIXE DEDUITES DES ALGORITHMES (B), (B'), (E), (E'), (G), (G')

CHAPITRE III

INTERPRETATION DU E-ALGORITHME ET DU PROCEDE D'INTERPOLATION INVERSE DE GERMAIN-BONNE-WIMP (GBW) GENERALISATION DU PROCEDE D'OVERHOLT

CHAPITRE IV

APPLICATIONS DES APPROXIMANTS DE PADE A LA RESOLUTION DES EQUATIONS NON LINEAIRES

IV.1. - Utilisation directe

IV.1.1. Mise en oeuvre du procédé de Nourein

IV.1.2. Techniques de calcul

* Procédure de Nourein

** Procédure de Claessens-Loisou-Wuytack

IV.2. - Utilisation réciproque

IV.3. - Essais numériques

IV.4. - Utilisation des approximants de type Padé

CHAPITRE V

PROCEDES D'ACCELERATION DE LA CONVERGENCE ASSOCIES AUX METHODES DE :

- KING
- SCHRÖDER

CHAPITRE VI

RESOLUTION DE SYSTEMES D'EQUATIONS NON LINEAIRES

VI.1. - Méthode d'Henrici

VI.2. - Méthode basée sur le E-algorithme

VI.3. - Essais numériques

INTRODUCTION

L'objet de ce travail est de dégager le lien qui existe entre les méthodes d'accélération de la convergence et celles de recherche de point fixe. Par exemple, les méthodes Δ^2 - d'Aitken et de Steffensen, sont basées sur la même procédure. La première sert à accélérer la convergence d'une suite donnée, la seconde est un procédé itératif qui permet de générer une suite qui convergera vers une solution d'une équation non linéaire.

Le premier chapitre traite des procédés d'extrapolations polynomiales et rationnelles. Après avoir rappelé les procédés d'extrapolation généralisés, nous étudions en détail deux procédés basés sur l'interpolation inverse. Ensuite nous donnons une extension de chacun d'eux à l'aide des fonctions de plusieurs variables. Ensuite nous appliquons ces algorithmes à une classe particulière de suites ce qui nous permet de retrouver certains procédés bien connus. Cette étude ne concerne que des fonctions ayant des racines simples. Enfin nous appliquons ces résultats aux fonctions qui admettent des racines multiples. Nous obtenons un procédé d'accélération de la convergence qui rentre dans une catégorie d'algorithmes regroupant les procédés d'Overholt et de Germain-Bonne-Wimp.

Le deuxième chapitre est consacré à l'étude des méthodes de résolution des équations non linéaires déduites des algorithmes du premier chapitre. Ces méthodes se partagent en deux catégories : celles de type Steffensen et celles de type sécante. Après avoir rappelé d'une façon détaillée certaines méthodes, nous dressons un tableau de comparaison. Celle-ci est faite sur le critère de l'indice d'efficacité. Nous terminons par l'étude du cas d'une racine multiple.

Dans le troisième chapitre nous rappelons le E-algorithme et nous en donnons une interprétation géométrique. En exploitant d'une part celle-ci et d'autre part le lien qui existe entre le E-algorithme et la transformation de Shanks nous exposons deux méthodes de résolution d'équations non linéaires. Après avoir rappelé la construction du procédé d'Overholt et donné une interprétation du procédé GBW (Germain-Bonne-Wimp), nous proposons une généralisation regroupant ces deux procédés à laquelle nous appliquons la procédure Θ.

Le quatrième chapitre montre comment les approximants de Padé donnent naissance à des méthodes itératives de point fixe. Cette idée a déjà été utilisée par Nourein et par Claessens-Loisou-Wuytack, mais leurs procédés n'utilisaient pas toute la table de Padé. C'est dans le but d'explorer toute la table, que nous proposons l'extrapolation de la fonction inverse, qui a déjà été utilisée par Traub.

Dans le cinquième chapitre nous essayons de faire un travail inverse à celui effectué lors des chapitres précédents. En partant des méthodes de King et de Schröder nous construisons deux procédés d'accélération de la convergence d'ordre quatre pour la première et d'ordre trois pour la seconde.

Le sixième et dernier chapitre expose une nouvelle méthode de résolution des systèmes d'équations non linéaires. Après avoir rappelé la méthode d'Henrici nous proposons une méthode basée sur le E-algorithme.

Nous supposerons dans tout ce travail que les application de $\mathbb{R}$ dans lui-même utilisées sont dérivables autant de fois qu'il sera nécessaire pour les démonstrations.

CHAPITRE I

PROCEDES D'EXTRAPOLATION INVERSES

INTRODUCTION

L'objet de ce chapitre est de donner un certain nombre de propriétés des procédés d'extrapolation inverses. Ces propriétés seront exploitées dans le chapitre II concernant l'étude des méthodes de point fixe déduites de ces algorithmes.

Soit $f(x) = 0$, une équation non linéaire, où f est une fonction $f : \mathbb{R} \to \mathbb{R}$. Soit x_* une racine de f. Nous désignons par p_1, p_2, p_3 les propriétés suivantes :

$p_1 : f(x_*) = 0$

$p_2 : f'(x_*) \neq 0$

$p_3 : f^{(m)}(x_*) \neq 0 \ (m \geq 2)$ et $f^{(i)}(x_*) = 0$ pour $1 \leq i \leq m-1$.

La première section est consacrée à l'extrapolation polynomiale inverse, correspondant au cas d'une fonction f satisfaisant p_1 et p_2. Le procédé obtenu noté (B) peut accélérer la convergence d'une suite $(x_n)_{n\in\mathbb{N}}$ pour laquelle on sait trouver f satisfaisant p_1 et p_2. Malheureusement, en général il est difficile d'exhiber une telle fonction. Cet obstacle sera surmonté par l'utilisation de fonctions de plusieurs variables. Nous obtenons une généralisation de l'algorithme (B).

Dans la seconde section nous nous intéressons à l'extrapolation rationnelle inverse, pour laquelle nous reprenons le travail fait ci-dessus.

La troisième section est consacrée au cas des fonctions f satisfaisant p_1 et p_3. Nous étudions en particulier un algorithme d'accélération de la convergence désigné par (G'). Ce procédé (G') entre dans une catégorie d'algorithmes regroupant le procédé d'Overholt et le procédé (B') de Germain-Bonne-Wimp.

Nous terminons par un ensemble d'essais numériques.

I.1. - PROCEDE D'EXTRAPOLATION POLYNOMIALE INVERSE
(Cas des fonctions f vérifiant p_1 et p_2)

I.1.1. - Rappels sur un procédé d'extrapolation polynomiale généralisé [2]

Soit $(y_n)_{n\in\mathbb{N}}$, une suite de nombres réels, convergeant vers y_*.

Soit $(x_n)_{n\in\mathbb{N}}$, une autre suite convergeant vers x_* connu.

Soit ϕ une fonction quelconque.

Notons par $\phi_n = \phi(x_n)$ et $\phi_* = \phi(x_*)$.

On considère le polynôme généralisé :

$$P_{k,\phi}^{(n)}(x) = \sum_{i=0}^{k} a_i^{(n,k)} [\phi(x) - \phi_*]^i .$$

$P_{k,\phi}^{(n)}$ sera déterminé par les conditions d'interpolation :

$$P_{k,\phi}^{(n)}(x_{n+i}) = y_{n+i},\ i = 0, 1, \ldots, k.$$

On s'intéresse à la valeur de $P_{k,\phi}^{(n)}(x_*) = a_0^{(n,k)}$, qui sera notée $T_k^{(n)}$. Ces quantités $T_k^{(n)}$ pour différentes valeurs de n et de k, sont calculées d'une façon récurrente par le schéma de Neville-Aitken :

$$(A) : \begin{cases} T_0^{(n)} = y_n,\ n = 0, 1, 2\ldots \\ \\ T_k^{(n)} = \dfrac{(\phi_{n+k}-\phi_*)\, T_{k-1}^{(n)} - (\phi_n-\phi_*)\, T_{k-1}^{(n+1)}}{\phi_{n+k} - \phi_n}, & n = 0, 1, 2, \ldots \\ & k = 1, 2, 3\ldots \end{cases}$$

On voit qu'une condition nécessaire et suffisante pour l'existence et l'unicité de $T_k^{(n)}$ est que :

$$\phi_i \neq \phi_j \ \forall i,j = n,\dots, n+k,\ i \neq j.$$

Remarque :

Le cas où (x_n) est une suite strictement décroissante à termes positifs convergeant vers 0, et où ϕ est une fonction strictement monotone, continue à droite de 0, et définie sur l'intervalle $[0, T]$, $T > 0$ a été étudié par P.J. Laurent [17].

I.1.2. - Procédé d'extrapolation polynomiale inverse

Soit $f(x) = 0$, une équation non linéaire dans $\mathbb{R}$.

Soit x_* une racine simple de f, c'est-à-dire que f vérifie les propriétés p_1 et p_2 au point x_*.

Dans un voisinage de x_*, f admet un inverse F [24]. Donc on a :

$$F(0) = x_*$$

Soit V_* un voisinage de x_* et W_0 un voisinage de 0, tels que :

$$x \in V_* \Rightarrow f(x) \in W_0$$

$$\text{et } y \in W_0 \Rightarrow F(y) \in V_*.$$

Soit $(x_n)_{n\in\mathbb{N}}$ une suite construite par un procédé itératif ϕ quelconque pour résoudre $f(x) = 0$.

On peut supposer que la suite (x_n) est entièrement dans V_*.

$$x_{n+1} = \phi(x_n),\ n = 0, 1, 2,...$$

$$x_* = \phi(x_*) = \phi_*$$

$$z_n = f(x_n) = f_n$$

$$f_* = f(x_*) = 0$$

La suite (z_n) est entièrement dans W_0 et de limite 0.

Dans le procédé (A) remplaçons y_n par x_n et ϕ par f, on a :

$$(B) : \begin{cases} T_0^{(n)} = x_n,\ n = 0, 1, 2,... \\ \\ T_k^{(n)} = \dfrac{f_{n+k}\, T_{k-1}^{(n)} - f_n\, T_{k-1}^{(n+1)}}{f_{n+k} - f_n},\ \begin{array}{l} n = 0, 1, 2,... \\ k = 1, 2, 3,... \end{array} \end{cases}$$

Ce procédé (B) est à la base de certaines méthodes de résolution des équations non linéaires obtenues par C. Brézinski dans [2], et que nous nous proposons d'étudier en détail dans le chapitre suivant.

I.1.3. - Etude du procédé (B)

Soit $Q_k^{(n)}(z) = \sum_{i=0}^{k} a_i^{(n,k)} z^i$ tel que pour $i = 0, 1,..., k$ on ait :

$$Q_k^{(n)}(z_{n+i}) = F(z_{n+i}),\ \text{où } F \text{ est l'inverse de } f,$$

$$z_{n+i} = f_{n+i},\ i = 0, 1,..., k.$$

L'erreur d'interpolation est donnée par [8] :

$$F(z) - Q_k^{(n)}(z) = \frac{F^{(k+1)}_{(\xi)}}{(k+1)!} \prod_{i=0}^{k} (z - z_{n+i}).$$

où $\xi \in I_z =]m_z, M_z[$ avec $m_z = \min\{z, z_n, \ldots, z_{n+k}\}$

et $M_z = \max\{z, z_n, \ldots, z_{n+k}\}$.

On remarque facilement que :

$$Q_k^{(n)}(f(x)) = P_{k,f}^{(n)}(x),$$

$$\text{et } Q_k^{(n)}(0) = P_{k,f}^{(n)}(x_*) = T_k^{(n)}$$

Par conséquent on a :

$$x_* - T_k^{(n)} = \frac{(-1)^{k+1}}{(k+1)!} F^{(k+1)}_{(\xi)} \prod_{i=0}^{k} f_{n+i}$$

où $\xi \in I_0 =]m_0, M_0[$ avec $m_0 = \min\{0, f_n, \ldots, f_{n+k}\}$

et $M_0 = \max\{0, f_n, \ldots, f_{n+k}\}$.

$f_{n+i} = f(x_{n+i}) = f'(\xi_i)\, e_{n+i}$, où $e_{n+i} = x_{n+i} - x_*$ et $\xi_i \in J_i$

où $J_i =]\min\{x_*, x_{n+i}\}, \max\{x_*, x_{n+i}\}[$.

On peut écrire l'erreur $T_k^{(n)} - x_*$ sous la forme

$$T_k^{(n)} - x_* = \frac{(-1)^k}{(k+1)!} F^{(k+1)}_{(\xi)} \prod_{i=0}^{k} f'(\xi_i) \prod_{i=0}^{k} e_{n+i}$$

où $\xi \in I_0$ et $\xi_i \in J_i$ définis précédemment.

Si $F^{(k+1)}$, la $(k+1)^{\text{ième}}$ dérivée de la réciproque de f est continue dans un voisinage de zéro, et si f', la première dérivée de f est aussi continue dans un voisinage de x_*, alors on a, d'après cette dernière forme de l'erreur, un certain nombre de propriétés concernant l'algorithme (B).

On pose $f'_* = f'(x_*)$.

Propriété 1 :

Pour k *fixé,*

$$T_k^{(n)} - x_* \sim (-1)^k \, (f'_*)^{k+1} \frac{F^{(k+1)}(0)}{(k+1)!} \prod_{i=0}^{k} e_{n+i}, \quad \textit{quand } n \to \infty.$$

Preuve :

On a vu que l'erreur s'écrit sous la forme :

$$T_k^{(n)} - x_* = (-1)^k \frac{F^{(k+1)}(\xi)}{(k+1)!} \prod_{i=0}^{k} f'(\xi_i) \prod_{i=0}^{k} e_{n+i}, \quad \text{où } \xi \in I_0$$

et $\xi_i \in J_i$.

Quand n tend vers l'infini ξ tend vers zéro et ξ_i tend vers x_*.

Donc, pour k fixé :

$$\lim_{n\to\infty} \frac{T_k^{(n)} - x_*}{\prod_{i=0}^{k} e_{n+i}} = (-1)^k \frac{F^{(k+1)}(0)}{(k+1)!} (f'_*)^{k+1}. \ \square$$

Soit $F^{(j)}$, la $j^{\text{ième}}$ dérivée de $F = f^{-1}$. On peut alors exprimer $F^{(j)}(0)$ par les différentes dérivées de f au point x_* [24].

$$F^{(j)}(0) = f_*^{\prime\,-j} \sum_{\ell=1}^{K_j} (-1)^{r_\ell} (j+r_\ell-1)! \prod_{i=2}^{j} \frac{A_i^{\beta_{i,\ell}}}{\beta_{i,\ell}!}$$

où $A_i = \dfrac{f_*^{(i)}}{i!f_*^{\prime}}$, $f_*^{(i)} = f^{(i)}(x_*)$ et où

K_j, r_ℓ, $\beta_{i,\ell}$ sont déterminés comme ce qui suit :

On résoud dans $\mathbb{N}^{j-1}$ l'équation

$$\sum_{i=2}^{j} (i-1)\ \beta_{i,\ell} = j-1$$

Cela donne K_j $(j-1)$-uplets solutions. On les indice de 1 à K_j par l'indice ℓ. Les r_ℓ sont calculés par la relation

$$r_\ell = \sum_{i=2}^{j} \beta_{i,\ell} \quad (\ell = 1,\ldots, K_j).$$

Pour $j = 1$, nous faisons la convention : $\beta_{i,\ell} = 0$, $\forall i$ et $\forall \ell$.

Exemple :

a) $j = 1$: $F'(0) = f_*^{\prime\,-1}$

b) $j = 2$: La dérivation classique donne :

$$F''(0) = f'^{-2}(-2!\ A_2).$$

Par la relation précédente nous avons :

$$F^{(2)}(0) = f_*^{\prime\,-2} \sum_{\ell=1}^{K_2} (-1)^{r_\ell} (1+r_\ell)! \prod_{i=2}^{2} \frac{A_i^{\beta_{i,\ell}}}{\beta_{i,\ell}!}$$

$$= f_*^{\prime\,-2} \sum_{\ell=1}^{K_2} (-1)^{r_\ell} (1+r_\ell)! \frac{A_2^{\beta_{2,\ell}}}{\beta_{2,\ell}!}$$

$\beta_{2,\ell}$ est solution de : $\beta_{2,\ell} = 1$. Donc on a un seul $\beta_{2,\ell}$. Par conséquent $K_2 = 1$ et $\ell = 1$. D'où

$$F^{(2)}(0) = f_*'^{-2}(-2!\ A_2).$$

c) j = 3 : La dérivation classique donne :

$$F^{(3)}(0) = (f_*')^{-3}\ (-3!\ A_3 + 4!\ \frac{A_2^2}{2!}).$$

Par la relation précédente nous obtenons

$$F^{(3)}(0) = (f')^{-3} \sum_{\ell=1}^{K_3} (-1)^{r_\ell}\ (2+r_\ell)!\ \prod_{i=2}^{3} \frac{A_i^{\beta_{i,\ell}}}{\beta_{i,\ell}!}$$

Déterminons r_ℓ, K_3, $\beta_{i,\ell}$

Les $\beta_{i,\ell}$ sont solutions de

$$\begin{cases} \beta_{2,\ell} + 2\ \beta_{3,\ell} = 2 \\ (\beta_{2,\ell}, \beta_{3,\ell}) \in \mathbb{N}^2 \end{cases}$$

$$\Rightarrow \left.\begin{array}{r} \beta_{2,\ell} = 2(1-\beta_{3,\ell}) \geq 0 \Rightarrow \beta_{3,\ell} \leq 1 \\ \beta_{3,\ell} \in \mathbb{N} \end{array}\right\} \Rightarrow \beta_{3,\ell} \in \{0, 1\}.$$

$$\left.\begin{array}{r} \beta_{3,\ell} = 0 \Rightarrow \beta_{2,\ell} = 2 \\ \beta_{3,\ell} = 1 \Rightarrow \beta_{2,\ell} = 0 \end{array}\right\} \Rightarrow (\beta_{2,\ell}, \beta_{3,\ell}) \in \{(0, 1), (2, 0)\}$$

Donc $K_3 = 2$ et $\ell \in \{1, 2\}$. Prenons $(\beta_{2,1}, \beta_{3,1}) = (0, 1)$ et $(\beta_{2,2}, \beta_{3,2}) = (2, 0)$.

On a $r_1 = \beta_{2,1} + \beta_{3,1} = 1$

et $r_2 = \beta_{2,2} + \beta_{3,2} = 2$.

$$F^{(3)}(0) = (f'_*)^{-3} \left((-1)^{r_1} (2+r_1)! \prod_{i=2}^{3} \frac{A_i^{\beta_{i,1}}}{\beta_{i,1}!} + (-1)^{r_2} (2+r_2)! \prod_{i=2}^{3} \frac{A_i^{\beta_{i,2}}}{\beta_{i,2}!}\right)$$

$$= (f'_*)^{-3}\left(-3!\ A_3 + 4!\ \frac{A_2^2}{2!}\right)$$

Nous obtenons donc le même résultat.

d) $j = 4$: Par dérivation classique on a

$$F^{(4)}(0) = (f'_*)^{-4} \left(-4!\ A_4 + 5!\ A_2A_3 - 6!\ \frac{A_2^3}{3!}\right).$$

La relation précédente donne

$$F^{(4)}(0) = (f'_*)^{-4} \sum_{\ell=1}^{K_4} (-1)^{r_\ell} (3+r_\ell)! \prod_{i=2}^{4} \frac{A_i^{\beta_{i,\ell}}}{\beta_{i,\ell}!}$$

Déterminons r_ℓ, K_ℓ, $\beta_{i,\ell}$.

Les $\beta_{i,\ell}$ sont solutions de

$$\begin{cases} \beta_{2,\ell} + 2\ \beta_{3,\ell} + 3\ \beta_{4,\ell} = 3 \\ (\beta_{2,\ell},\ \beta_{3,\ell},\ \beta_{4,\ell}) \in \mathbb{N}^3 \end{cases}$$

$$\left.\begin{array}{r} \Rightarrow \beta_{2,\ell} + 2\ \beta_{3,\ell} = 3(1-\beta_{4,\ell}) \Rightarrow \beta_{4,\ell} \leq 1 \\ \beta_{4,\ell} \in \mathbb{N} \end{array}\right\} \Rightarrow \beta_{4,\ell} \in \{0,\ 1\}$$

$$\left.\begin{array}{r} \beta_{4,\ell} = 0 \Rightarrow \beta_{2,\ell} + 2\ \beta_{3,\ell} = 3 \Rightarrow 2\ \beta_{3,\ell} \leq 3 \\ \beta_{3,\ell} \in \mathbb{N} \end{array}\right\} \Rightarrow \beta_{3,\ell} \leq 1 \Rightarrow \beta_{3,\ell} \in \{0,\ 1\}$$

$$\beta_{3,\ell} = 0 \Rightarrow \beta_{2,\ell} = 3$$
$$\beta_{3,\ell} = 1 \Rightarrow \beta_{2,\ell} = 1$$

Donc les triplets (1, 1, 0) et (3, 0, 0) sont solutions.

$$\left.\begin{array}{r}\beta_{4,\ell} = 1 \Rightarrow \beta_{2,\ell} + 2\,\beta_{3,\ell} = 0 \\ (\beta_{2,\ell}, \beta_{3,\ell}) \in \mathbb{N}^2\end{array}\right\} \Rightarrow \beta_{2,\ell} = \beta_{3,\ell} = 0.$$

d'où la solution (0, 0, 1).

Par conséquent $(\beta_{2,\ell}, \beta_{3,\ell}, \beta_{4,\ell}) \in \{(1, 1, 0), (3, 0, 0), (0, 0, 1)\}$.

Ainsi K_3 est égal à trois et ℓ appartient à $\{1, 2, 3\}$.

Prenons

$$\begin{aligned} (\beta_{2,1}, \beta_{3,1}, \beta_{4,1}) &= (1, 1, 0) \\ (\beta_{2,2}, \beta_{3,2}, \beta_{4,2}) &= (3, 0, 0) \\ \text{et } (\beta_{2,3}, \beta_{3,3}, \beta_{4,3}) &= (0, 0, 1). \end{aligned}$$

Les r_ℓ sont données par

$$\begin{aligned} r_1 &= \beta_{2,1} + \beta_{3,1} + \beta_{4,1} = 2 \\ r_2 &= \beta_{2,2} + \beta_{3,2} + \beta_{4,2} = 3 \\ r_3 &= \beta_{2,3} + \beta_{3,3} + \beta_{4,3} = 1. \end{aligned}$$

D'où l'on tire

$$F^{(4)}(0) = (f'_*)^{-4}\left(5!\, A_2A_3 - 6!\, \frac{A_2^3}{3!} - 4!\, A_4\right)$$

Propriété 2 :

Pour k *fixé* :

$$T_k^{(n)} - x_* \sim \frac{(-1)^k}{(k+1)!}\left(\sum_{\ell=1}^{K_{k+1}} (-1)^{r_\ell}(k+r_\ell)! \prod_{i=2}^{k+1} \frac{A_i^{\beta_{i,\ell}}}{\beta_{i,\ell}!}\right) \prod_{i=0}^{k} e_{n+i}.$$

quand $n \to \infty$.

Preuve :

Il suffit de remplacer $F^{(k+1)}(0)$ par son expression en fonction des dérivées de f dans la propriété 1. □

Cette propriété nous sera très utile dans la suite pour déterminer le coefficient asymptotique d'erreur de certaines méthodes de résolution des équations non linéaires.

Propriété 3 :

Si $|F^{(k+1)}(y)| \leq M$ (M *indépendant de* k) ; $\forall y \in W_0$, $\forall k$, *où* $M > 0$ *et où* W_0 *est un voisinage de zéro, alors* :

$T_k^{(n)} \to x_*$ *quand* $k \to \infty$, *pour tout* n

et $T_k^{(n)} \to x_*$ *quand* $n \to \infty$, *pour tout* k.

Preuve :

$$|T_k^{(n)}-x_*| = \frac{|F^{(k+1)}(\xi)|}{(k+1)!} \prod_{i=0}^{k} |f_{n+i}| \leq \frac{M}{(k+1)!} \prod_{i=0}^{k} |f_{n+i}| \text{ si } \xi \in W_0$$

Soit $\varepsilon > 0$, $\exists N_\varepsilon$; $\ell \geq N_\varepsilon \Rightarrow |f_\ell| \leq \varepsilon$, car f est supposée continue en x_* et (x_ℓ) converge vers x_*.

Soit $n \geq N_\varepsilon \Rightarrow n+i \geq N_\varepsilon$ pour $i = 0, 1, \ldots, k$. k étant quelconque.

Donc $|f_{n+i}| \leq \varepsilon$, $\forall i = 0, 1, \ldots, k$

Par conséquent l'erreur peut être majorée par :

$$|T_k^{(n)}-x_*| \leq \frac{M}{(k+1)!} \varepsilon^{(k+1)}.$$

Ce qui signifie que $(T_k^{(n)})_{n\in\mathbb{N}}$ converge vers x_* pour tout k.

Maintenant soit $k \geq N_\varepsilon \Rightarrow n+k \geq N_\varepsilon \Rightarrow |f_{n+k}| \leq \varepsilon$

Si $n \geq N_\varepsilon$ alors $n+i \geq N_\varepsilon$ et on a :

$$|T_k^{(n)}-x_*| \leq \frac{M\,\varepsilon^{(k+1)}}{(k+1)!} \to 0 \text{ quand } k \to \infty,\ \forall n \geq N_\varepsilon.$$

Si $n < N_\varepsilon$, alors $\exists N$ tel que $n + N = N_\varepsilon$, l'erreur dans ce cas peut être majorée par

$$|T_k^{(n)}-x^*| \leq \frac{M}{(k+1)!}\Big(\prod_{i=0}^{N} |f_{n+i}|\Big) \prod_{i=N+1}^{k} |f_{n+i}|$$

$$\leq \frac{M}{(k+1)!}\Big(\prod_{i=0}^{N} |f_{n+i}|\Big)\ \varepsilon^{(k-N)} \to 0 \text{ quand } k \to \infty.$$

D'où ; $T_k^{(n)}$ converge vers x_* quand $k \to \infty$ et ceci $\forall n$. □

<u>*Propriété 4*</u> :

Pour k *fixé,*

$$\frac{T_k^{(n+1)} - x_*}{T_k^{(n)} - x_*} \sim (\phi'_*)^{k+1},\ \textit{quand}\ n \to \infty,$$

où ϕ *est la fonction itérative telle que :* $x_{n+1} = \phi(x_n)$, $\forall n$ *et où* $\phi'_* = \phi'(x_*)$.

<u>Preuve</u> :

$$\frac{T_k^{(n+1)} - x_*}{T_k^{(n)} - x_*} = \frac{F^{(k+1)}(\xi)\prod_{i=0}^{k} f_{n+1+i}}{F^{(k+1)}(\eta)\prod_{i=0}^{k} f_{n+i}} = \frac{F^{(k+1)}(\xi)}{F^{(k+1)}(\eta)}\prod_{i=0}^{k}\frac{f_{n+1+i}}{f_{n+i}}$$

Or $\dfrac{f_{i+1}}{f_i} = \dfrac{f'_*\,e_{i+1}+O(e_{i+1}^2)}{f'_*\,e_i+O(e_i^2)}$, et $e_{i+1} = \phi'_*\,e_i + O(e_i^2)$.

D'où l'on tire $\frac{f_{i+1}}{f_i} = \phi'_* + O(e_i)$

Et comme ξ et η tendent vers zéro, quand n tend vers l'infini, il s'ensuit que :

$$\lim_{n\to\infty} \frac{T_k^{(n+1)}-x_*}{T_k^{(n)}-x_*} = \frac{F^{(k+1)}(0)}{F^{(k+1)}(0)} (\phi'_*)^{k+1} = (\phi'_*)^{k+1}. \square$$

<u>*Propriété 5*</u> :

> *Pour* k *fixé,*
>
> $$\frac{T_{k+1}^{(n)}-x_*}{T_k^{(n)}-x_*} \sim - \frac{f'_*}{(k+2)} \frac{F^{(k+2)}(0)}{F^{(k+1)}(0)} e_{n+k+1}, \text{ quand } n \to \infty.$$
>
> *De plus si* $\phi'_* \neq 0$,
>
> $$\frac{T_{k+1}^{(n)}-x_*}{T_k^{(n+1)}-x_*} \sim - \frac{f'_*}{(k+2)} \frac{1}{(\phi'_*)^{k+1}} \frac{F^{(k+2)}(0)}{F^{(k+1)}(0)} e_{n+k+1}, \text{ quand } n \to \infty.$$

<u>Preuve</u> :

$$\frac{T_{k+1}^{(n)}-x_*}{T_k^{(n)}-x_*} = - \frac{1}{(k+2)} \frac{F^{(k+2)}(\xi) \prod_{i=0}^{k+1} f'(\xi_i) \prod_{i=0}^{k+1} e_{n+i}}{F^{(k+1)}(\eta) \prod_{i=0}^{k} f'(\eta_i) \prod_{i=0}^{k} e_{n+i}}$$

$$= - \frac{1}{(k+2)} \frac{F^{(k+2)}(\xi)}{F^{(k+1)}(\eta)} \prod_{i=0}^{k} \frac{f'(\xi_i)}{f'(\eta_i)} f'(\xi_{k+1}) e_{n+k+1}.$$

où $\xi \in \,]\min\{0, f_n,\dots, f_{n+k+1}\}, \max\{0, f_n,\dots, f_{n+k+1}\}[$

$\eta \in \,]\min\{0, f_n,\dots, f_{n+k}\}, \quad \max\{0, f_n,\dots, f_{n+k}\}[$

$\xi_i \in \,]\min\{x_*, x_{n+i}\}, \quad \max\{x_*, x_{n+i}\}[$ pour $i = 0, 1,\dots, k+1$

et $\eta_i \in \,]\min\{x_*, x_{n+i}\}, \quad \max\{x_*, x_{n+i}\}[$ pour $i = 0, 1,\dots, k$.

$$\Longrightarrow \left\{\frac{T_{k+1}^{(n)}-x_*}{T_k^{(n)}-x_*}\right\} / e_{n+k+1} \to \frac{-f'_*}{(k+2)} \frac{F^{(k+2)}(0)}{F^{(k+1)}(0)}, \text{ quand } n \to \infty.$$

Et d'après la propriété 4 on a aussi

$$\left\{\frac{T_{k+1}^{(n)} - x_*}{T_k^{(n+1)}-x_*}\right\}/e_{n+k+1} = \left\{\frac{T_{k+1}^{(n)}-x_*}{T_k^{(n)}-x_*} \cdot \frac{T_k^{(n)} - x_*}{T_k^{(n+1)}-x_*}\right\}/e_{n+k+1} \to \frac{-f'_*}{(k+2)} \frac{1}{(\phi'_*)^{k+1}} \frac{F^{(k+2)}(0)}{F^{(k+1)}(0)},$$

quand n tend vers l'infini. □

Désignons par k l'indice des colonnes et par n l'indice des diagonales descendantes. Les quantités $T_k^{(n)}$ peuvent être rangées dans un tableau à double entrée.

$$\begin{array}{cccc}
T_0^{(0)} & & & \\
 & T_1^{(0)} & & \\
T_0^{(1)} & & T_2^{(0)} & \\
 & T_1^{(1)} & & T_3^{(0)} \\
T_0^{(2)} & & T_2^{(1)} & \vdots \\
 & T_1^{(2)} & \vdots & \\
T_0^{(3)} & \vdots & & \\
\vdots & & &
\end{array}$$

La propriété 5 montre bien que chaque colonne converge plus vite que la colonne précédente.

Remarques :

a) Si f et ϕ sont reliées par la relation :

$$\phi(x) = x + \lambda f(x), \lambda \neq 0, \lambda \in \mathbb{R}$$

alors on a : $f_n = \frac{1}{\lambda} \Delta x_n$, et le procédé (B) se réduit à :

$$(B') : \begin{cases} T_0^{(n)} = x_n, \; n = 0, 1, 2,\ldots \\ \\ T_k^{(n)} = \dfrac{\Delta x_{n+k} \, T_{k-1}^{(n)} - \Delta x_n \, T_{k-1}^{(n+1)}}{\Delta x_{n+k} - \Delta x_n}, \; n = 0, 1, 2,\ldots \\ \qquad\qquad\qquad\qquad\qquad\qquad k = 1, 2, 3,\ldots \end{cases}$$

C'est un procédé d'interpolation inverse obtenu et étudié par B. Germain-Bonne [11] et par J. Wimp [26].

b) Le procédé (B) peut servir pour accélérer la convergence d'une suite (x_n) de limite x_*, si l'on sait trouver une fonction f telle que :

$$P_1 : f(x_*) = 0$$

$$P_2 : f'(x_*) \neq 0.$$

Il est en général difficile de se trouver une telle fonction ; on peut remédier à cet inconvénient en utilisant des fonctions de plusieurs variables. Cette méthode sera exposée dans le paragraphe suivant.

I.1.4. - Une généralisation du procédé (B)

Soit $G : \mathbb{R}^{p+1} \to \mathbb{R}$
$x = (x^0, x^1,\ldots, x^p) \to G(x)$ tels que :

1) $G(t, t,\ldots, t) = 0 \; \forall t \in \mathbb{R}$

2) $G(x^0, x^1,\ldots, x^p) \neq 0 \; \forall x = (x^0, x^1,\ldots, x^p)$ tels que $\forall i \neq j, x^i \neq x^j$.

Ce genre de fonctions G a été utilisé par Germain-Bonne pour accélérer la convergence des suites [11].

Soit $(x_n)_{n\in\mathbb{N}}$ une suite générée par une fonction itérative ϕ.

$$x_{n+1} = \phi(x_n),\ n = 0,\ 1,\ 2,\ldots$$

$$x_* = \phi(x_*) \text{ où } x_* = \lim_{n\to\infty} x_n.$$

On considère la fonction :

$$\begin{aligned} C &: \mathbb{R} \to \mathbb{R} \\ &x \to C(x) = G(x,\ \phi(x),\ \phi^2(x),\ldots,\ \phi^p(x)) \end{aligned}$$

où $\phi^i(x) = \phi(\phi^{i-1}(x))$; $i = 1,\ 2,\ldots,\ p$

et $\phi^0(x) = x$.

Soit à résoudre l'équation :

$$C(x) = 0.$$

$$\begin{aligned} C_n = C(x_n) &= G(x_n,\ \phi(x_n),\ldots,\ \phi^{(p)}(x_n)) \\ &= G(x_n,\ x_{n+1},\ldots,\ x_{n+p}) \end{aligned}$$

Il est aisé de calculer C_n, même si ϕ n'est pas connue explicitement.

Par application de l'algorithme (B) à cette équation, nous obtenons l'algorithme suivant :

$$(C) : \begin{cases} T_0^{(n)} = x_n,\ n = 0,\ 1,\ 2,\ldots \\ \\ T_k^{(n)} = \dfrac{G(x_{n+k},\ldots,\ x_{n+k+p})\ T_{k-1}^{(n)} - G(x_n,\ldots,\ x_{n+p})\ T_{k-1}^{(n+1)}}{G(x_{n+k},\ldots,\ x_{n+k+p}) - G(x_n,\ldots,\ x_{n+p})},\ \begin{matrix} n = 0,\ 1,\ 2,\ldots \\ k = 1,\ 2,\ 3,\ldots \end{matrix} \end{cases}$$

Remarque :

La suite colonne $(T_1^{(n)})$ correspondant au cas p = 1 a été obtenue et étudiée par Germain-Bonne [11].

Soit $C(x) = G(x, \phi(x), \ldots, \phi^p(x)) = 0$.

Soit $x_n \to x_*$ tels que $x_{n+1} = \phi(x_n)$, $\forall n$ et $x_* = \phi(x_*)$.

Si C admet x_* comme racine simple et si C^{-1} l'inverse de C est de classe C^{k+1} alors on a :

Propriété 6 :

Pour k *fixé,*

$$T_k^{(n)} - x_* \sim (-1)^k \frac{(C^{-1})^{(k+1)}_{(0)}}{(k+1)!} (C'_*)^{k+1} \prod_{i=0}^{k} e_{n+i}, \text{ quand } n \to \infty.$$

Preuve :

Elle est identique à celle faite pour la propriété 1. □

Propriété 7 :

Pour k *fixé,*

$$\frac{T_{k+1}^{(n)} - x_*}{T_k^{(n)} - x_*} \sim - \frac{C'_*}{(k+2)} \frac{(C^{-1})^{(k+2)}(0)}{(C^{-1})^{(k+1)}(0)} e_{n+k+1}, \text{ quand } n \to \infty$$

Preuve :

Elle est identique à celle faite pour la propriété 5. □

D'après cette dernière propriété, ce procédé est capable d'accélérer la convergence des suites itératives, et chaque colonne converge plus vite que la colonne précédente.

Soit :

$$G_i'(x) = \frac{\partial G}{\partial x^i}(x) = \frac{\partial G}{\partial x^i}(x^0, x^1, \ldots, x^p) ;$$

la dérivée partielle de G par rapport à la variable x^i.

On a :

Propriété 8 :

Si (x_n) *est une suite à convergence linéaire, c'est-à-dire :*

$$\lim_{n\to\infty} \frac{e_{n+1}}{e_n} = \rho \text{ avec } \rho \neq 1, 0 \text{ et si } \sum_{i=0}^{p} G_i'(x_*, \ldots, x_*)\, \rho^i \neq 0$$

alors : $\lim_{n\to\infty} \frac{T_1^{(n)} - x_*}{x_n - x_*} = 0.$

Preuve :

La règle de l'algorithme (C) nous permet d'écrire :

$$T_1^{(n)} = x_n - \frac{x_{n+1} - x_n}{\frac{G(x_{n+1}, \ldots, x_{n+p+1})}{G(x_n, \ldots, x_{n+p})} - 1}$$

$$\Rightarrow T_1^{(n)} - x_* = x_n - x_* - \frac{x_{n+1} - x_n}{\frac{G(x_{n+1}, \ldots, x_{n+p+1})}{G(x_n, \ldots, x_{n+p})} - 1}$$

$$\Rightarrow \frac{T_1^{(n)} - x_*}{x_n - x_*} = 1 - \frac{\frac{x_{n+1} - x_*}{x_n - x_*} - 1}{\frac{G(x_{n+1}, \ldots, x_{n+p+1})}{G(x_n, \ldots, x_{n+p})} - 1}$$

$$G(x_n,\ldots,x_{n+p}) = \sum_{i=0}^{p} G_i'(\xi_i)e_{n+i},\ \xi_i = (x_*,\ldots,x_*,\ \theta_i\, x_{n+i} + (1-\theta_i)\, x_*,\ x_{n+i+1},\ldots,x_{n+p}) \text{ et } 0 < \theta_i < 1.$$

$$G(x_{n+1},\ldots,x_{n+p+1}) = \sum_{i=0}^{p} G_i'(\eta_i)e_{n+1+i},\ \eta_i = (x_*,\ldots,x_*,\ \delta_i\, x_{n+1+i} + (1-\delta_i)\, x_*,\ x_{n+i+2},\ldots,x_{n+p+1}) \text{ et } 0 < \delta_i < 1.$$

Donc on a :

$$\frac{G(x_{n+1},\ldots,x_{n+p+1})}{G(x_n,\ldots,x_{n+p})} = \frac{e_{n+1}}{e_n}\ \frac{G_0'(\eta_0)+G_1'(\eta_1)\dfrac{e_{n+2}}{e_{n+1}}+\ldots+G_p'(\eta_p)\dfrac{e_{n+1+p}}{e_{n+1}}}{G_0'(\xi_0)+G_1'(\xi_0)\dfrac{e_{n+1}}{e_n}+\ldots+G_p'(\xi_p)\dfrac{e_{n+p}}{e_n}}.$$

$$\Longrightarrow \frac{G(x_{n+1},\ldots,x_{n+p+1})}{G(x_n,\ldots,x_{n+p})} \to \rho\ \frac{\sum_{i=0}^{p} G_i'(x_*,\ldots,x_*)\ \rho^i}{\sum_{i=0}^{p} G_i'(x_*,\ldots,x_*)\ \rho^i} = \rho, \text{ quand } n \to \infty.$$

Par conséquent nous obtenons :

$$\frac{T_1^{(n)}-x_*}{e_n} \to 1 - \frac{\rho-1}{\rho-1} = 1 - 1 = 0.\ \square$$

<u>Remarque</u> :

Si x_* est une racine simple de $C(x) = G(x, \phi(x),\ldots, \phi^p(x)) = 0$, alors C^{-1}, l'inverse de C existe dans un voisinage de W_0.

Si de plus G et ϕ sont de classe C^{k+1}, alors C^{-1} est de classe C^{k+1} dans W_0, et d'après la propriété 3, chaque suite colonne $(T_k^{(n)})_{n\in\mathbb{N}}$ est une suite à convergence linéaire tel que :

$$\lim_{n\to\infty} \frac{T_k^{(n+1)}-x_*}{T_k^{(n)}-x_*} = \rho^{k+1}.$$

où $\rho = \phi'_*$.

Propriété 9 :

Si $\dfrac{T_{k-1}^{(n+1)}-x_*}{T_{k-1}^{(n)}-x_*} \to \rho^k$ *avec* $\rho \neq 0$ *et si* $\sum_{i=0}^{p} \rho^i\, G_i'(x_*,\ldots,\, x_*) \neq 0$, *alors*

$$\frac{T_k^{(n)}-x_*}{T_{k-1}^{(n)}-x_*} \to 0, \textit{ quand } n \to \infty \textit{ et } \frac{T_k^{(n)}-x_*}{T_{k-1}^{(n+1)}-x_*} \to 0, \textit{ quand } n \to \infty.$$

Preuve :

$$T_k^{(n)}-x_* = T_{k-1}^{(n)}-x_* - \frac{T_{k-1}^{(n+1)} - T_{k-1}^{(n)}}{\dfrac{G(x_{n+k},\ldots,\, x_{n+k+p})}{G(x_n,\ldots,\, x_{n+p})} - 1}$$

$$\Rightarrow \frac{T_k^{(n)}-x_*}{T_{k-1}^{(n)}-x_*} = \frac{1-\dfrac{T_{k-1}^{(n+1)} - x_*}{T_{k-1}^{(n)} - x_*}}{\dfrac{G(x_{n+k},\ldots,\, x_{n+k+p})}{G(x_n,\ldots,\, x_{n+p})} - 1} + 1$$

$$\frac{G(x_{n+k},\ldots,\, x_{n+k+p})}{G(x_n,\ldots,\, x_{n+p})} = \frac{\sum_{i=0}^{p} G_i'(\xi_i)\, e_{n+k+i}}{\sum_{i=0}^{p} G_i'(\eta_i)\, e_{n+i}} = \frac{e_{n+k}}{e_n}\, \frac{\sum_{i=0}^{p} G_i'(\xi_i)\, \dfrac{e_{n+k+i}}{e_{n+k}}}{\sum_{i=0}^{p} G_i'(\eta_i)\, \dfrac{e_{n+i}}{e_n}}.$$

$$\to \rho^k\, \frac{\sum_{i=0}^{p} G_i'(x_*,\ldots,\, x_*)\, \rho^i}{\sum_{i=0}^{p} G_i'(x_*,\ldots,\, x_*)\, \rho^i} = \rho^k, \text{ quand } n \to \infty.$$

En utilisant la condition de la propriété, on conclut que le rapport $(T_k^{(n)}-x_*)/(T_{k-1}^{(n)}-x_*)$ tend vers zéro, quand n tend vers l'infini.

Quant à $(T_k^{(n)}-x_*)/(T_{k-1}^{(n+1)}-x_*)$, il s'écrit

$$\frac{T_k^{(n)}-x_*}{T_{k-1}^{(n+1)}-x_*} = \frac{T_k^{(n)}-x_*}{T_{k-1}^{(n)}-x_*} \quad \frac{T_{k-1}^{(n)}-x_*}{T_{k-1}^{(n+1)}-x_*}.$$ D'après la remarque précédente

on a la conclusion. □

Exemple de fonctions G

Donnons quelques fonctions G qui vérifient les propriétés précédentes.

1) Soit $G : \mathbb{R}^2 \to \mathbb{R}$

$$(x, y) \to G(x, y) = y - x.$$
$$G(x, y) = 0 \Longleftrightarrow y = x.$$

Le procédé (C) se réduit au procédé (B').

2) Soit $H : \mathbb{R}^2 \to \mathbb{R}$ et $G : \mathbb{R}^2 \to \mathbb{R}$ telle que :
$G(x, y) = (y-x)\, H(x, y)$ où $H(x, y) \neq 0$ pour $x = y$.

Par exemple : $H(x, y) = e^{y-x}$.

3) $G : \mathbb{R}^3 \to \mathbb{R}$

$$(x, y, z) \to G(x, y, z) = |z-2y+x| + |Z-y|^2 + |y-x|^2$$
$$\Rightarrow G(x_n, x_{n+1}, x_{n+2}) = |\Delta^2 x_n| + |\Delta x_{n+1}|^2 + |\Delta x_n|^2.$$

4) $G : \mathbb{R}^{p+1} \to \mathbb{R}$

$$(x_0, x_1, \ldots, x_p) \to G(x_0, x_1, \ldots, x_p) = |\Delta^p x_0| + \sum_{i=0}^{p-1} (\Delta x_i)^2$$

$$\Rightarrow G(x_n, \ldots, x_{n+p}) = |\Delta^p x_n| + \sum_{i=0}^{p-1} (\Delta x_{n+i})^2.$$

Dans le paragraphe suivant nous appliquons les résultats obtenus précédemment à une classe de suites particulières données par :

$$x_n = f(n),\ \forall n.$$

I.1.5. - Suites de la forme $x_n = f(n)$

Soit (x_n) une suite de la forme :

$$x_n = f(n)\ \forall n$$
$$\ell im\ x_n = x_* = f(\infty).$$

Supposons que f est bijective, on a :

$$n = f^{-1}(x_n)$$
$$\infty = \ell im\ f^{-1}(x_n) = f^{-1}(x_*).$$

Donc on peut prendre x_* comme un pôle simple de f^{-1}.

Supposons que l'on puisse écrire

$$f^{-1}(x) = \frac{h(x)}{x-x_*},\ \text{où } h(x_*) \neq 0.$$

Soit g la fonction définie par :

$$g(x) = \frac{x-x_*}{h(x)} = \frac{1}{f^{-1}(x)}.$$

On a d'une part :

$$p_1 : g(x_*) = 0,$$

et d'autre part :

$$p_2 : g'(x_*) = \left.\frac{h(x)-h'(x)(x-x_*)}{h(x)^{\ 2}}\right|_{x_*} = \frac{1}{h(x_*)} \neq 0.$$

Par suite, on peut appliquer à g l'algorithme (B). Et comme

$g(x_n) = \frac{1}{f^{-1}(x_n)} = \frac{1}{n}$, alors l'algorithme (B) se réduit à :

$$(B'') : \begin{cases} T_0^{(n)} = f(n) \\ \\ T_k^{(n)} = T_{k-1}^{(n+1)} - \frac{n}{k}\,(T_{k-1}^{(n)} - T_{k-1}^{(n+1)}), \quad n = 0, 1, 2,\ldots \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad k = 1, 2, 3,\ldots \end{cases}$$

Si g^{-1}, l'inverse de g est de classe C^{k+1} dans un voisinage de zéro, on a :

$$T_k^{(n)} - x_* = \frac{(-1)^k}{(k+1)!} \frac{d^{k+1}}{d\tau^{k+1}} g^{-1}(\tau) \prod_{i=0}^{k} \frac{1}{n+i}$$

avec $\tau \in \left]\min_{i=0}^{k}\{0, \frac{1}{n+i}\}, \max_{i=0}^{k}\{0, \frac{1}{n+i}\}\right[= \left]0, \frac{1}{n}\right[$

Nous traitons un exemple numérique à la fin de ce chapitre qui montre que l'algorithme (B'') est intéressant pour de telles suites.

I.2. - PROCEDE D'EXTRAPOLATION RATIONNELLE INVERSE
(Cas des fonctions f vérifiant p_1 et p_2)

I.2.1. - Rappels sur un procédé d'extrapolation rationnelle généralisé [2]

Soit $(y_n)_{n\in\mathbb{N}}$, une suite de nombres réels et de limite y_*.

Soit $(x_n)_{n\in\mathbb{N}}$, une suite auxiliaire convergeant vers x_* connu.

Soit ϕ une fonction quelconque.

Notons par $\phi_n = \phi(x_n)$ et $\phi_* = \phi(x_*)$.

On considère la fraction rationnelle généralisée suivante :

$$R_{k,\phi}^{(n)}(x) = \frac{P_{k,\phi}^{(n)}(x)}{Q_{k,\phi}^{(n)}(x)}$$

où $P_{k,\phi}^{(n)}$ et $Q_{k,\phi}^{(n)}$ sont les polynomes généralisés suivants :

$$P_{k,\phi}^{(n)}(x) = \sum_{i=0}^{k} a_{k-i}^{(n,k)}[\phi(x)-\phi_*]^i$$

$$Q_{k,\phi}^{(n)}(x) = \sum_{i=0}^{k} b_{k-i}^{(n,k)}[\phi(x)-\phi_*]^i \text{ avec } b_k^{(n,k)} = 1.$$

Cette fraction $R_{k,\phi}^{(n)}(x)$ sera déterminée par les conditions d'interpolation suivantes :

$$R_{k,\phi}^{(n)}(x_{n+i}) = y_{n+i} \text{ pour } i = 0, 1, 2, \ldots, 2k.$$

Posons $X = \frac{1}{\phi(x)-\phi_*}$. Donc quand x tend vers x_*, X tend vers l'infini.

Ainsi on a :

$$R_{k,\phi}^{(n)}(x_*) = a_k^{(n,k)} = \lim_{X\to\infty} \frac{a_0^{(n,k)} + a_1^{(n,k)} X + \ldots + a_k^{(n,k)} X^k}{b_0^{(n,k)} + b_1^{(n,k)} X + \ldots + b_k^{(n,k)} X^k},\ b_k^{(n,k)} = 1.$$

Si la quantité $a_k^{(n,k)}$ existe, elle sera notée par la suite

$$\rho_{2k}^{(n)} = a_k^{(n,k)}.$$

La théorie des différences réciproques d'une fonction nous permet de calculer les quantités $\rho_{2k}^{(n)}$ d'une façon récurrente :

$$\rho_{-1}^{(n)} = 0,\ \rho_0^{(n)} = y_n,\ n = 0, 1, 2, \ldots$$

$$\rho_{k+1}^{(n)} = \rho_{k-1}^{(n+1)} + \frac{X_{n+k+1} - X_n}{\rho_k^{(n+1)} - \rho_k^{(n)}},\quad \begin{array}{l} n = 0, 1, 2, \ldots \\ k = 0, 1, 2, \ldots \end{array}$$

où $X_n = \dfrac{1}{\phi_n - \phi_*}$ et $X_{n+k+1} = \dfrac{1}{\phi_{n+k+1} - \phi_*}$.

Ce procédé est connu aussi sous le nom de **ρ-algorithme.**

D'où l'algorithme noté (D) :

$$(D) : \begin{cases} \rho_{-1}^{(n)} = 0, \ \rho_0^{(n)} = y_n \\[2ex] \rho_{k+1}^{(n)} = \rho_{k-1}^{(n+1)} + \dfrac{\dfrac{1}{\phi_{n+k+1} - \phi_*} - \dfrac{1}{\phi_n - \phi_*}}{\rho_k^{(n+1)} - \rho_k^{(n)}}, & \begin{array}{l} n = 0, 1, 2, \dots \\ k = 0, 1, 2, \dots \end{array} \end{cases}$$

Dans l'algorithme (D) seules les quantités $\rho_{2k}^{(n)}$ $(k = 0, 1, 2, \dots)$ sont intéressantes. Les quantités d'indice inférieur impair ne sont que des calculs intermédiaires. En effet, on a en posant

$$g(x) = \frac{1}{\phi(x) - \phi_*} :$$

$$\rho_{-1}^{(n)} = 0, \ \rho_0(x) = f(x), \ \rho_0^{(n)} = \rho_0(x_n) = f(x_n)$$

$$\rho_1(x, x_n) = \frac{g(x)-g(x_n)}{f(x)-f(x_n)} = \frac{g(x)-g(x_n)}{f(x) - \rho_0^{(n)}}$$

$$\Longrightarrow f(x) = \rho_0^{(n)} + \frac{g(x)-g(x_n)}{\rho_1(x, x_n)} \qquad (1)$$

$$\rho_{k+1}(x, x_n, \dots, x_{n+k}) = \rho_{k-1}^{(n)} + \frac{g(x) - g(x_{n+k})}{\rho_k(x, x_n, \dots, x_{n+k-1}) - \rho_k^{(n)}}$$

$$\Longrightarrow \rho_k(x, x_n, \dots, x_{n+k-1}) = \rho_k^{(n)} + \frac{g(x) - g(x_{n+k})}{\rho_{k+1}(x, x_n, \dots, x_{n+k}) - \rho_{k-1}^{(n)}} \qquad (2)$$

où $\rho_k^{(n)} = \rho_k(x_n, \dots, x_{n+k})$ et $\rho_{k-1}^{(n)} = \rho_{k-1}(x_n, \dots, x_{n+k-1})$.

Les relations (1) et (2) nous permettent d'écrire f(x) sous une forme de fraction continue :

$$f(x) = \rho_0^{(n)} + \frac{g(x)-g(x_n)|}{|\rho_1^{(n)}-\rho_{-1}^{(n)}} + \frac{g(x)-g(x_{n+1})|}{|\rho_2^{(n)}-\rho_0^{(n)}} + \ldots$$

Soit le convergeant $C_{2k}(x)$ de cette fraction continue :

$$C_{2k}(x) = \rho_0^{(n)} + \frac{g(x)-g(x_n)|}{|\rho_1^{(n)}-\rho_{-1}^{(n)}} + \ldots + \frac{g(x)-g(x_{n+2k})|}{|\rho_{2k+1}^{(n)}-\rho_{2k-1}^{(n)}}$$

D'après [3], le convergeant $C_{2k}(x)$ est une fraction rationnelle en g(x), dont le numérateur et le dénominateur sont de même degré k. De plus cette fraction interpole f aux points x_n, $x_{n+1}, \ldots, x_{n+2k}$ et elle vérifie

$$\lim_{x\to\infty} C_{2k}(x) = \rho_{2k}^{(n)}.$$

Quant aux $\rho_{2k+1}^{(n)}(k = 0, 1, 2, \ldots)$, ils ne sont que des calculs intermédiaires.

I.2.2. - Procédé d'extrapolation rationnelle inverse

Soit f(x) = 0, une équation non linéaire dans $\mathbb{R}$.

Soit x_* une racine simple de f, c'est-à-dire que f vérifie p_1 et p_2 au point x_*.

On utilise dans ce paragraphe les notations de I.1.1.

Dans le procédé (D) remplaçons y_n par x_n et ϕ par f, on obtient :

$$(E) : \begin{cases} \rho_{-1}^{(n)} = 0, \ \rho_0^{(n)} = x_n, \ n = 0, 1, 2, \ldots \\ \\ \rho_{k+1}^{(n)} = \rho_{k-1}^{(n+1)} + \dfrac{\dfrac{1}{f_{n+k+1}} - \dfrac{1}{f_n}}{\rho_k^{(n+1)} - \rho_k^{(n)}}, \ \begin{array}{l} n = 0, 1, 2, \ldots \\ k = 0, 1, 2, \ldots \end{array} \end{cases}$$

Ce procédé (E) est à la base de certaines méthodes de point fixe obtenues par C. Brézinski dans [2], et que nous étudierons en détail au chapitre II.

I.2.3. - Etude du procédé (E)

La fraction rationnelle généralisée qui a donné naissance au procédé (E) est :

$$R_{k,f}^{(n)}(x) = \frac{\sum_{i=0}^{k} a_{k-i}^{(n,k)} [f(x)]^i}{\sum_{i=0}^{k} b_{k-i}^{(n,k)} [f(x)]^i} \quad \text{avec } b_k^{(n,k)} = 1.$$

$$\lim_{x \to x_*} R_{k,f}^{(n)}(x) = a_k^{(n,k)} = \rho_{2k}^{(n)}.$$

Posons $X = f(x)$, $X_i = f(x_i)$ pour $i = n, \ldots, n+2k$.

Considérons les deux polynômes suivants :

$$P_k^{(n)}(X) = \sum_{i=0}^{k} a_{k-i}^{(n,k)} X^i \Rightarrow P_k^{(n)}(0) = a_k^{(n,k)}$$

$$Q_k^{(n)}(X) = \sum_{i=0}^{k} b_{k-i}^{(n,k)} X^i \Rightarrow Q_k^{(n)}(0) = b_k^{(n,k)} = 1.$$

Soit F la fonction inverse de f. Alors on a :

$$R_{k,f}^{(n)}(x_{n+i}) = x_{n+i} \Longleftrightarrow \frac{P_k^{(n)}(X_{n+i})}{Q_k^{(n)}(X_{n+i})} = F(X_{n+i}) \text{ pour } i = 0, 1, \ldots, 2k.$$

Désormais nous supposons que F n'a pas de pôles dans un voisinage de zéro. L'erreur d'interpolation est donnée par [3] :

$$F(X) - \frac{P_k^{(n)}(X)}{Q_k^{(n)}(X)} = \frac{1}{(2k+1)!} \frac{1}{[Q_k^{(n)}(X)]^2} \frac{d^{2k+1}}{d\tau^{2k+1}} \left[F(\tau)\ [Q_k^{(n)}(\tau)]^2\right] \prod_{i=0}^{2k} (X-X_{n+i})$$

avec $\tau \in]\min\{X, X_n,\ldots, X_{n+2k}\}, \max\{X, X_n,\ldots, X_{n+2k}\}[$.

Pour $X = 0$ et $X_{n+i} = f_{n+i} = f(x_{n+i})$, on a :

$$F(0) - \frac{P_k^{(n)}(0)}{Q_k^{(n)}(0)} = x_* - \rho_{2k}^{(n)} = - \frac{1}{(2k+1)!} \frac{d^{2k+1}}{d\tau^{2k+1}} \left[F(\tau)\ \{Q_k^{(n)}(\tau)\}^2\right] \prod_{i=0}^{2k} f_{n+i}$$

où $\tau \in]\min\{0, f_n,\ldots, f_{n+2k}\}, \max\{0, f_n,\ldots, f_{n+2k}\}[$.

Posons $D_{2k}^{(n)}(\tau) = \frac{d^{2k+1}}{d\tau^{2k+1}} \left[F(\tau)\ \{Q_k^{(n)}(\tau)\}^2\right]$ et $D_{2k}^{(n)} = \lim\limits_{\tau\to 0} D_{2k}^{(n)}(\tau)$.

Pour k fixé, on suppose que $D_{2k}^{(n)}$ converge vers D_{2k} (fini) quand n tend vers l'infini.

On peut montrer l'existence de D_{2k} par passage au cas confluent.

A partir de cette dernière forme de l'erreur, nous tirons un certain nombre de propriétés concernant le procédé (E).

Ces propriétés sont semblables à celles obtenues dans le cas polynomial. Elles seront aussi exploitées lors de l'étude de certaines méthodes de point fixe au chapitre suivant.

Propriété 10 :

Pour k *fixé,*

$$\rho_{2k}^{(n)} - x_* \sim (f'_*)^{2k+1} \frac{D_{2k}}{(2k+1)!} \prod_{i=0}^{2k} e_{n+i}, \quad n \to \infty.$$

Propriété 11 :

Pour k *fixé,*

$$\frac{\rho_{2k}^{(n+1)}-x_*}{\rho_{2k}^{(n)}-x_*} \sim (\phi'_*)^{2k+1}, \; n \to \infty \;;$$ *où* ϕ *est la fonction itérative génèrant la suite* (x_n).

Propriété 12 :

Pour k *fixé,*

$$\frac{\rho_{2k+2}^{(n)}-x_*}{\rho_{2k}^{(n)}-x_*} \sim \frac{f'^{\,2}_*}{(2k+2)(2k+3)} \frac{D_{2k+2}}{D_{2k}} e_{n+2k+1}\, e_{n+2k+2}, \; n \to \infty.$$

Les démonstrations de ces trois propriétés sont analogues à celles faites pour les propriétés concernant le procédé (B).

On peut ranger les quantités $\rho_k^{(n)}$ dans un tableau à double entrée, où k désignera l'indice des colonnes et n celui des diagonales descendantes :

$$\begin{array}{ccccc}
\rho_{-1}^{(0)} & & & & \\
 & \rho_0^{(0)} & & & \\
\rho_{-1}^{(1)} & & \rho_1^{(0)} & & \\
 & \rho_0^{(1)} & & \rho_2^{(0)} & \\
\rho_{-1}^{(2)} & & \rho_1^{(1)} & & \rho_3^{(0)} \\
 & \rho_0^{(2)} & & \rho_2^{(1)} & \vdots \\
\rho_{-1}^{(3)} & & \rho_1^{(2)} & \vdots & \\
 & \rho_0^{(3)} & \vdots & & \\
\rho_{-1}^{(4)} & \vdots & & & \\
\vdots & & & &
\end{array}$$

Les flèches indiquent le sens de progression des calculs.

Remarque :

Si f et ϕ sont reliés par la relation :

$$\phi(x) = x + \lambda f(x),\ \lambda \in \mathbb{R},\ \lambda \neq 0$$

alors on a $f(x_n) = \frac{1}{\lambda}\Delta x_n = f_n$, et l'algorithme (E) se réduit à :

$$(E') : \begin{cases} \rho_{-1}^{(n)} = 0,\ \rho_0^{(n)} = x_n,\ n = 0, 1, 2, \ldots \\ \rho_{k+1}^{(n)} = \rho_{k-1}^{(n+1)} + \lambda \dfrac{\left(\dfrac{1}{\Delta x_{n+k+1}} - \dfrac{1}{\Delta x_n}\right)}{\rho_k^{(n+1)} - \rho_k^{(n)}},\ \begin{matrix} n = 0, 1, 2, \ldots \\ k = 0, 1, 2, \ldots \end{matrix} \end{cases}$$

Le procédé (E) (respectivement (E')), d'après la propriété 12 peut accélérer la convergence d'une suite (x_n), si l'on sait trouver une fonction f satisfaisant au point x_* les deux propriétés :

$$p_1 = f(x_*) = 0$$
$$p_2 : f'(x_*) \neq 0.$$

Dans le cas où l'on ne sait pas trouver une telle fonction f, on peut y remédier par utilisation de fonctions de plusieurs variables. C'est ce qu'on va faire ci-dessous.

I.2.4. - Une généralisation du procédé (E)

Soit G comme en I.1.4. D'une façon parallèle à l'étude faite en I.1.4, on obtient :

$$(F) : \begin{cases} \rho_{-1}^{(n)} = 0,\ \rho_0^{(n)} = x_n,\ n = 0, 1, 2, \ldots \\ \rho_{k+1}^{(n)} = \rho_{k-1}^{(n+1)} + \dfrac{\dfrac{1}{G(x_{n+k+1}, \ldots, x_{n+k+1+p})} - \dfrac{1}{G(x_n, \ldots, x_{n+p})}}{\rho_k^{(n+1)} - \rho_k^{(n)}},\ \begin{matrix} n = 0, 1, 2, \ldots \\ k = 0, 1, 2, \ldots \end{matrix} \end{cases}$$

Posons $G_n = G(x_n, \ldots, x_{n+p})$, $G_{n+1} = G(x_{n+1}, \ldots, x_{n+1+p})$ et $G_{n+2} = G(x_{n+2}, \ldots, x_{n+2+p})$.

On a la :

Propriété 13 :

> *Si* (x_n) *est à convergence linéaire, c'est-à-dire* $\lim \frac{e_{n+1}}{e_n} = \rho$ *avec* $\rho \neq 1, 0$ *et si* $\sum_{i=0}^{p} \rho^i \, G_i'(x_*, \ldots, x_*) \neq 0$, *où*
>
> $$G_i'(x_*, \ldots, x_*) = \frac{\partial}{\partial x^i} G(x_*, \ldots, x_*), \text{ alors}$$
>
> $$\lim_{n\to\infty} \frac{\rho_2^{(n)} - x_*}{x_{n+1} - x_*} = 0.$$

Preuve :

$$\Delta\rho_1^{(n)} = \rho_1^{(n+1)} - \rho_1^{(n)} = \frac{G_{n+1}\, \Delta x_n \left[1 - \frac{G_{n+2}}{G_{n+1}}\right] - G_{n+2}\, \Delta x_{n+1} \left[1 - \frac{G_{n+1}}{G_n}\right]}{G_{n+1}\, G_{n+2}\, \Delta x_n\, \Delta x_{n+1}}$$

$$\rho_2^{(n)} = x_{n+1} + \frac{1 - \frac{G_{n+2}}{G_n}}{G_{n+2}\, \Delta\rho_1^{(n)}}$$

en remplaçant $\Delta\rho_1^{(n)}$ par son expression, on obtient :

$$\rho_2^{(n)} = x_{n+1} + \frac{(1 - \frac{G_{n+2}}{G_n})\, \Delta x_{n+1}}{(1 - \frac{G_{n+2}}{G_n}) - \frac{\Delta x_{n+1}}{\Delta x_n} \frac{G_{n+2}}{G_{n+1}}(1 - \frac{G_{n+1}}{G_n})}$$

Comme dans la preuve de la propriété 7 :

$$\lim_{n\to\infty} \frac{G_{n+2}}{G_{n+1}} = \rho, \ \lim_{n\to\infty} \frac{G_{n+1}}{G_n} = \rho \text{ et } \lim_{n\to\infty} \frac{G_{n+2}}{G_n} = \rho^2.$$

Or pour une suite à convergence linéaire on a aussi :

$$\rho = \lim_{n\to\infty} \frac{\Delta x_{n+1}}{\Delta x_n}.$$

En utilisant ces résultats on obtient :

$$\lim_{n\to\infty} \frac{\rho_2^{(n)} - x_\star}{e_{n+1}} = \lim_{n\to\infty} \left\{ 1 + \frac{(1 - \frac{G_{n+2}}{G_n})\ (\frac{e_{n+2}}{e_{n+1}} - 1)}{(1 - \frac{G_{n+2}}{G_{n+1}}) - \frac{\Delta x_{n+1}}{\Delta x_n} \frac{G_{n+2}}{G_{n+1}} (1 - \frac{G_{n+1}}{G_n})} \right\}$$

$$= 1 + \frac{(1-\rho^2)(\rho-1)}{(1-\rho) - \rho\rho(1-\rho)} = 1 + \frac{(1-\rho^2)(\rho-1)}{(1-\rho)(1-\rho^2)} = 0. \ \square$$

I.2.5. - Application à une classe de suite de la forme $x_n = f(n)$

Par un raisonnement analogue à celui fait au paragraphe I.1.5, nous obtenons l'algorithme :

$$(E'') : \begin{cases} \rho_{-1}^{(n)} = 0, \ \rho_0^{(n)} = x_n, \ n = 0, 1, 2, \ldots \\ \\ \rho_{k+1}^{(n)} = \rho_{k-1}^{(n+1)} + \dfrac{k+1}{\rho_k^{(n+1)} - \rho_k^{(n)}}, \ \begin{array}{l} n = 0, 1, 2, \ldots \\ k = 0, 1, 2, \ldots \end{array} \end{cases}$$

Cet algorithme s'obtient à partir du ρ-algorithme, en choisissant comme suite auxiliaire la suite $x_n = n$, $\forall n$. Cet algorithme a été utilisé par P. Wynn pour accélérer la convergence de certaines suites [27].

Si g est la fonction du paragraphe I.1.5. définie par :

$$g(x) = \frac{1}{f^{-1}(x)}$$

et si g est de classe C^{2k+1} ainsi que sa réciproque, alors l'erreur $\rho_{2k}^{(n)} - x_*$ prend la forme :

$$\rho_{2k}^{(n)} - x_* = \frac{1}{(2k+1)!} \frac{d^{2k+1}}{dy^{2k+1}} (g^{-1}(y) \left[B_k^{(n)}(y)\right]^2) \prod_{i=0}^{2k} \frac{1}{n+i}.$$

où $y \in \left]\min\{0, \frac{1}{n}, \ldots, \frac{1}{n+2k}\}, \max\{0, \frac{1}{n}, \ldots, \frac{1}{n+2k}\}\right[= \left]0, \frac{1}{n}\right[$.

Jusqu'ici on s'est intéressé aux fonctions f satisfaisant au point x_* les propriétés p_1 et p_2. Dans ce qui suit on va aborder le cas des fonctions f vérifiant p_1 et p_3. A partir de f nous construisons une fonction satisfaisant p_1 et p_2, à laquelle nous appliquons les algorithmes (B) et (E).

I.3. - CAS D'UNE RACINE MULTIPLE

Soit $f(x) = 0$, une équation non linéaire dans $\mathbb{R}$.

Soit x_* une racine multiple de f, c'est dire f vérifie au point x_* les propriétés p_1 et p_3 à savoir :

$$p_1 : f(x_*) = 0$$
$$p_3 : f^{(m)}(x_*) \neq 0,\ m \geq 2 \text{ et } f^{(i)}(x_*) = 0,\ 1 \leq i \leq m-1.$$

Alors f peut s'écrire sous la forme :

$$f(x) = (x-x_*)^m\, g(x),\ g(x_*) \neq 0.$$

Considérons la fonction $h(x) = \frac{f(x)}{f'(x)}$. Elle vérifie au point x_* les propriétés p_1 et p_2. En effet :

$$p_1 : h(x) = \frac{(x-x_*)^m\ g(x)}{m(x-x_*)^{m-1}\ g(x)+(x-x_*)^m\ g'(x)} = \frac{(x-x_*)\ g(x)}{mg(x)+(x-x_*)\ g'(x)}$$

$$\Rightarrow h(x_*) = 0$$

$$p_2 : h'(x) = \frac{m(g(x))^2 + (x-x_*)^2\ (g'(x))^2 - (x-x_*)^2\ g(x)\ g''(x)}{(mg(x) + (x-x_*)\ g'(x))^2}$$

$$h'(x_*) = \frac{m(g(x_*))^2}{m^2(g(x_*))^2} = \frac{1}{m},\ \text{car } g(x_*) \neq 0.$$

Donc x_* est une racine simple de h et nous pouvons appliquer les algorithmes (B) et (E) à la fonction h. Le problème pratique qui se pose est l'évaluation de la dérivée de f. King [15] en a proposé une approximation qui nous semble intéressante :

$$\frac{f(x-f(x))-f(x)}{-\ f(x)} \simeq f'(x).$$

Puisque l'accroissement f(x) tend vers 0, quand x tend vers x_*, alors on a bien :

$$\lim_{x\to x_*} \frac{f(x-f(x))-f(x)}{-\ f(x)} = f'(x_*).$$

I.3.1. - Etude de la fonction $G(x) = \frac{f(x)}{\frac{f(x-f(x))-f(x)}{-\ f(x)}}$

G vérifie p_1 et p_2 au point x_*. En effet :

$$f(x-f(x)) - f(x) = -\ f'(x)\ f(x) + \frac{f''(x)}{2!}\ f^2(x) - \frac{f^{(3)}(x)}{3!}\ (f(x))^3 + \ldots$$

$$= -\ f'(x)\ f(x) \left\{1 - \frac{f''(x)}{2}\ \frac{f(x)}{f'(x)} + \frac{f^{(3)}(x)}{6}\ \frac{(f(x))^2}{f'(x)} + \cdots\right\}$$

Posons $h(x) = \frac{f(x)}{f'(x)}$, d'où la relation :

$$f(x-f(x)) - f(x) = - f'(x)\, f(x) \left\{1 - \frac{f''(x)}{2} h(x) + \frac{f^{(3)}(x)}{6} f(x)\, h(x) + \dots\right\}$$

On peut écrire G sous la forme :

$$G(x) = \frac{-(f(x))^2}{f(x-f(x))-f(x)} = - \frac{(f(x))^2}{-f'(x)\, f(x) \left\{1 - \frac{f''(x)}{2} h(x) + \frac{f^{(3)}(x)}{6} f(x)\, h(x) + \dots\right\}}$$

$$= \frac{h(x)}{1-\epsilon(x)} \text{ où } \epsilon(x) = \quad h(x) \left\{\frac{f''(x)}{2} - \frac{f^{(3)}(x)}{6} f(x) + \dots\right\}.$$

$\epsilon(x) \to 0$ quand $x \to x_*$. Donc au voisinage de x_* :

$$G(x) = \frac{h(x)}{1-\epsilon(x)} = h(x) \{1 + \epsilon(x) + (\epsilon(x))^2 + \dots\}$$

$$\Rightarrow G'(x) = h'(x)\{1 + \epsilon(x) + \dots\} + h(x)\{\epsilon'(x) + 2\epsilon'(x)\, \epsilon(x) + \dots\}$$

$$= h'(x)\{1 + \epsilon(x) + (\epsilon(x))^2 + \dots\} + h(x)\, \epsilon'(x) \{1 + 2\epsilon(x) + 3(\epsilon(x)^2 + \dots\}$$

Les deux relations donnant les expressions de G et de G' nous permettent de dire que :

$$G(x_*) = 0,$$
$$G'(x_*) = \frac{1}{m} = h'(x_*),$$

et G satisfait p_1 et p_2.

Nous allons maintenant appliquer à la fonction G les algorithmes (B) et (E), afin de construire des méthodes de résolution des équations ayant une racine multiple. Ces méthodes seront étudiées en détail au chapitre suivant.

I.3.2. - Application de (B) à la fonction G

Soit $x_n \to x_*$. Posons $f_n = f(x_n)$ et $G_n = G(x_n)$.

On obtient l'algorithme :

$$
(G) : \begin{cases} T_0^{(n)} = x_n, \; n = 0, 1, 2, \ldots \\[2ex] T_k^{(n)} = \dfrac{f_{n+k}^2 [f(x_n - f_n) - f_n]\, T_{k-1}^{(n)} - f_n^2 [f(x_{n+k} - f_{n+k}) - f_{n+k}]\, T_{k-1}^{(n+1)}}{f_{n+k}^2 [f(x_n - f_n) - f_n] - f_n^2 [f(x_{n+k} - f_{n+k}) - f_{n+k}]} \\[2ex] n = 0, 1, 2, \ldots \quad \text{et } k = 1, 2, 3, \ldots \end{cases}
$$

Supposons f et ϕ reliées par la relation :

$$\phi(x) = x - f(x) \Longrightarrow f_n = - \Delta x_n$$

$$G_n = \frac{-(f(x_n))^2}{f(x_n - f(x_n)) - f(x_n)} = \frac{-(-\Delta x_n)^2}{f(x_n + \Delta x_n) + \Delta x_n} = \frac{-(\Delta x_n)^2}{f(x_{n+1}) + \Delta x_n} = \frac{-(\Delta x_n)^2}{-\Delta x_{n+1} + \Delta x_n} = \frac{(\Delta x_n)^2}{\Delta^2 x_n}.$$

Dans ces conditions l'algorithme (G) se réduit à l'algorithme (G') :

$$
(G') : \begin{cases} T_0^{(n)} = x_n \\[2ex] T_k^{(n)} = \dfrac{\dfrac{(\Delta x_{n+k})^2}{\Delta^2 x_{n+k}} T_{k-1}^{(n)} - \dfrac{(\Delta x_n)^2}{\Delta^2 x_n} T_{k-1}^{(n+1)}}{\dfrac{(\Delta x_{n+k})^2}{\Delta^2 x_{n+k}} - \dfrac{(\Delta x_n)^2}{\Delta^2 x_n}}, \quad \begin{array}{l} n = 0, 1, 2, \ldots \\ k = 1, 2, 3, \ldots \end{array} \end{cases}
$$

I.3.3. - Etude de l'algorithme (G)

On peut mettre l'algorithme (G) sous la forme :

$$T_0^{(n)} = x_n$$

$$T_k^{(n)} = \frac{G_{n+k}\, T_{k-1}^{(n)} - G_n\, T_{k-1}^{(n+1)}}{G_{n+k} - G_n}, \quad n = 0, 1, 2, \ldots \quad k = 1, 2, 3, \ldots$$

avec $G_n = -(f_n)^2/(f(x_n - f_n) - f_n)$ et $f_n = f(x_n)$.

La propriété 1 s'écrit sous la forme :

Propriété 14 :

Pour k *fixé,*

$$T_k^{(n)} - x_* \sim \frac{(-1)^k}{(k+1)!} \frac{(G^{-1})^{(k+1)}(0)}{m^{(k+1)}} \prod_{i=0}^{k} e_{n+i}, \quad n \to \infty$$

où $e_{n+i} = x_{n+i} - x_*$ *et* m *est la multiplicité de* x_*.

Remarque :

Pour obtenir cette propriété, il suffit de remplacer dans la propriété 1 f'_* par $G'_* = \frac{1}{m}$ et $F^{(k+1)}(0)$ par $(G^{-1})^{(k+1)}(0)$, où G^{-1} désigne l'inverse de G.

Si $x_{n+1} = \phi(x_n)$ alors $e_{n+1} = \phi'_* e_n + \ldots$ Et la propriété 14 prend la forme :

Propriété 15 :

Pour k *fixé,*

$$T_k^{(n)} - x_* \sim \frac{(-1)^k}{m^{k+1}} (\phi_*')^k \frac{(G^{-1})^{(k+1)}(0)}{(k+1)!} e_n^{k+1}, \; n \to \infty.$$

Cette propriété est importante pour étudier surtout l'algorithme (G').

En effet, si $\phi(x) = x - f(x)$ alors $\phi'(x) = 1 - f'(x)$ et $\phi_*' = 1$. Et on a la :

Propriété 16 :

Pour k *fixé,*

$$\frac{T_{k+1}^{(n)} - x_*}{T_k^{(n)} - x_*} \to 0, \; \textit{quand } n \to \infty.$$

Preuve :

$$\frac{T_{k+1}^{(n)} - x_*}{T_k^{(n)} - x_*} \sim -\frac{1}{m} \frac{(G^{-1})^{(k+2)}(0)}{(G^{-1})^{(k+1)}(0)} \frac{1}{(k+2)} e_n \to 0, \text{ quand } n \to \infty. \; \square$$

Cet algorithme est donc capable d'accélérer des suites itératives de la forme précédente (à savoir : $x_{n+1} = \phi(x_n)$ et $\phi_*' = 1$).

Remarque :

L'algorithme (G') est capable d'accélérer des suites itératives à convergence linéaire. La quantité $T_k^{(n)}$ est une approximation d'ordre k+1 de la limite x_*. Cet algorithme entre dans une catégorie d'algorithmes regroupant le procédé d'Overholt [20] et le procédé (B') d'interpolation inverse de Germain-Bonne-Wimp. En effet :

Soit (x_n) une suite de la forme :

$$x_{n+1} = x_* + a_1(x_n - x_*) + a_2(x_n - x_*)^2 + \ldots$$

avec $\quad 0 < |a_1| < 1$

et $\quad \dfrac{\Delta x_{n+1}}{\Delta x_n} \to a_1.$

Nous verrons au chapitre III que, si $T_{k-1}^{(n)} - x_* = O(e_n^k)$ alors toute approximation de a_1 du premier ordre, ou toute approximation de a_1^k du premier ordre, conduit à une approximation $T_k^{(n)}$ telle que :

$$T_k^{(n)} - x_* = O(e_n^{k+1}).$$

$$\text{Or ici } A_1 = \frac{\dfrac{(\Delta x_{n+k})^2}{\Delta^2 x_{n+k}}}{\dfrac{(\Delta x_n)^2}{\Delta^2 x_n}} = \frac{\Delta x_{n+k}}{\Delta x_n} \; \frac{\dfrac{\Delta x_{n+1}}{\Delta x_n} - 1}{\dfrac{\Delta x_{n+k+1}}{\Delta x_{n+k}} - 1}$$

$$\to a_1^k \, \frac{a_1 - 1}{a_1 - 1} = a_1^k$$

Ce procédé nous fournit pour des suites de la forme précédente des quantités $T_k^{(n)}$ telles que :

$$T_k^{(n)} - x_* = O(e_n^{k+1}).$$

I.3.4. - Application de l'algorithme (E) à G

L'application de (E) à G donne :

$$(H) : \begin{cases} \rho_{-1}^{(n)} = 0, \; \rho_0^{(n)} = x_n, \; n = 0, 1, 2, \ldots \\[2ex] \rho_{k+1}^{(n)} = \rho_{k-1}^{(n+1)} + \dfrac{\dfrac{f(x_n - f_n) - f_n}{f_n^2} - \dfrac{f(x_{n+k+1} - f_{n+k+1}) - f_{n+k+1}}{f_{n+k+1}^2}}{\rho_k^{(n+1)} - \rho_k^{(n)}}, \quad \begin{array}{l} n = 0, 1, \ldots \\ k = 0, 1, 2, \ldots \end{array} \end{cases}$$

Cet algorithme (H) se réduit à la forme suivante :

$$\phi(x) = x - f(x)$$

$$(H') : \begin{cases} \rho_{-1}^{(n)} = 0, \ \rho_0^{(n)} = x_n \\[2ex] \rho_{k+1}^{(n)} = \rho_{k-1}^{(n+1)} + \dfrac{\dfrac{\Delta^2 x_{n+k+1}}{(\Delta x_{n+k+1})^2} - \dfrac{\Delta^2 x_n}{(\Delta x_n)^2}}{\rho_k^{(n+1)} - \rho_k^{(n)}}, & n = 0, 1, 2, \dots \\ & k = 0, 1, 2, \dots \end{cases}$$

Pour l'algorithme (H) on a la propriété :

Propriété 17 :

Pour k *fixé,*

$$\rho_{2k}^{(n)} - x_* \sim \frac{1}{m^{2k+1}} \frac{D_{2k}}{(2k+1)!} \prod_{i=0}^{2k} e_{n+i}, \ n \to \infty \ ;$$

où m *est la multiplicité de* x_*, $D_{2k}^{(n)} = \dfrac{d^{2k+1}}{d\tau^{2k+1}} \left[G^{-1}(\tau)(B_k^n(\tau))^2 \right]_{\tau=0}$

et $D_{2k} = \lim\limits_{n \to \infty} D_{2k}^{(n)}$.

Preuve :

Pour obtenir cette propriété, il suffit de se reporter à la propriété 10 et de remplacer f'_* par $G'_* = \frac{1}{m}$ et F par G^{-1}, où G^{-1} est la fonction inverse de G. □

I.4. - CONCLUSION

Les deux procédés précédents d'interpolation inverse (B) et (E) peuvent être utilisés pour accélérer la convergence d'une suite $(x_n)_{n\in\mathbb{N}}$ si l'on sait choisir une fonction f telle que :

p_1 : $\lim_{n\to\infty} f(x_n) = f(x_*) = 0$ où $x_* = \lim_{n\to\infty} x_n$.

p_2 : $f'(x_*) \neq 0$.

Si f vérifie p_3 : $f^{(m)}(x_*) \neq 0$ et $f^{(i)}(x_*) = 0$ pour $1 \leq i \leq m-1$, on peut utiliser les procédés (G) et (H).

Si (x_n) est à convergence linéaire on peut utiliser (B), (E), (G) et (H) sous leurs formes réduites respectives (B'), (E'), (G') et (H').

L'importance capitale de ces algorithmes va se manifester le long du chapitre suivant. Car ils sont à la base des méthodes de point fixe que nous aborderons.

I.5. - ESSAIS NUMERIQUES

Les algorithmes (B), (B'), (B") et (G') sont programmés suivant le schéma ci-dessous :

$T_0^{(n)} = x_n$, $n = 0, 1, 2, \ldots$

$T_k^{(n)} = T_{k-1}^{(n)} + X_n(T_{k-1}^{(n+1)} - T_{k-1}^{(n)})/(X_{n+k}-X_n)$, $n = 0, 1, \ldots$

et $k = 1, 2, \ldots$

où

$X_i = f(x_i)$ pour (B),

$X_i = \Delta x_i$ pour (B')

$X_i = i+1$ pour (B")

$X_i = (\Delta x_i)^2/\Delta^2 x_i$ pour (G').

L'algorithme (E") suit la règle habituelle du ρ-algorithme avec $X_i = i$ comme suite auxiliaire.

Les deux derniers essais, portant sur (E') et (H'), utilisent le sous-programme RHO1 du ρ-algorithme de C. Brézinski [9] avec comme suites auxiliaires $(1/\Delta x_i)_{i\in\mathbb{N}}$ pour (E') et $(1/[(\Delta x_i)^2/\Delta^2 x_i])_{i\in\mathbb{N}}$ pour (H').

Application de l'algorithme (B) à $x_{n+1} = x_n - f(x_n)/f'(x_n)$ avec $f(x) = x - e^{-x}$

$x_0 = 0.5$ et $x_* = 0.567\ 143\ 290\ 409\ 783$

n	$T_0^{(n)} = x_n$	$T_1^{(n)}$	$T_2^{(n)}$
0	.5000000000000000+00		
1	.5663110031972181+00	.5671337407973724+00	
2	.5671431650348623+00	.5671432903908499+00	.5671432904097450+00
3	.5671432904097810+00	.5671432904097859+00	.5671432904097859+00
4	.5671432904097838+00	.5671432904097859+00	.5671432904097838+00

n	$T_3^{(n)}$	$T_4^{(n)}$
0	.5671432904097859+00	
1	.5671432904097859+00	.5671432904097859+00

Application de l'algorithme (B') à $x_{n+1} = e^{-x_n}$ avec $x_0 = 1$

$x_* = 0.567\ 143\ 290\ 409\ 783$

n	$T_0^{(n)} = x_n$	$T_1^{(n)}$	$T_2^{(n)}$
0	1.0000000000000000	.5822260969956228	.5671256979845161
1	.3678794411714423	.5717057675272521	.5671545188305761
2	.6922006275553464	.5686388058644661	.5671419398650336
3	.5004735005636368	.5676169948466354	.5671436140565280
4	.6062435350855974	.5672967524886339	.5671432389975992
5	.5453957859750270	.5671924278872064	
6	.5796123355033789		
7	.5601154613610891		

n	$T_3^{(n)}$	$T_4^{(n)}$	$T_5^{(n)}$
0	.5671503876193773	.5671432389174804	.5671432908868727
1	.5671439270495359	.5671432936999647	.5671432904214792
2	.5671433605196419	.5671432902243899	
3	.5671432973727664		

Application de (G') à $x_{n+1} = e^{-x_n}$, $x_0 = 1$

n	$T_0^{(n)}$	$T_1^{(n)}$	$T_2^{(n)}$
0	.10000000000000000D+01	.57515544467714300D+00	.56754563256268130D+00
1	.36787944117144230D+00	.56979632904412580D+00	.56708291351591080D+00
2	.69220062755534640D+00	.56797675389102100D+00	.56715543682261020D+00
3	.50047350056363680D+00	.56741431660348230D+00	.56714120235305420D+00
4	.60624353508655974D+00	.56722986281943690D+00	.56714368386017640D+00
5	.54539578597502700D+00	.56717123951144630D+00	.56714321997064790D+00
6	.57961233550337690D+00	.56715226077289020D+00	.56714330339533270D+00
7	.56011546136108910D+00	.56714617922336690D+00	
8	.57114311508017700D+00		
9	.56487934739104950D+00		
10	.56842872502906070D+00		

n	$T_3^{(n)}$	$T_4^{(n)}$	$T_5^{(n)}$
0	.56714698334554560D+00	.56714347525947260D+00	.56714329071686920D+00
1	.56714380229920550D+00	.56714328108867310D+00	.56714329042988320D+00
2	.56714333682553270D+00	.56714329100060420D+00	.56714329041029500D+00
3	.56714329561907830D+00	.56714329037669560D+00	.56714329040980410D+00
4	.56714329092580680D+00	.56714329041177560D+00	
5	.56714329046454490D+00		

n	$T_6^{(n)}$	$T_7^{(n)}$	$T_8^{(n)}$
0	.56714329042106570D+00	.5671432904097F480D+00	.5671432904097840D+00
1	.56714329040959490D+00	.56714329040978400D+00	
2	.56714329040978770D+00		

Applications des algorithmes (B'), (G') et (B") à : $x_n = e^{\frac{n}{n+1}}$

(suite à convergence logarithmique)

Algorithme (B') : $x_* = e = 2.718\ 281\ 828$

n	$T_0^{(n)} = x_n$	$T_1^{(n)}$	$T_2^{(n)}$	$T_3^{(n)}$
0	1.000000000000000	2.2034002223077708	2.3852782762111684	2.4823646763880391
1	1.6487212707001280	2.3378221997465822	2.4661206485100091	2.5324574081176920
2	1.9477340410546755	2.4195486221546356	2.5157219658125571	2.5653982457787647
3	2.117000016612 6743	2.4728614172655985	2.5491269325345038	2.5885270502295066
4	2.2255409284924674	2.5101706101299694	2.5731153043691135	
5	2.3009758906928246	2.5376863159245147		
6	2.3564184423836600			

n	$T_4^{(n)}$	$T_5^{(n)}$	$T_6^{(n)}$
0	2.5390487801421946	2.5760594840923658	2.6018252
1	2.5728963877794382	2.6001389604347635	
2	2.5962707849631718		

Algorithme (G') :

n	$T_0^{(n)} = x_n$	$T_1^{(n)}$	$T_2^{(n)}$
0	1.0000000000000000	2.5179319349747824	2.6480074233210409
1	1.6487212707001280	2.5970090015925073	2.6786281736300268
2	1.9477340410546755	2.6364788865559332	2.6938796417823232
3	2.1170000166126743	2.6592517194155531	2.7021846513140140
4	2.2255409284924674	2.6736264587009384	2.7070934396400974
5	2.3009758908928246	2.6832968061288342	2.7101843924290449
6	2.3564184423836600	2.6901207452089135	2.7122296115793174
7	2.3988752939670976	2.6951189546367087	
8	2.4324254542872074		
9	2.4596031111569494		
10	2.4820650846230117		

n	$T_3^{(n)}$	$T_4^{(n)}$	$T_5^{(n)}$
0	2.6914850986647087	2.7086789265479795	2.7148693541946445
1	2.7046122230288301	2.7136516789444353	2.7167429623854367
2	2.7105465437740172	2.7158350683458836	2.7175200972658111
3	2.7135665497638856	2.7168892632243111	2.7178751891378114
4	2.7152399956242503	2.7174412360530999	
5	2.7162298943312777		

n	$T_6^{(n)}$	$T_7^{(n)}$	$T_8^{(n)}$
0	2.7171217689002851	2.7179025652659572	2.7181631375931508
1	2.7177879594081602	2.7181292221429249	
2	2.7180516536393184		

Algorithme (B") :

n	$T_0^{(n)} = x_n$	$T_1^{(n)}$	$T_2^{(n)}$
0	1.0000000000000000		
		2.2974425414002559	
1	1.6487212707001280		2.66991810194557[illegible]
		2.5457595817637706	
2	1.9477340410546755		2.70383630460957[illegible]
		2.6247979432806708	
3	2.1170000166126743		2.71206452[illegible]
		2.6597045760116398	
4	2.2255409284924674		2.71504295606[illegible]
		2.6781507028946105	
5	2.3009758908928246		

n	$T_3^{(n)}$	$T_4^{(n)}$	$T_5^{(n)}$
0	2.71514[illegible]	2.7161519135[illegible]	2.71[illegible]
1	2.71755[illegible]	2.7186[illegible]	
2	2.7180[illegible]		

Application de (B) à $x_n = e^{\frac{n}{n+1}}$, $x_* = e = 2.718\ 281$

avec $f(x) = 1 - \mathrm{LOG}(x)$.

$$\begin{cases} f(e) = 0 \\ f'(e) = -\frac{1}{e}. \end{cases}$$

n	$T_0^{(n)}$	$T_1^{(n)}$	$T_2^{(n)}$
0	1.0000000000000000	2.2974425414002559	2.6699181019455293
1	1.6487212707001280	2.5457595817637717	2.7038363048095695
2	1.9477340410546759	2.6247979432866706	2.7120645250990896
3	2.1170000166126743	2.6597045760116382	2.7150429566605556
4	2.2255409284924674	2.6781507028946107	2.7163813724138406
5	2.3009758908928250	2.6890737513286764	2.7170717662186077
6	2.3564184423836605	2.6960732550511592	
7	2.3986752939670976		

n	$T_3^{(n)}$	$T_4^{(n)}$	$T_5^{(n)}$
0	2.7151423724309160	2.7181519135073995	2.7182781129228965
1	2.7175500052921029	2.7182570796869805	2.7182812310333915
2	2.7180213882220214	2.7182743306487027	2.7182816709999280
3	2.7181659267515536	2.7182789183682185	
4	2.7182224225598861		

n	$T_6^{(n)}$	$T_7^{(n)}$
0	2.7182817507184737	2.7182818272178639
1	2.7182818176554402	

(E") et x(n) = exp(n/(n+1)) : de limite $x_* = 2.718\ 281$

x_n	$\rho_2^{(n)}$	$\rho_4^{(n)}$
1.0000000000000000		
1.6487212707001280	2.7580791739154140	
1.9477340410546755	2.7279103584384889	2.7182501345261107
2.1170000166126743	2.7220972276965973	2.7182765630448590
2.2255409284924674	2.7201819060387295	2.7182803264370071
2.3009758908928246	2.7193653293671147	
2.3564184442383660		

$\rho_6^{(n)}$
2.7182816338218357

Algorithme (E') appliqué à $x = e^{-x}$.

	n	
$\rho_0^{(n)}$	1	.1000000000000000D+01
	2	.3678794411714423D+00
	3	.6922006275553464D+00
	4	.5004735005636368D+00
	5	.6062435350855974D+00
	6	.5453957859750270D+00
	7	.5796123355033789D+00
	8	.5601154613610891D+00
	9	.5711431150801770D+00
	10	.5648793473910495D+00
	11	.5684287250290607D+00
$\rho_2^{(n)}$	3	.5674416067776430D+00
	4	-.7521857962117659D+00
	5	.5986388647276892D+00
	6	.4805511815268300D+00
	7	.5882673737037885D+00
	8	.5478392119542277D+00
	9	.5755158660314246D+00
	10	.5616133920104687D+00

	n	
$\rho_4^{(n)}$	5	.5671433446650072D+00
	6	.5672384493804359D+00
	7	.5102813339213609D+00
	8	.5518201962496101D+00
	9	.5725027002944560D+00
	10	.5637809771061547D+00
$\rho_6^{(n)}$	7	.5671432875283628D+00

<u>Algorithme (H') appliqué à $x = e^{-x}$</u>.

	n	
$\rho_2^{(n)}$	3	.5676273757552500D+00
	4	-.7471923993107725D+00
	5	.5986920188620608D+00
	6	.4806072097636357D+00
	7	.5882756011549304D+00
	8	.5478432236464032D+00
	9	.5755168889354584D+00
	10	.5625757472939544D+00
$\rho_4^{(n)}$	5	.5671434378856193D+00
	6	.5673096813592793D+00
	7	.5103792754363482D+00
	8	.5518232379236524D+00
	9	.5725045901266192D+00
	10	.5640947176326883D+00
$\rho_6^{(n)}$	7	.5671432824716934D+00

CHAPITRE II

ETUDE DE METHODES DE POINT FIXE DEDUITES DES ALGORITHMES (B), (B'), (E), (E'), (G) ET (G')

INTRODUCTION

Dans le présent chapitre, composé de quatre parties, nous abordons des méthodes de point fixe construites à l'aide des algorithmes (B), (B'), (E), (E'), (G) et (G') et étudiés au chapitre précédent.

La première et la deuxième parties sont réservées aux méthodes destinées à résoudre des équations non linéaires à racine simple. Ainsi, dans la première partie, nous étudions en détail des méthodes d'interpolation polynomiale, déduites des procédés (B') et (B). Nous construisons d'autres méthodes par composition. Dans cette étude, nous déterminons l'ordre de convergence, les coefficients dominant et asymptotique et l'indice d'efficacité de chacune de ces méthodes. Dans la seconde partie, nous reprenons ce travail pour des méthodes d'interpolation rationnelle.

Quant à la troisième partie, elle est réservée à un rappel sur certaines méthodes, puis à une comparaison de toutes les méthodes citées précédemment. Cette comparaison est essentiellement basée sur le critère de l'indice d'efficacité.

Dans la dernière partie nous regardons de plus près certaines méthodes construites à l'aide des algorithmes (G'), (G), (H') et (H). Ces méthodes concernent particulièrement les équations à racine multiple.

Pour terminer, nous donnons quelques essais numériques.

II.0. - DEFINITIONS

1) *Ordre de convergence* d'une méthode de point fixe.

Soit (x_n) une suite itérative convergeant vers x_*.

$$\text{Si } \lim_{n\to\infty} \frac{|x_{n+1}-x_*|}{|x_n-x_*|^r} = C \text{ où } C \in \,]0, +\infty[$$

et $r > 0$, alors on dira que r est l'ordre de convergence de la suite (x_n), ou de la méthode itérative qui l'a générée.

Pour une autre définition beaucoup plus générale on pourra consulter [7].

2) *Coefficient asymptotique d'erreur*

Dans les conditions de la définition 1), on dira que C est le coefficient asymptotique d'erreur.

3) *Coefficient dominant d'erreur*

Soit ϕ une méthode de point fixe, basée sur p points, c'est-à-dire :

$$x_n = \phi(x_{n-1},\ldots, x_{n-p})$$

Soit $e_{n-i} = |x_{n-i}-x_*|$ pour $i = 0, 1, 2,\ldots, p$.

Soient $p_1, p_2, \ldots p_p$; p nombres réels tels que :

$$\lim_{n\to\infty} \frac{e_n}{\prod_{i=1}^{p} e_{n-i}^{p_i}} = K \text{ où } K \in \,]0, +\infty[,$$

alors K sera dit coefficient dominant d'erreur.

Remarque :

Si r existe selon la définition 1), il ne peut pas être strictement inférieur à 1. En effet, d'une part : $e_n \to 0$, $n \to \infty$; et d'autre part :

$$\frac{e_{n+1}}{e_n^r} = C + \phi_n \text{ où } \phi_n \to 0.$$

Soit $\alpha > 0$ tq $1 = r + \alpha$. On a $r < 1$.

$$\frac{e_{n+1}}{e_n} = \frac{C+\phi_n}{e_n^{\alpha}} \to +\infty, \ n \to \infty.$$

D'autre par si $r = 1$ alors $C \leq 1$. En effet, si $r = 1$ et si $C > 1$ alors

$$\exists N \text{ tel que } \frac{e_{n+1}}{e_n} > 1, \ \forall n \geq N$$

$\Rightarrow e_{n+1} > e_n$, $\forall n \geq N \Rightarrow (e_n)$ est une suite de nombres positifs strictement croissante à partir d'un certain rang.

Ce qui est en contradiction avec $x_n \to x_*$.

Propriété 0 :

Si $e_n \sim K \prod_{i=1}^{p} e_{n-i}^{p_i}$; $n \to \infty$, *alors* :

r *vérifie l'équation* :

$$P_p(r) = r^p - \sum_{i=1}^{p} p_i \, r^{p-i} = 0$$

K *et* C *sont reliés par* :

$$K = C^{Q_{p-1}(r)} \quad \text{où } Q_{p-1}(r) = \sum_{i=0}^{p-1} \left(1 - \sum_{j=0}^{i} p_j\right) r^{p-1-i}$$

avec $p_0 = 0$.

Preuve :

Elle est faite par récurrence sur p.

Si p = 1, on a : $e_n \sim K\, e_{n-1}^{p_1}$.

On obtient $r = p_1$ et $K = C$. Donc r satisfait

$$P_1(r) = r - p_1 = 0$$

De même $Q_0(r) = 1$.

$$\left.\begin{array}{l} \text{Si } p = 2\text{, on a : } e_n \sim K\, e_{n-1}^{p_1}\, e_{n-2}^{p_2} \\ \\ \text{Or, } \dfrac{e_n}{e_{n-1}^r} \sim C \Rightarrow e_n \sim C\, e_{n-1}^r \end{array}\right\} \Rightarrow \ldots$$

$$\ldots \left\{\begin{array}{l} C_{n-1}^{r-p_1} \sim K\, e_{n-2}^{p_2} \\ \\ e_{n-1} \sim C\, e_{n-2}^r \end{array}\right\} \Rightarrow C^{1-p_1+r} \sim K\, e_{n-2}^{p_2+p_1 r - r^2}$$

On obtient que r doit satisfaire :

$$P_2(r) = r^2 - p_1 r - p_2 = 0,$$

et que $Q_1(r) = r + (1-p_1)$.

Supposons que l'on ait P_j et Q_{j-1} pour $j \leq p-1$. Montrons ce résultat pour j = p.

$$\left.\begin{array}{l} e_n \sim K \prod_{i=1}^{p} e_{n-i}^{p_i} \\ \\ e_n \sim C\, e_{n-1}^{r} \end{array}\right\} \Rightarrow C\, e_{n-1}^{r-p_1} \sim K \prod_{i=2}^{p} e_{n-i}^{p_i} \Rightarrow e_{n-1} \sim \left(\frac{K}{C}\right)^{\frac{1}{r-p_1}} \prod_{i=2}^{p} e_{n-i}^{\frac{p_i}{r-p_1}}$$

$$\Rightarrow e_n \sim K' \prod_{i=1}^{p-1} e_{n-i}^{p_i'}, \text{ avec } K' = \left(\frac{K}{C}\right)^{\frac{1}{r-p_1}} \text{ et } p_i' = \frac{p_{i+1}}{r-p_1},\ 1 \le i \le p-1.$$

On applique l'hypothèse de récurrence. r doit satisfaire :

$$R_{p-1}(r) = r^{p-1} - p_1' r^{p-2} - \dots - p_{p-2}' r - p_{p-1}' = 0$$

$$\Longleftrightarrow P_p(r) = r^p - p_1 r^{p-1} - \dots - p_{p-1} r - p_p = 0.$$

Puis K' et C sont reliés par :

$$K' = C^{S_{p-2}(r)} \text{ où } S_{p-2}(r) = \sum_{i=0}^{p-2} \left(1 - \sum_{j=0}^{i} p_j'\right) r^{p-2-i} \text{ avec } p_0' = 0.$$

Comme $K' = \left(\frac{K}{C}\right)^{\frac{1}{r-p_1}}$ on a :

$$K = C^{1+(r-p_1)S_{p-2}(r)}.$$

$Q_{p-1}(r) = 1+(r-p_1)\, S_{p-2}(r)$?

$$1+(r-p_1)\, S_{p-2}(r) = 1 + (r-p_1) \sum_{i=0}^{p-2} \left(1-p_0 - \sum_{j=1}^{i} \frac{p_{j+1}}{r-p_1}\right) r^{p-2-i}$$

$$= 1 + \sum_{i=0}^{p-2} \left(r-p_1 - \sum_{j=1}^{i} p_{j+1}\right) r^{p-2-i}$$

$$= 1 + \sum_{i=0}^{p-2} \left(r - \sum_{j=0}^{i+1} p_j\right) r^{p-2-i}$$

$$= 1 + \sum_{i=1}^{p-1} \left(r - \sum_{j=0}^{i} p_j\right) r^{p-1-i} = 1 + \sum_{i=1}^{p-1} r^{p-i} - \sum_{i=1}^{p-1} \sum_{j=0}^{i} p_j\, r^{p-1-i}.$$

$$Q_{p-1}(r) = \sum_{i=0}^{p-1} (1 - \sum_{j=0}^{i} p_j)\, r^{p-1-i} = r^{p-1} - \sum_{i=1}^{p-1} \sum_{j=0}^{i} p_j\, r^{p-1-i} + \sum_{i=1}^{p-1} r^{p-1-i}$$

$$= 1 + \sum_{i=1}^{p-1} r^{p-i} - \sum_{i=1}^{p-1} \sum_{j=0}^{i} p_j\, r^{p-1-i}$$

Donc on obtient

$$Q_{p-1}(r) = 1 + (r-p_1)\, S_{p-2}(r). \ \square$$

4) *Indice d'efficacité d'une méthode*

Soit ϕ une méthode destinée à résoudre une équation non linéaire $f(x) = 0$. Soit r, s'il existe, l'ordre de convergence de ϕ. On supposera que $r > 1$.

Soit $p (p \in \mathbb{N}^*)$, le nombre total des évaluations de fonctions au cours d'une itération.

La quantité,

$$I(\phi) = r^{\frac{1}{p}}$$

sera dite : Indice d'efficacité de ϕ.

Le nombre $I(\phi)$ représente le facteur par lequel on multiplie le nombre de chiffres exacts significatifs par opération élémentaire.

La notion d'indice d'efficacité a été introduite par Ostrowski [19]. Pour une autre notion, on peut consulter [24].

On remarque que plus l'indice d'efficacité est grand et plus la vitesse de la convergence est élevée.

5) *Racine simple ou multiple*

Soit $f(x) = 0$, une équation non linéaire.

Soit x_* une racine de f.

a) x_* est dit racine simple de f si $f'(x_*) \neq 0$, ou encore si f vérifie p_1 et p_2 au point x_* (voir chapitre I).

b) x_* est dit racine multiple de f de multiplicité $m \geq 2$ si :

$$f^{(i)}(x_*) = 0 \text{ pour } 1 \leq i \leq m-1$$

$$\text{et } f^{(m)}(x_*) \neq 0,$$

ou encore si f vérifie au point x_* les propriétés p_1 et p_3 (voir chapitre I).

6) Soit (T) un procédé d'accélération de la convergence qui donnent des quantités $T_k^{(n)}$ pour différentes valeurs de n et de k à partir des initialisations

$$T_0^{(n)} = x_n,\ n = 0, 1, 2,\ldots$$

Soit ϕ une méthode de résolution de

$$f(x) = 0,$$

basée sur k+1 points.

Nous dirons que ϕ correspond à la suite colonne $(T_k^{(n)})_{n\in\mathbb{N}}$ du procédé (T) si

$$x_{n+k+1} = \phi(x_n,\ldots, x_{n+k}) = T_k^{(n)},\ n = 0, 1,\ldots$$

où $T_k^{(n)}$ est calculé par le procédé (T) à partir des initialisations $T_0^{(n)} = x_n$, n = 0, 1, 2,...

II.1. - METHODES D'INTERPOLATION POLYNOMIALE INVERSE
(Cas d'une racine simple)

II.1.1. - Méthodes de type-Steffensen

1) *Méthode* S(1) *ou méthode de Steffensen*

Le procédé de l'accélération de la convergence correspondant à cette méthode, est le procédé Δ^2-d'Aitken bien connu [1] ou, en d'autres termes, la suite colonne $(T_1^{(n)})_{n\in\mathbb{N}}$ du procédé (B').

En effet, soit une équation non linéaire :

$$f(x) = 0$$

$$\Longleftrightarrow x = \phi(x).$$

x_0 donné.

La n$^{\text{ième}}$ itération de S(1) est donnée par :

$$\left[\begin{array}{l} u_0 = x_{n-1} \\ u_1 = \phi(u_0) \\ u_2 = \phi(u_1) \\ \text{On applique (B') à } \{u_0, u_1, u_2\}. \\ \text{On prend :} \\ x_n = T_1^{(0)} = \dfrac{(\Delta u_1)\, u_0 - (\Delta u_0)\, u_1}{\Delta u_1 - \Delta u_0} \end{array}\right.$$

Si ϕ est de classe C^2 dans un voisinage de x_*, la racine simple de $f(x) = x - \phi(x)$, et si $\phi'_* = \phi'(x_*) \neq 1$ alors :

$$\lim_{n\to\infty} \frac{x_{n+1}-x_*}{(x_n-x_*)^2} = -\frac{1}{2}\,\frac{\phi'_*\ \phi''_*}{1-\phi'_*}$$

$S(1)$ est d'ordre $r_1=2$. $S(1)$ exige deux évaluations de fonctions par itération ; $p = 2$.

Son indice d'efficacité $I(S(1)) = r_1^{\frac{1}{p}} = \sqrt{2} = 1.414\ldots$

On peut retrouver le coefficient asymptotique ainsi que l'ordre de cette méthode en utilisant la propriété 5 (Chapitre I).

En effet :

$$\frac{x_{n+1}-x_*}{x_n-x_*} = \frac{T_1^{(0)}-x_*}{T_0^{(0)}-x_*} \sim -\frac{1}{2}\, f'_* \,\frac{F^{(2)}(0)}{F^{(1)}(0)}\, e_1 \text{ où } e_1 = u_1 - x_*$$

$$e_1 = u_1-x_* = \phi(u_0)-(x_*) = \phi(x_n)-\phi(x_*) = \phi'_*(x_n-x_*) + \ldots$$

Comme $f'_{(x)} = 1 - \phi'(x)$ et $F'(0) = \frac{1}{f'_*}$, $F''(0) = -\dfrac{f_*^{(2)}}{f_*'^{3}}$ alors on tire facilement :

$$\frac{x_{n+1}-x_*}{(x_n-x_*)^2} \sim \frac{1}{2}\,\frac{(1-f'_*)\, f_*^{(2)}}{f'_*} = -\frac{1}{2}\,\frac{\phi'_*\,\phi''_*}{1-\phi'_*}.$$

2) <u>*S(2) ou $1^{ère}$ méthode de type-Steffensen*</u>

La méthode $S(2)$ correspond en fait à la suite colonne $(T_2^{(n)})_{n\in\mathbb{N}}$ du procédé (B').

x_0 donné, sa $n+1^{\text{ième}}$ itération est donnée par :

$$\left[\begin{array}{l} u_0 = x_n \\ u_1 = \phi(u_0) \\ u_2 = \phi(u_1) \\ u_3 = \phi(u_2) \end{array}\right.$$

On applique l'algorithme (B') à l'ensemble $\{u_0, u_1, u_2, u_3\}$;

$$T_0^{(i)} = u_i \quad i = 0, 1, 2$$

$$T_i^{(j)} = \frac{\Delta u_{i+j}\; T_{i-1}^{(j)} - \Delta u_j\; T_{i-1}^{(j+1)}}{\Delta u_{i+j} - \Delta u_j}, \; i = 1, 2 \qquad \text{et} \quad j = 0, \ldots, 2-i$$

On prend enfin $x_{n+1} = T_2^{(0)}$.

La méthode S(2) décrite ci-dessus, exige 3 évaluations de fonctions par itération.

Comme $\phi(x) = x-f(x)$, alors on peut remplacer Δu_i par $f(u_i)$ et retrouver le procédé (B). D'après la propriété 1 (Chapitre I) :

$$x_{n+1}-x_* = T_2^{(0)}-x_* \sim (f'_*)^3 \frac{F^{(3)}(0)}{3!} (u_0-x_*)(u_1-x_*)(u_2-x_*)$$

$$\sim (1-f'_*)^3 (f'_*)^3 \frac{F^{(3)}(0)}{3!} (x_n-x_*)^3$$

d'où, en remplaçant $F^{(3)}$ par son expression en fonction des dérivées de f :

$$\frac{x_{n+1}-x_*}{(x_n-x_*)^3} \sim -\frac{1}{6}(1-f'_*)^3 \frac{(f'_* f_*^{(3)} - 3f_*''^2)}{f_*'^2}, \; n \to \infty.$$

S(2) est d'ordre $r_2 = 3$ et d'indice d'efficacité $I(S(2)) = \sqrt[3]{3} = 1.442\ldots$

L'indice d'efficacité de S(2) est légèrement supérieur à celui de S(1).

3) *S(k) ou $(k-1)^{ième}$ méthode de type-Steffensen, $k \geq 3$*

Cette méthode correspond à la suite colonne $(T_k^{(n)})_{n\in\mathbb{N}}$ du procédé (B'). x_0 étant donné, sa $(n+1)^{\text{ième}}$ étape est donnée par :

$u_0 = x_n$, $u_i = \phi(u_{i-1})$ pour $i = 1, 2, \ldots, k+1$.

On applique l'algorithme (B') à $\{u_0, u_1, \ldots, u_{k+1}\}$:

$$T_0^{(i)} = u_i, \quad i = 0, 1, \ldots, k$$

$$T_i^{(j)} = \frac{\Delta u_{i+j}\, T_{i-1}^{(j)} - \Delta u_j\, T_{i-1}^{(j+1)}}{\Delta u_{i+j} - \Delta u_j}, \quad i = 1, \ldots, k \quad j = 0, 1, \ldots, k-i.$$

Enfin on prend :

$$x_{n+1} = T_k^{(0)} \text{ ; et on note } x_{n+1} = S(k)(x_n).$$

Notons $e_i = x_i - x_*$ et $\epsilon_i = u_i - x$.

Comme $\Delta u_i = -f(u_i)$ alors on peut appliquer la propriété 1. (Chapitre I). On obtient :

$$e_{n+1} \sim (-1)^k\, (f'_*)^{k+1}\, \frac{F^{(k+1)}(0)}{(k+1)!} \prod_{i=0}^{k} \epsilon_i, \quad n \to \infty$$

Or $\epsilon_0 = e_n$ et $\epsilon_i \sim (1-f'_*)^i\, e_n$. D'où :

$$e_{n+1} \sim (-1)^k\, (f'_*)^{k+1}\, (1-f'_*)^{\frac{k(k+1)}{2}}\, \frac{F^{(k+1)}(0)}{(k+1)!}\, e_n^{k+1}.$$

Ou encore en exprimant $F^{(k+1)}$ par les dérivées de f (Chapitre I, I.1.3.) :

$$e_{n+1} \sim \frac{(-1)^k}{(k+1)!} (1-f'_*)^{\frac{k(k+1)}{2}} \left(\sum_{\beta} (-1)^r (j+r-1)! \prod_{i=2}^{j} \frac{A_i^{\beta_i}}{\beta_i !}\right) e_n^{k+1}.$$

S(k) est d'ordre $r_k = k+1$, exigeant k+1 évaluations de fonctions par itération. Son indice d'efficacité est :

$$I(S(k)) = (k+1)^{\frac{1}{k+1}}.$$

<u>Exemples</u> :

k	k+1	I(S(k))
1	2	1.414 214
2	3	1.442 25
3	4	1.414 214
4	5	1.379 73
5	6	1.348 01
6	7	1.320 47
7	8	1.296 84

II.1.2. - Méthodes composées de S(k)

On se limite dans ce paragraphe à l'étude de la composée $S_{12} = S(2) \circ S(1)$, où le symbole O signifie que, chaque itération se décompose en deux sous-étapes.

La première correspond à l'application de S(1) pour obtenir $S(1)(x_n)$. La seconde correspond à l'application de S(2) pour obtenir $x_{n+1} = S(2)(S(1)(x_n))$.

Etant donné x_0, sa $(n+1)^{\text{ième}}$ itération s'effectue suivant le schéma :

$$\begin{cases} u_0 = x_n,\ u_1 = \phi(u_0),\ u_2 = \phi(u_1). \\ S(1)(x_n) = \dfrac{(\Delta u_1)u_0 - (\Delta u_0)u_1}{\Delta u_1 - \Delta u_0} \\ V_0 = S(1)(x_n),\ V_1 = \phi(V_0),\ V_2 = \phi(V_1),\ V_3 = \phi(V_2) \\ T_0^{(j)} = V_j,\ j = 0,\ 1,\ 2 \\ T_i^{(j)} = \dfrac{\Delta V_{i+j}\ T_{i-1}^{(j)} - \Delta V_j\ T_{i-1}^{(j+1)}}{\Delta V_{i+j} - \Delta V_j},\ i = 1,\ 2 \qquad \text{et } j = 0,\ 1,\ 2-i. \\ x_{n+1} = T_2^{(0)} = S(2)(S(1)(x_n)) = S(2) \circ S(1)(x_n) = S_{12}(x_n). \end{cases}$$

Soit f tel que $f(x) = x-\phi(x)$. On a :

$$\Delta U_i = - f(U_i) \text{ et } \Delta V_i = - f(V_i)$$

Posons $e_i = x_i - x_*$, $\epsilon_i = V_i - x_*$, $\epsilon_i' = U_i - x_*$.

Notons par K_1 (respectivement K_2, K_{12}) le coefficient asymptotique correspondant à S_1 (respectivement, S_2, S_{12}).

D'après l'étude faite au paragraphe précédent :

$$e_{n+1} \sim K_2\ \epsilon_0^3,\ n \to \infty$$

$$\epsilon_0 \sim K_1\ e_n^2,\ n \to \infty$$

d'où :

$$e_{n+1} \sim K_2\ K_1^3\ e_n^6 = K_{12}\ e_n^6$$

$$K_{12} = K_2\ K_1^3 = - \frac{1}{24}\ (1-f_*')^5\ \frac{(f_*'')^2(f_*' f_*^{(3)} - 3f_*''^2)}{f_*'^4}.$$

S_{12} exige 5 évaluations de fonctions pour un ordre de convergence égal à 6. Son indice d'efficacité $I(S_{12})$ est :

$$I(S_{12}) = 6^{\frac{1}{5}} = 1.431$$

Remarque :

Pratiquement on perd sur l'indice d'efficacité ce que l'on gagne sur l'ordre de convergence. La cause principale est le nombre très grand d'évaluations de fonctions par itération.

Dans ce qui suit on va aborder des méthodes beaucoup plus efficaces, car elles exigent moins d'évaluations de fonctions. Ce sont les méthodes dites de type-sécante.

II.1.3. - Méthodes de type-sécante

1) *Méthode E(1) ou méthode de la sécante*

Cette méthode correspond à la suite colonne $(T_1^{(n)})_{n\in\mathbb{N}}$ du procédé (B).

Soit en effet une équation non linéaire :

$$f(x) = 0$$

de racine simple x_*.

x_0, x_1 donnés.

La $n+1^{\text{ième}}$ étape $(n \geq 1)$ de E(1) est donnée par :

$$\begin{cases} \text{On applique l'algorithme (B) à } \{x_{n-1}, x_n\}. \\ T_0^{(i)} = x_{n-1+i},\ i = 0, 1 \\ T_1^{(0)} = \dfrac{f_n T_0^{(0)} - f_{n-1} T_0^{(1)}}{f_n - f_{n-1}} \text{ ou } f_i = f(x_i) \text{ pour } i = n-1, n. \\ x_{n+1} = T_1^{(0)}. \end{cases}$$

ou plus simplement $x_{n+1} = \dfrac{f_n\, x_{n-1} - f_{n-1}\, x_n}{f_n - f_{n-1}}$, $n \geq 1$.

A part la première itération qui exige deux évaluations de fonctions, les autres n'en demandent qu'une seule.

La propriété 2 (Chapitre I) donne :

$$e_{n+1} \sim \frac{1}{2} \frac{f''_*}{f'_*} e_{n-1}\, e_n$$

Posons $K_1 = \frac{1}{2} \frac{f''_*}{f'_*}$. K_1 est le coefficient dominant de l'erreur.

Soit r_1 l'ordre de convergence de E(1). Soit C_1 le coefficient asymptotique de l'erreur de E(1).

D'après la propriété O($p=2$, $p_1 = p_2 = 1$) :

l'ordre r_1 est la racine réelle positive de l'équation :

$$E(1) : P_2(r) = r^2 - r - 1 = 0.$$

Les coefficients C_1 et K_1 sont reliés par :

$$C_1 = K_1^{\frac{1}{Q_1(r_1)}}$$

où $Q_1(r) = \sum_{j=0}^{1} (1 - \sum_{i=0}^{j} p_i) r^{1-j}$ avec $p_0 = \theta$, $p_1 = 1$

$\Rightarrow Q_1(r) = r.$

d'où $C_1 = K_1^{\frac{1}{r_1}}$.

L'ordre est $r_1 = 1.618...$ Le nombre p_1 d'évaluations de fonctions par itération étant $p_1 = 1$, l'indice d'efficacité de E(1) est égal à son ordre r_1

$$I(E(1)) = 1.618...$$

Dans ce qui suit je vais étudier deux méthodes de type-sécante, c'est-à-dire généralisant en quelque sorte la méthode de la sécante.

Nous allons voir que :

- l'ordre r_k est racine de l'équation :

  $$E(k) : r^{k+1} - (r^k + r^{k-1} + ... + 1) = 0.$$

- la suite $(r_k)_{k\in\mathbb{N}}$ est telle que :

  $$1 = r_0 < r_1 < r_2 < ... < r_k < ... < 2$$

  $$r_k \to 2.$$

- l'indice d'efficacité est $I(E(k)) = r_k$.

2) E(2) ou $1^{ère}$ méthode de type-sécante

La méthode E(2) correspond en fait à la suite colonne $(T_2^{(n)})_{n\in\mathbb{N}}$ du procédé (B). Après avoir donné x_0, x_1, x_2 sa $n^{ième}$ étape ($n \geq 3$) est donnée par :

$n^{\text{ième}}$ itération : On applique l'algorithme (B) à $\{x_{n-3}, x_{n-2}, x_{n-1}\}$:

$$T_0^{(i)} = x_{n-3+i},\ i = 0,\ 1,\ 2$$

$$T_j^{(i)} = \frac{f_{i+j}T_{j-1}^{(i)} - f_i T_{j-1}^{(i+1)}}{f_{i+j} - f_i},\ j = 1,\ 2 \qquad \text{et } i = 0,\dots,\ 2-j$$

où $f_i = f(x_{n-3+i})$.

$x_n = T_2^{(0)}$.

Cette méthode E(2) exige une seule évaluation de fonction par itération, sauf pour la première où elle demande 3 évaluations.

Si r_2 est son ordre de convergence et si I(E(2)) est son indice d'efficacité alors

$$I(E(2)) = r_2.$$

Détermination de r_2 et des coefficients dominant K_2 et asymptotique C_2 :

La propriété 2 (Chapitre I) nous permet d'écrire :

$$e_n = T_2^{(0)} - x_* \sim \frac{1}{6}(\Sigma(-1)^r(r+2)! \prod_{i=2}^{3} \frac{(A_i)^{\beta_i}}{\beta_i!})\ e_{n-3}\ e_{n-2}\ e_{n-1}$$

ou encore :

$$e_n \sim \frac{1}{6}\ (\frac{3f_*''^2 - f_*'\ f_*^{(3)}}{f_*'^2})\ e_{n-3}\ e_{n-2}\ e_{n-1}.$$

Ainsi :

$$K_2 = \frac{1}{6}\ \frac{3f_*''^2 - f_*'\ f_*^{(3)}}{f_*'^2}.$$

Dans la propriété 0 (Chapitre II) nous avons donc

$$p = 3,\ p_0 = 0,\ p_1 = p_2 = p_3 = 1$$

On obtient l'équation E(2) donnant l'ordre r_2 :

$$E(2) : r^3 - r^2 - r - 1 = 0 = P_3(r)$$

$$Q_2(r) = \sum_{i=0}^{2} (1 - \sum_{j=0}^{i} p_j)\, r^{2-i}$$

$$= r^2 + (1-p_1)r + (1-p_1-p_2) = r^2-1.$$

Et C_2 est donné par :

$$C_2 = K_2^{\frac{1}{r_2^2-1}}.$$

r_2 est l'unique racine réelle positive de p_3. La démonstration fera l'objet de la propriété 1).

Remarques :

1) Posons $f_i = f(x_i)$ et $f[x, y] = \frac{f(x)-f(y)}{x-y}$. La méthode E(2) peut se mettre sous la forme :

$$x_0,\ x_1,\ x_2 \text{ donnés}$$

$$x_n = x_{n-1} - \frac{f_{n-1}}{f[x_{n-1},\ x_{n-2}]} + \frac{f_{n-1}\ f_{n-2}}{f_{n-1} - f_{n-3}} \left\{\frac{1}{f[x_{n-1},\ x_{n-2}]} - \frac{1}{f[x_{n-2},\ x_{n-3}]}\right\},\ n \geq 3.$$

Elle se trouve chez Traub [24] sous le nom de méthode d'approximation inverse parabolique :

2) Soit F(x) = x, une équation de point fixe. Si, dans l'expression ci-dessus donnant x_n, on substitue F(x)-x à f(x), on obtient la méthode d'approximation parabolique de Popovski [22] :

$$x_0 \text{ donné},\ x_1 = F(x_0),\ x_2 = x_1 + \frac{x_1-x_0}{(x_1-x_0)-(F_1-x_1)}\,(F_1-x_1).$$

$$x_n = x_{n-1} + \left\{\left[\frac{\Delta x_{n-2}}{(F_{n-2}-x_{n-2})-(F_{n-1}-x_{n-1})} - \frac{\Delta x_{n-3}}{(F_{n-3}-x_{n-3})-(F_{n-2}-x_{n-2})}\right]\right.$$

$$\left.\frac{F_{n-1} - x_{n-1}}{(F_{n-3}-x_{n-3})-(F_{n-1}-x_{n-1})} + \frac{\Delta x_{n-2}}{(F_{n-2}-x_{n-2})-(F_{n-1}-x_{n-1})}\right\}$$

$(F_{n-1}-x_{n-1})$, $n \geq 3$. Où $F_i = F(x_i)$

Interprétation géométrique : E(2)

$$D_1 : y = \frac{f_0-f_1}{x_0-x_1}\,x + \frac{f_1x_0-f_0x_1}{x_0-x_1},$$

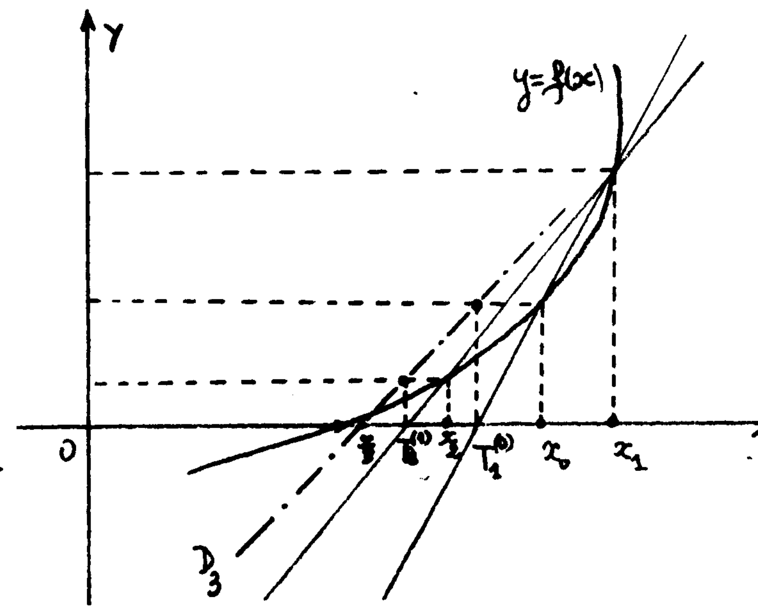
coupe OX en $T_1^{(0)}$.

$$D_2 : y = \frac{f_1-f_2}{x_1-x_2}\,x + \frac{f_2x_1-f_1x_2}{x_1-x_2}$$

coupe OX en $T_1^{(1)}$.

$$D_3 : y = \frac{f_0-f_2}{T_1^{(0)}-T_1^{(1)}}\,x + \frac{f_1T_1^{(0)}-f_0T_1^{(1)}}{T_1^{(0)}-T_1^{(1)}}$$

coupe OX en x_3



3) E(k) ou $(k-1)^{ième}$ *méthode de type-sécante*

La méthode E(k) correspond à la suite colonne $(T_k^{(n)})_{n\in\mathbb{N}}$ du procédé (B).

On se donne k+1 points : $x_0, x_1, x_2, \ldots, x_k$.

La $n^{ième}$ itération est donnée par l'application de l'algorithme (B) à l'ensemble $\{x_{n-k}, \ldots, x_{n-1}\ ;\ n > k\}$ suivant le schéma :

$$\left[\begin{array}{l}
T_0^{(i)} = x_{n-k-1+i} \text{ pour } i = 0, 1, \ldots, k \\[2ex]
T_j^{(i)} = \dfrac{f_{i+j}T_{j-1}^{(i)} - f_iT_{j-1}^{(i+1)}}{f_{i+j} - f_i}, \text{ pour } j = 1, 2, \ldots, k \\
\qquad\qquad\qquad\qquad\qquad\quad \text{et } i = 0, 1, \ldots, k-j \\[2ex]
\text{avec } f_\ell = f(x_{n-k-1+\ell}) \text{ pour } \ell = 0, 1, \ldots, k. \\[2ex]
\qquad x_n = T_k^{(0)}.
\end{array}\right.$$

D'après la propriété 1 (Chapitre I) :

$$e_n \sim (-1)^k\,(f_*')^{k+1}\,\frac{F^{(k+1)}(0)}{(k+1)!}\prod_{i=0}^{k} e_{n-k-1+i}$$

Notons par r_k l'ordre de convergence de cette méthode E(k), par C_k (respectivement K_k) le coefficient asymptotique (respectivement dominant) de l'erreur, et par I(E(k)) l'indice d'efficacité.

$$K_k = (-1)^k \; (f'_*)^{k+1} \; \frac{F^{(k+1)}(0)}{(k+1)!}.$$

Propriété 1 [24] :

r_k est l'unique racine réelle positive (une seule racine réelle : Règle de Descartes) de l'équation

$$E(k) : P_{k+1}(r) = r^{k+1} - r^k - r^{k-1} - \ldots - r-1 = 0$$

De plus $1 = r_0 < r_1 < r_2 < \ldots < r_k < \ldots < 2$ *et* $\lim\limits_{k\to\infty} r_k = 2.$

Preuve :

r_k vérifie l'équation E(k). En effet remplaçons dans la propriété 0 p et $p_i (i = 0, 1,\ldots, p)$ par leurs valeurs correspondantes, à savoir :

$p = k+1$, $p_0 = 0$ et $p_i = 1$ pour $i = 1, 2,\ldots, k+1$.

L'équation $P_{k+1}(r) = 0$, donnée par la propriété 0, se réduit à E(k).

D'après la propriété 0 on a aussi

$$Q_k(r) = r^k - r^{k-2} - 2r^{k-3} - \ldots - (k-2)r - (k-1).$$

Maintenant on va montrer par récurrence sur k que r_{k-1} est l'unique racine réelle positive de P_k et que P_k est strictement croissant dans l'intervalle $[r_{k-2}, 2]$.

Posons $P_0(r) = 1$ et $I_k = [r_k, 2]$.

Les polynômes P_{k-1} et P_k sont reliés par la relation :

$$P_k(r) = r\, P_{k-1}(r) - 1.$$

Si $k = 1$, $P_1(r) = r-1$ est strictement croissant dans $\mathbb{R}$, et en particulier dans $I_0 = [1, 2]$.

Si $k = 2$

$$P_2(r) = r\, P_1(r) - 1$$

$$P_1(r) \geq 0 \text{ et } P_1'(r) > 0 \text{ pour tout } r \in I_0.$$

$$\left.\begin{array}{l} P_2'(r) = r\, P_1'(r) + P_1(r) \\ \quad r \in I_0 \\ \quad P_1(r) \geq 0 \text{ et } P_1'(r) > 0 \end{array}\right\} \Longrightarrow P_2'(r) > 0 \text{ pour tout } r \in I_0.$$

Donc $P_2(r)$ est strictement croissant dans I_0.

Or $P_2(r_0) = P_2(1) = -1$ et $P_2(2) = 1$. En utilisant le théorème de la valeur intermédiaire on a :

$$\exists r_1 \in I_0 - \{1, 2\} \text{ tel que : } P_2(r_1) = 0.$$

Ce nombre r_1 est unique dans I_0, car P_2 est strictement croissant. On a aussi $P_2(r) \geq 0$ pour $r \in I_1$.

En dehors de l'intervalle I_0, P_2 ne s'annule pas. En effet : soit $r \in [2, +\infty]$ et $k \in 1$.

$$P_k(r) = r^k-(1 +\ldots+ r^{k-1}) = r^k - \frac{1-r^k}{1-r} = \frac{r^{k+1}-2r^k+1}{r-1}$$

$$\left.\begin{array}{r} P_k(r) > 0 \\ k \geq 2 \end{array}\right\} \Longleftrightarrow \left\{\begin{array}{r} r^k(r-2)+1 > 0 \\ r \geq 2 \end{array}\right.$$

Donc $P_k(r) > 0 \quad \forall r \geq 2$ et $\forall k \in \mathbb{N}^*$.

Soit $r \in [0, r_0[= [0, 1[$.

$$P_2(r) = r\, P_1(r)-1$$
$$\text{et } P_1(r) = r-1$$

Donc $P_2(r) < 0 \quad \forall r \in [0, r_0[$.

Supposons maintenant que pour $j \leq k-1$ on ait :

$P_j(r)$ strictement croissant dans I_{j-2},

$P_j(r) > 0$ pour $r \in [2, +\infty[$,

$P_j(r) < 0$ pour $r \in I_{j-2}$,

et r_{j-1} l'unique racine réelle positive de P_j dans l'intervalle I_{j-2}.

Montrons que ces propriétés restent vraies pour $j = k$.

$P_k(r) = r\, P_{k-1}(r) - 1 \Rightarrow P_k'(r) = P_{k-1}(r) + r\, P_{k-1}'(r)$.

Or sur I_{k-2}, $P_{k-1}(r) \geq 0$ et $P_{k-1}'(r) > 0$. Donc $P_k'(r) > 0$

$\Rightarrow P_k$ est strictement croissant sur I_{k-2}.

$$P_k(r_{k-2}) = r_{k-2}\, P_{k-1}(r_{k-2}) - 1 = -1$$
$$P_k(2) = 2\, P_{k-1}(2) - 1 = +1$$

En utilisant à nouveau le théorème de la valeur intermédiaire on a :

$\exists r_{k-1} \in I_{k-2} - \{r_{k-2}, 2\}$ tel que $P_k(r_{k-1}) = 0$.

r_{k-1} est unique dans I_{k-2} car P_k y est strictement croissant.

On a aussi $P_k(r) \geq 0$ pour $r \in I_{k-1}$.

On a vu que $P_k(r) > 0$ pour $r \in [2, +\infty]$.

Pour $r \in I_{k-2} - \{r_{k-2}\}$, $P_{k-1}(r) < 0$.

Comme $P_k(r) = r\,P_{k-1}(r)-1$ alors $P_k(r) < 0$ dans $I_{k-2}-\{r_{k-2}\}$.

Par conséquent parmi les réels positifs, seul r_k annule P_{k+1}.

On en tire aussi le fait que la suite $(r_k)_{k\in\mathbb{N}}$ est strictement croissante majorée par 2.

$\lim\limits_{k\to\infty} r_k = 2$. En effet :

$1 = r_0 < r_1 < r_2 <\ldots< r_k <\ldots< 2.$

Si $k \geq 1$, $r_k \neq 1$, et $r_k \neq 0$.

$$P_{k+1}(r_k) = r_k^{k+1}-(1 + r_k +\ldots+ r_k^k) = r_k^{k+1} - \frac{1-r_k^{k+1}}{1-r_k} = 0$$

$$\iff 2r_k^{k+1} = r_k^{k+2} +1 \iff 2 = r_k + \frac{1}{r_k^{k+1}}.$$

On sait que $r_1 = \frac{1+\sqrt{5}}{2} = 1.618\ldots$

Pour $k \geq 2$, $r_1 < r_k \Rightarrow \frac{1}{r_k} < \frac{1}{r_1} \Rightarrow \frac{1}{r_k^{k+1}} < (\frac{1}{r_1})^{k+1}$

Donc $2 = r_k + (\frac{1}{r_k})^{k+1} \leq r_k + (\frac{1}{r_1})^{k+1}$

$\Rightarrow 2 \leq \lim\limits_{k\to\infty} r_k + 0.$ □

On voit, d'après cette propriété que l'on améliore l'ordre de convergence lorsque k augmente. Mais il reste toujours inférieur à deux.

L'avantage principal de la méthode E(k) est qu'elle ne demande qu'une seule évaluation de fonction par itération, sauf à la première où elle exige la donnée de k+1 points et k+1 évaluations de fonctions. Par suite son indice d'efficacité I(E(k)) est égal à son ordre r_k.

$$I(E(k)) = r_k.$$

Quelques valeurs de r_k seront données à la fin de ce chapitre.

II.1.4. - Composition des méthodes E(k)

Soit un entier $k \geq 2$.

La méthode composée $E_{1\ldots k} = E(k)\ 0\ldots0\ E(1)$ est définie par : x_0 et x_1 étant donnés, la $n^{\text{ième}}$ étape est donnée par :

$$T_0^{(0)} = x_{n-2},\ T_0^{(1)} = x_{n-1},\ E_1 = \frac{f(x_{n-2})T_0^{(1)} - f(x_{n-1})T_0^{(0)}}{f(x_{n-2}) - f(x_{n-1})}$$

$$T_0^{(2)} = E_1,\ u_0 = x_{n-2},\ u_1 = x_{n-1},\ u_2 = E_1.$$

Pour $i = 1,\ldots, k$:

$$T_\ell^{(j)} = \frac{f_{j+\ell}\ T_{\ell-1}^{(j)} - f_j\ T_{\ell-1}^{(j+1)}}{f_{j+\ell} - f_j} \quad \text{pour } j = 1,\ldots, i \text{ ou } f_j = f(u_j).\quad \ell = i-j.$$

$$E_i = T_i^{(0)}$$

$i \neq k$ on reprend avec $T_0^{(i+1)} = E_i$

$i = k$ on prend $x_n = E_k$.

Soit $\epsilon_i = u_i - x_*$ et $e_n = x_n - x_*$.

Propriété 2 :

$$e_n \sim K_k \prod_{i=1}^{k-1} K_i^{2^{k-1-i}}\ e_{n-2}^{2^{k-1}}\ e_{n-1}^{2^{k-1}}.$$

L'ordre r *de convergence de* $E_{1\ldots k}$ *satisfait* :

$$r^2 - 2^{k-1}\ r - 2^{k-1} = 0$$

Il est donné par $r = 2^{k-2} + \sqrt{2^{k-2}(2^{k-2}+2)}$.

Preuve :

D'après l'étude précédente sur les méthodes E(k) :

$$e_n \sim K_k \prod_{i=0}^{k} \epsilon_i$$

$$\epsilon_0 = e_{n-2}$$

$$\epsilon_1 = e_{n-1}$$

$$\epsilon_i = K_{i-1} \prod_{j=0}^{i-1} \epsilon_j \quad i = 2,\dots, k$$

Posons $\epsilon_{k+1} = e_n$. On a :

$$\epsilon_i = K_{i-1} \prod_{j=0}^{i-1} \epsilon_j \quad i = 2,\dots, k+1.$$

On va montrer par récurrence sur i que

$$\epsilon_i = K_{i-1} \prod_{j=1}^{i-2} K_j^{2^{i-2-j}} \; \epsilon_0^{2^{i-2}} \; \epsilon_1^{2^{i-2}} \quad \text{pour } i = 2,\dots, k+1$$

avec la convention $\prod_{j=1}^{i-2} K_j^{2^{i-2-j}} = 1$ pour $i = 2$.

On a :

$$\epsilon_2 \sim K_1 \; \epsilon_0 \, \epsilon_1$$

$$\epsilon_3 \sim K_2 \; \epsilon_0 \, \epsilon_1 \, \epsilon_2 \sim K_2 \, K_1 \, \epsilon_0^2 \, \epsilon_1^2.$$

Supposons que l'on ait :

$$\epsilon_i \sim K_{i-1} \prod_{j=1}^{i-2} K_j^{2^{i-2-j}} \; \epsilon_0^{2^{i-2}} \; \epsilon_1^{2^{i-2}} \quad \text{pour } i \leq k.$$

La propriété reste vraie pour k+1. En effet :

$$\epsilon_{k+1} \sim K_k \prod_{i=0}^{k} \epsilon_i = K_k \, \epsilon_0 \, \epsilon_1 \prod_{i=2}^{k} \epsilon_i$$

Or par hypothèse de récurrence :

$$\epsilon_2 \sim K_1\, \epsilon_0\, \epsilon_1$$

$$\epsilon_3 \sim K_2\, K_1\, \epsilon_0^2\, \epsilon_1^2$$

$$\epsilon_4 \sim K_3\, K_2\, K_1^2\, \epsilon_0^{2^2}\, \epsilon_1^{2^2}$$

....

$$\epsilon_k \sim K_{k-1} \ldots K_{k-2} \ldots K_{k-3}^2 \ldots K_2^{2^{k-4}}\, K_1^{2^{k-3}}\, \epsilon_0^{2^{k-2}}\, \epsilon_1^{2^{k-2}}$$

$$\Rightarrow \prod_{i=2}^{k} \epsilon_i \sim K_{k-1}(K_{k-2}\, K_{k-2})(K_{k-3}^2\, K_{k-3}\, K_{k-3}) \ldots (K_{k-1}^{2^{i-2}} \ldots K_{k-i}\, K_{k-i}) \ldots$$

$$(K_1^{2^{k-3}} \ldots K_1 K_1)(\epsilon_0\, \epsilon_0^2\, \epsilon_0^{2^2} \ldots \epsilon_0^{2^{k-2}})(\epsilon_1\, \epsilon_1^2\, \epsilon_1^{2^2} \ldots \epsilon_1^{2^{k-2}})$$

Or $1 + 2 + 2^2 + \ldots + 2^m = \dfrac{2^{m+1}-1}{2-1} = 2^{m+1}-1$. Il s'ensuit que :

$$\prod_{i=2}^{k} \epsilon_i \sim K_{k-1}\, K_{k-2}^2\, K_{k-3}^{2^2} \ldots K_1^{2^{k-2}}\, \epsilon_0^{2^{k-1}-1}\, \epsilon_1^{2^{k-1}-1}$$

D'où $\epsilon_{k+1} \sim K_k \prod_{i=1}^{k-1} K_i^{2^{k-1-i}}\, \epsilon_0^{2^{k-1}}\, \epsilon_1^{2^{k-1}}$.

ou encore $e_n \sim K_k \prod_{i=1}^{k-1} K_i^{2^{k-1-i}}\, e_{n-2}^{2^{k-1}}\, e_{n-1}^{2^{k-1}}$

Posons $K_{12\ldots k} = K_k \prod_{i=1}^{k-1} K_i^{2^{k-1-i}}$, $p = 2$, $p_1 = 2^{k-1}$, $p_2 = 2^{k-1}$.

D'après la propriété 0 (Chapitre II) ; l'ordre r de cette méthode doit satisfaire l'équation :

$$P_2(r) = r^2 - p_1 r - p_2 = 0$$

Soit $\Delta' = 2^{k-2}.\, 2^{k-2} + 2^{k-1} = 2^{k-2}(2^{k-2} + 2)$. On a deux racines distinctes.

$r' = 2^{k-2} + \sqrt{\Delta'} > 0$

et $r'' = 2^{k-2} - \sqrt{\Delta'} < 0 \Longleftrightarrow 2^{k-2} < \sqrt{\Delta'} \Longleftrightarrow 2^{k-2} \cdot 2^{k-2} < 2^{k-2}(2^{k-2}+2)$

$\Longleftrightarrow 2^{k-2} < 2^{k-2} + 2 \Longleftrightarrow 0 < 2.$

Donc l'ordre r est donné par

$$r = 2^{k-2} + \sqrt{2^{k-2}(2^{k-2}+2)}.$$

On remarque que pour $k = 1$, r est réduit à celui de la méthode de la sécante E(1) : $r \simeq 1.618$. □

Puisque chaque E(i) nécessite une évaluation de fonction par itération, cette méthode exige k évaluations.

Soit $I(E_{1\ldots k})$ son indice d'efficacité.

$$I(E_{1\ldots k}) = \left[2^{k-2} + \sqrt{2^{k-2}(2^{k-2}+2)}\right]^{\frac{1}{k}}.$$

Soit la suite $(I(E_{1\ldots k}))_{k\in\mathbb{N}^*}$. On a :

$$I(E_{1\ldots k}) < 2$$
$$I(E_{1\ldots k}) \to 2.$$

En effet : $I(E_{1\ldots k}) < 2 \Longleftrightarrow 2^{k-2} + \sqrt{2^{k-2}(2^{k-2}+2)} < 2^k$

$\Longleftrightarrow \sqrt{2^{k-2}(2^{k-2}+2)} < 2^{k-2} \times 3$

$\Longleftrightarrow 2^{k-2}(2^{k-2}+2) < 9\ 2^{k-2}\ 2^{k-2}$

$\Longleftrightarrow 2^{k-2} + 2 < 9\ 2^{k-2}$

$\Longleftrightarrow 2 < 2^3\ 2^{k-2} = 2^{k+1}.$

$$I(E_{1\ldots k}) = \exp\left[\frac{k-2}{k}\,\mathrm{LOG}2 + \frac{\mathrm{LOG}\left(1 + \sqrt{1 + \frac{1}{2^{k-3}}}\right)}{k}\right]$$

$$\to \exp(\mathrm{LOG}2) = 2,\ k \to \infty.\ \square$$

Donnons les indices d'efficacité et les ordres de convergence des méthodes E_{12} et E_{123}.

Notons $r_{1,2}$ et $r_{1,2,3}$ les ordres respectifs de ces deux méthodes.

On a :

$$r_{1,2} = 1 + \sqrt{3} \simeq 2.732$$

$$r_{1,2,3} = 2(1 + \sqrt{2}) \simeq 4.828$$

$$I(E_{12}) = (1 + \sqrt{3})^{\frac{1}{2}} \simeq 1.653$$

$$I(E_{123}) = (2(1 + \sqrt{2}))^{\frac{1}{3}} \simeq 1.690$$

II.1.5. Autres méthodes composées

1) *Méthode* $E(2) \circ E(1) \circ \phi$

Soit ϕ une fonction itérative pour résoudre $f(x) = 0$.

La $n^{\text{ième}}$ itération : x_0 donné

$$\begin{cases} u_0 = x_{n-1} \\ u_1 = \phi(u_0) \\ E_1 = E(1)\ (u_0,\ u_1) \\ E_2 = E(2)\ (u_0,\ u_1,\ E_1) \\ x_n = E_2. \end{cases}$$

D'après l'étude précédente on a :

$\epsilon_i = u_i - x_* $ et $e_n = x_n - x_*$

$e_n \sim K_2(E_1 - x_*)\ \epsilon_1\ \epsilon_0$

$E_1 - x_* \sim K_1\ \epsilon_0\ \epsilon_1$

$\epsilon_1 \sim K_0\ \epsilon_0$ où $K_0 = \phi'_*$.

On obtient :

$$e_n \sim K_2\ K_1\ K_0^2\ e_{n-1}^4.$$

C'est une méthode d'ordre 4 exigeant 3 évaluations de fonctions.

Son indice d'efficacité

$I(E(2) \circ E(1) \circ \phi)$ est donnée par :

$I((E(2) \circ E(1) \circ \phi) = (4)^{\frac{1}{3}} \simeq 1.587$

2) *Composition de E_{12} avec elle-même*

Cette méthode consiste à composer la méthode E_{12} avec elle-même.

Notons cette composée $E_{12}^2 = E_{12} \circ E_{12}$.

On se donne x_0 et x_1 et on pose $z_0 = x_0$.

$n^{\text{ième}}$ itération :

$$\begin{cases} z_{n-1} = E_{12}(x_{n-1}, z_{n-2}) \\ x_n = E_{12}(z_{n-1}, x_{n-1}) \end{cases}$$

L'étude faite sur E_{12} nous permet d'écrire :

(i) : $e_n \sim K_2\, K_1\, \epsilon_{n-1}^2\, e_{n-1}^2$ avec $\epsilon_{n-1} = z_{n-1}-x_*$ et $e_{n-1} = x_{n-1}-x_*$

(ii) : $\epsilon_{n-1} \sim K_2\, K_1\, e_{n-1}^2\, \epsilon_{n-2}^2$ avec $\epsilon_{n-2} = z_{n-2}-x_*$

(iii) : $e_{n-1} \sim K_2\, K_1\, e_{n-2}^2\, \epsilon_{n-2}^2$ avec $e_{n-2} = x_{n-2}-x_*$

(iii) $\Rightarrow$ (iv) : $\epsilon_{n-2}^2 \sim \dfrac{e_{n-1}}{K_2K_1\, e_{n-2}^2}$

(ii) et (iv) $\Rightarrow$ $\epsilon_{n-1} \sim e_{n-1}^3\, e_{n-2}^{-2}$ (v)

(i) et (v) $\Rightarrow$ $e_n \sim K_2\, K_1\, e_{n-2}^{-4}\, e_{n-1}^8$.

Soit $p = 2$, $p_1 = 8$, $p_2 = -4$.

La propriété 0 (Chapitre II) nous donne, pour les valeurs de p, p_1 et p_2, l'équation que doit vérifier l'ordre de cette méthode :

$$P_2(r) = r^2 - 8r + 4 = 0.$$

Cette équation admet deux racines réelles positives distinctes :

$$r_{12} = 4 + 2\sqrt{3} \simeq 7.464$$

$$r'_{12} = 4 - 2\sqrt{3} \simeq 0.54.$$

Or pour une suite convergente (x_n) de limite x_*, admettant un ordre r, celui-ci ne peut pas être strictement inférieur à 1.

Par conséquent : $r_{12} \simeq 7.464$ est l'ordre cherché de la méthode E_{12}^2.

La méthode E_{12}^2 exige 4 évaluations de fonctions pour un ordre $r_{12} \simeq 7.464$. Donc son indice d'efficacité $I(E_{12}^2)$ est donné par :

$$I(E_{12}^2) = r_{12}^{\frac{1}{4}} = (4 + 2\sqrt{3})^{\frac{1}{4}} \simeq 1.68$$

3) *Composition de* E_{123} *avec elle-même*

Notons cette méthode par E^2_{123}.

On se donne x_0 et x_1 et on pose $z_0 = x_0$.

Sa $n^{\text{ième}}$ itération s'effectue de la façon suivante :

$$\begin{cases} z_{n-1} = E_{123}(x_{n-1}, z_{n-2}) \\ x_n = E_{123}(z_{n-1}, x_{n-1}) \end{cases}$$

L'étude faite sur E_{123} nous permet d'écrire :

(i) : $e_n \sim K_3\, K_2\, K_1^2\, e^4_{n-1}\, \epsilon^4_{n-1}$ ou $e_i = x_i - x_*$ et $\epsilon_i = z_i - x_*$.

(ii) : $\epsilon_{n-1} \sim K_3\, K_2\, K_1^2\, e^4_{n-1}\, \epsilon^4_{n-2}$

(iii) : $e_{n-1} \sim K_3\, K_2\, K_1^2\, e^4_{n-2}\, \epsilon^4_{n-2}$

(iii) $\Rightarrow$ (iv) $\epsilon^4_{n-2} \sim (K_3\, K_2\, K_1^2)^{-1}\, e_{n-1}\, e^{-4}_{n-2}$

(ii) et (iv) $\Rightarrow$ (v) $\epsilon_{n-1} \sim e^5_{n-1}\, e^{-4}_{n-2}$

(i) et (v) $\Rightarrow$ $e_n \sim K_3\, K_2\, K_1^2\, e^{24}_{n-1}\, e^{-16}_{n-2}$.

D'où par utilisation de la propriété 0 :

$$P_2(r) = r^2 - 24r + 16 = 0$$

C'est l'équation que doit satisfaire l'ordre $r^{(2)}_{1,2,3}$ de E^2_{123}.

On obtient la valeur

$$r^{(2)}_{1,2,3} = 12 + 8\sqrt{2} \simeq 23.312.$$

Car l'autre racine de P_2 est comprise strictement entre 0 et 1.

E_{123}^2 exige 6 évaluations de fonctions par itération.

$$I(E_{123}^2) = (r_{1,2,3}^{(2)})^{\frac{1}{6}} \simeq 1.714.$$

On donnera un exemple numérique de chacune de ces méthodes à la fin de ce chapitre.

II.2. - METHODE D'INTERPOLATION RATIONNELLE INVERSE (Cas d'une racine simple).

II.2.1. - Méthode de Type-Steffensen

1) T(1)

T(1) est la méthode itérative correspondant à la colonne $(\rho_2^{(n)})_{n\in\mathbb{N}}$ du procédé (E'). Elle est destinée à résoudre : $x = \phi(x) = x - f(x)$.

Etant donné x_0, sa $n^{\text{ième}}$ itération se déroule de la manière suivante :

$n^{\text{ième}}$ itération

$$\left[\begin{array}{l}
u_0 = x_{n-1},\ u_1 = \phi(u_0),\ u_2 = \phi(u_1),\ u_3 = \phi(u_2). \\[1em]
\rho_{-1}^{(i)} = 0 \text{ et } \rho_0^{(i)} = u_i \text{ pour } i = 0, 1, 2. \\[1em]
\rho_j^{(i)} = \rho_{j-2}^{(i+1)} + \dfrac{\dfrac{1}{\Delta u_i} - \dfrac{1}{\Delta u_{i+j}}}{\rho_{j-1}^{(i+1)} - \rho_{j-1}^{(i)}} \quad \begin{array}{l} j = 1, 2 \\ i = 0,\ldots, 2-j \end{array} \\[1em]
x_n = \rho_2^{(0)}.
\end{array}\right.$$

Posons $e_n = x_n - x_*$, $e_{n-i} = x_{n-i} - x_*$ et $\epsilon_i = u_i - x_*$ pour $i = 0, 1, 2, 3$.

Comme $f(u_i) = -\Delta u_i$, alors on peut appliquer la propriété 10 (Chapitre I).

On obtient :

$$e_n \sim \frac{(f'_*)^3}{6} D_2 \, \epsilon_0 \, \epsilon_1 \, \epsilon_2$$

Or $\epsilon_0 = e_{n-1}$, $\epsilon_1 \sim \phi'_* \, \epsilon_0$ et $\epsilon_2 \sim \phi'^2_* \, \epsilon_0$

On a donc :

$$e_n \sim \frac{(f'_*)^3}{6} \phi'^3_* \, D_2 \, e^3_{n-1}.$$

Dans ce cas simple on va déterminer le coefficient D_2

D_2 est donné par l'expression :

$$D_2^{(n)} = \frac{d^3}{d\tau^3} \left[F(\tau) \, [Q_1^{(0)}(\tau)]^2\right]\Big|_{\tau=0} \text{ et } D_2 = \lim_{n\to\infty} D_2^{(n)}.$$

F étant l'inverse de f et $Q_1^{(0)}(t)$ étant le dénominateur de la fraction $\dfrac{P_1^{(0)}(t)}{Q_1^{(0)}(t)}$ interpolant F en u_0, u_1, u_2.

On a $\dfrac{P_1^{(0)}(t)}{Q_1^{(0)}(t)} = \dfrac{a_0 t + a_1}{b_0 t + 1}$ tels que :

$$\begin{cases} f_0 \, a_0 + a_1 - u_0 \, f_0 \, b_0 = u_0 \\ f_1 \, a_0 + a_1 - u_1 \, f_1 \, b_0 = u_1 \Rightarrow b_0 = -\dfrac{\Delta u_1 \Delta f_0 - \Delta u_0 \Delta f_1}{\Delta(u_1 f_1)\Delta f_0 - \Delta(u_0 f_0)\Delta f_1} \\ f_2 \, a_0 + a_1 - u_2 \, f_2 \, b_0 = u_2 \end{cases}$$

Posons $h_i = u_i \, f_i$ et $h(x) = x \, f(x)$.

$$b_0 = -\frac{\Delta u_1 \Delta f_0 - \Delta u_0 \Delta f_1}{\Delta h_1 \Delta f_0 - \Delta h_0 \Delta f_1}$$

$$f_i = c_1\,\epsilon_i + c_2\,\epsilon_i^2 + \dots \qquad \text{avec } c_i = \frac{f_*^{(i)}}{i!}$$

$$h_i = a_1\,\epsilon_i + a_2\,\epsilon_i^2 + \dots \qquad a_i = \frac{h_*^{(i)}}{i!} = \frac{x_*\,f_*^{(i)} + i\,f_*^{(i-1)}}{i!}$$

$$= x_*\,c_i + c_{i-1}$$

$$\Delta u_1 \Delta f_0 - \Delta u_0 \Delta f_1 = (\epsilon_2-\epsilon_1)(\epsilon_1-\epsilon_0)(c_2\,(\epsilon_0-\epsilon_1) + \dots)$$

$$\Delta h_1 \Delta f_0 - \Delta h_0 \Delta f_1 = (\epsilon_2-\epsilon_1)(\epsilon_1-\epsilon_0)((a_1\,c_2 - c_1\,a_2)(\epsilon_0-\epsilon_1) + \dots)$$

D'où, en négligeant les puissances de ϵ_2 et ϵ_0 :

$$b_0 \sim -\frac{c_2}{a_1\,c_2 - c_1\,a_2} = \frac{f''_*}{2f'^2_*}.$$

Or $[F(\tau)(b_0\tau+1)^2]^{(3)} = F^{(3)}(\tau)(b_0\tau+1)^2 + 6b_0\,F^{(2)}(\tau)(b_0\tau+1) + 6b_0^2\,F'(\tau)$

$$\Rightarrow D_2 = F^{(3)}(0) + 6b_0\,F^{(2)}(0) + 6\,b_0^2\,F'(0)$$

$$\sim F^{(3)}(0) + 3\,\frac{f''_*}{f'^2_*}\,F^{(2)}(0) + \frac{3}{2}\,\frac{f''^2_*}{f'^4_*}\,F'(0)$$

Ainsi le coefficient asymptotique d'erreur de T(1) est donné par :

$$C = \left[\frac{(f'_*)^3}{6}\,F^{(3)}(0) + \frac{1}{2}\,f'_*\,f''_*\,F^{(2)}(0) + \frac{1}{4}\,\frac{f''^2_*}{f'_*}\,F'(0)\right]\,(\phi'_*)^3.$$

T(1) est donc une méthode d'ordre 3, exigeant trois évaluations de fonctions par itération. Et par conséquent son indice d'efficacité est :

$$I(T(1)) = 3^{\frac{1}{3}} \simeq 1.442$$

T(1) est donc comparable à la méthode S(2) vue au paragraphe I.1.1., 2).

2) *Méthode* T(k)

Elle correspond à la suite colonne $(\rho_{2k}^{(n)})_{n\in\mathbb{N}}$ du procédé (E').

Etant donné x_0, sa $n^{\text{ième}}$ itération est donnée par :

$$\left[\begin{array}{l} u_0 = x_{n-1},\ u_1 = \phi(u_0),\ldots,\ u_{2k} = \phi(u_{2k-1}),\ u_{2k+1} = \phi(u_{2k}) \\[2mm] \rho_{-1}^{(i)} = 0,\ \rho_0^{(i)} = u_i \text{ pour } i = 0,\ 1,\ldots,\ 2k. \\[2mm] \rho_j^{(i)} = \rho_{j-2}^{(i+1)} + \dfrac{\dfrac{1}{\Delta u_i} - \dfrac{1}{\Delta u_{j+1}}}{\rho_{j-1}^{(i+1)} - \rho_{j-1}^{(i)}},\ \begin{array}{l} j = 1,\ 2,\ldots,\ 2k \\ i = 0,\ldots,\ 2k-j \end{array} \\[2mm] x_n = \rho_{2k}^{(0)}. \end{array}\right.$$

D'après l'étude faite sur le procédé (E) et en tenant compte du fait que $\phi(x) = x - f(x)$ on a :

$$e_n \sim \frac{(f'_*)^{2k+1}}{(2k+1)!} \quad \phi'^{\,k(2k+1)}_* \; D_{2k}\; e_{n-1}^{2k+1}.$$

On voit que T(k) est comparable à S(2k), vue au paragraphe (I.1.1., 3)), car elles ont le même ordre de convergence et le même indice d'efficacité.

II.2.2. - Méthodes de type-sécante

1) *Méthode* F(1)

Elle correspond à la suite colonne $(\rho_2^{(n)})_{n\in\mathbb{N}}$ du procédé (E).

Après avoir choisi x_0, x_1, x_2 ; la $n^{\text{ième}}$ itération de F(1) se déroule de la manière suivante :

$n^{\text{ième}}$ itération : $$\left[\begin{array}{l} u_0 = x_{n-3},\ u_1 = x_{n-2},\ u_2 = x_{n-1} \\[2mm] \rho_{-1}^{(i)} = 0,\ \rho_0^{(i)} = u_i \quad i = 0,\ 1,\ 2. \\[2mm] \rho_j^{(i)} = \rho_{j-2}^{(i+1)} + \dfrac{\dfrac{1}{f_{i+j}} - \dfrac{1}{f_i}}{\rho_{j-1}^{(i+1)} - \rho_{j-1}^{(i)}},\ \begin{array}{l} j = 1,\ 2. \\ i = 0,\ldots,\ 2-j. \end{array} \\[2mm] x_n = \rho_2^{(0)}. \end{array}\right.$$

A partir de l'étude faite sur (E) on conclut :

$$e_n \sim \frac{(f'_*)^3}{6} D_2\, e_{n-3}\, e_{n-2}\, e_{n-1}$$

avec $D_2 = \lim\limits_n \frac{d^3}{d\tau^3}\left[F(\tau)\ (Q_1^{(0)}(\tau))^2\right]_{\tau=0} = F^{(3)}(0) + 3\,\frac{f''_*}{f'^2_*}\ F''(0) + \frac{3}{2}\,\frac{f''^2_*}{f'^4_*} F'(0)$

où F est l'inverse de f.

Comme nous avons vu au paragraphe précédent, la méthode F(1) est comparable à la méthode E(2). En effet, F(1) et E(2) exigent chacune une seule évaluation de fonctions par itération. De plus elles ont le même ordre et le même indice d'efficacité.

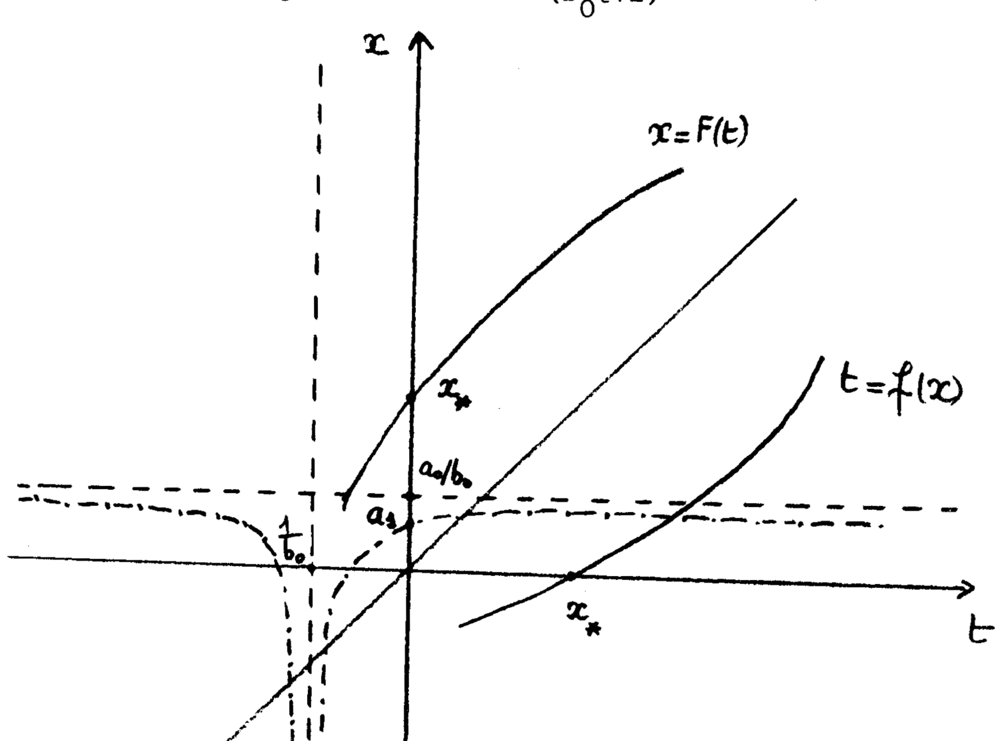
Désignons par r_2 l'ordre de F(1), C_1 (respectivement K_1) son coefficient asymptotique (respectivement dominant) d'erreur. On a :

$$K_1 = \frac{f'^3_*}{6}\, F^{(3)}(0) + \frac{1}{2}\, f'_*\, f''_*\, F^{(2)}(0) + \frac{1}{4}\,\frac{f''^2_*}{f'_*}\, F'(0) = \left(\frac{1}{2}\,\frac{f''_*}{f'_*}\right)^2 - \frac{f^{(3)}_*}{f'_*}.$$

$$C_1 = K_1^{\left(\frac{1}{r_2^2-1}\right)}$$

L'interprétation géométrique de F(1) est la suivante :

Posons $\phi(t) = \dfrac{a_0 t + a_1}{b_0 t + 1} \Rightarrow \phi'(t)\ \dfrac{a_0 - b_0 a_1}{(b_0 t+1)^2}$



Donc il s'agit d'une interpolation par une fonction homographique, puis d'une extrapolation en 0 :

$$\phi(0) = a_1 \simeq x_*.$$

2) *Méthode* F(k)

Elle correspond à la suite colonne $(\rho_{2k}^{(n)})_{n \in \mathbb{N}}$ du procédé (E).

Après avoir choisi $x_0, x_1, \ldots, x_{2k}$; la $n^{\text{ième}}$ itération de F(k) se déroule de la façon suivante :

$n^{\text{ième}}$ itération :

$$\begin{cases} u_i = x_{n-1-2k+i},\ i = 0, 1, \ldots, 2k. \\ \rho_{-1}^{(i)} = 0,\ \rho_0^{(i)} = u_i \quad i = 0, 1, \ldots, 2k \\ \rho_j^{(i)} = \rho_{j-2}^{(i+1)} + \dfrac{\dfrac{1}{f_{i+j}} - \dfrac{1}{f_i}}{\rho_{j-1}^{(i+1)} - \rho_{j-1}^{(i)}},\ j = 1, \ldots, 2k \quad i = 0, \ldots, 2k-j \\ x_n = \rho_{2k}^{(0)} \\ \text{où } f_i = f(u_i) \end{cases}$$

L'étude faite sur le procédé (E) nous permet de conclure :

$$e_n \sim \frac{(f_*')^{2k+1}}{(2k+1)!} D_{2k} \prod_{i=0}^{2k} e_{n-i-1} \qquad \text{où } e_n = x_n - x_*.$$

Cette erreur est de la forme étudiée dans la propriété 0 (Chapitre II).

On trouve donc que r existe et doit vérifier l'équation :

$$P_{2k+1}(r) = r^{2k+1} - \sum_{i=0}^{2k} r^{2k-i} = 0.$$

On donnera à la fin de ce chapitre quelques exemples.

A part la première itération exigeant 2k+1 évaluations de fonctions, les autres n'en demandent qu'une seule. Donc l'indice d'efficacité est égal à l'ordre.

De plus la méthode F(k) est comparable à la méthode E(2k), car elles ont le même ordre et le même indice d'efficacité.

Remarque :

Comme pour le paragraphe II.1. nous pourrons composer les méthodes T(k) et F(k) entre-elles pour en construire d'autres. Le travail à effectuer est toujours le même, car le développement asymptotique d'erreur se présente toujours sous la forme étudiée dans la propriété 0 (Chapitre II), à savoir :

$$e_n \sim K \sum_{i=0}^{p} e_{n-i}^{p_i}.$$

II.3. - COMPARAISON AVEC D'AUTRES METHODES DEJA CONNUES

Nous allons faire un rappel sur certaines méthodes de résolution des équations non linéaires afin de pouvoir les comparer aux méthodes que nous venons d'étudier.

II.3.1. - Rappels sur certaines méthodes

1) *Méthode de Newton*

Soit à résoudre

$$f(x) = 0$$

$$\Longleftrightarrow x = \phi(x) \text{ où } \phi(x) = x-f(x)$$

Soit x_* une racine simple de f. Nous avons :

$$x_* = \phi(x_*) = x_*-f(x_*) = x_* - \{f(x) + f'(x)(x_*-x) + \frac{f''(x)}{2!}(x_*-x)^2 + \ldots\}$$

$$= x_*-f(x) - f'(x)(x_*-x) - \frac{f''(x)}{2!}(x_*-x)^2 - \ldots$$

$$\Rightarrow f(x) = -\,[f'(x)(x_*-x) + \frac{f''(x)}{2!}(x_*-x)^2 + \ldots]$$

Dans le second membre, négligeons les termes en $(x_*-x)^i$ pour $i \geq 2$. Il vient donc :

$$f(x) = -\,f'(x)(x_*-x)$$

$$\Rightarrow x_* = x - f(x)/f'(x).$$

La méthode de Newton consiste donc à itérer sur la formule précédente. D'où :

$$\begin{cases} x_0 \text{ donné} \\ x_{n+1} = x_n - f_n/f'_n, \; n = 0, 1, 2, \ldots \end{cases}$$

L'erreur $e_{n+1} = x_{n+1}-x_*$ est telle que :

$$e_{n+1} \sim \frac{1}{2}\frac{f''_*}{f'_*}\, e_n^2 \text{ où } f'_* = f'(x_*) \text{ et } f''_* = f''(x_*).$$

En effet :

$$f_n = \sum_{i=1}^{\infty} c_i\, e_n^i,\; c_i = \frac{f_*^{(i)}}{i!}$$

$$\Rightarrow f'_n = \sum_{i=1}^{\infty} i\, c_i\, e_n^{i-1}$$

$$e_{n+1} = e_n - \frac{f_n}{f'_n} = e_n - \frac{\sum\limits_{i=1}^{\infty} c_i\, e_n^i}{\sum\limits_{i=1} i c_i\, e_n^{i-1}} = e_n - e_n \frac{C_1 + C_2\, e_n + C_3\, e_n^2 + \ldots}{C_1 + 2C_2\, e_n + 3C_3\, e_n^2 + \ldots}$$

$$= e_n - e_n \{1 - \frac{C_2}{C_1} e_n + 2\, (\frac{C_2 - C_1\, C_3}{C_1^2})\, e_n^2 + \ldots\}$$

$$= \frac{C_2}{C_1} e_n^2 - 2\, (\frac{C_2 - C_1\, C_3}{C_1^2})\, e_n^3 + \ldots$$

Pour obtenir e_{n+1} sous cette dernière forme, il suffit de diviser la série donnant f_n par celle donnant f'_n suivant l'ordre croissant des puissances de e_n. D'où l'on tire si $f'_* \neq 0$

$$e_{n+1} \sim \frac{1}{2} \frac{f''_*}{f'_*}\; e_n^2.$$

Puisque cette méthode exige deux évaluations de fonctions f et f' par itération, son indice d'efficacité est : $\sqrt{2} = 1.414\ldots$

2) *Méthode d'extrapolation de King*

King [16] a proposé une méthode d'extrapolation d'ordre quatre, nécessitant trois évaluations de fonction par itération. Elle est basée essentiellement sur le procédé Δ^2-d'Aitken.

Soit à résoudre l'équation de point fixe :

$$x = \phi(x).$$

Supposons que ϕ est suffisamment dérivable dans un voisinage de x_* et que $|\phi'_*| < 1$.

Nous nous proposons de construire la méthode de King.

Supposons que l'on ait construit x_n. Prenons :

$$x_{n+1} = \phi(x_n)$$
$$x_{n+2} = \phi(x_{n+1}).$$

Appliquons le procédé Δ^2-d'Aitken à $\{x_n, x_{n+1}, x_{n+2}\}$. On a :

$$\overline{x_{n+2}} = x_{n+1} - \frac{1}{1-K_1}(x_{n+1}-x_{n+2}),$$

où $K_1 = \dfrac{\Delta x_{n+1}}{\Delta x_n}$.

Si $e_i = x_i - x_*$ pour $i = n, n+1, n+2$ et si $\overline{e_{n+2}} = \overline{x_{n+2}} - x_*$, alors nous pouvons développer $\overline{e_{n+2}}$ en série. En effet :

$$e_{n+1} = K\,e_n + L\,e_n^2 + M\,e_n^3 + N\,e_n^4 + \ldots,$$

où $K = \dfrac{\phi'_*}{1!}$, $L = \dfrac{\phi''_*}{2!}$, $M\ \dfrac{\phi_*^{(3)}}{3!}$, $N = \dfrac{\phi_*^{(4)}}{4}$, $\phi_*^{(i)} = \phi^{(i)}(x_*)$.

$$e_{n+2} = K\,e_{n+1} + L\,e_{n+1}^2 + M\,e_{n+1}^3 + N\,e_{n+1}^4 + \ldots$$
$$= K^2\,e_n + L(K^2+K)e_n^2 + [L^2(2K)+M(K^3+K)]e_n^3 + [L^3+ML(3K^2+2K)+N(K^4+K)]e_n^4$$
$$\overline{e_{n+2}} = ([1-K]^{-1}\,[L(-K)])e_n^2 + ([1-K]^{-2}\,\,[L^2(-K^2-1) + M(K^3-K)])e_n^3 +$$
$$\{(1-K)^{-3}[L^3(-K^4-K^3-K^2-K) + ML(2K^5-K^4+K^3-2K) + N(-K^6+K^5+K^3-K^2)]\}e_n^4$$
$$= \overline{K}\,e_n^2 + \overline{L}\,e_n^3 + \overline{M}\,e_n^4 + \ldots$$

Maintenant construisons x_{n+3} par application de ϕ à $\overline{x_{n+2}}$.

$$x_{n+3} = \phi(\overline{x_{n+2}})$$

Nous avons, si $e_{n+3} = x_{n+3} - x_*$:

$$e_{n+3} = K\,\overline{e_{n+2}} + L\,\overline{e_{n+2}}^2 + M\,\overline{e_{n+2}}^3 + \ldots$$
$$= K(\overline{K}\,e_n^2 + \overline{L}\,e_n^3 + \overline{M}\,e_n^4) + L(\overline{K}\,e_n^2 + \ldots)^2 + \ldots$$
$$= K(\overline{K}\,e_n^2 + \overline{L}\,e_n^3) + (K\,\overline{M} + L\,\overline{K}^2)\,e_n^4 + \ldots$$

Donc, nous nous apercevons que $\overline{e_{n+2}}$ et e_{n+3} coïncident jusqu'à l'ordre trois compris au coefficient multiplicatif près K.

Soit $\tilde{K}$ une approximation connue de K, d'ordre $\alpha \geq 2$.

$$\tilde{K} = K + O(e_n^{\alpha}).$$

la combinaison :

$$\tilde{x}_{n+3} = \overline{x_{n+2}} - \left(\frac{1}{1-\tilde{K}}\right)(\overline{x_{n+2}} - x_{n+3}),$$

est telle que ; si $\tilde{e}_{n+3} = \tilde{x}_{n+3} - x_*$:

$$\tilde{e}_{n+3} = O(e_n^4).$$

En effet, nous avons :

$$\tilde{e}_{n+3} = \overline{e_{n+2}} - \left(\frac{1}{1-\tilde{K}}\right)(\overline{e_{n+2}} - e_{n+3})$$

$$= \overline{e_{n+2}} - \left(\frac{1}{1-K+O(e_n^{\alpha})}\right)((1-K)(\overline{K}\, e_n^2 + \overline{L}\, e_n^3) + O(e_n^4))$$

$$= \overline{e_{n+2}} - \left(\frac{1}{1-K} + O(e_n^{\alpha})\right)\ ((1-K)(\overline{K}\, e_n^2 + \overline{L}\, e_n^3) + O(e_n^4))$$

$$= \overline{K}\, e_n^2 + \overline{L}\, e_n^3 + O(e_n^4) - \{\overline{K}\, e_n^2 + \overline{L}\, e_n^3 + O(e_n^4)\}$$

$$= O(e_n^4).$$

Cette idée est à la base de la méthode de King.

Construisons maintenant une approximation $\tilde{K}$ à l'aide des termes de la suite x_n, x_{n+1}, x_{n+2}, $\overline{x_{n+2}}$, x_{n+3}.

D'abord nous examinons le rapport :

$$K_1 = \frac{\Delta x_{n+1}}{\Delta x_n} = \frac{\Delta e_{n+1}}{\Delta e_n} = K + L(1+K)\ e_n + \ldots$$

Or K_1 peut aussi s'écrire :

$$K_1 = \frac{\phi_{n+1}-\phi_n}{x_{n+1}-x_n}.$$

Remplaçons dans cette dernière expression de K_1, x_n par x_{n+1} et x_{n+1} par $\overline{x_{n+2}}$. Nous avons le rapport :

$$K_2 = \frac{\phi(\overline{x_{n+2}})-\phi_{n+1}}{\overline{x_{n+2}} - x_{n+1}} = \frac{x_{n+3}-x_{n+2}}{x_{n+2}-x_{n+1}} = \frac{e_{n+3}-e_{n+2}}{e_{n+2}-e_{n+1}}$$

$$\Longrightarrow K_2 = K + LK\, e_n + \ldots$$

Nous venons d'obtenir deux approximations de K d'ordre un.

Multiplions K_1 par K et K_2 par 1+K. Il vient :

$$KK_1 = K \quad K + LK(1+K)e_n + O(e_n^2)$$

$$(1+K)K_2 = (1+K)K + LK(1+K)e_n + O(e_n^2)$$

$$\Longrightarrow KK_1 - (1+K)K_2 = - K + O(e_n^2)$$

où encore :

$$K_2(1+K) - K_1K = K + O(e_n^2).$$

Mais K est toujours inconnu. Nous pouvons mettre à sa place K_2, d'où l'approximation :

$$\hat{K} = K_2(1+K_2) - K_1K_2 = K_2(1+K_2-K_1).$$

$\hat{K}$ est d'ordre deux. En effet :

$$\hat{K} = K_2(1+K_2-K_1) = (K+LKe_n +\ldots)(1+K+LKe_n +\ldots- K-L(1+K)e_n -\ldots)$$

$$= (K+LKe_n + O(e_n^2))(1 - Le_n + O(e_n^2))$$

$$= K - LKe_n + LKe_n + O(e_n^2)$$

$$= K + O(e_n^2).$$

Par conséquent la combinaison :

$$\overline{x_{n+3}} = \overline{x_{n+2}} - \left(\frac{1}{1-\hat{K}}\right)\left(\overline{x_{n+2}} - x_{n+3}\right),$$

est telle que, $\overline{e_{n+3}} = \overline{x_{n+3}} - x_*$:

$$\overline{e_{n+3}} = O(e_n^4).$$

Plus précisément nous obtenons l'erreur $\overline{e_{n+3}}$ sous la forme :

$$\overline{e_{n+3}} \sim \frac{L^3(-2K^2+K) + ML(K^3-K^2)}{(1-K)^3} e_n^4.$$

En effet, développons K_1 et K_2 jusqu'à l'ordre 3.

$$K_1 = K + L(1+K)e_n + (L^2+M(K^2+K+1))e_n^2 + [ML(2K+2) + N(K^3+K^2+K+1)]e_n^3.$$

$$K_2 = K + LK\, e_n + (1-K)^{-1}\,[L^2(-2K+1) + M(-K^3+K^2)]e_n^2 +$$

$$[1-K]^{-2}\,[L^3(-K-1) + ML(4K^3-4K^2-K+1) + N(K^5-2K^4+K^3)]e_n^3.$$

Par conséquent $\hat{K}$ peut s'écrire :

$$\hat{K} = K_2(1+K_2-K_1) = K+(1-K)^{-1}\,[L^2(-3K+1) + M(K^2-K)]e_n^2 + \ldots$$

$$= K+(1-K)^{-1}\, A\, e_n^2 + \ldots$$

D'où :

$$\overline{e_{n+3}} = \overline{e_{n+2}} - \frac{1}{1-\hat{K}}\left(\overline{e_{n+2}} - e_{n+3}\right)$$

$$= \overline{K}\, e_n^2 + \overline{L}\, e_n^3 + \overline{M}\, e_n^4 + \ldots - \left(\frac{1}{(1-K)-(1-K)^{-1}Ae_n^2 + \ldots}\right)\left((1-K)(\overline{K}e_n^2 + \overline{L}e_n^3)\right.$$

$$\left. + (\overline{M}(1-K) - L\overline{K}^2)e_n^4 + \ldots\right)$$

$$= \overline{K}\, e_n^2 + \overline{L}\, e_n^3 + \overline{M}\, e_n^4 - \left(\frac{1}{1-K}\right)\left(1 + \frac{A}{(1-K)^2}\, e_n^2 + \ldots\right)$$

$$\left((1-K)(\overline{K}e_n^2 + \overline{L}e_n^3) + (\overline{M}(1-K) - L\overline{K}^2)\, e_n^4 + \ldots\right)$$

$$= [\overline{M} - \frac{1}{1-K}(\overline{M}(1-K) - L\overline{K}^2) - \frac{1}{1-K}(\frac{A\overline{K}}{1-K})]e_n^4 + \ldots$$

$$= (\frac{L\overline{K}^2}{1-K} - \frac{A\overline{K}}{(1-K)^2})e_n^4 + \ldots$$

$$= \frac{(L(1-K)\overline{K}-A)\overline{K}}{(1-K)^2} e_n^4 + \ldots$$

Or on a :

$$\overline{K} = \frac{L(-K)}{1-K} \text{ et } A = L^2(-3K+1) + M(K^2-K)$$

$$\Rightarrow L(1-K)\,\overline{K} - A = L^2(-K) - L^2(-3K+1) - M(K^2-K)$$

$$= L^2(2K-1) - M(K^2-K)$$

$$\text{et } \overline{K}(L(1-K)\,\overline{K} - A) = \frac{1}{1-K}(L^3(-2K^2+K) + LM(K^3-K^2))$$

D'où l'on tire :

$$\overline{e_{n+3}} \sim \frac{L^3(-2K^2+K) + LM(K^3-K^2)}{(1-K)^3} e_n^4.$$

La méthode ainsi construite se résume de la manière suivante :

x_0 donné

<u>$n^{\text{ième}}$ itération</u>

$$\left[\begin{array}{l}
u_0 = x_{n-1} \\
u_1 = \phi(u_0) \\
u_2 = \phi(u_1) \\
\overline{u_2} = u_1 - (\frac{1}{1-K_1})(u_1-u_2) \text{ où } K_1 = \frac{\Delta u_1}{\Delta u_0} \\
u_3 = \phi(\overline{u_2}) \\
\overline{u_3} = \overline{u_2} - (\frac{1}{1-\hat{K}})(\overline{u_2} - u_3) \text{ où } \hat{K} = K_2(1+K_2-K_1) \text{ avec } K_2 = \frac{\Delta u_2}{\overline{u_2}-u_1}. \\
x_n = \overline{u_3}
\end{array}\right.$$

Pour construire x_n il faut évaluer ϕ trois fois. Comme l'ordre est quatre, l'indice d'efficacité de cette méthode est donc égal à :

$$\sqrt[3]{4} \simeq 1.587.$$

3) *Méthode de Jarratt* [13]

Soit à résoudre $f(x) = 0$.

Soient x_{n-1} et x_n deux points voisins de x_*.

On considère la fraction :

$$z(x) = \frac{x-a}{bx^2 + cx + d}$$

où a, b, c et d sont déterminés par les conditions :

$$\begin{aligned} f(x_{n-1}) &= z(x_{n-1}) \\ f(x_n) &= z(x_n) \\ f'(x_{n-1}) &= z'(x_{n-1}) \\ f'(x_n) &= z'(x_n) \end{aligned}$$

Soit $\phi = x-x_n$, et considérons la fonction :

$$g(\phi) = f(\phi + x_n) = f(x).$$

On a aussi :

$$z(\phi + x_n) = \frac{\phi + (x_n-a)}{b\phi^2 + (2bx_n + C)\phi + d + bx_n^2 + Cx_n}$$

Posons $\alpha = a-x_n$, $\beta = b$, $\gamma = 2bx_n + C$, $\delta = d + bx_n^2 + Cx_n$, et considérons la fraction :

$$y(\phi) = \frac{\phi - \alpha}{\beta\phi^2 + \gamma\phi + \delta}$$

On a évidemment :

$$y(0) = f_n$$
$$y'(0) = f'_n$$
$$y(\phi_{n-1}) = f_{n-1}$$
$$y'(\phi_{n-1}) = f'_{n-1}$$

où $f_n = f(x_n)$, $f_{n-1} = f(x_{n-1})$, $f'_n = f'(x_n)$, $f'_{n-1} = f'(x_{n-1})$ et $\phi_{n-1} = x_{n-1}-x_n$.

Ces dernières conditions d'interpolation donnent naissance au système suivant :

$$\begin{cases} \alpha & & & + f_n \quad \delta = 0 \\ \alpha + \phi^2_{n-1} f_{n-1} \beta & & + \phi_{n-1} f_{n-1} \gamma & + f_{n-1} \delta = \phi_{n-1} \\ & & f_n \quad \gamma & + f'_n \quad \delta = 1 \\ & + (\phi^2_{n-1} f'_{n-1} + 2\phi_{n-1} f_{n-1})\beta & + (\phi_{n-1} f'_{n-1} + f_{n-1})\gamma & + f'_{n-1} \delta = 1 \end{cases}$$

Ce système nous permet de déterminer α, β, γ et δ .

Nous nous intéressons au zéro de y, à savoir α. Nous prenons

$$x_{n+1} = x_n + \alpha.$$

En effet $y(\phi) = z(\phi+x_n) = 0$

$\Longleftrightarrow \phi + x_n$ est zéro de $z(x)$,

$\phi + x_n = x$ et $z(x)$ est une approximation de $f(x)$, en ce sens qu'elle est déterminée par les conditions d'interpolation. Donc on peut approximer x_* par $\phi + x_n$ (zéro de $z(x)$). Et comme α annulle y, donc $\alpha + x_n$ annulle $z(x)$. Par suite $x_n + \alpha$ est une approximation de x_*.

Après avoir calculé α à partir du système précédent, l'itéré x_{n+1} est donné par :

$$x_{n+1} = J(x_{n-1}, x_n) = x_n - \frac{\Delta x_{n-1} f_n \quad [f_{n-1}\Delta f_{n-1} - \Delta x_{n-1} \; f_n f'_{n-1}]}{2 f_n f_{n-1} \Delta f_{n-1} - \Delta x_{n-1}(f^2_n f'_{n-1} + f^2_{n-1} f'_n)}$$

De plus $e_{n+1} \sim (\frac{C_4}{C_1} - 2\,\frac{C_2C_3}{C_1^2} + (\frac{C_2}{C_1})^3)\; e_n^2\, e_{n-1}^2$,

où $C_i = \frac{f_*^{(i)}}{i!}$.

En effet, puisque $x_{n+1} = x_n + \alpha$, alors $\alpha = x_{n+1} - x_n$ et $a = x_{n+1}$.

Considérons le système suivant de 4 inconnues a, b, c, d, formé de cinq équations :

$$\begin{array}{llll}
a & & & = x_{n+1} \\
a + b\, x_n^2 f_n & + C\, x_n f_n & + df_n & = x_n \\
a + b\, x_{n-1}^2 f_{n-1} & + C\, x_{n-1} f_{n-1} & + df_{n-1} & = x_{n-1} \\
\quad b(x_n^2 f'_n + 2x_n f_n) & + C(x_n f'_n + f_n) & + df'_n & = 1 \\
\quad b(x_{n-1}^2 f'_{n-1} + 2x_{n-1}\, f_{n-1}) & + C(x_{n-1} f'_{n-1} + f_{n-1}) & + df'_{n-1} & = 1
\end{array}$$

On note S la matrice :

$$S = \begin{bmatrix}
x_n^2 f_n & x_n f_n & f_n \\
x_{n-1}^2 f_{n-1} & x_{n-1} f_{n-1} & f_{n-1} \\
x_n^2 f'_n + 2x_n f_n & x_n f'_n + f_n & f'_n \\
x_{n-1}^2 f'_{n-1} + 2x_{n-1} f_{n-1} & x_{n-1} f'_{n-1} + f_{n-1} & f'_{n-1}
\end{bmatrix}$$

Puisque nous nous intéressons à l'existence et à l'unicité de la fraction d'interpolation z(x), alors nécessairement

$$\begin{vmatrix}
x_{n+1} & 1 & 0 & 0 & 0 \\
x_n & 1 & & & \\
x_{n-1} & 1 & & S & \\
1 & 0 & & & \\
1 & 0 & & &
\end{vmatrix} = 0.$$

$$\Rightarrow x_{n+1} = \frac{\begin{vmatrix} x_n & \\ x_{n-1} & S \\ 1 & \\ 1 & \end{vmatrix}}{\begin{vmatrix} 1 & \\ 1 & \\ 0 & S \\ 0 & \end{vmatrix}}$$

Posons $E = (e_n, e_{n-1}, 1, 1)^T$, $A = (e_n^2 f_n, e_{n-1}^2 f_{n-1}, e_n^2 f'_n + 2e_n f_n,$

$e_{n-1}^2 f'_{n-1} + 2e_{n-1} f_{n-1})^T$,

$B = (e_n f_n, e_{n-1} f_{n-1}, e_n f'_n + f_n, e_{n-1} f'_{n-1} + f_{n-1})^T$

et $C = (f_n, f_{n-1}, f'_n, f'_{n-1})^T$.

L'erreur e_{n+1} peut s'écrire :

$$e_{n+1} = \frac{\begin{vmatrix} x_n & \\ x_{n-1} & S \\ 1 & \\ 1 & \end{vmatrix}}{\begin{vmatrix} 1 & \\ 1 & S \\ 0 & \\ 0 & \end{vmatrix}} - x_* = \frac{\begin{vmatrix} e_n & \\ e_{n-1} & S \\ 1 & \\ 1 & \end{vmatrix}}{\begin{vmatrix} 1 & \\ 1 & S \\ 0 & \\ 0 & \end{vmatrix}} = \frac{\det(E, A-2x_*B + x_*^2C, B - x_* C, C)}{\det(I, A-2x_*B + x_*^2C, B - x_* C, C)}$$

avec $I = (1, 1, 0, 0)^T$.

Et en utilisant les propriétés d'un déterminant, nous obtenons :

$$e_{n+1} = \frac{\begin{vmatrix} e_n & \\ e_{n-1} & \Delta \\ 1 & \\ 1 & \end{vmatrix}}{\begin{vmatrix} 1 & \\ 1 & \Delta \\ 0 & \\ 0 & \end{vmatrix}}, \text{ où } \Delta = [A^T, B^T, C^T]$$

Le développement de Taylor de f en x_* nous permet d'écrire :

$$e_{n+1} = \frac{\sum\limits_{i,j,\,k=1}^{\infty} C_i C_j C_k \, e_n e_{n-1} \begin{vmatrix} 1 & e_n^{i+1} & e_n^j & e_n^{k-1} \\ 1 & e_{n-1}^{i+1} & e_{n-1}^j & e_{n-1}^{k-1} \\ 1 & (i+2)e_n^{i+1} & (j+1)e_n^j & ke_n^{k-1} \\ 1 & (i+2)e_{n-1}^{i+1} & (j+1)e_{n-1}^j & ke_{n-1}^{k-1} \end{vmatrix}}{\sum\limits_{p,\,q,\,r=1}^{\infty} C_p C_q C_r \begin{vmatrix} 1 & e_n^{p+2} & e_n^{q+1} & e_n^r \\ 1 & e_{n-1}^{p+2} & e_{n-1}^{q+1} & e_{n-1}^r \\ 0 & (p+2)e_n^{p+1} & (q+1)e_n^q & re_n^{r-1} \\ 0 & (p+2)e_{n-1}^{p+1} & (q+1)e_{n-1}^q & re_{n-1}^{r-1} \end{vmatrix}} = \frac{N_{n,n-1}}{D_{n,n-1}}$$

Dans les deux séries nous ne retenons que les premiers déterminants non nuls.

Le premier déterminant non nul au dénominateur est celui correspondant au triplet (p, q, r) = (1, 1, 1).

$$D_{n,n-1} \sim C_1^3 \begin{vmatrix} 1 & e_n^3 & e_n^2 & e_n \\ 1 & e_{n-1}^3 & e_{n-1}^2 & e_{n-1} \\ 0 & 3\,e_n^2 & 2\,e_n & 1 \\ 0 & 3\,e_{n-1}^2 & 2\,e_{n-1} & 1 \end{vmatrix}$$

Au numérateur il y a quatre triplets qui correspondent à un même déterminant, à un signe près. Ce sont (1, 1, 4), (1, 3, 2), (2, 1, 3), (2, 2, 2).

Nous obtenons :

$$N_{n,n-1} \sim (C_1^2C_4 - 2\ C_1C_3C_2 + C_2^3) \begin{vmatrix} 1 & e_n^2 & e_n & e_n^3 \\ 1 & e_{n-1}^2 & e_{n-1} & e_{n-1}^3 \\ 1 & 3\ e_n^2 & 2\ e_n & 4\ e_n^3 \\ 1 & 3\ e_{n-1}^2 & 2\ e_{n-1} & 4\ e_{n-1}^3 \end{vmatrix} e_n e_{n-1}$$

$$\sim (C_1^2C_4 - 2\ C_1C_3C_2 + C_2^3)\ e_n e_{n-1} \begin{vmatrix} 1 & e_n^3 & e_n^2 & e_n \\ 1 & e_{n-1}^3 & e_{n-1}^2 & e_{n-1} \\ 1 & 4\ e_n^3 & 3\ e_n^2 & 2\ e_n \\ 1 & 4\ e_{n-1}^3 & 3\ e_{n-1}^2 & 2\ e_{n-1} \end{vmatrix}$$

Dans $D_{n,n-1}$ effectuons les quelques transformations suivantes :

$$D_{n,n-1} \sim \frac{C_1^3}{e_n e_{n-1}} \begin{vmatrix} 1 & e_n^3 & e_n^2 & e_n \\ 1 & e_{n-1}^3 & e_{n-1}^2 & e_{n-1} \\ 0 & 3\ e_n^3 & 2\ e_n^2 & e_n \\ 0 & 3\ e_{n-1}^3 & 2\ e_{n-1}^2 & e_{n-1} \end{vmatrix}$$

$$= \frac{C_1^3}{e_n e_{n-1}} \begin{vmatrix} 1 & e_n^3 & e_n^2 & e_n \\ 1 & e_{n-1}^3 & e_{n-1}^2 & e_{n-1} \\ 1 & 4\ e_n^3 & 3\ e_n^2 & 2\ e_n \\ 1 & 4\ e_{n-1}^3 & 3\ e_{n-1}^2 & 2\ e_{n-1} \end{vmatrix}$$

Par conséquent nous obtenons :

$$e_{n+1} \sim \left(\frac{C_4}{C_1} - 2\,\frac{C_2 C_3}{C_1^2} + \left(\frac{C_2}{C_1}\right)^3\right) e_n^2\, e_{n-1}^2.$$

L'ordre r de J vérifie l'équation :

$$r^2 - 2r - 2 = 0$$

Pour trouver cette équation, il suffit de se reporter à la propriété 0 (Chapitre II) avec $p = 2$, $p_1 = 2$, $p_2 = 2$.

r est donc donné par :

$$r = 1 + \sqrt{3} = 2.732\ldots$$

Chaque itération exige deux évaluations de fonctions f et f'.

Donc son indice d'efficacité est :

$$I(J) = \sqrt{1 + \sqrt{3}} \approx 1.653.$$

4) *Méthode de Popovski ou méthode J^2* [21]

Sa méthode consiste en la composition de J avec elle-même. Ainsi sa $n^{\text{ième}}$ itération est donnée par :

$$(x_0,\ x_1 \text{ donnés},\ z_0 = x_0).$$

$n^{\text{ième}}$ itération :

$$\begin{cases} z_{n-1} = J(x_{n-1}, z_{n-2}) \\ x_n \quad = J(z_{n-1}, x_{n-1}) \end{cases}$$

Comme J est comparable à E_{12}, J^2 est aussi comparable à E_{12}^2. Par conséquent nous avons :

$r(J^2) = r(E_{12}) = 7.464$

$I(J^2) = I(E_{12}) = (7.464)^{\frac{1}{4}} \sim 1.653$

La différence qu'il y a entre J^2 et E_{12}^2 est que J^2 exige le calcul de la dérivée de f.

5) *1ère forme confluente de l'ε-algorithme*

Soit à résoudre f(x) = 0 de racine simple x*.

Soit $g(t) = f^{-1}(\frac{1}{t}) = x$.

$$\lim_{t\to\infty} g(t) = f^{-1}(0) = x_*.$$

Dans sa thèse [4] Brézinski a appliqué la 1ère forme confluente de l'ε-algorithme à la résolution d'une telle équation, en cherchant la limite de g(t) pour t tendant vers l'infini.

La règle de la 1ère forme confluente est donnée par :

$$\varepsilon_{-1}(t) = 0, \ \varepsilon_0(t) = g(t)$$

$$\varepsilon_{k+1}(t) = \varepsilon_{k-1}(t) + \frac{1}{\varepsilon_k'(t)}, \ k = 0, 1, 2,\ldots$$

Donc on a : $\varepsilon_2(t) = g(t) - \frac{[g'(t)]^2}{g''(t)}$.

Or $g'(t) = -\frac{1}{t^2 f'(x)}$,

$$g''(t) = \frac{2t[f'(x)]^2 - f''(x)}{t^4[f'(x)]^3}$$

$$t = \frac{1}{f(x)}.$$

En portant les expressions de g(t), g'(t), g"(t) et t en fonction de x dans l'expression donnant $\varepsilon_2(t)$, on obtient :

$$\varepsilon_2 = x - \frac{f'(x)\, f(x)}{2[f'(x)]^2 - f''(x)f(x)}.$$

Par itération sur ε_2, on a la méthode :

$$x_{n+1} = x_n - \frac{f'_n\, f_n}{2f'^2_n - f_n f''_n},$$

où $f_n = f(x_n)$, $f'_n = f'(x_n)$ et $f''_n = f''(x_n)$.

Puisque $\varepsilon_2(t)$ ne converge pas plus vite que $\varepsilon_0(t)$ [4], cette méthode ne peut être qu'à convergence linéaire. En effet si $e_n = x_n - x_*$, $e_{n+1} = x_{n+1} - x_*$ et $c_i = \frac{f_*^{(i)}}{i!}$, alors on a :

$$e_{n+1} = e_n - \frac{f'_n\, f_n}{2f'^2_n - f_n f''_n} = e_n - \frac{e_n\left(\sum_{i=1}^{\infty} iC_i\, e_n^{i-1}\right)\left(\sum_{i=1}^{\infty} C_i\, e_n^{i-1}\right)}{2\left(\sum_{i=1}^{\infty} iC_i\, e_n^{i-1}\right)^2 - e_n\left(\sum_{i=1}^{\infty} C_i\, e_n^{i-1}\right)\left(\sum_{i=2}^{\infty} i(i-1)\, e_n^{i-2}\right)}$$

$$= e_n - e_n \frac{C_1^2 + 3C_1C_2\, e_n + \ldots}{2\{C_1^2 + C_1C_2\, e_n + \ldots}$$

$$= \frac{1}{2}\, e_n + \ldots$$

Donc $e_{n+1} \sim \frac{1}{2}\, e_n$.

En introduisant un facteur d'accélération et en calculant sa valeur optimale, Brézinski [4] a obtenu de cette manière la méthode connue de Schröder :

$$x_{n+1} = x_n - \frac{2\, f'_n\, f_n}{2f'^2_n - f_n f''_n}.$$

Cette fois-çi, on a

$$e_{n+1} = O(e_n^3), \text{ pour des racines simples.}$$

$$e_{n+1} = e_n - \frac{2\, e_n(\sum_{i=1}^{\infty} iC_i\, e_n^{i-1})(\sum_{i=1}^{\infty} C_i\, e_n^{i-1})}{2(\sum_{i=1}^{\infty} iC_i\, e_n^{i-1})^2 - e_n(\Sigma\, C_i\, e_n^{i-1})(\sum_{i=1}^{\infty} i(i=1)\, e_n^{i-2})}$$

$$= e_n - e_n \frac{C_1^2 + 3C_1C_2\, e_n + 4C_1C_3\, e_n^2 + \dots}{C_1^2 + 3C_1C_2e_n + (3C_1C_3 + 3C_2^2)\, e_n^2 + \dots}$$

$$= e_n - e_n\,\{1 + \frac{C_1C_3 - 3C_2^2}{C_1^2}\, e_n^2 + \dots\}$$

$$= \left[3(\frac{C_2}{C_1})^2 - \frac{C_3}{C_1}\right] e_n^3 + \dots$$

La méthode ainsi obtenue exige trois évaluations de fonctions, f, f' et f", pour atteindre l'ordre trois. Donc son indice d'efficacité est donné par : $\sqrt[3]{3} = 1.442\dots$

Dans le paragraphe suivant on va donner des tableaux de comparaison entre les méthodes étudiées.

II.3.2. - Tableaux de comparaison

On note :

NF = nombre d'évaluations de fonctions par itération.

R = ordre de convergence.

I = indice d'efficacité.

METHODES DE TYPE STEFFENSEN

METHODE	S(1)	S(2)	S(k)	S_{12}	$S_{1\dots k}$	KING	J	J^2
NF	2	3	k+1	5	$\frac{k(k+3)}{2}$	3	2	4
R	2	3	k+1	6	(k+1)!	4	2.732	7.464
I	1.414	1.442	$(1+k)^{1/(k+1)}$	1.431	$((k+1)!)^{2/k(k+3)}$	1.587	1.653	1.68

METHODES DE TYPE SECANTE

METHODE	E(1)	E(2)	E(k)	E_{12}	$E_{1\dots k}$	$E_2 o E_1 o \phi$	E_{12}^2	E_{123}^2	NEWTON	SCHRÖDER
NF	1	1	1	2	k	3	4	6	2	3
R	1.618	1.839	k=3 ; 1.923 k=4 ; 1.955	2.732	k=3 ; 4.828 k=4 ; 8.899	4	7.464	23.312	2	3
I	1.618	1.839	1.923 1.955	1.653	1.690 1.727	1.587	1.68	1.714	1.414	1.442

COMPARAISON D'INDICES D'EFFICACITE

METHODE	S(1) ; NEWTON	S_{12}	S(2) et Schröder	$E_2 o E_1 o \phi$ King	E(1)	E_{12} J	E_{12}^2 J^2	E_{123}	E_{123}^2	E_{1234}	E(2)	E(3)	E(4)
I	1.414	1.431	1.442	1.587	1.618	1.653	1.68	1.690	1.714	1.727	1.839	1.923	1.955

Remarque :

Dans la pratique il n'est pas toujours très réaliste de ne pas tenir compte du nombre d'évaluations de fonctions de la première itération parce que l'on risque, dans certains cas de s'arrêter au bout d'un nombre faible d'itérations. Supposons donc que x_0 ait un chiffre exact. On peut se demander combien il faudra d'itérations (soit K ce nombre) pour obtenir N chiffres exacts et combien on aura alors fait d'évaluations de fonctions (soit M ce nombre, $M = q + (K-1)p$ si la première itération nécessite q évaluations et les autres p). Soit une méthode d'ordre r où chaque itération demande p évaluations de fonctions sauf la première qui en demande q. La quantité

$$(r^K)^{\frac{1}{q+(K-1)p}}$$

représente le facteur par lequel on a multiplié le nombre de chiffres exacts par évaluation de fonction après K itérations. Si q est égal à p on retrouve l'indice d'efficacité tel qu'il a été donné au paragraphe (II.0) de ce chapitre.

Supposons que x_0 ait un chiffre exact. Combien faut-il d'itérations pour avoir N chiffres ?

$r^K = N$ (K est le nombre d'itérations)

$\Rightarrow K\ LOG(r) = LOG(N)$

Soit $K = \left\lfloor \frac{LOG(N)}{LOG(r)} + 0.5 \right\rfloor$, où $\lfloor x \rfloor$ désigne la partie entière de x

et $K \sim M$: évaluations de fonctions en tenant compte du nombre d'évaluations de la première itération.

.Pour la méthode E(k) on a

$$q = k+1,\ p = 1 \text{ et } M = k + K.$$

Si K est très grand on a

$$M \sim K.$$

METHODE	E(1)	E(2)	E(3)	E(4)
q	2	3	4	5
p	1	1	1	1
r	1.6180340	1.8392868	1.9275620	1.9659482
N	6	6	6	6
K	4	3	3	3
M	5	5	6	7
N	8	8	8	8
K	4	3	3	3
M	5	5	6	7
N	18	18	18	18
K	6	5	4	4
M	7	8	7	8

II.4. - METHODES DEDUITES DES PROCEDES (G'), (G), (H') ET (H) (Cas d'une racine multiple)

II.4.1. - Méthode SM(1)

Cette méthode correspond à la suite colonne $(T_1^{(n)})_{n\in\mathbb{N}}$ du procédé (G'). Etant donné x_0, sa $n^{\text{ième}}$ itération se déroule de la façon suivante :

$n^{\text{ième}}$ itération :

$$\begin{cases} u_0 = x_{n-1} \\ u_{i+1} = u_i - f(u_i) \text{ pour } i = 0,\ 1,\ 2. \\ x_n = \dfrac{\dfrac{(\Delta u_1)^2}{\Delta^2 u_1} u_0 - \dfrac{(\Delta u_0)^2}{\Delta^2 u_0} u_1}{\dfrac{(\Delta u_1)^2}{\Delta^2 u_1} - \dfrac{(\Delta u_0)^2}{\Delta^2 u_0}} \end{cases}$$

Soit $G(x) = -\frac{f^2(x)}{f(x-f(x))-f(x)}$. Posons $G_0 = G(u_0)$ et $G_1 = G(u_1)$;

x_n peut encore s'écrire :

$$x_n = \frac{G_1\, u_0 - G_0\, u_1}{G_1 - G_0}.$$

L'étude faite sur l'algorithme (B) appliqué à la fonction G, nous permet d'écrire :

$$e_n \sim \frac{1}{2}\, \frac{G''_*}{G'_*}\, \epsilon_0\, \epsilon_1,$$

où $\epsilon_0 = e_{n-1}$, $\epsilon_1 = u_1 - x_*$, $e_{n-1} = x_{n-1} - x_*$ et $e_n = x_n - x_*$.

Soit $m (m \geq 2)$ la multiplicité de x_*. Supposons que :

$$f(x) = (x-x_*)^m\, g(x) \text{ avec } g(x_*) \neq 0.$$

$$\Rightarrow \begin{cases} G'(x_*) = G'_* = \dfrac{1}{m} \\[2ex] G''(x_*) = G''_* = \dfrac{g_*\, f''_* - 2g'_*}{m^2 g_*} \text{ où } g_* = g(x_*),\ g'_* = g'(x_*) \text{ et } f''_* = f''(x_*) \end{cases}$$

Par conséquent nous avons :

$$e_n \sim \frac{1}{2}\, \frac{g_* f''_* - 2g'_*}{m g_*}\, e_{n-1}^2$$

C'est une méthode d'ordre 2, exigeant trois évaluations de fonctions par itération. Son indice d'efficacité est

$$I(SM(1)) = \sqrt[3]{2} = 1.260.$$

<u>Remarque</u> :

Le rapport $\frac{u_1-u_0}{G_1-G_0}$ est une approximation de m. En effet :

$$\frac{u_1-u_0}{G_1-G_0} = \frac{\Delta u_0}{\dfrac{(\Delta u_1)^2}{\Delta^2 u_1} - \dfrac{(\Delta u_0)^2}{\Delta^2 u_0}} = \frac{1}{\dfrac{\Delta G_0}{\Delta u_0}}$$

$$\text{Or } \frac{\Delta G_0}{\Delta u_0} = \frac{G'(u_0)\,\Delta u_0 + \frac{G''(u_0)}{2}(\Delta u_0)^2 + \ldots}{\Delta u_0} = G'(u_0) + \ldots$$

$$\text{et } G'(u_0) = G'_* + G''_*\, e_{n-1} + \ldots$$

$$= \frac{1}{m} + O(e_{n-1}).$$

Nous pouvons donc écrire :

$$\frac{\Delta u_0}{\Delta G_0} = \frac{1}{\frac{1}{m} + O(e_{n-1})} = \frac{m}{1+O(e_{n-1})} = m + O(e_{n-1}).$$

$$\text{Soit } m_n = \frac{\Delta u_0}{\Delta G_0} = m + O(e_{n-1}).$$

Par suite nous pouvons générer au cours des itérations de SM(1) des approximations m_n de m.

La $n^{\text{ième}}$ itération donnant à la fois une approximation de x_* et une approximation de m se résume par :

x_0 donné

<u>$n^{\text{ième}}$ itération</u> :

$$\left[\begin{array}{l}
u_0 = x_{n-1} \\
u_{i+1} = u_i - f(u_i) \text{ pour } i = 0,\ 1,\ 2 \\
x_n = \dfrac{\dfrac{(\Delta u_1)^2}{\Delta^2 u_1}\, u_0 - \dfrac{(\Delta u_0)^2}{\Delta^2 u_0}\, u_1}{\dfrac{(\Delta u_1)^2}{\Delta^2 u_1} - \dfrac{(\Delta u_0)^2}{\Delta^2 u_0}} \\
m_n = \dfrac{u_1 - u_0}{\dfrac{(\Delta u_1)^2}{\Delta^2 u_1} - \dfrac{(\Delta u_0)^2}{\Delta^2 u_0}}
\end{array}\right.$$

Remarque :

Comme x_* est une racine simple de G, on peut donc lui appliquer la méthode S(k). Cependant le nombre d'évaluations de fonctions par itération est grand. Par exemple S(1) exige 2 évaluations de G par itération, donc 4 évaluations de f. Et par conséquent l'indice d'efficacité de S(1) est

$$I(S(1)) = \sqrt[4]{2} < \sqrt[3]{2}.$$

L'essai fait avec cette méthode montre qu'elle est numériquement instable.

II.4.2. - Méthode TM(1)

Elle correspond à la colonne $(\rho_2^{(n)})_{n\in\mathbb{N}}$ du procédé (H').

x_0 donné. Sa $n^{\text{ième}}$ itération est donnée par :

$n^{\text{ième}}$ itération :

$$\begin{cases} u_0 = x_{n-1} \\ u_{i+1} = u_i - f(u_i) \text{ pour } i = 0,\ 1,\ 2,\ 3. \\ \rho_{-1}^{(i)} = 0,\ \rho_0^{(i)} = u_i,\ i = 0,\ 1,\ 2 \\ \rho_j^{(i)} = \rho_{j-2}^{(i+1)} + \dfrac{\dfrac{\Delta^2 u_{i+j}}{(\Delta u_{i+j})^2} - \dfrac{\Delta^2 u_i}{(\Delta u_i)^2}}{\rho_{j-1}^{(i+1)} - \rho_{j-1}^{(i)}},\ \begin{array}{l} j = 1,\ 2 \\ i = 0,\ 1, \ldots,\ 2-j. \end{array} \\ x_n = \rho_2^{(0)} \end{cases}$$

Soit G la fonction définie comme précédemment par :

$$G(x) = - \frac{f^2_{(x)}}{f(x-f(x)) - f(x)}.$$

L'itéré x_n est également le résultat que l'on obtient par application de l'algorithme (E) à la suite $\{u_0, u_1, u_2\}$, en prenant pour fonction, la fonction G. Donc nous pouvons écrire :

$$e_n \sim \frac{1}{m^3} \frac{D_2}{3!} \epsilon_0 \, \epsilon_1 \, \epsilon_2$$

où $\epsilon_0 = e_{n-1}$, $\epsilon_1 = u_1 - x_*$, $\epsilon_2 = u_2 - x_*$ et où D_2 est donné par la propriété 17 du chapitre I.

Comme $f'_* = 0$, alors on a :

$$\epsilon_1 \sim e_{n-1}$$

$$\text{et } \epsilon_2 \sim e_{n-1}$$

d'où la forme définitive de l'erreur :

$$e_n \sim \frac{1}{m^3} \frac{D_2}{6} e_{n-1}^3$$

TM(1) est d'ordre 3, et elle exige 4 évaluations de f par itération. Donc son indice d'efficacité est :

$$I(TM(1)) = \sqrt[4]{3} \simeq 1.316.$$

Dans ce cas là nous pouvons donner aussi une estimation de la multiplicité m de x_*. On prendra par exemple :

$$m_n = \frac{u_2 - u_0}{\dfrac{(\Delta u_2)^2}{\Delta^2 u_2} - \dfrac{(\Delta u_0)^2}{\Delta^2 u_0}}, \; n \geq 1$$

<u>Remarque</u> :

Nous pouvons construire des méthodes SM(K) et TM(K) d'un ordre arbitraire, associées aux suites colonnes $(T_k^{(n)})_{n\in\mathbb{N}}$ et $(\rho_{2k}^{(n)})_{n\in\mathbb{N}}$ des procédés (G') et (H'). Cependant comme le travail est identique à celui fait en II.1 et II.2, nous nous limitons aux méthodes SM(1) et TM(1).

II.4.3. - Méthodes EM(1) et EM(2)

a) Méthode EM(1)

Elle est obtenue par King [15]. Elle correspond en fait à la suite colonne $(T_1^{(n)})_{n\in\mathbb{N}}$ du procédé (G). C'est aussi la méthode E(1) appliquée à la fonction G(x). Par conséquent l'itéré x_n est donné par :

$$x_n = \frac{G_{n-2}\, x_{n-1} - G_{n-1}\, x_{n-2}}{G_{n-2} - G_{n-1}}.$$

$$\text{où } G_{n-1} = G(x_{n-1}) = \frac{-f^2(x_{n-1})}{f(x_{n-1}-f(x_{n-1})) - f(x_{n-1})}$$

$$\text{et } G_{n-2} = G(x_{n-2}) = \frac{-f^2(x_{n-2})}{f(x_{n-2}-f(x_{n-2})) - f(x_{n-2})}$$

La propriété 14 du chapitre I, nous permet d'écrire après quelques transformations :

$$e_n \sim \frac{1}{2m}\,\frac{g_* \, g_*'' - 2g_*'}{g_*}\, e_{n-1}\, e_{n-2} = \frac{1}{2}\,\frac{G_*''}{G_*'}\; e_{n-1}\, e_{n-2}.$$

où $g_* = g(x_*)$, $f_*'' = f''(x_*)$, $g'(x_*) = g_*'$,

$f(x) = (x-x_*)^m\, g(x)$ avec $g(x_*) \neq 0$.

Son ordre est celui de la méthode E(1). Mais l'indice d'efficacité de EM(1) n'est plus égal à son ordre. Il est donné par :

$$I(EM(1)) = \sqrt{I(E(1))} = \sqrt{\frac{1+\sqrt{5}}{2}} \simeq 1.272.$$

King [15] a développé l'erreur de la manière suivante :

$$e_n \sim \frac{1}{2}\,\frac{G_*''}{G_*'}\, e_{n-2}\, e_{n-1} + \left\{\frac{1}{6}\,\frac{G_*^{(3)}}{G_*'} - \left(\frac{1}{2}\,\frac{G_*''}{G_*'}\right)^2\right\} (e_{n-2} + e_{n-1})\, e_{n-2}\, e_{n-1}$$

$$\sim \{A + B(e_{n-2} + e_{n-1})\}\, e_{n-2}\, e_{n-1},$$

avec $A = \frac{1}{2}\,\frac{G_*''}{G_*'}$ et $B = \frac{1}{6}\,\frac{G_*^{(3)}}{G_*'} - \left(\frac{1}{2}\,\frac{G_*''}{G_*'}\right)^2.$

Nous pouvons également obtenir ce développement de la manière suivante :

$$e_{n-1} = x_{n-1}-x_*$$

$$e_{n-2} = x_{n-2}-x_*$$

$$G_{n-1} = G'_* \, e_{n-1} + \frac{G''_*}{2} e^2_{n-1} + \frac{G^{(3)}_*}{6} e^3_{n-1} + \ldots$$

$$G_{n-2} = G'_* \, e_{n-2} + \frac{G''_*}{2} e^2_{n-2} + \frac{G^{(3)}_*}{6} e^3_{n-2} + \ldots$$

$$G_{n-2} \, e_{n-1} = G'_* \, e_{n-2} \, e_{n-1} + \frac{G''_*}{2} e^2_{n-2} \, e_{n-1} + \frac{G^{(3)}_*}{6} e^3_{n-2} \, e_{n-1} + \ldots$$

$$G_{n-1} \, e_{n-2} = G'_* \, e_{n-1} \, e_{n-2} + \frac{G''_*}{2} e^2_{n-1} \, e_{n-2} + \frac{G^{(3)}_*}{6} e^3_{n-1} \, e_{n-2} + \ldots$$

$$G_{n-2} \, e_{n-1} - G_{n-1} \, e_{n-2} = e_{n-2}e_{n-1} \left\{ \frac{G''_*}{2}(e_{n-2}-e_{n-1}) + \frac{G^{(3)}_*}{6}(e_{n-2}-e_{n-1}) (e_{n-2}+e_{n-1}) + \ldots \right\}$$

$$\sim e_{n-2}e_{n-1} \, (e_{n-2}-e_{n-1}) \left\{ \frac{G''_*}{2} + \frac{G^{(3)}_*}{6}(e_{n-2} + e_{n-1}) \right\}$$

en négligeant les autres termes de la somme entre accolade comportant $e^i_{n-2} \, e^j_{n-1}$ avec $i+j \geq 3$.

$$G_{n-2} - G_{n-1} = (e_{n-2}-e_{n-1}) \left\{ G'_* + \frac{G''_*}{2} (e_{n-2} + e_{n-1}) + \ldots \right\}$$

$$\sim (e_{n-2}-e_{n-1})(G'_* + \frac{G'_*}{2} (e_{n-2} + e_{n-1})).$$

Posons $e = e_{n-2} + e_{n-1}$, on a

$$e_n = x_n - x_* = \frac{G_{n-2}e_{n-1} - G_{n-1}e_{n-2}}{G_{n-2} - G_{n-1}} = e_{n-2}e_{n-1}\,\frac{\frac{G''_*}{2} + \frac{G^{(3)}_*}{6}\,e + \ldots}{G'_* + \frac{G''_*}{2}\,e + \ldots}$$

$$= e_{n-2}\,e_{n-1}\,\frac{\frac{G''_*}{2G'_*} + \frac{G^{(3)}_*}{6G'_*}\,e + \ldots}{1 + \frac{G''_*}{2G'_*}\,e + \ldots}$$

$$\sim e_{n-2}e_{n-1}\left\{\frac{G''_*}{2G'_*} + \left(\frac{G^{(3)}_*}{6G'_*} - \left(\frac{G''_*}{2G'_*}\right)^2\right)(e_{n-2} + e_{n-1})\right\}$$

$$= e_{n-2}\,e_{n-1}\,\{A + B(e_{n-2} + e_{n-1})\}.$$

Cette dernière forme de l'erreur nous sera très utile pour la méthode suivante.

b) Méthode EM(2)

Elle correspond à la colonne $(T_2^{(n)})_{n\in\mathbb{N}}$ du procédé **(G)**. C'est aussi E(2) appliquée à G(x). Or on sait que E(2) correspond à la colonne $(T_2^{(n)})_{n\in\mathbb{N}}$ du procédé (B), dans lequel on a remplacé la fonction f par la fonction G.

L'itéré x_n fourni par EM(2) est donné par :

x_0, x_1, x_2 donnés

$$\begin{cases} T_0^{(i)} = x_{n-3+i}, \quad i = 0, 1, 2 \\[2ex] T_j^{(i)} = \dfrac{G_{i+j}\,T_{j-1}^{(i)} - G_i\,T_{j-1}^{(i+1)}}{G_{i+j} - G_i}, \quad \begin{array}{l} j = 1, 2 \\ i = 0,\ldots, 2-j \end{array} \\[2ex] x_n = T_2^{(0)} \end{cases}$$

où $G_\ell = G(x_\ell) = \dfrac{-\,f^2(x_\ell)}{f(x_\ell - f(x_\ell)) - f(x_\ell)}$.

Comme $T_1^{(0)}$ est obtenu par application de EM(1) à f ou E(1) à G, et $T_1^{(1)}$ est obtenu par application de EM(1) à f ou E(1) à G, alors on a :

$$T_1^{(0)}-x_* \sim \{A + B(e_{n-2} + e_{n-3})\} e_{n-2} e_{n-3}$$

$$T_1^{(1)}-x_* \sim \{A + B(e_{n-1} + e_{n-2})\} e_{n-1} e_{n-2}.$$

Nous pouvons écrire G_{n-1} et G_{n-3} sous les formes :

$$G_{n-1} = C_1 e_{n-1} + C_2 e_{n-1}^2 +\dots$$

$$G_{n-3} = C_1 e_{n-3} + C_2 e_{n-3}^2 +\dots$$

avec $C_i = \dfrac{G_*^{(i)}}{i!}$.

Par conséquent nous avons :

$$e_n = T_2^{(0)}-x_* = \frac{G_{n-1}(T_1^{(0)}-x_*) - G_{n-3}(T_1^{(1)}-x_*)}{G_{n-1} - G_{n-3}}$$

$$= \frac{e_{n-1}e_{n-2}e_{n-3}[\{AC_1+AC_2e_{n-1} + B(e_{n-2}+e_{n-3})C_1+\dots\}-\{AC_1+AC_2e_{n-3} + BC_1(e_{n-1}+e_{n-2})\}]}{C_1(e_{n-1}-e_{n-3}) + C_2(e_{n-1}-e_{n-3})(e_{n-1}+e_{n-3}) +\dots}$$

$$= \frac{e_{n-1}e_{n-2}e_{n-3}\ \{(AC_2-BC_1)(e_{n-1}-e_{n-3}) +\dots\}}{(e_{n-1}-e_{n-3})\ \{C_1 + C_2\ (e_{n-1}+e_{n-3}) +\dots\}}$$

$$\sim \frac{AC_2 - BC_1}{C_1} e_{n-1}e_{n-2}e_{n-3}.$$

Nous avons donné ce développement de l'erreur pour signaler cette autre possibilité sans passer par l'intermédiaire de la fonction inverse de G, comme nous l'avons fait lors de l'étude des méthodes E(k), et en particulier E(2). Il est évident que EM(2) a le même ordre de convergence que E(2). Mais son indice d'efficacité est :

$$I(EM(2)) = \sqrt{I(E(2))} \simeq 1.356$$

du fait que EM(2) exige deux évaluations de fonctions par itération.

Remarque :

Dans [13] King a proposé pour m l'approximation :

$$m_n = \frac{x_n - x_{n-1}}{G_n - G_{n-1}}, \text{ où } G(x) = \frac{-f^2(x)}{f(x-f(x)) - f(x)}.$$

On peut donc calculer des approximations successives de m aux cours des itérations des deux méthodes EM(1) et EM(2).

II.4.4. - Méthode FM(1)

Elle provient de l'itération successive sur $\rho_2^{(n)}$ du procédé (H).

Ainsi sa $n^{\text{ième}}$ itération est donnée par (après avoir donné x_0, x_1, x_2) :

$$\left[\begin{array}{l}
\rho_{-1}^{(i)} = 0, \ \rho_0^{(i)} = x_{n-3+i}, \ i = 0, 1, 2 \\[2ex]
\rho_j^{(i)} = \rho_{j-2}^{(i+1)} + \dfrac{\dfrac{1}{G_{i+j}} - \dfrac{1}{G_i}}{\rho_{j-1}^{(i+1)} - \rho_{j-1}^{(i)}}, \ \begin{array}{l} j = 1, 2 \\ i = 0, \ldots, 2-j \end{array} \\[2ex]
\text{où } G_i = G(x_{n-3+i}) = \dfrac{-[f(x_{n-3+i})]^2}{f(x_{n-3+i} - f_{n-3+i}) - f_{n-3+i}} \\[2ex]
\text{et } G_{i+j} = G(x_{n-3+i+j}) = \dfrac{-[f_{n-3+i+j}]^2}{f(x_{n-3+i+j} - f_{n-3+i+j}) - f_{n-3+i+j}} \\[2ex]
\text{Puis on prend } x_n = \rho_2^{(0)}.
\end{array}\right.$$

L'étude faite sur l'algorithme (H) nous permet de conclure que FM(1) est identique à EM(2). Ainsi si I(FM(1)) désigne son indice d'efficacité et si I(EM(2)) celui de EM(2), nous avons :

$$I(FM(1)) = I(EM(2) = \sqrt{I(E(2))} \simeq 1.356$$

où I(E(2)) désigne l'indice d'efficacité de E(2).

Remarque :

D'une façon parallèle à la construction des méthodes E(k) et F(k) nous pouvons tirer des algorithmes (G) et (H) des méthodes EM(k) et FM(k). Ces dernières sont en fait des méthodes E(k) et F(k) appliquées à la fonction G. Seuls les indices d'efficacité sont diminués. Nous avons en effet :

$$I(EM(k)) = \sqrt{I(E(k)}$$
$$I(FM(k)) = \sqrt{I(F(k)}$$
$$I(E(2(k)) = I(F(k))$$

où I(E(k)) (respectivement, I(F(k)), I(EM(k)), I(FM(k))) désigne l'indice d'efficacité de la méthode E(k) (respectivement, F(k), EM(k), FM(k)).

II.5. - CONCLUSION

Les méthodes E(k) (respectivement F(k)) et leurs composées $E_{1...k}$ (respectivement $F_{1...k}$) sont, parmi les méthodes que nous venons d'exposer, les méthodes les plus efficaces pour la résolution des équations non linéaires à racine simple. D'une part elles utilisent moins d'évaluations de fonctions et, d'autre part, leur indice d'efficacité coïncide avec leur ordre de convergence et il est voisin de 2. La seule fonction qui soit mise en jeu est la fonction f. Ses dérivées n'interviennent pas. Ces méthodes, comme nous l'avons vu, correspondent aux algorithmes (B) et (E).

Pour les équations non linéaires à racine multiple, nous retenons les méthodes EM(k) et FM(k), correspondant respectivement aux algorithmes (G) et (H).

ESSAIS NUMERIQUES

$\phi(x) = e^{-x}$

$f(x) = x - e^{-x}$

$x_* = 0.567\ 143\ 290\ 409\ 783$

S(1) appliquée à $x = \phi(x) = e^{-x}$

n	x_n
0	1.0000000000000000
1	.5822260969956230
2	.5671664379478828
3	.5671432904647697
4	.5671432904097839

S(2) appliquée à $\phi(x) = e^{-x}$

n	x_n
0	1.0000000000000000
1	.5671256979845161
2	.5671432904097839

S(k) appliquée à $\phi(x) = e^{-x}$

k=4

n	x_n
0	.1000000000000000D+01
1	.5671432389174804D+00
2	.5671432904097839D+00
3	.5671432904097839D+00
	.5671432904097839D+00

S(2) o S(1) : $(x) = e^{-x}$

n	x_n
0	.1000000000000000D+01
1	.5671432872857334D+00
2	.5671432904097839D+00
3	.5671432904097839D+00

E(1) appliquée à $x-e^{-x} = 0$; $x_0 = 0.2$, $x_1 = 0.3$

n	x_n
0	.2000000000000000
1	.3000000000000000
2	.5477724389307689
3	.5661666704128394
4	.5671398565287380
5	.5671432898028475
6	.5671432904097835
7	.5671432904097839

E(2) appliquée à $x-e^{-x} = 0$; $x_0 = 0.2$, $x_1 = 0.3$, $x_2 = 0.4$

n	x_n
0	.2000000000000000
1	.3000000000000000
2	.4000000000000000
3	.5669069341164280
4	.5671431840420590
5	.5671432904097543
6	.5671432904097839

E(k) appliquée à $f(x) = x-e^{-x} = 0$

$x_0 = 0.2$, $x_1 = 0.3$, $x_2 = 0.4$, $x_3 = 0.5$

n	x_n
0	.2000000000000000D+00
1	.3000000000000000D+00
2	.4000000000000000D+00
3	.5000000000000000D+00
4	.5671531711122229D+00
5	.5671432901290311D+00
6	.5671432904097839D+00

Ordres et indices de l'efficacité des méthodes E(K) E(K-1)... E(1)

K	Ordre R(K)	Indice d'efficacité I(K)
1	1.6180339887498946	1.6180339887498946
2	2.7320508075688772	1.6528916502810691
3	4.8284271247461901	1.6901888290514397
4	8.8989794855663559	1.7271696626725407
5	16.9442719099991554	1.7611833961191901

$E_{12} = E(2) \circ E(1)$ appliquée à $x-e^{-x} = 0$

$x_0 = 0.2$, $x_1 = 0.3$

n	x_n
0	.2000000000000000
1	.3000000000000000
2	.5671192029225770
3	.5671432904097202
4	.5671432904097839
5	.5671432904097839
6	.5671432904097839
7	.5671432904097839

$E_{12}^2 = E_{12} \circ E_{12}$ appliquée à $x-e^{-x} = 0$

$x_0 = z_0 = 0.2$, $x_1 = 0.3$

n	x_n	z_n
0	.2000000000000000	.2000000000000000
1	.3000000000000000	.5671192029225770
2	.5671432904097202	.5671432904097839
3	.5671432904097839	.5671432904097839
4	.5671432904097839	.5671432904097839
5	.5671432904097839	.5671432904097839
6	.5671432904097839	.5671432904097839
7	.5671432904097839	

$E_{123}^2 = E_{123} \circ E_{123}$ appliquée à $x-e^{-x} = 0$

$x_0 = z_0 = 0.2$, $x_1 = 0.3$

n	x_n	z_n
0	.20000000000000000+00	.20000000000000000+00
1	.30000000000000000+00	.56714939101052910+00
2	.56714329037255260+00	.56714329040978390+00
3	.56714329040978390+00	.56714329040978390+00
4	.56714329040978390+00	.56714329040978390+00
5	.56714329040978390+00	.56714329040978390+00
6	.56714329040978390+00	.56714329040978390+00
7	.56714329040978390+00	.56714329040978390+00
8	.56714329040978390+00	

F(1) $1^{ère}$ généralisation de la sécante

(Cas de l'extrapolation rationnelle)

n	x_n
0	.3000000000000000
1	.4000000000000000
2	.5000000000000000
3	.5672240346036927
4	.5671432656483726
5	.5671432904097802
6	.5671432904097839
7	.5671432904097839

T(1) appliquée à $x = e^{-x}$, $x_0 = 1$

n	x_n
0	.1000000000000000+01
1	.5674416067776432D+00
2	.567143290409917 4D+00
3	.5671432904097839D+00
4	.5671432904097839D+00

C(2) ou méthode de Schröder appliquée à $x-e^{-x} = 0$

$x_0 = 1$

n	x_n
0	1.0000000000000000
1	.5649192899718807
2	.5671432907130433
3	.5671432904097839
4	.5671432904097839
5	.5671432904097839
6	.5671432904097839
7	.5671432904097839

Méthode d'Aitken appliquée à $x = e^{-x}$

$x_0 = 1$

x_0	.1000000000000000D+01
x_1	.3678794411714423D+00
x_2	.6922006275553464D+00
$\bar{x}_2$	.5822260969956230D+00
x_3	.5586533647552658D+00
x_4	.5719787922616897D+00
$\bar{x}_4$	.5671664379478828D+00
x_5	.5671301625908005D+00
x_6	.5671507358131089D+00
$\bar{x}_6$	.5671432904647697D+00
x_7	.5671432903785990D+00
x_8	.5671432904274762D+00
$\bar{x}_8$	.5671432904097839D+00

Méthode de King appliquée à : $x = e^{-x}$

$x_0 = 1$

x_0	.1000000000000000D+01
x_1	.3678794411714423D+00
x_2	.6922006275553464D+00
$\bar{x}_2$	.5822260969956230D+00
x_3	.5586533647552658D+00
$\bar{x}_3$	.5670621447099630D+00
x_4	.5671893135162463D+00
x_5	.5671171893143813D+00
$\bar{x}_5$	.5671432910855523D+00
x_6	.5671432900265264D+00
$\bar{x}_6$	.5671432904097839D+00
x_7	.5671432904097839D+00
x_8	.5671432904097839D+00
$\bar{x}_8$	.5671432904097839D+00
x_9	.5671432904097839D+00
$\bar{x}_9$	.5671432904097839D+00
x_{10}	.5671432904097839D+00
x_{11}	.5671432904097839D+00
$\bar{x}_{11}$	.5671432904097839D+00
x_{12}	.5671432904097839D+00
$\bar{x}_{12}$	.5671432904097839D+00

Méthode F(2) appliquée à $x-e^{-x} = 0$

$x_0 = 0$, $x_1 = 1.$, $x_2 = 2.$, $x_3 = 2.5$, $x_4 = 1.5$

n	x_n
0	.000000000000D+00
1	.100000000000D+01
2	.200000000000D+01
3	.250000000000D+01
4	.150000000000D+01
5	.566302792345D+00
6	.567142059031D+00
7	.567143290413D+00
8	.567143290410D+00

Méthode T(2) appliquée à $x = e^{-x}$

$x_0 = 1$

n	x_n
0	.1000000000000000D+01
1	.5671433446650072D+00
2	.5671432904097839D+00
3	.5671432904097839D+00
4	.5671432904097839D+00

(J^2) appliquée à : $x-e^{-x} = 0$

$x_1 = 0.2$ et $x_2 = 0.3$

n	x_n	z_n
1	.2000000000000000D+00	.2000000000000000D+00
2	.3000000000000000D+00	.5671678582744503D+00
3	.5671432904098493D+00	.5671432904097839D+00
4	.5671432904097839D+00	.5671432904097839D+00
5	.5671432904097839D+00	.5671432904097839D+00
6	.5671432904097839D+00	.5671432904097839D+00
7	.5671432904097839D+00	.5671432904097839D+00
8	.5671432904097839D+00	.5671432904097839D+00
9	.5671432904097839D+00	.5671432904097839D+00
10	.5671432904097839D+00	

(J) appliquée à : $x-e^{-x} = 0$

$x_1 = 0.2$, $x_2 = 0.3$

n	x_n
1	.20000000000000000D+00
2	.30000000000000000D+00
3	.56716785827445030D+00
4	.56714329040984930D+00
5	.56714329040978390D+00
6	.56714329040978390D+00
7	.56714329040978390D+00
8	.56714329040978390D+00
9	.56714329040978390D+00
10	.56714329040978390D+00

Soit r_k (respectivement s_k) l'ordre de convergence de E(k) (respectivement F(k)).

$$s_k = r_{2k}$$

r_k est racine de $P_k(r) = r^{k+1} - \sum_{i=0}^{k} r^{k-i}$

s_k est racine de $P_{2k}(r) = r^{2k+1} - \sum_{i=0}^{2k} r^{2k-i}$

Une seule racine réelle (règle de Descartes) au plus.

$P_k(1) < 0$ et $P_k(2) > 0$.

k	r_k	s_k
1	1.618 034 0	1.839 286 8
2	1.839 562 8	1.965 948 2
3	1.927 562 0	1.991 964 2
4	1.965 948 2	1.998 029 5
5	1.983 582 8	1.999 510 4
6	1.991 694 2	1.999 877 8
7	1.996 031 2	1.999 969 5
8	1.998 029 5	1.999 992 4
9	1.999 018 6	1.999 998 1
10	1.999 510 4	1.999 999 5

$$f(x) = (x-1)^2 (x^2 + 1) = 0$$

$$(x) = x - f(x),\ x_* = 1,\ m_* = 2$$

SM(1)

n	x_n	m_n
1	.70000000000000000D+00	.17354898198958520D+01
2	.97211536819167790D+00	.20429694620367480D+01
3	.10002713444715840D+01	.19984272516895400D+01
4	.10000001764244370D+01	-.13817870511760370D-75
5	.10000001764243740D+01	-.13817870511760370D-75
6	.10000001764243120D+01	-.13817870511760370D-75
7	.10000001764242500D+01	-.13817870511760370D-75
8	.10000001764241880D+01	-.13817870511760370D-75
9	.10000001764241260D+01	-.13817870511760370D-75
10	.10000001764240640D+01	

EM(1)

n	x_n	m_n
1	.50000000000000000D+00	
2	.60000000000000000D+00	.20960792308989080D+01
3	.10519402212611810D+01	.18636579213462030D+01
4	.10018273550504820D+01	.19292731995583840D+01
5	.10000629917406900D+01	.19292731995583840D+01
6	.10000000582068200D+01	.19292731995583840D+01
7	.99999999899841070D+00	.19292731995583840D+01
8	.10000000582068200D+01	.19292731995583840D+01
9	.10000000582068200D+01	.19292731995583840D+01
10	.10000000582068200D+01	.19292731995583840D+01

$f(x_n)$
.21760000000000010D+00
.56830984660227250D-02
.66906680162232410D-05
.79364187035130200D-08
.67760682422487520D-14
.20063620733708530D-17
.67760682422487520D-14
.67760682422487520D-14
.67760682422487520D-14

TM(1)

n	x_n
1	.9000000000000000D+00
2	.1003454638079201D+01
3	.9999998442416489D+00
4	.9999998442416004D+00
5	.9999998442415519D+00
6	.9999998442415033D+00
7	.9999998442414588D+00
8	.9999998442414063D+00
9	.9999998442413578D+00
10	.9999998442413693D+00

EM(2)

n	x_n	m_n	$f(x_n)$
1	.5000000000000000D+00		.3125000000000000D+00
2	.6000000000000000D+00	.2096079230898908D+01	.2176000000000001D+00
3	.7000000000000000D+00	.1700170768879415D+01	.1341000000000000D+00
4	.8409327334472735D+00	.1801503090799151D+01	.4319543607287627D-01
5	.9915740083457274D+00	.2025627207736296D+01	.1408032654269764D-03
6	.1000558154289501D+01	.2007467528057491D+01	.6234202893764534D-06
7	.9999980288694639D+00	.2007467528057491D+01	.7770695863833343D-11
8	.9999999999240330D+00	.2007467528057491D+01	.1154204082793340D-19
9	.9999999999240330D+00	.2007467528057491D+01	.1154204082793340D-19
10	.9999999999240330D+00	.2007467528057491D+01	.1154204082793340D-19

CHAPITRE III

INTERPRETATION DU E-ALGORITHME

ET DU PROCEDE GBW

GENERALISATION DU PROCEDE D'OVERHOLT

INTRODUCTION

Ce chapitre se compose de cinq parties.

Dans la première, nous donnons les deux conceptions qui ont permis à C. Brézinski [6] et à T. Havie [12] de formuler les règles du E-algorithme.

Dans la deuxième partie, nous introduisons une interprétation géométrique du E-algorithme, basée sur une interprétation de R.C. Johnson [14] et de R.R. Tucker [26] de la transformation de Shanks. Cela nous ramène à la recherche de point fixe par cet algorithme, d'où la formulation de nouvelles méthodes. Ainsi, le E-algorithme ou la transformation de Shanks engendre des méthodes de recherche de points fixes d'un ordre k arbitraire. Il donne également une extension de la méthode de la sécante. Cependant la méthode obtenue par itération sur $E_2^{(n)}$ (deuxième colonne du tableau du E-algorithme) a un ordre inférieur à celui de la méthode de la sécante. Elle est donc moins intéressante.

La troisième partie est réservée à un rappel de la construction du procédé d'Overholt [18]. Cette construction a permi d'interpréter le procédé (B') (Chapitre I) dû à Germain-Bonne [11] dans la quatrième partie.

La cinquième partie est consacrée à une généralisation des procédés d'Overholt, de Germain-Bonne-Wimp et de Richardson. Après avoir donné quelques résultats de convergence et d'accélération de la convergence concernant cette généralisation ; nous lui appliquons la procédure Θ. Nous terminons par l'étude du procédé ainsi obtenu.

III.1. - E-ALGORITHME

Il a été obtenu indépendamment par C. Brézinski [6] et T. Havie [12].

Nous allons ci-dessous le construire d'une façon regroupant l'une ou l'autre manière par laquelle ces auteurs l'ont introduit.

Soit (S_n) une suite de la forme

$$S_n = S + \sum_{i\geq 1} a_i\ g_i(n),\ \forall n\ ;$$

où S, a_1, $a_2,\ldots$, $a_k,\ldots$ sont inconnus, $g_i(n)$ donnés.

Nous nous intéressons à l'inconnue S. Nous construisons des quantités $E_k^{(n)}$ pour différentes valeurs de n et de k. Nous les considérons comme des approximations de S. Cette procédure est obtenue par élimination successive des inconnues a_i.

III.1.1. - Elimination de a_1

Considérons les deux égalités

$$S_n = S + \sum_{i\geq 1} a_i\ g_i(n)$$

$$S_{n+1} = S + \sum_{i\geq 1} a_i\ g_i(n+1)$$

Multiplions la première par $g_1(n+1)$ et la seconde par $g_1(n)$. Cela donne

$$\left.\begin{array}{l} g_1(n+1)S_n = g_1(n+1)S + \sum_{i\geq 1} a_i\ g_1(n+1)\ g_i(n) \\ g_1(n)S_{n+1} = g_1(n)S + \sum_{i\geq 1} a_i\ g_1(n)\ g_i(n+1) \\ g_1(n+1) - g_1(n) \neq 0 \end{array}\right\} \Rightarrow \ldots$$

$$\ldots \frac{g_1(n+1)S_n - g_1(n)S_{n+1}}{g_1(n+1) - g_1(n)} = S + \sum_{i\geq 2} a_i\ \frac{g_1(n+1)g_i(n) - g_1(n)g_i(n+1)}{g_1(n+1) - g_1(n)}$$

Posons $$E_1^{(n)} = \frac{g_1(n+1)S_n - g_1(n)S_{n+1}}{g_1(n+1) - g_1(n)}, \forall n$$

et $$g_{1,i}^{(n)} = \frac{g_1(n+1)g_i(n) - g_1(n)g_i(n+1)}{g_1(n+1) - g_1(n)}, \forall n, \forall i \geq 2.$$

D'où $$E_1^{(n)} = S + \sum_{i\geq 2} a_i\, g_{1,i}^{(n)}, \forall n.$$

Nous obtenons la suite $(E_1^{(n)})$ sous une forme identique à celle de la suite de départ $(S_n)_{n\in\mathbb{N}}$. Nous pouvons reprendre le travail fait en III.1.1. afin d'éliminer a_2.

II.1.2. - Elimination de a_2

Soient

$$\left.\begin{aligned} E_1^{(n)} &= S + \sum_{i\geq 2} a_i\, g_{1,i}^{(n)} & \times\, g_{1,2}^{(n+1)} \\ E_1^{(n+1)} &= S + \sum_{i\geq 2} a_i\, g_{1,i}^{(n+1)} & \times\, g_{1,2}^{(n)} \end{aligned}\right\} \Rightarrow \ldots$$

$$\ldots\; g_{1,2}^{(n+1)} E_1^{(n)} - g_{1,2}^{(n)} E_1^{(n+1)} = (g_{1,2}^{(n+1)} - g_{1,2}^{(n)})S + \sum_{i\geq 3} a_i (g_{1,2}^{(n+1)} g_{1,i}^{(n)} - g_{1,2}^{(n)} g_{1,i}^{(n+1)})$$

Si $g_{1,2}^{(n+1)} - g_{1,2}^{(n)} \neq 0$, $\forall n$, on obtient

$$\frac{g_{1,2}^{(n+1)} E_1^{(n)} - g_{1,2}^{(n)} E_1^{(n+1)}}{g_{1,2}^{(n+1)} - g_{1,2}^{(n)}} = S + \sum_{i\geq 3} a_i \frac{g_{1,2}^{(n+1)} g_{1,i}^{(n)} - g_{1,2}^{(n)} g_{1,i}^{(n+1)}}{g_{1,2}^{(n+1)} - g_{1,2}^{(n)}}, \forall n$$

Posons

$$E_2^{(n)} = \frac{g_{1,2}^{(n+1)} E_1^{(n)} - g_{1,2}^{(n)} E_1^{(n+1)}}{g_{1,2}^{(n+1)} - g_{1,2}^{(n)}}, \forall n$$

$$g_{2,i}^{(n)} = \frac{g_{1,2}^{(n+1)} g_{1,i}^{(n)} - g_{1,2}^{(n)} g_{1,i}^{(n+1)}}{g_{1,2}^{(n+1)} - g_{1,2}^{(n)}}, \forall n, \forall i \geq 3.$$

D'où $\quad E_2^{(n)} = S + \sum_{i\geq 3} a_i\, g_{2,i}^{(n)}, \ \forall n$

La suite $(E_2^{(n)})_{n\in\mathbb{N}}$ a une forme identique à (S_n). Par un raisonnement inductif on construit des suites $(E_k^{(n)})_{n\in\mathbb{N}}$ ayant la forme de (S_n).

III.1.3. - Elimination de a_k

Supposons que l'on ait éliminé a_{k-1} et obtenu

$$E_{k-1}^{(n)} = S + \sum_{i\geq k} a_i\, g_{k-1,i}^{(n)}, \ \forall n.$$

Soient

$$\left.\begin{aligned} E_{k-1}^{(n)} &= S + \sum_{i\geq k} a_i\, g_{k-1,i}^{(n)} \qquad \times\ g_{k-1,k}^{(n+1)} \\ E_{k-1}^{(n+1)} &= S + \sum_{i\geq k} a_i\, g_{k-1,i}^{(n+1)} \times\ g_{k-1,k}^{(n)} \end{aligned}\right\} \Rightarrow \ldots$$

$$\ldots\ g_{k-1,k}^{(n+1)}E_{k-1}^{(n)} - g_{k-1,k}^{(n)}E_{k-1}^{(n+1)} = (g_{k-1,k}^{(n+1)} - g_{k-1,k}^{(n)})S$$

$$+ \sum_{i\geq k+1} a_i\,(g_{k-1,k}^{(n+1)}g_{k-1,i}^{(n)} - g_{k-1,k}^{(n)}g_{k-1,i}^{(n+1)})$$

Si $g_{k-1,k}^{(n+1)} - g_{k-1,k}^{(n)} \neq 0$, on obtient

$$\frac{g_{k-1,k}^{(n+1)}E_{k-1}^{(n)} - g_{k-1,k}^{(n)}E_{k-1}^{(n+1)}}{g_{k-1,k}^{(n+1)} - g_{k-1,k}^{(n)}} = S + \sum_{i\geq k+1} a_i\, \frac{g_{k-1,k}^{(n+1)}g_{k-1,i}^{(n)} - g_{k-1,k}^{(n)}g_{k-1,i}^{(n+1)}}{g_{k-1,k}^{(n+1)} - g_{k-1,k}^{(n)}}, \ \forall n$$

Posons $\quad E_k^{(n)} = \dfrac{g_{k-1,k}^{(n+1)}E_{k-1}^{(n)} - g_{k-1,k}^{(n)}E_{k-1}^{(n+1)}}{g_{k-1,k}^{(n+1)} - g_{k-1,k}^{(n)}}, \ \forall n$

$$g_{k,i}^{(n)} = \frac{g_{k-1,k}^{(n+1)}g_{k-1,i}^{(n)} - g_{k-1,k}^{(n)}g_{k-1,i}^{(n+1)}}{g_{k-1,k}^{(n+1)} - g_{k-1,k}^{(n)}}, \ \forall n, \ \forall i \geq k+1.$$

D'où $\qquad E_k^{(n)} = S + \sum_{i\geq k+1} a_i\, g_{k,i}^{(n)}, \ \forall n.$

Si $a_i = 0$, $\forall i \geq k+1$, nous obtenons la propriété fondamentale de cet algorithme

Propriété 1 :

Si $S_n = S + a_1\, g_1(n) +\ldots+ a_k\, g_k(n)$, $\forall n \geq N$

alors $E_k^{(n)} = S$, $\forall n \geq N$

La règle du E-algorithme se résume par :

$$(E) : \begin{cases} E_0^{(n)} = S_n,\ g_{0,i}^{(n)} = g_i(n),\ n = 0, 1, 2,\ldots,\ ;\ i = 1, 2, 3,\ldots \\[2ex] E_k^{(n)} = \dfrac{g_{k-1,k}^{(n+1)} E_{k-1}^{(n)} - g_{k-1,k}^{(n)} E_{k-1}^{(n+1)}}{g_{k-1,k}^{(n+1)} - g_{k-1,k}^{(n)}},\ n = 0, 1, 2,\ldots\ ;\ k = 1, 2,\ldots \\[2ex] g_{k,i}^{(n)} = \dfrac{g_{k-1,k}^{(n+1)} g_{k-1,i}^{(n)} - g_{k-1,k}^{(n)} g_{k-1,i}^{(n+1)}}{g_{k-1,k}^{(n+1)} - g_{k-1,k}^{(n)}},\ n = 0, 1, 2,\ldots\ ;\ k = 1, 2,\ldots \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad i = k+1,\ldots \end{cases}$$

C. Brézinski a obtenu cet algorithme en partant d'une suite (S_n) de la forme

$$S_n = S + a_1\, g_1(n) +\ldots+ a_k\, g_k(n),\ \forall n.$$

Il a considéré le système

$$S + a_1\, g_1(n+j) +\ldots+ a_k\, g_k(n+j) = S_{n+j},\ j = 0, 1,\ldots, k.$$

Ce qui donne

$$S = \begin{vmatrix} S_n & g_1(n) \ldots\ldots\ldots g_k(n) \\ S_{n+1} & g_1(n+1) \\ \vdots & \vdots \\ S_{n+k} & g_1(n+k) \ldots\ldots\ldots g_k(n+k) \end{vmatrix} \Big/ \begin{vmatrix} 1 & g_1(n) \ldots\ldots\ldots g_k(n) \\ \vdots & \vdots \\ 1 & g_1(n+k) \ldots\ldots\ldots g_k(n+k) \end{vmatrix}$$

Pour une suite (S_n) quelconque, il a posé

$$E_k^{(n)} = \begin{vmatrix} S_n & g_1(n) \ldots\ldots g_k(n) \\ \vdots & \\ S_{n+k} & g_1(n+k) \ldots\ldots g_k(n+k) \end{vmatrix} \Big/ \begin{vmatrix} 1 & g_1(n) \ldots\ldots\ldots g_k(n) \\ \vdots & \vdots \\ 1 & g_1(n+k) \ldots\ldots\ldots g_k(n+k) \end{vmatrix}$$

Grâce à l'identité de Sylvester, il a formulé les règles ci-dessus du E-algorithme.

Nous allons montrer par récurrence sur k et en partant des règles du E-algorithme que la quantité $E_k^{(n)}$ est donnée par un rapport de deux déterminants.

k=1

$$E_1^{(n)} = \frac{g_1(n+1)S_n - g_1(n)S_{n+1}}{g_1(n+1) - g_1(n)} = \frac{\begin{vmatrix} S_n & S_{n+1} \\ g_1(n) & g_1(n+1) \end{vmatrix}}{\begin{vmatrix} 1 & 1 \\ g_1(n) & g_1(n+1) \end{vmatrix}} \text{ et } g_{1,i}^{(n)} = \frac{\begin{vmatrix} g_i(n) & g_i(n+1) \\ g_1(n) & g_1(n+1) \end{vmatrix}}{\begin{vmatrix} 1 & 1 \\ g_1(n) & g_1(n+1) \end{vmatrix}}$$

k=2

$$E_2^{(n)} = \frac{g_{1,2}^{(n+1)} E_1^{(n)} - g_{1,2}^{(n)} E_1^{(n+1)}}{g_{1,2}^{(n+1)} - g_{1,2}^{(n)}} = \frac{\begin{vmatrix} g_2(n+1) & g_2(n+2) \\ g_1(n+1) & g_1(n+2) \end{vmatrix} \begin{vmatrix} S_n & S_{n+1} \\ g_1(n) & g_1(n+1) \end{vmatrix} - \cdots}{\begin{vmatrix} g_2(n+1) & g_2(n+2) \\ g_1(n+1) & g_1(n+2) \end{vmatrix} \begin{vmatrix} 1 & 1 \\ g_1(n) & g_1(n+1) \end{vmatrix} - \cdots}$$

$$\cdots \frac{\begin{vmatrix} g_2(n) & g_2(n+1) \\ g_1(n) & g_1(n+1) \end{vmatrix} \begin{vmatrix} S_{n+1} & S_{n+2} \\ g_1(n+1) & g_1(n+2) \end{vmatrix}}{\begin{vmatrix} g_2(n) & g_2(n+1) \\ g_1(n) & g_1(n+1) \end{vmatrix} \begin{vmatrix} 1 & 1 \\ g_1(n+1) & g_1(n+2) \end{vmatrix}}$$

Dans le premier déterminant et le troisième du numérateur (resp. du dénominateur) échangeons la première ligne et la dernière. Le résultat obtenu est égal à : (Identité de Sylvester).

$$\begin{vmatrix} S_n & S_{n+1} & S_{n+2} \\ g_1(n) & g_1(n+1) & g_1(n+2) \\ g_2(n) & g_2(n+1) & g_2(n+2) \end{vmatrix} \quad g_1(n+1) \text{ au numérateur}$$

et à

$$\begin{vmatrix} 1 & 1 & 1 \\ g_1(n) & g_1(n+1) & g_1(n+2) \\ g_2(n) & g_2(n+1) & g_2(n+2) \end{vmatrix} \quad g_1(n+1) \text{ au dénominateur.}$$

D'où

$$E_2^{(n)} = \frac{\begin{vmatrix} S_n & S_{n+1} & S_{n+2} \\ g_1(n) & g_1(n+1) & g_1(n+2) \\ g_2(n) & g_2(n+1) & g_2(n+2) \end{vmatrix}}{\begin{vmatrix} 1 & 1 & 1 \\ g_1(n) & g_1(n+1) & g_1(n+2) \\ g_2(n) & g_2(n+1) & g_2(n+2) \end{vmatrix}}$$

On fait le même raisonnement pour obtenir

$$g_{2,i}^{(n)} = \begin{vmatrix} g_i(n) & g_i(n+1) & g_i(n+2) \\ g_1(n) & g_1(n+1) & g_1(n+2) \\ g_2(n) & g_2(n+1) & g_2(n+2) \end{vmatrix} \Big/ \begin{vmatrix} 1 & 1 & 1 \\ g_1(n) & g_1(n+1) & g_1(n+2) \\ g_2(n) & g_2(n+1) & g_2(n+2) \end{vmatrix}$$

Supposons que pour $j \leq k-1$.

$$E_j^{(n)} = \begin{vmatrix} S_n & \cdots\cdots & S_{n+j} \\ g_1(n) & \cdots\cdots & g_1(n+j) \\ \cdots & \cdots & \cdots \\ g_j(n) & \cdots\cdots & g_j(n+j) \end{vmatrix} \Bigg/ \begin{vmatrix} 1 & \cdots\cdots & 1 \\ g_1(n) & \cdots\cdots & g_1(n+j) \\ \cdots & \cdots & \cdots \\ g_j(n) & \cdots\cdots & g_j(n+j) \end{vmatrix}$$

et

$$g_{j,i}^{(n)} = \begin{vmatrix} g_i(n) & \cdots\cdots & g_i(n+j) \\ g_1(n) & \cdots\cdots & g_1(n+j) \\ \cdots & \cdots & \cdots \\ g_j(n) & \cdots\cdots & g_j(n+j) \end{vmatrix} \Bigg/ \begin{vmatrix} 1 & \cdots\cdots & 1 \\ g_1(n) & \cdots\cdots & g_1(n+j) \\ \cdots & \cdots & \cdots \\ g_j(n) & \cdots\cdots & g_j(n+j) \end{vmatrix}$$

Montrons que ce résultat reste vrai pour k.

$$E_k^{(n)} = \frac{g_{k-1,k}^{(n+1)} E_{k-1}^{(n)} - g_{k-1,k}^{(n)} E_{k-1}^{(n+1)}}{g_{k-1,k}^{(n+1)} - g_{k-1,k}^{(n)}}$$

$$= \frac{\begin{vmatrix} g_k(n+1) & \cdots\cdots & g_k(n+k) \\ g_1(n+1) & \cdots\cdots & g_1(n+k) \\ \cdots & \cdots & \cdots \\ g_{k-1}(n+1) & \cdots\cdots & g_{k-1}(n+k) \end{vmatrix} \begin{vmatrix} S_n & \cdots\cdots & S_{n+k-1} \\ g_1(n) & \cdots\cdots & g_1(n+k-1) \\ \cdots & \cdots & \cdots \\ g_{k-1}(n) & \cdots\cdots & g_{k-1}(n+k-1) \end{vmatrix} - \cdots}{\begin{vmatrix} g_k(n+1) & \cdots\cdots & g_k(n+k) \\ g_1(n+1) & \cdots\cdots & g_1(n+k) \\ \cdots & \cdots & \cdots \\ g_{k-1}(n+1) & \cdots\cdots & g_{k-1}(n+k) \end{vmatrix} \begin{vmatrix} 1 & \cdots\cdots & 1 \\ g_1(n) & \cdots\cdots & g_1(n+k-1) \\ \cdots & \cdots & \cdots \\ g_{k-1}(n) & \cdots\cdots & g_{k-1}(n+k-1) \end{vmatrix} - \cdots}$$

$$\cdots \frac{\begin{vmatrix} g_k(n) & \cdots\cdots & g_k(n+k-1) \\ g_1(n) & \cdots\cdots & g_1(n+k-1) \\ \cdots & \cdots & \cdots \\ g_{k-1}(n) & \cdots\cdots & g_{k-1}(n+k-1) \end{vmatrix} \begin{vmatrix} S_{n+1} & \cdots\cdots & S_{n+k} \\ g_1(n+1) & \cdots\cdots & g_1(n+k) \\ \cdots & \cdots & \cdots \\ g_{k-1}(n+1) & \cdots\cdots & g_{k-1}(n+k) \end{vmatrix}}{\begin{vmatrix} g_k(n) & \cdots\cdots & g_k(n+k-1) \\ g_1(n) & \cdots\cdots & g_1(n+k-1) \\ \cdots & \cdots & \cdots \\ g_{k-1}(n) & \cdots\cdots & g_{k-1}(n+k-1) \end{vmatrix} \begin{vmatrix} 1 & \cdots\cdots & 1 \\ g_1(n+1) & \cdots\cdots & g_1(n+k) \\ \cdots & \cdots & \cdots \\ g_{k-1}(n+1) & \cdots\cdots & g_{k-1}(n+k) \end{vmatrix}}$$

Dans le numérateur (respectivement le dénominateur) échangeons dans le premier et le troisième déterminant la première et la dernière ligne. Appliquons l'identité de Sylvester au numérateur (respect. dénominateur) obtenu. Après simplification par

$$\begin{vmatrix} g_1(n+1) \ldots\ldots\ldots\ldots g_1(n+k-1) \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ g_{k-1}(n+1) \ldots\ldots\ldots g_{k-1}(n+k-1) \end{vmatrix}$$

On obtient

$$E_k^{(n)} = \begin{vmatrix} S_n & S_{n+1} \ldots\ldots S_{n+k} \\ g_1(n) \ldots\ldots\ldots\ldots g_1(n+k) \\ \ldots\ldots\ldots\ldots\ldots \\ g_k(n) \ldots\ldots\ldots\ldots g_k(n+k) \end{vmatrix} \bigg/ \begin{vmatrix} 1 \ldots\ldots\ldots\ldots\ldots 1 \\ g_1(n) \ldots\ldots\ldots\ldots g_1(n+k) \\ \ldots\ldots\ldots\ldots\ldots \\ g_k(n) \ldots\ldots\ldots\ldots g_k(n+k) \end{vmatrix}$$

Un raisonnement analogue pour $g_{k,i}^{(n)}$, $i \geq k+1$ donne :

$$g_{k,i}^{(n)} = \begin{vmatrix} g_i(n) \ldots\ldots\ldots\ldots g_i(n+k) \\ g_1(n) \ldots\ldots\ldots\ldots g_1(n+k) \\ \ldots\ldots\ldots\ldots\ldots \\ g_k(n) \ldots\ldots\ldots\ldots g_k(n+k) \end{vmatrix} \bigg/ \begin{vmatrix} 1 \ldots\ldots\ldots\ldots\ldots 1 \\ g_1(n) \ldots\ldots\ldots\ldots g_1(n+k) \\ \ldots\ldots\ldots\ldots\ldots \\ g_k(n) \ldots\ldots\ldots\ldots g_k(n+k) \end{vmatrix}$$

III.2. - INTERPRETATION GEOMETRIQUE DU E-ALGORITHME

Ce paragraphe est basé essentiellement sur l'article R.C. Johnson [14] concernant la transformation de Shanks. Nous avons étendu ses résultats au E-algorithme qui est en quelque sorte une généralisation de la transformation de Shanks. En effet il suffit de choisir $g_i(n) = \Delta S_{n+i-1}$ pour retrouver celle-ci. Deux constructions géométriques du E-algorithme sont exposées.

La première consiste en la recherche du point d'intersection entre le premier vecteur de la base canonique de $\mathbb{R}^{k+1}$ (soit D_1 ce vecteur) et l'hyperplan $P_k^{(U)}$ passant par les points :

$$\begin{bmatrix} S_{n+i} \\ g_1(n+i) \\ \cdot \\ \cdot \\ \cdot \\ g_k(n+i) \end{bmatrix} \qquad \text{pour } i = 0, 1, \ldots, k.$$

La seconde construction est obtenue à partir de la première par une "rotation" ramenant l'axe D_1 de $\mathbb{R}^{k+1}$ sur sa diagonale $\mathcal{D}$. Puis on cherche l'intersection de l'hyperplan image de $P_k^{(U)}$ par cette rotation avec $\mathcal{D}$. La première construction donne immédiatement $E_k^{(n)}$. La seconde peut aboutir à la détermination de $E_k^{(n)}$ si le vecteur normal à l'hyperplan satisfait à la condition de normalisation.

III.2.1. - Première construction géométrique

Soit $(S_n)_{n\in\mathbb{N}}$ une suite. Soit $(g_i)_{i\in\mathbb{N}}$ une famille de fonctions données de n.

On considère l'ensemble $S_n^{(k)} = \{S_n, S_{n+1}, \ldots, S_{n+k}\}$, et les k premières fonctions $g_i (i = 1, 2, \ldots, k)$ aux points n, n+1, ..., n+k.

Soient $u^{(m)} = (S_{n+m}, g_1(n+m), \ldots, g_k(n+m))^T$ pour $m = 0, 1, 2, \ldots, k$. $u^{(m)} \in \mathbb{R}^{k+1}$

Soit $P_k^{(U)}$ l'hyperplan euclidien passant par $u^{(m)}$, $m = 0, 1, \ldots, k$ (où l'on a supposé $u^{(m)}$, $m = 0, 1, \ldots, k$ forment un ensemble de k+1 points affinement indépendants).

$\dim P_k^{(U)} = k$ (la dimension de l'espace vectoriel associé).

Si $P_k^{(U)}$ existe et si $P_k^{(U)}$ rencontre l'axe D_1 de $\mathbb{R}^{k+1}$, alors ce point d'intersection est $E_k^{(n)}$ obtenu par application du E-algorithme à $S_n^{(k)}$. En effet ;

Soient $\Delta u^{(m)} = u^{(m+1)} - u^{(m)}$, $m = 0, 1, \ldots, k-1$. Le système $\{\Delta u^{(m)}\}_{m = 0, 1, \ldots, k-1}$ est une base de l'espace vectoriel associé à $P_k^{(U)}$.

Soit $p = (E, 0, \dots, 0)^T$ ou $E \in \mathbb{R}$. Soit enfin U le vecteur normal à $P_k^{(U)}$.

$$U^T \cdot \Delta u^{(m)} = 0$$
$$m = 0, 1, \dots, k$$

Si $p \in P_k^{(U)}$, $p-u^{(m)}$ $(m = 0, 1, \dots, k) \in$ l'espace vectoriel associé.

D'où :

$$\left.\begin{array}{l} U^T \cdot (p-u^{(m)}) = 0 \\ m = 0, 1, \dots, k \end{array}\right\} \Longleftrightarrow \left\{\begin{array}{l} U^T \cdot u^{(m)} = U^T \cdot p \\ m = 0, 1, \dots, k \end{array}\right.$$

$$\Longleftrightarrow \left\{\begin{array}{l} u_0 S_n + u_1 g_1(n) + \dots + u_k g_k(n) = u_0 E \\ \dots\dots\dots\dots\dots\dots\dots\dots \\ u_0 S_{n+k} + u_1 g_1(n+k) + \dots + u_k g_k(n+k) = u_0 E \end{array}\right.$$

où $(u_0, u_1, \dots, u_k)^T = U$.

Posons $a_i = -\dfrac{u_i}{u_0}$ $(u_0 \neq 0)$. On obtient le système :

$$\left\{\begin{array}{l} E + a_1 g_1(n) + \dots + a_k g_k(n) = S_n \\ \dots\dots\dots\dots\dots\dots\dots\dots \\ E + a_1 g_1(n+k) + \dots + a_k g_k(n+k) = S_{n+k} \end{array}\right.$$

Il est de k+1 équations à k+1 inconnues ; $E, a_1, \dots, a_k$.

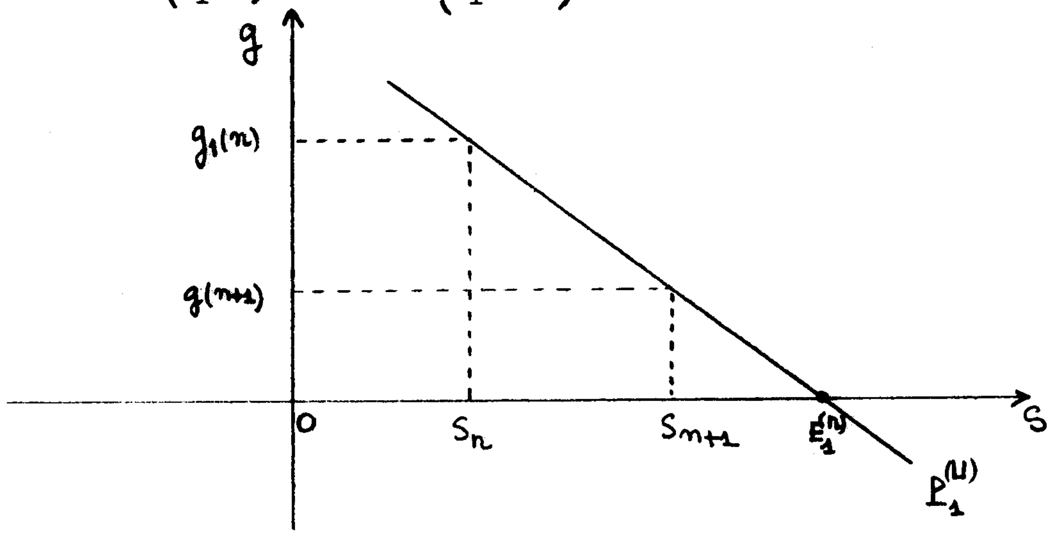
E est donné par

$$E = \begin{vmatrix} S_n & \dots\dots & S_{n+k} \\ g_1(n) & \dots\dots & g_1(n+k) \\ & & \\ g_k(n) & \dots\dots & g_k(n+k) \end{vmatrix} \Bigg/ \begin{vmatrix} 1 & \dots\dots & 1 \\ g_1(n) & \dots\dots & g_1(n+k) \\ & & \\ g_k(n) & \dots\dots & g_k(n+k) \end{vmatrix}$$

$\Rightarrow E = E_k^{(n)}$.

Dans le plan cartésien (k=1) on a :

$$u^{(0)} = \begin{pmatrix} S_n \\ \\ g_1(n) \end{pmatrix}, \quad u^{(1)} = \begin{pmatrix} S_{n+1} \\ \\ g_1(n+1) \end{pmatrix}$$



$E_1^{(n)}$ est une extrapolation linéaire en g = 0.

III.2.2. - Deuxième construction géométrique

Soit R l'application linéaire définie par la matrice

$$R = \begin{bmatrix} 1 & 0 & \cdots & \cdots & 0 \\ 1 & 1 & 0 & \cdots & 0 \\ \vdots & & \ddots & & \\ \vdots & & & \ddots & \\ 1 & \cdots & \cdots & \cdots & 1 \end{bmatrix} \in M_{(k+1,k+1)}(\mathbb{R}).$$

Posons $P_k^{(V)} = R\ P_k^{(U)}$, $v^{(m)} = R\ u^{(m)}$, $q = Rp$.

$$\dim P_k^{(V)} = \dim P_k^{(U)} \text{ (car R est inversible).}$$

R transforme l'axe D_1 de $\mathbb{R}^{k+1}$ en sa diagonale $\mathcal{D}$.

Si $P_k^{(V)}$ rencontre $\mathcal{D}$, alors ce point d'intersection est

$$q = Rp = E(1,\dots,1)^T,\ E \in \mathbb{R}.$$

En effet, soit V le vecteur normal de $P_k^{(V)}$. On peut écrire

$$V^T \cdot v^{(k)} = \dots = V^T \cdot v^{(0)} = V^T \cdot q$$

ou encore

$$V^T R\, u^{(k)} = \dots = V^T R\, u^{(0)} = V^T Rp$$

$$\Rightarrow (R^T V)^T u^{(k)} = \dots = (R^T V)^T u^{(0)} = (R^T V)^T p$$

$\Rightarrow R^T V$ est normal à $P_k^{(U)}$, on peut donc prendre

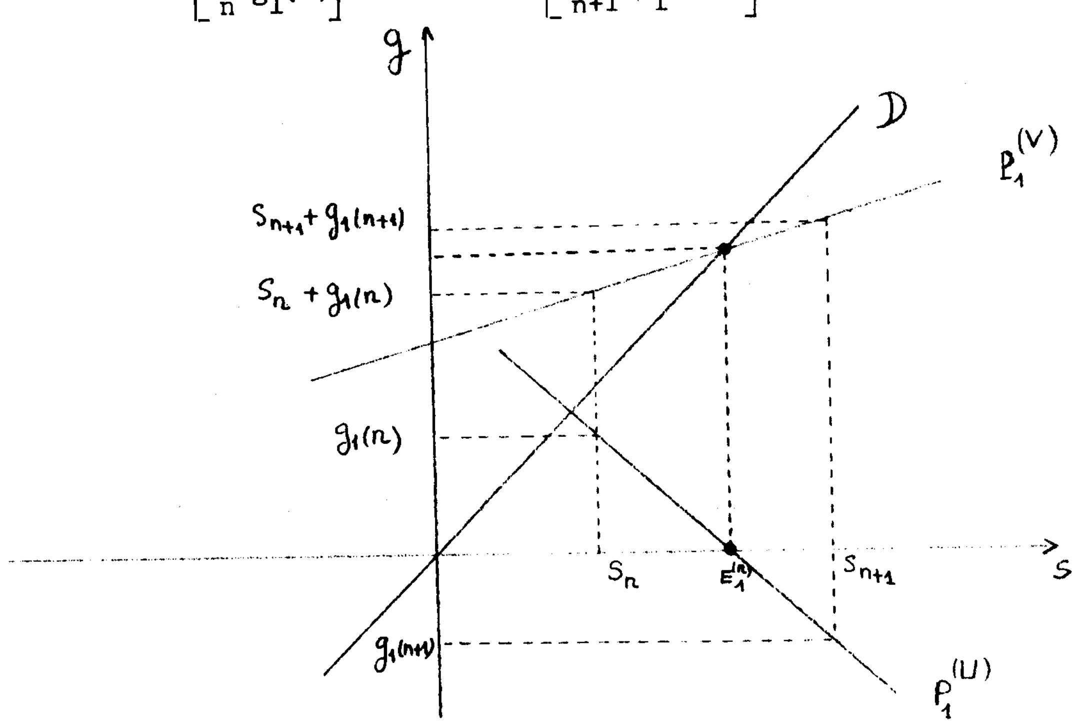
$$U = R^T V$$

$$\Rightarrow V = (R^T)^{-1} U$$

Donc $(R^T)^{-1}\, U$ est normal à $P_k^{(V)}$.

Dans le plan cartésien on a :

$$v^{(0)} = \begin{bmatrix} S_n \\ \\ S_n + g_1(n) \end{bmatrix}, \quad v^{(1)} = \begin{bmatrix} S_{n+1} \\ \\ S_{n+1} + g_1(n+1) \end{bmatrix}$$



Le vecteur $v^{(m)}$ est donné par

$$v^{(m)} = (S_{n+m}, S_{n+m} + g_1(n+m), \ldots, S_{n+m} + \sum_{i=1}^{k} g_i(n+m))^T.$$

Remarque :

L'application G définie par la matrice

$$G = \begin{bmatrix} 0 & \cdots & 0 & 1 \\ & & \cdot^{\cdot^{\cdot}} & \\ 1 & 0 & \cdots & 0 \end{bmatrix} \in \mathcal{M}(k+1,k+1)(\mathbb{R})$$

laisse invariant tout point de la diagonale $\mathcal{D}$ de $\mathbb{R}^{k+1}$. L'hyperplan $P_k'^{(V)} = G\, P_k^{(V)}$ passe par les points

$$v'^{(m)} = (S_{n+m} + \sum_{i=1}^{k} g_i(n+m), S_{n+m} + \sum_{i=1}^{k-1} g_i(n+m), \ldots, S_{n+m})^T, m = 0, 1, 2, \ldots, k$$

Il rencontre $\mathcal{D}$ en

$$q = E_k^{(n)}(1, \ldots, 1)^T.$$

Posons $S'_{n+m} = S_{n+m} + \sum_{i=1}^{k} g_i(n+m)$. $v'^{(m)}$ devient

$$v'^{(m)} = (S'_{n+m}, S'_{n+m} + g'_1(n+m), \ldots, S'_{n+m} + \sum_{i=1}^{k} g'_i(n+m))$$

où $\quad g'_i(n+m) = - g_{k+1-i}(n+m).$

En appliquant le E-algorithme à la suite (S'_n) avec la famille g'_i. on obtient :

$$E_k^{(n)'} = E_k^{(n)}.$$

où

$$E_k^{(n)'} = \begin{vmatrix} S_n + \sum_{i=1}^{k} g_i(n) & - g_k(n) \ldots\ldots -g_1(n) \\ \ldots\ldots & \ldots\ldots \\ S_{n+k} + \sum_{i=1}^{k} g_i(n+k) & - g_k(n+k) \ldots -g_1(n+k) \end{vmatrix} \Bigg/ \begin{vmatrix} 1 & -g_k(n) \ldots\ldots -g_1(n) \\ \ldots & \ldots\ldots \\ 1 & -g_k(n+k) \ldots -g_1(n+k) \end{vmatrix}$$

$$= \begin{vmatrix} S_n & g_1(n) \ldots\ldots\ldots g_k(n) \\ \ldots\ldots & \ldots\ldots\ldots\ldots\ldots \\ S_{n+k} & g_1(n+k) \ldots\ldots g_k(n+k) \end{vmatrix} \Big/ \begin{vmatrix} 1 & g_1(n) \ldots\ldots\ldots\ldots\ldots g_k(n) \\ \ldots & \ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ 1 & g_1(n+k) \ldots\ldots\ldots\ldots g_k(n+k) \end{vmatrix}$$

$$= E_k^{(n)}.$$

Ce résultat est loin d'être évident sur la première construction.

Définition :

Le vecteur V normal à $P_k^{(V)}$ satisfait à la condition de normalisation si $\sum_{i=0}^{k} V_i = 1$ où les V_i sont les composantes de V.

Posons $G_i(n) = \sum_{j=0}^{i} g_j(n)$ avec $g_0(n) = 0$.

V est normal à $P_k^{(V)}$ donne ;

$$\left.\begin{array}{l} V^T v^{(m)} = V^T q \\ m = 0, \ldots, k \end{array}\right\} \Longleftrightarrow \left\{\begin{array}{l} V_0(S_{n+m}+G_0(n+m)) + \ldots + V_k(S_{n+m}+G_1(n+m)) = (\sum_{i=0}^{k} V_i)\, E \\ m = 0, 1, \ldots, k \end{array}\right.$$

La condition de normalisation nous permet d'écrire :

$$\left\{\begin{array}{l} S_{n+m} + V_1\, G_1(n+m) + \ldots + V_k\, G_k(n+m) = E \\ V_0 = 1 - \sum_{i=1}^{k} V_i \\ m = 0, 1, \ldots, k \end{array}\right.$$

$$\left\{\begin{array}{l} E - V_1\, G_1(n+m) - \ldots - V_k\, G_k(n+m) = S_{n+m} \\ m = 0, 1, \ldots, k \\ V_0 = 1 - \sum_{i=1}^{k} V_i \end{array}\right.$$

Posons $p_1 = -V_1, \ldots, p_k = -V_k$. D'où le système

$$\begin{cases} E + p_1\, G_1(n) + \ldots + p_k\, G_k(n) = S_n \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ E + p_1\, G_1(n+k) + \ldots + p_k\, G_k(n+k) = S_{n+k} \end{cases}$$

Il est de k+1 équations à k+1 inconnues ; $E, p_1, \ldots, p_k$.

$$\Rightarrow E = \begin{vmatrix} S_n \ldots\ldots\ldots S_{n+k} \\ G_1(n) \ldots\ldots\ldots G_1(n+k) \\ \ldots\ldots\ldots\ldots\ldots \\ G_k(n) \ldots\ldots\ldots G_k(n+k) \end{vmatrix} \Big/ \begin{vmatrix} 1 \ldots\ldots\ldots 1 \\ G_1(n) \ldots\ldots\ldots G_1(n+k) \\ \ldots\ldots\ldots\ldots\ldots \\ G_k(n) \ldots\ldots\ldots G_k(n+k) \end{vmatrix}$$

or $G_{i+1}(m) - G_i(m) = g_{i+1}(m)$. Dans le numérateur (resp. le dénominateur) de ce dernier rapport retranchons la deuxième ligne à la troisième et la $i^{\text{ième}}$ ligne à la $(i+1)^{\text{ième}}$ pour $3 \leq i \leq k-1$. On obtient :

$$E = \begin{vmatrix} S_n \ldots\ldots\ldots S_{n+k} \\ g_1(n) \ldots\ldots\ldots g_1(n+k) \\ \ldots\ldots\ldots\ldots\ldots \\ g_k(n) \ldots\ldots\ldots g_k(n+k) \end{vmatrix} \Big/ \begin{vmatrix} 1 \ldots\ldots\ldots 1 \\ g_1(n) \ldots\ldots\ldots g_1(n+k) \\ \ldots\ldots\ldots\ldots\ldots \\ g_k(n) \ldots\ldots\ldots g_k(n+k) \end{vmatrix}$$

Donc $E = E_k^{(n)}$.

Par conséquent nous aboutissons à la détermination de la quantité $E_k^{(n)}$ donnée par le E-algorithme appliqué à $s_n^{(k)} = \{S_n, \ldots, S_{n+k}\}$.

Inversement soit $(S_n)_{n\in\mathbb{N}}$ donnée par

$$S_n = S + a_1\, g_1(n) + \ldots + a_k\, g_k(n),\ \forall n.$$

Nous savons que $E_k^{(n)} = S$, $\forall n$.

Nous allons montrer que cette deuxième construction donne le même résultat.

$$S = S_n - a_1\, g_1(n) - \ldots - a_k\, g_k(n),\ \forall n.$$

$$\left.\begin{array}{l} V^T v^{(m)} = V^T q \\ \\ m = 0, 1, \ldots, k \end{array}\right\} \Longleftrightarrow \left\{\begin{array}{l} (\sum_{i=0}^{k} V_i) S_{n+m} + (\sum_{i=1}^{k} V_i) g_1(n+m) + \ldots + (\sum_{i=j}^{k} V_i)\ g_j(n+m) + \ldots \\ \qquad \ldots + V_k\ g_k(n+m) = (\sum_{i=0}^{k} V_i)\ E \end{array}\right.$$

On peut choisir V tel que :

$$\left\{\begin{array}{l} \sum_{i=0}^{k} V_i = 1 \\ \\ \sum_{i=1}^{k} V_i = -\ a_1 \\ \\ \ldots\ldots\ldots\ldots \\ \\ \qquad V_k = -\ a_k \end{array}\right.$$

$$\Longrightarrow \left\{\begin{array}{l} V_0 = 1 + a_1 \\ V_1 = a_2 - a_1 \\ V_2 = a_3 - a_2 \\ \ldots\ldots\ldots\ldots \\ V_{k-1} = a_k - a_{k-1} \\ V_k = -\ a_k \end{array}\right.$$

Le vecteur normal obtenu ne dépend pas de n. L'hyperplan $P_k^{(V)}$, ainsi obtenu, passe par $S(1,\ldots, 1)^T$, $\forall n$.

<u>Exemple</u> :

Soit la suite géométrique

$S_r = a + b_1\ \rho_1^r + \ldots + b_k\ \rho_k^r \quad (\rho_i \neq 1,\ \forall i).$

Choisissons $g_i(r) = \rho_i^r$ pour $i = 1, 2, \ldots, k$.

Le vecteur normal V est donné par :

$$V = (1 + b_1,\ b_2 - b_1, \ldots,\ b_k - b_{k-1},\ -\ b_k)^T.$$

$$(1+b_1)S_r + (b_2-b_1)(S_r+\rho_1^r) + \ldots + (b_k-b_{k-1})(S_r+\sum_{i=1}^{k-1} \rho_i^r) + (-b_k)(\sum_{i=1}^{k} \rho_i^r + S_r) = a.$$

$\Longrightarrow E_{k'}^{(r)} = a$, $\forall r$, $\forall k'$ tel que $k' \geq k$.

Soit $\{S_r'\}$ la suite donnée par

$$S_r' = S_r + \rho_1^r + \ldots + \rho_k^r.$$

$$\Longrightarrow S_r' = a + (1+b_1)\rho_1^r + \ldots + (1+b_k)\rho_k^r.$$

$$= a + a_1' \, g_1'(r) + \ldots + a_k' \, g_k'(r)$$

où $a_i' = -1 - b_{k+1-i}$ et où $g_i'(r) = -\rho_{k+1-i}^r$ pour $i = 1, \ldots, k$.

Le vecteur normal V' est

$$V' = (-b_k, b_k-b_{k-1}, \ldots, b_2-b_1, 1+b_1)^T.$$

L'équation de l'hyperplan $P_k'^{(V)}$ est

$$-b_k \, x_0 + (b_k-b_{k-1}) \, x_1 + \ldots + (b_2-b_1) \, x_{k-1} + (b_1+1) \, x_k = a$$

$$\Longrightarrow -b_k \, S_r' + (b_k-b_{k-1})(S_r'-\rho_k^r) + \ldots + (b_2-b_1)(S_r'-\sum_{i=k}^{2} \rho_i^r) + (b_1+1)(S_r'-\sum_{i=k}^{1} \rho_i^r) = a.$$

Si on note $E_k'^{(n)}$ le résultat obtenu par application du E-algorithme à S_r' avec $g_i'(r) = -g_{k+1-i}(r)$, on obtient :

$$E_k'^{(n)} = E_k^{(n)}.$$

III.2.3. - Point fixe et E-algorithme

D'après cette deuxième construction, le E-algorithme peut servir à estimer le point fixe d'une application. En effet cet algorithme est une généralisation des procédés d'extrapolation qui, nous l'avons vu dans les chapitres précédents, fournissent des méthodes itératives de point fixe. Il va donc nous servir à obtenir de nouvelles méthodes itératives pour ce problème.

Par exemple si nous choisissons $g_i(n) = (\Delta S_n)^i$, le E-algorithme se réduit à l'algorithme (B') (Chapitre I) qui engendre une méthode itérative d'un ordre arbitraire (Chapitre II).

Si maintenant, nous prenons $g_i(n) = \Delta S_{n+i-1}$, le E-algorithme se réduit à la transformation de Shanks $e_k(S_n)$. Si $S_{n+1} = F(S_n)$ et F suffisamment dérivable dans un voisinage de S (son point fixe) Johnson [12] a montré grâce à la deuxième construction que

$$e_k^{(n)} = S + O(S_n - S)^{k+1},$$

En effet, posons $e_n = S_n - S \Rightarrow S_n = S + e_n$ (2.1)

$$S_{n+1} = S + \sum_{i\geq 1} a_i\, e_n^i \tag{2.2}$$

Retranchons à (2.2), le produit de (2.1) par a_1. On obtient

$$S_{n+1} - a_1\, S_n = (1-a_1)S + \sum_{i\geq 2} a_i\, e_n^i \tag{2.3}$$

$$S_{n+2} - a_1\, S_{n+1} = (1-a_1 S + \sum_{i\geq 2} a_i\, e_{n+1}^i$$

$$= (1-a_1)S + a_1^2\, a_2\, e_n^2 + \ldots \tag{2.4}$$

Soustrayons à (2.4), le produit de 2.3) par a_1^2. Cela donne

$$S_{n+2} - a_1\, (1+a_1)S_{n+1} + a_1^3 S_n = (1-a^1)(1-a_1^2)S + a_3^{(3)}\, e_n^3 + \ldots$$

ou encore :

$$S_{n+2} - C_1^{(2)}\, S_{n+1} + C_2^{(2)}\, S_n = \left[\prod_{i=1}^{2}(1-a_1^i)\right] S + a_3^{(3)}\, e_n^3 + \ldots \tag{2.5}$$

Supposons que pour $1 \leq j \leq k-1$, on ait

$$S_{n+j}-C_1^{(j)}S_{n+j-1}+\dots+(-1)^i C_i^{(j)} S_{n+j-i}+\dots+(-1)^j C_j^{(j)} S_n = \left[\prod_{i=1}^{j}(1-a_1^i)\right] S + a_{j+1}^{(j+1)} e_n^{j+1}+\dots \qquad (2.6)$$

$$S_{n+j+1}-C_1^{(j)}S_{n+j}+\dots+(-1)^i C_i^{(j)} S_{n+j-i+1}+\dots$$

$$\dots+(-1)^j C_j^{(j)} S_{n+1} = S\prod_{i=1}^{j}(1-a_1^i)+a_1^{j+1} a_{j+1}^{(j+1)} e_n^{j+1}+\dots \qquad (2.7)$$

formons maintenant (2.7) - a_1^{j+1} (2.6) $\Rightarrow$

$$S_{n+j+1}+(C_1^{(j)}+a_1^{j+1})\ S_{n+j}+\dots+(-1)^i\ (C_i^{(j)}+a_1^{j+1}\ C_{i-1}^{(j)})\ S_{n+j-i+1}+\dots$$

$$+(-1)^{j+1}\ a_1^{j+1}\ C_j^{(j)}\ S_n = S\prod_{i=1}^{j+1}(1-a_1^i)+a_{j+2}^{(j+2)}\ e_n^{j+2}+\dots \qquad (2.8)$$

Dans l'égalité (2.8) posons

$$j+1 = k$$

$$C_1^{(j)}+a_1^{j+1} = C_1^{(k)}$$

....

$$C_i^{(j)}+a_1^{j+1}\ C_{i-1}^{(j)} = C_i^{(k)}$$

....

$$a_1^{j+1}\cdot C_j^{(j)} = C_k^{(k)},$$

d'où

$$C_k^{(k)}\ (-1)^k\ S_n+\dots+(-1)^i\ C_i^{(k)}\ S_{n+k-i}+\dots+S_{n+k} = S\prod_{i=1}^{k}(1-a_1^i)+a_{k+1}^{(k+1)}\ e_n^{k+1}+\dots \qquad (2.9)$$

La deuxième construction, pour $g_i(n) = \Delta S_{n+i-1}$, correspond aux vecteurs $v^{(m)} = (S_{n+m}, S_{n+m+1},\dots, S_{n+m+k})^T$. L'hyperplan $P_k^{(V)}$, dans ce cas, admet comme vecteur normal

$$V = \frac{1}{\prod_{i=1}^{k}(1-a_1^i)}\ ((-1)^k\ C_k^{(k)},\dots,\ (-1)^i\ C_i^{(k)},\dots,\ -\ C_1^{(k)},\ 1)^T.$$

Si $P_k^{(V)}$ rencontre D en $E_k^{(n)}(1,\dots, 1)^T$, alors

$$\frac{1}{\prod_{i=1}^{k}(1-a_1^i)} \left\{(-1)^k C_k^{(k)} S_{n+m} + \ldots + (-1)^i C_i^{(k)} S_{n+m+k-i} + \ldots + S_{n+m+k}\right\}$$

$$= E_k^{(n)} \qquad (2.10) \; ; \; m = 0, 1, \ldots, k.$$

Les égalités (2.9) et (2.10) donnent

$$e_k^{(n)} = E_k^{(n)} = S + \frac{a_{k+1}^{(k+1)}}{\prod_{i=1}^{k}(1-a_1^i)} e_n^{k+1} + \ldots$$

Le E-algorithme ($g_i(n) = \Delta S_{n+i-1}$) ou la transformation de Shanks nous fournit une méthode itérative d'un ordre arbitraire pour résoudre des équations non linéaires. On a la méthode :

$x = F(x)$

x_0 donné

La $n^{\text{ième}}$ itération se déroule de la manière suivante :

$u_0 = x_{n-1}$

$u_{i+1} = F(u_i)$ pour $i = 0, 1, 2, \ldots, 2k-1.$

On applique

soit l'ε-algorithme : $\varepsilon_0^{(i)} = u_i$, $i = 0, 1, \ldots, 2k.$

soit le E-algorithme : $E_0^{(i)} = u_i$, $g_{0,j}^{(i)} = \Delta u_{i+j-1}$, $j = 1, 2, \ldots, k$

et $i = 0, 1, \ldots, k.$

On prend

$x_n = e_k^{(0)} = \varepsilon_{2k}^{(0)} = E_k^{(0)}.$

D'après l'étude précédente, si F est de classe C^{k+1} dans un voisinage de x_*

$$x_n - x_* = O(x_{n-1} - x_*)^{k+1}.$$

L'indice d'efficacité est donné par

$$I_k = (k+1)^{\frac{1}{2k}}.$$

Si k = 1, elle se réduit à la méthode de Steffensen qui est d'ordre deux et d'indice d'efficacité $I_1 = \sqrt{2} = 1.414...$

Remarque :

Soit f une fonction telle que

$$f(x_*) = 0$$
$$\text{et } F(x) = x + f(x).$$

Le choix $g_j(i) = \Delta u_{i+j-1}$ devient d'une façon naturelle

$$g_j(i) = f(u_{i+j-1}).$$

Dans ce cas nous proposons la méthode suivante :

$$x_0,\ x_1, \ldots,\ x_{2k-1} \text{ donnés.}$$

La $n^{\text{ième}}$ itération est ($2k \leq n$).

$$\left[\begin{array}{l}
E_0^{(i)} = x_{n-2k+i},\ i = 0, 1, \ldots, k \\
g_{0,j}^{(i)} = f(x_{n-2k-1+i+j}),\ i = 0, 1, \ldots, k \text{ et } j = 1, 2, \ldots, k. \\
E_j^{(i)} = \dfrac{g_{j-1,j}^{(i+1)} E_{j-1}^{(i)} - g_{j-1,j}^{(i)} E_{j-1}^{(i+1)}}{g_{j-1,j}^{(i+1)} - g_{j-1,j}^{(i)}},\ j = 1, 2, \ldots, k \text{ et } i = 0, 1, \ldots, k-j \\
g_{\ell,j}^{(i)} = \dfrac{g_{\ell-1,\ell}^{(i+1)} g_{\ell-1,j}^{(i)} - g_{\ell-1,\ell}^{(i)} g_{\ell-1,j}^{(i+1)}}{g_{\ell-1,\ell}^{(i+1)} - g_{\ell-1,\ell}^{(i)}},\ \ell = 0, \ldots, k-1\ ;\ j = \ell+1, \ldots, k\ ; \\
\qquad\qquad\qquad\qquad\qquad\qquad i = 0, 1, \ldots, k-j
\end{array}\right.$$

$$\left[\begin{array}{l}\text{On prend}\\ x_n = E_k^{(0)}.\end{array}\right.$$

Si k = 1 cette méthode se réduit à la méthode de la sécante. Si k ≥ 2, elle peut être considérée comme une extension de la sécante. Mais pour k = 2 nous allons voir qu'elle n'est pas intéressante.

Etude de la méthode basée sur la deuxième colonne du E-algorithme :

x_0, x_1, x_2, x_3 donnés.

Pour $n \geq 0$, x_{n+4} est donné par

$$x_{n+4} = \begin{vmatrix} x_n & x_{n+1} & x_{n+2} \\ f_n & f_{n+1} & f_{n+2} \\ f_{n+1} & f_{n+2} & f_{n+3} \end{vmatrix} \Big/ \begin{vmatrix} 1 & 1 & 1 \\ f_n & f_{n+1} & f_{n+2} \\ f_{n+1} & f_{n+2} & f_{n+3} \end{vmatrix}$$

Posons $e_i = x_i - x_*$ pour $i = n, n+1, n+2, n+3, n+4$.

Supposons $e_{i+1} = C\, e_i^r$ (C une constante positive) où r est l'ordre de cette méthode. Supposons $r > 1$, et $r \neq 2$.

$$e_{n+1} = Ce_n^r$$

$$e_{n+2} = C\, e_{n+1}^r = C^{r+1}\, e_n^{r^2}$$

$$e_{n+3} = C\, e_{n+2}^r = C^{r+1}\, e_{n+1}^{r^2} = C^{r^2+r+1}\, e_n^{r^3}$$

$$e_{n+4} = C\, e_{n+3}^r = C^{r+1}\, e_{n+2}^{r^2} = C^{r^2+r+1}\, e_{n+1}^{r^3} = C^{r^3+r^2+r+1}\, e_n^{r^4}.$$

$$e_{n+4} = \begin{vmatrix} e_n & e_{n+1} & e_{n+2} \\ f_n & f_{n+1} & f_{n+2} \\ f_{n+1} & f_{n+2} & f_{n+3} \end{vmatrix} \Big/ \begin{vmatrix} 1 & 1 & 1 \\ f_n & f_{n+1} & f_{n+2} \\ f_{n+1} & f_{n+2} & f_{n+3} \end{vmatrix} = \frac{N}{D}$$

D peut s'écrire (en négligeant dans $f_i = \Sigma\, c_j\, e_i^j$, les puissances e_i^j, $j \geq 2$) sous la forme :

$$D = C_1^2 \begin{vmatrix} 1 & 1 & 1 \\ e_n & e_{n+1} & e_{n+2} \\ e_{n+1} & e_{n+2} & e_{n+3} \end{vmatrix} = C_1^2 \, C \, e_n^{r+1} \begin{vmatrix} 1 & 1 & 1 \\ 1 & C\xi & C^{r+1}\xi^{r+1} \\ 1 & C^r\xi^r & C^{r^2+r}\xi^{r(r+1)} \end{vmatrix}$$

où $\xi = e_n^{r-1}$.

On obtient

$$D = - C_1^2 \, C^2 \, e_n^{2r} + O(e_n^{r^2+1})$$

Dans N on développe la deuxième ligne jusqu'à l'ordre deux pour éviter la coolinéarité des deux premières lignes. D'où :

$$N = C_1 \, C_2 \begin{vmatrix} e_n & e_{n+1} & e_{n+2} \\ e_n^2 & e_{n+1}^2 & e_{n+2}^2 \\ e_{n+1} & e_{n+2} & e_{n+3} \end{vmatrix} = C_1 \, C_2 \, e_n \, e_{n+1} \, e_{n+2} \begin{vmatrix} 1 & 1 & 1 \\ e_n & e_{n+1} & e_{n+2} \\ \dfrac{e_{n+1}}{e_n} & \dfrac{e_{n+2}}{e_{n+1}} & \dfrac{e_{n+3}}{e_{n+2}} \end{vmatrix}$$

$$= C_1 \, C_2 \, e_n \, e_{n+1} \, e_{n+2} \, N' \text{ où } N' = \begin{vmatrix} 1 & 1 & 1 \\ e_n & e_{n+1} & e_{n+2} \\ \dfrac{e_{n+1}}{e_n} & \dfrac{e_{n+2}}{e_{n+1}} & \dfrac{e_{n+3}}{e_{n+2}} \end{vmatrix}.$$

$$N' = C \, e_n^r \begin{vmatrix} 1 & 1 & 1 \\ 1 & C\xi & C^{r+1}\xi^{r+1} \\ 1 & C^{r-1}\xi^{r-1} & C^{r^2-1}\xi^{(r-1)(r+1)} \end{vmatrix} \qquad \text{où } \xi = e_n^{r-1}.$$

Afin de développer N' suivant les puissances réelles de ξ, nous distinguons deux cas.

<u>1^{er} cas</u> : $r < 2 \Rightarrow r-1 < 1$

ξ petit $\Rightarrow$ ξ est négligeable devant ξ^{r-1}, ξ^{r+1} devant $\xi^{(r+1)(r-1)}$, $\xi^{(r+1)(r-1)}$ devant ξ^{r-1}, ξ^{r^2} devant ξ^{r-1}.

N' s'écrit :

$$N' = C\, e_n^r \{C^{r-1}\xi^{r-1} + O(\xi)\}$$

$$= C^r\, e_n^{r^2-r+1} + O(e_n^{2r-1}).$$

N devient

$$N = C_1\, C_2\, C^{r+2}\, e_n^{r^2+r+1}\, N'$$

$$= C_1\, C_2\, C^{2r+2}\, e_n^{2r^2+2} + O(e_n^{r^2+3r}).$$

e_{n+4} s'exprime comme

$$e_{n+4} = \frac{N}{D} = \frac{C_1\, C_2\, C^{2r+2}\, e_n^{2r^2+2}}{-\, C_1^2\, C^2\, e_n^{2r}} = -\frac{C_2}{C_1}\, C^{2r}\, e_n^{2r^2-2r+2}$$

Or $e_{n+4} = C^{r^3+r^2+r+1}\, e_n^{r^4}$

D'où

$-\frac{C_1}{C_2}\, C^{r^3+r^2-r+1} = e_n^{2-2r+2r^2-r^4}$. Cette égalité ne peut être satisfaite que si

$$P(r) = r^4 - 2r^2 + 2r - 2 = 0.$$

Si r existe il doit donc satisfaire

$$1 < r < 2$$
$$\text{et } P(r) = 0$$

Considérons la fonction $P(t) = t^4 - 2t^2 + 2t - 2.$

$$P'(t) = 4t^3 - 4t + 2$$

$$P''(t) = 12t^2-4 = 4(3t^2-1) = 12(t + \sqrt{1/3})(t - \sqrt{1/3}).$$

On a le tableau de variation suivant :

t	0		$\sqrt{1/3}$		1		$+\infty$
$P''(t)$	$-$		0	$+$	8	$+$	
$P'(t)$	$+2$	$\searrow$	$+\ 0.48$	$\nearrow$	$+2$	$\nearrow$	$+\infty$
Signe $(P'(t))$	$+$					$+$	
$P(t)$	-2		$\nearrow$				$+\infty$

P est strictement croissante sur $[0, +\infty[$. D'après ce tableau P rencontre l'axe des abscisses en un point unique.

Notons ce point par r.

$P(1) = -1$, $P(3/2) = 0.87$.

$r \in]1, 3/2[$.

Nous concluons, si $r < 2$, que cette méthode est moins intéressante que la méthode de la sécante puisque celle-ci est d'ordre 1.618.

<u>$2^{\text{ième}}$ cas</u> : $r > 2$.

On vient de voir que

$$e_{n+4} = C_1\ C_2\ e_n\ e_{n+1}\ e_{n+2}\ \frac{N'}{D}.$$

D n'est pas modifié. Par contre N' devient

$$N' = C^2\ e_n^{r^2-r} + O(e_n^{r^2-r+1}).$$

Cela est dû au fait que ξ^{r-1} est négligeable devant ξ et que ξ^{α} $(\alpha > 1)$ est négligeable devant ξ.

$$\left.\begin{array}{l} \text{D'où } e_{n+4} = \dfrac{C_2}{C_1}\, e_{n+1}\, e_{n+2} \\[2ex] e_{n+4} = C^{r^2+r+1}\, e_{n+1}^{r^3} \end{array}\right\} \Longrightarrow \ldots$$

$$\ldots \; \frac{C_1}{C_2}\, C^{r^2+r} = e_{n+1}^{1+r-r^3}.$$

Cette égalité ne peut se produire que si

$$Q(r) = r^3-r-1 = 0.$$

Si r existe dans ce deuxième cas, il doit satisfaire :

$$r > 2$$
$$\text{et } Q(r) = 0.$$

Soit la fonction $Q(t) = t^3-t-1$.

$Q'(t) = 3t^2-1 = 3(t + \sqrt{1/3})(t - \sqrt{1/3})$.

On a le tableau de variation suivant :

t	0		$\sqrt{1/3}$		1		2		$+\infty$
$Q'(t)$	-1	$-$	0	$+$			$+$		
$Q(t)$		$\searrow$	$\searrow$		-1	$\nearrow$	$+5$	$\nearrow$	$+\infty$

D'après ce tableau r annulant Q appartient à]-1, 2[. Donc il n'existe pas de r vérifiant

$$r > 2$$
$$\text{et } Q(r) = 0.$$

Par conséquent ce cas est à rejeter.

En résumé, l'ordre r de cette méthode est la seule racine réelle positive de $P(t) = t^4-2t^2+2t-2 = 0$. Soit $r = 1.286...$

Remarque :

Nous avons supposé $r \neq 2$. Si $r = 2$ nous arrivons à

$e_n^9 \sim C^{te}$. Ce qui est impossible puisque e_n tend vers zéro.

En effet :

$$e_{n+1} = C\, e_n^2 \; ; \; e_{n+2} = C^3\, e_n^4 \; ; \; e_{n+3} = C^7\, e_n^8 \; ; \; e_{n+4} = C^{15}\, e_n^{16}.$$

Avec $g_i(n) = f_{n+i-1}$, on a :

$$x_{n+4} = \frac{g_{1,2}^{(n+1)} E_1^{(n)} - g_{1,2}^{(n)} E_1^{(n+1)}}{g_{1,2}^{(n+1)} - g_{1,2}^{(n)}}.$$

Or $E_1^{(n)}$ (respectivement $E_1^{(n+1)}$) est le résultat obtenu par l'application de $E(1)$ (Chapitre II, paragraphe II.1.3., 1)) à x_n x_{n+1} (resp. x_{n+1}, x_{n+2}). Dans le chapitre II (paragraphe II.4.3., a)) nous avons développé l'erreur $E_1^{(n)}-x_*$ comme suit :

$$E_1^{(n)}-x_* = [a + b(e_n + e_{n+1})]\, e_n\, e_{n+1}$$

On a aussi

$$E_1^{(n+1)}-x_* = [a + b(e_{n+1} + e_{n+2})]\, e_{n+1}\, e_{n+2}$$

Vu les relations précédentes liant les e_i

$$E_1^{(n)}-x_* = C\, e_n^3\, [a + b\, e_n + O(e_n^2)]$$

$$\text{et } E_1^{(n+1)}-x_* = C^4\, e_n^6\, [a + O(e_n^2)].$$

Quant à $g_{1,2}^{(n)}$, elle est donnée par

$$g_{1,2}^{(n)} = \frac{\begin{vmatrix} f_{n+1} & f_{n+2} \\ f_n & f_{n+1} \end{vmatrix}}{\begin{vmatrix} 1 & 1 \\ f_n & f_{n+1} \end{vmatrix}} = \frac{\begin{vmatrix} C_1 e_{n+1} & C_1 e_{n+2} \\ C_1 e_n & C_1 e_{n+1} \end{vmatrix}}{\begin{vmatrix} 1 & 1 \\ C_1\, e_n & C_1 e_{n+1} \end{vmatrix}} = C_1\; C \frac{\begin{vmatrix} e_n^2 & e_{n+1}^2 \\ e_n & e_{n+1} \end{vmatrix}}{\begin{vmatrix} 1 & 1 \\ e_n & e_{n+1} \end{vmatrix}} = - C_1\, C\, e_n\, e_{n+1}$$

$$\Rightarrow g_{1,2}^{(n)} = - C_1\; C^2\; e_n^3.$$

On a aussi

$$g_{1,2}^{(n)} = - C_1\; C^5\; e_n^6.$$

Par suite

$$g_{1,2}^{(n+1)} - g_{1,2}^{(n)} \sim C_1\; C^2\; e_n^3.$$

L'erreur e_{n+4} s'écrit

$$e_{n+4} = \frac{g_{1,2}^{(n+1)}(E_1^{(n)} - x_*) - g_{1,2}^{(n)}(E_1^{(n+1)} - x_*)}{g_{1,2}^{(n+1)} - g_{1,2}^{(n)}}$$

$$= \frac{- C_1\; C^5\; e_n^6\; [a + be_n + O(e_n^2)]\; C\; e_n^3 + C_1\; C^2\; e_n^3\; [a + O(e_n^2)]\; C^4\; e_n^6}{C_1\; C^2\; e_n^3}$$

$$= - b\; C^4\; e_n^7.$$

Or $e_{n+4} = C^{15}\; e_n^{16}$. D'où

$$e_n^9 \sim - \frac{b}{C^{11}}, \text{ absurde.}$$

III.3. - PROCEDE D'OVERHOLT [20]

Soit (S_n) une suite de la forme

$$d_{n+1} = \sum_{i\geq 1} a_i \, d_n^i$$

où $d_i = S_i - S$ avec $S = \lim S_n$.

$$\left.\begin{aligned} S_{n+1} &= S + a_1 \, d_n + a_2 \, d_n^2 + \dots \\ S_n &= S + d_n \end{aligned}\right\} \Rightarrow \frac{S_{n+1} - a_1 S_n}{1 - a_1} = S + \frac{a_2}{1-a_1} d_n^2 + \dots$$

Or a_1 est en général inconnu. Soit $\overline{a_1}$ une approximation de a_1 d'ordre α $(\alpha \geq 1)$.

$$\frac{S_{n+1} - \overline{a_1} S_n}{1 - \overline{a_1}} = S + O(d_n^2).$$

En effet

$$\frac{S_{n+1} - \overline{a_1} S_n}{1 - \overline{a_1}} = \frac{S_{n+1} - S_n + S_n(1-\overline{a_1})}{1 - \overline{a_1}} = S_n + \frac{\Delta S_n}{1 - \overline{a_1}}$$

$$= S_n + \frac{\Delta d_n}{1 - a_1 + O(d_n^{\alpha})} = S + d_n + [(a_1 - 1) d_n + O(d_n^2)] \frac{1}{1-a_1} [1 + O(d_n^{\alpha})]$$

$$= S + d_n + [- d_n + O(d_n^2)] \, [1 + O(d_n^{\alpha})]$$

$$= S + d_n + [- d_n + O(d_n^2) + O(d_n^{\alpha+1}) + O(d_n^{\alpha+2})]$$

$$= S + d_n + \{- d_n + O(d_n^2)\} = S + O(d_n^2).$$

Utilisons S_{n+2} pour construire une telle approximation $\overline{a_1}$.

$$\overline{a_1} = \frac{\Delta S_{n+1}}{\Delta S_n} = \frac{\Delta d_{n+1}}{\Delta d_n} = \frac{a_1 d_n + a_2 \left(1 + \frac{a_1^2}{a_1 - 1}\right) d_n^2 + \dots}{d_n + \frac{a_2}{a_1 - 1} d_n^2 + \dots} = a_1 + a_2 \, (1 + a_1) \, d_n + \dots$$

On obtient

$$V_1^{(n)} = \frac{S_{n+1} - \frac{\Delta S_{n+1}}{\Delta S_n} S_n}{1 - \frac{\Delta S_{n+1}}{\Delta S_n}} = S + \frac{a_1\ a_2}{a_1 - 1} d_n^2 + \ldots$$

Maintenant considérons la suite $(V_1^{(n)})_{n\in\mathbb{N}}$.

$$V_1^{(n)} = S + a_{1,2}\ d_n^2 + a_{1,3}\ d_n^3 + \ldots$$

$$V_1^{(n+1)} = S + a_{1,2}\ d_{n+1}^2 + a_{1,3}\ d_{n+1}^3 + \ldots$$

$$= S + a_1^2\ a_{1,2}\ d_n^2 + a'_{1,3}\ d_n^3 + \ldots$$

cherchons à nouveau une approximation de a_1. On peut se contenter de $\frac{\Delta S_{n+1}}{\Delta S_n}$.

Cependant il semble naturel d'utiliser des nouveaux termes de la suite, S_{n+3} en particulier. D'où la nouvelle approximation

$$\overline{a_1} = \frac{\Delta S_{n+2}}{\Delta S_{n+1}} = a_1 + a_2\ (1 + a_1)\ d_{n+1} + \ldots$$

$$= a_1 + a_1\ a_2\ (1 + a_1)\ d_n + \ldots$$

$$\Rightarrow \overline{a}_1^2 = a_1^2 + 2a_1^2\ a_2\ (1 + a_1)\ d_n + \ldots$$

$\overline{a}_1^2$ est une approximation de a_1^2 d'ordre suffisant pour que :

$$V_2^{(n)} = \frac{V_1^{(n+1)} - \overline{a}_1^2 V_1^{(n)}}{1 - \overline{a}_1^2} = S + O(d_n^3).$$

$$V_2^{(n)} = V_1^{(n)} + \frac{V_1^{(n+1)} - V_1^{(n)}}{1 - \overline{a}_1^2} = S + a_{1,2}\ d_n^2 + O(d_n^3) + [(a_1^2 - 1)a_{1,2}\ d_n^2 + O(d_n^3)]$$

$$\frac{1}{1-a_1^2}\ [1 + O(d_n)]$$

$$= S + a_{1,2}\ d_n^2 + O(d_n^3) + [-\ a_{1,2}\ d_n^2 + O(d_n^3)][1 + O(d_n)]$$

$$= S + a_{1,2}\ d_n^2 + O(d_n^3) + [-\ a_{1,2}\ d_n^2 + O(d_n^3)] = S + O(d_n^3).$$

A partir de la suite $(V_2^{(n)})_{n\in\mathbb{N}}$ on construit de la même manière la suite $(V_3^{(n)})_{n\in\mathbb{N}}$.

Par récurrence on va construire $(V_k^{(n)})_{n\in\mathbb{N}}$. Supposons que l'on ait $(V_{k-1}^{(n)})$ telle que

$$V_{k-1}^{(n)} = S + a_{k-1,k}\, d_n^k + a_{k-1,k+1}\, d_n^{k+1} + \dots$$

$$\Rightarrow V_{k-1}^{(n+1)} = S + a_{k-1,k}\, d_{n+1}^k + a_{k-1,k+1}\, d_{n+1}^{k+1} + \dots$$

$$d_{n+1} = a_1\, d_n + a_2\, d_n^2 + \dots$$

$$d_{n+1}^k = a_1^k\, d_n^k + k\, a_1^{k-1}\, a_2\, d_n^{k+1} + \dots$$

$$d_{n+1}^{k+1} = a_1^{k+1}\, d_n^{k+1} + \dots$$

$$V_{k-1}^{(n+1)} = S + a_1^k\, d_n^k\, a_{k-1,k} + (k\, a_1^{k-1}\, a_2\, a_{k-1,k} + a_1^k\, a_{k-1,k+1})\, d_n^{k+1} + \dots$$

$$\Rightarrow \frac{V_{k-1}^{(n+1)} - a_1^k V_{k-1}^{(n)}}{1 - a_1^k} = S + \frac{a_1^{k-1}}{1 - a_1^k}\,[a_{k-1,k}\, a_2\, k + (a_1 - 1)\, a_{k-1,k+1}]\, d_n^{k+1} + \dots$$

Considérons à nouveau une approximation de a_1. Soit

$$\overline{a_1} = \frac{\Delta S_{n+k}}{\Delta S_{n+k-1}} = a_1 + a_2\,(1 + a_1)\, d_{n+k-1} + \dots$$

$$= a_1 + a_1^{k-1}\, a_2\,(1 + a_1)\, d_n + \dots$$

$$\Rightarrow \overline{a}_1^k = a_1^k + k\, a_1^{2k-2}\, a_2 (1+a_1)\, d_n + \dots$$

On obtient

$$V_k^{(n)} = \frac{V_{k-1}^{(n+1)} - \overline{a}_1^k V_{k-1}^{(n)}}{1 - \overline{a}_1^k} = S + O(d_n^{k+1}).$$

$$V_k^{(n)} = V_{k-1}^{(n)} + \frac{\Delta V_{k-1}^{(n)}}{1 - a_1^{-k}}$$

$$= S + a_{k-1,k}\, d_n^k + O(d_n^{k+1}) + [(a_1^k - 1)a_{k-1,k}\, d_n^k + O(d_n^{k+1})]\, \frac{1}{1-a_1^k}\, [1 + O(d_n)]$$

$$= S + a_{k-1,k}\, d_n^k + O(d_n^{k+1}) + \{- a_{k-1,k}\, d_n^k + O(d_n^{k+1})\}\, \{1 + O(d_n)\}$$

$$= S + O(d_n^{k+1}).$$

D'où l'algorithme d'Overholt suivant :

$$\begin{cases} V_0^{(n)} = S_n, \; n = 0, 1, 2, \ldots \\[2ex] V_k^{(n)} = \dfrac{(\Delta S_{n+k})^k V_{k-1}^{(n)} - (\Delta S_{n+k-1})^k V_{k-1}^{(n+1)}}{(\Delta S_{n+k})^k - (\Delta S_{n+k-1})^k}, \; \begin{array}{l} k = 1, 2, 3, \ldots \\ n = 0, 1, 2, \ldots \end{array} \end{cases}$$

<u>Remarque</u> :

Dans [7] et récemment dans [26] on trouve des méthodes itératives d'un ordre arbitraire construites à l'aide de cet algorithme. De telles méthodes sont comparables aux méthodes S(k) (Chapitre II). Nous en rappelons ci-dessous la construction : soit à résoudre une équation non linéaire $x = \phi(x)$

x_0 donné

la $n^{\text{ième}}$ itération est

$$\left[\begin{array}{l} u_0 = x_{n-1} \\ u_{i+1} = \phi(u_i), \; i = 0, 1, \ldots, k \\ V_0^{(i)} = u_i \qquad i = 0, 1, \ldots, k \\[2ex] \text{On applique l'algorithme d'Overholt à la suite finie} \\ \{u_0, u_1, \ldots, u_{k+1}\}. \\[2ex] \text{On prend} \\ x_n = V_k^{(0)} \end{array}\right.$$

On a :

$$x_n - x_* = O(x_{n-1} - x_*)^{k+1}.$$

Dans ce qui suit nous donnons une interprétation du procédé (B') (Chapitre I) de Germain-Bonne. Elle est basée essentiellement sur la construction inductive faite par Overholt pour obtenir son algorithme et que nous venons d'exposer ci-dessus.

III.4. - INTERPRETATION DU PROCEDE D'EXTRAPOLATION INVERSE DE GERMAIN-BONNE (B') (CHAPITRE I)

Considérons à nouveau la suite

$$S_{n+1} = S + \sum_{i=1}^{\infty} a_i (S_n - S)^i, \ \forall n.$$

$$S_n = S + d_n$$

$$S_{n+1} = S + d_{n+1} = S + \sum_{i\geq 1} a_i \, d_n^i.$$

$$\Longrightarrow \frac{S_{n+1} - a_1 S_n}{1 - a_1} = S + a_2 \, d_n^2 + a_3 \, d_n^3 + \ldots$$

La quantité $\dfrac{S_{n+1} - a_1 S_n}{1 - a_1}$ est une approximation de S, d'ordre deux.

Or, généralement a_1 est inconnu. Pour atteindre l'ordre deux, une approximation $\overline{a_1}$ de a_1 d'ordre un est suffisante (III.3).

$$\overline{a_1} = \frac{\Delta S_{n+1}}{\Delta S_n} = a_1 + a_2 \, (1 + a_1) \, d_n + \ldots$$

On obtient

$$T_1^{(n)} = \frac{S_{n+1} - \dfrac{\Delta S_{n+1}}{\Delta S_n} S_n}{1 - \dfrac{\Delta S_{n+1}}{\Delta S_n}} = S + a_{1,2} \, d_n^2 + a_{1,3} \, d_n^3 + \ldots$$

où $a_{1,2} = \dfrac{a_1 \, a_2}{a_1 - 1}.$

Remarque :

$$\varepsilon_2^{(n)} = V_1^{(n)} = T_1^{(n)}, \ \forall n \ ;$$

ce qui montre que la transformation de Shanks, le procédé d'Overholt et le procédé d'interpolation inverse de Germain-Bonne sont trois extensions du procédé Δ^2 d'Aitken.

Considérons maintenant la suite $(T_1^{(n)})_{n\in\mathbb{N}}$.

$$T_1^{(n)} = S + a_{1,2}\, d_n^2 + a_{1,3}\, d_n^3 + \ldots$$

$$T_1^{(n+1)} = S + a_{1,2}\, d_{n+1}^2 + a_{1,3}\, d_{n+1}^3 + \ldots$$

$$= S + a_1^2\, a_{1,2}\, d_n^2 + (a_1^3\, a_{1,3} + 2\, a_1\, a_2\, a_{1,2})\, d_n^3 + \ldots$$

$$\Longrightarrow \frac{T_1^{(n+1)} - a_1^2\, T_1^{(n)}}{1 - a_1^2} = S + \frac{2\, a_1\, a_2\, a_{1,2} + a_1^2(a_1 - 1)\, a_{1,3}}{1 - a_1^2}\, d_n^3 + \ldots$$

Mais a_1 est inconnu. Donc a_1^2 l'est aussi. Lors de la construction du procédé d'Overholt, on s'est intéressé à la recherche d'une nouvelle approximation de a_1 :

$$\overline{a_1} = \frac{\Delta S_{n+2}}{\Delta S_{n+1}} = a_1 + O(d_n).$$

On va maintenant chercher directement une approximation de a_1^2. Pour cela on garde l'ancienne approximation $\overline{a_1}$ de a_1 dans le procédé d'Overholt et, avec sa nouvelle approximation on fabrique :

$$\overline{a_1^2} = \frac{\Delta S_{n+1}}{\Delta S_n}\, \frac{\Delta S_{n+2}}{\Delta S_{n+1}} = \frac{\Delta S_{n+2}}{\Delta S_n}.$$

$$\Longrightarrow \overline{a_1^2} = \frac{\Delta S_{n+2}}{\Delta S_n} = \{a_1 + a_2(1 + a_1)\, d_n + \ldots\}\, \{a_1 + a_1\, a_2\, (1 + a_1)\, d_n + \ldots\}$$

$$= a_1^2 + (1 + a_1)^2\, a_1\, a_2\, d_n + \ldots$$

On obtient

$$T_2^{(n)} = \frac{T_1^{(n+1)} - \bar{a}_1^2 T_1^{(n)}}{1 - \bar{a}_1^2} = T_1^{(n)} + \frac{\Delta T_1^{(n)}}{1 - \bar{a}_1^2}$$

$$= S + a_{1,2}\, d_n^2 + a_{1,3}\, d_n^3 + \ldots + [a_{1,2}(a_1^2-1)\, d_n^2 + (2a_1\, a_2\, a_{1,2} + a_{1,3}(a_1^3-1)\, d_n^3 + \ldots$$

$$\frac{1}{1 - a_1^2} [1 + \frac{(1+a_1)^2}{1 - a_1^2}\, a_1\, a_2\, d_n + \ldots]$$

$$= S + \frac{a_1}{1 + a_1} [a_2\, a_{1,2}\, (1 + a_1) - a_1\, a_{1,3}]\, d_n^3 + \ldots$$

La suite obtenue $(T_2^{(n)})$ est la deuxième colonne du procédé (B') (Chapitre I).

De proche en proche on construit de cette façon toutes les autres colonnes.

Considérons maintenant la suite $(T_{k-1}^{(n)})_{n\in\mathbb{N}}$.

$$T_{k-1}^{(n)} = S + a_{k-1,k}\, d_n^k + a_{k-1,k+1}\, d_n^{k+1} + \ldots$$

$$T_{k-1}^{(n+1)} = S + a_{k-1,k} d_{n+1}^k + a_{k-1,k+1}\, d_{n+1}^{k+1} + \ldots$$

$$= S + a_1^k\, a_{k-1,k}\, d_n^k + (k\, a_1^{k-1}\, a_2\, a_{k-1,k} + a_1^{k+1}\, a_{k-1,k+1})\, d_n^{k+1} + \ldots$$

$$\Rightarrow \frac{T_{k-1}^{(n+1)} - a_1^k T_{k-1}^{(n)}}{1 - a_1^k} = S + \frac{a_1^{k-1}}{1-a_1^k}\, [(ka_2)\, a_{k-1,k} + a_1\, a_{k-1,k+1}(a_1-1)]\, d_n^{k+1} + \ldots$$

a_1 est toujours inconnu. a_1^k l'est aussi. Or au cours de la construction des $V_j^{(n)}$, $j = 1, 2, \ldots, k-1$ on avait fait les approximations successives

$\frac{\Delta S_{n+1}}{\Delta S_n}, \frac{\Delta S_{n+2}}{\Delta S_{n+1}}, \ldots, \frac{\Delta S_{n+k-1}}{\Delta S_{n+k-2}}$ de a_1. Soit $\frac{\Delta S_{n+k}}{\Delta S_{n+k-1}}$ sa nouvelle approximation.

Considérons

$$\overline{a}_1^k = \frac{\Delta S_{n+1}}{\Delta S_n} \times \frac{\Delta S_{n+2}}{\Delta S_{n+1}} \times \ldots \times \frac{\Delta S_{n+k}}{\Delta S_{n+k-1}} = \frac{\Delta S_{n+k}}{\Delta S_n}$$

$$\Rightarrow \quad \overline{a}_1^k = a_1^k + a_1^{k-1}\, a_2 (1 + a_1)(1 + a_1 + \ldots + a_1^{k-1})\, d_n + \ldots$$

la quantité

$$T_k^{(n)} = \frac{T_{k-1}^{(n+1)} - \overline{a}_1^k\, T_{k-1}^{(n)}}{1 - \overline{a}_1^k}$$

est une approximation de S d'ordre k+1.

$$T_k^{(n)} = \frac{T_{k-1}^{(n+1)} - \overline{a}_1^k\, T_{k-1}^{(n)}}{1 - \overline{a}_1^k} = T_{k-1}^{(n)} + \frac{\Delta T_{k-1}^{(n)}}{1 - \overline{a}_1^k}$$

$$= S + a_{k-1,k}\, d_n^k + a_{k-1,k+1}\, d_n^{k+1} + \ldots$$

$$+ \{(a_1^k - 1)\, a_{k-1,k}\, d_n^k + [k\, a_1^{k-1}\, a_2\, a_{k-1,k} + a_{k-1,k+1}(a_1^{k+1} - 1)]\, d_n^3 + \ldots\}$$

$$\frac{1}{1 - a_1^k} \{1 + \frac{a_1^{k-1}}{1 - a_1^k}\, a_2 (1 + a_1)(1 + a_1 + \ldots + a_1^{k-1})\, d_n + \ldots\}$$

$$= S + \frac{a_1^{k-1}}{1 - a_1^k} [a_1\, a_{k-1,k+1}(1 - a_1) + a_2\, a_{k-1,k}\, \{a_1^k + 2\,(a_1^2 + \ldots + a_1^{k-1})$$

$$+ a_1 (2-k)\}]\, d_n^{k+1} + \ldots$$

Pour une telle suite ce procédé donne des approximations d'ordre de plus en plus élevé. Il est donc comparable au procédé d'Overholt (III.3). Ces deux algorithmes sont bien adaptés aux suites

$$S_{n+1} = F(S_n)$$

$$= S + \frac{F'(S)}{1!} (S_n - S) + \frac{F^{('')}(S)}{2!} (S_n - S)^2 + \ldots$$

Dans ce qui suit nous donnons un formalisme regroupant ces deux procédés.

III.5. - UNE GENERALISATION DU PROCEDE D'OVERHOLT

Il sera question ici de donner une généralisation du procédé d'Overholt, et du procédé ((B'), Chapitre I) d'interpolation inverse de Germain-Bonne-Wimp et aussi du procédé d'extrapolation de Richerdson. Les propriétés de convergence et d'accélération de la convergence du procédé obtenu seront examinées. Puis on terminera par l'application de la procédure θ à ce procédé.

III.5.1. - Généralisation

Soit (S_n) une suite de la forme

$$S_n = S + \sum_{i\geq 1} a_i\, g_i(n), \ \forall n.$$

Les fonctions g_i sont données. On suppose qu'elles satisfont aux propriétés suivantes :

$$(Q_1) : g_i(n+1) = \sum_{j\geq i} b_{ij}\, g_j(n), \ \forall i \geq 1, \ \forall n$$

$$(Q_2) : g_i(n)\, g_j(n) = \sum_{\ell\geq i+j} c_{i\ell}^{(j)}\, g_\ell(n), \ \forall i \geq 1, \ \forall j \geq 1, \ \forall n.$$

$$(Q_3) : \frac{g_j(n)}{g_i(n)} = \sum_{\ell\geq j-i} d_{i\ell}^{(j)}\, g_\ell(n), \ \forall j > i \geq 1, \ \forall n.$$

La construction de ce procédé est basé sur l'élimination successive des $g_i(n)$ suivant des indices i croissants.

a) *Elimination de* $g_1(n)$

$$S_n = S + a_1\, g_1(n) + a_2\, g_2(n) + \ldots$$

$$S_{n+1} = S + a_1\, g_1(n+1) + a_2\, g_2(n+1) + \ldots$$

$$g_i(n+1) = b_{ii}\ g_i(n) + b_{ii+1}\ g_{i+1}(n) + \ldots \text{ pour } i = 1, 2, \ldots \quad \Rightarrow$$

$$\ldots \Rightarrow S_{n+1} = S + a_1\ b_{11}\ g_1(n) + (a_1\ b_{12} + a_2\ b_{22})\ g_2(n) + \ldots$$

Formons la combinaison

$$S_{n+1} - b_{11}\ S_n = (1 - b_{11})S + (a_1\ b_{12} + a_2(b_{22}-b_{11}))\ g_2(n) + \ldots$$

Si $b_{11} \neq 1$, on obtient

$$\frac{S_{n+1}-b_{11}S_n}{1 - b_{11}} = S + \frac{a_1b_{12}+a_2(b_{22}-b_{11})}{1 - b_{11}}\ g_2(n) + \ldots$$

En général b_{11} est inconnu. Mais d'après (Q_1)

$$\frac{g_1(n+1)}{g_1(n)} = b_{11} + b_{12}\ \frac{g_2(n)}{g_1(n)} + b_{13}\ \frac{g_3(n)}{g_1(n)} + \ldots$$

or d'après (Q_3)

$$\frac{g_2(n)}{g_1(n)} = d_{11}^{(2)}\ g_1(n) + d_{12}^{(2)}\ g_2(n) + \ldots$$

$$\frac{g_3(n)}{g_1(n)} = d_{12}^{(3)}\ g_2(n) + \ldots$$

.....

Par conséquent

$$\frac{g_1(n+1)}{g_1(n)} = b_{11} + b_{12}\ d_{11}^{(2)}\ g_1(n) + (b_{12}\ d_{12}^{(2)} + b_{13}\ d_{12}^{(3)})\ g_2(n) + \ldots$$

Nous pouvons approcher b_{11} par $\frac{g_1(n+1)}{g_1(n)}$. Cette approximation est justifiée si les g_i tendent vers zéro lorsque n tend vers l'infini.

Dans

$$\frac{S_{n+1}-b_{11}S_n}{1-b_{11}},$$

remplaçons b_{11} par $\frac{g_1(n+1)}{g_1(n)}$, cela donne

$$W_1^{(n)} = \frac{S_{n+1} - \frac{g_1(n+1)}{g_1(n)} S_n}{1 - \frac{g_1(n+1)}{g_1(n)}}.$$

La quantité $W_1^{(n)}$ peut s'écrire :

$$W_1^{(n)} = S + a_{12}\, g_2(n) + a_{1,3}\, g_3(n) + \dots \quad \forall n.$$

En effet :

$$S_{n+1} - \frac{g_1(n+1)}{g_1(n)} S_n = S + a_1\, b_{11}\, g_1(n) + (a_1\, b_{12} + a_2\, b_{22})\, g_2(n) + \dots$$

$$- (b_{11} + b_{12}\, d_{11}^{(2)}\, g_1(n) + (b_{12}\, d_{12}^{(2)} + b_{13}\, d_{12}^{(3)}) g_2(n) + \dots)(S + a_1 g_1(n) + a_2 g_2(n) + \dots)$$

$$= \{1 - (b_{11} + b_{12}\, d_{11}^{(2)}\, g_1(n) + \dots)\} S + [a_1\, b_{12} + a_2(b_{22} - b_{11})]\, g_2(n)$$

$$- a_1\, b_{12}\, d_{11}^{(2)}\, [g_1(n)]^2 + \dots$$

D'après (Q_2)

$$[g_1(n)]^2 = C_{11}^{(1)}\, g_2(n) + \dots$$

Donc

$$S_{n+1} - \frac{g_1(n+1)}{g_1(n)} S_n = \{1 - \{b_{11} + b_{12} d_{11}^{(2)} g_1(n) + \dots\}\} S + \{a_1 b_{12}(1 - d_{11}^{(2)} C_{11}^{(1)}) + a_2(b_{22} - b_{11})\} g_2(n)$$

$$+ \dots$$

$$\frac{S_{n+1} - \frac{g_1(n+1)}{g_1(n)} S_n}{1 - \frac{g_1(n+1)}{g_1(n)}} =$$

$$\frac{\{1-(b_{11}+b_{12}d_{11}^{(2)}g_1(n)+..)\}S + \{a_1b_{12}(1-d_{11}^{(2)}c_{11}^{(1)}) + a_2(b_{22}-b_{11})\}\ g_2(n) + \ldots}{1-(b_{11}+b_{12}\ d_{11}^{(2)}\ g_1(n) + \ldots)}$$

$$= S + \frac{1}{1-b_{11}} \frac{\{a_1b_{12}(1-d_{11}^{(2)}c_{11}^{(1)}) + a_2(b_{22}-b_{11})\}\ g_2(n) + \ldots}{1 - \frac{b_{12}d_{11}^{(2)}}{1-b_{11}}\ g_1(n) + \ldots}$$

$$= S + \frac{a_1b_{12}(1-d_{11}^{(2)}c_{11}^{(1)}) + a_2(b_{22}-b_{11})}{1-b_{11}}\ g_2(n) + \ldots$$

Par conséquent nous pouvons écrire

$$W_1^{(n)} = S + \sum_{i\geq 2} a_{1i}\ g_i(n),\ \forall n.$$

La suite $(W_1^{(n)})_{n\in\mathbb{N}}$ ainsi obtenue a donc une forme identique à la suite initiale (S_n).

A nouveau nous éliminons $g_2(n)$ suivant le schéma précédent.

b) Elimination de $g_2(n)$

$$W_1^{(n)} = S + a_{12}\ g_2(n) + a_{13}\ g_3(n) + \ldots$$

$$W_1^{(n+1)} = S + a_{12}\ g_2(n+1) + a_{13}\ g_3(n+1) + \ldots$$

D'après (Q_1)

$$g_2(n+1) = b_{22}\ g_2(n) + b_{23}\ g_3(n) + \ldots$$

$$g_3(n+1) = b_{33}\ g_3(n) + \ldots$$

....

Ainsi $W_1^{(n+1)} = S + a_{12}\, b_{22}\, g_2(n) + (a_{12}\, b_{23} + a_{13}\, b_{33})\, g_3(n) + \ldots$

Formons la combinaison

$$W_1^{(n+1)} - b_{22}\, W_1^{(n)} = (1-b_{22})S + \{a_{12}\, b_{23} + a_{13}(b_{33}-b_{22})\}\, g_3(n) + \ldots$$

Si $b_{22} \neq 1$, on obtient

$$\frac{W_1^{(n+1)} - b_{22}\, W_1^{(n)}}{1 - b_{22}} = S + \frac{a_{12}\, b_{23} + a_{13}(b_{33}-b_{22})}{1 - b_{22}}\, g_3(n) + \ldots$$

En général b_{22} est inconnu. Mais d'après (Q_1)

$$\frac{g_2(n+1)}{g_2(n)} = b_{22} + b_{23}\,\frac{g_3(n)}{g_2(n)} + b_{24}\,\frac{g_4(n)}{g_2(n)} + \ldots$$

Or d'après (Q_3)

$$\frac{g_3(n)}{g_2(n)} = d_{21}^{(3)}\, g_1(n) + d_{22}^{(3)}\, g_2(n) + \ldots$$

$$\frac{g_4(n)}{g_2(n)} = d_{22}^{(4)}\, g_2(n) + \ldots$$

....

D'où

$$\frac{g_2(n+1)}{g_2(n)} = b_{22} + b_{23}\, d_{21}^{(3)} g_1(n) + (b_{23}\, d_{22}^{(3)} + b_{24}\, d_{22}^{(4)})\, g_2(n) + \ldots$$

A la place de b_{22} dans

$$\frac{W_1^{(n+1)} - b_{22}\, W_1^{(n)}}{1 - b_{22}}$$

utilisons $\dfrac{g_2(n+1)}{g_2(n)}$. Cela donne

$$W_2^{(n)} = \frac{W_1^{(n+1)} - \dfrac{g_2(n+1)}{g_2(n)}\, W_1^{(n)}}{1 - \dfrac{g_2(n+1)}{g_2(n)}}.$$

La quantité $W_2^{(n)}$ est telle que

$$W_2^{(n)} = S + a_{23}\, g_3(n) + a_{24}\, g_4(n) + \dots \;\forall n.$$

En effet :

$$W_1^{(n+1)} - \frac{g_2(n+1)}{g_2(n)} W_1^{(n)} = S + a_{12}\, b_{22}\, g_2(n) + (a_{12}\, b_{23} + a_{13}\, b_{33})\, g_3(n) + \dots$$

$$- (b_{22} + b_{23}\, d_{21}^{(3)}\, g_1(n) + (b_{23}\, d_{22}^{(3)} + b_{24}\, d_{22}^{(4)})\, g_2(n) + \dots)$$

$$(S + a_{12}\, g_2(n) + a_{13}\, g_3(n) + \dots)$$

$$= \{1-(b_{22}+b_{23}d_{21}^{(3)}g_1(n)+\dots)\}S + \{a_{12}b_{23}+a_{13}(b_{33}-b_{22})\}g_3(n) - a_{12}b_{23}d_{21}^{(3)}g_1(n)g_2(n) + .$$

D'après (Q_2)

$$g_1(n)\, g_2(n) = c_{13}^{(2)}\, g_3(n) + \dots$$

Donc

$$W_1^{(n+1)} - \frac{g_2(n+1)}{g_2(n)} W_1^{(n)} = \{1 - (b_{22} + b_{23}\, d_{21}^{(3)}\, g_1(n) + \dots\}S$$

$$+ \{a_{12}\, b_{23}\, (1 - d_{21}^{(3)}\, c_{13}^{(2)}) + a_{13}(b_{33}-b_{22})\}\, g_3(n) + \dots$$

et

$$\frac{W_1^{(n+1)} - \dfrac{g_2(n+1)}{g_2(n)} W_1^{(n)}}{1 - \dfrac{g_2(n+1)}{g_2(n)}} = S + \frac{1}{1-b_{22}} \frac{\{a_{12}b_{23}(1-d_{21}^{(3)}c_{13}^{(2)}) + a_{13}(b_{33}-b_{22})\}\, g_3(n) + \dots}{1 - \dfrac{b_{23}\, d_{21}^{(3)}}{1-b_{22}}\, g_1(n) + \dots}$$

$$= S + \frac{a_{12}\, b_{23}(1 - d_{21}^{(3)}\, c_{13}^{(2)}) + a_{13}(b_{33}-b_{22})}{1 - b_{22}}\, g_3(n) + \dots$$

Nous pouvons conclure que

$$W_2^{(n)} = S + a_{23}\, g_3(n) + a_{24}\, g_4(n) + \ldots \quad \forall n.$$

Ainsi de proche en proche nous construisons les suites $(W_k^{(n)})_{n\in\mathbb{N}}$ pour différentes valeurs de k.

c) *Elimination de* $g_k(n)$

Supposons que l'on ait

$$W_{k-1}^{(n)} = S + a_{k-1,k}\, g_k(n) + a_{k-1,k+1}\, g_{k+1}(n) + \ldots \quad \forall n.$$

Par une construction analogue à celle faite en a) ou b), on construit la quantité $W_k^{(n)}$ donnée par

$$W_k^{(n)} \; \frac{W_{k-1}^{(n+1)} - \dfrac{g_k(n+1)}{g_k(n)} W_{k-1}^{(n)}}{1 - \dfrac{g_k(n+1)}{g_k(n)}}, \quad \forall n.$$

$W_k^{(n)}$ vérifie

$$W_k^{(n)} = S + a_{k,k+1}\, g_{k+1}(n) + a_{k,k+2}\, g_{k+2}(n) + \ldots \quad \forall n.$$

Pour cela il suffit de faire jouer dans b) à $W_{k-1}^{(n)}$ le rôle de $W_1^{(n)}$ et à $g_k(n)$ celui de $g_2(n)$.

En résumé, pour une suite (S_n) quelconque et pour les fonctions g_i, on a donc le procédé (W) suivant :

$$(W) : \begin{cases} W_0^{(n)} = S_n, \ n = 0, 1, 2, \ldots \\[2ex] W_k^{(n)} = \dfrac{g_k(n+1) W_{k-1}^{(n)} - g_k(n) W_{k-1}^{(n+1)}}{g_k(n+1) - g_k(n)}, \ k = 1, 2, \ldots, \ ; \ n = 0, 1, 2, \ldots \end{cases}$$

Propriété 2

Si $S_n = S + \sum_{i\geq 1} a_i \, g_i(n)$, $\forall n$,

si les $g_i(n)$ *satisfont* (Q_1), (Q_2) et (Q_3),
et si $b_{ii} \neq 1$, $\forall i$, *où* b_{ii} *est le premier coefficient donné par* (Q_1),
alors :

$$W_k^{(n)} = S + a_{k,k+1} \, g_{k+1}(n) + \ldots, \ \forall n, \ \forall k \geq 1.$$

Exemples :

a) $S_n = S + a_1 \lambda^n + a_2 \lambda^{2n} + a_3 \lambda^{3n} + \ldots$ avec $|\lambda| < 1$.

Par exemple :

$S_n = e^{x^n}$,

$S_n = f(x^n)$, où f est une fonction analytique au voisinage de zéro.

On peut choisir $g_i(n) = \lambda^{in}$.

(Q_1) : $g_i(n+1) = \lambda^{i(n+1)} = \lambda^i \lambda^{in} = \lambda^i \, g_i(n)$
$\quad b_{ii} = \lambda^i \neq 1$.

(Q_2) : $g_i(n) \, g_j(n) = \lambda^{in} \lambda^{jn} = \lambda^{(i+j)n} = g_{i+j}(n)$.

(Q_3) : $\dfrac{g_j(n)}{g_i(n)} = \dfrac{\lambda^{jn}}{\lambda^{in}} = \lambda^{(j-i)n} = g_{j-i}(n)$; $j > i \geq 1$.

En appliquant le procédé (W) nous obtenons

$$W_k^{(n)} = S + a_{k,k+1} \, \lambda^{(k+1)n} + \ldots, \ \forall n, \ \forall k \geq 1.$$

b) Soit $(S_n)_{n\in\mathbb{N}}$ donnée par :

$S_{n+1} = F(S_n) = f(S_n) + S_n$

$\iff \Delta S_n = f(S_n)$

$S_n \to S,\ n \to \infty$; $F(S) = S$ et $f(S) = 0$

Supposons f inversible ;

$$S_n = f^{-1}(\Delta S_n) = S + a_1\, \Delta S_n + a_2(\Delta S_n)^2 + \ldots + a_i\, (\Delta S_n)^i + \ldots$$

où $a_i = \dfrac{(f^{-1})^{(i)}(0)}{i!}$

On peut choisir dans ce cas $g_i(n) = (\Delta S_n)^i$. (W) devient :

$$\begin{cases} W_0^{(n)} = S_n \\[2ex] W_k^{(n)} = \dfrac{(\Delta S_{n+1})^k W_{k-1}^{(n)} - (\Delta S_n)^k W_{k-1}^{(n+1)}}{(\Delta S_{n+1})^k - (\Delta S_n)^k}, \ \forall n,\ \forall k \geq 1. \end{cases}$$

$$(Q_2)\ :\ g_i(n)\, g_j(n) = (\Delta S_n)^i\, (\Delta S_n)^j = (\Delta S_n)^{i+j} = g_{i+j}(n)$$

$$(Q_3)\ :\ \frac{g_j(n)}{g_i(n)} = \frac{(\Delta S_n)^j}{(\Delta S_n)^i} = (\Delta S_n)^{j-i} = g_{j-i}(n),\ \forall j > i \geq 1.$$

Mais il est difficile d'obtenir (Q_1). En effet :

$$(Q_1)\ :\ g_i(n+1) = (\Delta S_{n+1})^i$$

Or $\Delta S_{n+1} = \Delta F(S_n) = \dfrac{F'(S_n)}{1!}\, \Delta S_n + \dfrac{F''(S_n)}{2!}\, (\Delta S_n)^2 + \ldots$

et

$$\left.\begin{aligned} F^{(j)}(S_n) &= \sum_{i\geq 0} \frac{F^{(i+j)}}{i!}\, (S_n - S)^i \\[2ex] S_n &= S + \sum_{i\geq 1} a_i (\Delta S_n)^i \end{aligned}\right\} \Longrightarrow \quad F^{(j)}(S_n) = F^{(j)}(S) + \sum_{i\geq 1} b_i (\Delta S_n)^i$$

Donc on peut écrire :

$$\Delta S_{n+1} = \sum_{i\geq 1} C_i(\Delta S_n)^i \text{ avec } C_i \text{ indépendant de } n.$$

Par conséquent

$$g_i(n+1) = \sum_{j\geq i} b_{ij}(\Delta S_n)^j$$

$$= \sum_{j\geq i} b_{ij}\ g_j(n) \text{ ou } b_{ij} \text{ indépendant de } n.$$

En appliquant le procédé (W), nous obtenons :

$$W_k^{(n)} = S + a_{k,k+1}\ (\Delta S_n)^{k+1} + \ldots\ \forall n,\ \forall k \geq 1.$$

Remarque :

Le procédé (W) correspondant au choix $g_i(n) = (\Delta S_n)^i$ est comparable au procédé d'Overholt (III.3) et au procédé (B') (Chapitre I) (III.4). En effet $g_k(n+1)/g_k(n) = (\Delta S_{n+1}/\Delta S_n)^k$ est une approximation de a_1^k où a_1 est le coefficient donnée par :

$$S_{n+1} = S + a_1\ (S_n-S) + a_2\ (S_n-S)^2 + \ldots \text{ (Par exemple dans III.3).}$$

c) Soit $(S_n)_{n\in\mathbb{N}}$ donnée par :

$$S_n = S + a_1\ h_n + a_2\ h_n^2 + a_3\ h_n^3 + \ldots \text{ où } h_n = \frac{h}{2^n}.$$

On prend $g_i(n) = h_n^i$.

les g_i vérifient

(Q_1) : $$g_i(n+1) = h_{n+1}^i = \left(\frac{h}{2^{n+1}}\right)^i = \left(\tfrac{1}{2}\right)^i\ g_i(n)$$

(Q_2) : $$g_i(n)\ g_j(n) = h_n^i\ h_n^j = h_n^{i+j} = g_{i+j}(n)$$

(Q_3) : $$g_j(n)/g_i(n) = h_n^j/h_n^i = h_n^{j-i} = g_{j-i}(n).$$

On obtient

$$W_k^{(n)} = S + a_{k,k+1}\, h_n^{k+1} + \ldots$$

$$= S + a_{k,k+1}\left(\frac{h}{2^n}\right)^{k+1} + \ldots$$

Si $h_n = \lambda^n$ on retrouve le premier exemple traité.

L'algorithme (W) est une généralisation du procédé d'Overholt ; en effet si nous choisissons

$$g_k(n) = (\Delta S_{n+k-1})^k \text{ pour } k \geq 1 \text{ et pour tout } n,$$

alors (W) devient

$$\begin{cases} W_0^{(n)} = S_n,\ n = 0, 1, 2, \ldots \\ W_k^{(n)} = \dfrac{(\Delta S_{n+k})^k\, W_{k-1}^{(n)} - (\Delta S_{n+k-1})^k\, W_{k-1}^{(n+1)}}{(\Delta S_{n+k})^k - (\Delta S_{n+k-1})^k},\ k = 1, 2, \ldots ;\ n = 0, 1, 2, \ldots \end{cases}$$

c'est le procédé d'Overholt (III.3).

Le choix $g_k(n) = \prod_{i=0}^{k-i} \Delta S_{n+i}$ correspond au procédé (B') de Germain-Bonne (III.4).

Le choix $g_k(n) = \prod_{i=0}^{k-1} x_{n+i}$ correspond au procédé de Richardson :

$$\begin{cases} W_0(n) = S_n,\ n = 0, 1, 2, \ldots \\ W_k^{(n)} = \dfrac{x_{n+k}\, W_{k-1}^{(n)} - x_n\, W_{k-1}^{(n+1)}}{x_{n+k} - x_n},\ k = 1, 2, 3, \ldots ;\ n = 0, 1, 2, \ldots \end{cases}$$

Les choix correspondant aux procédés d'Overholt et de Germain-Bonne, vérifient (Q_1), (Q_2) et (Q_3). En effet, dans l'exemple b) on a vu que ΔS_{n+1} possède un développement limité par rapport à ΔS_n. Par suite ΔS_n en possède un par rapport à ΔS_{n+1}. D'une manière générale ΔS_q et ΔS_p possèdent un développement limité l'un par rapport à l'autre.

* <u>Choix $g_i(n) = (\Delta S_{n+i-1})^i$</u>

(Q_1) : $g_i(n+1) = (\Delta S_{n+i})^i = \sum_{j\geq i} b_{ij}(\Delta S_{n+i-1})^j = b_{ii}\, g_i(n) + \sum_{j\geq i+1} b_{ij}(\Delta S_{n+i-1})^j$

Or ΔS_{n+i-1} possède un développement limité par rapport à ΔS_{n+i}.

D'où

$$g_i(n+1) = b_{ii}\, g_i(n) + b_{ii+1}\, \alpha_{i+1}\, g_{i+1}(n) + \sum_{j\geq i+2} \alpha_j (\Delta S_{n+i})^j$$

On développe maintenant ΔS_{n+i} par rapport à ΔS_{n+i+1}

.... etc.

On obtient

$$g_i(n+1) = \sum_{j=0}^{\infty} \beta_{ij}\, g_{i+j}(n).$$

(Q_2) : supposons $j \geq i$

$$g_i(n)\, g_j(n) = (\Delta S_{n+i-1})^i\, (\Delta S_{n+j-1})^j.$$

ΔS_{n+i-1} possède un développement limité par rapport à $\Delta S_{n+i+j-1}$ et ΔS_{n+j-1} en possède un par rapport à $\Delta S_{n+i+j-1}$.

Don on a : $$g_i(n)\, g_j(n) = \sum_{\ell=0}^{\infty} \gamma_{i+j+\ell}\, g_{i+j+\ell}(n).$$

(Q_3) : supposons $j > i \geq 1$

$$\frac{g_j(n)}{g_i(n)} = \frac{(\Delta S_{n+j-1})^j}{(\Delta S_{n+i-1})^i}.$$

On développe d'abord ΔS_{n+j-1} par rapport à ΔS_{n+i-1}. Le résultat obtenu est

$$\frac{g_j(n)}{g_i(n)} = \sum_{\ell \geq j-i} \delta_\ell (\Delta S_{n+i-1})^\ell$$

Or ΔS_{n+i-1} possède lui aussi un développement par rapport à $\Delta S_{n+j-i-1}$.

D'où la propriété (Q_3).

** Choix $g_i(n) = \prod_{\ell=0}^{i-1} \Delta S_{n+\ell}$

Par la même technique que le choix précédent on montre que les $g_i(n)$ vérifient (Q_1), (Q_2) et (Q_3).

A titre d'indication vérifions (Q_1) :

$$g_i(n+1) = \prod_{\ell=0}^{i-1} \Delta S_{n+1+\ell} = \prod_{\ell=1}^{i} \Delta S_{n+\ell} = \left(\prod_{\ell=1}^{i-1} \Delta S_{n+\ell}\right) \Delta S_{n+i}$$

$$= g_i(n) \frac{\Delta S_{n+i}}{\Delta S_n}$$

$$\Delta S_{n+i} = \sum_{j=1}^{\infty} \theta_j (\Delta S_n)^j$$

D'où

$$g_i(n+1) = \theta_1 \, g_i(n) + \theta_2 \left(\prod_{\ell=1}^{i-1} \Delta S_{n+\ell}\right)(\Delta S_n)^2 + \ldots$$

Puis un développement de ΔS_n par rapport à ΔS_{n+i} donne :

$$g_i(n+1) = \theta_1 \, g_i(n) + \theta_2 \, \theta_2' \, g_{i+1}(n) + \ldots$$

De proche en proche nous obtenons la propriété (Q_1).

D'après la propriété 2 précédente.

La quantité $W_k^{(n)}$ donnée par le procédé d'Overholt est telle que :

$$W_k^{(n)} = S + a_{k,k+1} \, (\Delta S_{n+k})^{k+1} + \ldots$$

et celle donnée par Germain-Bonne-Wimp est telle que :

$$W_k^{(n)} = S + a_{k,k+1} \prod_{i=0}^{k} \Delta S_{n+i-1}$$

On retrouve donc les propriétés fondamentales bien connues de ces deux procédés.

III.5.2. - Etude du procédé (W)

III.5.2.1. - Résultats de convergence

Ce procédé est quasi-linéaire :

$$W_k^{(n)}(a\ S_n+b) = a\ W_k^{(n)}(S_n) + b\ ;\ a,\ b \in \mathbb{R}$$

Par récurrence sur k on a :

k=0 $$W_0^{(n)}(a\ S_n+b) = a\ S_n+b = a\ W_0^{(n)}\ (S_n) + b$$

Supposons que

$$W_{k-1}^{(n)}\ (a\ S_n+b) = a\ W_{k-1}^{(n)}(S_n) + b$$

Montrons que cette propriété reste vrai pour k.

$$W_k^{(n)}(a\ S_n+b) = \frac{g_k(n+1)W_{k-1}^{(n)}(a\ S_n+b) - g_k(n)W_{k-1}^{(n+1)}(a\ S_{n+1}+b)}{g_k(n+1) - g_k(n)}$$

$$= \frac{a[g_k(n+1)W_{k-1}^{(n)}(S_n) - g_k(n)W_{k-1}^{(n+1)}(S_{n+1})] + b(g_k(n+1)-g_k(n))}{g_k(n+1) - g_k(n)}$$

$$= a\ W_k^{(n)}(S_n) + b.$$

Théorème 1 :

Si $\lim\limits_{n\to\infty} W_{k-1}^{(n)} = S$ *et si, il existe* α *et* β *tels que :*

$0 < \alpha < 1 < \beta$ *et* $g_k(n+1)/g_k(n) \notin [\alpha, \beta]$, $\forall n > N$, *alors* :

$\lim\limits_{n\to\infty} W_k^{(n)} = S.$

Preuve :

La règle de l'algorithme nous permet d'écrire :

$$W_k^{(n)} = W_{k-1}^{(n)} - \frac{\Delta W_{k-1}^{(n)}}{\dfrac{g_k(n+1)}{g_k(n)} - 1} \text{ où } \Delta W_{k-1}^{(n)} = W_{k-1}^{(n+1)} - W_{k-1}^{(n)}.$$

$\Delta W_{k-1}^{(n)}$ converge vers zéro et $\dfrac{g_k(n+1)}{g_k(n)}$ ne converge pas vers un nous donne la conclusion. □

Si la suite $(g_k(n))_{n\in\mathbb{N}}$ converge vers $\ell(k)$, alors pour que $\dfrac{g_k(n+1)}{g_k(n)}$ ne converge pas vers un, il faut que $\ell(k)$ soit nul.

Théorème 2

Si $\lim\limits_{n\to\infty} S_n = S$ *et si* $\lim\limits_{n\to\infty} \dfrac{g_k(n+1)}{g_k(n)} = b_{kk} \neq 1$, $\forall k$, *alors* :

$\lim\limits_{n\to\infty} W_k^{(n)} = S$, $\forall k$.

Preuve :

Par récurrence sur k et par application successive du théorème 1 nous obtenons celui-ci. □

III.5.2.2. Accélération de la convergence

Théorème 3 :

Sous les conditions du théorème 2,

$$\left\{\frac{W_{k-1}^{(n+1)}-S}{W_{k-1}^{(n)}-S} \to b_{kk},\ n \to \infty\right\} \Rightarrow \{W_k^{(n)} - S = 0\,(W_{k-1}^{(n)}-S),\ n \to \infty\}$$

Preuve :

$$W_k^{(n)}-S = W_{k-1}^{(n)} - \frac{W_{k-1}^{(n+1)}-W_{k-1}^{(n)}}{\dfrac{g_k(n+1)}{g_k(n)} - 1} - S$$

$$= W_{k-1}^{(n)} - (W_{k-1}^{(n)}-S)\,\frac{\left\{\dfrac{W_{k-1}^{(n+1)}-S}{W_{k-1}^{(n)} - S} - 1\right\}}{\dfrac{g_k(n+1)}{g_k(n)} - 1} - S$$

$$\Longrightarrow \frac{W_k^{(n)}-S}{W_{k-1}^{(n)}-S} = 1 - \frac{\dfrac{W_{k-1}^{(n+1)}-S}{W_{k-1}^{(n)} - S} - 1}{\dfrac{g_k(n+1)}{g_k(n)} - 1}$$

$$= - \frac{\dfrac{W_{k-1}^{(n+1)}-S}{W_{k-1}^{(n)} - S} - \dfrac{g_k(n+1)}{g_k(n)}}{\dfrac{g_k(n+1)}{g_k(n)} - 1} \to - \frac{b_{kk}-b_{kk}}{b_{kk}-1} = 0.$$

D'où la conclusion du théorème. □

Dans la règle de l'algorithme, posons

$$D_{k-1}^{(n)} = - g_k(n) \frac{\Delta W_{k-1}^{(n)}}{\Delta g_k(n)} \text{ où } \Delta \text{ porte sur l'indice n.}$$

D'où $W_k^{(n)} = W_{k-1}^{(n)} + D_{k-1}^{(n)}$.

$$\left.\begin{array}{l} \Longrightarrow \Delta W_k^{(n)} = \Delta W_{k-1}^{(n)} + \Delta D_{k-1}^{(n)} \\ \\ \Delta W_{k-1}^{(n)} \neq 0 \end{array}\right\} \Longrightarrow \frac{\Delta W_k^{(n)}}{\Delta W_{k-1}^{(n)}} = 1 + \frac{\Delta D_{k-1}^{(n)}}{\Delta W_{k-1}^{(n)}}$$

D'où le théorème de l'accélération de la convergence :

Théorème 4 :

$$\lim_{n\to\infty} \frac{\Delta W_k^{(n)}}{\Delta W_{k-1}^{(n)}} = 0 \Longleftrightarrow \lim_{n\to\infty} \frac{\Delta D_{k-1}^{(n)}}{\Delta W_{k-1}^{(n)}} = -1.$$

Ce théorème nous ramène à la procédure θ [5], que nous exposons dans le paragraphe suivant.

III.5.3. - Procédure θ

III.5.3.1. - Rappel sur cette procédure

Soit S l'ensemble des suites réelles ou complexes. Soit S' le sous-ensemble de S des suites $(b_n)_{n\in\mathbb{N}}$ virifiant :

$$\Delta b_n = b_{n+1} - b_n \neq 0, \forall n.$$

Soit f une application définie par :

$$\begin{aligned} f : S \times S' &\to S \\ ((a_n), (b_n)) &\to f((a_n), (b_n)) = (c_n). \end{aligned}$$

où $C_n = a_n - \frac{\Delta a_n}{\Delta b_n} b_n$, $\forall n$.

Soit un algorithme de la forme :

$C_n = a_n + b_n$ où $(a_n) \in S$ et $(b_n) \in S'$.

Soit θ la procédure consistant à remplacer C_n par

$$\theta(C_n) = a_n - \frac{\Delta a_n}{\Delta b_n} b_n = f((a_n), (b_n))_n = f(a_n, b_n)$$

où l'indice extérieur aux parenthèses correspond au $n^{\text{ième}}$ terme de la suite $f((a_n), (b_n))$.

$\theta(C_n)$ est dit θ-dérivé de C_n, ou de type-θ associé à C_n.

C_n est dit θ-primitif de $\theta(C_n)$.

Pour plus d'information sur cette procédure se reporter à [5].

(W) *est un θ-dérivé*

Soit (u) le procédé

$$(u) : \begin{cases} u_0^{(n)} = S_n \\ u_k^{(n)} = u_{k-1}^{(n)} + A_k\, g_k(n) \text{ ; où } A_k \text{ est une constante non nulle.} \\ n = 0, 1, 2, \ldots ; k = 1, 2, \ldots \end{cases}$$

Ce procédé se réduit à

$$u_k^{(n)} = S + \sum_{i=1}^{k} A_i\, g_i(n)$$

Appliquons la procédure θ à (u). Cela donne

$$\theta(u_k^{(n)}) = u_{k-1}^{(n)} - \frac{\Delta u_{k-1}^{(n)}}{\Delta g_k(n)} g_k(n).$$

En posant $W_k^{(n)} = \theta(u_k^{(n)})$ et $W_{k-1}^{(n)} = u_{k-1}^{(n)}$, dans la relation précédente on obtient la règle de (W).

III.5.3.2. - Application à (W)

Appliquons la procédure θ à

$$W_k^{(n)} = W_{k-1}^{(n)} + D_{k-1}^{(n)}.$$

On obtient

$$\theta(W_k^{(n)}) = W_{k-1}^{(n)} - \frac{\Delta W_{k-1}^{(n)}}{\Delta D_{k-1}^{(n)}} D_{k-1}^{(n)}.$$

Dans cette dernière relation, posons

$$\theta_k^{(n)} = \theta(W_k^{(n)}) \text{ et } \theta_{k-1}^{(n)} = W_{k-1}^{(n)}.$$

D'où le procédé de type-θ associé à (W) :

$$(\theta) : \begin{cases} \theta_0^{(n)} = S_n \\ \theta_k^{(n)} = \theta_{k-1}^{(n)} - \dfrac{\Delta\theta_{k-1}^{(n)}}{\Delta D_{k-1}^{(n)}} D_{k-1}^{(n)}, \; k = 1, 2, \ldots \; ; \; n = 0, 1, 2, \ldots \\ D_{k-1}^{(n)} = - \dfrac{\Delta\theta_{k-1}^{(n)}}{\Delta g_k(n)} g_k(n), \end{cases}$$

Ce nouvel algorithme peut se mettre sous la forme :

$$(\theta) : \begin{cases} \theta_0^{(n)} = S_n, \; n = 0, 1, 2, \ldots \\ \theta_k^{(n)} = \theta_{k-1}^{(n)} - \dfrac{\Delta\theta_{k-1}^{(n)}}{1 - \mu_k(n) \dfrac{\Delta\theta_{k-1}^{(n+1)}}{\Delta\theta_{k-1}^{(n)}}}, \; n = 0, 1, 2, \ldots \; ; \; k = 1, 2, \ldots \\ \mu_k(n) = \dfrac{g_k(n+1)}{g_k(n)} \; \dfrac{\Delta g_k(n)}{\Delta g_k(n+1)}, \; n = 0, 1, 2, \ldots \; ; \; k = 1, 2, \ldots \end{cases}$$

La quantité $\mu_k(n)$ peut aussi s'écrire :

$$\mu_k(n) = \frac{\dfrac{g_k(n+1)}{g_k(n)} - 1}{\dfrac{g_k(n+2)}{g_k(n+1)} - 1}.$$

Si $\dfrac{g_k(n+1)}{g_k(n)}$ converge vers une limite finie b_k différente de un, alors $\mu_k(n)$ converge vers un. De plus si $\dfrac{\Delta\theta_{k-1}^{(n+1)}}{\Delta\theta_{k-1}^{(n)}} \notin]\alpha, \beta[$ où $0 < \alpha < 1 < \beta$, à partir d'un certain rang N, alors on a :

Théorème 5 :

Si $(\theta_{k-1}^{(n)})$ *converge vers* S *quand* $n \to \infty$, *si il existe* $0 < \alpha < 1 < \beta$ *et* N *tel que* $\dfrac{\Delta\theta_{k-1}^{(n+1)}}{\Delta\theta_{k-1}^{(n)}} \notin]\alpha, \beta[$, $\forall n \geq N$; *et si* $\mu_k(n) \to 1$ *quand* $n \to \infty$, *alors* $\theta_k^{(n)}$ *converge vers* S *quand* $n \to \infty$.

Preuve : Evidente.

Théorème 6 :

Sous les hypothèses du théorème 5 et si $\lim\limits_{n\to\infty} \dfrac{\Delta\theta_{k-1}^{(n+1)}}{\Delta\theta_{k-1}^{(n)}} = \rho_{k-1} \in]-1, +1[- \{0\}$

alors $\lim\limits_{n\to\infty} \dfrac{\theta_k^{(n)}-S}{\theta_{k-1}^{(n)}-S} = 0 \iff \lim\limits_{n\to\infty} \dfrac{\theta_{k-1}^{(n+1)}-S}{\theta_{k-1}^{(n)}-S} = \rho_{k-1}$.

Preuve : D'après la relation de l'algorithme nous avons :

$$\theta_k^{(n)}-S = \theta_{k-1}^{(n)}-S - (\theta_{k-1}^{(n)}-S)\,\frac{\dfrac{\theta_{k-1}^{(n+1)}-S}{\theta_{k-1}^{(n)}-S} - 1}{1-\mu_k(n)\,\dfrac{\Delta\theta_{k-1}^{(n+1)}}{\Delta\theta_{k-1}^{(n)}}}$$

$$\Longrightarrow \frac{\theta_k^{(n)}-S}{\theta_{k-1}^{(n)}-S} = 1 - \frac{\dfrac{\theta_{k-1}^{(n+1)}-S}{\theta_{k-1}^{(n)}-S} - 1}{1-\mu_k(n)\,\dfrac{\Delta\theta_{k-1}^{(n+1)}}{\Delta\theta_{k-1}^{(n)}}}$$

Par passage à la limite nous obtenons

$$\ell im\,\frac{\theta_k^{(n)}-S}{\theta_{k-1}^{(n)}-S} = 0 \Longleftrightarrow \ell im\,\frac{\theta_{k-1}^{(n+1)}-S}{\theta_{k-1}^{(n)}-S} = \rho_{k-1}. \ \square$$

Exemple

Soit $(S_n)_{n\in\mathbb{N}}$ une suite à convergence linéaire, c'est-à-dire :

$$\ell im_{n\to\infty}\,\frac{S_{n+1}-S}{S_n-S} = \rho,\ \rho \in\,]-1, +1[,\ \rho \neq 0.$$

où $S = \ell im_{n\to\infty} S_n$.

Pour une telle suite nous avons aussi :

$$\frac{\Delta S_{n+1}}{\Delta S_n} = \frac{S_{n+1}-S}{S_n-S}\,\frac{\dfrac{S_{n+2}-S}{S_{n+1}-S} - 1}{\dfrac{S_{n+1}-S}{S_n-S} - 1} \to \rho\,\frac{\rho-1}{\rho-1} = \rho.$$

Prenons dans le procédé (W) $g_k(n) = (\Delta S_n)^k$.

Les procédés (W) et (θ) se réduisent à :

$$(W) : \begin{cases} W_0^{(n)} = S_n,\ n = 0, 1, 2,\dots \\[2ex] W_k^{(n)} = \dfrac{(\Delta S_{n+1})^k\, W_{k-1}^{(n)} - (\Delta S_n)^k\, W_{k-1}^{(n+1)}}{(\Delta S_{n+1})^k - (\Delta S_n)^k},\ n = 0, 1, 2,\dots\ ;\ k = 1, 2, 3,\dots \end{cases}$$

$$(\theta) : \begin{cases} \theta_0^{(n)} = S_n, \; n = 0,\, 1,\, 2,\ldots \\[2ex] \theta_k^{(n)} = \theta_{k-1}^{(n)} - \dfrac{\Delta\theta_{k-1}^{(n)}}{1-\mu_k(n)\,\dfrac{\Delta\theta_{k-1}^{(n+1)}}{\Delta\theta_{k-1}^{(n)}}}, \; k = 1,\, 2,\ldots \; ; \; n = 0,\, 1,\, 2,\ldots \\[4ex] \mu_k^{(n)} = \left(\dfrac{\Delta S_{n+1}}{\Delta S_n}\right)^k \dfrac{\Delta(\Delta S_n)^k}{\Delta(\Delta S_{n+1})^k} \end{cases}$$

Pour ce procédé de type θ particulier nous avons

Théorème 7 :

Si $(S_n)_{n\in\mathbb{N}}$ *est à convergence linéaire alors*

$\dfrac{\theta_k^{(n)}-S}{\theta_{k-1}^{(n)}-S}$ *converge vers zéro quand* n *tend vers l'infini, quelque soit* k.

Preuve :

Par récurrence sur k nous allons montrer que :

$$\lim_{n\to\infty} \frac{\theta_{k-1}^{(n+1)}-S}{\theta_{k-1}^{(n)}-S} = \rho \Rightarrow \lim_{n\to\infty} \frac{\theta_{k}^{(n+1)}-S}{\theta_{k}^{(n)}-S} = \rho .$$

où encore que

$$\lim_{n\to\infty} \frac{\Delta\theta_{k-1}^{(n+1)}}{\Delta\theta_{k-1}^{(n)}} = \rho \Rightarrow \lim_{n\to\infty} \frac{\Delta\theta_{k}^{(n+1)}}{\Delta\theta_{k}^{(n)}} = \rho .$$

Comme $\mu_k(n)$ converge vers 1, le théorème 6 nous permettra de conclure.

Pour k=0, (S_n) est à convergence linéaire.

$$\frac{\theta_1^{(n+1)}-S}{\theta_1^{(n)}-S}=\frac{\theta_0^{(n+1)}-S}{\theta_0^{(n)}-S}\;\frac{1-\{(\theta_0^{(n+2)}-S)/(\theta_0^{(n+1)}-S)-1\}/\{1-\mu_1(n+1)\;\Delta\theta_0^{(n+2)}/\Delta\theta_0^{(n+1)}\}}{1-\{(\theta_0^{(n+1)}-S)/(\theta_0^{(n)}-S)-1\}/\{1-\mu_1(n)\;\Delta\theta_0^{(n+1)}/\Delta\theta_0^{(n)}\}}.$$

$$\to \rho\,\frac{1-\dfrac{\rho-1}{1-\rho}}{1-\dfrac{\rho-1}{1-\rho}}=\rho.$$

Donc $(\theta_1^{(n)})_n$ est à convergence linéaire.

Supposons que la propriété est satisfaite jusqu'à k-1. Montrons la pour k.

$$\frac{\theta_k^{(n+1)}-S}{\theta_k^{(n)}-S}=\frac{\theta_{k-1}^{(n+1)}-S}{\theta_{k-1}^{(n)}-S}\;\frac{1-\left\{\dfrac{\theta_{k-1}^{(n+2)}-S}{\theta_{k-1}^{(n+1)}-S}-1\right\}\Big/\left\{1-\mu_k(n+1)\;\dfrac{\Delta\theta_{k-1}^{(n+2)}}{\Delta\theta_{k-1}^{(k+1)}}\right\}}{1-\left\{\dfrac{\theta_{k-1}^{(n+1)}-S}{\theta_{k-1}^{(n)}-S}-1\right\}\Big/\left\{1-\mu_k(n)\;\dfrac{\Delta\theta_{k-1}^{(n+1)}}{\Delta\theta_{k-1}^{(n)}}\right\}}$$

$$\to \rho\,\frac{1-\dfrac{\rho-1}{1-\rho}}{1-\dfrac{\rho-1}{1-\rho}}=\rho.$$

Donc $(\theta_k^{(n)})_{n\in\mathbb{N}}$ est à convergence linéaire. □

Méthode basée sur $\varepsilon_4^{(n)}$, appliquée à $x = e^{-x}$ de point fixe

$x_* = 0.567\ 143\ 290\ 409$ et $x_1 = 1$

1	.1000000000000000D+01
2	.5675269237743473D+00
3	.5671432904102058D+00
4	.5671432904095446D+00
5	.5671432904099196D+00
6	.5671432904097069D+00
7	.5671432904098275D+00
8	.5671432904097591D+00
9	.5671432904097979D+00
10	.5671432904097760D+00

Méthode basée sur $E_2^{(n)}$, appliquée à $x-e^{-x} = 0$

donnés	1	.1000000000000000D+01
	2	.2000000000000000D+00
	3	.3000000000000000D+00
	4	.4000000000000000D+00
	5	.5584837479805750D+00
	6	.5409799456300528D+00
	7	.5660598643490965D+00
	8	.5663345105162569D+00
	9	.5671388983923524D+00
	10	.5671393985878969D+00

Procédé (W) avec $g_i(n) = (\Delta S_n)^i$ appliqué à la suite : $x_{n+1} = e^{-x_n}$ $n \geq 2$

avec $x_1 = 1$. Sa limite est $x_* = 0.567\ 143\ 290\ 409$

n	x_n	Diagonale descendante(+)	Précision relative
1	.1000000000000000D+01	.1000000000000000D+01	.10D+01
2	.3678794411714423D+00	.3678794411714423D+00	.10D+01
3	.6922006275553464D+00	.5822260969956230D+00	.37D+00
4	.5004735005636368D+00	.5679469284970467D+00	.25D-01
5	.6062435350855974D+00	.5671048955523903D+00	.15D-02
6	.5453957859750270D+00	.5671419214222825D+00	.65D-04
7	.5796123355033769D+00	.5671433300646206D+00	.25D-05
8	.5601154613610891D+00	.5671432912069836D+00	.69D-07
9	.5711431150801770D+00	.5671432903981970D+00	.14D-08
10	.5648793473910495D+00	.5671432904096723D+00	.20D-10
11	.5684287250290607D+00	.5671432904097846D+00	.20D-12
12	.5664147331468833D+00	.5671432904097839D+00	.13D-14

(+) $W_0^{(1)}$, $W_0^{(2)}$, $W_1^{(1)}$, $W_2^{(1)}$, $W_3^{(1)}$, $W_4^{(1)}$, $W_5^{(1)}$, $W_6^{(1)}$, $W_7^{(1)}$.

CHAPITRE IV

APPLICATION DES APPROXIMANTS DE PADE A LA RESOLUTION DES EQUATIONS NON LINEAIRES

INTRODUCTION

L'utilisation des approximants de Padé pour résoudre des équations non linéaires a été examinée par plusieurs auteurs. Récemment Nourein [18], puis Claessens-Loisou-Wuytack [10] ont donné deux techniques de calcul ne mettant en jeu que l'approximant de Padé [1/p]. *En fait c'est la seule possibilité, car ils s'intéressent aux zéros du numérateur de l'approximant* [q/p]. *Si* $q \geq 2$, *le nombre de zéros réels, s'ils existent, peut être supérieur ou égal à deux, ce qui rend le choix d'un zéro difficile.*

Il reste une autre possibilité, c'est l'extrapolation. Dans le cas d'une racine simple c'est la fonction inverse qui sera approximée par un approximant de Padé après avoir effectué un changement de variable convenable. Malgré la complexité des calculs, tout approximant de Padé [p/q] *est une approximation de* x_*, *la racine de l'équation* $f(x) = 0$ *à l'ordre* $p+q+1$.

Nous terminons par quelques essais numériques.

IV.1. - UTILISATION DIRECTE DES APPROXIMANTS DE PADE

Soit à résoudre

$$f(x) = 0.$$

Soit x_* une racine réelle simple de f. C'est-à-dire : $f(x_*) = 0$ et $f'(x_*) \neq 0$.

IV.1.1. - Mise en oeuvre du procédé de Nourein [18]

Soit w une approximation de x_*.

Soit la fonction g définie par

$$g(t) = f(w + t)$$

Résoudre $f(x) = 0$ revient à résoudre $g(t) = 0$.

Soit t_* une racine réelle simple de g. Alors $w + t_* = x_*$.

En utilisant le développement de Taylor de g au voisinage de zéro, on a

$$\begin{aligned} g(t) &= g(0) + \frac{g'(0)}{1!}\, t + \ldots + \frac{g^{(r)}(0)}{r!}\, t^r + \ldots \\ &= c_0 + c_1 t + \ldots + c_r\, t^r + \ldots, \end{aligned}$$

avec $c_i = c_i(w) = \dfrac{g^{(i)}(0)}{i!} = \dfrac{f^{(i)}(w)}{i!}$

L'idée est d'approximer g par un approximant de Padé [q/p]. Puis de chercher un zéro de cet approximant. Or si $q \geq 2$ on risque d'avoir plus d'une racines réelles et donc on ne saura pas laquelle il faut prendre pour approcher t_*. Donc on doit prendre dors-et-déjà q égal à un.

Soit P l'approximant de Padé [1/p]. On a

$$P(t) = \frac{a_0 + a_1 t}{1+b_1 t + \ldots + b_p t^p} = g(t) + O(t^{p+2}).$$

$$P(t) = 0 \iff a_0 + a_1 t = 0 \iff t = -a_0/a_1.$$

Posons $\tilde{t} = -a_0/a_1$. $\tilde{t}$ est donc une approximation de t_* car P est une approximation de g. Par conséquent $w + \tilde{t}$ est une approximation de x_*.

Posons $\tilde{w} = w + \tilde{t}$. Donc à partir d'une approximation w de x_*, on vient de construire une nouvelle approximation $\tilde{w}$. Et on a

$$\tilde{w}-x_* = w+\tilde{t}-x_* = \frac{f(\tilde{t}+w)}{f'(\xi)} \text{ avec } \xi \in \,]\min(x_*, \tilde{t}+w), \max(x_*, \tilde{t}+w)[.$$

$$= \frac{g(\tilde{t})}{f'(\xi)} = O(\tilde{t}^{p+2})$$

$$= O(w-x_*)^{p+2}.$$

Car d'une part on a

$$g(\tilde{t}) = f(w+\tilde{t}) = O(\tilde{t}^{p+2}),$$

et d'autre part on a

$$f(w) = f'(\eta)\,(w-x_*) = g(\tilde{t}) + f'(\gamma)\tilde{t} \text{ avec } \eta \in \,]\min(x_*, w), \max(x_*, w)[$$
$$\text{et } \gamma \in \,]\min(w, w+\tilde{t}), \max(w, w+\tilde{t})[$$
$$= O(\tilde{t}^{p+2}) + f'(\gamma)\tilde{t}$$
$$= \tilde{t}(f'(\gamma) + O(t^{p+1}))$$

$$\Rightarrow \tilde{t} = \frac{f'(\eta)}{f'(\gamma)+O(\tilde{t}^{p+1})}\,(w-x_*)$$

L'approximation $\tilde{w}$ est donnée aussi par

$$\tilde{w} = w - a_0/a_1.$$

Les coefficients a_0 et a_1 sont déterminés comme ce qui suit

$$\frac{a_0 + a_1 t}{1+b_1 t + \ldots + b_p t^p} = c_0 + c_1 t + \ldots + c_{p+1} t^{p+1} + O(t^{p+2})$$

$$\Rightarrow \quad \begin{cases} a_0 = c_0 \\ a_1 = c_1 + b_1\, c_0 \\ 0 = c_m + c_{m-1}\, b_1 + \ldots + c_0 b_m \qquad \text{pour } m = 2, 3, \ldots, p \\ 0 = c_{p+1} + c_p b_1 + \ldots + c_1 b_p \end{cases}$$

Le système $\begin{cases} c_1 b_1 + c_0 b_2 & = - c_2 \\ c_2 b_1 + c_1\, b_2 + c_0 b_3 & = - c_3 \\ \cdots\cdots\cdots\cdots\cdots\cdots & \\ c_p b_1 + c_{p-1} b_2 + c_{p-2} b_3 + \ldots + c_1 b_p & = - c_{p+1}, \end{cases}$

nous donne

$b_1 = - \tilde{H}_p / H_p$, où

$$H_p = \begin{vmatrix} c_1 & c_0 & 0 & \cdots & 0 \\ c_2 & \ddots & \ddots & \ddots & \vdots \\ \cdots & \cdots & \cdots & \cdots & 0 \\ c_{p-1} & \cdots & \cdots & \ddots & c_0 \\ c_p & c_{p-1} & \cdots & \cdots & c_1 \end{vmatrix}$$

et où

$$\tilde{H}_p = \begin{vmatrix} c_2 & c_0 & 0 & \cdots & 0 \\ c_3 & c_1 & \ddots & \ddots & \vdots \\ \cdots & \cdots & \cdots & \cdots & 0 \\ \cdots & \cdots & \cdots & c_1 & c_0 \\ c_{p+1} & c_{p-1} & \cdots & \cdots & c_1 \end{vmatrix}$$

D'où l'on tire $a_1 = c_1 - c_0\, \tilde{H}_p / H_p$,

$$\text{et } \tilde{w} = w - a_0/a_1 = w - \frac{c_0}{c_1 - c_0\, \tilde{H}_p / H_p} = w - \frac{\dfrac{c_0}{c_1}}{1 - \dfrac{c_0}{c_1}\, \tilde{H}_p / H_p}$$

En résumé, à partir d'une approximation w on a obtenu une nouvelle approximation d'ordre p+2, à savoir $\tilde{w}$. Donc on peut itérer en utilisant la fonction

$$F_p(w) = - \frac{u(w)}{1 - u(w)\,\dfrac{\tilde{H}_p(w)}{H_p(w)}}$$

où $u(w) = \dfrac{f(w)}{f'(w)} = \dfrac{c_0}{c_1}$, $H_p(w) = H_p$ et $\tilde{H}_p(w) = \tilde{H}_p$.

C'est le procédé itératif de Nourein [18]. Il coïncide avec d'autres procédés connus, pour les premières valeurs de p.

Posons $D_r = D_r(w) = \dfrac{f^{(r)}(w)}{f'(w)}$ pour $r = 2, 3, \ldots$

a) $p = 0$, F_0 est le procédé de Newton

$$F_0(w) = w - u(w)$$

b) $p = 1$, F_1 est le procédé itératif de Halley-Schröder.

$$F_1(w) = w - \frac{u(w)}{1 - \frac{1}{2}\, u(w)\, D_2} \quad \text{avec } D_2 = \frac{f''(w)}{f'(w)}.$$

c) $p = 2$, F_2 est le procédé itératif de Kiss [18]

$$F_2(w) = w - u(w)\,(1 - \tfrac{1}{2}\, u(w)\, D_2)/(1 - u(w)\, D_2 + (u(w))^2\, D_3/6).$$

avec $D_2 = f''(w)/f'(w)$ et $D_3 = f^{(3)}(w)/f'(w)$.

Soit x_0 un point voisin de x_*, la suite $(x_n)_{n\in\mathbb{N}}$ construite par F_p, c'est-à-dire $x_{n+1} = F_p(x_n)$, est telle que

$$x_{n+1} - x_* = O(x_n - x_*)^{p+2}.$$

Dans ce qui suit on donne deux techniques de calcul de F_p dues à Nourein et à Claessens-Loisou-Wuytack.

IV.1.2. - Techniques de Calcul

* *Procédure de Nourein*

Soit A_{p+1-q} le déterminant d'ordre p+1-q défini par

$$A_{p+1-q} = \begin{vmatrix} c_1 & c_0 & 0 & \cdots & 0 \\ c_2 & c_1 & c_0 & \ddots & \vdots \\ \cdots & \cdots & \cdots & \cdots & 0 \\ c_{p-q} & c_{p-1-q} & \cdots c_1 & \cdots & c_0 \\ c_{p+1-q}^{(q)} & \cdots & \cdots & c_2^{(q)} & c_1^{(q)} \end{vmatrix}$$

Multiplions la dernière ligne par $-\frac{c_0}{c_1^{(q)}}$ et ajoutons à l'avant dernière. Développons le déterminant ainsi obtenu suivant la dernière colonne. On obtient

$$\frac{c_0}{c_1^{(q)}} A_{p+1-q} = c_0 A_{p+1-(q+1)} = c_0 A_{p-q}$$

$$\Rightarrow A_{p+1-q} = c_1^{(q)} A_{p-q}$$

$$A_{p-q} = \begin{vmatrix} c_1 & c_0 & 0 & \cdots & 0 \\ c_2 & c_1 & c_0 & \ddots & \vdots \\ \cdots & \cdots & \cdots & \cdots & 0 \\ c_{p-q-1} & \cdots & \cdots & \cdots & c_0 \\ c_{p-q}^{(q+1)} & \cdots & \cdots & \cdots & c_1^{(q+1)} \end{vmatrix}$$ avec $c_{p-q-i}^{(q+1)} = c_{p-q-i} - \frac{c_0}{c_1^{(q)}} c_{p+1-q-i}^{(q)}$

pour i = 0, 1,..., p-q-1
et i ≥ 0.

ces dernières relations peuvent encore s'écrire

$$c_{p-\ell}^{(q+1)} = c_{p-\ell} - \frac{c_0}{c_1^{(q)}} c_{p+1-\ell}^{(q)} \text{ pour } \ell = q, \ldots, p-1.$$

Elles sont vraies pour $q = 1, 2, \ldots, p-1, p$. Il en est de même de la relation liant les déterminants A_{p+1-q} et A_{p-q}.

$$A_{p+1-q} = c_1^{(q)} A_{p+1-(q+1)} = c_1^{(q)} c_1^{(q+1)} A_{p+1-(q+2)}$$

$$= \ldots = \prod_{i=0}^{p} c_1^{(q+i)} A_1.$$

$$= \prod_{i=0}^{p} c_1^{(q+i)} \text{ avec } A_1 = 1.$$

Pour $q = 0$ on a

$$A_{p+1} = \prod_{i=0}^{p} c_1^{(i)}.$$

Si on choisit $c_i^{(0)} = c_i$ pour $i = 0, 1, \ldots, p+1$, alors on a d'une part

$$A_{p+1} = c_1 H_p - c_0 \tilde{H}_p,$$

et d'autre part

$$A_p = H_p.$$

Or $A_p = \prod_{i=0}^{p-1} c_1^{(i)}$, donc on a

$$A_{p+1} = c_1^{(p)} A_p \Rightarrow c_1^{(p)} = \frac{A_{p+1}}{A_p}.$$

$\tilde{H}_p/H_p$ est donné par

$$-\frac{c_0}{c_1} \tilde{H}_p/H_p = c_1^{(p)} - 1.$$

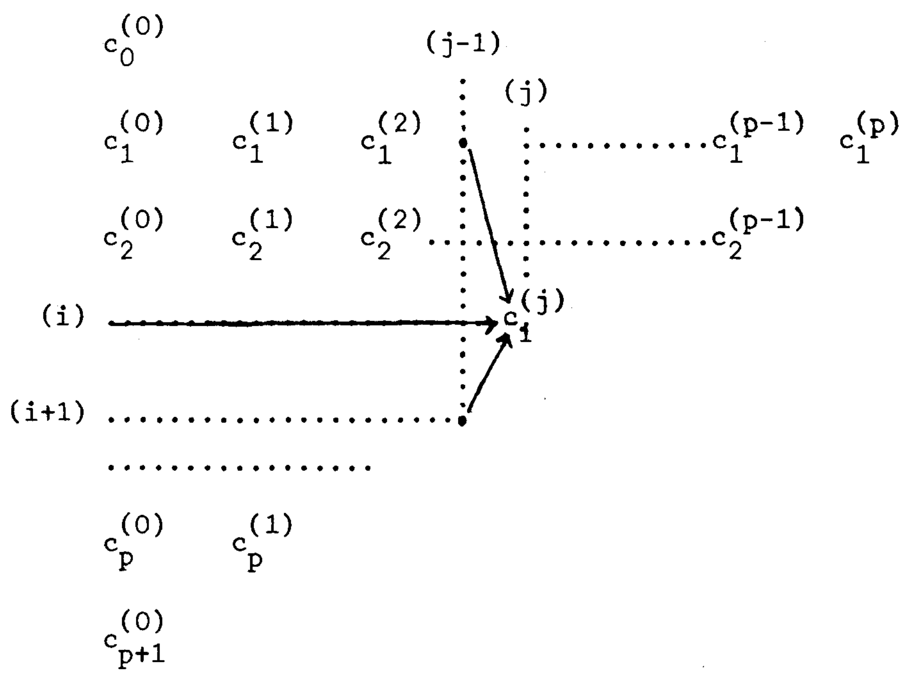
Par conséquent $F_p(w)$ est donnée par

$$F_p(w) = w - \frac{c_0(w)}{c_1^{(p)}}.$$

On a la procédure suivante pour calculer $c_1^{(p)}$:

$$
\begin{cases}
c_i^{(0)} = c_i, \; i = 0, 1, \ldots, p+1 \\[2ex]
c_m^{(q)} = c_m - c_{m+1}^{(q-1)} \cdot \dfrac{c_0}{c_1^{(q-1)}}, \; m = 1, 2, \ldots, p+1-q \text{ et } q = 1, 2, \ldots, p
\end{cases}
$$

Les quantités $c_i^{(j)}$ peuvent être présentées dans le tableau suivant



$$
c_i^{(j)} = c_i - c_{i+1}^{(j-1)} \cdot \frac{c_0}{c_1^{(j-1)}}
$$

Cette technique nécessite pour calculer $c_1^{(p)}$: $\frac{p(p+1)}{2}$ soustractions, $\frac{p(p+1)}{2}$ multiplications et p divisions. Pour passer dans le tableau de la colonne i à la colonne i+1 on doit diviser par le facteur $c_1^{(i)}$. Cela peut entraîner une instabilité numérique.

** *Procédure de Claessens-Loisou-Wuytack* [10]

Considérons le déterminant

$$A_{p+1} = \begin{vmatrix} c_1 & c_0 & 0 & \cdots & 0 \\ c_2 & c_1 & c_0 & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ \vdots & & & \ddots & c_0 \\ c_{p+1} & \cdots & \cdots & c_2 & c_1 \end{vmatrix} = c_1 A_p - c_0 \tilde{H}_p = c_1 H_p - c_0 \tilde{H}_p$$

$$\Longrightarrow \frac{\tilde{H}_p}{H_p} = \frac{c_1}{c_0} - \frac{1}{c_0} \frac{A_{p+1}}{A_p}.$$

La fonction F_p s'écrit alors comme

$$F_p(w) = w - c_0\, A_p / A_{p+1}.$$

Soit w_p le déterminant suivant

$$w_p = \begin{vmatrix} c'_1 & 1 & 0 & \cdots & \cdots & 0 \\ c'_2 & c'_1 & \ddots & 0 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \ddots & 0 \\ \vdots & & & \ddots & \ddots & 1 \\ c'_p & \cdots & \cdots & \cdots & \cdots & c'_1 \end{vmatrix} \quad \text{où } c'_i = \frac{c_i}{c_0}.$$

On a $c_0^p\, w_p = A_p$ et $c_0^{p+1}\, w_{p+1} = A_{p+1}$

$$\Longrightarrow \frac{1}{c_0} \frac{A_p}{A_{p+1}} = \frac{c_0^p}{c_0^{p+1}} \frac{A_p}{A_{p+1}} = w_p / w_{p+1}.$$

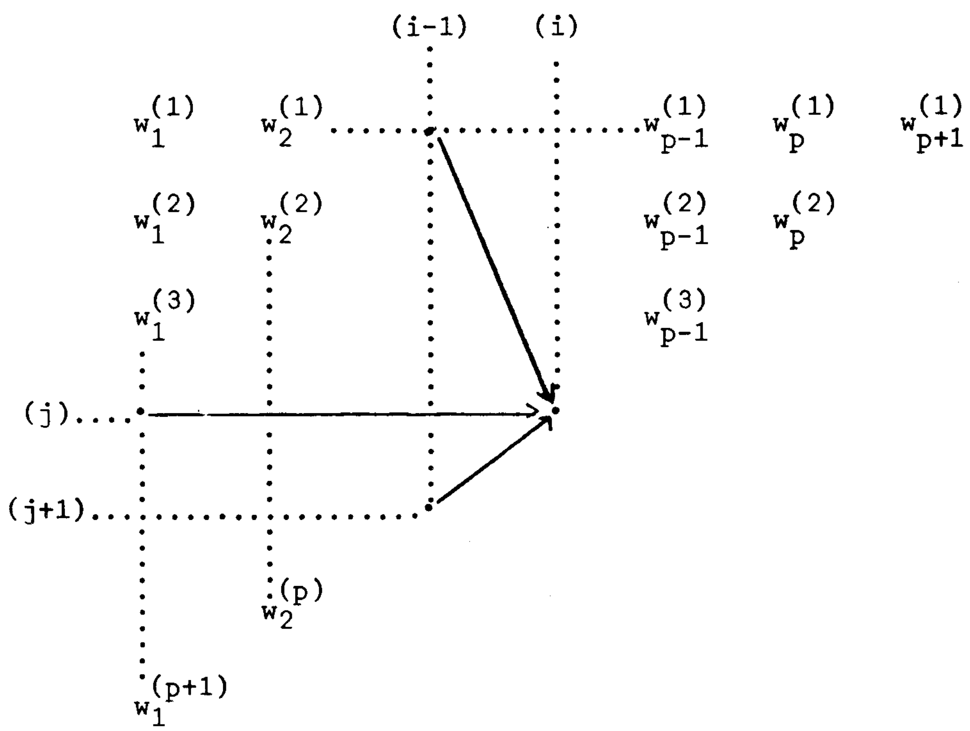
Par conséquent F_p peut s'écrire

$$F_p(w) = w - w_p / w_{p+1}.$$

On a l'algorithme suivant pour déterminer w_p et w_{p+1} :

$$\begin{cases} w_1^{(j)} = c'_j = c_j/c_0, \; j = 1,\ldots,\; p+1 \\ w_i^{(j)} = w_1^{(j)}\, w_{i-1}^{(1)} - w_{i-1}^{(j+1)}, \; i = 2,\ldots,\; p+1 \text{ et } j = 1,\ldots,\; p+2-i \\ w_i = w_i^{(1)}, \; i = 1,\ldots,\; p+1 \end{cases}$$

Les quantités $w_i^{(j)}$ peuvent être rangées dans le tableau suivant :



Les relations de l'algorithme précédent sont obtenues de la manière suivante :

Soit $w_i^{(1)}$ ($w_0^{(1)} = 1$) le déterminant d'ordre (i)

$$w_i^{(1)} = \begin{vmatrix} c'_1 & 1 & 0 & \cdots & 0 \\ c'_2 & c'_1 & 1 & \ddots & \vdots \\ \cdots & \cdots & \cdots & \cdots & 0 \\ & & & & 0 \\ & & & \ddots & 1 \\ c'_i & c'_{i-1} & \cdots & c'_2 & c'_1 \end{vmatrix} \quad \text{pour } i = 1, 2, \ldots, p+1.$$

Soit $k \in \{1, 2, \ldots, p+1\}$. Le déterminant $w_k^{(1)}$ est une combinaison linéaire des $w_i^{(1)}$ pour $i = 1, 2, \ldots, k-1$.

$$w_k^{(1)} = \begin{vmatrix} c_1' & 1 & 0 \ldots\ldots 0 \\ c_2' & c_1' & 1 \quad 0 \ldots 0 \\ \cdots & \cdots & \cdots \\ c_k' & c_{k-1}' & \ldots c_1' \end{vmatrix} = c_1' \begin{vmatrix} c_1' & 1 & 0 \ldots\ldots 0 \\ c_2' & c_1' & 1 \quad 0 \ldots 0 \\ \cdots & \cdots & \cdots \\ c_{k-1}' & c_{k-2}' & \ldots c_1' \end{vmatrix} - \begin{vmatrix} c_2' & 1 & 0 \ldots\ldots 0 \\ c_3' & c_1' & 1 \quad 0 \ldots 0 \\ \cdots & \cdots & \cdots \\ c_k' & c_{k-2}' & \ldots\ldots c_1' \end{vmatrix}$$

$$= c_1' w_{k-1}^{(1)} - c_2' w_{k-2}^{(1)} + \begin{vmatrix} c_3' & 1 & 0 \ldots\ldots 0 \\ c_4' & c_1' & 1 \quad 0 \ldots 0 \\ \cdots & \cdots & \cdots \\ c_k' & c_{k-3}' & \ldots c_1' \end{vmatrix}$$

$$= c_1' w_{k-1}^{(1)} - c_2' w_{k-2}^{(1)} + c_3' w_{k-3}^{(1)} - \begin{vmatrix} c_4' & 1 & 0 \ldots\ldots 0 \\ c_5' & c_1' & 1 \quad 0 \ldots 0 \\ \cdots & \cdots & \cdots \\ c_k' & c_{k-4}' & \ldots\ldots c_1' \end{vmatrix}$$

$$= \ldots = \sum_{i=1}^{k} c_i' \, w_{k-i}^{(1)} \quad \text{avec } w_0^{(1)} = 1.$$

Soit maintenant le déterminant $w_i^{(j)}$ d'ordre (i) défini par

$$w_i^{(j)} = \begin{vmatrix} c_j' & 1 & 0 \ldots\ldots\ldots 0 \\ c_{j+1}' & c_1' & 1 \ddots \quad \vdots \\ \vdots & & \ddots \quad 0 \\ \vdots & & \ddots \quad 1 \\ c_{i+j-1}' & c_{i-1}' & \ldots\ldots\ldots c_1' \end{vmatrix} = \begin{vmatrix} c_j' & 1 \quad 0 \ldots\ldots\ldots 0 \\ c_{j+1}' & \\ \vdots & w_{i-1}^{(1)} \\ c_{i+j-1}' & \end{vmatrix}$$

$$= c_j' w_{i-1}^{(1)} - \begin{vmatrix} c_{j+1}' & 1 & 0 \ldots\ldots\ldots 0 \\ c_{j+2}' & c_1' & 1 \ddots \quad \vdots \\ \vdots & & \ddots \quad 0 \\ \vdots & & \ddots \quad 1 \\ c_{i+j-1}' & c_{i-2}' & \ldots\ldots\ldots c_1' \end{vmatrix} = c_j' w_{i-1}^{(1)} - w_{i-1}^{(j+1)}$$

$= w_1^{(j)}\, w_{i-1}^{(1)} - w_{i-1}^{(j+1)}$ avec $w_1^{(j)} = c'_j$.

Pour $i = p+1$ on a besoin de connaître $w_1^{(j)}$, $w_p^{(1)}$ et $w_p^{(j+1)}$.

Or le coefficient c'_j d'indice le plus grand est c'_{p+1}.

Donc pour que le déterminant $w_p^{(j+1)}$ ait un sens il faut que

$$p+j \leq p+1$$

$$\Rightarrow j = 1$$

Donc seuls $w_p^{(1)}$ et $w_p^{(2)}$ sont utilisés par la relation précédente, d'où le calcul de $w_{p+1}^{(1)}$.

Pour $i \in \{2,\ldots, p+1\}$, le coefficient c'_{i+j-1} doit avoir un indice $i+j-1 \leq p+1 \Rightarrow j \leq p+2-i$, d'où les relations de récurrence de l'algorithme précédent.

On voit que cette technique utilise également $\frac{p(p+1)}{2}$ additions et $\frac{p(p+1)}{2}$ multiplications, après avoir au départ effectué p divisions.

Dans le paragraphe suivant on va voir que tout approximant de Padé est une approximation de x_*.

IV.2. - UTILISATION RECIPROQUE DES APPROXIMANTS DE PADE

Nous allons exposer une technique analogue à celle introduite par Traub [24] pour construire des méthodes itératives d'un ordre arbitraire.

Soit à résoudre $f(x) = 0$. Soit x_* une racine simple de f.

Soit F l'application inverse de f ; $F = f^{-1}$ et $F(0) = x_*$.

Soit w une approximation connue de x_*. Posons $\tau = f(w)$ et considérons l'application g définie par :

$$g(t) = F(\tau - t)$$

On a $g(\tau) = F(\tau-\tau) = F(0) = x_*$.

Supposons que l'on puisse développer $g(t)$ comme suit :

$$g(t) = c_0 + c_1 t + c_2 t^2 + \ldots + c_i t^i + \ldots$$

où $c_i = (-1)^i \dfrac{F^{(i)}(\tau)}{i!}$

Considérons l'approximant de Padé

$$[m/n]_g(t) = \frac{a_0 + a_1 t + \ldots + a_m t^m}{1 + b_1 t + \ldots + b_n t^n}.$$

Par définition de l'approximant de Padé, on a

$$[m/n]_g(t) = g(t) + O(t^{m+n+1}).$$

Comme $[m/n]_g(t)$ est une approximation de $g(t)$, alors $[m/n]_g(\tau)$ sera une approximation de $g(\tau)$ et donc de x_*.

Par la suite on a

$$\begin{aligned} [m/n]_g(\tau) &= g(\tau) + O(\tau^{m+n+1}) \\ &= x_* + O([f(w)]^{m+n+1}) \\ &= x_* + O((w-x_*)^{m+n+1}). \end{aligned}$$

où encore

$$\frac{a_0 + a_1[f(w)] + \ldots + a_m[f(w)]^m}{1 + b_1[f(w)] + \ldots + b_n[f(w)]^n} = x_* + O((w-x_*)^{m+n+1}).$$

Les coefficients b_i et a_i de l'approximant de Padé sont déterminés par les équations :

$$\begin{cases} c_{m-n+1}\, b_n + \ldots + c_m\, b_1 = c_{m+1} \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ c_m\, b_n + \ldots\ldots + c_{m+n-1}\, b_1 = c_{m+n} \end{cases}$$

$$\begin{cases} a_0 = c_0 \\ a_1 = c_0\, b_1 + c_1 \\ a_2 = c_0\, b_2 + c_1\, b_1 + c_2 \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ a_m = c_{m-n}\, b_n + \ldots + c_{m-1}\, b_1 + c_m \end{cases}$$

avec la convention : $c_i = 0$ pour $i < 0$.

Les coefficients c_i sont donnés par

$$c_i = (-1)^i \frac{F^{(i)}(\tau)}{i!}.$$

Et on sait d'après [24] que la $i^{\text{ième}}$ dérivée de F peut s'exprimer en fonction des dérivées de f.

$$\tau = f(w)$$

$$F^{(i)}(\tau) = [f'(w)]^{-i} \sum_{\ell=1}^{K_i} (-1)^{r_\ell}\, (i+r_\ell-1)! \prod_{j=2}^{i} \frac{A_j^{\beta_{j,\ell}}}{\beta_{j,\ell}!}$$

avec $(\beta_{2,\ell},\ldots,\beta_{i,\ell})$ solution de l'équation

$$(*) : \begin{cases} \sum_{j=2}^{i} (j-1)\, \beta_{j,\ell} = i-1 \\ \beta_{j,\ell} \in \mathbb{N} \quad \text{pour } j = 2,\ldots, i : \end{cases}$$

K_i est le nombre des $(i-1)$-uplets solutions de $(*)$.

et $r_\ell = \sum_{j=2}^{i} \beta_{j,\ell}$ pour $\ell = 1,\ldots, K_i$

On rappelle aussi que $A_i = \frac{f^{(i)}(w)}{i!f'(w)}$.

Détermination des premières dérivées de F.

Nous avons calculé au chapitre I les quatres premières dérivées de F au point 0. Le travail est le même pour tout autre point.

Pour $\tau = f(w)$ on a :

$$F'(\tau) = \frac{1}{f'(w)}$$

$$F''(\tau) = -\frac{f''(w)}{[f'(w)]^3}$$

$$F^{(3)}(\tau) = \frac{3[f''(w)]^2 - f'(w)\ f^{(3)}(w)}{[f'(w)]^5}$$

$$F^{(4)}(\tau) = \frac{10\ f'(w)f''(w)f^{(3)}(w) - 15[f''(w)]^3 - f'(w)\ f^{(4)}(w)}{[f'(w)]^7}$$

etc...

On voit que l'expression de ces dérivées devient de plus en plus compliquée ce qui limite l'utilisation de cette méthode. On va voir que, par ce moyen, on retrouve certaines méthodes connues.

a) *Méthode de Newton*

Par itération sur l'approximant de Padé [1/0] nous obtenons une méthode d'ordre deux ; c'est la méthode de Newton.

$$\begin{aligned}[1/0]_g(t) &= g(t) + O(t^2) \\ &= c_0 + c_1 + O(t^2) \\ &= a_0 + a_1 t.\end{aligned}$$

Donc on a

$$a_0 = c_0 = w$$

$$a_1 = c_1 = -\frac{F'(\tau)}{1!} = -\frac{1}{f'(w)}.$$

Posons $\tilde{w} = [1/0]_g(\tau)$. On a

$$\tilde{w} = w - \frac{f(w)}{f'(w)} = g(\tau) + O(\tau^2)$$

$$= x_* + O((w-x_*)^2).$$

D'où l'on tire la fonction itérative de Newton

$$\phi_{1,0}(x) = x - \frac{f(x)}{f'(x)}.$$

b) *Une autre méthode d'ordre deux* : $\phi_{0,1}(x)$.

L'approximant de Padé [0/1] est tel que

$$[0/1]_g(t) = \frac{a_0}{1+b_1 t} = c_0 + c_1 t + O(t^2)$$

$$\Rightarrow \begin{cases} a_0 = c_0 = w \\ b_1 = -\dfrac{c_1}{c_0} = \dfrac{1}{wf'(w)}. \end{cases}$$

Par suite $[0/1]_g(\tau)$ est donné par

$$\tilde{w} = [0/1]_g(\tau) = w - \frac{wf(w)}{wf'(w) + f(w)}$$

D'où l'on a la fonction itérative $\phi_{0,1}$

$$\phi_{0,1}(x) = x - \frac{xf(x)}{xf'(x) + f(x)}.$$

Remarque :

Soit $h(x) = xf(x)$. La fonction $\phi_{0,1}(x)$ n'est rien d'autre que le processus de Newton appliqué à $h(x)$.

c) ***Méthode de Schröder*** : $\phi_{1,1}(x)$

Elle correspond à l'approximant de Padé $[1/1]_g(t)$.

$$[1/1]_g(t) = \frac{a_0 + a_1 t}{1 + b_1 t} = c_0 + c_1 t + c_2 t^2 + O(t^3)$$

$$\Longrightarrow \begin{cases} a_0 = c_0 = w \\ a_1 = c_1 + b_1\, c_0 \\ 0 = c_2 + b_1\, c_1 \end{cases} \Longrightarrow \begin{cases} a_0 = w \\ a_1 = (c_1^2 - c_0\, c_2)/c_1 \\ b_1 = -\dfrac{c_2}{c_1} \end{cases}$$

L'approximant $[1/1]_g(\tau)$ s'écrit

$$[1/1]_g(\tau) = \frac{c_0 c_1 + (c_1^2 - c_0 c_2)\tau}{c_1 - c_2\,\tau}$$

$$= \frac{F(\tau)\,[F'(\tau) + F''(\tau)\,\tau/2] - [F'(\tau)]^2\tau}{F'(\tau) + F''(\tau)\,\tau/2}$$

$$= F(\tau) - \frac{[2\,F'(\tau)]^2\,\tau}{2F'(\tau) + F''(\tau)\tau}$$

$$= w - 2\,\frac{f(w)\,f'(w)}{2[f'(w)]^2 - f(w)f''(w)}$$

Par conséquent on a la méthode de Schröder :

$$\phi_{1,1}(x) = x - \frac{2\,f(x)\,f'(x)}{2[f'(x)]^2 - f(x)f''(x)}.$$

Par définition de l'approximant de Padé $[1/1]_g(\tau)$ on a :

$$[1/1]_g(\tau) = \phi_{1,1}(w) = g(\tau) + O(\tau^3)$$

$$= x_* + O(w-x)^3.$$

d) *Méthode $\phi_{2,0}$ d'ordre 3*

Elle correspond à l'approximant $[2/0]_g(t)$. Elle est donnée par

$$\phi_{2,0}(x) = x - \frac{f(x)}{f'(x)} - \frac{1}{2}\,\frac{f''(x)}{[f'(x)]^2}\,[f(x)]^2.$$

e) $\phi_{0,2}$ d'ordre 3

Elle correspond à l'approximant de Padé $[0/2]_g(t)$.

$$[0/2]_g(\tau) = g(\tau) + O(\tau^3)$$

$$= x_* + O(w-x_*)^3.$$

$$[0/2]_g(t) = g(t) + O(t^3)$$

$$= \frac{a_0}{1+b_1t+b_2t^2} = c_0 + c_1 t + c_2 t^2 + O(t^3).$$

$$\Longrightarrow \begin{cases} a_0 = c_0 \\ 0 = b_1\, c_0 + c_1 \\ 0 = b_2\, c_0 + b_1\, c_1 + c_2 \end{cases} \Longrightarrow \begin{cases} a_0 = c_0 \\ b_1 = -\, c_1/c_0 \\ b_2 = -\dfrac{c_2}{c_0} + (\dfrac{c_1}{c_0})^2 \end{cases}$$

$$[0/2]_g(t) = \frac{c_0}{1 - \dfrac{c_1}{c_0}t + \left[(\dfrac{c_1}{c_0})^2 - \dfrac{c_2}{c_0}\right]t^2} = w - \frac{w[2wf'^2t + (wf'' + 2f')t^2]}{2\,w^2f'^3 + 2wf'^2t + (wf''+2f')t^2}$$

où les dérivées $f^{(i)}$ sont calculées au point w.

D'où la fonction itérative suivante

$$\phi_{0,2}(x) = x - \frac{x(2xf'^2f + (xf''+2f')f^2)}{2x^2f'^3 + 2xf'^2f + (xf''+2f')f^2}.$$

f) Méthode $\phi_{m,n}$ d'ordre m+n+1

D'une manière générale l'approximant de Padé $[m/n]_g(t)$ donne naissance à une méthode itérative $\phi_{m,n}$ d'ordre m+n+1.

En effet $\tilde{w}$ est tel que

$$\tilde{w} = [m/n]_{g}(\tau) = [m/n]_{g}(f(w)) = g(\tau) + O(\tau^{m+n+1})$$

$$= x_* + O[(w-x_*)^{m+n+1}].$$

$\phi_{m,n}$ est donnée par

$$\phi_{m,n}(x) = [m/n]_g (f(x))$$

et $\quad x_{k+1} = \phi_{m,n}(x_k)$, $k \geq 0$ avec x_0 donné.

Grâce à l'ε-algorithme et à la règle de la croix on peut déterminer toute la table de Padé. En partant d'un point w voisin de x_* on construit les approximants [m/n] qui sont d'ordre m+n+1. Les coefficients de la série sont donnés par

$$c_i = (-1)^i \frac{F^{(i)}}{i!} (f(w)),\ i = 0, 1, 2,...$$

IV.3. - ESSAIS NUMERIQUES

1) *Procédures de Nourein et Claessens-Loisou-Wuytack*

Soit à résoudre

$$e^{-x}-x = 0$$

Donc

$f(x) = e^{-x}-x$, et g est définie par $g(t) = f(w+t)$

$$= e^{-w-t}-w-t = (e^{-w}-w) + (-e^{-w}-1)t + \frac{e^{-w}}{2!} t^2 + ... + (-1)^{n-w} \frac{e}{n!} t^n + ...$$

$$c_0 = e^{-w}-w$$

$$c_1 = - e^{-w}-1$$

$$c_2 = \frac{e^{-w}}{2!} \text{ et } c_{i+1} = - \frac{c_i}{(i+1)} \text{ pour } i \geq 2.$$

On a vu au paragraphe IV.1 que $F_p(w) = x_* + O(w-x_*)^{p+2}$.

Dans cet exemple on va s'intéresser au calcul de $F_p(w)$ pour $p = 1, 2,..., 10$.

$w = 1.$ $\qquad x_* = 0.567\ 143\ 290\ 409\ 783$

Procédure de Nourein

p	$F_p(w)$
1	.5649192899718807D+00
2	.5671105680984343D+00
3	.5671554363940288D+00
4	.5671445821480874D+00
5	.5671433139251416D+00
6	.5671432829887815D+00
7	.5671432895990245D+00
8	.5671432903945244D+00
9	.5671432904144171D+00
10	.5671432904102933D+00

Procédure de Claessens-Leisou-Wuytack

p	$F_p(w)$
1	.5649192899718807D+00
2	.5671105680984343D+00
3	.5671554363940288D+00
4	.5671445821480874D+00
5	.5671433139251416D+00
6	.5671432829887812D+00
7	.5671432895990245D+00
8	.5671432903945244D+00
9	.5671432904144174D+00
10	.5671432904102933D+00

2) *Utilisation réciproque*

Soit à résoudre :

$$\mathrm{LOG}(x) = 0 \quad \text{et} \quad x > 0 \quad (x_* = 1).$$

$$g(t) = e^{\tau - t} \qquad \text{où } \tau = \mathrm{LOG}(w)$$

$$= \sum_{i=0}^{\infty} (-1)^i \frac{e^{\tau}}{i!} t^i = \sum_{i=0}^{\infty} (-1)^i \frac{w}{i!} t^i$$

$$\Rightarrow c_i = (-1)^i \frac{e^{\tau}}{i!} = (-1)^i \frac{w}{i!}, \ i \geq 0.$$

En partant de $w = 2$ et en utilisant l'ε-algorithme on a calculé la moitié de la table de Padé et on a obtenu les résultats suivants :

n	[n/0] $\phi_{n,0}(w)$
0	.2000000000000000D+01
1	.6137056388801092D+00
2	.1094158652798311D+01
3	.9831504354686675D+00
4	.1002380693683924D+01
5	.9997199820546387D+00
6	.1000028052662506D+01
7	.9999975471948983D+00
8	.1000000190292256D+01
9	.9999999867305360D+00
10	.1000000000840359D+01
11	.9999999999512517D+00

n	[n/1] $\phi_{n,1}(w)$
1	.9705023393250572D+00
2	.1003984978231017D+01
3	.9995456240064893D+00
4	.1000044657245457D+01
5	.9999961486467277D+00
6	.1000000295716195D+01
7	.9999999795451650D+00
8	.1000000001287029D+01
9	.9999999999257374D+00
10	.1000000000000395 6D+01

n	[n/2] $\phi_{n,2}(w)$
2	.10002286006975850D+01
3	.99998228119824930D+00
4	.10000012653066050D+01
5	.99999991696966840D+00
6	.10000000050190510D+01
7	.99999999971939550D+00
8	.10000000000014573D+01
9	.99999999999992939D+00

n	[n/3] $\phi_{n,3}(w)$
3	.99999922301960230D+00
4	.10000000418791080D+01
5	.99999999784776180D+00
6	.10000000000104775D+01
7	.99999999999517750D+00
8	.10000000000000210D+01

n	[n/4] $\phi_{n,4}(w)$
4	.10000000014745440D+01
5	.99999999993920480D+00
6	.10000000000024300D+01
7	.99999999999999065D+00

n	[n/5] $\phi_{n,5}(w)$
5	.99999999999821590D+00
6	.10000000000000059D+01

IV.4. - UTILISATION DES APPROXIMANTS DE TYPE-PADE [28]

On utilise dans ce paragraphe les notations du paragraphe IV.2.

On définit une fonctionnelle c associée à g et agissant sur l'espace des polynômes par :

$$c(x^i) = c_i = (-1)^i \frac{F^{(i)}(\tau)}{i!}, \quad i = 0, 1, \ldots$$

L'idée de base est

$$c\left(\frac{1}{1-xt}\right) = c(1 + xt + x^2t^2 + \ldots) = c_0 + c_1 t + c_2 t^2 + \ldots$$
$$= g(t),$$

où c agit sur la variable x et où t est un paramètre.

Soit P un polynome qui est un approximant de la fonction $(1-xt)^{-1}$. La fonction g est approximée par c(P). On va s'intéresser à la valeur de c(P) en τ puisque $g(\tau) = x_*$.

Soit v un polynome arbitraire de degré k donné par

$$v(x) = a_0 + a_1 x + \ldots + a_k x^k \qquad (a_k \neq 0)$$

Considérons $\frac{v(x)-v(t)}{x-t}$. On a

$$\frac{v(x)-v(t)}{x-t} = a_1 + a_2(x+t) + \ldots + a_k(x^{k-1} + x^{k-2}t + \ldots + t^{k-1})$$

$$c(\frac{v(x)-v(t)}{x-t}) = a_1\, c_0 + a_2(c_1 + c_0 t) + \ldots + a_k(c_{k-1} + c_{k-2}t + \ldots + c_0 t^{k-1})$$

$$= \sum_{i=0}^{k-1} (\sum_{j=0}^{k-1-i} a_{1+i+j}\, c_j) t^i = w(t)$$

Soit $\tilde{w}(t) = t^{k-1}\, w(t^{-1}) = \sum_{i=0}^{k-1} (\sum_{j=0}^{k-1-i} a_{1+i+j}\, c_j) t^{k-i-1}$

$$= (\sum_{j=0}^{k-1} a_{1+j}\, c_j) t^{k-1} + (\sum_{j=0}^{k-2} a_{2+j}\, c_j) t^{k-2} + \ldots + a_k\, c_0.$$

Posons $A_0 = a_k\, c_0, \ldots, A_{k-2} = \sum_{j=0}^{k-2} a_{2+j}\, c_j,, A_{k-1} = \sum_{j=0}^{k-1} a_{1+j}\, c_j.$

On a :

$$\tilde{w}(t) = A_0 + A_1 t + \ldots + A_{k-1} t^{k-1}.$$

On a aussi

$$\tilde{w}(t) = t^{k-1} w(t^{-1}) = t^{k-1}\, c(\frac{v(x)-v(t^{-1})}{x-t^{-1}}) = c(\frac{t^k v(x) - t^k v(t^{-1})}{xt-1})$$

$$= - c(\frac{t^k v(x) - t^k v(t^{-1})}{1-xt}) = - t^k\, c(\frac{v(x)}{1-xt}) + t^k v(t^{-1})\, g(t)$$

Posons $\tilde{v}(t) = t^k v(t^{-1})$. Cela donne

$$\frac{\tilde{w}(t)}{\tilde{v}(t)} = g(t) - \frac{t^k}{\tilde{v}(t)}\, c(\frac{v(x)}{1-xt}).$$

Le polynome $\tilde{v}(t)$ s'écrit aussi comme

$$\tilde{v}(t) = a_k + a_{k-1}t + \ldots + a_0 t^k.$$

Considérons maintenant la fraction ϕ_k définie par :

$$\phi_k(t) = \frac{\tilde{w}(t)}{\tilde{v}(t)}$$
$$= \frac{A_0 + A_1 t + \ldots + A_{k-1}t^{k-1}}{a_k + a_{k-1}t + \ldots + a_0 t^k}.$$

$$\phi_k(t) - g(t) = - t^k\, c(\frac{v(x)}{\tilde{v}(t)(1-xt)}).$$

$$\phi_k(\tau) - g(\tau) = \phi_k(\tau) - x_* = - \tau^k\, c(\frac{v(x)}{\tilde{v}(\tau)(1-x\tau)})$$

$$\phi_k(f(w)) - x_* = - [f(w)]^k\, c(\frac{v(x)}{\tilde{v}(f(w))(1-xf(w))}) \text{ avec } \tau = f(w).$$

$$= O(w-x_*)^k.$$

Soit $w' = \phi_k(f(w))$. w' est donc une approximation d'ordre k de x_*. D'où l'idée d'itérer sur la fonction ϕ_k.

Soit x_0 un point donné. L'itéré x_n $(n \geq 1)$ est construit de la manière suivante :

$$x_n = \phi_k(f(x_{n-1})) = \frac{A_0 + A_1 f_{n-1} + \ldots + A_{k-1} f_{n-1}^{k-1}}{a_k + a_{k-1} f_{n-1} + \ldots + a_0 f_{n-1}^k}$$

où $f_{n-1} = f(x_{n-1})$.

ϕ_k est d'ordre k.

Exemples de fonction ϕ_k

a) $k = 1$: ϕ_1 est d'ordre un. Elle est donnée par :

$$\phi_1(x) = \frac{a_1 x}{a_1 + a_0 f(x)}.$$

b) $k = 2$: ϕ_2 est d'ordre deux. Elle est donnée par :

$$\phi_2(x) = \frac{a_2 x + (a_1 x + a_2 c_1) f(x)}{a_2 + a_1 f(x) + a_0 f^2(x)} \quad \text{avec } c_1 = -\frac{1}{f'(x)}$$

$$= x - f(x) \frac{a_0 f(x) - a_2 c_1}{a_2 + a_1 f(x) + a_0 f^2(x)}$$

$$= x - \frac{f(x)}{f'(x)} \frac{(a_2 + a_0 f(x) f'(x))}{a_2 + a_1 f(x) + a_0 f^2(x)}.$$

On remarque que le choix $a_0 = 0$, $a_1 = 0$ et a_2 quelconque ($\neq 0$) correspond à la méthode de Newton.

c) $k = 3$: ϕ_3 est d'ordre trois. Elle est donnée par :

$$\phi_3(x) = \frac{a_3 c_0 + (a_2 c_0 + a_3 c_1) f(x) + (a_1 c_0 + a_2 c_1 + a_3 c_2)\, f^2(x)}{a_3 + a_2\, f(x) + a_1\, f^2(x) + a_0\, f^3(x)}$$

$$= \frac{[a_3 + a_2 f(x) + a_1 f^2(x) + a_0 f^3(x)]x + f(x)[a_3 c_1 + a_2 c_1 + a_3 c_2 f(x) - a_0 f^2(x)]}{a_3 + a_2 f(x) + a_1\, f^2(x) + a_0\, f^3(x)}$$

$$= x - f(x) \frac{a_0 f^2(x) - a_3 c_1 - a_2 c_1 - a_3 c_2 f(x)}{a_3 + a_2 f(x) + a_1 f^2(x) + a_0 f^3(x)}.$$

Or $c_1 = -\frac{1}{f'(x)}$ et $c_2 = -\frac{f''(x)}{2f'(x)}$. Par suite on a

$$\phi_3(x) = x - f(x) \frac{2a_0 f^2(x) f'^2(x) + 2(a_2 + a_3) f'(x) + a_3 f(x) f''(x)}{2f'^2(x)(a_3 + a_2 f(x) + a_1 f^2(x) + a_0 f^3(x))}.$$

On remarque que le choix $a_0 = 0$, $a_1 = 0$, $a_2 = 0$ et a_3 quelconque ($\neq 0$) correspond à

$$\phi_3(x) = x - \frac{f(x)}{f'(x)} - \frac{f''(x)}{2f'(x)} \frac{f(x)}{f'(x)}.$$

Approximants de type-Padé

$$\phi_2(x) = x - \frac{f(x)}{f'(x)} \frac{a_2 + a_0 x\, f(x)\, f'(x)}{a_2 + a_1 f(x) + a_0 [f(x)]^2}$$

$$x_{n+1} = \phi_2(x_n), \; n = 0, 1, \ldots$$

a) $a_0 = a_1 = a_2 = 1$

n	x_n
0	.1000000000000000D+01
1	.5758749581081601D+00
2	.5671429689086451D+00
3	.5671432904097832D+00
4	.5671432904097839D+00
5	.5671432904097839D+00
6	.5671432904097839D+00
7	.5671432904097839D+00
8	.5671432904097839D+00
9	.5671432904097839D+00

b) $a_0 = 1{,}5$; $a_1 = 2$; $a_2 = 1{,}7$

n	x_n
0	.1000000000000000D+01
1	.6113588657411457D+00
2	.5679378165406811D+00
3	.5671435639050669D+00
4	.5671432904098164D+00
5	.5671432904097839D+00
6	.5671432904097839D+00
7	.5671432904097839D+00
8	.5671432904097839D+00
9	.5671432904097839D+00

c) $a_0 = 20$; $a_1 = 3{,}5$; $a_2 = 1{,}5$

n	x_n
0	.1000000000000000D+01
1	.2579681412493455D+00
2	.7161262680170734D-01
3	.4220077197158868D-01
4	.4173330707025799D-01
5	.4172915035303854D-01
6	.4172911379666936D-01
7	.4172911347520494D-01
8	.4172911347237810D-01
9	.4172911347235324D-01

On remarque que ce dernier choix des coefficients $a_i (i = 0, 1, 2)$ ne donne pas la bonne solution.

CHAPITRE V

PROCEDES D'ACCELERATION DE LA CONVERGENCE ASSOCIES AUX METHODES DE

- KING
- SCHRODER

INTRODUCTION

A partir de certaines méthodes de point fixe nous allons maintenant construire des procédés d'accélération de la convergence.

Nous nous intéressons essentiellement aux méthodes de King [16] et de Schröder [4].

Nous avons vu au chapitre II que la méthode de King est d'ordre quatre et que celle de Schröder est d'ordre trois. A l'aide des théories de l'interpolation polynômiale et des différences divisées d'une fonction, nous allons donner deux procédés d'ordre quatre pour la première et d'ordre trois pour la seconde.

V.1. - POSITION DE LA QUESTION

Soit à résoudre une équation non linéaire

$$f(x) = 0.$$

Soit ϕ_r une méthode itérative basée sur p points et faisant intervenir dans son expression les dérivées de f jusqu'à l'ordre q :

$$x_n = \phi_r(x_{n-1},\dots, x_{n-p}, f, f',\dots, f^{(q)}),$$

r étant l'ordre de convergence de ϕ_r on a :

$$x_n-x_* = O(x_{n-1}-x_*)^r,$$

où x_* est une racine de f.

Etant donnée une suite $(S_n)_{n\in\mathbb{N}}$, de limite S_*, on va approcher les dérivées de f par des quantités formées à l'aide d'un nombre fini de termes de (S_n). Soit $[f^{(j)}]$ une telle approximation pour $j = 0, 1,\dots, q$.

On considère la transformation de suite $(S_n) \to (T_n)$ avec

$$T_n = \phi_r(S_n,\dots, S_{n-p+1}, [f],\dots, [f^{(q)}])$$

Le problème est de choisir les approximations $[f^{(j)}]$ afin d'obtenir un proécédé d'accélération de la convergence de degré r, c'est-à-dire tel que :

$$T_n-S_* = O(S_n-S_*)^r.$$

Dans les deux paragraphes suivants nous tentons de répondre d'une certaine manière à cette question pour les méthodes de King et de Schröder.

V.2. - PROCEDE ASSOCIE A LA METHODE DE KING [16]

Soit une équation de la forme :

$$x = F(x).$$

En prenant $f(x) = x - F(x)$, on peut mettre la méthode de King, vue au chapitre II, paragraphe II.3.1, sous la forme :

$$\overline{x_{n+3}} = \phi_4(x_n, x_{n+1}, x_{n+2}, \overline{x_{n+2}}, f).$$

On sait que $\overline{x_{n+3}}-x_* = O(x_n-x_*)^4$.

Soit une suite $(S_n)_{n\in\mathbb{N}}$ à convergence linéaire. On sait que pour de telles suites il existe une fonction F et il existe un entier N tels que :

$$S_{n+1} = F(S_n),\ \forall n \geq N \qquad [11]$$

Donc à partir de S_n, S_{n+1} et S_{n+2}, il est aisé de construire $\overline{S_{n+2}}$:

La difficulté consiste dans le calcul de $F(\overline{S_{n+2}})$, car F n'est pas connue d'une façon explicite.

Une idée simple pour éviter cette difficulté est d'interpoler F en S_n, S_{n+1} et S_{n+2} puis d'approcher $F(\overline{S_{n+2}})$ par la valeur de son interpolant au point $\overline{S_{n+2}}$.

Considérons les deux points (S_n, S_{n+1}) et (S_{n+1}, S_{n+2}). Soit $P_1^{(n)}$ le polynôme passant par ces deux points. On a :

$$P_1^{(n)}(\overline{S_{n+2}}) = \overline{S_{n+2}}.$$

$\overline{S_{n+2}}$ est un point fixe de $P_1^{(n)}$.

Pour avoir une meilleure approximation de $F(\overline{S_{n+2}})$, nous devons faire appel à d'autres termes de la suite (S_n).

Pour cela prenons les points (S_n, S_{n+1}), (S_{n+1}, S_{n+2}) et (S_{n+2}, S_{n+3}).

Dans ce cas, le polynôme d'interpolation est de degré deux.

Soit $P_2^{(n)}$ ce polynôme.

$P_2^{(n)}(S_{n+i}) = S_{n+1+i}$; $i = 0, 1, 2.$

$P_0^{(i)} = S_{n+1+i}$; $i = 0, 1, 2.$

$$x = \overline{S_{n+2}} = \frac{S_{n+2}\, S_n - S_{n+1}^2}{S_{n+2} - 2S_{n+1} + S_n}$$

$$P_j^{(i)}(x) = \frac{(S_{n+i+j} - x)P_{j-1}^{(i)}(x) - (S_{n+i} - x)P_{j-1}^{(i+1)}(x)}{S_{n+i+j} - S_{n+i}}$$

pour $j = 1, 2$ et $i = 0, 1, \ldots, 2-j$.

$P_2^{(n)} = P_2^{(n)}(\overline{S_{n+2}}) \simeq F(\overline{S_{n+2}})$.

L'erreur d'interpolation nous permet d'écrire :

$$F(\overline{S_{n+2}}) - P_2^{(n)} = \frac{F^{(3)}(\xi)}{6}(S_n - \overline{S_{n+2}})(S_{n+1} - \overline{S_{n+2}})(S_{n+2} - \overline{S_{n+2}})$$

$$= O(e_n^3), \quad \text{avec } \xi \in \left]\min_i\{S_{n+i}, \overline{S_{n+2}}\}, \max_i\{S_{n+i}, \overline{S_{n+2}}\}\right[$$

Posons $K_1^1 = \dfrac{\Delta S_{n+1}}{\Delta S_n} = K_1$

$$K_*^* = \frac{P_2^{(n)} - S_{n+2}}{\overline{S_{n+2}} - S_{n+1}} = \frac{F(\overline{S_{n+2}}) - S_{n+2} + O(e_n^3)}{\overline{S_{n+2}} - S_{n+1}}$$

$$= K_* + O(e_n^2).$$

Or lors de l'étude de la méthode de King au chapitre II, nous avons vu que

$$\hat{K} = K_*(1 - K_* + K_1) = K + O(e_n^2).$$

$P_3^{(n)}$ est déterminé par les conditions d'interpolation

$P_3^{(n)}(S_{n+i}) = S_{n+i+1}$; $i = 0, 1, 2, 3.$

Le calcul de $P_3^{(n)} = P_3^{(n)}\ (\overline{S_{n+2}})$ se fait comme précédemment par le schéma de Neville-Aitken.

Dans ce cas on a

$$F(\overline{S_{n+2}}) - P_3^{(n)} = O(e_n^4)$$

Considérons le procédé

$$T_n = \overline{S_{n+2}} - \frac{1}{1-\tilde{K}}\,(\overline{S_{n+2}}-P_3^{(n)}). \tag{V.2.K'}$$

Cette fois-ci on a

$$T_n - S_* = O(S_n - S_*)^4.$$

En fait toute approximation $Q^{(n)}$ de $F(\overline{S_{n+2}})$ d'ordre $\beta \geq 4$ conduit à la même conclusion. A savoir

$$T_n - S_* = O(S_n - S_*)^4.$$

En effet

$$\left.\begin{aligned} T_n - S_* &= \overline{e_{n+2}} - \frac{1}{1-\tilde{K}}\,(\overline{S_{n+2}}-Q^{(n)}) \\ F(\overline{S_{n+2}}) - Q^{(n)} &= O(e_n^{\beta}) \\ &\quad \beta \geq 4 \end{aligned}\right\} \Longrightarrow \ldots$$

$$\ldots\ T_n - S_* = \underbrace{\overline{e_{n+2}} - \frac{1}{1-\tilde{K}}\,(\overline{S_{n+2}} - F(\overline{S_{n+2}}))}_{\downarrow \atop O(e_n^4)} - \underbrace{\frac{1}{1-\tilde{K}}\,(F(\overline{S_{n+2}}) - Q^{(n)})}_{\downarrow \atop O(e_n^{\beta})}$$

$$= O(e_n^4).$$

Définissons la quantité $\tilde{K}$ par

$$\tilde{K} = K^*_*(1 - K^*_* + K^1_1)$$

$$= (K_* + O(e_n^2))(1 - K_* + K_1 + O(e_n^2))$$

$$\Rightarrow \tilde{K} = K + O(e_n^2).$$

On pose

$$T_n = \overline{S_{n+2}} - \frac{1}{1-\tilde{K}} (\overline{S_{n+2}} - P_2^{(n)}) \qquad \text{(V.2.K)}$$

Ainsi la suite (S_n) est transformée en (T_n). Nous obtenons

$$T_n - S_* = O(S_n - S_*)^3,$$

alors que la méthode de King est d'ordre quatre.

En effet nous avons :

$$\left.\begin{array}{l} T_n - S_* = (\overline{S_{n+2}} - S_*) - \dfrac{1}{1-\tilde{K}} (\overline{S_{n+2}} - P_2^{(n)}) \\ \overline{e_{n+2}} = \overline{S_{n+2}} - S_* \end{array}\right\} \Longrightarrow \ldots$$

$$\ldots\ T_n - S_* = \underbrace{\overline{e_{n+2}} - \frac{1}{1-\tilde{K}} (\overline{S_{n+2}} - F(\overline{S_{n+2}}))}_{\downarrow \atop O(e_n^4)} - \underbrace{\frac{1}{1-\tilde{K}} (F(\overline{S_{n+2}}) - P_2^{(n)})}_{\downarrow \atop O(e_n^3)}$$

D'où le résultat.

On voit qu'une approximation de $F(S_{n+2})$ d'ordre trois ne suffit pas pour obtenir l'ordre quatre. On s'intéresse donc maintenant à un polynôme d'interpolation de degré trois. Soit

$$P_3^{(n)}(x) = a_0^{(n)} + a_1^{(n)}\, x + a_2^{(n)}\, x^2 + a_3^{(n)}\, x^3.$$

Remarque :

A partir des termes $\{S_n, S_{n+1}, S_{n+2}, S_{n+3}, S_{n+4}\}$, les procédés d'Overholt et de Germain-Bonne donnent une quantité $T_3^{(n)}$ telle que :

$$T_3^{(n)}-S_* = O(S_n-S_*)^4.$$

Par conséquent le procédé que nous venons d'étudier leur est comparable.

V.3. - PROCEDES ASSOCIES A LA METHODE DE SCHRODER

Nous avons étudié en détail au chapitre II, la méthode de Schröder. Elle peut être mise sous la forme

$$x_{n+1} = \phi_3(x_n, f, f', f'').$$

On a

$$x_{n+1}-x_* = O(x_n-x_*)^3.$$

Dans ce paragraphe nous nous proposons de former à l'aide des termes d'une suite (S_n) de limite S_*, des quantités approchant $f(x_n)$, $f'(x_n)$ et $f''(x_n)$, de telle sorte que l'on ait

$$T_n-S_* = O(S_n-S_*)^3,$$

où T_n est donné par :

$$T_n = \phi_3(S_n, [f], [f'], [f'']),$$

avec $[f]$ approximation de $f(S_n)$, $[f']$ approximation de $f'(S_n)$ et $[f'']$ approximation de $f''(S_n)$.

V.3.1. - Procédé d'ordre deux associé à la méthode de Schröder

Soit $(S_n)_{n\in\mathbb{N}}$ une suite à convergence linéaire.

Il existe F et il existe N tels que

$$S_{n+1} = F(S_n), \ \forall n \geq N$$

Soit f définie par

$$f(x) = F(x) - x$$

Pour $n \geq N$ on a

$$\Delta S_n = f(S_n),$$
$$\text{et } f(S_*) = 0.$$

Donc nous prenons tout simplement pour [f] la quantité ΔS_n

$$[f] = \Delta S_n = f(S_n).$$

Quant à $f'(S_n)$ elle sera approchée par la différence divisée de f aux points S_n et S_{n+1}.

$$[f'] = [S_n, S_{n+1}, f] = \frac{f(S_{n+1})-f(S_n)}{S_{n+1} - S_n}.$$

De même $f''(S_n)$ est approchée par la différence divisée de f aux points S_n, S_{n+1}, S_{n+2}.

$$[f''] = 2[S_n, S_{n+1}, S_{n+2}, f] = 2\,\frac{[S_{n+1}, S_{n+2}, f] - [S_n, S_{n+1}, f]}{S_{n+2} - S_n}.$$

Posons $K_n = \dfrac{\Delta S_{n+1}}{\Delta S_n}$.

$$[f'] = \frac{f(S_{n+1})-f(S_n)}{S_{n+1} - S_n} = \frac{\Delta S_{n+1} - \Delta S_n}{\Delta S_n} = K_n - 1$$

$$[f''] = 2\,\frac{[S_{n+1}, S_{n+2}, f] - [S_n, S_{n+1}, f]}{S_{n+2} - S_n} = 2\,\frac{(K_{n+1}-1)-(K_n-1)}{\Delta S_{n+1} + \Delta S_n}$$

$$= 2\,\frac{K_{n+1}-K_n}{\Delta S_n (K_n+1)} = \frac{2}{\Delta S_n}\;\frac{K_{n+1}-K_n}{K_n+1}.$$

La quantité T_n est donnée par :

$$T_n = \phi_3(S_n, [f], [f'], [f''])$$

$$= S_n - \frac{2[f]\,[f']}{2[f']^2-[f][f'']}$$

$$= S_n - \frac{2\Delta\, S_n\, (K_n - 1)}{2(K_n-1)^2 - 2(\frac{K_{n+1}-K_n}{K_n + 1})}$$

D'où

$$\begin{cases} T_n = S_n - \dfrac{\Delta S_n (K_n - 1)}{(K_n-1)^2 - \dfrac{\Delta K_n}{K_n+1}}, \quad n \geq 0 \\[2ex] K_n = \dfrac{\Delta S_{n+1}}{\Delta S_n}. \end{cases} \tag{V.3.S}$$

Ce procédé n'est que de degré deux. En effet

$$f_n = f(S_n) = c_1\, e_n + c_2\, e_n^2 + c_3\, e_n^3 + \ldots$$

où $e_n = S_n - S_*$.

Notons, $e_{n+1} = S_{n+1}-S_*$ et $e_{n+2} = S_{n+2}-S_*$.

$$e_{n+1} = (c_1+1)\, e_n + c_2\, e_n^2 + c_3\, e_n^3 + \ldots$$

$$e_{n+2} = (c_1+1)^2\, e_n + c_2(c_1+1)(c_1+2)\, e_n^2 + \ldots$$

$$K_n = \frac{\Delta S_{n+1}}{\Delta S_n} = \frac{\Delta e_{n+1}}{\Delta e_n} = (c_1+1) + c_2(c_1+2)\, e_n + \ldots$$

$$K_{n+1} = \frac{\Delta S_{n+2}}{\Delta S_{n+1}} = \frac{\Delta e_{n+2}}{\Delta e_{n+1}} = (c_1+1) + c_2(c_1+2)\, e_{n+1} + \ldots$$

$$= (c_1+1) + c_2(c_1+1)(c_1+2)\, e_n + \dots$$

Posons $N_n = \Delta S_n(K_n-1)$

et $D_n = (K_n-1)^2 - \dfrac{\Delta K_n}{K_n+1}$

Nous avons :

$$N_n = c_1^2\, e_n + c_1\, c_2(c_1+3)\, e_n^2 + \dots,$$

$$\text{et } D_n = c_1^2 + c_1\, c_2(2c_1 + 3)\, e_n + \dots$$

Par suite

$$\frac{N_n}{D_n} = e_n - c_2\, e_n^2 + \dots$$

Par conséquent nous obtenons :

$$T_n - S_* = e_n - \frac{N_n}{D_n} = c_2\, e_n^2 + \dots$$

Ce procédé, mettant en jeu S_n, S_{n+1}, S_{n+2} et S_{n+3} ne nous fournit que des quantités d'ordre deux. Or on sait que les procédés d'Overholt et de Germain-Bonne, qui utilisent quatre termes de la suite, donnent des quantités d'ordre trois.

Cette perte de l'ordre est dûe au fait que les deux développements en série suivants ne coïncident pas en général :

$$[S_n, S_{n+1}, \dots, S_{n+i}, f] = \frac{f^{(i)}(S_n)}{i!} + a\, e_n + \dots = \frac{f_*^{(i)}}{i!} + b\, e_n + \dots$$

$$f^{(i)}(S_n) = f_*^{(i)} + f_*^{(i+1)}\, e_n + \dots$$

$$b\, i! \neq f_*^{(i+1)} \qquad \text{(en général).}$$

On ne veut pas utiliser davantage de termes de la suite. D'où l'idée de construire une nouvelle approximation de $f'(S_n)$ grâce aux termes de la suite utilisé déjà dans le procédé ci-dessus.

C'est l'objet du paragraphe suivant.

V.3.2. - Procédé d'ordre trois associé à la méthode de Schröder

On remarque que

$$[S_n, S_{n+1}, f] = c_1 + (2c_2 + c_1 c_2)\, e_n + \ldots$$

$$f'(S_n) = c_1 + 2c_2\, e_n + \ldots$$

$$\left.\begin{array}{r} \Delta S_n = c_1 e_n + c_2 e_n^2 + \ldots \\ [S_n, S_{n+1}, S_{n+2}, f] = c_2 + O(e_n) \end{array}\right\} \Rightarrow \Delta S_n [S_n, S_{n+1}, S_{n+2}, f] = c_1 c_2 e_n + O(e_n^2).$$

Par conséquent on a

$$[S_n, S_{n+1}, f] - \Delta S_n [S_n, S_{n+1}, S_{n+2}, f] = c_1 + 2c_2\, e_n + O(e_n^2).$$

Soit X_n défini par :

$$X_n = K_n - 1 - \frac{\Delta K_n}{K_n + 1} = c_1 + 2c_2\, e_n + O(e_n^2).$$

La quantité X_n coïncide avec $f'(S_n)$ jusqu'à l'ordre un compris.

Dans la méthode de Schröder remplaçons $f(S_n)$ par ΔS_n, $f'(S_n)$ par X_n et $f''(S_n)$ par $2[S_n, S_{n+1}, S_{n+2}, f]$, on obtient le procédé :

$$\left\{\begin{array}{l} T_n = S_n - \dfrac{\Delta S_n \left\{K_n - 1 - \dfrac{\Delta K_n}{K_n + 1}\right\}}{\left\{K_n - 1 - \dfrac{\Delta K_n}{K_n + 1}\right\}^2 - \dfrac{\Delta K_n}{K_n + 1}} \\ \\ K_n = \dfrac{\Delta S_{n+1}}{\Delta S_n} \end{array}\right. \qquad (V.3.S')$$

Cette fois-ci on a :

$$T_n - S_* = O(S_n - S_*)^3.$$

En effet, en posant $N_n = \Delta S_n \left\{ K_n - 1 - \frac{\Delta K_n}{K_n+1} \right\}$

et $D_n = \left\{ K_n - 1 - \frac{\Delta K_n}{K_n+1} \right\}^2 - \frac{\Delta K_n}{K_n+1}$, nous avons les développements

$$N_n = \{c_1^2 + 3c_1c_2e_n + O(e_n^2)\}e_n.$$

$$\text{et } D_n = \{c_1^2 + 3c_1\, c_2\, e_n + O(e_n^2)\}.$$

D'où

$$\frac{N_n}{D_n} = \frac{\{c_1^2 + 3c_1c_2e_n + O(e_n^2)\}e_n}{c_1^2 + 3c_1C_2e_n + O(e_n^2)} = O(e_n^3) + e_n.$$

Par conséquent nous obtenons :

$$T_n - S_* = S_n - S_* + \frac{N_n}{D_n} = e_n - \frac{N_n}{D_n} = O(e_n^3).$$

CONCLUSION

Ces deux derniers paragraphes montrent bien que toute méthode de point fixe peut donner naissance à un procédé d'accélération de la convergence. La seule difficulté réside dans le choix des approximations convenables afin d'obtenir un degré d'accélération égal à l'ordre de convergence de la méthode de point fixe.

$x_{n+1} = e^{-x_n}$, $x_0 = 1$

```
.1000000000000000D+01
.3678794411714423D+00
.6922006275553464D+00
.5004735005636368D+00
.6062435350855974D+00
.5453957859750270D+00
.5796123355033789D+00
.5601154613610891D+00
.5711431150801770D+00
.5648793473910495D+00
.5684287250290607D+00
.5664147331468833D+00
.5675566373282835D+00
```

Δ^2-d'Aidken

```
.5822260969956230D+00
.5717057675272521D+00
.5686388058644661D+00
.5676169948466354D+00
.5672967524886339D+00
.5671924278872064D+00
.5671591338340007D+00
.5671483792269584D+00
.5671449285298513D+00
```

(V.3.S)

```
.6095773746848443D+00
.5797914333724451D+00
.5712839863062001D+00
.5684524511514806D+00
.5675674724359099D+00
.5672790620823158D+00
.5671870722760146D+00
.5671573515978921D+00
.5671478169837048D+00
```

(V.3.S')

```
.5703518941051280D+00
.5667847973690096D+00
.5672266373779075D+00
.5671301608208084D+00
.5671458848652666D+00
.5671428385464257D+00
.5671433750062943D+00
.5671432752044506D+00
.5671432932068919D+00
```

(V.2.K)

```
.5687396202387979D+00
.5668230647093972D+00
.5671976079659024D+00
.5671330229068560D+00
.5671451225988394D+00
.5671429521730719D+00
.5671433516846631D+00
.5671432791884351D+00
.5671432924522726D+00
```

(V.2.K')

```
.5682811983840369D+00
.5669430423373699D+00
.5671799354233819D+00
.5671366496364626D+00
.5671445043828668D+00
.5671430693304666D+00
.5671433307718020D+00
.5671432830505661D+00
.5671432917526326D+00
```

CHAPITRE VI

RESOLUTION DE SYSTEMES D'EQUATIONS NON LINEAIRES

INTRODUCTION

Dans ce chapitre nous nous intéressons à la résolution de systèmes d'équations non linéaires.

Après avoir rappelé la méthode d'Henrici [8] nous en proposons une autre basée sur le E*-algorithme [6].*

VI.1. - METHODE D'HENRICI [8]

Soit F une application de $\mathbb{R}^k$ dans $\mathbb{R}^k$ telle qu'il existe $x_* \in \mathbb{R}^k$ tel que

$$x^* = F(x^*).$$

Soit f l'application définie par

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

x^* est donc une racine de f. Supposons de plus $f'(x^*)$ inversible. Supposons que f admette un inverse $\mathcal{F}$. Soit $x^{(0)}$ un point donné. Supposons $x^{(n)}$ construit. L'itéré $x^{(n+1)}$ sera obtenu de la manière suivante :

$$\begin{cases} u^{(0)} = x^{(n)} \\ u^{(i+1)} = F(u^{(i)}),\ i = 0, 1, \ldots, k \\ U_0 = (\Delta u^{(0)}, \ldots, \Delta u^{(k-1)}) \text{ matrice } k \times k \\ U_1 = (\Delta u^{(1)}, \ldots, \Delta u^{(k)}) \quad \text{matrice } k \times k \\ \Delta U_0 = U_1 - U_0 \quad\quad \text{matrice } k \times k \\ x^{(n+1)} = u^{(0)} - U_0 (\Delta U_0)^{-1}\, \Delta u^{(0)}. \end{cases}$$

Cette méthode peut être construite de la façon suivante :

$$u^{(i+1)} = F(u^{(i)}) = u^{(i)} + f(u^{(i)}) \Longleftrightarrow \Delta u^{(i)} = f(u^{(i)})$$

$$\Longleftrightarrow u^{(i)} = \mathcal{F}(\Delta u^{(i)}).$$

Considérons maintenant le développement de Taylor de $\mathcal{F}$ dans un voisinage de zéro. On a

$$\begin{aligned} u^{(i)} = \mathcal{F}(\Delta u^{(i)}) &= \mathcal{F}(0) + \mathcal{F}'(0)\, \Delta u^{(i)} + O(||\Delta u^{(i)}||^2) \\ &= x^* + A\, \Delta u^{(i)} + O(||\Delta u^{(i)}||^2) \end{aligned}$$

où $A = \mathcal{F}'(0)$ et où $O(||\Delta u^{(i)}||^2)$ désigne un vecteur de $\mathbb{R}^k$ tel qu'il existe une constante positive K finie telle que

$$O(||\Delta u^{(i)}||^2) \leq K\, ||\Delta u^{(i)}||^2.$$

Il s'agit dans tout ce chapitre de la norme Euclidienne.

Dans ce développement négligeons les termes du second ordre. Cela donne

$$u^{(i)} = x^* + A\, \Delta u^{(i)} \quad \text{pour } i = 0, 1, \ldots, k.$$

$$\Rightarrow \Delta u^{(i)} = A\, \Delta^2 u^{(i)} \quad \text{pour } i = 0, 1, \ldots, k-1.$$

$$\Rightarrow (\Delta u^{(0)}, \ldots, \Delta u^{(k-1)}) = A\, (\Delta^2 u^{(0)}, \ldots, \Delta^2 u^{(k-1)})$$

$$\Rightarrow \quad U_0 = A\, \Delta U_0.$$

Si la matrice ΔU_0 est inversible, alors on a

$$A = U_0 (\Delta U_0)^{-1}.$$

Donc la matrice $U_0(\Delta U_0)^{-1}$ ainsi obtenue peut être considérée comme une approximation du Jacobien $\mathcal{F}'$ calculé au point zéro. Or x^* est donné par

$$\begin{aligned} x^* &= u^{(0)} - A\, \Delta u^{(0)} \\ &= u^{(0)} - U_0(\Delta U_0)^{-1}\, \Delta u^{(0)}. \end{aligned}$$

D'où l'idée de prendre

$$x^{(n+1)} = u^{(0)} - U_0(\Delta U_0)^{-1} \Delta u^{(0)}.$$

VI.2. - METHODE BASEE SUR LE E-ALGORITHME

Nous avons vu au chapitre III, que le E-algorithme est basé sur l'élimination de Gauss qui est également utilisée pour inverser une matrice. Donc nous allons linéariser le système puis appliquer le E-algorithme au système ainsi obtenu.

Reprenons toutes les notations du paragraphe VI.1.

Pour $\ell \in \{1, 2,..., k\}$, considérons le système

$$\begin{cases} u_\ell^{(0)} = x_\ell^* + \sum_{j=1}^{k} \Delta u_j^{(0)} A_\ell^{(j)} + O(||\Delta u^{(0)}||^2)_\ell \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ u_\ell^{(k)} = x_\ell^* + \sum_{j=1}^{k} \Delta u_j^{(k)} A_\ell^{(j)} + O(||\Delta u^{(k)}||^2)_\ell \end{cases}$$

où $\Delta u_j^{(i)} = u_j^{(i+1)} - u_j^{(i)}$ et $(A_\ell^{(1)},..., A_\ell^{(k)})$ désigne la $\ell^{\text{ième}}$ ligne de la matrice A.

Si on néglige le terme $O(||\Delta u^{(i)}||^2)_\ell$ alors, en appliquant le E-algorithme avec le choix $g_j(i) = \Delta u_j^{(i)} = u_j^{(i+1)} - u_j^{(i)}$, on a :

$$x_\ell^* = E_k^{(0)} (u_\ell^{(0)}).$$

Par conséquent x^* est donné par

$$x^* = \begin{bmatrix} x_1^* \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ x_k^* \end{bmatrix} = \begin{bmatrix} E_k^{(0)} (u_1^{(0)}) \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ E_k^{(0)} (u_k^{(0)}) \end{bmatrix}$$

Posons

$$E_j^{(i)} = \begin{bmatrix} E_j^{(i)}\ (u_1^{(i)}) \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ E_j^{(i)}\ (u_k^{(i)}) \end{bmatrix} \quad \text{pour } j = 1, 2,\dots k \text{ et } i = 0, 1,\dots, k-j$$

L'algorithme proposé est :

$x^{(0)}$ donné.

$(n+1)^{\text{ième}}$ itération

$$\left[\begin{array}{ll}
u^{(0)} = x^{(n)} & \\
u^{(i+1)} = F(u^{(i)}),\ i = 0, 1,\dots, k. & \\
E_0^{(i)} = u^{(i)},\ i = 0, 1,\dots, k & \\
g_{0,j}^{(i)} = \Delta u_j^{(i)} = u_j^{(i+1)} - u_j^{(i)},\ j = 1,\dots, k \text{ et } i = 0, 1,\dots, k. & \\
E_j^{(i)} = \dfrac{g_{j-1,j}^{(i+1)}\ E_{j-1}^{(i)} - g_{j-1,j}^{(i)}\ E_{j-1}^{(i+1)}}{g_{j-1,j}^{(i+1)} - g_{j-1,j}^{(i)}}, & \begin{array}{l} j = 1,\dots, k \\ i = 0,\dots, k-j \end{array} \\
g_{j,\ell}^{(i)} = \dfrac{g_{j-1,j}^{(i+1)}\ g_{j-1,\ell}^{(i)} - g_{j-1,j}^{(i)}\ g_{j-1,\ell}^{(i+1)}}{g_{j-1,j}^{(i+1)} - g_{j-1,j}^{(i)}}, & \begin{array}{l} j = 1,\dots, k-1 \\ i = 0, 1,\dots, k+1-j \\ \ell = j+1,\dots, k \end{array} \\
x^{(n+1)} = E_k^{(0)} &
\end{array}\right.$$

<u>Remarque</u> :

$E_j^{(i)}$ est un vecteur de $\mathbb{R}^k$ et $g_{j,\ell}^{(i)}$ est un élément de $\mathbb{R}$.

On a les deux propriétés suivantes :

Propriété 1 :

Si $f(x) = Ax-b$ *où* A *est une matrice carrée inversible de dimension* k *et* $b \in \mathbb{R}^k$ *alors*

$$x^* = E_k^{(0)} = A^{-1}b$$

Preuve :

Soit $x^{(0)}$ donné.

$$u^{(i+1)} = u^{(i)} + Au^{(i)} - b, \quad i = 0, 1, \ldots, k$$
$$\Rightarrow \Delta u^{(i)} = Au^{(i)} - b \quad , i = 0, 1, \ldots, k$$
$$\Rightarrow A^{-1} \Delta u^{(i)} = u^{(i)} - A^{-1} b, \quad i = 0, 1, \ldots, k$$

$$\Rightarrow u^{(i)} = A^{-1} b + A^{-1} \Delta u^{(i)}, \quad i = 0, 1, \ldots, k$$

En appliquant le E-algorithme à chaque composante de $u^{(i)}$, on obtient $x_\ell^* = E_k^{(0)}(u_\ell^{(0)})$ pour $\ell \in \{1, 2, \ldots, k\}$. Puisque $E_k^{(0)}(u_\ell^{(0)})$ est la $\ell^{\text{ième}}$ composante de $E_k^{(0)}$ on a

$$x^* = E_k^{(0)}.$$

Propriété 2 :

Le vecteur $E_j^{(i)}$ *est donné par un rapport de deux déterminants*

$$E_j^{(i)} = \begin{vmatrix} u^{(i)} & \ldots\ldots & u^{(i+j)} \\ \Delta u_1^{(i)} & \ldots\ldots & \Delta u_1^{(i+j)} \\ \ldots & \ldots\ldots & \ldots \\ \Delta u_j^{(i)} & \ldots\ldots & \Delta u_j^{(i+j)} \end{vmatrix} \Bigg/ \begin{vmatrix} 1 & \ldots\ldots & 1 \\ \Delta u_1^{(i)} & \ldots\ldots & \Delta u_1^{(i+j)} \\ \ldots & \ldots\ldots & \ldots \\ \Delta u_j^{(i)} & \ldots\ldots & \Delta u_j^{(i+j)} \end{vmatrix}$$

pour $j = 1, \ldots, k$ *et* $i = 0, \ldots, k-j$, *où le déterminant du numérateur est développable suivant la première ligne.*

Preuve :

Soit $\{e_1, \ldots, e_k\}$ la base canonique de $\mathbb{R}^k$.

$$E_j^{(i)} = (E_j^{(i)}(u_1^{(i)}), \ldots, E_j^{(i)}(u_k^{(i)}))^T = \sum_{\ell=1}^{k} E_j^{(i)}(u_\ell^{(i)}) e_\ell$$

Or $E_j^{(i)}(u_\ell^{(i)})$ est donné par :

$$E_j^{(i)}(u_\ell^{(i)}) = \begin{vmatrix} u_\ell^{(i)} & \ldots & u_\ell^{(i+j)} \\ u_1^{(i)} & \ldots & u_1^{(i+j)} \\ \ldots & \ldots & \ldots \\ u_j^{(i)} & \ldots & u_j^{(i+j)} \end{vmatrix} \Big/ \begin{vmatrix} 1 & \ldots & 1 \\ u_1^{(i)} & \ldots & u_1^{(i+j)} \\ \ldots & \ldots & \ldots \\ u_j^{(i)} & \ldots & u_j^{(i+j)} \end{vmatrix}$$

Par conséquent on a

$$E_j^{(i)} = \begin{vmatrix} \sum_{\ell=1}^{k} u_\ell^{(i)} e_\ell & \ldots & \sum_{\ell=1}^{k} u_\ell^{(i+j)} e_\ell \\ u_1^{(i)} & \ldots & u_1^{(i+j)} \\ \ldots & \ldots & \ldots \\ u_j^{(i)} & \ldots & u_j^{(i+j)} \end{vmatrix} \Big/ \begin{vmatrix} 1 & \ldots & 1 \\ u_1^{(i)} & \ldots & u_1^{(i+j)} \\ \ldots & \ldots & \ldots \\ u_j^{(i)} & \ldots & u_j^{(i+j)} \end{vmatrix}$$

$$= \begin{vmatrix} u^{(i)} & \ldots & u^{(i+j)} \\ u_1^{(i)} & \ldots & u_1^{(i+j)} \\ \ldots & \ldots & \ldots \\ u_j^{(i)} & \ldots & u_j^{(i+j)} \end{vmatrix} \Big/ \begin{vmatrix} 1 & \ldots & 1 \\ u_1^{(i)} & \ldots & u_1^{(i+j)} \\ \ldots & \ldots & \ldots \\ u_j^{(i)} & \ldots & u_j^{(i+j)} \end{vmatrix} \quad \square$$

D'après les essais numériques suivants cette méthode basée sur le E-algorithme semble être d'ordre deux.

VI.3. - ESSAIS NUMERIQUES

1) E-*algorithme*

Soit à trouver la solution unique $x = -1$ et $y = 1$ du système

$$\begin{cases} x = -\dfrac{y^4}{4} - \dfrac{3}{4} \\ y = -\,0{,}405\ e^{1-x^2} + 1{,}405 \end{cases}$$

Les itérations de base convergent lentement car les valeurs propres du Jacobien calculées à la solution valent ± 0.9.

En partant de $x_0 = y_0 = 0$ on obtient

n	x_n	y_n
0	.0000000000D+00	.0000000000D+00
1	-.8461302364D+00	.6391240269D+00
2	-.9555582817D+00	.9653968000D+00
3	-.9965530579D+00	.9976928503D+00
4	-.9999865591D+00	.9999929514D+00
5	-.1000000000D+01	.1000000000D+01
6	-.1000000000D+01	.1000000000D+01
7	-.1000000000D+01	.1000000000D+01

Et en partant de $x_0 = -\,1.5$ et $y_0 = +\,1.5$ on obtient :

n	x_n	y_n
0	-.1500000000D+01	.1500000000D+01
1	-.6584667980D+00	.1068472321D+01
2	-.1070941074D+01	.1081818858D+01
3	-.8747120124D+00	.9075320785D+00
4	-.9823800200D+00	.9882110891D+00
5	-.9996584541D+00	.9998174633D+00
6	-.9999999882D+00	.1000000029D+01
7	-.1000000000D+01	.1000000000D+01
8	-.1000000000D+01	.1000000000D+01
9	-.1000000000D+01	.1000000000D+01

Considérons maintenant le cas suivant :

$$\begin{cases} x = \frac{y^2}{2} + x - \frac{1}{2} \\ y = \sin(x) + \sin(y-1) + 1 \end{cases}$$

dont une solution est x = 0, y = 1. On a

$$F'(x) = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

Partant de $x_0 = 0.5$ et $y_0 = -1$ on trouve :

n	x_n	y_n
0	.5000000000D+00	-.1000000000D+01
1	.1718476897D+00	.1298087201D+01
2	-.1089737540D+00	.1090164629D+01
3	.5511584212D-05	.1000056665D+01
4	-.4806216993D-12	.1000000004D+01
5	.1110223287D-15	.1000000000D+01

et en partant de $x_0 = -2$ et $y_0 = 2$ on obtient :

n	x_n	y_n
0	-.2000000000D+01	.2000000000D+01
1	.6896687619D+00	.1802759524D+01
2	-.2578267556D+00	.1227809177D+01
3	-.7479227404D-04	.9974208867D+00
4	.3717262761D-07	.1000007035D+01
5	-.4857132174D-15	.1000000000D+01
6	.1942911238D-15	.1000000000D+01

2) La méthode d'Henrici appliquée au système

$$\begin{cases} x = -\dfrac{y^4}{4} - \dfrac{3}{4} \\ y = -0{,}405\ e^{1-x^2} + 1{,}405 \end{cases}$$

donne les résultats suivants :

*) $x_0 = 0$, $y_0 = 0$

n	x_n	y_n
0	.0000000000D+00	.0000000000D+00
1	-.8461302364D+00	.8391240269D+00
2	-.9555582817D+00	.9653968000D+00
3	-.9965530579D+00	.9976928503D+00
4	-.9999865591D+00	.9999929514D+00
5	-.1000000000D+01	.1000000000D+01

**) $x_0 = -1{,}5$, $y_0 = 1{,}5$.

n	x_n	y_n
0	-.1500000000D+01	.1500000000D+01
1	-.6584667980D+00	.1068472321D+01
2	-.1070941074D+01	.1081818858D+01
3	-.8747120124D+00	.9075320785D+00
4	-.9823600200D+00	.9882110891D+00
5	-.9996584541D+00	.9998174633D+00
6	-.9999999882D+00	.1000000029D+01
7	-.1000000000D+01	.1000000000D+01

BIBLIOGRAPHIE

[1] A.C. AITKEN, *"On Bernouilli's numerical solution of algebraic equations"*. Proc. Roy. Soc. Edinburgh, V. 46, 1926, pp. 289-305.

[2] C. BREZINSKI, *"Généralisation des extrapolations polynomiales et rationnelles"*. RAIRO 6ème Année, R.1, 1972, pp. 61-66.

[3] C. BREZINSKI, *"Les fractions continues, connexion avec l'ε-algorithme la table de Padé et les différences réciproques"*. Publication n° 68, Janv. 1976, Laboratoire de calcul de Lille I.

[4] C. BREZINSKI, *"Méthodes d'accélération de la convergence en analyse numérique"*. Thèse d'Etat 1971, Grenoble.

[5] C. BREZINSKI, *"Some new convergence acceleration methods"*. Math. Comp, à paraître.

[6] C. BREZINSKI, "A *general extrapolation*". Numer. Math. 35, pp. 175-187, 1980.

[7] C. BREZINSKI, *"Accélération de la convergence en analyse numérique"*. Lectures Notes in mathematics n° 584, Springer-Verlag, Heidelberg, 1977.

[8] C. BREZINSKI, *"Analyse numérique discrète"*. Publication. Laboratoire de Calcul de Lille I.

[9] C. BREZINSKI, *"Algorithmes d'accélération de la convergence, étude numérique"*. Editions technip, 1978.

[10] G. CLAESSENS, G. LOISOU and L. WUYTACK, Scientific notes : *"Comments on a root finding method using Padé approximation"*. BIT 17, (1977), pp. 360-361.

[11] B. GERMAIN-BONNE, *"Estimation de la limite de suites et formalisation de procédés d'accélération de la convergence"*. Thèse d'Etat 1978, Lille I.

[12] T. HÅVIE, *"Generalized Neville type extrapolation schemes"*. BIT 19, (1979), pp. 204-213.

[13] P. JARRATT, *"A rational iteration function for solving equations"*. Comp. J. 9 (1966), pp. 304-307.

[14] R.C. JOHNSON, *"Alternative approach to Padé approximants"*. Mathematics Department, Durham University, England.

[15] R.F. KING, *"A secant method for multiple roots"*. BIT 17 (1977), pp. 321-328.

[16] R.F. KING, *" An extrapolation method of order four for linear sequences"*. SIAM Journal Numer. Anal. Vol. 16, n° 5, Oct. 1979.

[17] P.J. LAURENT, *"Etude de procédés d'extrapolation en analyse numérique"*. Thèse d'Etat 1964, Grenoble.

[18] A.M. NOUREIN, *"Root determination by use of Pade approximants"*. BIT 16 (1976), pp. 291- 297.

[19] A.M. OSTROWSKI, *"Solution of equations in euclidean and Banach spaces"*. Thrid edition. A.P. New-York and London.

[20] K.J. OVERHOLT, *"Extended Aitken acceleration"*. BIT (1965), pp. 122-132.

[21] P.B. POPOVSKI, *"A note on KING's fifth-order family of methods for solving equation"*. BIT 21 (1981), pp. 129-130.

[22] P.B. POPOVSKI, *"Method of parabolic approximation for solving the equation* $x = \phi(x)$". Intern. J. Computer Maths. 1981, section B, Vol. 9, pp. 243-248.

[23] D. SHANKS, *"Non linear transformations of divergent and slowly convergent sequences"*. J. Math. Phys. 34 (1955), pp. 1-42.

[24] J.F. TRAUB, *"Iterative methods for the solution of equations"*. Prentice Hall-INC-England Cliffs, N.J.

[25] R.R. TUCKER, *"A geometric derivation of D. Shanks* e_k *transform"*. from, the Faculty review, Vol. 65, n° 3, Bulletin OS the NORTH, Carolina A et T State Univ., Spring, 1973.

[26] J. WIMP, *"Sequences transformations and their applications"*. Academic press 1981.

[27] P. WYNN, *"On a procrustean technique for the numerical transformation of slowly convergent sequences and series"*, Proc. Cambridge Phil. Soc. 52, (1956).

[28] C. BREZINSKI, *"Rational approximation to power series"*. Séminaire d'Analyse Numérique et d'Optimisation 1976-1977". Publication n° 102, Oct. 1977 du Laboratoire de Calcul de Lille I.