

HABILITATION A DIRIGER LES RECHERCHES

Quelques contributions sur l'approximation  
rationnelle et les méthodes d'extrapolation

Ana C. Matos  
Laboratoire Paul Painlevé UMR 8524  
Université des Sciences et Technologies de Lille  
59655 Villeneuve d'Ascq CEDEX  
e-mail : Ana.Matos@univ-lille1.fr



# Table des matières

<b>Introduction</b>	<b>4</b>
<b>1 Méthodes d'extrapolation</b>	<b>9</b>
1.1 Introduction . . . . .	9
1.2 Construction de procédés d'extrapolation basés sur des estimations d'erreur . . .	10
1.2.1 Définitions . . . . .	10
1.2.2 Quelques algorithmes d'extrapolation . . . . .	11
1.2.3 Composition d'opérateurs . . . . .	18
1.3 Opérateurs aux différences et méthodes d'extrapolation . . . . .	21
1.3.1 Opérateurs aux différences à coefficients constants . . . . .	22
1.3.2 Quelques classes d'opérateurs aux différences généraux . . . . .	25
1.3.3 Propriétés d'accélération de $T$ basées sur le comportement asymptotique des solutions des équations aux différences linéaires . . . . .	31
1.4 Nouvelle propriété d'accélération pour le E-algorithme . . . . .	35
1.4.1 Définitions et rappels . . . . .	36
1.4.2 Propriété d'accélération . . . . .	37
<b>2 Propriétés de convergence et d'accélération de quelques généralisations des   approximants de Padé</b>	<b>41</b>
2.1 Introduction . . . . .	41
2.2 Polynômes orthogonaux au sens des moindres carrés et applications . . . . .	43
2.2.1 Définitions et propriétés . . . . .	43
2.2.2 Calcul récursif . . . . .	45
2.2.3 Applications . . . . .	46
2.3 Approximants de type Cauchy : propriétés d'accélération et nouvelles transfor- mations de suites . . . . .	47

2.3.1	Propriété d'accélération pour les séries lacunaires . . . . .	48
2.3.2	Nouvelle transformation de suites basée sur A.T.C . . . . .	49
2.3.3	Propriété d'accélération pour des séries périodico-linéaires . . . . .	50
2.3.4	Remarques sur le choix d'un dénominateur polynomial . . . . .	52
2.4	Approximants de type Padé généralisés . . . . .	54
2.4.1	Définitions et propriétés . . . . .	54
2.4.2	Résultat général de convergence . . . . .	56
2.4.3	Résultats de convergence pour différents types de fonctionnelles d'interpolation . . . . .	57
2.4.4	Représentation intégrale de l'erreur . . . . .	62
2.4.5	ATPG pour des fonctions de Stieltjes généralisées. Relation avec les approximants de Baker-Gammel . . . . .	65
<b>3</b>	<b>Approximation rationnelle de séries orthogonales</b>	<b>69</b>
3.1	Introduction . . . . .	69
3.2	Approximants de Frobenius-Padé : calcul et propriétés d'accélération . . . . .	71
3.2.1	Calcul récursif de suites d'approximants . . . . .	72
3.2.2	Convergence de suites de colonnes dans la table de Padé-Legendre . . . . .	76
3.3	Approximants de Frobenius-Padé simultanés . . . . .	79
3.3.1	Définitions et notations . . . . .	80
3.3.2	Relations de récurrence à trois termes . . . . .	82
3.3.3	Un algorithme de type Kronecker . . . . .	83
3.3.4	Calcul de suites diagonales et diagonales régulières . . . . .	84
3.4	Généralisations de l'approximation de Frobenius-Padé à plusieurs variables . . . . .	86
3.4.1	Définitions . . . . .	86
3.4.2	Un algorithme récursif . . . . .	89
3.4.3	Structure de la matrice de Frobenius-Padé . . . . .	91
3.4.4	Approximants de Frobenius-Padé mixtes . . . . .	94
3.4.5	Conclusion et perspectives . . . . .	100

# Introduction

L'essentiel de mes travaux de recherche se répartit sur deux domaines : les méthodes d'extrapolation et l'approximation rationnelle, le but étant dans les deux cas de construire une meilleure approximation de la limite d'une suite de valeurs (réelles, complexes ou vectorielles) ou de la valeur d'une fonction en un point.

Le premier chapitre concerne les méthodes d'extrapolation. Ce travail s'insère dans la continuité du travail fait en thèse. Un nouveau formalisme pour présenter et déduire les méthodes d'extrapolation a été développé. Cette nouvelle approche fournit une méthode de construction systématique des algorithmes d'extrapolation et permet de mieux comprendre son mécanisme. Elle est basée sur la transformation générale de suites  $T_n = L(S_n/D_n)/L(1/D_n)$ , où  $(S_n)$  est la suite pour laquelle on veut calculer la limite,  $(D_n)$  une estimation d'erreur et  $L$  un opérateur aux différences. Une grande majorité des transformations de suites et algorithmes d'extrapolation entrent dans ce formalisme, en considérant différents choix pour l'opérateur  $L$  et différentes estimations d'erreur  $(D_n)$ . Ce cadre a permis de généraliser certaines transformations de suites existantes et d'en proposer des nouvelles.

J'ai ensuite exploré ce formalisme pour approfondir l'étude des méthodes d'extrapolation en m'intéressant à deux problèmes complémentaires :

1. étant donné un opérateur  $L$  et une estimation d'erreur  $(D_n)$ , déterminer le noyau de la transformation  $T$  associée et la classe de suites qu'elle accélère ;
2. étant donnée une classe  $\mathcal{C}$  de suites  $(S_n)$  pour lesquelles on connaît un développement asymptotique de l'erreur (ou les premiers termes de ce développement), comment construire un opérateur  $L$  pour que la transformation de suites correspondante accélère la convergence de cette classe.

La théorie des opérateurs aux différences et les résultats sur le comportement asymptotique des solutions des équations aux différences linéaires ont permis d'étudier ces problèmes et les résultats obtenus sont présentés au Chapitre 1. On termine ce chapitre par un résultat concernant une propriété d'accélération de convergence de suites scalaires ou vectorielles d'un algorithme très général d'extrapolation - le E-algorithme. Ce résultat est obtenu en utilisant des identités de déterminants et la théorie des matrices totalement positives.

Un autre domaine auquel je me suis intéressée est l'approximation d'une fonction  $f(z)$  pour laquelle on connaît le développement en série (ou les premiers termes de ce développement),  $f(z) = \sum_{i=0}^{\infty} c_i g_i(z)$  (en particulier des développements de Taylor, en série de fonctions ou en série de polynômes orthogonaux), par des suites d'approximants rationnels et des généralisations. Ce problème est lié au précédent dans la mesure où les algorithmes pour le calcul récursif de

suites de ces approximants sont des algorithmes d'extrapolation et l'on s'intéresse à l'étude des propriétés que l'on doit imposer à la suite des coefficients ou à la fonction  $f$  pour que la suite des approximants converge vers  $f(z)$  plus vite que la suite des sommes partielles de la série.

Dans le Chapitre 2 je me suis intéressée à différentes généralisations des approximants de Padé et de type Padé, dans le but soit d'améliorer les propriétés de stabilité numérique, soit d'améliorer les propriétés de convergence ou de vitesse de convergence, soit encore de considérer des développements en série plus généraux.

La première généralisation concerne les propriétés de stabilité numérique : sachant que les approximants de Padé sont très sensibles à des perturbations dans les coefficients de la série, l'idée est, au lieu d'imposer qu'un certain nombre de coefficients du développement de l'erreur soient nuls, de minimiser la somme des carrés d'un plus grand nombre. On a ainsi défini les *polynômes orthogonaux au sens des moindres carrés*, donné leurs propriétés et des algorithmes de calcul. Sont construits, basés sur ces polynômes, les *approximants de type Padé au sens des moindres carrés*, pour lesquels de bonnes propriétés de stabilité ont été obtenues.

Je me suis aussi intéressée aux *approximants de type Cauchy d'une série*  $f(z) = \sum_{n=0}^{\infty} a_n z^n$  - généralisation des approximants de type Padé qui correspond à choisir comme dénominateur une fonction analytique  $g$ . J'ai obtenu des propriétés d'accélération de la convergence de ces approximants (par rapport à la suite des sommes partielles de la série) pour des séries lacunaires et des séries périodico-linéaires. A partir de propriétés sur la suite des coefficients du développement de  $f$ , des conditions sur la fonction  $g$  sont données pour que la suite des approximants de type Cauchy converge plus vite et, dans le cas où ces conditions ne seraient pas vérifiées, de nouvelles transformations de suites basées sur ces approximants sont proposées.

Dans la dernière partie de ce chapitre, je me suis intéressée à une autre généralisation des approximants de type Padé aux fonctions données par un développement en une série de fonctions,  $f(z) = \sum_{n=0}^{\infty} c_n g_n(z)$  (la famille  $\{g_i(z)\}_i$  peut être, par exemple, une famille de polynômes orthogonaux). Les *approximants de type-Padé généralisés* consistent à remplacer la fonction génératrice de la famille  $\{g_i(z)\}_i$  par un polynôme vérifiant certaines propriétés d'interpolation. Motivés par les bons résultats numériques de ces approximants, j'ai étudié leurs propriétés de convergence. Différentes classes de fonctionnelles d'interpolation  $L_k$  qui définissent ces approximants ont été considérées, et j'ai étudié quelles propriétés de la fonction génératrice  $G(x, z)$  de la famille  $\{g_i(z)\}_i$ , de la forme linéaire associée et de la suite des coefficients du développement de  $f$ , étaient suffisantes pour que la suite d'approximants converge vers  $f$ . J'ai aussi obtenu une représentation intégrale de l'erreur de ces approximants pour des fonctions de Stieljes généralisées, permettant d'estimer la vitesse de convergence de certaines suites d'approximants.

Le dernier Chapitre est consacré à la généralisation des approximants de Padé à des séries orthogonales,  $f(z) = \sum_{n=0}^{\infty} f_n P_n(z)$ , où  $\{P_n\}_n$  est une famille de polynômes orthogonaux. De nouveaux approximants rationnels - les *approximants de Frobenius-Padé* - ont été définis et généralisés au cas de fonctions vectorielles et de fonctions de plusieurs variables. Différents algorithmes récursifs pour le calcul de certaines suites de ces approximants ont été développés (généralisant des algorithmes pour les approximants de Padé). Dans le cas de fonctions de deux variables, je me suis intéressée à deux types de problèmes : calcul de la valeur en un point d'une suite d'approximants et calcul des coefficients du numérateur et dénominateur d'une suite donnée d'approximants. Pour le premier problème un algorithme récursif de calcul (basé sur le E-

algorithme) est obtenu, pour le second j'ai montré que, pour certaines suites d'approximants, la matrice du système à résoudre pour obtenir les coefficients du dénominateur avait une structure de "displacement rank". Ceci permet le développement d'algorithmes rapides de calcul de ces coefficients. Quelques résultats de convergence et d'accélération de convergence par rapport à la suite des sommes partielles de la série ont été obtenus dans le cas de séries de Legendre d'une variable. Les méthodes spectrales de résolution d'équations aux dérivées partielles ont été une motivation pour l'étude de ces approximants et conduiront certainement à des applications. En effet, ces méthodes fournissent des approximations de la solution sous forme d'une somme partielle d'une série orthogonale, et en présence de singularités de la solution l'approximation n'est pas très bonne. Des résultats numériques très prometteurs ont été obtenus et montrent que ces approximants de Frobenius-Padé accélèrent la convergence des sommes partielles. D'autres applications sont en cours de développement.

Une liste de mes publications ayant servi de base à la rédaction de ce document, notée de [A1] à [A13], est rajoutée en annexe à la fin de ce document. Dans un esprit de synthèse, j'ai présenté, tout au long de ce document, les résultats essentiels de mon travail sans donner les démonstrations. J'ai essayé de les motiver, de dégager les idées et techniques qui me semblaient essentielles et de présenter les conséquences et conclusions plus importantes. Pour les lecteurs qui s'intéresseront à plus de détails, et dans un souci de clarté, j'ai inscrit à côté de chaque proposition, théorème ou corollaire une référence [Axx] indiquant la publication dans laquelle on peut trouver les développements du résultat correspondant. Également, un grand nombre des algorithmes développés tout le long de ce travail ont été programmés et leurs bonnes propriétés de convergence ont été illustrées dans différents exemples numériques que je n'ai pas présentés ici. Le lecteur est donc renvoyé à la publication correspondante.





# Chapitre 1

## Méthodes d'extrapolation

Ce chapitre est basé sur les publications [A4], [A6] et [A9].

### 1.1 Introduction

L'étude et l'intérêt pour les méthodes d'extrapolation date de plusieurs décennies. Les premières méthodes utilisées pour accélérer la convergence de suites furent les procédés de sommation linéaire développés par Euler, Cesaro, Hausdorff et autres (voir, par exemple [95]). Parmi elles on trouve le procédé d'extrapolation de Richardson basé sur l'extrapolation polynomiale en zéro [96] et qui a été à l'origine de la méthode de Romberg pour accélérer la convergence de la méthode des trapèzes [98]. La première méthode d'accélération de la convergence non linéaire a été le procédé  $\Delta^2$  d'Aitken [2]. Il a été généralisé par Shanks en 1955 [101], et un an après Wynn a proposé son  $\varepsilon$ -algorithm [122] pour son implémentation. A partir de là, beaucoup d'autres algorithmes d'extrapolation ont été proposés et étudiés; voir [26, 120] pour une présentation générale et [21] pour un peu d'histoire.

Un cadre assez général a été construit et développé le long de toutes ces années pour la théorie des méthodes d'extrapolation (voir [47, 120] et le premier chapitre de [26]). La situation était un peu différente du point de vue de la construction pratique des méthodes d'extrapolation, i.e. il n'existait pas de méthode systématique pour obtenir les algorithmes d'extrapolation : ces algorithmes étaient présentés séparément, sans lien logique entre eux.

Notre but a été, dans la première partie de ce chapitre, de proposer une approche systématique des algorithmes d'extrapolation et de leur construction (cf. [A6]). Cette approche a déjà été utilisée dans [104] et [105] pour les transformées de Levin, mais son formalisme de base est dû à Weniger [117]. Il nous a semblé un outil très intéressant et puissant qui méritait d'être exploité. Il est basé sur des estimations d'erreur (ou de reste) et des opérateurs d'annihilation. La forme générale des transformations de suites considérées est

$$T_n = \frac{L(S_n/D_n)}{L(1/D_n)} \quad n \in \mathbb{N}, \quad (1.1)$$

où  $(S_n)$  est la suite dont on veut calculer la limite,  $(D_n)$  une estimation d'erreur  $L$  un opérateur linéaire aux différences. Nous montrerons que cette approche conduit à une meilleure

compréhension des mécanismes des algorithmes d'extrapolation, donne un cadre général où tous les procédés actuellement connus peuvent être inclus et fournit de nouveaux algorithmes et de nouveaux résultats théoriques.

Dans la deuxième partie de ce chapitre (cf. [A9]) on étudie les noyaux et on obtient des propriétés d'accélération des transformations de suites de la forme (1.1) pour différents choix de l'opérateur  $L$ , en utilisant des résultats de la théorie des opérateurs linéaires. On s'intéressera aussi au problème suivant : étant donné une classe de suites ayant un développement asymptotique de l'erreur connu, comment peut-on construire un opérateur  $L$  pour lequel la transformation correspondante accélère la convergence de cette classe de suites ? Différentes applications sont données.

Dans la dernière partie (cf. [A4]) nous considérons une des méthodes d'extrapolation la plus générale (aussi bien pour des suites de scalaires que de vecteurs) - le E-algorithme [12]. Nous donnons un résultat d'accélération de convergence des colonnes de cet algorithme pour des classes de suites telles que l'erreur  $(S_n - S)_n$  admette un développement asymptotique dans une échelle de comparaison vérifiant une certaine propriété déterminantale. Ce résultat est généralisé au cas vectoriel.

## 1.2 Construction de procédés d'extrapolation basés sur des estimations d'erreur

### 1.2.1 Définitions

DEFINITION 1.1. [121, p.196] :

Soit  $\mathcal{S}$  l'ensemble des suites complexes. Un opérateur aux différences  $L$  est une application linéaire de  $\mathcal{S}$  dans  $\mathcal{S}$

$$L : u = (u_n) \mapsto L(u) = ((L(u))_n).$$

Un tel opérateur est représenté par une matrice infinie, ou en d'autres termes, par la suite  $(L_n)$  des formes linéaires qui à  $u$  font correspondre le  $n$ -ème terme  $(L(u))_n$  de la suite  $L(u)$ , . Par souci de simplicité, on écrira par la suite  $L(u)_n$  ou  $L_n(u)$  indifféremment. On sait [5] que la forme la plus générale d'un opérateur aux différences est donnée par

$$L_n(u) = \sum_{i=-p_n}^{q_n} G_i(n)u_{n+i},$$

où  $p_n$  et  $q_n$  sont des entiers non négatifs qui peuvent dépendre de  $n$ ,  $u_i = 0$  pour  $i < 0$ , et où les  $G_i$  sont des fonctions de  $n$  qui peuvent aussi dépendre de suites auxiliaires données.

DEFINITION 1.2. [117, p.212] :

$L$  est un opérateur aux différences d'annihilation pour la suite  $a = (a_n)$  si

$$\exists N \text{ tel que } \forall n \geq N, L_n(a) = 0$$

(on supposera  $N = 0$  sans perte de généralité).

DEFINITION 1.3. [117, p.212] : La suite  $(D_n)$  est appelée une estimation d'erreur de la suite  $(S_n)$  si  $\forall n, S_\infty - S_n = a_n D_n$ , où  $(a_n)$  est une suite inconnue et  $S_\infty$  un scalaire (qui est usuellement la limite de la suite si  $\lim a_n D_n = 0$ , sinon on l'appelle l'anti-limite).

La première étape dans la construction d'une méthode d'extrapolation consiste à supposer que la suite admet un certain comportement ; en d'autres termes, on construit un algorithme capable de calculer la limite exacte  $S_\infty$  (ou l'anti-limite) pour une famille de suites ayant le même comportement, en utilisant un opérateur aux différences d'annihilation (dans une approche un peu plus générale que celle utilisée dans [117]). Supposons que la suite  $S = (S_n)$  vérifie

$$\forall n \quad S_\infty - S_n = a_n D_n$$

où  $D = (D_n)$  est une suite connue et  $a = (a_n)$  une suite inconnue. On veut construire une transformation de suites  $T : (S_n) \mapsto (T_n)$  telle que  $\exists N, \forall n \geq N, T_n = S_\infty$ . On suppose qu'il existe une suite  $b = (b_n)$  telle que  $L(b)$  soit connu et  $L$  soit un opérateur aux différences d'annihilation pour  $a - b$ . Alors on a

$$\frac{S_\infty}{D_n} - \frac{S_n}{D_n} - b_n = a_n - b_n.$$

En appliquant  $L$  et par linéarité on obtient  $\forall n \geq N$

$$S_\infty L_n(1/D) - L_n(S/D) - L_n(b) = L_n(a - b) = 0 \quad \Leftrightarrow \quad S_\infty = \frac{L_n(S/D) + L_n(b)}{L_n(1/D)}.$$

On peut alors définir la transformation  $T : (S_n) \mapsto (T_n)$  par

$$T_n = \frac{L_n(S/D) + L_n(b)}{L_n(1/D)},$$

et par construction on a  $\forall n \geq N, T_n = S_\infty$  ssi  $\forall n \geq N, S_\infty - S_n = a_n D_n$ . La forme la plus simple de cette transformation correspond à  $b = (0)$ , ce qui sera le cas souvent par la suite.

Les cas vectoriel, matriciel et confluent peuvent aussi être inclus dans ce cadre général (pour plus de détail voir [A6], [32])

### 1.2.2 Quelques algorithmes d'extrapolation

On va maintenant considérer différents choix pour l'opérateur linéaire  $L$  et quelques choix particuliers pour les estimations d'erreur  $(D_n)$ . On retrouve quelques transformations connues et cela nous permettra de proposer quelques généralisations naturelles. Les paragraphes (a) à (h) correspondent à un choix de l'opérateur  $L$  indépendant de la suite  $(S_n)$  ; dans les paragraphes (i) à (k),  $L$  dépend de  $(S_n)$ . Le cas où  $L$  est une combinaison linéaire de plusieurs opérateurs est considéré dans (l). Ces résultats sont exposés d'une façon plus complète dans [A6].

#### (a) Une transformation simple

Le choix le plus simple consiste à prendre pour l'opérateur  $L$  l'identité ce qui donne

$$T_n = S_n + b_n D_n.$$

Le noyau de cette transformation est l'ensemble de suites de la forme  $S_\infty - S = b_n D_n$ . Une telle transformation a été étudiée dans [27].

(b) La procédure  $\Theta$ 

Si  $\forall n, a_n = a, b_n = 0$ , et si  $L = \Delta$  (l'opérateur aux différences) on obtient

$$T_n = \frac{D_{n+1}S_n - D_nS_{n+1}}{D_{n+1} - D_n} = S_n - \frac{\Delta S_n}{\Delta D_n} D_n,$$

qui est la procédure  $\Theta$  étudiée dans [14].

Quand  $D_n = x_n - x$ , où  $(x_n)$  est une suite connue convergeant vers une limite connue  $x$ , alors le deuxième procédé standard de Germain-Bonne [59] est retrouvé. Pour des choix particuliers de la suite  $(x_n)$ , on peut construire de nouveaux procédés d'accélération basés sur des tests de convergence et sur des estimations d'erreur. Ceci a été fait dans [20] et développé dans [79].

## (c) Procédés de sommation

Si  $L_n(u) = \sum_{i=0}^{k_n} G_i(n)u_i$  où les  $G_i(n)$  sont des quantités fixées, si  $L(b) = 0$ , et  $\forall n, D_n = 1$  alors  $T$  est un procédé de sommation linéaire, comme défini par exemple dans [120]. La convergence de la suite  $(T_n)$  vers  $S_\infty$  pour toute suite  $(S_n)$  convergeant vers  $S_\infty$  est donnée par le théorème de Toeplitz. Les propriétés d'accélération de ce procédé ont été étudiées dans [119].

## (d) Transformations colonnes et transformations diagonales

Si  $b_n = 0, D_n = \Delta S_n$  et  $L = \Delta^k$ , alors pour  $k = 1$  on obtient le procédé  $\Delta^2$  d'Aitken, et  $k = 2$  correspond à la deuxième colonne du  $\theta$ -algorithme [8]. Le cas d'une valeur quelconque de  $k$  a été étudié par Drummond [49].

Cet exemple nous montre qu'il est intéressant de considérer le cas où les formes linéaires définissant  $L$  dépendent d'un second entier  $k$ . On les désigne par  $L_{n,k}$ , et on considère comme précédemment l'opérateur  $L$  défini par la suite  $(L_n = L_{n,k})_n$  pour une valeur fixée de  $k$  ou l'opérateur  $L'$  défini en inversant les rôles de  $n$  et de  $k$ , c'est-à-dire, par la suite  $(L'_k = L_{n,k})_k$  ( $n$  fixé). Regardons sur un exemple les deux choix alternatifs que l'on obtient :

(i)  $k$  fixé,  $n$  varie : dans ce cas,  $L = \Delta^k$ ,

$$T_n = \Delta^k(S_n/\Delta S_n)/\Delta^k(1/\Delta S_n), \quad n = 0, 1, \dots$$

et on parle de *transformation colonne*;

(ii)  $n$  fixé,  $k$  varie : dans ce cas,  $L'_k(u) = \Delta^k u_n$ , ce qui donne

$$T'_k = \Delta^k(S_n/\Delta S_n)/\Delta^k(1/\Delta S_n), \quad k = 0, 1, \dots$$

et on parle de *transformation diagonale*.

On explique facilement les noms donnés à ces transformations si l'on pose

$$T_k^{(n)} = \Delta^k(S_n/\Delta S_n)/\Delta^k(1/\Delta S_n),$$

et l'on dispose ces quantités dans un tableau bidimensionnel comme suit :

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

$n$  indique le plus petit indice des termes de la suite  $S$  utilisés dans le calcul de  $T_k^{(n)}$  et  $k$  a un rapport avec le nombre de termes de  $S$  utilisés dans ce calcul, et donc avec la complexité de la transformation de suites  $T$ .

(e) *Transformations de Levin*

Si  $b_n = 0$  et  $a_n$  est un polynôme de degré  $k - 1$  dans la variable  $(n + 1)$ , alors  $L = \Delta^k$  est un opérateur aux différences d'annihilation pour  $(a_n)$ . Cette remarque a été utilisée par Levin [75] pour construire différentes transformations de suites basées sur différents choix de la suite  $(D_n)$  et par Weniger [117] pour les généraliser. Les transformations de Levin peuvent être généralisées en prenant pour  $a_n$  un polynôme de degré  $k - 1$  en  $x_n$ . Dans ce cas, l'opérateur aux différences d'annihilation est l'opérateur des différences divisées  $\delta_k$  (qui sera défini au paragraphe suivant) et les transformations de Levin apparaissent comme une généralisation de la méthode de Richardson. La généralisation du procédé de Richardson définie par Sidi [102] rentre aussi dans notre formalisme, aussi bien que la méthode de Drummond [49]. Les connexions entre ces différentes méthodes sont discutées plus en détail dans [117, Weniger, pp.236–238] and [26, BRZ, pp.116–119].

(f) *Extrapolation de Richardson*

On rappelle que le procédé de Richardson [96] permet de calculer la limite d'une suite  $(S_n)$  vérifiant

$$S_n = S_\infty - c_1 x_n - \cdots - c_k x_n^k \quad (1.2)$$

où  $(x_n)$  est une suite auxiliaire. Donc

$$S_\infty - S_n = (c_1 + \cdots + c_k x_n^{k-1}) x_n,$$

ce qui consiste à prendre  $D_n = x_n$  et  $a_n = c_1 + \cdots + c_k x_n^{k-1}$ . L'opérateur  $\delta_k$  des différences divisées d'ordre  $k$  aux points  $x_i$  est un opérateur d'annihilation pour la suite  $(a_n)$ . (On rappelle que cet opérateur est défini récursivement par

$$\delta_{k+1}(u_n) = \frac{\delta_k(u_{n+1}) - \delta_k(u_n)}{x_{n+k+1} - x_n}$$

avec  $\delta_0(u_n) = u_n$ ). En utilisant les notations de (d) on peut construire un tableau bidimensionnel par

$$T_k^{(n)} = \frac{\delta_k(S_n/x_n)}{\delta_k(1/x_n)},$$

avec  $T_0^{(n)} = S_n$ . Par construction, si la suite  $(S_n)$  a la forme (1.2), alors  $\forall n, T_k^{(n)} = S_\infty$ . On peut montrer ([A4]) en utilisant la relation de récurrence des différences divisées que ces quantités  $T_k^{(n)}$  sont exactement les mêmes que celles calculées par le schéma d'extrapolation de Richardson

$$\begin{cases} U_k^{(n)} = \frac{x_{n+k}U_{k-1}^{(n)} - x_nU_{k-1}^{(n+1)}}{x_{n+k} - x_n} & k \geq 1, \quad n \geq 0 \\ U_0^{(n)} = S_n & n \geq 0 \end{cases}$$

Ce schéma est plus simple que son implémentation via les différences divisées car c'est une relation entre les  $T_k^{(n)}$  au lieu de deux relations de récurrence séparées pour les numérateurs et les dénominateurs. Cependant, si l'on veut appliquer la méthode simultanément à plusieurs suites avec la même suite auxiliaire  $(x_n)$  il sera préférable d'utiliser le schéma précédent basé sur les différences divisées (car les dénominateurs sont les mêmes).

### (g) Le procédé d'Overholt

Le procédé d'Overholt [92] est défini par

$$\begin{cases} V_{k+1}^{(n)} = \frac{\Delta(V_k^{(n)})/(\Delta S_{n+k})^{k+1}}{\Delta(1/(\Delta S_{n+k})^{k+1})} & n \geq 0, \quad k \geq 1 \\ V_0^{(n)} = S_n & n \geq 0 \end{cases} .$$

Il rentre dans notre formalisme car on voit immédiatement que  $\forall n, V_{k+1}^{(n)} = S_\infty$  ssi il existe une constante  $c_k$  telle que

$$S_\infty - V_k^{(n)} = c_k(\Delta S_{n+k})^{k+1}.$$

Dans ce cas ce qui est intéressant c'est que ce formalisme permet de généraliser ce procédé : si l'on remplace la constante  $c_k$  par un polynôme de degré  $m_k - 1$  en  $n$  on obtient

$$V_{k+1}^{(n)} = \frac{\Delta^{m_k}(V_k^{(n)})/(\Delta S_{n+k})^{k+1}}{\Delta^{m_k}(1/(\Delta S_{n+k})^{k+1})}.$$

Si  $c_k$  est remplacé par un polynôme en  $x_n$ , alors on doit utiliser les différences divisées au lieu de l'opérateur  $\Delta$ . Ces nouvelles transformations, aussi bien que leurs propriétés d'accélération restent à étudier.

### (h) Le E-algorithme

Considérons maintenant des exemples un peu plus compliqués : on suppose que les suites  $G_i$  dans la définition de l'opérateur général  $L$  dépendent elles mêmes de quelques termes de la suite  $(S_n)$ . On choisit

$$L_n(u) = L'_k(u) = \begin{vmatrix} u_n & u_{n+1} & \cdots & u_{n+k} \\ g_1(n) & g_1(n+1) & \cdots & g_1(n+k) \\ \vdots & \vdots & & \vdots \\ g_k(n) & g_k(n+1) & \cdots & g_k(n+k) \end{vmatrix},$$

avec  $(g_i(n))_n$  des suites auxiliaires données (qui peuvent dépendre de  $(S_n)$ ), et on définit la transformation  $T$  comme ci-dessus,

$$T_n = \frac{L_n(S/D)}{L_n(1/D)} \quad n \in \mathbb{N}, \quad T'_k = \frac{L'_k(S/D)}{L'_k(1/D)} \quad k \in \mathbb{N}.$$

Si pour tout  $n$ ,  $D_n = 1$ , et si l'opérateur  $L$  est donné par la suite  $(L_n)$  de formes linéaires définies ci-dessus, alors la transformation  $T$  correspond à la colonne  $k$  du E-algorithme [12, 62] avec les  $(g_i(n))$  comme suites auxiliaires. Comme  $L$  est l'opérateur d'annihilation pour  $a_n = \sum_{i=1}^k c_i g_i(n)$ ,  $c_i$  étant des constantes, on obtiendra

$$T_n = S_\infty \quad \Leftrightarrow \quad S_\infty - S = \sum_{i=1}^k c_i g_i(n).$$

Si  $L'$  est l'opérateur défini par la suite de formes  $(L'_k)$ , alors  $T$  correspond à la  $n$ -ème diagonale du E-algorithme.

Le E-algorithme est l'un des algorithmes d'extrapolation les plus généraux : presque toutes les transformations de suites existantes sont des cas particuliers correspondant à différents choix des suites auxiliaires  $(g_i(n))_n$  (un résultat nouveau d'accélération de ces colonnes sera donné dans la section 1.3.3). Un cas particulier très important est celui correspondant au choix suivant

$$L_n(u) = L'_k(u) = \begin{vmatrix} u_n & u_{n+1} & \cdots & u_{n+k} \\ \Delta S_n & \Delta S_{n+1} & \cdots & \Delta S_{n+k} \\ \vdots & \vdots & & \vdots \\ \Delta S_{n+k-1} & \Delta S_{n+k} & \cdots & \Delta S_{n+2k-1} \end{vmatrix}.$$

Si  $D_n = 1$  et  $b_n = \sum_{i=1}^k c_i \Delta S_{n+i-1}$  avec  $c_i$  des constantes arbitraires, alors la transformation correspondante  $T$  coïncide avec la  $k$ -ème colonne de la transformation de Shanks [101], ou, ce qui est équivalent, la  $2k$ -ème colonne de l' $\varepsilon$ -algorithme de Wynn [122]. Dans le cas où  $L'$  est défini par la suite  $(L'_k)$ ,  $T$  correspond à la  $n$ -ème diagonale de la transformation de Shanks ou de l' $\varepsilon$ -algorithme. Les première et seconde généralisations de l' $\varepsilon$ -algorithme [9] rentrent aussi dans ce cadre en remplaçant l'opérateur  $\Delta$  par un opérateur plus général [10, 99].

### (i) Approximants de Padé et de type-Padé

Considérons maintenant un choix particulier pour les suites  $S = (S_n)$  : soit  $S_n$  la somme partielle d'ordre  $n$  d'une série de puissances  $S_\infty = f(z) = \sum_{i=0}^{\infty} c_i z^i$ . Comme on a  $S_\infty - S_n = c_{n+1} z^{n+1} + \cdots$ , on peut choisir  $D_n = 1$  et on définit

$$L_p(u) = L'_q(u) = \begin{vmatrix} z^q u_{p-q} & z^{q-1} u_{p-q+1} & \cdots & u_p \\ c_{p-q+1} & c_{p-q+2} & \cdots & c_{p+1} \\ \vdots & \vdots & & \vdots \\ c_p & c_{p+1} & \cdots & c_{p+q} \end{vmatrix},$$

avec la convention  $c_i = 0$  pour  $i < 0$ . Si  $b_p = \sum_{i=1}^q c_{p+i} z^{p+i}$  et l'opérateur  $L$  est défini par la suite  $(L_p)$ , alors  $T_p$  est l'approximant de Padé  $[p/q]$  de la série  $f$  au point  $z$  [33]. En effet, on

obtient

$$f(z)L(1) - L(S) = L((c_{p+1}z^{p+1} + \dots)_p).$$

En utilisant la définition de  $L$  on voit facilement que

$$L((c_{p+1}z^{p+1} + \dots)_p) = \begin{vmatrix} z^q(c_{p-q+1}z^{p-q+1} + \dots) & \dots & c_{p+1}z^{p+1} + \dots \\ c_{p-q+1} & \dots & c_{p+1} \\ \vdots & & \vdots \\ c_p & \dots & c_{p+q} \end{vmatrix} = z^q O(z^{p+1}),$$

ce qui correspond à la propriété d'ordre d'approximation des approximants de Padé, et pour une valeur fixe de  $q$  on a  $L_p(S)/L_p(1) = [p/q]_f(z)$ . On a le même résultat si  $p$  est fixe et  $q$  varie.

Si maintenant on considère le choix

$$L_n(u) = a_0^{(k)} z^k u_n + a_1^{(k)} z^{k-1} u_{n+1} + \dots + a_k^{(k)} u_{n+k},$$

alors, pour  $b_n = 0$  et  $D_n = 1$ , on obtient facilement

$$f(z)L_n(1) - L_n(S) = O(z^{n+k+1}).$$

Ceci montre que  $L_n(S)/L_n(1)$  est l'approximant de type-Padé  $(n + k/k)$  de  $f$  avec polynôme générateur  $v_k(z) = a_0^{(k)} + \dots + a_k^{(k)} z^k$  [11], car  $L_n(1)$  est un polynôme de degré  $k$  et  $L_n(S)$  un polynôme de degré  $n + k$ .

Les approximants de Padé et de type Padé rentrent donc aussi dans notre formalisme.

### (j) le $\theta$ -algorithme

Le  $\theta$ -algorithme [8] peut s'écrire sous la forme suivante

$$\theta_{2k+2}^{(n)} = \frac{\Delta(\theta_{2k}^{(n+1)}/D_{2k+1}^{(n)})}{\Delta(1/D_{2k+1}^{(n)})}$$

avec

$$\theta_{2k+1}^{(n)} = \theta_{2k-1}^{(n+1)} + D_{2k}^{(n)}, \quad D_k^{(n)} = 1/(\theta_k^{(n+1)} - \theta_k^{(n)})$$

et  $\theta_{-1}^{(n)} = 0, \theta_0^{(n)} = S_n$ . Donc cet algorithme rentre dans notre formalisme car il correspond à l'itération de notre procédure de base avec  $S = (\theta_{2k}^{(n+1)})$ ,  $D = (D_{2k+1}^{(n)})$  et  $b = (0)$ .

De l'expression précédente on obtient d'une façon triviale que  $\forall n, \theta_{2k+2}^{(n)} = S_\infty$  ssi il existe une constante  $c_k$  telle que

$$S_\infty - \theta_{2k}^{(n+1)} = c_k D_{2k+1}^{(n)} = c_k / \Delta \theta_{2k+1}^{(n)}.$$

Ce cadre permet, une fois encore, d'envisager la construction de nouvelles transformations : si  $c_k$  est remplacé par un polynôme de degré  $m_k - 1$  en  $n$ , on obtient la généralisation suivante du  $\theta$ -algorithme

$$\theta_{2k+2}^{(n)} = \frac{\Delta^{m_k}(\theta_{2k}^{(n+1)}/D_{2k+1}^{(n)})}{\Delta^{m_k}(1/D_{2k+1}^{(n)})}.$$



Et si  $c_k$  est remplacé par un polynôme en  $x_n$ , alors on doit utiliser les différences divisées au lieu de l'opérateur  $\Delta$ . L'étude de ces transformations et leurs propriétés est encore un problème ouvert.

**(k)** *Le  $\rho$ -algorithme*

L'extrapolation à l'infini par une fonction rationnelle consiste à supposer que la suite  $S$  admet la forme suivante :

$$S_n = \frac{S_\infty x_n^k + c_1 x_n^{k-1} + \cdots + c_k}{x_n^k + b_1 x_n^{k-1} + \cdots + b_k}.$$

Cela conduit à une généralisation du  $\rho$ -algorithme de Wynn [123] dont les quantités sont définies par un quotient de deux déterminants :

$$\rho_{2k}^{(n)} = \frac{\begin{vmatrix} 1 & S_n & x_n & x_n S_n & \cdots & x_n^{k-1} & x_n^{k-1} S_n & x_n^k S_n \\ 1 & S_{n+1} & x_{n+1} & x_{n+1} S_{n+1} & \cdots & x_{n+1}^{k-1} & x_{n+1}^{k-1} S_{n+1} & x_{n+1}^k S_{n+1} \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots \\ 1 & S_{n+2k} & x_{n+2k} & x_{n+2k} S_{n+2k} & \cdots & x_{n+2k}^{k-1} & x_{n+2k}^{k-1} S_{n+2k} & x_{n+2k}^k S_{n+2k} \end{vmatrix}}{\begin{vmatrix} 1 & S_n & x_n & x_n S_n & \cdots & x_n^{k-1} & x_n^{k-1} S_n & x_n^k \\ 1 & S_{n+1} & x_{n+1} & x_{n+1} S_{n+1} & \cdots & x_{n+1}^{k-1} & x_{n+1}^{k-1} S_{n+1} & x_{n+1}^k \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots \\ 1 & S_{n+2k} & x_{n+2k} & x_{n+2k} S_{n+2k} & \cdots & x_{n+2k}^{k-1} & x_{n+2k}^{k-1} S_{n+2k} & x_{n+2k}^k \end{vmatrix}}.$$

Dans notre formalisme, cela correspond à la transformation  $T$  avec  $D_n = 1$  et  $L_n(u) = \begin{vmatrix} 1 & S_n & x_n & x_n S_n & \cdots & x_n^{k-1} & x_n^{k-1} S_n & x_n^k u_n \end{vmatrix}$  (pour simplifier l'écriture nous avons remplacé le déterminant par sa première ligne). Ces quantités peuvent être calculées récursivement par le  $\rho$ -algorithme dont les règles sont

$$\rho_{k+1}^{(n)} = \rho_{k-1}^{(n+1)} + (x_{n+k+1} - x_n) / (\rho_k^{(n+1)} - \rho_k^{(n)})$$

avec  $\rho_{-1}^{(n)} = 0$  and  $\rho_0^{(n)} = S_n$ . Ces quantités sont les différences réciproques qui jouent le rôle, en interpolation rationnelle et en fractions continues, des différences divisées en interpolation polynômiale.

**(1)** *Transformations composites*

Les transformations composites rentrent aussi dans notre cadre théorique et encore une fois ce formalisme permet de les généraliser en proposant de nouvelles transformations.

Soient  $L^{(1)}, L^{(2)}, \dots, L^{(k)}$  des opérateurs aux différences et  $(b_i^{(n)})$  des suites données. On définit l'opérateur aux différences  $L$  par

$$L_n = \sum_{i=1}^k b_i^{(n)} L_n^{(i)}$$

et la transformation  $T$  comme précédemment par  $T_n = L_n(S/D)/L_n(1/D)$ . Si on définit les transformations  $T^{(i)} : (S_n) \mapsto (T_i^{(n)})$  par

$$T_i^{(n)} = L_n^{(i)}(S/D)/L_n^{(i)}(1/D),$$

on a

$$T_n = \sum_{i=1}^k c_i^{(n)} T_i^{(n)} \text{ avec } c_i^{(n)} = b_i^{(n)} \frac{L_n^{(i)}(1/D)}{L_n(1/D)}.$$

Alors  $\sum_{i=1}^k c_i^{(n)} = 1$  et l'on retrouve les transformations composites introduites et étudiées en [18].

Pour que ces transformations soient intéressantes, on doit pouvoir choisir les  $(b_i^{(n)})$  de façon à ce que le noyau de  $T$  contienne les noyaux des transformations  $T^{(i)}$ . Nous rappelons que le noyau d'une transformation  $T$  est l'ensemble des suites pour lesquelles  $T(S_n) = S_\infty \quad \forall n \geq N$ . Dans notre formalisme il s'écrit

$$\text{Ker}(T) = \{(S_n) : \forall n \in \mathbb{N} \quad S_n - S_\infty = a_n D_n, \text{ and } \exists N \in \mathbb{N} \quad \forall n \geq N \quad L(a_n) = 0\}.$$

On a le résultat

PROPOSITION 1.4. **[A6]** *Soit l'opérateur  $L$  défini par*

$$L_n = \begin{vmatrix} L_n^{(1)} & \cdots & L_n^{(k)} \\ \Delta T_1^{(n)} & \cdots & \Delta T_k^{(n)} \\ \vdots & & \vdots \\ \Delta T_1^{(n+k-2)} & \cdots & \Delta T_k^{(n+k-2)} \end{vmatrix}$$

et  $\forall n, D_n = 1$ . Soit  $T$  la transformation qui lui est associée. Alors

$$\text{Ker}(T) \supset \cup_{i=1}^k \text{Ker}(T^{(i)}).$$

Cette transformation, qui est définie en [18], généralise la transformation de Shanks, que l'on retrouve pour le choix  $T_i^{(n)} = S_{n+i-1}$ .

Une autre généralisation possible consiste à prendre  $L_n(u) = \sum_{i=1}^k b_i^{(n)} L_n(u/D^{(i)})$  avec  $D^{(i)}$  des suites données.

Cette présentation n'est pas exhaustive. D'autres transformations et méthodes peuvent être déduites de ce formalisme général. Pour une étude plus complète voir **[A6]**.

### 1.2.3 Composition d'opérateurs

Dans cette section on notera par  $L(u_n)$  le  $n$ -ème élément de la suite  $(L(u))$  et les différents opérateurs seront notés par  $L^{(i)}$ .

On suppose, comme dans le paragraphe précédent, que  $S_\infty - S_n = a_n D_n$ . Dans la pratique, la plupart du temps on ne connaît pas un opérateur d'annihilation pour la suite  $(a_n)$ . Donc on commence par choisir un opérateur  $L^{(1)}$  et l'on considère la transformation de suites associée

$$T_1^{(n)} = L^{(1)}(S_n/D_n)/L^{(1)}(1/D_n) \text{ qui vérifie } S_\infty - T_1^{(n)} = L^{(1)}(a_n)/L^{(1)}(1/D_n).$$

Si on pose  $a_1^{(n)} = L^{(1)}(a_n)$  et  $D_1^{(n)} = 1/L^{(1)}(1/D_n)$ , alors

$$S_\infty - T_1^{(n)} = a_1^{(n)} D_1^{(n)}.$$

On peut alors choisir un autre opérateur  $L^{(2)}$  (qui peut coïncider avec le premier), ce qui conduit à une deuxième transformation

$$T_2^{(n)} = L^{(2)}(T_1^{(n)}/D_1^{(n)})/L^{(2)}(1/D_1^{(n)}) \text{ avec } S_\infty - T_2^{(n)} = L^{(2)}(a_1^{(n)})/L^{(2)}(1/D_1^{(n)}).$$

Ces choix d'opérateurs peuvent être guidés, comme on le verra dans la section suivante, par des informations sur le développement asymptotique de l'erreur.

On peut ainsi définir une procédure itérative

PROPOSITION 1.5. **[A6]** On pose  $T_0^{(n)} = S_n$ ,  $D_0^{(n)} = D_n$ , et on définit pour  $k = 0, 1, \dots$ ,

$$T_{k+1}^{(n)} = \frac{L^{(k+1)}(T_k^{(n)}/D_k^{(n)})}{L^{(k+1)}(1/D_k^{(n)})} \text{ et } D_{k+1}^{(n)} = 1/L^{(k+1)}(1/D_k^{(n)}).$$

L'erreur de la suite  $(T_{k+1}^{(n)})_n$  est donné par

$$S_\infty - T_{k+1}^{(n)} = a_{k+1}^{(n)} D_{k+1}^{(n)} \text{ avec } a_{k+1}^{(n)} = L^{(k+1)}(a_k^{(n)}).$$

De plus

$$T_k^{(n)} = \frac{L^{(k)} \cdots L^{(1)}(S_n/D_n)}{L^{(k)} \cdots L^{(1)}(1/D_n)}, \quad a_k^{(n)} = L^{(k)} \cdots L^{(1)}(a_n) \text{ et } D_k^{(n)} = 1/L^{(k)} \cdots L^{(1)}(1/D_n).$$

Ceci permettra, dans des cas particuliers et pour des choix différents des  $L^{(i)}$ , d'obtenir des propriétés d'accélération pour ces méthodes.

Considérons l'exemple suivant : si  $\forall k$ ,  $L^{(k)}$  est l'opérateur  $\Delta$  alors

$$T_{k+1}^{(n)} = T_k^{(n)} - \frac{\Delta T_k^{(n)}}{\Delta D_k^{(n)}} D_k^{(n)} \quad (\Delta \text{ agit sur l'indice } n)$$

et

$$D_{k+1}^{(n)} = -D_k^{(n)} D_k^{(n+1)} / \Delta D_k^{(n)}.$$

Ceci est un cas particulier de l'algorithme (30) donné par Homeier [64] quand les quantités  $\Delta r_k^{(n)}$  sont toutes égales à 1.

Une autre application de cette procédure itérative est

PROPOSITION 1.6. **[A6]** *On considère la famille de transformations définie par*

$$T_k^{(n)} = \frac{L^{(k)}(S_n/D_n)}{L^{(k)}(1/D_n)}$$

où les opérateurs  $L^{(k)}$  vérifient la relation de récurrence suivante

$$L^{(k+1)}(u_n) = \alpha_n^{(k)} L^{(k)}(u_{n+1}) + \beta_n^{(k)} L^{(k)}(u_n)$$

avec  $L^{(0)}(u_n) = u_n$ ,  $(\alpha_n^{(k)})$  et  $(\beta_n^{(k)})$  des suites données. Alors

$$T_{k+1}^{(n)} = \frac{\mathcal{L}^{(k+1)}(T_k^{(n)}/D_k^{(n)})}{\mathcal{L}^{(k+1)}(1/D_k^{(n)})},$$

avec

$$\begin{aligned} \mathcal{L}^{(k+1)}(u_n) &= \alpha_n^{(k)} u_{n+1} + \beta_n^{(k)} u_n \\ D_k^{(n)} &= 1/L^{(k)}(1/D_n) = 1/\mathcal{L}^{(k)} \mathcal{L}^{(k-1)} \dots \mathcal{L}^{(1)}(1/D_n). \end{aligned}$$

Ceci montre que  $T_{k+1}^{(n)}$  peut être obtenue par composition de différents opérateurs. Comme exemple, considérons les transformations de Levin introduites dans **(e)**. L'algorithme qui les implémente donné dans [105] nous permet d'écrire

$$T_k^{(n)} = \frac{\Delta^k((n+1)^{k-1} S_n/D_n)}{\Delta^k((n+1)^{k-1}/D_n)} = \frac{L^{(k)}(S_n/D_n)}{L^{(k)}(1/D_n)}$$

avec

$$L^{(k+1)}(u_n) = L^{(k)}(u_{n+1}) - \frac{(n+1)(n+k+1)^{k-1}}{(n+k+2)^k} L^{(k)}(u_n), \quad L^{(0)}(u_n) = u_n$$

et  $(D_n)$  choisi selon la transformation de Levin considérée. On conclut que cet algorithme correspond à notre procédé itératif précédent en posant

$$\alpha_n^{(k)} = 1 \quad \text{and} \quad \beta_n^{(k)} = -\frac{(n+1)(n+k+1)^{k-1}}{(n+k+2)^k}.$$

On peut généraliser cette procédure d'itération en considérant des suites  $D_k^{(n)}$  arbitraires (et pas données par la relation de récurrence précédente). Ceci permettra d'inclure des transformations comme le E-algorithme, le procédé d'Overholt ou le  $\theta$ -algorithme. On obtiendra ainsi des transformations  $T_k$  définies par

$$T_k^{(n)} = \frac{L^{(k)}(b_{k-1}^{(n)} L^{(k-1)}(\dots L^{(2)}(b_1^{(n)} L^{(1)}(S_n/D_n) \dots))}{L^{(k)}(b_{k-1}^{(n)} L^{(k-1)}(\dots L^{(2)}(b_1^{(n)} L^{(1)}(1/D_n) \dots))}.$$

avec  $b_k^{(n)} = 1/D_k^{(n)} L^{(k)}(1/D_{k-1}^{(n)})$ ,  $a_k^{(n)} = b_k^{(n)} L^{(k)}(a_{k-1}^{(n)})$ .

Les noyaux de ces transformations composées ont été étudiés dans [29], et quand tous les opérateurs  $L^{(i)}$  sont égaux à  $\Delta$ , la théorie est développée dans [34].

Différents exemples de transformations de suites qui rentrent dans ce cadre sont données dans **[A6]**.

### 1.3 Opérateurs aux différences et méthodes d'extrapolation

Le formalisme développé dans la section précédente nous a fourni une façon systématique de construire une grande partie des méthodes d'extrapolation existantes, de généraliser et construire de nouvelles méthodes et de mieux comprendre le mécanisme de l'extrapolation. Nous allons maintenant exploiter ce formalisme dans le sens de l'accélération de la convergence, c'est-à-dire, obtenir les propriétés du noyau et celles d'accélération pour des transformations de suites de la forme générale

$$\begin{cases} T: \mathcal{S} & \longrightarrow \mathcal{S} \\ S = (S_n) & \longmapsto T(S) = (T(S_n)) = (T_n) : T_n = \frac{L(S_n/D_n)}{L(1/D_n)}, \quad n \in \mathbb{N} \end{cases} \quad (1.3)$$

où  $\mathcal{S}$  est l'ensemble des suites de nombres complexes. L'ensemble des résultats est traité en détail dans [A9].

Dans l'étude des méthodes d'extrapolation, il y a essentiellement deux types de problèmes qui se posent.

1) Etant donnée une transformation de suites  $T$  (dans notre nouvelle approche, ceci est équivalent à la donnée d'un opérateur aux différences  $L$  et d'une estimation d'erreur  $(D_n)$ ), déterminer :

- son noyau, c'est-à-dire, l'ensemble des suites pour lesquelles  $T(S_n) = S_\infty$ ,  $n \geq N$ . Dans notre formalisme ceci équivaut à déterminer les suites  $(a_n)$  pour lesquelles on a  $L(a_n) = 0$ ,  $\forall n \geq N$ . La structure du noyau s'obtient immédiatement à partir du choix de l'opérateur linéaire  $L$  : ceci a été fait dans [25] pour des opérateurs menant à des algorithmes d'extrapolation connus et dans [29] pour la composition d'opérateurs ;
- la classe de suites accélérées par la transformation  $T$ , c'est-à-dire, les suites  $(S_n)$  pour lesquelles  $\lim_{n \rightarrow \infty} (T(S_n) - S_\infty)/(S_n - S_\infty) = 0$ . Dans notre formalisme, comme

$$\frac{T_n - S_\infty}{S_n - S_\infty} = \frac{L((S_n - S_\infty)/D_n)}{L(1/D_n)} \frac{1}{a_n D_n} = \frac{L(a_n)/a_n}{L(1/D_n)D_n},$$

l'ensemble des suites accélérées par  $T$  et que l'on notera par  $\mathcal{A}(T)$  est donné par

$$\mathcal{A}(T) = \left\{ (S_n) : S_n - S_\infty = a_n D_n \quad n \in \mathbb{N} \quad \text{et} \quad \lim_{n \rightarrow \infty} \frac{L(a_n)/a_n}{L(1/D_n)D_n} = 0 \right\}.$$

On dira que  $L$  est *un opérateur d'accélération pour*  $(S_n) \in \mathcal{A}(T)$ .

2) Le deuxième problème est le suivant : étant donné une classe  $\mathcal{C}$  de suites  $(S_n)$  vérifiant

$$S_n - S_\infty = a_n D_n, \quad \forall n \in \mathbb{N} \quad (1.4)$$

où  $(D_n)$  est une suite connue et  $(a_n)$  vérifie certaines propriétés, construire une transformation de suites de la forme (1.3) (ou, d'une façon équivalente, trouver un opérateur aux différences  $L$ ) pour laquelle

- (a)  $\forall (S_n) \in \mathcal{C}, (S_n) \in \text{Ker}(T)$  ;  
ou, dans le cas où cela n'est pas possible,

(b)  $\forall (S_n) \in \mathcal{C}, (S_n) \in \mathcal{A}(T)$ .

Etant donné  $(S_n)$  de la forme (1.4), à partir des propriétés de la suite  $(a_n)$  il est parfois possible, à partir des propriétés de la suite  $(a_n)$ , de trouver un opérateur aux différences d'annihilation pour  $(a_n)$  et donc de calculer la valeur exacte de  $S_\infty$ . Par exemple, si  $(a_n)$  est un polynôme de degré  $k-1$  en  $n$ , alors on peut choisir  $L = \Delta^k$ ; ou encore, si  $a_n = c_1 + c_2x_n + \dots + c_kx_n^{k-1}$ , et  $D_n = x_n$ ,  $n \in \mathbb{N}$  ( $(x_n)$  étant une suite donnée) alors  $L = \delta_k$  l'opérateur de différences divisées d'ordre  $k$  aux points  $x_i$  est un choix possible; d'autres choix sont considérés dans [25]. Mais ce sont les cas les plus simples.

Le cas général consiste à supposer que  $(a_n)$  admet un développement asymptotique dans une échelle de comparaison  $\{g_i(n)\}_{i=0}^\infty$ , i.e.,

$$\forall k \quad a_n = \sum_{i=0}^k \alpha_i g_i(n) + o(g_k(n)) \quad (n \rightarrow \infty),$$

où  $\{g_i(n)\}_{i=0}^\infty$  est une famille de suites vérifiant  $g_{i+1}(n) = o(g_i(n))$  ( $n \rightarrow \infty$ )  $\forall i$ , et  $\alpha_i$  constantes (non connues). Il n'est pas possible de trouver dans ce cas un opérateur aux différences d'annihilation pour  $(a_n)$  et en utilisant notre formalisme, on ne peut donc pas trouver une transformation de suites telle que  $\forall n \geq N \quad T(S_n) = S_\infty$ . Dans ce cas nous nous intéressons aux propriétés d'accélération de la transformation, c'est-à-dire, à la caractérisation des suites  $(S_n)$  pour lesquelles  $\lim_{n \rightarrow \infty} (L(a_n)/a_n)/(L(1/D_n)D_n) = 0$ .

Ces deux problèmes sont complémentaires et nous intéresseront tout le long de cette section. Comme on a interprété les transformations de suites en termes d'opérateurs aux différences, on utilisera des propriétés de ces opérateurs pour obtenir des résultats qui contribueront à la résolution des deux problèmes précédents.

Dans le cas du premier problème on considère plusieurs classes d'opérateurs aux différences  $L$  et, à partir des propriétés des équations aux différences associées  $L(u_n) = 0$ , on étudie, pour la transformation correspondante  $T$  (1.3) :

- les propriétés du noyau (en complétant les travaux exposés à la section 1.2);
- les propriétés d'accélération de  $T$ .

Il y a beaucoup d'équations linéaires aux différences pour lesquelles on ne dispose pas d'une forme compacte de la solution, mais on connaît son développement asymptotique (ou les premiers termes de ce développement). Concernant le deuxième problème, on utilise cette information pour construire une méthode d'extrapolation pour une suite  $(S_n)$  donnée pour laquelle on connaît les premiers termes du développement asymptotique de  $(a_n)$ , et on obtient la vitesse de convergence de la suite transformée.

Commençons par considérer les transformations de suites associées à un opérateur aux différences à coefficients constants.

### 1.3.1 Opérateurs aux différences à coefficients constants

On considère un opérateur aux différences d'ordre  $k$  de la forme

$$L^{(k)}(u_n) = \sum_{i=0}^k p_i^{(k)} u_{n+i}, \tag{1.5}$$

où les quantités  $p_i^{(k)}$  sont indépendantes de  $n$ . Son polynôme caractéristique étant donné par

$$P_k(x) = \sum_{i=0}^k p_i^{(k)} x^i = (x - \lambda_1)^{s_1} \cdots (x - \lambda_m)^{s_m}, \quad (1.6)$$

alors on sait (voir, par exemple [73]) qu'un ensemble de  $k$  solutions linéairement indépendantes de l'équation aux différences associée  $L^{(k)}(u_n) = 0$  est

$$\forall n \in \mathbb{N} \quad u_n^{(i,j)} = n^i \lambda_j^n, \text{ for } 0 \leq i \leq s_j - 1, \quad 1 \leq j \leq m.$$

On définit

$$\mathcal{S}^* = \{(D_n) \in \mathcal{S} : \forall n \in \mathbb{N} \ D_n \neq 0\}.$$

Soit  $(S_n) \in \mathcal{S}$  vérifiant (1.4) avec  $(D_n) \in \mathcal{S}^*$  et  $T^{(k)}$  la transformation de suites correspondante :

$$\left\{ \begin{array}{l} T^{(k)} : \mathcal{S} \longrightarrow \mathcal{S} \\ S = (S_n) \longmapsto T^{(k)}(S) = (T^{(k)}(S_n))_n = (T_n^{(k)})_n : T_n^{(k)} = \frac{L^{(k)}(S_n/D_n)}{L^{(k)}(1/D_n)}, \quad n \in \mathbb{N} \end{array} \right. \quad (1.7)$$

On va décrire le noyau et obtenir les propriétés d'accélération de ces transformations à partir des propriétés des solutions des équations aux différences.

### (i) Noyau

**PROPOSITION 1.7. [A9]** *Soit  $L^{(k)}$  un opérateur de la forme (1.5) avec un polynôme caractéristique  $P_k(x)$  donné par (1.6),  $(D_n) \in \mathcal{S}^*$  et  $T^{(k)}$  la transformation correspondante (1.7). Alors*

$$(S_n) \in \text{Ker}(T^{(k)}) \Leftrightarrow \text{pour } n \in \mathbb{N}, \quad S_n - S_\infty = a_n D_n \quad \text{avec } a_n = \sum_{i=1}^m q_i(n) \lambda_i^n \text{ et}$$

$q_i(n) \in \Pi_{s_i-1}$  (l'ensemble des polynômes de degré inférieur ou égal à  $s_i-1$ ).

Ceci permet d'obtenir

**COROLLAIRE 1.8. [A9]** *Supposons que  $s_1 \leq s_2 \leq \cdots \leq s_m$  et considérons la suite  $(S_n)$  vérifiant :*

$$S_n = S_\infty + r_n \times \left[ \lambda_1^n \left( c_0^{(1)} + c_1^{(1)} \frac{1}{n} + \cdots + c_{s_1-1}^{(1)} \frac{1}{n^{s_1-1}} \right) + \cdots + \lambda_m^n \left( c_0^{(m)} + \cdots + c_{s_1-1}^{(m)} \frac{1}{n^{s_1-1}} \right) \right], \quad (1.8)$$

avec  $(r_n)$  une suite connue ( $(r_n) \in \mathcal{S}^*$ ) et  $c_i^{(j)}$  des constantes (inconnues). On choisit  $D_n = n^{-s_1+1} r_n$  pour  $n \in \mathbb{N}$ . Si  $L^{(k)}(1/D_n) \neq 0$  pour  $n \in \mathbb{N}$ , alors en appliquant à  $(S_n)$  la transformation  $T^{(k)}$  donnée par (1.7), on obtient

$$\forall n \in \mathbb{N} \quad T_n^{(k)} = S_\infty \quad .$$

**(ii) Propriétés d'accélération**

L'erreur des suites transformées (1.7), i.e. la différence entre la valeur de  $S_\infty$  (que l'on veut calculer) et son approximation  $T_n^{(k)}$  peut s'écrire sous la forme

$$T_n^{(k)} - S_\infty = \frac{L^{(k)}((S_n - S_\infty)/D_n)}{L^{(k)}(1/D_n)} = \frac{L^{(k)}(a_n)}{L^{(k)}(1/D_n)} \quad n \in \mathbb{N}. \quad (1.9)$$

Pour obtenir la vitesse de convergence de cette suite on doit considérer deux cas distincts.

$$(a) \quad \boxed{\lim_{n \rightarrow \infty} D_n/D_{n+1} = \sigma \text{ et } P_k(\sigma) \neq 0}$$

Dans ce cas on obtient  $\lim_{n \rightarrow \infty} D_n L^{(k)}(1/D_n) = P_k(\sigma) \neq 0$  et donc

$$\lim_{n \rightarrow \infty} (T_n^{(k)} - S_\infty) = \frac{1}{P_k(\sigma)} \lim_{n \rightarrow \infty} D_n L^{(k)}(a_n).$$

On obtiendra accélération de la convergence si

$$\lim_{n \rightarrow \infty} \frac{L^{(k)}(a_n)}{a_n} = 0. \quad (1.10)$$

On peut alors démontrer le résultat suivant

**THÉORÈME 1.9.** [A9] *Soit  $(S_n) \in \mathcal{S}$  vérifiant :*

$$S_n - S_\infty = r_n \times \left[ \lambda_1^n \sum_{i=0}^{\infty} c_i^{(1)} \frac{1}{n^i} + \cdots + \lambda_m^n \sum_{i=0}^{\infty} c_i^{(m)} \frac{1}{n^i} \right], \quad (1.11)$$

avec

- $|\lambda_1| = |\lambda_2| = \cdots = |\lambda_m| = 1$ ;
- $\lim_{n \rightarrow \infty} r_n/r_{n+1} = \sigma$  et  $P_k(\sigma) \neq 0$ .

Si dans (1.7) l'opérateur  $L^{(k)}$  vérifie (1.5) et (1.6) avec  $s_1 \leq s_2 \leq \cdots \leq s_m$ , et si l'on choisit  $D_n = n^{-s_1+1} r_n$ , on obtient

$$\left| T_n^{(k)} - S_\infty \right| = O(n^{-2s_1} |r_n|) \quad (n \rightarrow \infty).$$

$$(b) \quad \boxed{\lim_{n \rightarrow \infty} D_n/D_{n+1} = \sigma \text{ et } P_k(\sigma) = 0}$$

Ce cas est moins simple que le précédent car, comme  $\lim_{n \rightarrow \infty} D_n L^{(k)}(1/D_n) = 0$ , (1.10) n'est plus une condition suffisante d'accélération. On doit donc imposer des conditions supplémentaires sur  $(D_n)$  et sur le polynôme caractéristique  $P_k$  pour obtenir l'ordre de convergence de  $(D_n L^{(k)}(1/D_n))$ .

**PROPOSITION 1.10.** [A9] *Supposons que la suite  $(D_n)$  vérifie :*

$$\frac{D_n}{D_{n+1}} = \sigma + \sigma_n \quad \text{avec } P_k(\sigma) = p_0^{(k)} + p_1^{(k)} \sigma + \cdots + p_k^{(k)} \sigma^k = 0, \quad \lim_{n \rightarrow \infty} \sigma_n = 0.$$

De plus, supposons que l'une des conditions suivantes est satisfaite :



a)  $\lim_{n \rightarrow \infty} \sigma_{n+1}/\sigma_n = 1$  et  $\sigma$  est une racine simple de  $P_k$  ;

ou

b)  $\lim_{n \rightarrow \infty} \sigma_{n+1}/\sigma_n = \sigma_* \neq 1$  et  $q_{k-1}(\sigma) - \sigma_* q_{k-1}(\sigma\sigma_*) \neq 0$

(avec  $q_{k-1}(x) = p_1^{(k)} + p_2^{(k)}x + \dots + p_k^{(k)}x^{k-1}$ ).

Alors l'opérateur aux différences  $L^{(k)}$  avec polynôme caractéristique  $P_k$  vérifie

$$L^{(k)}\left(\frac{1}{D_n}\right) \sim C \frac{\sigma_n}{D_n} \quad (n \rightarrow \infty) \quad (C \neq 0). \quad (1.12)$$

À partir de ce résultat on obtient facilement :

THÉORÈME 1.11. [A9] Soit  $(S_n)$  une suite de la forme  $S_n - S_\infty = a_n D_n$ ,  $n \in \mathbb{N}$  où

-  $(D_n)$  satisfait les conditions de la proposition précédente ;

-  $a_n = n^{s_1-1} \left( \sum_{i=0}^{\infty} c_i^{(1)} \frac{\lambda_1^n}{n^i} + \dots + \sum_{i=0}^{\infty} c_i^{(m)} \frac{\lambda_m^n}{n^i} \right)$  pour  $n \in \mathbb{N}$  ;

-  $|\lambda_1| = |\lambda_2| = \dots = |\lambda_m|$ .

Alors la suite  $(T_n^{(k)})_n$  donnée par (1.7), où l'opérateur  $L^{(k)}$  est donné par (1.5) et (1.6), vérifie :

$$\frac{T_n^{(k)} - S_\infty}{S_n - S_\infty} \sim \frac{C}{n^{2s_1} \sigma_n} \quad (n \rightarrow \infty),$$

et donc

$$(S_n) \in \mathcal{A}(T) \text{ ssi } \lim_{n \rightarrow \infty} n^{2s_1} \sigma_n = \infty.$$

Les démonstrations étant un peu longues et techniques, sont omises et pourront être consultées dans [A9].

### 1.3.2 Quelques classes d'opérateurs aux différences généraux

On va maintenant étudier la structure du noyau et les propriétés d'accélération des méthodes d'extrapolation correspondant à des opérateurs aux différences dont les coefficients dépendent de  $n$ . Comme dans le cas précédent on omet les démonstrations, renvoyant le lecteur à [A9].

#### (i) Opérateurs linéaires de premier ordre

Soient  $p(n)$  et  $q(n)$  deux polynômes de degré  $r$  et  $s$  respectivement. On peut donc les écrire sous la forme

$$\begin{cases} p(n) &= a \prod_{i=1}^r (n - \alpha_i), & \alpha_i \in \mathbb{R}, \\ q(n) &= b \prod_{j=1}^s (n - \beta_j), & \beta_j \in \mathbb{R}. \end{cases}$$

On pose

$$\alpha = \max_{1 \leq i \leq r} \alpha_i, \beta = \max_{1 \leq j \leq s} \beta_j, \text{ et } N \in \mathbb{N} \text{ tels que } N - 1 \leq \max(\alpha, \beta) < N. \quad (1.13)$$

On considère l'opérateur linéaire de la forme

$$L(u_n) = u_{n+1} - \frac{p(n)}{q(n)}u_n, \text{ pour } n \geq N. \quad (1.14)$$

La solution de l'équation aux différences  $L(u_n) = 0$ ,  $n \geq N$  est donnée dans [77]

$$u_n = \left(\frac{a}{b}\right)^n \prod_{i=1}^r \Gamma(n - \alpha_i) / \prod_{j=1}^s \Gamma(n - \beta_j), n \geq N,$$

ce qui nous permet d'obtenir immédiatement le noyau de la transformation de suites correspondante.

**THÉORÈME 1.12. [A9]** *Soit  $(D_n) \in \mathcal{S}^*$  et  $L$  un opérateur linéaire de la forme (1.14). Alors le noyau de la transformation correspondante  $T$  est donné par*

$$\text{Ker}(T) = \{(S_n) \in \mathcal{S} : S_n - S_\infty = a_n D_n, n \geq N \text{ avec } a_n \text{ vérifiant}$$

$$a_n = \left(\frac{a}{b}\right)^n \prod_{i=1}^r \Gamma(n - \alpha_i) / \prod_{j=1}^s \Gamma(n - \beta_j)$$

$$\text{avec } r, s, a, b, \alpha_i, \beta_j \in \mathbb{R} \text{ et } N \text{ définis par (1.13)}\}$$

Comme on peut écrire

$$\frac{T_n - S_\infty}{S_n - S_\infty} = \frac{L(a_n)}{a_n} \frac{1/D_n}{L(1/D_n)} = \frac{\frac{a_{n+1}}{a_n} - \frac{p(n)}{q(n)}}{\frac{D_n}{D_{n+1}} - \frac{p(n)}{q(n)}} \quad (1.15)$$

on obtient le résultat d'accélération suivant :

**THÉORÈME 1.13. [A9]** *Soit  $(S_n)$  une suite vérifiant  $S_n - S_\infty = a_n D_n$ ,  $n \geq N$ , avec  $(D_n) \in \mathcal{S}^*$  et  $\lim_{n \rightarrow \infty} a_{n+1}/a_n = 1 \neq \lim_{n \rightarrow \infty} D_n/D_{n+1}$ . On considère la transformation*

$$T_n = \frac{L(S_n/D_n)}{L(1/D_n)} \quad n \geq N, \quad (1.16)$$

avec  $L$  vérifiant (1.14). Alors, si  $\lim_{n \rightarrow \infty} p(n)/q(n) = 1$  on obtient

$$\lim_{n \rightarrow \infty} (T_n - S_\infty)/(S_n - S_\infty) = 0,$$

c'est-à-dire, si  $S_\infty$  est fini, alors  $(S_n) \in \mathcal{A}(T)$ .

De (1.15) on peut conclure que l'on obtiendra de bonnes propriétés d'accélération pour  $(T_n)$  si on peut fournir une bonne approximation de  $(a_{n+1}/a_n)$  par une fonction rationnelle de  $n$ ,  $(p(n)/q(n))$ . En effet,

THÉORÈME 1.14. [A9] Soit  $(S_n)$  une suite vérifiant  $S_n - S_\infty = a_n D_n$ ,  $n \geq N$  où

a)  $(a_n)$  vérifie :

$$a_{n+1}/a_n \sim \sum_{i=0}^{\infty} d_i n^{-i} \text{ où } d_i, \quad i = 0, \dots, 2k \text{ sont connus ;}$$

b)  $(D_n)$  vérifie :

$$D_n/D_{n+1} \sim \sum_{i=0}^{\infty} e_i n^{-i} \text{ avec } e_i = d_i, \quad i = 0, \dots, l-1; \quad e_l \neq d_l \quad (l < 2k).$$

Soit  $[k/k]$  l'approximant de Padé [11] de la fonction  $f(x) = \sum_{i=0}^{\infty} d_i x^i$ ,

$$[k/k]_f(x) = \frac{P(x)}{Q(x)}.$$

On considère l'opérateur  $L$  de la forme (1.14) avec  $r = s = k$  et

$$\begin{aligned} p(n) &= P(1/n)n^k, \\ q(n) &= Q(1/n)n^k. \end{aligned}$$

Alors la transformation de suites correspondante définie par (1.16) accélère la convergence de  $(S_n)$ . De plus, la vitesse de convergence est donnée par

$$\frac{T_n - S_\infty}{S_n - S_\infty} = O(n^{l-2k-1}) \quad (n \rightarrow \infty).$$

Si  $(D_n/D_{n+1})$  n'a pas un développement asymptotique en puissances de  $(1/n)$  mais vérifie

$$D_n/D_{n+1} = e_0 + \sum_{i=0}^l e_i g_i(n) + o(g_l(n)) \quad (g_{i+1}(n) = o(g_i(n)) \quad (n \rightarrow \infty))$$

(par exemple, si  $D_n = n^\alpha (\log n)^\beta$ , alors  $D_n/D_{n+1} = 1 + c_1/n + c_2/(n \log n) + o(n^{-1}(\log n)^{-1})$ ), on peut encore déterminer le comportement asymptotique de  $(D_n/D_{n+1} - p(n)/q(n))$  et obtenir de nouveaux résultats d'accélération en utilisant le même type de techniques.

## (ii) Opérateurs aux différences à coefficients polynômiaux

Considérons maintenant des classes d'opérateurs aux différences de la forme

$$L(u_n) = \sum_{i=0}^l \lambda_i(n) u_{n+i}, \quad \text{où les } \lambda_i(n) \text{ sont des polynômes en } n,$$

pour lesquels on peut obtenir  $l$  solutions linéairement indépendantes de l'équation aux différences homogène  $L(u_n) = 0$ . On va donner, pour la transformation  $T$  correspondante, le noyau et les propriétés d'accélération. On montrera que certaines de ces méthodes d'extrapolation sont bien

adaptées à l'accélération de la convergence de suites logarithmiques (c'est-à-dire, des suites vérifiant  $\lim_{n \rightarrow \infty} (S_{n+1} - S_\infty)/(S_n - S_\infty) = 1$  et donc convergeant lentement) pour lesquelles on a un développement asymptotique de l'erreur en termes des suites  $\left\{ \frac{1}{n(n+1) \cdots (n+i)} \right\}_{i=0}^\infty$ . Pour simplifier les notations, on définit les opérateurs élémentaires suivants :

$$E(u_n) = u_{n+1}, \quad \Omega(u_n) = nu_{n+1}, \quad \pi(u_n) = n\Delta u_n.$$

La composition de ces opérateurs donne

$$E^r(u_n) = u_{n+r}, \quad \Omega^r(u_n) = n(n+1) \cdots (n+r-1)u_{n+r}, \quad \pi^r(u_n) = \pi(\pi^{r-1}(u_n)).$$

On va considérer deux classes d'opérateurs  $L$ .

a)  $L(u_n) = (\Omega^l + \lambda_1 \Omega^{l-1} + \cdots + \lambda_{l-1} \Omega + \lambda_l)(u_n), \quad \lambda_i \text{ donnés.}$

Si on calcule les racines du polynôme  $p_l(x) = x^l + \lambda_1 x^{l-1} + \cdots + \lambda_l$ , on peut écrire  $L$  comme composition d'opérateurs simples. On considère les deux cas suivants :

$$L_1(u_n) = (\Omega - \alpha_1)(\Omega - \alpha_2) \cdots (\Omega - \alpha_l)(u_n) \text{ avec } \alpha_i \neq \alpha_j \quad \forall i \neq j; \quad (1.17)$$

$$L_2(u_n) = (\Omega - \alpha)^l(u_n). \quad (1.18)$$

Pour le cas général  $L(u_n) = (\Omega - \alpha_1)^{s_1}(\Omega - \alpha_2)^{s_2} \cdots (\Omega - \alpha_l)^{s_l}(u_n)$ , les résultats s'obtiennent facilement à partir de ceux pour  $L_1$  et  $L_2$ .

A partir de la théorie des opérateurs linéaires aux différences (voir, par exemple [7]) on obtient

$$L_1(u_n) = 0 \Leftrightarrow u_n = \frac{1}{(n-1)!} \sum_{i=1}^l A_i \alpha_i^n, \quad A_i \text{ constantes};$$

$$L_2(u_n) = 0 \Leftrightarrow u_n = \frac{\alpha^n}{(n-1)!} \sum_{i=0}^{l-1} A_i n^i, \quad A_i \text{ constantes.}$$

La structure du noyau de la transformation  $T$  correspondante est alors donnée par :

THÉORÈME 1.15. [A9] *On considère les transformations de suites*

$$T_n^{(i)} = \frac{L_i(S_n/D_n)}{L_i(1/D_n)}, \quad n \in \mathbb{N}, \quad i = 1, 2, \quad (1.19)$$

avec  $L_i, i = 1, 2$  donnés par (1.17) et (1.18). Alors

$$a) (S_n) \in \text{Ker}(T^{(1)}) \Leftrightarrow S_n - S_\infty = \frac{D_n}{(n-1)!} \sum_{i=1}^l A_i \alpha_i^n \text{ pour } n \in \mathbb{N};$$

$$b) (S_n) \in \text{Ker}(T^{(2)}) \Leftrightarrow S_n - S_\infty = \frac{D_n \alpha^n}{(n-1)!} \sum_{i=0}^{l-1} A_i n^i \text{ pour } n \in \mathbb{N}.$$

Pour obtenir les propriétés d'accélération de ces transformations, on a besoin de déterminer le comportement asymptotique de suites  $(L_i(u_n))$   $i = 1, 2$  à partir de quelques propriétés de la suite  $(u_n)$ . En étudiant ce comportement pour des suites  $(u_n)$  vérifiant :

- $\lim_{n \rightarrow \infty} n \frac{u_{n+1}}{u_n} = \beta, \quad \beta \neq \alpha_i$
  - $\lim_{n \rightarrow \infty} \frac{u_{n+1}}{u_n} = C \neq 0$
  - $u_n = \frac{\alpha^n}{(n-1)!} \frac{1}{n^i}$
  - $u_n = \frac{1}{(n-1)!} \sum_{i=1}^m A_i \alpha_i^n$  avec  $|\alpha_1| > |\alpha_2| > \dots > |\alpha_m|$ ,
- on peut montrer les résultats suivants

THÉORÈME 1.16. [A9] Soit  $(S_n)$  une suite vérifiant

$$S_n - S_\infty = r_n \sum_{i=0}^{\infty} A_i \frac{\alpha_i^n}{(n-1)!} \quad n \in \mathbb{N}, \text{ avec } |\alpha_1| > |\alpha_2| > \dots$$

On considère la transformation de suites  $T^{(1)}$  donnée par (1.19) avec  $L_1$  défini par (1.17) et  $D_n = r_n$  pour  $n \in \mathbb{N}$ . Alors

$$\frac{T_n^{(1)} - S_\infty}{S_n - S_\infty} = \begin{cases} O\left(\frac{\alpha_{l+1}^n}{n^l(n-1)!}\right) & \text{si } \lim_{n \rightarrow \infty} \frac{r_n}{r_{n+1}} = c \neq 0 \\ O\left(\frac{\alpha_{l+1}^n}{(n-1)!}\right) & \text{si } \lim_{n \rightarrow \infty} \frac{nr_n}{r_{n+1}} = \beta \neq \alpha_i \quad i = 1, \dots, l. \end{cases}$$

THÉORÈME 1.17. [A9] Soit  $(S_n)$  une suite vérifiant :

$$S_n - S_\infty = \frac{r_n \alpha^n}{(n-1)!} \sum_{i=0}^{\infty} A_i \frac{1}{n^i} \quad \text{avec } \lim_{n \rightarrow \infty} \frac{r_n}{r_{n+1}} = r \neq 0$$

Si on applique à  $(S_n)$  la transformation de suites  $T^{(2)}$  donnée par (1.19) où  $L_2$  a la forme (1.18) et  $D_n = r_n/n^l$ ,  $n \in \mathbb{N}$ , alors  $(T_n^{(2)})$  converge vers  $S_\infty$  plus vite que  $(S_n)$ . De plus, la vitesse de convergence est donnée par

$$\frac{T_n^{(2)} - S_\infty}{S_n - S_\infty} = O\left(\frac{1}{n^{2l+2}}\right) \quad (n \rightarrow \infty).$$

Considérons maintenant une autre classe d'opérateurs :

b)  $L(u_n) = (\pi - \alpha_1)(\pi - \alpha_2) \cdots (\pi - \alpha_l)(u_n) \quad \alpha_i \text{ constantes}$

On peut montrer [77] que les suites  $(u_n)$  pour lesquelles  $L(u_n) = 0$  sont de la forme

$$u_n = \frac{1}{(n-1)!} \sum_{i=1}^l A_i \Gamma(n + \alpha_i),$$

( $\Gamma(x)$  est la fonction Gamma) ce qui nous permet d'obtenir le noyau de la transformation  $T$ .

THÉORÈME 1.18. [A9] Soit  $T$  la transformation de suites correspondant à un opérateur  $L$  de la forme

$$L(u_n) = (\pi - \alpha_1)(\pi - \alpha_2) \cdots (\pi - \alpha_l)(u_n), \quad \alpha_i \text{ constantes données.} \quad (1.20)$$

Alors le noyau de  $T$  est donné par

$$\text{Ker}(T) = \left\{ (S_n) : S_n - S_\infty = \frac{D_n}{(n-1)!} \sum_{i=1}^l A_i \Gamma(n + \alpha_i) \right\}.$$

Ces opérateurs  $L$ , aussi appelés *opérateurs aux différences d'Euler*, peuvent s'écrire sous la forme plus explicite suivante :

$$L(u_n) = \frac{\Gamma(n+l)}{\Gamma(n)} \Delta^l u_n + p_{l-1} \frac{\Gamma(n+l-1)}{\Gamma(n)} \Delta^{l-1} u_n + \cdots + p_1 n \Delta u_n + p_0 u_n,$$

la relation entre les  $p_i$  et les  $\alpha_i$  étant :  $\alpha_1, \alpha_2, \dots, \alpha_r$  sont les racines du polynôme

$$p_*(x) = x(x-1) \cdots (x-r+1) + p_{r-1} x(x-1) \cdots (x-r+2) + \cdots + p_1 x + p_0.$$

Comme dans le cas précédent, on étudie le comportement asymptotique des suites  $(L(u_n))$  pour

$$\begin{aligned} - \frac{u_{n+1}}{u_n} &= 1 + \frac{C}{n} + \alpha_n, \quad \alpha_n \sim \frac{C_1}{n^2} \quad (n \rightarrow \infty) \\ - \frac{u_{n+1}}{u_n} &= \rho \neq 1, \end{aligned}$$

et on obtient les résultats d'accélération suivants :

**THÉORÈME 1.19. [A9]** *Soit  $(S_n)$  une suite de la forme :*

$$S_n - S_\infty = D_n \left( A_1 \frac{\Gamma(n+\alpha_1)}{(n-1)!} + \cdots + A_r \frac{\Gamma(n+\alpha_r)}{(n-1)!} + \cdots \right) \quad n \in \mathbb{N},$$

avec

$$a) \lim_{n \rightarrow \infty} \frac{\Gamma(n+\alpha_i)}{\Gamma(n+\alpha_{i-1})} = 0 \quad \forall i;$$

$$b) (D_n) \text{ vérifie : } \lim_{n \rightarrow \infty} \left( \frac{D_n}{D_{n+1}} - 1 \right) n \neq \alpha_i \quad i = 1, \dots, r.$$

Si on applique à  $(S_n)$  la transformation de suites  $T^{(r)}$  définie à partir de  $L^{(r)}(u_n) = (\pi - \alpha_1) \cdots (\pi - \alpha_r)(u_n)$ , on obtient

$$\frac{T_n^{(r)} - S_\infty}{S_n - S_\infty} \sim C \frac{\Gamma(n+\alpha_{r+1})}{\Gamma(n+\alpha_1)} \quad (n \rightarrow \infty).$$

**THÉORÈME 1.20. [A9]** *Soit  $(S_n)$  une suite ayant un développement asymptotique de l'erreur de la forme :*

$$S_n - S_\infty = \frac{C_1}{n} + \frac{C_2}{n(n+1)} + \cdots + \frac{C_r}{n(n+1) \cdots (n+r-1)} + \cdots \quad n \in \mathbb{N}.$$

On considère la transformation de suites  $T^{(r)}$  définie par

$$a) D_n = \frac{1}{n(n+1) \cdots (n+r-1)} \text{ pour } n \in \mathbb{N};$$

$$b) L^{(r)}(u_n) = \pi(\pi-1) \cdots (\pi-(r-1))(u_n) \quad (\text{i.e., } \alpha_i = i, \quad i = 0, \dots, r-1).$$

Alors la suite  $(T_n^{(r)} = T^{(r)}(S_n))$  converge vers  $S_\infty$  plus vite que  $(S_n)$ . De plus, on obtient :

$$\frac{T_n^{(r)} - S_\infty}{S_n - S_\infty} = O\left(\frac{1}{n^r}\right) \quad (n \rightarrow \infty).$$

Pour la démonstration de ces résultats voir [A9].

De ces deux derniers théorèmes on conclut que les méthodes d'extrapolation correspondantes aux opérateurs de la forme (1.20) ont de bonnes propriétés d'accélération quand elles sont appliquées à certaines classes de suites à convergence logarithmique.

### 1.3.3 Propriétés d'accélération de $T$ basées sur le comportement asymptotique des solutions des équations aux différences linéaires

Pour les deux classes d'opérateurs  $L$  considérées dans la section précédente, on disposait d'une base de solutions (sous forme explicite) pour l'équation aux différences linéaire homogène associée  $L(u_n) = 0$ , et ceci nous a permis d'étudier les propriétés d'accélération de la méthode d'extrapolation  $T$ . Mais pour un opérateur général  $L$  d'ordre  $k$  et coefficients non constants, il n'y a pas de solution générale sous forme compacte. On peut, néanmoins, pour un opérateur  $L$  donné, trouver un ensemble indépendant de solutions pour  $L(u_n) = 0$  en utilisant des méthodes de calcul aux différences (voir par exemple [1, 73]) : réduction d'ordre, fonctions génératrices, etc. Et, à partir d'une base de solutions, en utilisant les mêmes techniques que dans la section précédente, trouver le noyau et déterminer les propriétés d'accélération de la transformation  $T$ .

Mais si l'on ne peut pas trouver une base de solutions, on peut utiliser des techniques différentes pour obtenir le développement asymptotique d'un ensemble linéairement indépendant de solutions pour certaines classes d'opérateurs [7]. À partir de là on pourra :

- donner les propriétés d'accélération de la transformation  $T$  associée à  $L$  ;
- proposer une méthode de construction d'un opérateur  $L$  et de la transformation correspondante  $T$  pour accélérer la convergence d'une classe de suites  $(S_n)$  pour lesquelles l'erreur  $(S_n - S_\infty)$  possède un développement asymptotique donné.

C'est ce que l'on va présenter maintenant, en considérant une classe d'opérateurs  $\mathcal{L}$  définie par

DEFINITION 1.21.  $L \in \mathcal{L}$  ssi

$$L(u_n) = \Delta^k u_n + p_{k-1}(n)\Delta^{k-1}u_n + \cdots + p_1(n)\Delta u_n + p_0(n)u_n, \quad (1.21)$$

où les  $p_i(n)$ ,  $i = 0, \dots, k-1$  vérifient la condition suivante :

les fonctions  $f_i$  définies par  $f_i(t) = p_i(1/t)t^{-k+i}$ ,  $i = 0, \dots, k-1$  sont analytiques dans un voisinage de 0. Donc les  $p_i(n)$  admettent le développement asymptotique suivant :

$$p_i(n) = \frac{1}{n^{k-i}} \left( C_0^{(i)} + \frac{C_1^{(i)}}{n} + \cdots \right) \quad i = 0, \dots, k-1. \quad (1.22)$$

DEFINITION 1.22. La suite  $(u_n)$  vérifie la propriété  $\mathcal{P}$  ssi

$$\frac{u_{n+1}}{u_n} = 1 + \frac{\alpha}{n} + r_n \quad \text{avec } r_n = o\left(\frac{1}{n}\right) \quad (n \rightarrow \infty), \quad \Delta^i r_n = o\left(\frac{1}{n^{i+1}}\right) \quad (n \rightarrow \infty). \quad (1.23)$$

Un exemple de suite  $(u_n)$  vérifiant  $\mathcal{P}$  est une suite qui admet le développement suivant :

$$\frac{u_{n+1}}{u_n} = 1 + \frac{\alpha_1}{n} + \frac{\alpha_2}{n^2} + \cdots \quad (n \rightarrow \infty)$$

On peut montrer que si  $L \in \mathcal{L}$  et  $(u_n)$  vérifie  $\mathcal{P}$  alors

$$L(u_n) \sim \frac{A_k}{n^k} u_n \quad (n \rightarrow \infty) \quad \text{avec } A_k = \sum_{i=0}^k \alpha^{(i)} C_0^{(i)},$$

où  $\alpha^{(0)} = 1$ ,  $\alpha^{(i)} = \prod_{j=0}^{i-1} (\alpha - j)$ ;  $C_0^{(k)} = 1$ ;  $C_0^{(i)}$  donnés par (1.21).

Soit  $L \in \mathcal{L}$  et supposons que  $(u_n)$  est une solution de  $L(u_n) = 0$  pour laquelle on connaît les premiers termes du développement asymptotique :

$$u_n = c_1 g_1(n) + c_2 g_2(n) + r_n \quad n \in \mathbb{N}, \quad g_2(n) = o(g_1(n)), r_n = o(g_2(n)) \quad (n \rightarrow \infty), \quad (1.24)$$

avec  $(g_i(n))$ ,  $i = 1, 2$ ,  $(r_n)$  vérifiant  $\mathcal{P}$  et  $c_1 \neq 0$ . On peut alors montrer que

$$L(g_1(n)) \sim C n^{-k} g_2(n) \quad (n \rightarrow \infty), \quad (1.25)$$

ce qui permet d'obtenir le résultat d'accélération suivant :

**THÉORÈME 1.23. [A9]** *Soit  $(S_n)$  une suite vérifiant  $S_n - S_\infty = a_n D_n$  avec*

$$a_n = b_1 g_1(n) + \rho_n, \quad \rho_n = o(g_1(n)) \quad (n \rightarrow \infty), \quad (g_1(n)), (\rho_n) \text{ vérifiant } \mathcal{P}.$$

*Soit  $L$  un opérateur dans  $\mathcal{L}$  pour lequel on connaît le développement asymptotique d'une solution de  $L(u_n) = 0$  vérifiant*

$$u_n \sim \sum_{i=0}^{\infty} \alpha_i g_i(n), \quad \text{avec } (g_i(n)) \quad i = 1, \dots \quad \text{vérifiant la propriété } \mathcal{P}.$$

*On suppose que  $\rho_n \sim K g_2(n)$   $(n \rightarrow \infty)$  et on considère la transformation  $T$  correspondant à cet opérateur :*

$$T_n = L(S_n/D_n)/L(1/D_n). \quad (1.26)$$

a) *Si  $\lim_{n \rightarrow \infty} D_n/D_{n+1} = \lambda \neq 1$  alors*

$$\frac{T_n - S_\infty}{S_n - S_\infty} \sim C n^{-k} \frac{g_2(n)}{g_1(n)} \quad (n \rightarrow \infty) \quad (C \text{ constante });$$

b) *Si  $\lim_{n \rightarrow \infty} D_n/D_{n+1} = 1$  et  $(1/D_n)$  vérifie  $\mathcal{P}$  avec  $A_k \neq 0$  alors*

$$\frac{T_n - S_\infty}{S_n - S_\infty} \sim C \frac{g_2(n)}{g_1(n)} \quad (n \rightarrow \infty) \quad (C \text{ constante}).$$

Ce résultat peut être généralisé au cas où l'on connaît  $k$  solutions linéairement indépendantes de  $L(u_n) = 0$ .

**THÉORÈME 1.24. [A9]** *Soit  $(S_n)$  une suite vérifiant*

$$S_n - S_\infty = D_n \left( a_1 g_1^{(1)}(n) + a_2 g_1^{(2)}(n) + \dots + a_k g_1^{(k)}(n) + \rho_n \right),$$

*avec  $g_1^{(i+1)}(n) = o(g_1^{(i)}(n))$ ,  $\rho_n = o(g_1^{(k)}(n))$   $(n \rightarrow \infty)$ .*

*On considère un opérateur  $L \in \mathcal{L}$  pour lequel on connaît une base de solutions  $(u_n^{(i)})$   $i = 1, \dots, k$ , chacune s'écrivant*

$$u_n^{(i)} \sim \sum_{j=1}^{\infty} \alpha_j^{(i)} g_j^{(i)}(n), \quad g_{j+1}^{(i)}(n) = o(g_j^{(i)}(n)) \quad (n \rightarrow \infty) \quad \forall j \in \mathbb{N}, j = 1, \dots, k.$$

*On suppose que :*



- (a)  $g_2^{(i+1)}(n) = o(g_2^{(i)}(n)) \quad (n \rightarrow \infty), \quad i = 1, \dots, k-1;$   
 (b)  $g_2^{(1)}(n) = o(g_1^{(k)}(n)) \quad (n \rightarrow \infty); \quad \rho_n \sim K g_2^{(1)}(n) \quad (n \rightarrow \infty);$   
 (c)  $(g_j^{(i)}(n)) \quad i = 1, \dots, k, \quad j = 1, 2, \dots$  vérifient  $\mathcal{P}$ .

Alors

1. si  $(D_n)$  vérifie  $\lim_{n \rightarrow \infty} D_n/D_{n+1} = \lambda \neq 1$ , la transformation de suites (1.26) accélère la convergence de  $(S_n)$ . De plus, cette accélération est donnée par :

$$\frac{T_n - S_\infty}{S_n - S_\infty} \sim C n^{-k} \frac{g_2^{(1)}(n)}{g_1^{(1)}(n)} \quad (n \rightarrow \infty).$$

2. si  $(1/D_n)$  vérifie  $\mathcal{P}$ , la vitesse de convergence de  $(T_n)$  est donnée par

$$\frac{T_n - S_\infty}{S_n - S_\infty} \sim C \frac{g_2^{(1)}(n)}{g_1^{(1)}(n)} \quad (n \rightarrow \infty).$$

Pour pouvoir déterminer la classe de suites  $(S_n)$  qui peut être accélérée par la méthode d'extrapolation (1.26) correspondante à un opérateur  $L \in \mathcal{L}$  ou pour pouvoir construire un opérateur pour lequel  $(T_n)$  accélère la convergence d'une classe de suites  $(S_n)$ , on a besoin d'obtenir le comportement asymptotique des solutions de  $L(u_n) = 0$ . Cela est fait en détail dans [A9], basé sur les idées développées dans [7]. En utilisant des techniques l'analyse asymptotique locale on obtient

THÉORÈME 1.25. Soit  $L \in \mathcal{L}$  et  $(u_n^{(i)})_n$  une solution de  $L(u_n) = 0$

1. si  $(\alpha_i - \alpha_j)$  ne sont pas entiers, pour  $i \neq j$ , alors on a  $k$  solutions linéairement indépendantes qui admettent le développement en série de Frobenius

$$u_n^{(i)} \sim \sum_{k=0}^{\infty} A_k^{(i)} \frac{\Gamma(n)}{\Gamma(n+k-\alpha_i)} \quad (n \rightarrow \infty).$$

Comme  $\frac{\Gamma(n)}{\Gamma(n+k-\alpha)} \sim n^{\alpha-k} \quad (n \rightarrow \infty)$  on a  $u_n^{(i)} \sim C_i n^{\alpha_i} \quad (n \rightarrow \infty) \quad i = 1, \dots, k$ .

2. Soit  $\Psi(n) = \Gamma'(n)/\Gamma(n)$  la fonction Digamma (qui possède le développement asymptotique suivant :  $\Psi(n) \sim \log(n) + c + \sum_{i=1}^{\infty} b_i n^{-i} \quad (n \rightarrow \infty)$  )

- (a) si  $\alpha_i = \alpha_{i+1} = \dots = \alpha_{i+j}$  alors  $L(u_n) = 0$  admet  $j+1$  solutions linéairement indépendantes vérifiant

$$\begin{aligned} u_n^{(i)} &\sim \sum_{k=0}^{\infty} A_k^{(i)} \frac{\Gamma(n)}{\Gamma(n+k-\alpha_i)} \quad (n \rightarrow \infty) \\ u_n^{(i+1)} &\sim \sum_{k=0}^{\infty} A_k^{(i+1)} \frac{\Gamma(n)}{\Gamma(n+k-\alpha_i)} + \sum_{k=0}^{\infty} B_k^{(i)} \Psi(n) \frac{\Gamma(n)}{\Gamma(n+k-\alpha_i)} \\ &\dots \\ u_n^{(i+j)} &\sim \sum_{k=0}^{\infty} A_k^{(i+j)} \frac{\Gamma(n)}{\Gamma(n+k-\alpha_i)} + \dots + \sum_{k=0}^{\infty} B_k^{(i,j)} (\Psi(n))^j \frac{\Gamma(n)}{\Gamma(n+k-\alpha_i)} \end{aligned}$$

(b) si  $\alpha_{l+1} - \alpha_l = n_l$  entier,  $l = i, \dots, i+j$  alors  $L(u_n) = 0$  admet  $j+1$  solutions linéairement indépendantes vérifiant

$$\begin{aligned} u_n^{(i)} &\sim \sum_{k=0}^{\infty} A_k^{(i)} \frac{\Gamma(n)}{\Gamma(n+k-\alpha_i)}; \\ u_n^{(i+1)} &\sim \sum_{k=0}^{\infty} A_k^{(i+1)} \frac{\Gamma(n)}{\Gamma(n+k-\alpha_{i+1})} + \sum_{k=0}^{\infty} B_k^{(i,1)} \Psi(n) \frac{\Gamma(n)}{\Gamma(n+k-\alpha_i)}; \\ \dots &\dots \\ u_n^{(i+j)} &\sim \sum_{k=0}^{\infty} A_k^{(i+j)} \frac{\Gamma(n)}{\Gamma(n+k-\alpha_{i+j})} + \sum_{k=0}^{\infty} B_k^{(i,1)} \Psi(n) \frac{\Gamma(n)}{\Gamma(n+k-\alpha_{i+j-1})} + \dots + \\ &+ \sum_{k=0}^{\infty} B_k^{(i,j)} (\Psi(n))^j \frac{\Gamma(n)}{\Gamma(n+k-\alpha_i)}. \end{aligned}$$

Pour ces deux derniers cas on obtient le comportement asymptotique suivant

$$u_n^{(i+l)} \sim C_{i+l} (\log(n))^l n^{\alpha_i} \quad (n \rightarrow \infty) \quad l = 0, \dots, j.$$

**Exemples :** regardons maintenant deux exemples de construction d'une transformation de suites pour accélérer la convergence d'une famille de suites donnée, basés sur les techniques exposées ci-dessus.

(1) On considère la famille de suites  $(S_n)$  vérifiant

$$S_n - S_{\infty} = D_n \left( \frac{b_1}{n^{\alpha_1}} + \frac{b_2}{n^{\alpha_2}} + \dots + \frac{b_k}{n^{\alpha_k}} + \rho_n \right) \quad n \in \mathbb{N},$$

avec

$$0 < \alpha_1 < \alpha_2 < \dots < \alpha_k, \quad \alpha_k - \alpha_1 < 1, \quad \rho_n = o\left(\frac{1}{n^{\alpha_k}}\right) \quad (n \rightarrow \infty).$$

Pour construire un opérateur  $L \in \mathcal{L}$  qui a  $k$  solutions linéairement indépendantes vérifiant

$$u_n^{(i)} \sim \sum_{k=0}^{\infty} A_k^{(i)} \frac{\Gamma(n)}{\Gamma(n+k+\alpha_i)} \quad (n \rightarrow \infty), \quad i = 1, \dots, k,$$

on procède de la façon suivante :

– on considère le polynôme

$$P(\alpha) = (\alpha + \alpha_1) \cdots (\alpha + \alpha_k) = \alpha(\alpha - 1) \cdots (\alpha - k + 1) + a_{k-1} \alpha \cdots (\alpha - k + 2) + \dots + a_1 \alpha + a_0.$$

– on construit récursivement l'ensemble suivant de fonctions analytiques en 0 :

$$p_i^*(t) = (-1)^k q_i(t) + (-1)^k \sum_{j=i+1}^k d_{ij} (-1)^j p_j^*(t), \quad i = k-1, \dots, 0$$

avec

$$\begin{aligned} p_k^*(t) &= (-1)^k; \\ q_i(t) &\text{ fonctions analytiques en 0 vérifiant } q_i(0) = a_i; \\ d_{ij} &\text{ quantités définies dans [A9] .} \end{aligned}$$

– On pose

$$p_i(t) = (-1)^i t^{i-k} p_i^* \left( \frac{1}{t} \right), \quad i = 0, \dots, k,$$

et on considère l'opérateur  $L$  dans  $\mathcal{L}$  correspondant.

Si  $\lim_{n \rightarrow \infty} D_n/D_{n+1} \neq 1$  alors on peut montrer que les conditions du théorème 1.24 sont vérifiées et conclure que  $(T_n)$  accélère la convergence de  $(S_n)$ . De plus,

$$\frac{T_n - S_\infty}{S_n - S_\infty} \sim C n^{-k-1} \quad (n \rightarrow \infty) \text{ où } C \text{ est une constante.}$$

(2) On considère maintenant la famille de suites de la forme

$$S_n - S_\infty = D_n \left( b_1 \frac{(\log n)^l}{n^a} + b_2 \frac{(\log n)^{l-1}}{n^a} + \dots + b_{l+1} \frac{1}{n^a} + \rho_n \right) \quad n \in \mathbb{N},$$

$$\text{où } \rho_n = o\left(\frac{1}{n^a}\right) \quad (n \rightarrow \infty).$$

On considère le polynôme  $P(\alpha)$  dont les racines sont  $a, a+1, \dots, a+l-1$  et on construit l'opérateur  $L$  comme dans l'exemple précédent. On peut montrer que

$$\frac{T_n - S_\infty}{S_n - S_\infty} \sim C n^{-k-1} \quad (n \rightarrow \infty) \text{ où } C \text{ est une constante.}$$

Tous ces résultats montrent les voies qui ont été ouvertes par ce nouveau formalisme pour présenter les méthodes d'extrapolation. Il serait maintenant intéressant de faire l'étude numérique (notamment la stabilité) des nouvelles méthodes d'extrapolation proposées tout le long de ce chapitre et de comparer leurs performances. La prospection de problèmes appliqués où l'on rencontre des suites ayant un développement asymptotique du type considéré dans les théorèmes ci-dessus est en cours.

Un autre problème ouvert et qui serait intéressant à étudier est de déterminer les propriétés d'accélération de convergence de l'itération de la procédure

$$\left\{ \begin{array}{l} T: \mathcal{S} \quad \longrightarrow \quad \mathcal{S} \\ S = (S_n) \quad \longmapsto \quad T(S) = (T(S_n)) = (T_n) : \quad T_n = \frac{L(S_n/D_n)}{L(1/D_n)}, \quad n \in \mathbb{N}. \end{array} \right.$$

## 1.4 Nouvelle propriété d'accélération pour le E-algorithme

Pour finir ce chapitre, on va s'intéresser à l'un des algorithmes les plus généraux d'extrapolation et qui a aussi été interprété dans le formalisme des sections précédentes - le **E-algorithme**. On va obtenir une propriété d'accélération des colonnes de cet algorithme pour des familles de suites ayant un développement asymptotique de l'erreur dans une famille  $(g_i(n))_n, i \geq 1$  vérifiant une relation déterminantale. Ce résultat se généralise au cas de suites vectorielles.

Commençons par quelques rappels.

### 1.4.1 Définitions et rappels

Soit  $(S_n)_{n \in \mathbb{N}}$  une suite de nombres complexes vérifiant

$$S_n = S_\infty + a_1 g_1(n) + a_2 g_2(n) + \cdots + a_k g_k(n), \quad n \in \mathbb{N}, \quad (1.27)$$

avec les  $g_i$  des suites données. Si on veut calculer  $S_\infty$ , on écrit (1.27) pour l'indice variant de  $n$  à  $n+k$

$$S_{n+i} = S_\infty + a_1 g_1(n+i) + a_2 g_2(n+i) + \cdots + a_k g_k(n+i), \quad i = 0, \dots, k, \quad (1.28)$$

et, si le système (1.28) est régulier,  $S_\infty$  est donné par

$$S_\infty = \frac{\begin{vmatrix} S_n & g_1(n) & \cdots & g_k(n) \\ S_{n+1} & g_1(n+1) & \cdots & g_k(n+1) \\ \vdots & \vdots & \cdots & \vdots \\ S_{n+k} & g_1(n+k) & \cdots & g_k(n+k) \end{vmatrix}}{\begin{vmatrix} 1 & g_1(n) & \cdots & g_k(n) \\ 1 & g_1(n+1) & \cdots & g_k(n+1) \\ \vdots & \vdots & \cdots & \vdots \\ 1 & g_1(n+k) & \cdots & g_k(n+k) \end{vmatrix}}. \quad (1.29)$$

Pour simplifier les notations on identifiera chacun de ces déterminants par sa première ligne, i.e.

$$S_\infty = \frac{\begin{vmatrix} S_n & g_1(n) & \cdots & g_k(n) \end{vmatrix}}{\begin{vmatrix} 1 & g_1(n) & \cdots & g_k(n) \end{vmatrix}}.$$

Si la suite  $(S_n)_{n \in \mathbb{N}}$  n'a pas la forme exacte (1.27) alors la valeur de  $S_\infty$  donnée par (1.29) dépend de  $k$  et  $n$ , et on note

$$E_k(S_n) = \frac{\begin{vmatrix} S_n & g_1(n) & \cdots & g_k(n) \end{vmatrix}}{\begin{vmatrix} 1 & g_1(n) & \cdots & g_k(n) \end{vmatrix}}. \quad (1.30)$$

Le E-algorithme est un algorithme récursif proposé indépendamment par plusieurs auteurs mais les deux approches les plus générales sont dues à Brezinski dans [12] et Hävie dans [62]. Il permet le calcul des quantités  $E_k(S_n)$  sans calculer les déterminants qui apparaissent dans (1.30). Il peut aussi être interprété comme un cas particulier d'une généralisation due à Mühlbach [93] du schéma de Neville Aitken pour le calcul récursif du polynôme d'interpolation. Il est donné par les règles suivantes

$$\begin{aligned} E_0^{(n)} &= S_n, \quad n = 0, 1, \dots, \\ g_{0,i}^{(n)} &= g_i(n), \quad i = 1, 2, \dots \quad \text{and } n = 0, 1, \dots \end{aligned}$$

Pour  $k = 1, 2, \dots$  et  $n = 0, 1, \dots$

$$\begin{aligned} E_k^{(n)} &= E_{k-1}^{(n)} - \frac{E_{k-1}^{(n+1)} - E_{k-1}^{(n)}}{g_{k-1,k}^{(n+1)} - g_{k-1,k}^{(n)}} g_{k-1,k}^{(n)}, \\ g_{k,i}^{(n)} &= g_{k-1,i}^{(n)} - \frac{g_{k-1,i}^{(n+1)} - g_{k-1,i}^{(n)}}{g_{k-1,k}^{(n+1)} - g_{k-1,k}^{(n)}} g_{k-1,k}^{(n)}, \quad i = k+1, k+2, \dots \end{aligned}$$

Les quantités  $E_k^{(n)}$  sont normalement disposées dans un tableau à double entrée

$$\begin{array}{ccccccc} E_0(S_0) & = & S_0 & & & & \\ E_0(S_1) & = & S_1 & E_1(S_0) & & & \\ E_0(S_2) & = & S_2 & E_1(S_1) & E_2(S_0) & & \\ E_0(S_3) & = & S_3 & E_1(S_2) & E_2(S_1) & E_3(S_0) & \\ & \vdots & & \vdots & \vdots & \vdots & \ddots \end{array}$$

Si les suites  $g_i$  sont à convergence linéaire et et si elles forment une échelle de comparaison, Brezinski et Sidi ont montré que chaque colonne du tableau précédent convergeait plus vite que la précédente.

**THÉORÈME 1.26. [12, 103]** *Supposons que  $S_n = S_\infty + a_1g_1(n) + a_2g_2(n) + \dots$ . Si*

$$\lim_{n \rightarrow \infty} \frac{g_i(n+1)}{g_i(n)} = b_i \neq 1, \quad \forall i = 1, 2, \dots \text{ avec } b_i \neq b_j, \forall i \neq j,$$

alors  $\lim_{n \rightarrow \infty} E_k^{(n)} = S_\infty, \quad k = 0, 1, \dots$ . Si, de plus,

$$g_{i+1}(n) = o(g_i(n))(n \rightarrow \infty), \quad \forall i = 1, 2, \dots,$$

alors

$$E_k^{(n)} - S_\infty \sim a_{k+1} \prod_{i=1}^k \frac{b_{k+1} - b_i}{1 - b_i} g_{k+1}(n) \quad (n \rightarrow \infty), \quad \forall k = 0, 1, \dots$$

et donc  $(E_k^{(n)} - S_\infty)/(E_{k-1}^{(n)} - S_\infty) = \mathcal{O}(g_{k+1}(n)/g_k(n)) \quad (n \rightarrow \infty)$ .

### 1.4.2 Propriété d'accélération

Le précédent théorème ne peut pas être appliqué dans le cas (très fréquent en analyse numérique) où on a un développement asymptotique dans l'échelle de comparaison  $1/n, 1/n^2, 1/n^3, \dots$ . On va obtenir le même résultat d'accélération mais avec d'autres conditions suffisantes, ce qui permettra d'englober d'autres familles  $\{g_i\}$  que celles considérées dans le théorème précédent.

En appliquant des identités de déterminants (notamment l'identité de Sylvester) et des résultats de [70] sur les matrices totalement positives, on peut montrer :

**THÉORÈME 1.27. [A4]** *Soient  $\{(g_i(n))_n, i = 1, \dots\}$  une famille de suites vérifiant :*

$$(\mathcal{H}) \left\{ \begin{array}{l} (i) \lim_{n \rightarrow \infty} (g_{i+1}(n)/g_i(n)) = 0, i = 0, 1, \dots; \\ (ii) \forall p \in \mathbb{N}, \forall i \in \mathbb{N}, \exists N \in \mathbb{N} \text{ tel que } n \geq N \Rightarrow \begin{vmatrix} g_{i+p}(n) & \cdots & g_i(n) \\ \vdots & \vdots & \vdots \\ g_{i+p}(n+p) & \cdots & g_i(n+p) \end{vmatrix} \geq 0. \end{array} \right.$$

Alors

$$\forall k \in \mathbb{N}, \forall l \geq k + 1, \forall i \in \mathbb{N} \quad \lim_{n \rightarrow \infty} \frac{|g_{i+l+1}(n) \ g_{i+k}(n) \ \cdots \ g_i(n)|}{|g_{i+l}(n) \ g_{i+k}(n) \ \cdots \ g_i(n)|} = 0.$$

Ce théorème permet d'obtenir une nouvelle propriété d'accélération pour le E-algorithme appliqué à une suite  $(S_n)_{n \in \mathbb{N}}$  pour laquelle on connaît un développement asymptotique dans une échelle de comparaison.

**THÉORÈME 1.28.** [A4] *Soit  $(S_n)_{n \in \mathbb{N}}$  une suite convergeant vers  $S_\infty$  telle que*

$$S_n = S_\infty + a_1 g_1(n) + a_2 g_2(n) + \dots$$

avec  $\{(g_i(n))_n\}_i$  une famille vérifiant  $(\mathcal{H})$  et  $g_0(n) := 1$ . Alors

$$\lim_{n \rightarrow \infty} \frac{E_{k+1}^{(n)} - S_\infty}{E_k^{(n)} - S_\infty} = 0, \quad k = 0, 1, 2, \dots$$

La démonstration fait intervenir l'identité de Sylvester et une astucieuse manipulation de lignes et colonnes de déterminants. Pour les détails consulter [A4].

**Remarque 1 :** les suites auxiliaires  $(g_{k,i}(n))_n$  obtenues par application du E-algorithme peuvent s'exprimer comme un quotient de deux déterminants [12] et on obtient alors

$$g_{k,i}(n) = (-1)^k \frac{\begin{vmatrix} g_i(n) & g_k(n) & \cdots & g_1(n) \\ g_k(n) & \cdots & g_1(n) & 1 \end{vmatrix}}{\begin{vmatrix} g_k(n) & \cdots & g_1(n) & 1 \end{vmatrix}}.$$

La condition **(ii)** de la propriété  $\mathcal{H}$  implique alors que

$$(-1)^k g_{k,i}(n) \geq 0, \quad \forall i \in \mathbb{N}, \quad \forall k \in \mathbb{N}, \quad \forall n \geq N,$$

et cette condition est facilement vérifiable pendant les calculs du E-algorithme.

**Remarque 2 :** On peut montrer que la propriété  $\mathcal{H}$  est vérifiée par un large ensemble de familles de suites  $\{(g_i(n))_n\}$ , parmi lesquelles figurent :

- (1)  $g_1(n) = c_n$ ,  $g_i(n) = (-1)^i \Delta^i c_n$ , pour  $i \geq 2, \forall n \in \mathbb{N}$ , où  $(c_n)$  est une suite de moments de Stieltjes ( $c_n := \int_0^1 x^n d\alpha(x)$ ,  $\text{support}(\alpha) = [0, 1]$ ).
- (2)  $g_i(n) = K(\alpha_n, \beta_i)$ , avec  $(\alpha_n)$  une suite de réels croissante,  $(\beta_i)$  une suite de réels décroissante,  $K$  une fonction totalement positive [70], c'est-à-dire,

$$\forall r \in \mathbb{N}, \text{ pour tout } x_1 < x_2 < \cdots < x_m, \quad y_1 < y_2 < \cdots < y_m, \quad x_i, y_i \in \mathbb{R}, 1 \leq m \leq r,$$

$$K \begin{pmatrix} x_1 & x_2 & \cdots & x_m \\ y_1 & y_2 & \cdots & y_m \end{pmatrix} = \begin{vmatrix} K(x_1, y_1) & K(x_1, y_2) & \cdots & K(x_1, y_m) \\ K(x_2, y_1) & K(x_2, y_2) & \cdots & K(x_2, y_m) \\ \vdots & \vdots & & \vdots \\ K(x_m, y_1) & K(x_m, y_2) & \cdots & K(x_m, y_m) \end{vmatrix} \geq 0.$$

Dans ce cas la condition **(ii)** est satisfaite.

Si, par exemple,  $K(x, y) = x^y$ , alors  $g_i(n) = \alpha_n^{\beta_i}$ . Si de plus  $\lim_{n \rightarrow \infty} \alpha_n = 0$  et  $\beta_i > \beta_{i+1}, \forall i$  alors la condition **(i)** est aussi satisfaite. Cela inclut le cas particulier  $g_i(n) = n_i^\beta$ ,  $0 > \beta_1 > \beta_2 > \cdots$ .

- (3)  $g_i(n) = n^{\alpha_i} (\log n)^{\beta_i}$  avec

- (a)  $\alpha_1 > \alpha_2 > \alpha_3 > \cdots, \beta_i \in \mathbb{R}$

ou

(b) si  $\alpha_{i_0} = \alpha_{i_0+1}$  pour un  $i_0$  donné, alors  $\beta_{i_0} > \beta_{i_0+1}$ .

Un résultat d'accélération généralisant le théorème 1.28 au cas de suites vectorielles est donné dans [A4]. Les conditions sont un peu plus longues et difficiles à vérifier, mais on peut montrer que, si les composantes des suites vectorielles  $\{(g_i(n))_n\}$  appartiennent à la famille

$$\mathcal{F} = \left\{ (f_j(n))_n : f_j(n) = n^{\alpha_j} (\log n)^{\beta_j}, \quad \alpha_j < 0, \alpha_j \geq \alpha_{j+1}; \quad \text{si } \alpha_j = \alpha_{j+1} \text{ alors } \beta_j > \beta_{j+1} \right\}$$

alors les conditions du théorème sont vérifiées et on a

$$\lim_{n \rightarrow \infty} \frac{\|E_k^{(n)} - S_\infty\|}{\|E_{k-1}^{(n)} - S_\infty\|} = 0.$$

Ce résultat est très intéressant car la famille de suites  $\mathcal{F}$  apparaît souvent dans le développement de l'erreur dans les méthodes de l'Analyse Numérique. C'est par exemple le cas des suites obtenues par la méthode de la puissance pour le calcul du vecteur PageRank de Google (voir [31]).





## Chapitre 2

# Propriétés de convergence et d'accélération de quelques généralisations des approximants de Padé

Ce chapitre est basé sur les publications [A1], [A2], [A3], [A5], [A7] et [A8].

### 2.1 Introduction

Dans ce chapitre nous nous sommes intéressés à différents types de généralisations des approximants de Padé et de type Padé. Nous commençons par rappeler brièvement les définitions. Une vaste littérature concerne ces approximants, dont les propriétés et applications essentielles peuvent être consultés dans [11] et [4].

Soit  $f$  une série entière

$$f(t) = \sum_{i=0}^{\infty} c_i t^i.$$

On définit la forme linéaire  $c$  sur l'espace des polynômes complexes par

$$\begin{aligned} c(x^i) &= c_i \in \mathbb{C}, & i = 0, 1, \dots \\ &= 0, & i < 0. \end{aligned}$$

Soit  $v_k$  un polynôme arbitraire de degré  $k$ , on définit

$$w_k(t) = a_0 + \dots + a_{k-1} t^{k-1} \text{ avec } a_i = c(x^{-i-1} v_k(x)) \quad i = 0, \dots, k-1.$$

On pose

$$\tilde{v}_k(t) = t^k v_k(t^{-1}) \quad \text{et} \quad \tilde{w}_k(t) = t^{k-1} w_k(t^{-1}).$$

Alors on peut montrer que

$$f(t) - \tilde{w}_k(t)/\tilde{v}_k(t) = O(t^k) \quad (t \rightarrow 0).$$

La fonction rationnelle  $\tilde{w}_k(t)/\tilde{v}_k(t)$  est appelé *approximant de type Padé de  $f$*  et on le note par  $(k - 1/k)_f(t)$  [11]. De plus on peut montrer que

$$\begin{aligned} f(t) - \frac{\tilde{w}_k(t)}{\tilde{v}_k(t)} &= \frac{t^k}{\tilde{v}_k(t)} c \left( \frac{v_k(x)}{1 - xt} \right) \\ &= \frac{t^k}{\tilde{v}_k(t)} c \left( \left( 1 + xt + \dots + x^{k-1}t^{k-1} + \frac{x^k t^k}{1 - xt} \right) v_k(x) \right), \end{aligned}$$

ce qui permet d'obtenir

$$f(t)\tilde{v}_k(t) - \tilde{w}_k(t) = t^k \sum_{i=0}^{\infty} c(x^i v_k(x)) t^i.$$

De cette forme linéarisée de l'erreur on conclut que, si le polynôme  $v_k$  (appelé le *polynôme générateur de  $(k - 1/k)$* ) vérifie

$$c(x^i v_k(x)) = 0 \quad \text{pour } i = 0, \dots, k - 1,$$

alors on obtient

$$f(t) - \tilde{w}_k(t)/\tilde{v}_k(t) = O(t^{2k}).$$

Dans ce cas  $v_k$  est le polynôme orthogonal formel  $P_k$  de degré  $k$  par rapport à la forme linéaire  $c$  et  $\tilde{w}_k(t)/\tilde{v}_k(t)$  est l'*approximant de Padé  $[k - 1/k]_f$  de  $f$* .

Les généralisations que nous avons étudiées prennent en compte différents types de problèmes et ont pour but soit d'améliorer les propriétés de convergence et la vitesse de convergence de la suite des sommes partielles vers la fonction, soit d'améliorer les propriétés de stabilité numérique, soit encore de considérer des développements en série plus généraux que la série entière.

La première généralisation (section **2.2**) concerne les problèmes de stabilité numérique. Comme il est expliqué dans [78], les approximants de Padé sont très sensibles à des perturbations dans les coefficients ( $c_i$ ) de la série. Pour pallier cette difficulté, l'idée nous est venue de prendre pour polynôme  $v_k$  un polynôme qui, au lieu de vérifier  $c(x^i v_k(x)) = 0$  pour  $i = 0, \dots, k - 1$  (polynôme orthogonal), minimise  $\sum_{i=0}^m [c(x^i v_k(x))]^2$ , ( $m > k - 1$ ) essayant ainsi d'augmenter la stabilité numérique et la précision des approximants. Nous avons été amené à définir les *polynômes orthogonaux au sens des moindres carrés*, étudier leurs propriétés, donner des algorithmes de calcul, et à construire les *approximants de type Padé au sens des moindres carrés* pour lesquels des bonnes propriétés de stabilité ont été obtenues. Une autre application de ces polynômes aux formules de quadrature a aussi été développée.

Une deuxième généralisation (définie en [23]) (section **2.3**) a consisté à prendre comme dénominateur de l'approximant une fonction analytique  $g(z)$  (pas nécessairement un polynôme) : les *approximants de type Cauchy*. Nous nous sommes intéressés à leurs propriétés de convergence pour certaines familles de séries  $f$  (séries lacunaires, séries périodico-linéaires, ...) et aussi à l'accélération de convergence. Notre souci étant de construire des approximants qui convergent plus vite que les sommes partielles de la série, nous avons donné des conditions suffisantes sur la suite des coefficients de la série et sur la fonction  $g(z)$  pour obtenir l'accélération de convergence, i.e.,  $\lim_{n \rightarrow \infty} (\mathcal{C}_n(z) - f(z)) / (f_n(z) - f(z)) = 0$ , où  $(\mathcal{C}_n(z))$  est la suite d'approximants et  $(f_n(z))$  la suite des sommes partielles. Des nouvelles transformations de suites basées sur ces approximants ont été proposées et leurs propriétés d'accélération ont été démontrées.

La troisième généralisation (section 2.4) a consisté à considérer une famille de fonctions plus générale que les séries entières,  $f(z) = \sum_{n=0}^{\infty} f_n g_n(z)$ , où  $\{g_i(z)\}$  est une famille de fonctions (par exemple, les polynômes orthogonaux) - les *approximants de type Padé généralisés* ont été définis dans [24]. Les bons résultats numériques obtenus pour certaines suites de ces approximants ont motivé notre étude de leurs propriétés de convergence pour certaines classes de fonctions : on a considéré différentes familles  $\{g_i(z)\}_{i \in \mathbb{N}}$  et, à partir de propriétés sur les suites de coefficients  $(f_i)$  du développement de  $f$  et sur la fonction génératrice  $G(x, t)$  de la famille  $\{g_i(z)\}_i$ , nous avons estimé la vitesse de convergence de certaines suites de ces approximants.

Les sujets traités sont loin d'être clos. Différents problèmes évoqués le long du chapitre restent ouverts et font toujours partie de mes recherches actuelles.

## 2.2 Polynômes orthogonaux au sens des moindres carrés et applications

### 2.2.1 Définitions et propriétés

Soit  $c$  la forme linéaire définie sur l'espace des polynômes complexes par

$$\begin{aligned} c(x^i) &= c_i \in \mathbb{C}, \quad i = 0, 1, \dots \\ &= 0, \quad i < 0. \end{aligned}$$

$\{P_k\}$  forme une *famille de polynômes orthogonaux (formels) par rapport à  $c$*  ssi  $\forall k$

- $P_k$  est de degré exact  $k$  ;
- $c(x^i P_k(x)) = 0$  pour  $i = 0, \dots, k-1$ .

Une telle famille existe si pour tout  $k$ , le déterminant de Hankel

$$H_k^{(0)} = \begin{vmatrix} c_0 & c_1 & \cdots & c_{k-1} \\ c_1 & c_2 & \cdots & c_k \\ \cdots & \cdots & \cdots & \cdots \\ c_{k-1} & c_k & \cdots & c_{2k-2} \end{vmatrix}$$

est non nul. Ces polynômes satisfont la plupart des propriétés des polynômes orthogonaux usuels, c'est-à-dire, ceux correspondant à une forme  $c$  donnée par

$$c(x^i) = \int_a^b x^i d\alpha(x),$$

où  $\alpha$  est bornée non décroissante sur  $[a, b]$ . Pour plus de détails sur la théorie de ces polynômes, consulter [11]. Nous allons maintenant généraliser et définir les *polynômes orthogonaux au sens des moindres carrés*  $R_k(x)$  de degré exact  $k$ . Ils sont définis à une constante multiplicative près. Nous supposons, pour l'instant et pour fixer les idées, qu'il sont unitaires. On définit alors les polynômes

$$R_k(x) = b_0 + b_1 x + \cdots + b_{k-1} x^{k-1} + b_k x^k \quad \text{avec } b_k = 1,$$

qui minimisent la fonction

$$\Phi(b_0, \dots, b_{k-1}) = \sum_{i=0}^m [c(x^i R_k(x))]^2 \quad \text{avec } m > k-1.$$

(pour  $m = k - 1$  on retrouve les polynômes orthogonaux formels). Ces polynômes sont biorthogonaux et dépendent de la valeur de  $m$  (pour simplifier les notations on ne notera pas explicitement cette dépendance). Le minimum de  $\Phi$  est atteint pour les  $(b_i)$  vérifiant  $\partial\Phi/\partial b_j = 0$  pour  $j = 0, \dots, k - 1$ . Si l'on pose  $\gamma_n = (c_n, \dots, c_{n+m})^T$ , les coefficients de  $R_k(x)$  sont obtenus en résolvant le système

$$b_0(\gamma_0, \gamma_j) + \dots + b_{k-1}(\gamma_{k-1}, \gamma_j) = -(\gamma_k, \gamma_j), \quad j = 0, \dots, k - 1.$$

Donc  $R_k$  existe et est unique ssi la matrice  $A_k$  de ce système est inversible. Si on note  $X = (1, x, \dots, x^{k-1})$  et  $\gamma$  le second membre de ce système, on a la représentation

$$R_k(x) = \frac{\begin{vmatrix} A_k & \gamma \\ X & x^k \end{vmatrix}}{|A_k|}.$$

Si l'on pose

$$B_k = \begin{pmatrix} c_0 & \dots & \dots & c_{k-1} \\ \dots & \dots & \dots & \dots \\ c_m & \dots & \dots & c_{m+k-1} \end{pmatrix},$$

alors  $A_k = B_k^T B_k$ ,  $\gamma = B_k^T \gamma_k$  et l'on retrouve la solution d'un système d'équations linéaires au sens des moindres carrés.

Comme  $R_k$  est défini à un facteur multiplicatif près, selon les applications en vue on peut choisir différentes normalisations : en particulier, comme on le verra ci-dessous, le choix  $b_0 = 1$  simplifie les algorithmes de calcul. C'est aussi le choix fait dans les méthodes de Lanczos pour la résolution d'un système d'équations linéaires. Pour une valeur fixée de  $m$  indépendante de  $k$  on ne peut construire qu'une famille finie de polynômes orthogonaux au sens des moindres carrés (que l'on notera POMC)  $R_0, \dots, R_{m+1}$ . Un problème intéressant et encore ouvert serait d'étudier le cas où  $m$  dépend de  $k$  (par exemple  $m = pk + q$ ) ce qui permettra la construction d'une suite infinie de polynômes.

On peut montrer quelques propriétés importantes vérifiées par ces polynômes

**PROPOSITION 2.1. [A5, A10]** *Soit  $R_k(x)$  le polynôme orthogonal formel aux sens des moindres carrés correspondant à la normalisation  $b_k = 1$  et à une forme linéaire  $c$  définie par*

$$c_i = c(x^i) = \int_a^b x^i d\alpha(x), \quad i = 0, 1, \dots, \quad (2.1)$$

avec  $\alpha$  bornée non décroissante dans  $[a, b]$ . On pose

$$w(x, \mu) = \int_a^b y^\mu \left( \sum_{j=0}^m x^j y^j \right) d\alpha(y).$$

Alors le polynôme  $R_k$  est biorthogonal dans le sens de [66, 67], c'est-à-dire,

$$c(R_k(x)w(x, i)) = \int_a^b R_k(x)w(x, i)d\alpha(x) = 0 \quad \text{pour } i = 0, \dots, k - 1.$$

En appliquant des résultats de [66, 67] sur les polynômes biorthogonaux, on peut obtenir le résultat suivant sur les racines de ces polynômes  $R_k(x)$ .

**THÉORÈME 2.2. [A5, A10]** *Supposons vérifiées les conditions de la proposition 2.1. On définit*

- $X_i = (1, x_i, \dots, x_i^m)^T$ , où les  $x_i$  sont des points distincts de  $[a, b]$ , et  $\gamma_n = (c_n, \dots, c_{n+m})^T$  ;
- $\mathcal{X}_k = \begin{pmatrix} X_1^T \\ \vdots \\ X_k^T \end{pmatrix}$  et  $\Gamma_k = (\gamma_0, \dots, \gamma_{k-1})$  ;
- la matrice

$$\mathcal{I}_k = \mathcal{X}_k \Gamma_k = \begin{pmatrix} (\gamma_0, X_1) & (\gamma_1, X_1) & \cdots & (\gamma_{k-1}, X_1) \\ (\gamma_0, X_2) & (\gamma_1, X_2) & \cdots & (\gamma_{k-1}, X_2) \\ \cdots & \cdots & \cdots & \cdots \\ (\gamma_0, X_k) & (\gamma_1, X_k) & \cdots & (\gamma_{k-1}, X_k) \end{pmatrix}.$$

*Si  $A_k$  est inversible et si  $\mathcal{I}_k$  est de rang  $k$ , alors le POMC existe, et a  $k$  racines distinctes dans  $[a, b]$ .*

En utilisant des résultats de la théorie des matrices [70] (notamment sur la totale positivité des matrices de Vandermonde généralisées et de certaines matrices de Hankel) on peut simplifier les conditions de ce théorème dans le cas particulier suivant.

**COROLLAIRE 2.3. [A5]** *Supposons que  $0 \leq a < b$  et que  $c$  vérifie la condition (2.1). Alors si  $A_k$  est inversible,  $R_k$  possède  $k$  racines distinctes dans  $[a, b]$ .*

Remarque : Dans le cas où  $0 \in [a, b]$  des exemples montrent que  $\det I_k$  peut être nul.

Ces résultats sont très importants dans le développement de formules de quadrature, comme on le verra dans les applications.

## 2.2.2 Calcul récursif

Les polynômes  $R_k$  peuvent être calculés récursivement en inversant la matrice  $A_k$  du système qui donne ses coefficients par la méthode de bordage. Cette méthode (expliquée en détail dans [51]) consiste à calculer l'inverse d'une matrice d'ordre  $k + 1$  de la forme

$$A_{k+1} = \begin{pmatrix} A_k & u_k \\ v_k & a_k \end{pmatrix}, \quad u_k \in \mathbb{R}^k, \quad v_k \in \mathbb{R}^{1 \times k}, \quad a_k \in \mathbb{R},$$

à partir de l'inverse connue de  $A_k$ ,  $A_k^{-1}$ , en utilisant la formule :

$$A_{k+1}^{-1} = \begin{pmatrix} A_k^{-1} + A_k^{-1} u_k \beta_k^{-1} v_k A_k^{-1} & -A_k^{-1} u_k \beta_k^{-1} \\ -\beta_k^{-1} v_k A_k^{-1} & \beta_k^{-1} \end{pmatrix}, \quad \text{avec } \beta_k = a_k - v_k A_k^{-1} u_k.$$

Si au lieu de choisir la normalisation  $b_k = 1$  on impose  $b_0 = 1$ , le système qui donne les coefficients  $b'_i$  du polynôme  $R_k$  s'écrit

$$b'_1(\gamma_1, \gamma_j) + \cdots + b'_k(\gamma_k, \gamma_j) = -(\gamma_0, \gamma_j), \quad j = 1, \dots, k, \quad (2.2)$$

et l'on voit facilement que l'on peut appliquer la méthode de bordage pour calculer récursivement non seulement les inverses mais aussi la solution. En effet, si  $A'_k$  est la matrice du système (2.2) et  $d'_k$  le second membre, alors on a

$$A'_{k+1} = \begin{pmatrix} A'_k & u'_k \\ v'_k & a_k \end{pmatrix}, \quad d'_{k+1} = \begin{pmatrix} d'_k \\ f'_k \end{pmatrix},$$

avec

$$\begin{aligned} u'_k &= ((\gamma_{k+1}, \gamma_1), \dots, (\gamma_{k+1}, \gamma_k))^T; \\ v'_k &= ((\gamma_1, \gamma_{k+1}), \dots, (\gamma_k, \gamma_{k+1})); \\ a_k &= (\gamma_{k+1}, \gamma_{k+1}); \\ d'_k &= ((\gamma_0, \gamma_1), \dots, (\gamma_0, \gamma_k))^T; \\ f'_k &= (\gamma_0, \gamma_{k+1}). \end{aligned}$$

Si l'on pose  $z'_k = (b'_1, \dots, b'_k)^T$  on obtient

$$z'_{k+1} = \begin{pmatrix} z'_k \\ 0 \end{pmatrix} + \frac{f'_k - v'_k z'_k}{\beta'_k} \begin{pmatrix} -A'^{-1}_k u'_k \\ 1 \end{pmatrix},$$

avec  $\beta'_k = a'_k - v'_k A'^{-1}_k u'_k$ , ce qui permet de développer les algorithmes récursifs. La méthode de bordage ne peut être utilisée que si les  $\beta'_k$  sont non nuls. Si ce n'est pas le cas, on peut rajouter un bloc de lignes et colonnes jusqu'à ce que  $\beta_k$  (maintenant une matrice carrée) soit inversible. Cette méthode de bordage par blocs a été donnée dans [28].

D'autres algorithmes récursifs pour le calcul de ces polynômes peuvent être développés en utilisant

- soit le Compact Recursive Projection Algorithm (CPRA) [17];
- soit les formules récursives de la théorie de la biorthogonalité [24].

### 2.2.3 Applications

La motivation de l'étude de ces polynômes a été d'améliorer les propriétés de stabilité numérique des approximants de Padé. En effet il est bien connu que les approximants de Padé peuvent être très sensibles à des perturbations des coefficients de la série  $f$ . D'où l'idée de prendre comme dénominateur de l'approximant  $v_k = R_k$  le polynôme orthogonal au sens des moindres carrés de degré  $k$ , au lieu de prendre  $\tilde{v}_k(t) = t^k v_k(t^{-1})$  où  $v_k$  est le polynôme orthogonal usuel (qui, on rappelle, vérifie  $c^i(v_k) = 0, i = 0, \dots, k-1$ ). Un tel choix diminue bien sûr l'ordre d'approximation, mais, quand on fait les calculs, les  $c_i$  ne sont que des approximations des vraies valeurs. On pense ainsi augmenter leur stabilité et aussi leur précision,  $\sum_{i=0}^m [c(x^i v_k(x))]^2$  étant minimisé par le choix  $v_k = R_k$ . On définit l'approximant de type Padé au sens des moindres carrés comme l'approximant de type Padé correspondant au choix  $v_k = R_k$ .

Les algorithmes récursifs de la section précédente ont été programmés et quelques résultats numériques encourageants ont été donnés dans [A5].

Une autre application importante des POMC concerne les formules de quadrature et a été considérée dans [A5] : dans les conditions du corollaire 2.3, les racines des polynômes  $R_k(x)$  sont distinctes et permettent la construction de formules de quadrature de type interpolation, pour

lesquelles l'erreur de troncature vérifie une certaine propriété de minimisation. Les exemples numériques montrent que ces formules sont moins sensibles aux perturbations de la suite des moments  $(c_i)$ .

D'autres applications sont envisageables à partir de la généralisation suivante : au lieu de minimiser  $\sum_{i=0}^m [c(x^i R_k(x))]^2$  on peut introduire des poids  $p_i$  et minimiser

$$\Phi^*(b_0, \dots, b_{k-1}) = \sum_{i=0}^m p_i [c(x^i R_k^*(x))]^2,$$

avec  $p_i > 0$ ,  $i = 0, \dots, m$ . Il est facile de voir que si l'on choisit le produit scalaire  $(\gamma_i, \gamma_j)^* = \sum_{k=0}^m p_k c_{i+k} c_{j+k}$ , la solution du problème peut être calculée comme dans le cas précédent et toutes les propriétés des polynômes correspondants sont encore vérifiées. Il sera intéressant d'étudier comment choisir les  $p_i$  de sorte que l'on améliore les approximations obtenues. Par exemple, si la suite des moments  $c_i$  est connue avec une précision décroissante, on peut espérer que les approximants de type Padé au sens des moindres carrés construits avec une suite décroissante de poids donnent un meilleur résultat.

Considérons maintenant la deuxième généralisation.

## 2.3 Approximants de type Cauchy : propriétés d'accélération et nouvelles transformations de suites

Commençons par rappeler la définition des *approximants de type Cauchy* (A.T.C.) donnée dans [23].

DEFINITION 2.4. Soit  $f(z) = \sum_{n=0}^{\infty} a_n z^n$  une série entière de rayon de convergence  $R$ . L'approximant de type Cauchy d'ordre  $n$  de  $f$  avec fonction génératrice  $g(z)$  est défini par

$$\mathcal{C}_n(z) = \frac{h_n(z)}{g(z)}, \quad n \in \mathbb{N}, \tag{2.3}$$

où

- $g(z) = \sum_{i=0}^{\infty} b_i z^i$  est une série entière de rayon de convergence  $R_1 > R$ ;
- $h_n(z) = \sum_{i=0}^n c_i z^i$ ,  $n \in \mathbb{N}$ ,  $c_i = a_0 b_i + a_1 b_{i-1} + \dots + a_i b_0$ ,  $i \geq 0$ .

Ces approximants généralisent les approximants de type Padé [11, 23] : si  $g(z)$  est un polynôme de degré  $k$ , alors  $\mathcal{C}_n(z)$  est l'approximant de type Padé de  $f$ ,  $(n/k)_f$ , avec polynôme générateur  $v(z) = z^k g(z^{-1})$ , ce qui veut dire que  $\mathcal{C}_n(z)$  est une fonction rationnelle de dénominateur de degré  $k$  et telle que  $\mathcal{C}_n(z) = f_n(z) + \mathcal{O}(z^{n+1})$  (où, on rappelle,  $f_n(z)$  est la somme partielle de la série  $f$ ). Nous nous sommes intéressés à l'étude des propriétés d'accélération de convergence de suites de ces approximants. Pour certaines classes de fonctions  $f$  définies à partir de propriétés de la suite des coefficients  $(a_n)$ , nous avons donné des conditions suffisantes sur la fonction  $g(z)$  de façon que les suites de A.T.C. convergent vers  $f$  plus vite que la suite des sommes partielles  $f_n$ .

Le premier résultat a été donné par Brezinski [23] pour le cas où la suite des coefficients est à convergence linéaire :

**THÉORÈME 2.5.** Soit  $f(z) = \sum_{n=0}^{\infty} a_n z^n$  une série entière vérifiant  $\lim_{n \rightarrow \infty} a_{n+1}/a_n = a \neq 0$  et  $g(z) = \sum_{i=0}^{\infty} b_i z^i$ . Supposons que

(i)  $\exists b : |b| < |a|$  et  $\lim_{n \rightarrow \infty} b_{n+1}/b_n = b$ ;

(ii)  $g(1/a) = 0$

(iii)  $\exists c > 0$  ( $c \leq 2|a|$ )  $\exists N \in \mathbb{N} \forall n \geq N |c_{n+1}/c_n| < c$ .

Alors pour tout  $z$  tel que  $|z| < 1/(2|a|)$  et  $g(z) \neq 0$  on obtient

$$\lim_{n \rightarrow \infty} \frac{C_n(z) - f(z)}{f_n(z) - f(z)} = 0.$$

Nous avons généralisé ce résultat au cas des séries lacunaires et des séries périodico-linéaires. Nous avons complété l'étude en proposant, quand les hypothèses sur  $g$  données dans les théorèmes ne sont pas satisfaites, des nouvelles transformations de suites basées sur ces approximants et qui accélèrent la convergence des sommes partielles. Nous avons aussi obtenu, en imposant des conditions supplémentaires sur  $g$ , de meilleurs résultats d'accélération pour  $f$  dans les conditions du théorème 2.5.

Commençons par étudier le cas des séries lacunaires.

### 2.3.1 Propriété d'accélération pour les séries lacunaires

**DEFINITION 2.6.** Soit  $f(z) = \sum_{n=0}^{\infty} a_n z^n$  une série entière avec  $(a_n)$  vérifiant :

- $\exists p, q \in \mathbb{N}, \lim_{k \rightarrow \infty} a_{p(k+1)+q}/a_{pk+q} = a \neq 0$ ,
- $\exists n_0 \in \mathbb{N}$ , pour  $n \geq n_0$  et  $n \neq kp + q$ ,  $a_n = 0$ .

$f$  est une série lacunaire. La suite  $(a_n)$  est une suite lacunaire.

On se place maintenant dans le cas d'une série  $f$  lacunaire. Pour comparer la vitesse de convergence de suites d'approximants de type Cauchy  $(C_n(z))_n$  avec celle de la suite des sommes partielles  $f_{pk+q}(z) = \sum_{j=0}^{pk+q} a_j z^j$ ,  $n \in \mathbb{N}$ , de la série  $f(z)$ , on va utiliser le résultat suivant :

**THÉORÈME 2.7.** [113] Soit  $f(z) = \sum_{n=0}^{\infty} a_n z^n$  une série entière de rayon de convergence  $R$  telle que  $(a_n)_n$  est une suite lacunaire et  $g(z) = \sum_{n=0}^{\infty} b_n z^n$  une autre série entière de rayon de convergence  $R_1 > R$ . On définit

$$g_i(z) = \sum_{n=0}^{\infty} b_{np+i} z^{np+i}, \quad 0 \leq i \leq p-1. \quad (2.4)$$

Soit

$$h(z) = f(z)g(z) = \sum_{n=0}^{\infty} c_n z^n.$$

Alors

$$\lim_{k \rightarrow \infty} \frac{c_{pk+q+i}}{a_{pk+q}} = g_i(1/a), \quad 0 \leq i \leq p-1;$$

si, de plus,  $g_i(1/a) \neq 0$  alors  $\lim_{k \rightarrow \infty} c_{p(k+1)+q+i}/c_{pk+q+i} = a$ .



À partir de ce théorème, on peut calculer la limite des suites

$$\left( \frac{\Delta \mathcal{C}_{pk+q+j}(z)}{\Delta \mathcal{C}_{pk+q+j-1}(z)} \right)_k, \quad \left( \frac{\Delta \mathcal{C}_{pk+q+j-1}(z)}{\Delta f_{pk+q-1}(z)} \right)_k, \quad 0 \leq j \leq p-1,$$

ce qui nous permettra d'obtenir les propriétés d'accélération des A.T.C. pour une classe de séries lacunaires.

**THÉORÈME 2.8. [A1]** Soient  $f(z)$ ,  $g(z)$ ,  $g_i(z)$ ,  $1 \leq i \leq p-1$  et  $h(z)$  définies comme dans le théorème 2.7 et  $(\mathcal{C}_n(z))_n$  la suite d'approximants de type Cauchy avec fonction génératrice  $g(z)$ .

Supposons que :

**(H1)**  $\exists j \in \{0, \dots, p-1\}$ ,  $g_j(1/a) = 0$ ;

**(H2)**  $\exists \rho > 0, \exists n_0 \in \mathbb{N}, \forall n \geq n_0 \quad |c_{n+1}/c_n| < \rho$ .

Alors, pour tout  $z$  vérifiant :  $|z| < \min(1/\rho, |a|^{-1/p})$ ,  $g(z) \neq 0$  on obtient

$$\lim_{k \rightarrow \infty} \frac{\mathcal{C}_{pk+q+i}(z) - f(z)}{f_{pk+q}(z) - f(z)} = 0, \quad 0 \leq i \leq p-1.$$

### 2.3.2 Nouvelle transformation de suites basée sur A.T.C

Considérons maintenant le cas où

$$g_j(1/a) \neq 0, \quad 0 \leq j \leq p-1. \quad (2.5)$$

Les hypothèses du théorème précédent n'étant plus vérifiées, on ne peut pas conclure que la suite  $(\mathcal{C}_n(z))_n$  converge plus vite que  $(f_n(z))_n$ . On va donc proposer une nouvelle transformation qui accélère la convergence de  $(f_n(z))_n$ . On commence par comparer le comportement asymptotique du rapport des erreurs avec celui du rapport des différences pour les suites  $(\mathcal{C}_n(z))_n$  et  $(f_n(z))_n$ , c'est-à-dire, calculer, sous certaines hypothèses,

$$\lim_{k \rightarrow \infty} \frac{\mathcal{C}_{pk+q+j}(z) - f(z)}{f_{pk+q}(z) - f(z)} \text{ et } \lim_{k \rightarrow \infty} \frac{\mathcal{C}_{pk+q+j}(z) - \mathcal{C}_{p(k-1)+q+j}(z)}{\Delta f_{pk+q-1}(z)}$$

(où l'opérateur  $\Delta$  agit sur la variable  $k$ ). Ceci nous amène à la construction d'une nouvelle transformation de suites basée sur les A.T.C. et qui a les propriétés d'accélération résumées dans le résultat suivant :

**THÉORÈME 2.9. [A1]** Soient  $f(z)$ ,  $g(z)$ ,  $g_i(z)$ ,  $1 \leq i \leq p-1$  et  $h(z)$  définies comme dans le théorème 2.7 et  $(\mathcal{C}_n(z))_n$  la suite d'approximants de type Cauchy avec fonction génératrice  $g(z)$ .

Supposons que

**(H1\*)**  $g_j(1/a) \neq 0, \quad 0 \leq j \leq p-1$ , ce qui nous permet de définir les quantités

$$r_j(z) = \frac{1}{g_j(1/a)} \left( a \sum_{m=0}^{j-1} z^{m+p-j} g_m(1/a) + \sum_{m=j}^{p-1} z^{m-j} g_m(1/a) \right), \quad 0 \leq j \leq p-1;$$

**(H2\*)**  $r_0(z) \neq g(z)/g_0(1/a)$ ;  $r_{j+1}(z) \neq g(z)az^{p-j-1}/g_{j+1}(1/a), \quad 0 \leq j \leq p-1$ .

Alors pour tout  $z$  tel que  $g(z) \neq 0$  et  $|z| < |a|^{-1/p}$ , la transformation

$$T_{pk+q+j}(z) = f_{pk+q}(z) - \frac{C_{pk+q+j}(z) - f_{pk+q}(z)}{C_{pk+q+j}(z) - C_{p(k+1)+q+j}(z) - \Delta f_{pk+q-1}(z)} \Delta f_{pk+q-1}(z), \quad 0 \leq j \leq p-1,$$

accélère la convergence de  $(f_{pk+q}(z))_k$  :

$$\lim_{k \rightarrow \infty} \frac{T_{pk+q+j}(z) - f(z)}{f_{pk+q}(z) - f(z)} = 0, \quad 0 \leq j \leq p-1.$$

On remarque que les hypothèses **(H1\*)** et **(H2\*)** ne dépendent que de la valeur de  $a$  et pas des coefficients de  $f$  et donc, une fois choisi  $g(z)$ , on peut déterminer pour chaque valeur de  $a$  l'ensemble  $\mathcal{E}_a$  des points  $z$  pour lesquels on obtient accélération de convergence.

Le cas  $p = 2$ , qui apparaît souvent dans le développement en série entière de fonctions analytiques ou méromorphes, a été étudié plus en détail dans [A1]. En écrivant les conditions **(H1)**, **(H2)**, **(H1\*)** et **(H2\*)** pour ce cas on a pu donner des fonctions  $g(z)$  vérifiant ces hypothèses et déterminer les ensembles  $\mathcal{E}_a$  correspondants. Des résultats numériques illustrent les bonnes propriétés d'accélération des approximants de type Cauchy et des nouvelles transformations proposées.

### 2.3.3 Propriété d'accélération pour des séries périodico-linéaires

On considère maintenant une autre famille de séries entières.

DEFINITION 2.10.  $(a_n)_n$  est une suite à convergence périodico-linéaire de période  $p$  ssi

- (i)  $\exists p \in \mathbb{N} \quad \forall i : 0 \leq i \leq p-1, \quad \exists \alpha_i \in \mathbb{C} : \lim_{k \rightarrow \infty} a_{kp+i+1}/a_{kp+i} = \alpha_i,$
- (ii)  $\alpha_i \neq 0, 1, \quad 0 \leq i \leq p-1.$

Une série entière  $f(z) = \sum_{n=0}^{\infty} a_n z^n$  est dite périodico-linéaire si la suite des coefficients l'est.

Ces suites ont été introduites et étudiées en détail dans [47]. On va procéder comme dans la section précédente et commencer par donner des conditions suffisantes sur  $g$  pour que la suite correspondante de A.T.C. converge plus vite que la suite des sommes partielles.

Soit  $f$  une série périodico-linéaire de rayon de convergence  $R$ . On lui associe les  $p$  séries entières :

$$f^{(q)}(z) = \sum_{k=0}^{\infty} a_{kp+q} z^{kp+q}, \quad 0 \leq q \leq p-1.$$

Pour  $|z| < R$  la série  $\sum_{k=0}^{\infty} a_{kp+q} z^{kp+q}$  converge et  $f(z) = \sum_{q=0}^{p-1} f^{(q)}(z)$ . Pour une fonction génératrice  $g(z)$  on définit

$$h^{(q)}(z) = f^{(q)}(z)g(z) = \sum_{n=0}^{\infty} c_n^{(q)} z^n, \quad 0 \leq q \leq p-1, \quad |z| < R.$$

On a alors

$$h(z) = f(z)g(z) = \sum_{q=0}^{p-1} h^{(q)}(z) = \sum_{n=0}^{\infty} c_n z^n \text{ avec } c_n = \sum_{q=0}^{p-1} c_n^{(q)}.$$

Pour obtenir la vitesse de convergence de la suite des A.T.C. on a besoin de connaître le comportement de la suite  $(c_n)$ . En utilisant le théorème 2.7 l'on obtient le comportement des suites  $(c_n^{(q)})_n$ ,  $0 \leq q \leq p-1$ , et par la suite celui de  $(c_n)$ . A partir de résultats auxiliaires sur le comportement des suites des rapports  $((u_{kp+q} - u)/(v_{kp+q} - v))_k$  en fonction du comportement des suites  $(\Delta u_{kp+q}/\Delta v_{kp+q})_k$  pour certaines suites  $(u_n)$  et  $(v_n)$  convergeant respectivement vers  $u$  et  $v$ , on obtient un résultat d'accélération pour les suites d'approximants de type Cauchy :

**THÉORÈME 2.11. [A3]** *Soit  $f$  une série périodico-linéaire de période  $p$  avec  $\lim_{n \rightarrow \infty} a_{n+p}/a_n = a \neq 0$ . Soit  $g(z)$  une série entière de rayon de convergence  $R > 1/|a|$ . On considère la suite  $(C_n(z))_n$  des A.T.C. de  $f$  avec fonction génératrice  $g$ . On définit :*

$$d_q = \alpha_q \sum_{j=0}^{p-1} \frac{g_j(1/a)}{\alpha_q \alpha_{q-1} \cdots \alpha_{q-j}} \quad (\alpha_i = \alpha_{i+p} \text{ pour } i < 0), \quad q = 0, \dots, p-1.$$

Soit  $A$  tel que  $\forall n \quad |a_{n+1}/a_n| \leq A$  et supposons que :

1.  $\exists q \in \{0, 1, \dots, p-1\} : d_q = 0$ ;
2.  $\exists B > 0, \exists N \in \mathbb{N}, \forall n \geq N \quad |c_{n+1}/c_n| \leq B$ .

Alors pour tout  $z$  vérifiant  $|z| < \min(R, 1/B, 1/2A)$ ,  $g(z) \neq 0$ , on obtient

$$\lim_{k \rightarrow \infty} \frac{C_{pk+i}(z) - f(z)}{f_{pk+i}(z) - f(z)} = 0, \quad 0 \leq i \leq p-1.$$

Si pour tout  $q$  tel que  $0 \leq q \leq p-1$ ,  $d_q \neq 0$ , on ne peut pas conclure du théorème précédent que la suite des A.T.C. accélère la convergence de  $(f_n(z))_n$ . On va donc proposer une nouvelle transformation de suites basée sur les A.T.C. et sur une modification de la procédure  $\theta$  [14] qui produit accélération dans ce cas.

**THÉORÈME 2.12. [A3]** *Soient  $f(z)$ ,  $g(z)$  dans les conditions du théorème 2.11 et on définit de la même façon les quantités  $d_i$  et les approximants  $(C_n(z))_n$ . Supposons que*

- (H1')**  $d_i \neq 0, \quad 0 \leq i \leq p-1; g(z) \neq 0$ . On peut alors définir les quantités
- $\rho_q = 1 + r_q + r_q r_{q+1} + \cdots + r_q r_{q+1} \cdots r_{q+p-2}, \quad 0 \leq q \leq p-1$  avec  $r_i = \alpha_i d_{i+1}/d_i z$ ,  $0 \leq i \leq p-1$  ( $d_p = d_0$ );  $r_i = r_{i-p}$  si  $i \geq p$ ;
  - $\sigma_q = 1 + s_q + s_q s_{q+1} + \cdots + s_q s_{q+1} \cdots s_{q+p-2}, \quad 0 \leq q \leq p-1$  avec  $s_i = \alpha_i z, 0 \leq i \leq p-1$ ;  $s_i = s_{i-p}$  si  $i \geq p$ ;
- (H2')** pour  $0 \leq i \leq p-1, \sigma_i \neq 0$  et  $d_i \rho_i \neq g(z) \sigma_i$ .

Alors pour  $z$  vérifiant  $|z| < \min(R, |a|^{-1/p})$ , la suite

$$T_n(z) = f_n(z) - \frac{C_n(z) - f_n(z))(f_n(z) - f_{n-p}(z))}{(C_n(z) - C_{n-p}(z)) - (f_n(z) - f_{n-p}(z))} \quad n \in \mathbb{N}$$

converge vers  $f(z)$  plus vite que  $(f_n(z))_n$

**Remarque :** si la période  $p$  n'est pas connue, on peut combiner ces transformations avec un algorithme de détermination de la période d'une suite asymptotiquement périodique [47].

Les hypothèses **(H1')**, **(H2')** sont faciles à vérifier et, d'après les exemples numériques de **[A3]**, cette transformation a de bonnes propriétés d'accélération. Il sera donc intéressant de trouver de applications de ces méthodes à des séries entières (venues de la physique ou de problèmes de mathématiques appliquées) convergeant lentement.

### 2.3.4 Remarques sur le choix d'un dénominateur polynomial

Considérons maintenant le choix d'un dénominateur polynomial, i.e., l'approximant est, dans ce cas, un approximant de type Padé. On va mesurer la vitesse de convergence de  $(\mathcal{C}_n(z) - f(z))$ , ce qui nous permettra de donner un choix des polynômes du dénominateur de façon à augmenter la vitesse de convergence au fur et à mesure que l'on augmente le degré du polynôme. Pour cela on va comparer la vitesse de convergence de  $(\mathcal{C}_n(z))_n$  avec celle des sommes partielles. On définit  $R_n(z)$  par

$$\frac{\mathcal{C}_n(z) - f(z)}{f_n(z) - f(z)} = \frac{\Delta \mathcal{C}_n(z)}{\Delta f_n(z)} R_n(z), \quad \mathcal{C}_n(z) = \left( \sum_{i=0}^n c_i z^i \right) / g(z).$$

Supposons que

$$\exists R^{(1)}(z), R^{(2)}(z) > 0 \quad \exists n_0 \in \mathbb{N} \quad \forall n \geq n_0 \quad 0 < R^{(1)}(z) < |R(z)| < R^{(2)}(z). \quad (2.6)$$

Cette condition n'est pas restrictive. En effet, on peut montrer qu'une condition suffisante pour que (2.6) soit satisfaite est que

$$\exists c > 0 \quad \exists N \in \mathbb{N} \quad \forall n \geq N \quad |c_{n+1}/c_n| \leq c.$$

Si (2.6) est satisfaite, alors une condition suffisante d'accélération est

$$\lim_{n \rightarrow \infty} \frac{\Delta \mathcal{C}_n(z)}{\Delta f_n(z)} = \lim_{n \rightarrow \infty} \frac{c_{n+1}}{a_{n+1}} \frac{1}{g(z)} = 0,$$

ce qui est vérifié si  $g(1/a) = 0$ .

Pour chaque  $n \geq 2$ , il existe une infinité de polynômes vérifiant  $g(1/a) = 0$ . Il est intéressant de savoir si l'on peut choisir les coefficients de ces polynômes de façon que, en augmentant leur degré, on augmente aussi la vitesse de convergence des suites d'approximants correspondants. En d'autres termes, on se pose le problème suivant : si  $(\mathcal{C}_n^{(k)}(z))_n$  est la suite d'approximants avec polynôme générateur  $p_k(z)$  de degré  $k$ ,

$$\mathcal{C}_n^{(k)}(z) = \sum_{j=0}^n c_j^{(k)} z^j / p_k(z) \quad n \in \mathbb{N}, \quad \text{avec } p_k(z) = 1 + \alpha_1 z + \cdots + \alpha_k z^k,$$

comment choisir  $p_{k+1}(z)$  de façon à ce que  $(\mathcal{C}_n^{(k+1)}(z))_n$  converge vers  $f(z)$  plus vite que  $(\mathcal{C}_n^{(k)}(z))_n$ . Comme

$$\frac{\Delta \mathcal{C}_n(z)}{\Delta f_n(z)} = \frac{c_{n+1}}{a_{n+1}} \frac{1}{g(z)} \quad n \in \mathbb{N},$$

on peut comparer les vitesses de convergence de  $(\mathcal{C}_n^{(k+1)}(z))_n$  et  $(\mathcal{C}_n^{(k)}(z))_n$  en comparant l'ordre de convergence des suites correspondantes  $(c_n^{(k+1)}/a_n)_n$  et  $(c_n^{(k)}/a_n)_n$ . Or

$$\frac{c_n^{(k)}}{a_n} = 1 + \alpha_1 \frac{a_{n-1}}{a_n} + \alpha_2 \frac{a_{n-2}}{a_n} + \cdots + \alpha_k \frac{a_{n-k}}{a_n},$$

et donc il suffit de choisir les coefficients  $\alpha_j, 1 \leq j \leq k$  du polynôme  $p_k$  de façon à éliminer les coefficients du développement asymptotique de  $(c_n^{(k)}/a_n)_n$ . On obtient alors le résultat suivant :

THÉORÈME 2.13. [A2] Soit  $f(z) = \sum_{n=0}^{\infty} a_n z^n$  une série entière avec  $(a_n)$  vérifiant

- $\lim_{n \rightarrow \infty} a_{n+1}/a_n = a \neq 0$  ;
- pour  $i = 1, \dots, kmax$  on a

$$\frac{a_{n-i}}{a_n} = \frac{1}{a^i} + \sum_{j=0}^{kmax} b_j^{(i)} g_j(n) + o(g_{kmax}(n)) \quad (n \rightarrow \infty)$$

où les  $b_j^{(i)}$  sont des constantes connues,  $g_{j+1}(n) = o(g_j(n)) \quad (n \rightarrow \infty)$  pour  $1 \leq j \leq kmax - 1$ .

On construit les polynômes  $p_i(z) = 1 + \alpha_1 z + \dots + \alpha_i z^i$ ,  $1 \leq i \leq kmax$ , où les coefficients sont solution du système

$$\begin{pmatrix} a^{i-1} & a^{i-2} & \dots & 1 \\ b_1^{(1)} & b_1^{(2)} & \dots & b_1^{(i)} \\ \dots & \dots & \dots & \dots \\ b_{i-1}^{(1)} & b_{i-1}^{(2)} & \dots & b_{i-1}^{(i)} \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_i \end{pmatrix} = \begin{pmatrix} -a^i \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

On considère les approximants de type Cauchy de  $f$ ,  $\mathcal{C}_n^{(i)}(z) = \frac{\sum_{j=0}^n c_j^{(i)} z^j}{p_i(z)}$ ,  $n \in \mathbb{N}$ ,  $1 \leq i \leq kmax$ .

Alors

1.  $\frac{c_n^{(i)}}{a_n} \sim L_i g_i(n) \quad (n \rightarrow \infty)$ ,  $1 \leq i \leq kmax$ ,  $L_i = \sum_{j=1}^i \alpha_j b_i^{(j)}$  ;

2. si de plus  $\forall i \quad L_i \neq 0$ ,

alors, en posant  $\mathcal{C}_n^{(0)}(z) = f_n(z)$ , on obtient

$$\lim_{n \rightarrow \infty} \frac{\mathcal{C}_n^{(i)}(z) - f(z)}{\mathcal{C}_n^{(i-1)}(z) - f(z)} = 0 \quad 1 \leq i \leq kmax,$$

pour tout  $z$  vérifiant  $|z| < \min\left(\frac{1}{2|a|}, \frac{1}{2c^{(i)}}\right)$ ,  $1 \leq i \leq kmax$

La construction des polynômes  $p_i(z)$  proposés dans le résultat précédent nécessite la connaissance des quantités  $(b_i^{(j)})$ . On peut montrer que dans le cas où

$$\frac{a_{n-1}}{a_n} \sim \frac{1}{a} + \frac{b_1^{(1)}}{n} + \frac{b_2^{(1)}}{n^2} + \dots + \frac{b_i^{(1)}}{n^i} + \dots \quad (n \rightarrow \infty),$$

on obtient facilement des formules récursives pour le calcul des  $b_i^{(j)}$  (voir [A2]). Ce cas est assez général car il englobe une grande classe de séries entières.

Dans [A1], [A2], [A3] différents exemples numériques illustrent tous ces résultats théoriques : ils confirment les très bonnes propriétés d'accélération des A.T.C. et montrent la grande variété de fonctions vérifiant les conditions des théorèmes.

## 2.4 Approximants de type Padé généralisés

### 2.4.1 Définitions et propriétés

Soit  $f$  une fonction analytique définie dans un domaine  $A \subseteq \mathbb{C}$  par une série de fonctions

$$f(t) = \sum_{i=0}^{\infty} c_i g_i(t), \quad t \in A.$$

Soit  $G(x, t)$  la fonction génératrice de la famille  $\{g_i(t)\}_i$ , c'est-à-dire,  $G(x, t) = \sum_{i=0}^{\infty} x^i g_i(t)$ . On définit la forme linéaire  $c$  par ses moments de la forme suivante :

$$c(x^i) = c_i \quad i \in \mathbb{N}.$$

Alors on a formellement  $f(t) = c(G(x, t))$ ,  $t \in A$ , où  $c$  agit sur la variable  $x$  (cette convention sera valable tout le long du chapitre).

**DEFINITION 2.14.** *L'approximant de type Padé généralisé d'ordre  $n$  de la série  $f$  (ou plus court, ATPG),  $(n)_f^G$  est défini de la façon suivante [24] :*

- on fixe  $t \in A$  et on considère le polynôme  $q_n(x, t)$  de degré inférieur ou égal à  $n$  qui vérifie les conditions d'interpolation suivantes :

$$L_i(q_n(x, t)) = L_i(G(x, t)) \quad i = 0, \dots, n,$$

où les  $L_i$  sont des formes linéaires agissant sur la variable  $x$ . Pour garantir l'existence et l'unicité du polynôme d'interpolation, les formes linéaires doivent être indépendantes, i.e. satisfaire

$$D_n^{(0)} = \begin{vmatrix} L_0(1) & L_0(x) & \cdots & L_0(x^n) \\ L_1(1) & L_1(x) & \cdots & L_1(x^n) \\ \cdots & \cdots & \cdots & \cdots \\ L_n(1) & L_n(x) & \cdots & L_n(x^n) \end{vmatrix} \neq 0 \quad \forall n \in \mathbb{N}$$

- on remplace  $G$  par son approximation  $q_n$  et on construit l'approximant

$$(n)_f^G(t) = c(q_n(x, t)) \quad n \in \mathbb{N}.$$

Une généralisation similaire des approximants de Padé a été proposée pour une classe de fonctions ayant une représentation sous forme intégrale - les approximants de Baker-Gammel [4].

De la définition du polynôme d'interpolation on obtient

$$q_n(x, t) = \begin{vmatrix} 0 & 1 & x & \cdots & x^n \\ L_0(G(x, t)) & L_0(1) & L_0(x) & \cdots & L_0(x^n) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ L_n(G(x, t)) & L_n(1) & L_n(x) & \cdots & L_n(x^n) \end{vmatrix} / D_n^{(0)},$$

ce qui permet de représenter l'approximant de type Padé généralisé comme un quotient de deux déterminants

$$c(q_n(x, t)) = - \begin{vmatrix} 0 & c_0 & c_1 & \cdots & c_n \\ L_0(G(x, t)) & L_0(1) & L_0(x) & \cdots & L_0(x^n) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ L_n(G(x, t)) & L_n(1) & L_n(x) & \cdots & L_n(x^n) \end{vmatrix} / D_n^{(0)} \Leftrightarrow$$

$$c(q_n(x, t)) = c_0 g_0(t) + c_1 g_1(t) + \cdots + c_n g_n(t) + e_n(t) \text{ avec}$$

$$e_n(t) = - \begin{vmatrix} 0 & c_0 & c_1 & \cdots & c_n \\ L_0(x^{n+1} G_n(x, t)) & L_0(1) & L_0(x) & \cdots & L_0(x^n) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ L_n(x^{n+1} G_n(x, t)) & L_n(1) & L_n(x) & \cdots & L_n(x^n) \end{vmatrix} / D_n^{(0)}.$$

Ceci montre la propriété principale des ATPG : les  $n + 1$  premiers termes de leur développement dans la famille  $\{g_i(t)\}$  coïncident avec ceux de  $f(t)$ . On peut écrire l'approximant sous la forme

$$c(q_n(x, t)) = a_0 L_0(G(x, t)) + a_1 L_1(G(x, t)) + \cdots + a_n L_n(G(x, t)) \quad (2.7)$$

et on a

$$a_0 L_0(x^i) + a_1 L_1(x^i) + \cdots + a_n L_n(x^i) = c(x^i) \quad i = 0, \dots, n.$$

Si les formes linéaires sont définies par  $L_i(g) = g(x_i)$ ,  $i \geq 0$  (où les  $x_i$  sont des abscisses fixées), et la fonction génératrice est donnée par  $G(x, t) = (1 - xt)^{-1}$  alors ces approximants coïncident avec les approximants de type Padé de polynôme générateur  $v(x) = \prod (x - x_i)$  (d'où leur nom).

On peut généraliser cette définition (comme dans [24]) et construire une table d'approximants de type Padé généralisés de la façon suivante. On veut construire un approximant de la forme

$$(k + 1/n)_f^G(t) = \sum_{i=0}^k b_i g_i(t) + a_0 L_0(G(x, t)) + \cdots + a_n L_n(G(x, t))$$

pour lequel le développement en série des  $\{g_i(t)\}$  coïncide avec celui de  $f(t)$  aussi loin que possible, c'est-à-dire, jusqu'à l'ordre  $k + n + 1$ . Cette condition d'ordre peut s'écrire

$$\begin{cases} a_0 L_0(x^{k+1}) + a_1 L_1(x^{k+1}) + \cdots + a_n L_n(x^{k+1}) & = & c_{k+1} \\ \cdots & \cdots & \cdots \\ a_0 L_0(x^{k+1+n}) + a_1 L_1(x^{k+1+n}) + \cdots + a_n L_n(x^{k+1+n}) & = & c_{k+1+n} \end{cases}$$

un système de  $(n + 1)$  équations à  $(n + 1)$  inconnues qui donne les coefficients  $a_i$ . Les coefficients  $b_i$  seront donnés par

$$b_i = c_i - \sum_{j=0}^n a_j L_j(x^i) \quad i = 0, \dots, k.$$

L'existence et l'unicité de ces approximants est garantie par la condition

$$D_n^{(k+1)} = \begin{vmatrix} L_0(x^{k+1}) & L_0(x^{k+2}) & \cdots & L_0(x^{n+k+1}) \\ L_1(x^{k+1}) & L_1(x^{k+2}) & \cdots & L_1(x^{n+k+1}) \\ \cdots & \cdots & \cdots & \cdots \\ L_n(x^{k+1}) & L_n(x^{k+2}) & \cdots & L_n(x^{n+k+1}) \end{vmatrix} \neq 0 \quad \forall k, n \in \mathbb{N}.$$

On construit ainsi toute une table d'approximants  $(k + 1/n)_f^G$ ,  $k \geq -1$ ,  $n \geq 0$ , qui vérifient aussi un problème d'interpolation : en effet, si on pose

$$x^{k+1}G_{k+1}(x, t) = G(x, t) - \sum_{i=0}^k x^i g_i(t) = \sum_{i=k+1}^{\infty} x^i g_i(t),$$

et si  $q_n(x, t)$  est le polynôme de degré  $n$  qui vérifie les conditions d'interpolation

$$L_i(q_n(x, t)) = L_i(G_{k+1}(x, t)) \quad i = 0, \dots, n,$$

on montre que  $(k + 1/n)_f^G$  peut s'obtenir en remplaçant, dans  $c(G(x, t))$ ,  $G$  par le polynôme  $q_n$ .

Les conditions d'existence et unicité de ces approximants ont été étudiées dans [24] et un programme en MATHEMATICA pour le calcul récursif de ces approximants a été développé dans [97]. Les bons résultats numériques obtenus ont motivé notre étude de la convergence de ces approximants. Nous nous sommes intéressés uniquement à la convergence de suites de la forme  $((n)_f^G(t))_n = ((0/n)_f^G)_n$ . L'étude de différentes suites dans la table définie ci-dessus (suites diagonales, convergence des lignes ...) reste un problème ouvert. Néanmoins, quelques uns des résultats obtenus restent valables pour des suites  $((k + 1/n)_f^G)_n$  ( $k > -1$ ) (car si on impose à  $G(\cdot, t)$  d'être analytique dans un disque  $D(0, \beta)$ , cette condition sera aussi satisfaite par  $G_k(\cdot, t)$ ).

On va donc étudier la convergence des suites  $((n)_f^G(t))_n$  de ATPG pour deux choix de formes linéaires  $L_i$  :

- (i)  $L_i(f) = f(x_i)$ , où  $(x_i)$  est une suite arbitraire de points (si le point est répété on considère les dérivées) ;
- (ii)  $L_i(f) = \int_C f(z) \overline{p_i(z)} w(z) |dz|$ , où  $\{p_i(z)\}$  est la famille de polynômes orthonormaux sur  $C$  par rapport à la fonction poids  $w(z)$ .

Nous donnerons des conditions sur la fonction génératrice  $G$  et sur la forme linéaire  $c$  pour que l'on ait  $\lim_{n \rightarrow \infty} (n)_f^G(t) = f(t)$  pour  $t$  dans un sous-ensemble de  $\mathbb{C}$ . Nous illustrerons ces résultats par quelques exemples dans le cas où  $G$  est la fonction génératrice d'une famille de polynômes orthogonaux classiques.

Nous obtiendrons ensuite, pour le choix des fonctionnelles **(i)**, une représentation intégrale de l'erreur qui permettra d'obtenir des bornes pour l'erreur des suites  $((n)_f^G(t))_n$ . Nous considérerons différents choix de  $G$  et obtiendrons les résultats de convergence correspondants.

## 2.4.2 Résultat général de convergence

De la définition des ATPG, on voit que pour étudier la convergence d'une suite d'approximants, il suffit d'obtenir des conditions de convergence d'une suite de polynômes d'interpolation  $P_n(x; t)$  et de continuité de la forme  $c$ . La forme linéaire  $c$  est définie sur l'espace des polynômes, mais on peut l'étendre, par continuité, à des espaces plus grands. On va définir trois espaces fonctionnels normés pour lesquels  $c$  est définie et continue. Mais d'abord on a besoin de la propriété suivante :

**DEFINITION 2.15.** *La forme linéaire  $c$  vérifie la propriété  $(\mathcal{P}_B)$  où  $B$  est un intervalle de  $\mathbb{R}$  ou une région de  $\mathbb{C}$  ssi pour toute suite  $(p_n)_n$  de polynômes ( $p_n$  polynôme de degré  $n$ ) on a l'implication suivante :*

$$\lim_{n \rightarrow \infty} p_n = 0 \text{ uniformément dans } B \quad \Rightarrow \quad \lim_{n \rightarrow \infty} c(p_n) = 0$$



Cette propriété est vérifiée, par exemple, si les moments  $c_i = c(x^i)$  vérifient  $c_i = \int_a^b f(x)q_i(x)w(x)dx$  avec  $\{q_i(x)\}$  la suite de polynômes orthonormaux dans  $[a, b]$  par rapport à la fonction poids  $w(x)$ .

Soit  $c$  une forme linéaire donnée par ses moments  $c(x^i) = c_i$ ,  $i \geq 0$ . Selon les propriétés de  $c$ , on va définir trois espaces dans lesquels on peut montrer que  $c$  est continue (voir [A7]).

1.  $c$  vérifie  $\mathcal{P}_B$  pour  $B = I = [a, b]$

On définit  $\mathcal{E}_1(I) = \{ f : [a, b] \rightarrow \mathbb{R}$  limite uniforme dans  $[a, b]$  d'une suite de polynômes  $(p_n)$  telle que  $\lim_{n \rightarrow \infty} c(p_n)$  existe  $\}$  et on considère la norme du max sur  $I$ ;

2.  $\limsup_{n \rightarrow \infty} |c_n|^{1/n} = r < \infty$

On définit, pour  $R > r$  l'espace

$\mathcal{E}_2(R) = L^2(D_R) =$  l'espace des fonctions analytiques dans  $D_R = \{z : |z| \leq R\}$ , telles que  $\int_{D_R} |f|^2 dx dy < \infty$  avec la norme  $L^2$ ;

3.  $\limsup_{n \rightarrow \infty} |c_n|^{1/n} = \infty$ ,  $c$  vérifie  $(\mathcal{P}_B)$  sur  $D_R$

On définit  $\mathcal{E}_3(R) =$  l'espace des fonctions entières  $\sum_{n=0}^{\infty} a_n z^n$  pour lesquelles la série  $\sum_{i=0}^{\infty} a_i c_i$  converge. On prend comme norme  $\|g\| = \sup_{|z| \leq R} |g(z)|$ .

On obtient alors le résultat général de convergence suivant :

**THÉORÈME 2.16.** [A7] *Soit  $f$  une fonction analytique dans  $A \subset \mathbb{C}$  représentée par la série de fonctions*

$$f(z) = \sum_{i=0}^{\infty} c_i g_i(z) \quad z \in A, \quad (2.8)$$

*et  $G(x, t)$  une fonction génératrice de la famille  $\{g_i(z)\}$ . On définit la forme linéaire  $c$  par ses moments :  $c(x^i) = c_i$ ,  $i \geq 0$ . On considère une suite de polynômes de degré  $\leq n$  en  $x$ ,  $P_n(x; t)$  vérifiant les conditions d'interpolation*

$$L_i(P_n(x; t)) = L_i(G(x, t)) \quad i = 0, \dots, n.$$

*Soit  $\mathcal{E}$  l'un des espaces fonctionnels normés considérés ci-dessus ( $\mathcal{E}_1, \mathcal{E}_2$  ou  $\mathcal{E}_3$ ). Alors, si pour  $t \in A_* \subset A$ ,*

- la fonction génératrice  $G(\cdot, t)$  appartient à  $\mathcal{E}$ ;
- la suite  $\{P_n(\cdot; t)\}$  converge vers  $G(\cdot, t)$  dans  $\mathcal{E}$

*alors la suite correspondante de ATPG converge vers  $f$  dans  $A_*$ , i.e.*

$$\forall t \in A_* \quad \lim_{n \rightarrow \infty} (n)_f^G(t) = f(t).$$

### 2.4.3 Résultats de convergence pour différents types de fonctionnelles d'interpolation

On va utiliser des résultats de convergence de la théorie de l'interpolation et le théorème 2.16 pour obtenir des résultats de convergence pour des suites particulières de ATPG. Étant

données une fonction génératrice  $G$  d'une famille  $\{g_i(n)\}_{i \in \mathbb{N}}$ , une fonction  $f$  développée dans la famille  $\{g_i(n)\}_{i \in \mathbb{N}}$  et la forme linéaire associée  $c$ , on va déterminer, pour différents types de fonctionnelles d'interpolation  $L_k$ , les conditions à imposer à  $G$  et  $c$  pour que la suite  $((n)_f^G(t))_n$  converge vers  $f$ . On étudiera deux types de formes  $L_k$ .

## 1. Interpolation de Lagrange et de Hermite

On considèrera trois choix pour les abscisses d'interpolation

### (a) série d'interpolation

THÉOREME 2.17. [A7] Soient  $\zeta_1, \zeta_2, \dots, \zeta_k$  des points de  $\mathbb{C}$  et  $\mathcal{L}_\rho$  la lemniscate définie par

$$|(z - \zeta_1)(z - \zeta_2) \cdots (z - \zeta_k)| < \rho^k.$$

Soit  $f$  une fonction analytique de la forme

$$f(z) = \sum_{i=0}^{\infty} c_i g_i(z) \quad z \in A,$$

où  $c$  est la forme linéaire associée et  $G(x, t)$  la fonction génératrice correspondante. Supposons que

$$(i) \limsup_{n \rightarrow \infty} |c_n|^{1/n} = r < \infty;$$

$$(ii) \forall t \in A_* \subset A, G(\cdot, t) \text{ est analytique dans } \mathcal{L}_\rho \supseteq D_{R_*}, \quad R_* > r.$$

Si l'on considère une suite de points  $(z_i)_i$  vérifiant

$$\lim_{n \rightarrow \infty} z_{nk+i} = \zeta_i, \quad 1 \leq i \leq k; \quad z_i \in \mathcal{L}_\rho \quad \forall i,$$

et si  $P_n(x; t)$  est le polynôme de degré  $\leq n$  qui interpole  $G$  aux points  $z_i$ ,  $i = 1, \dots, n+1$  ( $t$  considéré comme un paramètre), alors la suite des ATPG correspondante converge vers  $f$  dans  $A_*$ .

Quelle forme ont ces ATPG ?

– si les points d'interpolation sont tous distincts, on obtient par la formule de Lagrange

$$c(P_n(\cdot, t)) = \sum_{i=0}^n c(l_i) G(\zeta_i, t)$$

et donc les approximants sont des combinaisons linéaires des fonctions  $G(\zeta_i, t)$ ,  $i = 1, \dots, n+1$ ,

– si tous les points d'interpolation coïncident, les lemniscates sont réduites à des cercles et  $P_n(x; t)$  est le polynôme de Taylor  $P_n(x; t) = \sum_{i=0}^n \frac{\partial^i}{\partial x^i} (G(x, t)) \big|_{x=\zeta} \frac{x^i}{i!}$ . Dans ce cas, les

approximants sont des combinaisons linéaires des fonctions  $\left\{ \frac{\partial^i}{\partial x^i} (G(\zeta, t)) \right\}_{i=1}^{n+1}$ .

On remarque encore une fois que si  $g_i(t) = t^i \quad \forall i$ , la fonction génératrice est donnée par  $G(x, t) = 1/(1 - xt)$  et l'approximant sera une fonction rationnelle - un approximant de type Padé.

### (b) ensemble triangulaire de points d'interpolation

On considère maintenant le cas plus général où les fonctionnelles  $L_k$  dépendent de  $n$ , i.e. on a l'ensemble suivant de points d'interpolation

$$\begin{cases} \beta_1^{(0)} \\ \beta_1^{(1)}, \beta_2^{(1)} \\ \dots \quad \dots \quad \dots \\ \beta_1^{(n)}, \beta_2^{(n)}, \dots, \beta_{n+1}^{(n)} \\ \dots \quad \dots \quad \dots \end{cases} \quad (2.9)$$

Un résultat connu de convergence est le suivant :

**THÉORÈME 2.18.** [115] *Soit  $C$  un ensemble de points fermé borné dont le complémentaire est connexe régulier. Soit  $x = \Phi(z)$  la fonction qui transforme  $K$  dans  $|w| > 1$  de façon à ce que les points à l'infini correspondent, et soit  $\Delta$  la capacité de  $C$ . Supposons que les points (2.9) n'aient pas de point limite extérieur à  $C$  et vérifient la relation*

$$\lim_{n \rightarrow \infty} \left| (z - \beta_1^{(n)})(z - \beta_2^{(n)}) \cdots (z - \beta_{n+1}^{(n)}) \right|^{1/(n+1)} = \Delta |\phi(z)|,$$

uniformément dans chaque sous-ensemble fermé borné dans  $K$ .

Soit  $f$  une fonction analytique dans  $C$ . Alors la suite de polynômes  $p_n(z)$  de degrés respectifs  $n$  interpolant  $f(z)$  aux points  $\beta_1^{(n)}, \dots, \beta_{n+1}^{(n)}$  vérifie

$$\lim_{n \rightarrow \infty} p_n(z) = f(z) \quad \text{uniformément pour } z \in C.$$

Si l'on considère différents choix particuliers pour les suites de points (2.9) et différentes fonctions génératrices, en appliquant le théorème (2.16) on obtient de nouveaux résultats de convergence pour des suites de ATPG. Regardons, par exemple, le choix des  $\beta_i^{(n)}$  comme racines de polynômes orthogonaux classiques. D'autres cas sont considérés dans [A7].

**COROLLAIRE 2.19.** [A7] *Soient  $f$  une fonction analytique dans un domaine  $A$  donnée par  $f(t) = \sum_{n=0}^{\infty} c_n t^n$ ,  $t \in A$ ,  $G$  la fonction génératrice et  $c$  la forme linéaire associées. Supposons que*

- pour  $t \in A_* \subset A$ , la fonction génératrice  $G(\cdot, t)$  est continue dans  $[-1, 1]$  avec module de continuité  $w(\delta)$  vérifiant  $w(\delta) = o(|\log \delta|^{-1})$ ;
- $c$  vérifie  $\mathcal{P}_{[-1,1]}$ ;
- pour  $t \in A_*$ ,  $G(\cdot, t)$  appartient à  $\mathcal{E}_1([-1 + \epsilon, 1 - \epsilon])$  ( $\epsilon < 1/2$ ).

Alors si les  $L_k$  sont les fonctionnelles d'interpolation aux racines des polynômes de Jacobi  $P_n^{(\alpha, \beta)}(x)$  ( $\alpha, \beta$  fixés), la suite correspondante de ATPG vérifie

$$\forall t \in A_* \quad \lim_{n \rightarrow \infty} (n)_f^G(t) = f(t).$$

### (c) distribution régulière des points d'interpolation $\beta_k^{(n)}$

Supposons que les points  $\beta_k^{(n)}$  vérifient

$$\lim_{n \rightarrow \infty} \left| (z - \beta_1^{(n)}) \cdots (z - \beta_{n+1}^{(n)}) \right|^{1/(n+1)} = \sigma(z),$$

pour  $z$  dans un certain ensemble de  $\mathbb{C}$ . Par exemple, soient  $\beta_1^{(n)}, \dots, \beta_{n+1}^{(n)}$  les  $n+1$  racines du polynôme de Tchebychev  $T_{n+1}(z)$ . Alors

$$T_{n+1}^*(z) = (z - \beta_1^{(n)}) \cdots (z - \beta_{n+1}^{(n)}) = \frac{1}{2^n} T_{n+1}(z),$$

et l'on peut montrer que [43]

$$\sigma(z) = \lim_{n \rightarrow \infty} |T_n^*(z)|^{1/n} = \rho/2 \quad \text{pour } z \in \mathcal{F}_\rho \text{ uniformément,} \quad (2.10)$$

où  $\mathcal{F}_\rho$  est l'ellipse de foyers  $\pm 1$  et demi-axes  $a = (\rho + \rho^{-1})/2$  et  $b = (\rho - \rho^{-1})/2$ . On peut montrer en utilisant un résultat de la théorie de l'interpolation et le théorème 2.16 que

**COROLLAIRE 2.20.** [A7] Soient  $f(t) = \sum_{n=0}^{\infty} c_n g_n(t)$ ,  $t \in A$ , une fonction analytique,  $G(x, t)$  la fonction génératrice des  $\{g_n(t)\}_n$  et  $c$  la forme linéaire associée. On suppose que

- $c$  vérifie  $\limsup_{n \rightarrow \infty} |c_n|^{1/n} = r$ ,
- $\forall t \in A_* \subset A$ ,  $G(\cdot, t)$  est analytique dans une région  $S$  qui contient une ellipse  $\mathcal{F}_\rho$  ( $\rho > 1$ ), et on note  $E_\rho$  l'intérieur de  $\mathcal{F}_\rho$ ,
- $\exists R > r : D_R \subset E_\rho$ .

Si l'on choisit comme points d'interpolation  $\beta_k^{(n)}$  les racines des polynômes de Tchebychev, la suite correspondante de ATPG,  $((n)_f^G(t))_n$  converge vers  $f$  dans  $A_*$ .

On obtient un résultat similaire si on utilise les racines des polynômes de Legendre car  $\lim_{n \rightarrow \infty} |P_n(z)|^{1/n} = \rho \quad \forall z \in \mathcal{F}_\rho$  ( $P_n(z)$  est le  $n$ -ème polynôme de Legendre).

Regardons quelques exemples d'application des théorèmes précédents.

(i) fonction donnée par une série de Legendre

Soit  $f(z)$  la série orthogonale

$$f(z) = \sum_{n=0}^{\infty} c_n P_n(z), \quad P_n(z) \text{ polynômes de Legendre,} \quad (2.11)$$

et  $c_n = \int_{-1}^1 f(x) P_n(x) dx$ ,  $n = 0, 1, \dots$ . Supposons que  $\lim_{n \rightarrow \infty} \sup |c_n|^{1/n} = 1/\rho$ . Alors le développement en série converge pour  $t$  dans l'ellipse  $\mathcal{F}_\rho$ . Une fonction génératrice des polynômes de Legendre est

$$G(x, t) = \sum_{n=0}^{\infty} x^n P_n(t) = (1 - 2tx + x^2)^{-1/2}.$$

Pour  $t \in \mathcal{F}_\rho$ ,  $G(x, t)$  est analytique dans  $D_R$ , pour tout  $R$  inférieur ou égal à 1 et donc appartient à  $\mathcal{E}_2(R)$ . On choisit  $R_*$  tel que  $1 > R_* > 1/\rho$ , et une suite de points d'interpolation  $(x_i)_i$  dans  $D_{R_*}$  convergeant vers 0. On remplace  $G(x, t)$  par les polynômes d'interpolation de Lagrange  $P_n(x; t) = \sum_{i=0}^n l_i(x) (1 - 2tx_i + x_i^2)^{-1/2}$  et donc  $f(t)$  est approchée par

$$(n)_f^G(t) = c(P_n(\cdot; t)) = \sum_{i=0}^n c(l_i(\cdot)) (1 - 2tx_i + x_i^2)^{-1/2}.$$

En appliquant le théorème précédent on montre que cette suite converge vers  $f(t)$  pour  $t \in \mathcal{F}_\rho$ .

(i) fonction donnée par une série d'Hermite

Soit  $f$  donnée par la série d'Hermite

$$f(t) = \sum_{n=0}^{\infty} c_n H_n(t) \quad t \in A, \text{ avec } c_n = \int_{-\infty}^{+\infty} f(x) e^{-x^2} H_n(x) dx.$$

Une fonction génératrice des polynômes d'Hermite est

$$G(x, t) = \sum_{n=0}^{\infty} \frac{1}{n!} x^n H_n(t) = e^{2xt-x^2}.$$

Pour  $t \in \mathbb{C}$ ,  $G(\cdot, t)$  est une fonction entière. Dans ce cas on montre que  $G(\cdot, t)$  appartient à  $\mathcal{E}_3(R)$ ,  $R > 0$ . Si  $(x_i)$  est une suite de points d'interpolation vérifiant  $\lim_{n \rightarrow \infty} x_n = 0$ ,  $x_n \in D_R \quad \forall n$ , alors la suite correspondante d'approximants est une combinaison linéaire d'exponentielles

$$(n)_f^G(t) = c(P_n(\cdot, t)) = \sum_{i=0}^n e^{-x_i^2} c(l_i(\cdot)) e^{2x_i t}.$$

Cette suite de ATPG converge vers  $f(t)$  dans  $A$  si  $c$  vérifie  $\mathcal{P}$ .

## 2. Développements orthogonaux

On considère maintenant un second type de fonctionnelles d'interpolation, de la forme

$$L_k(g) = \int_C g(z) \overline{p_k(z)} w(z) |dz|,$$

où  $C$  est une courbe de Jordan rectifiable,  $w(z)$  est une fonction réelle, non négative et uniformément bornée dans  $C$ , et  $\{p_k(z)\}$  la suite des polynômes orthonormaux sur  $C$  par rapport à la fonction poids  $w(z)$ . Dans ce cas le polynôme d'interpolation  $P_n(x; t)$  est donné par

$$P_n(x; t) = \sum_{k=0}^n a_k p_k(x), \quad a_k = \int_C G(z, t) \overline{p_k(z)} w(z) |dz|,$$

et c'est le polynôme de degré  $n$  de meilleure approximation de  $G(x, t)$  (fonction de  $x$ ) au sens des moindres carrés, c'est-à-dire, celui qui minimise

$$\int_C |G(x, t) - p_n(x; t)|^2 w(z) |dz|.$$

Donc la convergence de la suite  $(P_n(\cdot; t))_n$  vers  $G(\cdot, t)$  dépend de la convergence des développements orthogonaux. En utilisant des résultats de [115] combinés avec le théorème 2.16 on obtient :

**COROLLAIRE 2.21. [A7]** Soient  $f$  une fonction analytique donnée par  $f(z) = \sum_{i=0}^{\infty} c_i g_i(z)$ ,  $G(x, t)$  la fonction génératrice et  $c$  la forme linéaire associée. Supposons que :

- pour  $t \in A_* \subset A$ , la fonction  $G(x, t)$  de  $x$  est analytique dans  $[-1, 1]$  et l'on note  $\mathcal{F}_\rho$  la plus grande ellipse de foyers  $\pm 1$  pour laquelle  $G(\cdot, t)$  est analytique pour  $t \in A_*$  ;

- $c$  vérifie  $\limsup_{n \rightarrow \infty} |c_n|^{1/n} = r < \infty$
- on peut choisir  $R_* > r$  tel que  $D_{R_*} \subset \mathcal{F}_\rho$ .

On considère le polynôme d'interpolation de degré  $n$  en  $x$ ,  $P_n(x; t)$ , avec

$$L_k(g) = \int_{-1}^1 g(x) \frac{P_n^{(\alpha, \beta)}(x)}{h_n^{(\alpha, \beta)}} (1-x)^\alpha (1+x)^\beta dx,$$

(i.e., la somme partielle d'ordre  $n$  du développement de  $G(\cdot, t)$  en série de Jacobi). Alors on obtient

$$\forall t \in A_* \quad \lim_{n \rightarrow \infty} (n^G)_f(t) = f(t).$$

Considérons un exemple d'application de ce corollaire.

Soit  $f$  une fonction analytique représentée par  $f(t) = \sum_{i=0}^{\infty} c_i t^i$  avec  $\lim_{n \rightarrow \infty} |c_n|^{1/n} = r < 1$ . Alors la fonction génératrice est donnée par  $G(x, t) = \sum_{i=0}^{\infty} x^i t^i = \frac{1}{1-xt}$ ,  $x \neq \frac{1}{t}$ . Pour  $|t| < \alpha < 1$ , on a  $1/|t| > 1/\alpha$ , et donc  $G(x, t)$  (fonction de  $x$ ) est analytique pour  $|x| < 1/\alpha$ . On remplace  $G(x, t)$  par

$$P_n(x; t) = a_0 p_0(x) + a_1 p_1(x) + \cdots + a_n p_n(x)$$

où les  $\{p_i(x)\}$  sont les polynômes de Legendre. La suite correspondante de ATPG est donnée par

$$c(P_n(\cdot, t)) = \sum_{i=0}^n a_i c(p_i(\cdot)), \text{ avec } a_i = \int_{-1}^1 \frac{\overline{p_i(z)}}{1-zt} dz.$$

Par des calculs simples on obtient

$$c(P_n(\cdot; t)) = \sum_{k=0}^n \alpha_k R_{\log}^k(t),$$

avec  $R_{\log}^i(t) = \frac{\log(\frac{1+t}{1-t}) - S_i(t)}{t^{i+1}}$ ,  $S_i(t)$  la somme partielle de la série de Taylor de  $\log\left(\frac{1+t}{1-t}\right)$ . On montre que pour  $|t| < \alpha < 1$ , les conditions du corollaire précédent sont satisfaites et la suite de ATPG converge vers  $f(t)$ .

Les résultats précédents sont très généraux et ouvrent la voie d'une étude plus détaillée dans des cas particuliers. Pour des fonctions données par leur développement en série de polynômes orthogonaux, le choix des points d'interpolation de façon à obtenir de meilleurs résultats de convergence reste un problème ouvert.

#### 2.4.4 Représentation intégrale de l'erreur

Dans le cas où les fonctionnelles  $L_k$  sont de la forme

$$L_i(G(\cdot, t)) = G(z_{ni}, t) \text{ pour } i = 0, \dots, n,$$

on peut, en utilisant des résultats d'analyse complexe et des résultats de la théorie d'interpolation, obtenir une représentation intégrale de l'erreur pour des fonctions génératrices vérifiant

certaines conditions, ce qui nous permettra d'obtenir des bornes sur la vitesse de convergence de certaines suites de ATPG.

Considérons donc comme précédemment une fonction  $f(z)$  définie dans un domaine  $\mathcal{D} \subset \mathbb{C}$  par un développement en série

$$\forall z \in \mathcal{D} : f(z) = \sum_{n=0}^{\infty} c_n g_n(z) \text{ avec } \limsup_{n \rightarrow \infty} |c_n|^{1/n} = \frac{1}{R}, R > 1, \quad (2.12)$$

et soit  $G(x, z) = \sum_{i=0}^{\infty} x^i g_i(z)$  la fonction génératrice de la famille  $\{g_n(z)\}$ . On associe à  $f$  une fonction  $g$  définie par la série entière

$$g(t) = \sum_{n=0}^{\infty} c_n t^n, \text{ qui est analytique pour } |t| < R.$$

On définit la forme linéaire  $c$  dans l'espace  $\mathcal{H}_\alpha$  des fonctions holomorphes dans  $D_{1/\alpha} = D(0, 1/\alpha)$  par :

$$\forall k \in \mathcal{H}_\alpha : c(k) = \frac{1}{2\pi i} \int_{|x|=r} g(x) k(1/x) \frac{dx}{x} \text{ avec } \alpha < r < R. \quad (2.13)$$

Il est facile de voir que  $c(x^n) = c_n, \forall n \geq 0$ . De plus, si  $A \subset \mathcal{D}$  est un domaine tel que pour  $z \in A, G(\cdot, z) \in \mathcal{H}_\alpha$  pour  $\alpha > 0$ , alors, pour  $z \in A$  on obtient, par des arguments de convergence uniforme, que

$$c(G(x, z)) = \sum_{n=0}^{\infty} c_n g_n(z) = f(z).$$

Si  $q_n(\cdot, z)$  est le polynôme de degré au plus  $n$  qui interpole  $G(\cdot, z)$  dans l'ensemble de points  $\{z_{ni}\}_{i=0}^n$ , alors

$$(n)_f^G(z) = c(q_n(x, z)) = \frac{1}{2\pi i} \int_{|x|=r} g(x) q_n\left(\frac{1}{x}, z\right) \frac{dx}{x},$$

et l'erreur des ATPG s'écrit

$$\forall z \in A \quad f(z) - (n)_f^G(z) = \frac{1}{2\pi i} \int_{|x|=r} g(x) \left[ G\left(\frac{1}{x}, z\right) - q_n\left(\frac{1}{x}, z\right) \right] \frac{dx}{x}.$$

La vitesse de convergence peut ainsi être mesurée par la vitesse de convergence des polynômes d'interpolation et en utilisant la formule de l'erreur d'interpolation [43] on obtient

$$e_n(z) = f(z) - (n)_f^G(z) = -\frac{1}{4\pi^2} \int_{|x|=r} g(x) \int_{\Gamma} \frac{(x^{-1} - z_{n0}) \cdots (x^{-1} - z_{nn})}{(t - z_{n0}) \cdots (t - z_{nn})} \frac{G(t, z)}{xt - 1} dt \, dx. \quad (2.14)$$

En considérant des abscisses d'interpolation  $(z_{ni})_{i=0}^n$  ayant un certain comportement à la limite on peut obtenir des majorations d'erreur, comme le théorème suivant :

**THÉORÈME 2.22. [A8]** Soit  $f(z) = \sum_{n=0}^{\infty} c_n g_n(z), z \in \mathcal{D} \subset \mathbb{C}$  avec  $\lim_{n \rightarrow \infty} |c_n/c_{n+1}| = R$ . Soit  $G(\cdot, z)$  la fonction génératrice de la famille  $\{g_n(z)\}$ . On considère les suites de ATPG,  $((n)_f^G(z))_n$ , correspondantes à des fonctionnelles d'interpolation aux points  $\{z_{ni}, i = 0, \dots, n\}_n$ . On suppose que :

(H1) pour  $z \in A \subset \mathcal{D}$ ,  $G(\cdot, z)$  est analytique dans  $D_{1/\alpha}$  (pour un  $\alpha > 0$  donné).

(H2)  $\{z_{ni}, i = 0, \dots, n\}_{n \in \mathbb{N}} \subset D_{1/\alpha}$  vérifient

$$\lim_{n \rightarrow \infty} |(z - z_{n0}) \cdots (z - z_{nn})|^{1/(n+1)} = \sigma(z) \quad \text{pour tout } z \in D(0, 1/\alpha). \quad (2.15)$$

(H3)  $C_\rho = \{z : \sigma(z) = \rho\}$  est une courbe simple, fermée, rectifiable et l'on pose

$$\begin{aligned} \rho_M &= \sup \{ \rho : C_\rho \subset D_{1/\alpha}; z_{ni} \in \text{Int}(C_\rho), \forall i, n \}, \\ \rho_m &= \inf \{ \rho : C_\rho \subset \text{Int}(C_{\rho_M}), C_\rho \subset \text{Ext}(D_{1/R}) \}. \end{aligned}$$

( $\text{Int}(C_\rho)$  l'intérieur de  $C_\rho$  et  $\text{Ext}(C_\rho)$  l'extérieur de  $C_\rho$ )

Alors

$$\forall z \in A : \limsup_{n \rightarrow \infty} |f(z) - (n)_f^G(z)|^{1/(n+1)} \leq \frac{\rho_m}{\rho_M}.$$

Regardons quels résultats on peut obtenir sur la convergence des ATPG à partir de ce théorème général en considérant deux suites différentes de points d'interpolation vérifiant (2.15).

(1) Premier choix :

**COROLLAIRE 2.23. [A8]** Soient  $f$  et  $G$  dans les conditions du théorème 2.22. On construit la suite de ATPG correspondant au choix suivant de points d'interpolation :

$$z_{ni} = z_n \quad i = 0, \dots, n \quad \text{avec} \quad \lim_{n \rightarrow \infty} z_n = 0.$$

Alors

$$\lim_{n \rightarrow \infty} |f(z) - (n)_f^G(z)|^{1/(n+1)} \leq \frac{\alpha}{R}.$$

(2) Deuxième choix : Soient  $\{z_{ni}\}_{i=0}^n$  les racines de  $p_{n+1}(\mu, z)$ , pour  $n \in \mathbb{N}$ , avec  $\{p_n(z)\}$  une suite de polynômes orthogonaux par rapport à une mesure de Borel  $\mu$  à support compact et vérifiant  $\lim_{n \rightarrow \infty} |p_n(\mu, z)|^{1/n} = e^{g_\Omega(z, \infty)}$  localement uniformément pour  $z \in \mathbb{C} \setminus \text{Co}(S(\mu))$  (avec  $g_\Omega(z, \infty)$  la fonction de Green avec pôle à l'infini de  $\Omega$  composante non bornée de  $\mathbb{C} \setminus \text{supp}(\mu)$ ).

Alors, dans certains cas particuliers on peut, en utilisant des résultats sur le comportement asymptotique des suites de polynômes orthogonaux [112], déterminer la forme de  $C_\rho$  et calculer les valeurs de  $\rho_m$  et  $\rho_M$ . Cela a été fait dans [A8] en considérant différentes fonctions génératrices  $G$ .

Regardons, par exemple, un développement en série de Legendre. On a

$$G(x, z) = \frac{1}{\sqrt{1 - 2xz + x^2}} = \sum_{n=0}^{\infty} x^n P_n(z).$$

Si l'on fixe  $z \in \mathcal{E}_\rho$  ( $\rho > 1$  ( $\mathcal{E}_\rho$  l'ellipse de foyers  $\pm 1$  et demi-axes  $a = \frac{1}{2}(\rho + \rho^{-1})$ ,  $b = \frac{1}{2}(\rho - \rho^{-1})$ )),  $G(\cdot, z)$  est analytique dans  $x \in D(0, 1/\rho)$ , et donc en appliquant le théorème 2.22 on obtient



COROLLAIRE 2.24. [A8] Soit  $f$  donnée par

$$f(z) = \sum_{n=0}^{\infty} a_n P_n(z) \text{ avec } \limsup_{n \rightarrow \infty} |a_n|^{1/n} = \frac{1}{R} \quad (R > 1). \quad (2.16)$$

On fixe  $\rho$  et  $\rho_*$  vérifiant  $\rho < R$ ,  $\rho_* > \rho$  et on choisit les points d'interpolation de la façon suivante :

$$z_{ni} = y_{ni}/\rho_* \quad i = 0, \dots, n, \quad (2.17)$$

où les  $y_{ni}$  sont les racines du polynôme de Tchebychev de degré  $n+1$ ,  $T_{n+1}(z)$ . Alors la suite correspondante de ATPG a le comportement asymptotique suivant

$$\limsup_{n \rightarrow \infty} |f(z) - (n)G_f(t)(z)|^{1/n} \leq \frac{\rho}{R} \left( 1 + \sqrt{1 + \left(\frac{R}{\rho_*}\right)^2} \right) / \left( 1 + \sqrt{1 - \left(\frac{\rho}{\rho_*}\right)^2} \right) \quad \forall z \in \mathcal{E}_\rho. \quad (2.18)$$

Après avoir fixé une valeur de  $\rho_*$  supérieure ou égale à  $R$ , ce résultat permet de déterminer l'ensemble des points pour lesquels on a convergence géométrique de la suite des approximants : par exemple, si l'on prend  $\rho_* = R$  on a convergence géométrique pour  $z \in \mathcal{E}_\rho$  avec  $\rho \leq R/(1+\sqrt{2})$ .

**Remarque :** un résultat similaire est obtenu si l'on prend comme  $\{y_{ni}\}$  les racines des polynômes de Legendre.

D'autres cas (développement en série de Tchebychev, Laguerre, Hermite) ont été étudiés dans [A8] où l'on a obtenu des résultats sur la vitesse de convergence de suites  $((n)G_f(z))_n$ .

### 2.4.5 ATPG pour des fonctions de Stieltjes généralisées. Relation avec les approximants de Baker-Gammel

On va maintenant obtenir des résultats de convergence pour des fonctions de Stieltjes généralisées, c'est-à-dire, des fonctions  $f$  ayant la forme suivante :

$$f(z) = \int_{\Delta} G(t, z) d\alpha(t), \quad (2.19)$$

où  $\alpha$  est une fonction bornée, non décroissante, avec un nombre infini de valeurs différentes sur  $\Delta$  intervalle compact de  $\mathbb{R}^+$ . Dans le cas particulier où  $G(t, z) = (1 + tz)^{-1}$ ,  $f$  est alors une fonction de Stieltjes. Si  $G(t, z)$  est la fonction génératrice de la famille  $\{g_i(z)\}_{i=0}^{\infty}$  on a formellement

$$f(z) = \int_{\Delta} \left( \sum_{i=0}^{\infty} t^i g_i(z) \right) d\alpha(t) = \sum_{i=0}^{\infty} \left( \int_{\Delta} t^i d\alpha(t) \right) g_i(z) = \sum_{i=0}^{\infty} c_i g_i(z).$$

Supposons que pour  $z \in C$  une région compacte de  $\mathbb{C}$ ,  $G(\cdot, z)$  soit analytique dans  $A = \mathbb{R}^+ \times ]-a, a[$ . Alors, en utilisant la représentation de Cauchy, on obtient

$$f(z) = \frac{1}{2\pi i} \int_{\Gamma} G(x, z) \int_{\Delta} \frac{d\alpha(t)}{x-t} dx = \frac{1}{2\pi i} \int_{\Gamma} G(x, z) g(-1/x) \frac{dx}{x}, \quad (2.20)$$

où  $\Gamma$  est un contour simple fermé de  $A$  qui contient  $\Delta$  dans son intérieur et  $g(x) = \int_{\Delta} \frac{d\alpha(t)}{1+xt}$  est une fonction de Stieltjes.

Une façon de construire des approximants pour des fonctions de Stieltjes généralisées qui soient des combinaisons linéaires des fonctions  $G(u_i, z)$  ( $u_i \in A$ ) et  $g_i^*(z) = \frac{\partial^i}{\partial u^i} G(u, z) |_{u=0}$  a été proposée par Baker et Gammel : les *approximants de Baker-Gammel de  $f$*  [4] consistent à remplacer  $g$  par ses approximants de Padé. On obtient un approximant de la forme

$$G^{[m+j/m]}(z) = \sum_{i=0}^j \beta_i g_i^*(z) + \sum_{i=1}^m \alpha_i G(u_i, z). \quad (2.21)$$

On montre facilement (voir [4]) que leur développement en termes des  $\{g_i(z)\}$  coïncide avec celui de  $f$  jusqu'à l'ordre  $2m+j$ , et un résultat de convergence peut être obtenu à partir des résultats de convergence des approximants de Padé des fonctions de Stieltjes. La construction de ces approximants nécessite le calcul des pôles et des résidus de  $[m+j/m]_f(z)$ , ce qui implique une grande quantité de calculs.

L'approche des ATPG consiste à remplacer  $G(\cdot, z)$  dans (2.20) par son polynôme d'interpolation, au lieu de remplacer  $g$  par son approximant de Padé. Comme vu précédemment, la forme générale de ces approximants contient (2.21) pour le choix de points d'interpolation suivant :  $u_i$ ,  $i = 1, \dots, m$  et  $u_0$  répété  $j+1$  fois. L'ordre d'approximation est  $m+j$  dans le cas des ATPG, et est donc inférieure à celui des approximants de Baker-Gammel, mais on obtient une diminution importante des calculs.

On a obtenu des bornes supérieures sur le comportement asymptotique des erreurs des ATPG pour des fonctions de Stieltjes généralisées, en considérant des choix particuliers des points d'interpolation. En utilisant des résultats fondamentaux d'analyse complexe et des théorèmes de convergence précédents on peut montrer :

THÉORÈME 2.25. [A8] Soit  $f$  de la forme

$$f(z) = \int_{\Delta} G(t, z) d\alpha(t), \quad (2.22)$$

avec  $G(\cdot, z)$  analytique dans  $A = \mathbb{R}^+ \times ]-a, a[$  pour  $z \in C$  compact de  $\mathbb{C}$ . On considère une suite de points d'interpolation  $\{z_{ni}\}_{i=0}^n \subset A$  vérifiant

$$\lim_{n \rightarrow \infty} \left| \prod_{i=0}^n (z - z_{ni}) \right|^{1/(n+1)} = \sigma(z) \quad \forall z \in A. \quad (2.23)$$

Alors la suite correspondante de ATPG vérifie

$$\limsup_{n \rightarrow \infty} |f(z) - (n)_f^G(z)|^{1/(n+1)} \leq \frac{\max_{t \in \Delta} \sigma(t)}{\min_{x \in \Gamma} \sigma(x)},$$

pour tous les contours  $\Gamma$  dans  $A$  tels que  $\forall n \in \mathbb{N} \quad \{z_{ni}\}_{i=0}^n \subset \text{Int}(\Gamma)$  et  $\Gamma \cap \Delta = \emptyset$ .

THÉORÈME 2.26. [A8] Soit  $f$  de la forme

$$f(z) = \int_{\Delta} G(t, z) d\alpha(t), \quad (2.24)$$

avec  $G(\cdot, z)$  analytique dans  $A = \mathbb{R}^+ \times ]-a, a[$  pour  $z \in C$  compact de  $\mathbb{C}$ . Soit  $\{\pi_n(z)\}_{n \in \mathbb{N}}$  la suite de polynômes orthonormaux sur  $\Delta$  par rapport à la mesure  $d\alpha(t)$ . On construit la suite de ATPG correspondants aux suites de points d'interpolation suivantes :  $\forall n \in \mathbb{N}$ ,  $\{z_{ni}\}_{i=0}^n$  sont les racines de  $\pi_{n+1}(z)$ . Alors, si on pose  $\Omega = \mathbb{C} \setminus \Delta$  et  $g_\Omega(z, \infty)$  la fonction de Green avec pôle à l'infini de  $\Omega$ , l'erreur des approximants a le comportement asymptotique suivant :

$$\forall z \in \mathbb{C} \quad \limsup_{n \rightarrow \infty} |f(z) - (n)_f^G(z)|^{1/2n} \leq \frac{1}{\min_{x \in \Gamma} e^{g_\Omega(x, \infty)}}$$

pour tout  $\Gamma$  tel que

$$\Delta \subset \text{Int}(\Gamma), \quad \Gamma \subset A, \quad \Gamma \cap \Delta = \emptyset.$$

Pour des fonctions données par leur développement en série dans une famille  $\{g_i(z)\}_i$ , les résultats de convergence obtenus le long de cette partie pour certaines suites d'approximants de type Padé généralisés montrent les bonnes propriétés d'approximation de ces approximants. Néanmoins, quelques problèmes importants restent ouverts, notamment celui du choix des points d'interpolation (i.e., les fonctionnelles  $L_k$ ) de façon à obtenir une meilleure vitesse de convergence et l'étude d'autres suites d'approximants de la table, ... Ces sujets font toujours partie de mes recherches actuelles.

Nous avons considéré, dans ce chapitre, une généralisation des approximants de type Padé aux séries de fonctions, et en particulier aux séries orthogonales. Nous proposerons, au chapitre suivant, une autre généralisation - cette fois ci des approximants de Padé (et non plus type Padé) aux séries orthogonales réelles, vectorielles et à plusieurs variables.



## Chapitre 3

# Approximation rationnelle de séries orthogonales

Ce chapitre est basé sur les publications [A11], [A12] et [A13]

### 3.1 Introduction

Tout le long de ce chapitre, nous allons nous intéresser à l'approximation de fonctions définies par des développements en série orthogonales, au moyen d'approximants rationnels qui généralisent les approximants de Padé (approximants rationnels pour des séries entières).

Soit  $f$  une fonction donnée par une série orthogonale dans un domaine  $\mathcal{D}$

$$f(z) = \sum_{k=0}^{\infty} f_k P_k(z) \quad z \in \mathcal{D}, \quad (3.1)$$

où  $\{P_k\}$  est une famille de polynômes orthogonaux dans l'intervalle  $[a, b]$  par rapport à une fonction poids  $w(x) \geq 0$  et

$$f_k = c_k(f) = \frac{1}{\|P_k\|_2^2} \int_a^b f(x) P_k(x) w(x) dx,$$

c'est-à-dire, (3.1) est le développement de Fourier de  $f$  par rapport au système  $\{P_k\}$ .

Des généralisations du concept d'approximant de Padé à une série orthogonale ont été proposées dans la littérature et conduisent à deux classes essentiellement différentes d'approximants rationnels :

**1) L'approximant de Padé non linéaire**  $\Phi_{L,M}$  d'ordre  $(L, M)$  de la série (3.1) est une fonction rationnelle (numérateur de degré  $L$ , dénominateur de degré  $M$ ) dont le développement de Fourier par rapport au système  $\{P_k\}$ ,

$$\Phi_{L,M}(z) = \phi_0 P_0(z) + \phi_1 P_1(z) + \cdots + \phi_n P_n(z) + \cdots$$

vérifie  $\phi_i = f_i \quad i = 0, \dots, L + M$ . L'approximant est donc déterminé par la solution d'un système d'équations non linéaires

$$c_k(\Phi_{L,M}) = c_k(f) \quad k = 0, 1, \dots, L + M. \quad (3.2)$$

Ce système n'a pas toujours de solution. Dans le cas d'un développement en série de Legendre, une méthode pour le résoudre est donnée dans [58] et pour une série de Tchebychev, un algorithme pour le calcul des coefficients de l'approximant est développé dans [94].

**2) L'approximant de Frobenius-Padé**  $[L/M]_f^P = \frac{N}{D}$  (ou approximant de Padé linéaire) d'ordre  $(L, M)$  de la série (3.1) est une fonction rationnelle de type  $(L, M)$ , où les polynômes du numérateur et dénominateur vérifient

$$c_k(Df - N) = 0 \quad k = 0, \dots, L + M.$$

Ceci conduit à un système d'équations linéaires dont la solution correspond aux coefficients des polynômes. Le cas d'un développement en série de Tchebychev a été étudié longuement dans [4] et [36].

Les approximants non linéaires ont des avantages par rapport aux approximants de Frobenius-Padé en ce qui concerne l'ordre d'approximation : en effet, on vérifie facilement que pour obtenir un ordre d'approximation (coïncidence des coefficients dans le développement en série des  $\{P_k\}$ ) de  $L + M$  on doit connaître les  $L + 2M + 1$  premiers coefficients de la série pour construire  $[L/M]_f^P$  et seulement  $L + M + 1$  coefficients pour construire  $\Phi_{L,M}$ . Mais ils ont aussi des grands inconvénients : ils sont bien plus difficiles à calculer et les propriétés d'existence et d'unicité sont difficiles à obtenir. Différents résultats sur la convergence de ces approximants ont été obtenus : en [60] les auteurs ont comparé des suites diagonales  $(\Phi_{n+j,n})_n$  et  $([n+j/n]_j^P)_n$  et obtenu des bornes sur leur vitesse de convergence pour des fonctions de type Markov. Le problème de la convergence des colonnes de la table de Padé généralisée a aussi été étudiée dans [108], [109] et [110], où un analogue du théorème de Montessus de Ballore a été montré pour les approximants non linéaires.

Dans ce chapitre nous allons étudier les *approximants de Frobenius-Padé*. Nous commencerons par nous intéresser au calcul de ces approximants pour une fonction  $f$  de la forme (3.1). Nous développerons deux algorithmes récursifs pour le calcul de suites de ces approximants disposées dans une table analogue à la table de Padé. À partir du comportement asymptotique de la suite des coefficients  $(f_k)_k$  de  $f$ , nous estimerons la vitesse de convergence des colonnes de la table de Frobenius-Padé. (Une étude plus complète de ces approximants a été faite par J. Matos dans sa thèse que j'ai co-dirigée [88]).

Les résultats numériques obtenus en programmant ces algorithmes ont été très encourageants, et donc nous avons généralisé la notion d'approximant de Frobenius-Padé au cas des fonctions vectorielles : dans la section **3.3** nous définirons les *approximants de Frobenius-Padé simultanés* et nous proposerons différents algorithmes de calcul de différentes suites d'approximants dans la table.

Finalement nous considérerons des séries orthogonales à deux variables (section **3.4**). Nous généraliserons les différentes définitions des approximants de Padé à plusieurs variables pour des

séries entières au cas de séries orthogonales. Nous nous intéresserons à deux types de problèmes : calcul des valeurs d'une suite d'approximants en un point donné par un algorithme récursif et calcul des coefficients d'une suite d'approximants. Pour ce dernier problème, qui sera équivalent à la résolution d'un système linéaire, on obtiendra pour certains cas particuliers une structure de la matrice du système ("displacement rank structure") qui permettra le développement d'algorithmes rapides de calcul.

## 3.2 Approximants de Frobenius-Padé : calcul et propriétés d'accélération

Soit  $f$  donnée par son développement (3.1) dans un domaine  $\mathcal{D}$ . Nous voulons calculer une approximation rationnelle de  $f$  de la forme

$$[p/q]_f^P(z) = \frac{N^{[p/q]}(z)}{D^{[p/q]}(z)}, \text{ avec}$$

$$\begin{cases} N^{[p/q]}(z) = \sum_{i=0}^p a_i P_i(z) \\ D^{[p/q]}(z) = \sum_{j=0}^q b_j P_j(z) \end{cases},$$

où  $N^{[p/q]}$  et  $D^{[p/q]}$  vérifient la propriété suivante :

$$D^{[p/q]}(z)f(z) - N^{[p/q]}(z) = \sum_{k=p+q+1}^{\infty} e_k P_k(z) \text{ pour } z \in \mathcal{D}. \quad (3.3)$$

$[p/q]_f^P$  est appelé *l'approximant de Frobenius-Padé* d'ordre  $(p, q)$  de la série orthogonale  $f$ . On pose

$$P_j(z)f(z) = \sum_{k=0}^{\infty} h_{kj} P_k(z) \text{ avec } h_{kj} = \frac{1}{\|P_k\|_2^2} \int_a^b f(x)P_j(x)P_k(x)w(x)dx. \quad (3.4)$$

En développant (3.3) nous obtenons

$$\sum_{k=0}^{\infty} \left( \sum_{j=0}^q b_j h_{kj} \right) P_k(z) - \sum_{k=0}^p a_k P_k(z) = \mathcal{O}(P_{p+q+1}(z)),$$

(la notation  $\mathcal{O}(P_{p+q+1}(z))$  signifie que le premier terme non nul dans le développement orthogonal de la fonction a un indice supérieur ou égal à  $p+q+1$ ). Les coefficients de  $D^{[p/q]}(z)$  et de  $N^{[p/q]}(z)$  sont donc solution des deux systèmes linéaires suivants :

$$\begin{cases} \sum_{j=0}^q b_j h_{kj} = 0 & k = p+1, \dots, p+q; \\ \sum_{j=0}^q b_j h_{kj} = a_k & k = 0, \dots, p. \end{cases} \quad (3.5)$$

On remarque que, une fois calculés les  $(b_j)$ , les  $(a_i)$  s'obtiennent immédiatement. Les  $(b_j)$  sont solution d'un système homogène de  $q$  équations et  $q+1$  inconnues. Il y a donc toujours une solution non triviale et nous serons intéressé par la solution correspondant à  $b_q \neq 0$ . Nous pouvons construire ces approximants pour différentes valeurs de  $p \geq 0$  et  $q \geq 0$  et les placer dans une table bidimensionnelle analogue à la table de Padé - *la table de Frobenius-Padé*. Pour simplifier les notations on introduit les déterminants suivants.

DEFINITION 3.1.

$$H_{l+1,p}^n = \begin{vmatrix} h_{n,0} & h_{n,1} & \cdots & h_{n,l-1} & h_{n,p} \\ h_{n+1,0} & h_{n+1,1} & \cdots & h_{n+1,l-1} & h_{n+1,p} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ h_{n+l,0} & h_{n+l,1} & \cdots & h_{n+l,l-1} & h_{n+l,p} \end{vmatrix} \quad \text{avec } p \geq l. \quad (3.6)$$

On vérifie immédiatement que, si  $H_{q+1,q}^{p+1} \neq 0$ , alors il y a une solution du système avec  $b_q \neq 0$ , et l'on peut donner des expressions déterminantales pour le numérateur et le dénominateur de l'approximant :

$$D^{[p/q]}(z) = \begin{vmatrix} h_{p+1,0} & h_{p+1,1} & \cdots & h_{p+1,q} \\ \cdots & \cdots & \cdots & \cdots \\ h_{p+q,0} & h_{p+q,1} & \cdots & h_{p+q,q} \\ P_0(z) & P_1(z) & \cdots & P_q(z) \end{vmatrix} \quad \text{et}$$

$$N^{[p/q]}(z) = \begin{vmatrix} h_{p+1,0} & h_{p+1,1} & \cdots & h_{p+1,q} \\ \cdots & \cdots & \cdots & \cdots \\ h_{p+q,0} & h_{p+q,1} & \cdots & h_{p+q,q} \\ \sum_{i=0}^p h_{i,0} P_i(z) & \sum_{i=0}^p h_{i,1} P_i(z) & \cdots & \sum_{i=0}^p h_{i,q} P_i(z) \end{vmatrix}.$$

Si  $H_{q+1,q}^{p+1} \neq 0$  pour tout  $p, q \geq 0$  la table est dite *normale* et on montre facilement que

- le polynôme du dénominateur a un degré exact  $q$  et le numérateur un degré exact  $p$ , avec :  $b_q = H_{q+1,q}^{p+1} a_q = \sum_{j=0}^q b_j h_{p,j} = (-1)^q H_{q+1,q}^p$  ;
- le coefficient du premier terme dans le développement de l'erreur est donné par  $e_{p+q+1} = \sum_{j=0}^q b_j h_{p+q+1,j} = H_{q+1,q}^{p+1} \neq 0$ .

Le cas d'une table non normale entraîne, comme dans le cas de la table de Padé, l'existence de blocks d'approximants identiques (la condition d'ordre étant vérifiée pour des polynômes de degré  $p_1 < p$ ,  $q_1 < q$ ). Nous nous placerons, pour la suite, dans le cas d'une table normale, et nous développerons des algorithmes récursifs de calcul de suites d'approximants.

### 3.2.1 Calcul récursif de suites d'approximants

Commençons tout d'abord par le calcul des coefficients d'un approximant. Pour cela, nous avons besoin des quantités  $h_{k,j}$  définies par (3.4) et qui constituent la matrice des coefficients des systèmes à résoudre pour calculer les coefficients des polynômes du numérateur et dénominateur. On commence par développer un algorithme récursif pour les calculer, en nous inspirant des idées de [63].

Rappelons qu'une famille de polynômes orthogonaux vérifie une relation de récurrence à trois termes de la forme

$$P_{k+1}(x) = (\alpha_k x + \beta_k) P_k(x) - \gamma_k P_{k-1}(x), \quad k = 0, 1, \dots,$$

qui sera la base de tous les calculs récursifs qui suivent. En utilisant cette relation nous pouvons obtenir

$$\begin{aligned} P_{k+1}(x) P_j(x) &= \frac{\alpha_k}{\alpha_j} P_k(x) P_{j+1}(x) - \frac{\alpha_k}{\alpha_j} \beta_j P_k(x) P_j(x) + \\ &+ \frac{\alpha_k}{\alpha_j} \gamma_j P_k(x) P_{j-1}(x) + \beta_k P_k(x) P_j(x) - \gamma_k P_{k-1}(x) P_j(x). \end{aligned}$$



En multipliant par  $f(x)w(x)$  et en intégrant on obtient

$$\frac{\mu_{k+1}}{\mu_k} h_{k+1,j} = \frac{\alpha_k}{\alpha_j} h_{k,j+1} + \left( \beta_k - \frac{\alpha_k}{\alpha_j} \beta_j \right) h_{k,j} + \frac{\alpha_k}{\alpha_j} \gamma_j h_{k,j-1} - \gamma_k \frac{\mu_{k-1}}{\mu_k} h_{k-1,j}, \quad j \geq 1, k \geq 0. \quad (3.7)$$

où  $\mu_k = \|P_k\|_2^2$  pour  $k \geq 0$ . Cette relation, accompagnée des initialisations

$$h_{j0} = f_j, \quad h_{j,-1} = 0 \quad j = 0, 1, \dots, \quad , h_{0,j} = \frac{\mu_j}{\mu_0} f_j, \quad j \geq 0,$$

permet le calcul récursif des quantités  $h_{kj}$ , en progressant de gauche à droite et de haut en bas comme l'indique le tableau ci-dessous

$$\begin{array}{cccccccc} h_{0,-1} & h_{00} & \cdots & \cdots & h_{0,j-1} & h_{0j} & h_{0,j+1} & \cdots \\ h_{1,-1} & h_{1,0} & \cdots & \cdots & & \cdots & & \cdots \\ \vdots & \vdots & \ddots & \ddots & \vdots & \vdots & \vdots & \vdots \\ \cdots & \cdots & \cdots & \cdots & h_{k-1,j-1} & \boxed{h_{k-1,j}} & \searrow & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \boxed{h_{k,j-1}} & h_{k,j} & \boxed{h_{k,j+1}} & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & h_{k+1,j-1} & \boxed{h_{k+1,j}} & \nearrow & \vdots & \vdots \end{array}$$

De plus, on a besoin de calculer que la partie triangulaire supérieure (ou inférieure) car  $h_{kj} = \frac{\mu_j}{\mu_k} h_{jk} (\forall j, k \geq 1)$ .

Une fois calculées les quantités  $h_{kj}, k = 0, \dots, p+q, j = 0, \dots, q$  par la relation de récurrence (on vérifie que pour cela on doit connaître les  $p+2q+1$  premiers coefficients de la série), si l'on veut calculer l'approximant de Frobenius-Padé  $[p/q]_f^P$ , les coefficients ( $b_j$ ) du dénominateur sont donnés par la solution du système

$$\begin{pmatrix} h_{p+1,0} & h_{p+1,1} & \cdots & \cdots & h_{p+1,q} \\ h_{p+2,0} & h_{p+2,1} & \cdots & \cdots & h_{p+2,q} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ h_{p+q,0} & h_{p+q,1} & \cdots & \cdots & h_{p+q,q} \end{pmatrix} \begin{pmatrix} b_0 \\ b_1 \\ \vdots \\ b_q \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix},$$

et les coefficients ( $a_i$ ) directement par

$$a_j = h_{j,0} b_0 + h_{j,1} b_1 + \cdots + h_{j,q} b_q, \quad j = 0, 1, \dots, p.$$

Si l'on veut calculer ces coefficients pour une suite d'approximants alors on peut tirer profit des expressions déterminantales des polynômes et utiliser des identités de déterminants pour développer des algorithmes de calcul récursif de suites d'approximants : calcul d'une colonne en fonction des précédentes (algorithmes de type Frobenius) et calcul d'une anti-diagonale (algorithmes de type Kronecker).

**(a) Algorithmes de type Frobenius**

En utilisant l'identité de Sylvester (voir, par exemple, [4]) nous avons obtenu une relation de récurrence pour le calcul des déterminants  $H_{q,l}^n$ .

THÉORÈME 3.2. [A11] *Supposons que  $\forall p, \forall q \geq 0$ ,  $H_{q,q-1}^{p+1} \neq 0$ . Alors*

$$H_{l+1,p}^n = \frac{H_{l,p}^{n+1} H_{l,l-1}^n - H_{l,l-1}^{n+1} H_{l,p}^n}{H_{l-1,l-2}^{n+1}} \quad l \geq 1, \quad p \geq l, \quad n \in \mathbb{N}, \quad (3.8)$$

avec les initialisations

$$H_{0,-1}^{p+1} = 1, \quad H_{1,l}^{p+1} = h_{p+1,l}, \quad l \geq 0, \quad \forall p \geq 0.$$

On introduit maintenant les quantités suivantes

DEFINITION 3.3.

$$D_l^{[p/q]}(z) = \begin{vmatrix} h_{p+1,0} & h_{p+1,1} & \cdots & h_{p+1,q-1} & h_{p+1,l} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ h_{p+q,0} & h_{p+q,1} & \cdots & h_{p+q,q-1} & h_{p+q,l} \\ P_0(z) & P_1(z) & \cdots & P_{q-1}(z) & P_l(z) \end{vmatrix}, \quad l \geq q,$$

$$N_l^{[p/q]}(z) = \begin{vmatrix} h_{p+1,0} & h_{p+1,1} & \cdots & h_{p+1,q-1} & h_{p+1,l} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ h_{p+q,0} & h_{p+q,1} & \cdots & h_{p+q,q-1} & h_{p+q,l} \\ \sum_{i=0}^p h_{i,0} P_i(z) & \sum_{i=0}^p h_{i,1} P_i(z) & \cdots & \sum_{i=0}^p h_{i,q-1} P_i(z) & \sum_{i=0}^p h_{i,l} P_i(z) \end{vmatrix}, \quad l \geq q.$$

On remarque que  $D_q^{[p/q]}(z) = D^{[p/q]}(z)$  et  $N_q^{[p/q]}(z) = N^{[p/q]}(z)$  pour tout  $p, q \geq 0$ . En appliquant à ces quantités l'identité de Sylvester on peut obtenir :

PROPOSITION 3.4. **Identités de type Frobenius [A11] :**

$$D_l^{[p/q]}(z) = \frac{D_l^{[p/q-1]}(z) H_{q,q-1}^{p+1} - D_{q-1}^{[p/q-1]}(z) H_{q,l}^{p+1}}{H_{q-1,q-2}^{p+1}} \quad p \geq 0, q \geq 1$$

$$N_l^{[p/q]}(z) = \frac{N_l^{[p/q-1]}(z) H_{q,q-1}^{p+1} - N_{q-1}^{[p/q-1]}(z) H_{q,l}^{p+1}}{H_{q-1,q-2}^{p+1}} \quad p \geq 0, q \geq 1$$

avec les initialisations

$$D_l^{[p/0]}(z) = P_l(z), \quad N_l^{[p/0]}(z) = \sum_{i=0}^p h_{i,l} P_i(z).$$

Basé sur les relations de récurrence obtenues ci-dessus et sur les identités de type Frobenius, un algorithme de calcul de la table de Frobenius-Padé a été proposé et programmé dans [A11]. À partir des coefficients  $f_j, j = 0, \dots, 2M$ , l'algorithme calcule, par colonnes, les approximants

dans la table ci-dessous :

$[0/0]^P$	$[0/1]^P$	$[0/2]^P$	$\dots$	$[0/M-1]^P$	$[0/M]^P$
$[1/0]^P$	$[1/1]^P$	$[1/2]^P$	$\dots$	$[1/M-1]^P$	
$[2/0]^P$	$[2/1]^P$	$[2/2]^P$	$\dots$	$[2/M-1]^P$	
$\vdots$	$\vdots$	$\vdots$	$\dots$		
$\vdots$	$\vdots$	$[2M-4/2]^P$			
$\vdots$	$[2M-3/1]^P$				
$\vdots$	$[2M-2/1]^P$				
$[2M-1/0]^P$					
$[2M/0]^P$					

### (b) Algorithmes de type Kronecker

Nous avons aussi obtenu un algorithme pour le calcul d'une suite d'approximants dans une anti-diagonale  $[N-m/m]_f^P$ ,  $m = 0, 1, \dots, N$  (avec  $N$  fixé), c'est-à-dire, une suite d'approximants ayant le même ordre d'approximation ( $\mathcal{O}(P_{N+2}(z))$ ) et un degré de dénominateur croissant. On pose

$$[N-m/m]_f^P(z) = p_m(z)/q_m(z) \quad m = 0, 1, \dots, N,$$

où les polynômes  $p_m$  et  $q_m$  vérifient :

$$\begin{cases} \deg(p_m) = N - m, & \deg(q_m) = m; \\ q_m(z)f(z) - p_m(z) = e_{m,N+1}P_{N+1}(z) + \mathcal{O}(P_{N+2}(z)). \end{cases} \quad (3.9)$$

Nous avons cherché une relation de récurrence à 4 termes de la forme

$$\begin{cases} p_{j+1}(z) &= p_{j-2}(z) + \alpha_j(z)p_{j-1}(z) + \beta_j(z)p_j(z) \\ q_{j+1}(z) &= q_{j-2}(z) + \alpha_j(z)q_{j-1}(z) + \beta_j(z)q_j(z) \end{cases}$$

où  $\alpha_j(z)$  et  $\beta_j(z)$  sont des polynômes de premier degré dont les coefficients sont calculés de sorte que (3.9) soient satisfaites. Cela aboutit à un système linéaire de dimension 4 dont la solution nous donne les  $\alpha_j(z)$ ,  $\beta_j(z)$ .

A titre d'exemple, pour le cas d'une série de Legendre, on obtient le résultat suivant

**PROPOSITION 3.5. [A11]** *Soit  $f$  une fonction donnée par son développement en série de Legendre  $f(z) = \sum_{i=0}^{\infty} f_i L_i(z)$  pour  $z \in \mathcal{D}$  un domaine du plan complexe. Soit  $N$  fixé. Supposons vérifiée la condition de normalité  $H_{m+1,m}^{N-m} \neq 0$  pour  $m = 0, 1, \dots, N$ . Alors l'anti-diagonale  $[N-m/m]_f^L$ ,  $m = 0, \dots, N$  de la table de Padé-Legendre peut être calculée récursivement par l'algorithme de type Kronecker suivant :*

$$\begin{cases} p_{j+1}(z) &= p_{j-2}(z) + \alpha_j(z)p_{j-1}(z) + \beta_j(z)p_j(z) \\ q_{j+1}(z) &= q_{j-2}(z) + \alpha_j(z)q_{j-1}(z) + \beta_j(z)q_j(z) \end{cases} \quad \text{for } j = 0, \dots, N-1,$$

avec  $[N-j/j]_f^L(z) = p_j(z)/q_j(z)$  et, pour  $j \geq 0$ ,  $\alpha_j(z), \beta_j(z)$  définis par  $\alpha_j(z) = a_{0,j} +$

$a_{1,j}z, \beta_j(z) = b_{0,j} - b_{1,j}z$  où

$$\left\{ \begin{array}{l} a_{1,j} = -\frac{2N-2j+3}{N-j+2} \frac{c_{j-2,N-j+2}}{c_{j-1,N-j+1}} \\ b_{1,j} = -\frac{e_{j-1,N+1}}{e_{j,N+1}} a_{1,j}, \quad b_{1,0} = \frac{a_{1,0}}{c_{0,n+1}} \\ a_{0,j} = -\frac{1}{c_{j-1,N-j+1}} \left( \frac{N-j+1}{2N-2j+1} c_{j,N-j} b_{1,j} + \frac{N-j+1}{2N-2j+1} c_{j-1,N-j} a_{1,j} + c_{j-2,N-j+1} \right) \\ b_{0,j} = -\frac{1}{c_{j,N-j}} \left( c_{j-1,N-j} a_{0,j} + \frac{N-j}{2N-2j-1} (b_{1,j} c_{j,N-j-1} + a_{1,j} c_{j-1,N-j-1}) + \frac{N-j+1}{2N-2j+3} c_{j-1,N-j+1} a_{1,j} + c_{j-2,N-j} \right) \\ e_{j+1,i} = e_{j-2,i} + a_{0,j} e_{j-1,i} + b_{0,j} e_{j,i} + \\ + a_{1,j} \left[ \frac{1}{2i-1} e_{j-1,i-1} + \frac{i-1}{2i+3} e_{j-1,i+1} \right] + b_{1,j} \left[ \frac{1}{2i-1} e_{j,i-1} + \frac{i-1}{2i+3} e_{j,i+1} \right], \quad i \geq N+1 \end{array} \right.$$

avec les initialisations

$$\left\{ \begin{array}{l} p_{-2}(z) = L_{N+2}(z) \quad p_{-1}(z) = L_{N+1}(z) \quad p_0(z) = \sum_{i=0}^N f_i L_i(z) \\ q_{-2}(z) = 0 \quad q_{-1}(z) = 0 \quad q_0(z) = 1. \end{array} \right.$$

**Remarque :** on peut montrer que la condition de normalité implique que les calculs de cet algorithme peuvent être menés jusqu'au bout sans "breakdown".

Des relations de récurrence à 4 termes dans la table de Frobenius-Padé ont été trouvées et étudiées dans [88], où différents parcours dans la table pour calculer des suites d'approximants ont été proposés, avec des exemples numériques.

### 3.2.2 Convergence de suites de colonnes dans la table de Padé-Legendre

Nous nous intéressons maintenant à la convergence de suites d'approximants de Frobenius-Padé de la forme  $(([n/q]_f^P)_n)$  (suites de colonnes dans la table décrite ci-dessus). Nous allons considérer des séries de Legendre, mais des résultats similaires peuvent être obtenus pour d'autres séries orthogonales en utilisant le même type de techniques.

Étant donnée une série de Legendre  $f(z) = \sum_{k=0}^{\infty} f_k L_k(z)$ , son domaine de convergence  $\mathcal{D}$  est connu dans les cas suivants [43] :

- si  $\limsup_{k \rightarrow \infty} |f_k|^{1/k} = 1/r$  avec  $r > 1$  alors la série converge vers  $f$  à l'intérieur de l'ellipse  $\mathcal{E}_r$  (foyers  $\pm 1$ , demi-axes  $a = \frac{1}{2}(r + r^{-1})$ ,  $b = \frac{1}{2}(r - r^{-1})$ ) et donc  $\mathcal{D} = \mathcal{E}_r$  ;
- si  $f$  est analytique dans  $-1 \leq x \leq 1$  avec  $\limsup_{k \rightarrow \infty} |f_k|^{1/k} = 1$  alors la série converge vers  $f$  dans  $-1 < x < 1$  et alors  $\mathcal{D} = ]-1, 1[$ .

La vitesse de convergence des sommes partielles d'une série de Legendre peut être mesurée par l'ordre de convergence de la suite des coefficients  $(f_k)_k$  : si  $f(z) = \sum_{k=0}^{\infty} f_k L_k(z)$  pour  $z \in \mathcal{D}_f$ ,  $g(z) = \sum_{k=0}^{\infty} g_k L_k(z)$  pour  $z \in \mathcal{D}_g$  et  $\lim_{k \rightarrow \infty} \frac{g_k}{f_k} = 0$ , alors, pour  $z \in \mathcal{D}_f \cap \mathcal{D}_g$ , la suite des sommes partielles de  $g$  converge plus vite que celle de  $f$ .

Pour étudier la vitesse de convergence des colonnes de la table de Padé-Legendre d'une fonction donnée par son développement en série de Legendre, nous allons écrire les approximants comme les sommes partielles d'une certaine série de Legendre et obtenir le comportement

asymptotique de la suite des coefficients correspondante. Fixons  $q$  et considérons la suite :

$$[n/q]_f^L(z) = \left( \sum_{i=0}^n a_i^{(n)} L_i(z) \right) / \left( \sum_{j=0}^q b_j^{(n)} L_j(z) \right) = \sum_{i=0}^n c_i^{(n)}(z) L_i(z) \quad n \in \mathbb{N},$$

avec

$$c_i^{(n)}(z) = \frac{a_i^{(n)}}{\sum_{j=0}^q b_j^{(n)} L_j(z)}, \quad 0 \leq i \leq n.$$

Pour obtenir la vitesse de convergence de  $([n/q]_f^L(z))_n$ , il suffit de déterminer le comportement asymptotique de la suite  $(c_i^{(n)}(z))_n$  quand  $i \rightarrow \infty$ ,  $n \rightarrow \infty$  ( $i \leq n$ ). Pour cela, on commence par étudier le comportement asymptotique des suites  $(h_{k,j})_k$  quand  $k \rightarrow \infty$  ( $j$  fixé) à partir du comportement de  $(f_n)_n$ . Par récurrence on peut montrer que

PROPOSITION 3.6. [A11] *Supposons que la suite  $(f_n)_n$  vérifie*

$$h_{n,0} = f_n = \rho^n \left[ \alpha_0 + \frac{\beta_0}{n} + \mathcal{O}\left(\frac{1}{n^2}\right) \right], \quad \rho \neq 1, \alpha_0 \neq 0. \quad (3.10)$$

Alors

$$h_{n,j} = \rho^n \left[ \alpha_j + \frac{\beta_j}{n} + \mathcal{O}\left(\frac{1}{n^2}\right) \right] \quad (n \rightarrow \infty) \text{ avec}$$

$$\alpha_1 = \alpha_0(\rho + \frac{1}{\rho})/2, \text{ et } \alpha_j = \frac{2j-1}{j} \alpha_{j-1}(\rho + 1/\rho)/2 - \frac{j-1}{j} \alpha_{j-2}.$$

De plus, pour tout  $j$ ,  $\alpha_j \neq 0$ .

PROPOSITION 3.7. [A11] *Considérons une suite  $(f_n)_n$  ayant le comportement asymptotique suivant :*

$$h_{n,0} = f_n = \frac{\alpha}{n^p} \left( 1 + \frac{\beta}{n} + \frac{1}{n^2} \left( \sum_{i=0}^k \frac{c_{0i}}{n^i} + \mathcal{O}\left(\frac{1}{n^{k+1}}\right) \right) \right) \quad (n \rightarrow \infty). \quad (3.11)$$

Alors

$$\forall j \geq 1 \quad h_{n,j} = \frac{\alpha}{n^p} \left( 1 + \frac{\beta}{n} + \frac{1}{n^2} \left( \sum_{i=0}^k \frac{c_{ji}}{n^i} + \mathcal{O}\left(\frac{1}{n^{k+1}}\right) \right) \right) \quad (n \rightarrow \infty),$$

où  $c_{10} = c_{00} + \frac{1}{2}(p+1)^2$  et pour  $j \geq 2$ ,  $c_{j0} = \frac{2j-1}{j}(c_{j-1,0} + \frac{1}{2}(p+1)^2) - \frac{j-1}{j}c_{j-2,0}$ .

De ces résultats on peut obtenir le comportement asymptotique des coefficients du numérateur  $(a_i^{(n)})(n \rightarrow \infty, i \leq n)$  et des coefficients du dénominateur  $(b_i^{(n)}), i = 0, \dots, q$  pour pouvoir ensuite estimer la vitesse de convergence des approximants. On considère d'abord le cas  $q = 1, 2$  pour lesquels on peut obtenir la forme explicite de ces coefficients et considérons ensuite le cas général  $q > 2$ .

(i)  $\boxed{q = 1, 2}$

En utilisant la forme des  $(h_{n,0})_n$ , les relations de récurrence et un peu de calcul formel, nous obtenons, à partir de l'expression explicite des coefficients  $(a_i^{(n)})$  et  $(b_i^{(n)})$  leur comportement asymptotique, ce qui permet d'obtenir :

**THÉORÈME 3.8. [A11]** Soit  $f(z) = \sum_{i=0}^{\infty} f_i L_i(z)$  donnée par une série de Legendre dans un domaine  $\mathcal{D}$  de convergence de la série. On suppose que  $H_{2,1}^n \neq 0 \quad \forall n \in \mathbb{N}$  et on considère la première colonne de la table de Padé-Legendre,

$$[n/1]_f^L(z) = \sum_{i=0}^n c_i^{(n)}(z) L_i(z).$$

Alors

1. si la suite  $(f_i)_i$  vérifie (3.10), alors

$$c_i^{(n)}(z) \sim K(z) \frac{\rho^i}{i} (i \rightarrow \infty) \quad (i \leq n),$$

(avec  $K(z)$  indépendant de  $i$ ). Donc pour  $z \in \mathcal{D}$ , la suite  $([n/1]_f^L(z))_n$  converge vers  $f(z)$  comme la suite des sommes partielles d'une série de Legendre dont la suite des coefficients est d'ordre  $\mathcal{O}\left(\frac{\rho^n}{n}\right)$ ,

2. si la suite des coefficients  $(f_i)_i$  vérifie (3.11), alors le comportement asymptotique de  $c_i^{(n)}(z)$  est donné par

$$c_i^{(n)}(z) \sim K(z) \frac{1}{i^{p+2}} (i \rightarrow \infty), (i \leq n),$$

avec  $K(z)$  indépendant de  $i$ . Donc pour  $z \in \mathcal{D}$ , la suite  $([n/1]_f^L(z))_n$  converge vers  $f(z)$  comme la suite des sommes partielles d'une série de Legendre dont la suite des coefficients est d'ordre  $\mathcal{O}\left(\frac{1}{n^{p+2}}\right)$ .

**THÉORÈME 3.9. [A11]** Soit  $f$  donnée par une série de Legendre  $f(z) = \sum_{i=0}^{\infty} f_i L_i(z)$  dans son domaine  $\mathcal{D}$  de convergence. Supposons que la suite  $(f_i)_i$  vérifie (3.11). Alors la deuxième colonne de la table de Padé-Legendre,  $([n/2]_f^L(z))_n$  converge vers  $f$  dans  $\mathcal{D}$  comme les sommes partielles d'une série de Legendre dont la suite des coefficients a l'ordre  $\mathcal{O}\left(\frac{1}{n^{p+4}}\right)$ .

(ii)  $q > 2$

Le calcul des premiers termes du développement des coefficients de  $[n/q]_f^L$  ( $i < q$ ),  $b_i^{(n)}$ , à partir de leur expression explicite comme un quotient de deux déterminants devient très difficile quand  $q > 2$ . Nous procédons alors d'une façon différente. Supposons que l'on puisse calculer des quantités  $(b_{i,0}, i = 0, \dots, q-1)$  telles que

$$h_{n,0}b_{0,0} + h_{n,1}b_{1,0} + h_{n,q-1}b_{q-1,0} + h_{n,q} = \epsilon(n) \text{ avec } \epsilon(n) = \mathcal{O}\left(\frac{1}{n^{\alpha+q+1}}\right) (n \rightarrow \infty).$$

En utilisant le développement de  $(h_{ni})_n$ ,  $i \leq q$  et en regroupant les termes de mêmes puissances en  $(1/n)$ , nous pouvons montrer que ceci est équivalent à supposer que le système

$$\begin{cases} b_{10} + b_{20} + \dots + b_{q0} + 1 = 0 \\ b_{10}c_{1i} + b_{20}c_{2i} + \dots + b_{q0}c_{qi} + c_{0i} = 0, \quad i = 1, \dots, q-1 \end{cases}$$

a une solution. On pose  $b_i^{(n)} = b_{i0} + \epsilon_i(n)$  pour  $i = 0, \dots, q-1$ , et en regardant le système vérifié par les  $b_i^{(n)}$ , on obtient pour  $i = 0, \dots, q-1$ ,

$$\epsilon_i(n) = \frac{|h_{n+1,0} \ h_{n+1,1} \ \cdots \ h_{n+1,i-1} \ \epsilon(n+1) \ h_{n+1,i+1} \ \cdots \ h_{n+1,q-1}|}{|h_{n+1,0} \ h_{n+1,1} \ \cdots \ h_{n+1,i-1} \ h_{n+1,i} \ h_{n+1,i+1} \ \cdots \ h_{n+1,q-1}|} \sim \frac{K_i}{n^{q+1}} \quad (n \rightarrow \infty).$$

De même, pour les coefficients du numérateur. Ces résultats sont regroupés dans le théorème suivant :

**THÉORÈME 3.10.** [A11] *Soit  $f$  donnée par une série de Legendre  $f(z) = \sum_{k=0}^{\infty} f_k L_k(z)$  pour  $z \in \mathcal{D}$  domaine du plan complexe. Supposons que la suite des coefficients  $(f_k)_k$  vérifie (3.11). On construit la table de Padé-Legendre et l'on pose*

$$[n/q]_f^L(z) = \sum_{i=0}^n c_i^{(n)}(z) L_i(z).$$

Pour  $q > 2$  fixé la suite  $(c_i^{(n)}(z))_i$  vérifie

$$c_i^{(n)}(z) \sim \frac{K}{i^{p+q+2}} \quad (i \rightarrow \infty) \quad (i \leq n).$$

Pour  $z \in \mathcal{D}$ , chaque colonne de la table de Padé-Legendre converge vers  $f$  comme les sommes partielles d'une série de Legendre dont la suite des coefficients est d'ordre  $\mathcal{O}\left(\frac{1}{n^{p+q+2}}\right)$  ( $n \rightarrow \infty$ ).

Dans [A11] les algorithmes des sections précédentes ont été programmés et différents exemples numériques y sont proposés, illustrant ces théorèmes de convergence et les bonnes propriétés d'accélération des approximants de Padé-Legendre, même quand la fonction a une singularité dans l'intervalle.

Une des applications de ces approximants concerne la résolution par des méthodes spectrales des équations aux dérivées partielles, où l'on obtient comme approximation de la solution une somme partielle d'une série orthogonale, que l'on peut essayer d'améliorer en utilisant les approximants de Frobenius-Padé (voir par exemple [50, 68, 69]), surtout en présence de singularités.

### 3.3 Approximants de Frobenius-Padé simultanés

Les idées d'approximation rationnelle simultanée de fonctions données par leur développement en série entière ont été introduites et étudiées par De Bruin dans [45], des méthodes de calcul de ces approximants développées dans [44, 46], et dans [61] des analogues des identités de Frobenius ont été obtenues, permettant de développer des algorithmes récursifs de calcul de suites "antidiagonales" comme dans la table de Padé (algorithme de Kronecker). Ces idées ont été étudiées plus en détail et ensuite généralisées par l'introduction des polynômes orthogonaux vectoriels et les polynômes orthogonaux matriciels (voir par exemple [106, 114]), où l'ensemble des paramètres

libres est limité (*indices réguliers*), ce qui permet l'obtention d'une table bien structurée. En nous basant sur ces deux concepts d'approximation rationnelle simultanée et/ou vectorielle, nous allons maintenant étendre la notion d'approximation de Frobenius-Padé à la construction d'approximants rationnels pour des fonctions vectorielles données par leur développement en série orthogonale - *les approximants de Frobenius-Padé simultanés*.

Après avoir donné la définition de ces approximants pour des fonctions vectorielles de dimension  $d$ , nous nous restreindrons au cas  $d = 2$  pour développer des algorithmes récursifs de calcul de certaines suites dans la table : nous obtiendrons des relations de récurrence à trois termes entre les numérateurs (dénominateurs) de suites d'approximants, nous établirons des relations entre des approximants ayant la même précision (similaire à l'antidiagonale dans le cas de la table de Padé classique), ainsi que des algorithmes de calcul de suites diagonales. Pour obtenir une augmentation plus rapide de la précision et une table à deux paramètres (au lieu de 4) nous nous intéresserons à *l'approximation vectorielle à indices réguliers*, qui consiste à diminuer le nombre de degrés de liberté : on impose le même degré aux deux numérateurs et la précision est équidistribuée sur les deux composantes.

### 3.3.1 Définitions et notations

Les notations correspondent à une généralisation triviale de ce qui a été fait dans la section précédente, avec des indices supérieurs indiquant la composante de la fonction. On considère donc une série vectorielle

$$\mathbb{F}(z) = (f^1(z), f^2(z), \dots, f^d(z)) \in \mathbb{C}^d[[z]] .$$

Soit  $(P_k)_{k \geq 0}$  un système de polynômes orthogonaux par rapport à la fonction poids  $w(x)$  dans l'intervalle  $[a, b]$ , et l'on suppose que chaque composante  $f^j$  appartient à  $\mathcal{L}_2((a, b), w)$  et est donnée par une série orthogonale

$$f^j(z) = \sum_{k=0}^{\infty} f_k^j P_k(z), \quad f_k^j = \frac{1}{\|P_k\|_2^2} \int_a^b f^j(x) P_k(x) w(x) dx, \quad j = 1, \dots, d.$$

On va construire un approximant rationnel de  $\mathbb{F}(z)$  de la façon suivante :

- on définit deux multi-indices  $\mathbf{p} = (p_1, p_2, \dots, p_d)$  and  $\mathbf{q} = (q_1, q_2, \dots, q_d)$  ;
- les polynômes du numérateur sont notés par  $\text{Num}(z) = (N^1(z), \dots, N^d(z))$  et leurs degrés  $\deg(N^j(z)) \leq p_j, \quad j = 1, \dots, d$  ;
- le polynôme du dénominateur  $D(z)$  a un degré au plus  $q = q_1 + \dots + q_d$  ;
- ces polynômes vérifient certaines conditions d'ordre/précision, c'est-à-dire, l'erreur  $\mathbb{R}\text{em}(z) = (R^1(z), R^2(z), \dots, R^d(z))$  vérifie

$$R^j(z) = D(z)f^j(z) - N^j(z) = e_{p_j+q_j+1}^j P_{p_j+q_j+1}(z) + \dots, \quad j = 1, \dots, d, \quad (3.12)$$

(pour chaque composante de  $\mathbb{R}\text{em}(z)$ , les  $p_j + q_j$  premiers coefficients du développement par rapport au système orthogonal  $(P_k)_{k \geq 0}$  sont nuls).

On appelle *approximant de Frobenius-Padé simultané* la fonction rationnelle

$$[\mathbf{p}, \mathbf{q}] \text{ ou } [p_1, \dots, p_d; q_1, \dots, q_d] = \text{Num}(z)/D(z).$$



Si l'on pose

$$N^j(z) = \sum_{i=0}^{p_j} a_i^j P_i(z), \quad j = 1, \dots, d, \quad D(z) = \sum_{i=0}^q b_i P_i(z),$$

et si l'on définit les quantités  $(h_{ki}^j), j = 1, \dots, d$ , par

$$f^j(z)P_i(z) = \sum_{k=0}^{\infty} h_{ki}^j P_k(z), \quad \text{avec } h_{ki}^j = \frac{1}{\|P_k\|_2^2} \int_a^b f^j(x)P_i(x)P_k(x)w(x)dx,$$

alors les coefficients des numérateurs et du dénominateur sont solution d'un système (analogue au cas scalaire) qui a toujours une solution :

$$\sum_{l=0}^q b_l h_{kl}^j = a_k^j, \quad k = 0, \dots, p_j, \quad j = 1, \dots, d, \quad (3.13)$$

$$\sum_{l=0}^q b_l h_{kl}^j = 0 \quad k = p_j + 1, \dots, p_j + q_j, \quad j = 1, \dots, d. \quad (3.14)$$

Les quantités  $h_{k,i}^j$  peuvent être calculées récursivement (comme dans la section précédente) à partir des données  $f_i^j, i \geq 0, j = 1, \dots, d$ , en utilisant la relation de récurrence à trois termes de la famille  $(P_k)_{k \geq 0}$  de polynômes.

On voit facilement que, à un facteur multiplicatif près, le dénominateur  $D(z)$  admet la représentation déterminantale

$$D(z) = \begin{vmatrix} h_{p_1+1,0}^1 & h_{p_1+1,1}^1 & \cdots & h_{p_1+1,q}^1 \\ \cdots & \cdots & \cdots & \cdots \\ h_{n_1,0}^1 & h_{n_1,1}^1 & \cdots & h_{n_1,q}^1 \\ \vdots & \vdots & \vdots & \vdots \\ h_{p_d+1,0}^d & h_{p_d+1,1}^d & \cdots & h_{p_d+1,q}^d \\ \cdots & \cdots & \cdots & \cdots \\ h_{n_d,0}^d & h_{n_d,1}^d & \cdots & h_{n_d,q}^d \\ P_0(z) & P_1(z) & \cdots & P_q(z) \end{vmatrix},$$

et les numérateurs

$$N^j(z) = \begin{vmatrix} h_{p_1+1,0}^1 & h_{p_1+1,1}^1 & \cdots & h_{p_1+1,q}^1 \\ \cdots & \cdots & \cdots & \cdots \\ h_{n_1,0}^1 & h_{n_1,1}^1 & \cdots & h_{n_1,q}^1 \\ \vdots & \vdots & \vdots & \vdots \\ h_{p_d+1,0}^d & h_{p_d+1,1}^d & \cdots & h_{p_d+1,q}^d \\ \cdots & \cdots & \cdots & \cdots \\ h_{n_d,0}^d & h_{n_d,1}^d & \cdots & h_{n_d,q}^d \\ \sum_{i=0}^{p_j} h_{i,0}^j P_i(z) & \sum_{i=0}^{p_j} h_{i,1}^j P_i(z) & \cdots & \sum_{i=0}^{p_j} h_{i,q}^j P_i(z) \end{vmatrix} \quad j = 1, \dots, d.$$

Pour les multi-indices  $\mathbf{p}, \mathbf{q}$ , on définit les déterminants  $q \times q$  (avec  $q = \sum_1^d q_k$ ),

$$H_{\mathbf{p}, \mathbf{q}} = \begin{vmatrix} h_{p_1+1,0}^1 & h_{p_1+1,1}^1 & \cdots & h_{p_1+1,q-1}^1 \\ \cdots & \cdots & \cdots & \cdots \\ h_{n_1,0}^1 & h_{n_1,1}^1 & \cdots & h_{n_1,q-1}^1 \\ \vdots & \vdots & \vdots & \vdots \\ h_{p_d+1,0}^d & h_{p_d+1,1}^d & \cdots & h_{p_d+1,q-1}^d \\ \cdots & \cdots & \cdots & \cdots \\ h_{n_d,0}^d & h_{n_d,1}^d & \cdots & h_{n_d,q-1}^d \end{vmatrix}.$$

Si  $H_{\mathbf{p}, \mathbf{q}} \neq 0$  alors le système (3.14) est de rang maximum  $q$  et admet une famille à un paramètre de solutions qui peut être obtenue par la règle de Cramer. Ceci est équivalent au fait que l'approximant est uniquement déterminé, et aussi au fait que le dénominateur et les numérateurs sont de degrés exacts  $(p_1, \dots, p_d, q)$ . Pour la suite, on se place dans ce cas de régularité :  $H_{\mathbf{p}, \mathbf{q}} \neq 0$  pour tout  $\mathbf{p}, \mathbf{q}$ .

Une fois donnée cette définition générale, nous allons par la suite nous restreindre au cas  $d = 2$  et nous intéresser au calcul de suites d'approximants.

Il y a plusieurs façons de choisir des suites d'approximants de Frobenius-Padé simultanés car on a plusieurs paramètres : le degré du dénominateur  $q = q_1 + q_2$ , les degrés des numérateurs  $\mathbf{p} = (p_1, p_2)$ , l'ordre d'approximation  $(p_1 + q_1, p_2 + q_2)$ . Du point de vue de l'approximation de  $(f^1, f^2)$  il semble naturel et intéressant de considérer les suites suivantes :

- on fixe le degré du dénominateur  $q = 2m + k$  ( $k = 0, 1$ ) et on impose le même ordre d'approximation  $p_i + q_i = n$ ,  $i = 1, 2$  aux deux fonctions. Les polynômes des numérateurs auront les degrés  $\mathbf{p}_n = (p_1, p_2)$  avec  $p_j = n - m - 1$ , ( $q_j = m + 1$ ) pour  $j = 1, \dots, k$ ,  $p_j = n - m$ , ( $q_j = m$ ) pour  $j = k + 1, \dots, 2$ , et l'on considère des "suites verticales"  $([\mathbf{p}_n, \mathbf{q}])_{n \in \mathbb{N}}$  ( $q$  fixé) ;
- on impose le même degré aux polynômes du numérateur  $p_1 = p_2 = p$  et, si  $q = 2m + k$ , on définit le multi-indice régulier  $\mathbf{q} = (q_1, q_2)$  par

$$\begin{aligned} q_j &= m + 1, & n_j &= p + m + 1, & j &= 1, \dots, k, \\ q_j &= m & n_j &= p + m, & j &= k + 1, \dots, 2, \end{aligned}$$

On considère alors des suites "diagonales régulières"  $([\mathbf{p}, \mathbf{q}])_{q \in \mathbb{N}}$  ;

- on fixe l'ordre d'approximation  $\mathbf{n} = (n_1, n_2)$  à partir du nombre de coefficients disponibles du développement des  $(f^1, f^2)$  et l'on construit une suite d'approximants en augmentant le degré du dénominateur de 0 à  $n_1 + n_2$  tout en décroissant le degré des numérateurs (suites "antidiagonales").

Pour ces différents types de suites, nous allons développer des algorithmes récursifs.

### 3.3.2 Relations de récurrence à trois termes

A partir de la représentation déterminantale des numérateurs et du dénominateur des approximants de Frobenius-Padé simultanés et en utilisant l'*identité de Jacobi* appliquée à un déterminant  $D$ ,

$$D_{i_1, i_2; j_1, j_2} D = D_{i_1; j_1} D_{i_2; j_2} - D_{i_1; j_2} D_{i_2; j_1} \text{ avec } 1 \leq i_1, i_2, j_1, j_2 \leq n,$$

(où  $D_{m;l}$  représente le déterminant obtenu à partir de  $D$  en éliminant la ligne  $l$  et la colonne  $m$ ) nous pouvons obtenir différentes relations de récurrence à trois termes faisant intervenir deux approximants d'une colonne et un approximant de la colonne précédente. Plus précisément,

**PROPOSITION 3.11.** **[A12]** *Si on note  $T[p_1, p_2; q_1, q_2]$  les numérateurs ou le dénominateur de l'approximant  $F$ - $P$  simultané  $[p_1, p_2; q_1, q_2]$ , alors les relations de récurrence à 3 termes suivantes sont vérifiées :*

$$\begin{aligned} T[p_1, p_2; q_1, q_2] &= \mu_A^0 T[p_1, p_2; q_1 + 1, q_2 - 1] + \mu_A^1 T[p_1, p_2; q_1, q_2 - 1], & \text{(A)} \\ T[p_1, p_2; q_1, q_2] &= \mu_B^0 T[p_1, p_2; q_1 - 1, q_2 + 1] + \mu_B^1 T[p_1, p_2; q_1 - 1, q_2], & \text{(B)} \\ T[p_1 + 1, p_2; q_1, q_2] &= \mu_C^0 T[p_1, p_2; q_1, q_2] + \mu_C^1 T[p_1 + 1, p_2; q_1 - 1, q_2], & \text{(C)} \\ T[p_1, p_2 + 1; q_1, q_2] &= \mu_D^0 T[p_1, p_2; q_1, q_2] + \mu_D^1 T[p_1, p_2 + 1; q_1, q_2 - 1]. & \text{(D)} \end{aligned}$$

avec  $\mu_A^i, \mu_B^i, \mu_C^i, \mu_D^i$ ,  $i = 0, 1$  des constantes définies par des quotients de déterminants de Hankel généralisés.

Ces relations peuvent être combinées et utilisées de façon astucieuse pour permettre de calculer toute la table de Frobenius-Padé. En effet, si l'on définit la colonne  $q$  de la table l'ensemble des approximants de la forme  $[p_1, p_2; q_1, q_2]$  avec  $q_1 + q_2 = q$ , on peut montrer que

- à partir de la connaissance d'un seul élément de la colonne  $q$  et d'éléments adéquats de la colonne précédente on peut calculer toute la colonne en utilisant **(A)**, **(B)**, **(C)**, **(D)** ;
- pour obtenir un premier approximant dans la colonne  $q$  on peut
  1. utiliser l'un des algorithmes développés ci-dessous qui permettent d'augmenter le degré du dénominateur (suite diagonale ou antidiagonale) ;
  2. construire l'approximant  $[p_1, p_2; q, 0]$  (ou  $[p_1, p_2; 0, q]$ ) qui correspond à l'approximant de Frobenius-Padé  $[p_1, q] = N/D$  de la fonction scalaire  $f^1$  (resp.  $[p_2, q]$  de  $f^2$ ) et la somme partielle de  $f^2 D$  ( $f^1 D$ ) par un algorithme pour les fonctions scalaires (voir [86, 88]) ;
- les coefficients des relations de la proposition 3.11 étant des quotients de déterminants de Hankel généralisés, nous pouvons développer des formules récursives pour leur calcul. Ils peuvent aussi être calculés en imposant les conditions d'ordre et de degré du nouvel approximant

Plus de détails sur le calcul récursif des approximants sont donnés dans **[A12]**. On peut ainsi développer des algorithmes qui permettent le calcul d'une suite quelconque d'approximants dans la table : ils consistent en une utilisation successive de relations de récurrence à trois termes entre les dénominateurs (et numérateurs) d'approximants "contigus" (i.e. des approximants  $[p_1, p_2; ; q_1, q_2]$  et  $[p'_1, p'_2; ; q'_1, q'_2]$  tels que  $\sum |p'_i - p_i| + \sum |q'_i - q_i| = 1$ ). En particulier on obtient facilement un algorithme pour calculer une suite "verticale" ( $(\mathbf{p}_n, \mathbf{q})_{n \in \mathbb{N}}$ , i.e. avec un degré de dénominateur constant  $q_1 + q_2$ ).

### 3.3.3 Un algorithme de type Kronecker

Intéressons-nous maintenant à la construction de suites ayant un ordre de convergence fixé (dans l'approximation scalaire classique ceci correspond aux antidiagonales de la table). Nous fixons  $n = n_1 = n_2$  et voulons calculer la suite d'approximants

$$\text{AFP}_k(z) = \text{Num}_k(z)/D_k(z), \quad k = 1, 2, \dots, 2n,$$

vérifiant la condition d'ordre de convergence suivante :

$$R_k^j(z) = D_k(z)f^j(z) - N_k^j(z) = \mathcal{O}(P_{n+1}), \quad j = 1, 2; \quad k = 1, 2, \dots, 2n,$$

avec

$$\begin{cases} \deg(N_k^1(z)) &= n - [(k+1)/2], \\ \deg(N_k^2(z)) &= n - [k/2] \quad k = 0, \dots, 2n, \\ \deg(D_k(z)) &= k. \end{cases}$$

Ceci signifie que la suite des degrés des numérateurs et dénominateurs est

$$(p_1, p_2, q) = (n, n, 0), (n-1, n, 1), (n-1, n-1, 2), (n-2, n-1, 3), \dots, (0, 1, 2n-1), (0, 0, 2n).$$

Nous cherchons une relation de récurrence vérifiée simultanément par les numérateurs, dénominateurs et restes, de la forme

$$T_{k+1}(z) = \sum_{i=1}^l c_i(z)T_{k+1-i}(z), \quad \text{avec } c_i(z) = \alpha_i + \beta_i z.$$

Les termes linéaires  $c_i(z)$  sont calculés à partir des conditions d'ordre et des conditions sur les degrés des numérateurs. Nous obtenons alors le résultat suivant :

**THÉORÈME 3.12.** [A12] *Soit  $\mathbb{F}(z) = (f^1(z), f^2(z))$  une fonction donnée par son développement en série orthogonale ; considérons la suite d'approximants de Frobenius-Padé simultanés  $(AFP_k)_{k \geq 0}$ . On suppose cette suite normale. Si*

$$\det \begin{pmatrix} e_{k-1, n+1}^1 & e_{k-2, n+1}^1 \\ e_{k-1, n+1}^2 & e_{k-2, n+1}^2 \end{pmatrix} \neq 0 \quad \text{pour tout } k,$$

(où les quantités  $e_{m,l}^j$  sont définies par  $R_k^j(z) = D_k(z)f^j(z) - N_k^j(z) = \sum_{i=n+1}^{\infty} e_{k,i}^j P_i(z)$ ,  $j = 1, 2$ ), alors les numérateurs et dénominateurs de ces approximants peuvent être calculés par une relation de récurrence à 6 termes de la forme :

$$T_{k+1}(z) = (\alpha_1^k + \beta_1^k z)T_k(z) + (\alpha_2^k + \beta_2^k z)T_{k-1}(z) + (\alpha_3^k + \beta_3^k z)T_{k-2}(z) + \alpha_4^k T_{k-3}(z) + \alpha_5^k T_{k-4}(z)$$

où  $\beta_1^k := 1$  et les autres coefficients sont solution d'un système inversible connu. Si  $S_k^j(z)$  sont les sommes partielles du développement en série, les initialisations nécessaires sont

$$\begin{cases} D_{-j}(z) = 1 & j = 1, \dots, 4 \\ N_{-j}^1 = \begin{cases} S_n^1(z) & \text{if } j = 1 \\ S_{n+1}^1(z) & \text{if } j = 2, 3 \\ S_{n+2}^1(z) & \text{if } j = 4 \end{cases} \\ N_{-j}^2 = \begin{cases} S_{n+1}^2(z) & \text{if } j = 1, 2 \\ S_{n+2}^2(z) & \text{if } j = 3, 4. \end{cases} \end{cases}$$

### 3.3.4 Calcul de suites diagonales et diagonales régulières

Du point de vue de l'approximation, une autre classe intéressante de suites d'approximants est celle des suites diagonales : l'idée est d'augmenter la précision en augmentant simultanément

les degrés des numérateurs et dénominateurs. Fixons  $p_1$  et  $p_2$  et considérons la suite d'approximants

$$[p_1, p_2; 0, 0], [p_1, p_2; 1, 0], [p_1 + 1, p_2; 1, 0], [p_1 + 1, p_2; 1, 1], [p_1 + 1, p_2 + 1; 1, 1], \dots,$$

c'est-à-dire, la suite  $(AFP_m)_{m \geq 0}$  définie par

$$(AFP_{4k+1}, AFP_{4k+2}, AFP_{4k+3}, AFP_{4k+4}) = \quad (3.15)$$

$$= ([p_1 + k, p_2 + k; k + 1, k], [p_1 + k + 1, p_2 + k; k + 1, k], \\ [p_1 + k + 1, p_2 + k; k + 1, k + 1], [p_1 + (k + 1), p_2 + (k + 1); k + 1, k + 1]). \quad (3.16)$$

On remarque que de l'étape  $k$  à l'étape  $k + 1$  nous devons calculer 4 termes, et que l'on augmente le degré du dénominateur de 2, le degré des numérateurs de 1 et la précision d'un ordre de 2 pour chaque fonction. Nous pouvons montrer le résultat suivant :

**THÉOREME 3.13.** [A12] *La suite d'approximants de Frobenius-Padé simultanés définie par (3.15) peut être calculée par une relation de récurrence à 6 termes de la forme*

$$T_{m+1}(z) = \alpha_1^m T_m(z) + (\alpha_2^m + \beta_2^m z) T_{m-1}(z) + (\alpha_3^m + \beta_3^m z) T_{m-2}(z), \\ + (\alpha_4^m + \beta_4^m z) S_{m-3}(z) + \alpha_5^m S_{m-4}(z)$$

où  $\alpha_j^m, \beta_j^m$  sont des constantes solution d'un système triangulaire, et  $T_i(z)$  représente soit les numérateurs soit le dénominateur des approximants. Cette relation est minimale par rapport au nombre de termes.

La démonstration est donnée dans [A12]. L'idée est la suivante : on essaie d'obtenir le nouveau terme  $T_{m+1}(z)$  de la suite en commençant par le dernier terme calculé  $T_m(z)$  multiplié par un facteur linéaire  $(\alpha_1^m + \beta_1^m z)$  et on additionne successivement les termes précédents de la suite multipliés par des facteurs linéaires jusqu'à ce que le nombre d'inconnues soit supérieur ou égal au nombre de conditions d'ordre et de degré des polynômes. On écrit alors ce système de conditions et l'on montre qu'il a une solution.

Considérons une autre façon de définir des suites diagonales. En suivant les idées d'"approximation régulière" et d'indices réguliers développées dans [106, 114] pour les approximants de Padé et de Padé-Hermite, le but est d'améliorer la précision rapidement en considérant une suite d'approximants dont les numérateurs ont le même degré et pour lesquels l'approximation est régulièrement distribuée. Plus précisément,

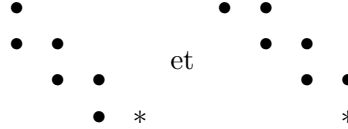
$$p_1 = p_2, \quad \begin{cases} q_1 = [q/2] + \varepsilon_q, & \varepsilon_q = 1 \text{ si } q \text{ impair, } 0 \text{ si } q \text{ pair.} \\ q_2 = [q/2] \end{cases}$$

Pour définir ces approximants on a besoin seulement de 2 indices et donc ils peuvent être placés dans un tableau bi-dimensionnel similaire à la table de Padé, les colonnes correspondants à  $q$  constant et les lignes à  $p$  constant. On a les notations suivantes :

$$\begin{cases} [p, q] = \text{Num}_{p,q}/D_{p,q}, \\ \text{Num}_{p,q} = (N_{p,q}^1, N_{p,q}^2), \quad \text{deg}(N_{p,q}^i) = p, \quad i = 1, 2, \quad \text{deg}(D_{p,q}) = q, \\ \text{Rem}_{p,q} = \mathbb{F}D_{p,q} - \text{Num}_{p,q} = (O(P_{p+[q/2]+\varepsilon_q+1}), O(P_{p+[q/2]+1})). \end{cases}$$

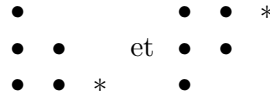
Dans [A12] nous avons obtenu des relations de récurrence reliant des approximants dans un escalier descendant et dans une diagonale. Plus précisément, en représentant par “•” les approximants connus et par “\*” l’approximant à calculer,

- nous avons obtenu des relations de récurrence reliant les numérateurs  $N_{p,q}^1, N_{p,q}^2$ , les dénominateurs  $D_{p,q}$  et les restes  $R_{p,q}^1$  ou  $R_{p,q}^2$  situés comme suit dans la table :



Les coefficients des relations sont des polynômes de degré 0, 1 et 2 (pour un seul coefficient).

- nous avons obtenu une relation diagonale à 8 termes qui permet le calcul de la suite  $([p+k, q+k])_k$  ;
- pour pouvoir initialiser les relations précédentes on a obtenu des relations qui permettent de calculer un approximant  $[p, q]$  à partir d’approximants des colonnes  $q-1$  et  $q-2$ , reliant les approximants



Une remarque du point de vue calcul est importante : les coefficients des relations de récurrence s’obtiennent en résolvant des systèmes de dimension 7 – 11 de matrice “presque” Hessenberg. Cette matrice est obtenue à partir des conditions de précision-ordre, ce qui fait que, quand on calcule une suite d’approximants, d’une étape à l’autre la matrice est la même à 1 ou 2 lignes près (selon les cas). Le calcul des coefficients est ainsi très simplifié car l’on peut profiter des calculs de l’étape précédente.

Ceci montre l’intérêt des algorithmes que l’on pourra déduire de ces relations de récurrence pour construire des suites d’approximants ayant un grand ordre de précision, sans passer par le calcul d’approximants moins intéressants.

Il serait maintenant intéressant de programmer ces relations et d’étudier les propriétés de convergence et de stabilité numérique de ces approximants. Ce travail est en cours.

## 3.4 Généralisations de l’approximation de Frobenius-Padé à plusieurs variables

### 3.4.1 Définitions

Différentes généralisations de l’approximation de Padé (pour des séries entières) au cas de plusieurs variables ont été étudiées par A. Cuyt [38, 39, 40, 41, 42], P. Guillaume [52, 53], et d’autres auteurs (voir, par exemple, [65], [76], [35], ...); propriétés des différents approximants, propriétés de convergence et algorithmes de calcul y ont été développées. En nous inspirant de ces différentes approches, nous avons généralisé les idées de l’approximation de Frobenius-Padé

au cas de deux variables, c'est-à-dire, étant donnée une fonction  $f$ ,

$$f(x, y) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} c_{ij} P_i(x) P_j(y),$$

où

$$\left\{ \begin{array}{l} \{P_i\} \text{ est un système de polynômes orthogonaux dans } [a, b] \text{ par rapport à} \\ \text{la fonction poids } w \\ \gamma_{ij} = \|P_i\| \|P_j\|, \quad \|P\|^2 = \int_a^b P(x)^2 w(x) dx \\ c_{ij} = \frac{1}{\gamma_{ij}} \int_a^b \int_a^b f(x, y) P_i(x) P_j(y) w(x) w(y) dx dy \end{array} \right. ,$$

construire des approximants rationnels de  $f$ ,  $P(x, y)/Q(x, y)$ , où  $P(x, y)$  et  $Q(x, y)$  sont des polynômes

$$\left\{ \begin{array}{l} P(x, y) = \sum_{(i,j) \in N} a_{ij} P_i(x) P_j(y) \\ Q(x, y) = \sum_{(i,j) \in D} b_{ij} P_i(x) P_j(y) \end{array} \right.$$

vérifiant

$$f(x, y)Q(x, y) - P(x, y) = \sum_{(i,j) \in (\mathbb{N}^2 \setminus E)} d_{ij} P_i(x) P_j(y) \quad (3.17)$$

où

- $D \subset \mathbb{N}^2$  a  $m$  éléments  $(k_1, l_1), \dots, (k_m, l_m)$ ,
- $E$  est l'ensemble des indices des termes qui s'annulent dans le développement de l'erreur,  $N \subset E$  avec  $n$  éléments  $(i_1, j_1), (i_2, j_2), \dots, (i_n, j_n)$ ,  $\text{card}(N) = n$ ,
- $H = E \setminus N$  a  $m - 1$  éléments  $(i_{n+1}, j_{n+1}), \dots, (i_{n+m-1}, j_{n+m-1})$ .

On définit alors l'approximant de Frobenius-Padé de  $f$  par

$$R(x, y) = \frac{P(x, y)}{Q(x, y)}.$$

Si l'on pose, pour  $(k, l) \in \mathbb{N}^2$ ,

$$P_k(x) P_l(y) f(x, y) = \sum_{(i,j) \in \mathbb{N}^2} h_{ij}^{kl} P_i(x) P_j(y),$$

on obtient

$$(fQ - P)(x, y) = \sum_{(i,j) \in \mathbb{N}^2} \left( \sum_{(k,l) \in D} b_{kl} h_{ij}^{kl} \right) P_i(x) P_j(y) - \sum_{(i,j) \in N} a_{ij} P_i(x) P_j(y).$$

La condition (3.17) s'écrit alors

$$\sum_{(k,l) \in D} b_{kl} h_{ij}^{kl} = 0, \quad (i, j) \in E \setminus N \quad (3.18)$$

$$\sum_{(k,l) \in D} b_{kl} h_{ij}^{kl} = a_{ij}, \quad (i, j) \in N. \quad (3.19)$$

Comme  $\text{card}(D) = \text{card}(H) + 1$ , le système (3.18) est un système homogène de  $(m - 1)$  équations et  $m$  inconnues et donc a toujours une solution non triviale  $(b_{kl})_{(k,l) \in D}$  (les coefficients du

dénominateur). Si le rang de la matrice du système est  $m - 1$ , alors la solution est unique à un facteur multiplicatif près. Les coefficients  $(a_{ij})_{(i,j) \in N}$  du polynôme du numérateur sont donnés immédiatement par (3.19). On conclut ainsi que la partie principale du calcul d'un approximant de Frobenius-Padé à plusieurs variables concerne la résolution du système (3.18) et c'est donc à ce système que nous allons nous intéresser par la suite.

En utilisant le même type de techniques que dans le cas à une variable on obtient les relations suivantes qui permettent le calcul récursif des quantités  $(h_{ij}^{kl})$  :

$$\alpha_k h_{ij}^{k+1,l} = \alpha_{i-1} h_{i-1,j}^{kl} + \beta_i h_{ij}^{kl} + \gamma_{i+1} h_{i+1,j}^{kl} - \beta_k h_{ij}^{kl} - \gamma_l h_{ij}^{k-1,l}, \quad (3.20)$$

$$\alpha_l h_{ij}^{k,l+1} = \alpha_{j-1} h_{i,j-1}^{kl} + \beta_j h_{ij}^{kl} + \gamma_{j+1} h_{i,j+1}^{kl} - \beta_l h_{ij}^{kl} - \gamma_l h_{ij}^{k,l-1}, \quad (3.21)$$

avec  $h_{ij}^{00} = c_{ij}$ ,  $i, j \geq 0$  et  $h_{ij}^{kl} \gamma_{ij} = h_{kl}^{ij} \gamma_{kl}$ . ( $\gamma_{ij} = \|P_i\|^2 \|P_j\|^2$ ).

Supposons maintenant que nous voulions construire une suite d'approximations de  $f$ . Plus précisément,

- soit calculer les valeurs d'une suite d'approximants dans un point donné  $(x_0, y_0)$  - ce problème est étudié dans la section suivante ;
- soit calculer les coefficients des dénominateurs et numérateurs d'une suite d'approximants.

Pour définir un approximant on a besoin de spécifier trois ensembles d'indices : l'ensemble  $N$  des indices qui apparaissent dans le numérateur, l'ensemble  $D$  des indices qui apparaissent dans le dénominateur et celui correspondant aux termes dont le coefficient s'annule dans l'erreur,  $E$ . Ces ensembles doivent seulement vérifier

$$\text{card}(E) = \text{card}(D) + \text{card}(N) - 1, \quad N \subset E,$$

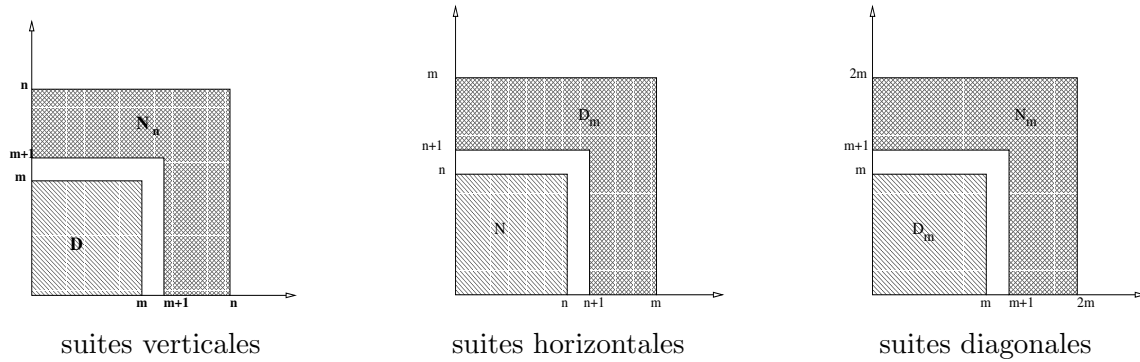
et donc peuvent être assez généraux. On a alors différentes façons de définir des suites d'approximants : fixer l'ensemble  $N$  correspondant au numérateur et augmenter le cardinal de  $D$  ("suites horizontales"), fixer l'ensemble  $D$  et augmenter le cardinal de  $N$  ("suites verticales"), augmenter simultanément les cardinaux de  $N$  et  $D$  ("suites diagonales"),... Pour pouvoir obtenir des algorithmes récursifs de calcul de la suite d'approximants on va considérer les deux choix suivants pour les ensembles d'indices

1. soient  $N, D$  et  $E$  vérifiant

- $E = \{(i, j) : 0 \leq i \leq m-1, 0 \leq j \leq m-1\}$ , avec  $\text{card}(E) = m^2$
- $N = \{(i, j) : 0 \leq i \leq m-l, 0 \leq j \leq m-l\}$
- $D = \{(i, j) : (m-l < i \leq m-1 \text{ et } 0 \leq j \leq m-1) \text{ ou } (0 \leq i \leq m-1 \text{ et } m-l < j \leq m-1)\}$ .

On peut aussi échanger les rôles de  $N$  et  $D$ . Les approximants construits à partir d'ensembles d'indices de ce type sont appelés *approximants de type produit tensoriel*. On définit des suites d'approximants correspondants aux choix d'indices indiqués dans les figures suivantes

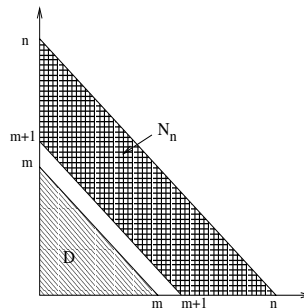




2. soient  $N, D$  et  $E$  vérifiant

- $E = \{(i, j) : 0 \leq i + j \leq m - 1\}$ ,  $\text{Card}(E) = \frac{m(m+1)}{2}$
- $N = \{(i, j) : 0 \leq i + j \leq m - l\}$
- $D = \{(i, j) : m - l + 1 \leq i + j \leq m - l\} \cup \{(0, 0)\}$ .

Les approximants construits à partir de ces ensembles d'indices sont appelés *approximants homogènes*. On peut ainsi définir comme dans le cas précédent des suites verticales, horizontales, diagonales d'approximants. Par exemple, pour les suites verticales, les ensembles d'indices considérés sont dessinés dans la figure :



suites verticales

On peut montrer que (voir [A13]), si l'on énumère de façon adéquate les ensembles d'indices concernés, pour les suites d'approximants considérées ci-dessus, le système à résoudre pour le calcul des coefficients du dénominateur d'un approximant de la suite s'obtient à partir du précédent (dans la suite) en rajoutant un paquet de lignes et colonnes. Ceci permet l'utilisation de la méthode de bordage pour la résolution de la suite des systèmes et donc le développement d'algorithmes récursifs de calcul.

Revenons maintenant au premier type de problème, c'est-à-dire, le calcul de la valeur en un point d'une suite d'approximants.

### 3.4.2 Un algorithme récursif

En utilisant la règle de Cramer pour résoudre les systèmes (3.19) and (3.18) on obtient la représentation sous forme de déterminant du numérateur et du dénominateur de l'approximant

de Frobenius-Padé :

$$Q(x, y) = \begin{vmatrix} P_{k_1}(x)P_{l_1}(y) & P_{k_2}(x)P_{l_2}(y) & \cdots & P_{k_m}(x)P_{l_m}(y) \\ h_{i_n+1j_{n+1}}^{k_1l_1} & h_{i_n+1j_{n+1}}^{k_2l_2} & \cdots & h_{i_n+1j_{n+1}}^{k_ml_m} \\ h_{i_n+2j_{n+2}}^{k_1l_1} & h_{i_n+2j_{n+2}}^{k_2l_2} & \cdots & h_{i_n+2j_{n+2}}^{k_ml_m} \\ \vdots & \vdots & \vdots & \vdots \\ h_{i_n+m-1j_{n+m-1}}^{k_1l_1} & h_{i_n+m-1j_{n+m-1}}^{k_2l_2} & \cdots & h_{i_n+m-1j_{n+m-1}}^{k_ml_m} \end{vmatrix}$$

$$P(x, y) = \sum_{(i,j) \in N} a_{ij} P_i(x) P_j(y) = \sum_{(i,j) \in N} \left( \sum_{(k,l) \in D} b_{kl} h_{ij}^{kl} \right) P_i(x) P_j(y) =$$

$$= \begin{vmatrix} \sum_{(i,j) \in N} h_{ij}^{k_1l_1} P_i(x) P_j(y) & \sum_{(i,j) \in N} h_{ij}^{k_2l_2} P_i(x) P_j(y) & \cdots & \sum_{(i,j) \in N} h_{ij}^{k_ml_m} P_i(x) P_j(y) \\ h_{i_n+1j_{n+1}}^{k_1l_1} & h_{i_n+1j_{n+1}}^{k_2l_2} & \cdots & h_{i_n+1j_{n+1}}^{k_ml_m} \\ h_{i_n+2j_{n+2}}^{k_1l_1} & h_{i_n+2j_{n+2}}^{k_2l_2} & \cdots & h_{i_n+2j_{n+2}}^{k_ml_m} \\ \vdots & \vdots & \vdots & \vdots \\ h_{i_n+m-1j_{n+m-1}}^{k_1l_1} & h_{i_n+m-1j_{n+m-1}}^{k_2l_2} & \cdots & h_{i_n+m-1j_{n+m-1}}^{k_ml_m} \end{vmatrix}.$$

En s'inspirant des idées développées en [38] pour le calcul récursif des approximants de Padé à plusieurs variables, et en manipulant par des combinaisons linéaires astucieuses les lignes et colonnes des déterminants ci-dessus on peut montrer que (voir **[A13]**) :

$$\frac{P(x, y)}{Q(x, y)} = \frac{\begin{vmatrix} s_1(n) & s_1(n+1) & \cdots & s_1(n+m-1) \\ g_1(n) & g_1(n+1) & \cdots & g_1(n+m-1) \\ \vdots & \vdots & \vdots & \vdots \\ g_{m-1}(n) & g_{m-1}(n+1) & \cdots & g_{m-1}(n+m-1) \end{vmatrix}}{\begin{vmatrix} 1 & 1 & \cdots & 1 \\ g_1(n) & g_1(n+1) & \cdots & g_1(n+m-1) \\ \vdots & \vdots & \vdots & \vdots \\ g_{m-1}(n) & g_{m-1}(n+1) & \cdots & g_{m-1}(n+m-1) \end{vmatrix}},$$

avec

$$\begin{cases} s_u(n) = \sum_{r=1}^n h_{i_r j_r}^{k_u l_u} P_{i_r}(x) P_{j_r}(y) / P_{k_u}(x) P_{l_u}(y), & u = 1, \dots, m, n \geq 0 \\ \Delta s_u(n) = s_u(n+1) - s_u(n) \end{cases}$$

$$g_i(n) = s_{i+1}(n) - s_i(n), \quad i = 1, 2, \dots, m-1.$$

Il est bien connu que les quotients de déterminants de cette forme peuvent être calculés récursivement par le E-algorithme [12] (voir Chapitre 1 section **1.4**).

Ceci permet de calculer récursivement la valeur en un point fixé  $(x, y)$  d'une suite d'approximants. Pour cela on fixe deux suites d'indices :

- $\{(i_1, j_1), (i_2, j_2), \dots, (i_n, j_n), \dots\}$  qui correspond aux indices des polynômes  $P_{i_u}(x) P_{j_u}(y)$  qui apparaissent dans les numérateurs et aux indices qui seront annulés dans le développement de l'erreur ;
- $\{(k_1, l_1), (k_2, l_2), \dots, (k_m, l_m), \dots\}$  qui correspond aux indices des polynômes  $P_{k_u}(x) P_{l_u}(y)$  apparaissant dans le dénominateur.

On pose, pour  $n, m \geq 0$ ,

$$N_n = \{(i_1, j_1), (i_2, j_2), \dots, (i_n, j_n)\} \text{ et } D_m = \{(k_1, l_1), (k_2, l_2), \dots, (k_m, l_m)\},$$

et l'on définit pour  $n \geq 0, m \geq 0$  l'approximant de Frobenius-Padé à deux variables par

$$S_{n,m}(x, y) = P(x, y)/Q(x, y)$$

avec  $N = N_n$ ,  $D = D_m$  et  $E = N_{n+m}$ . On a donc montré que, en appliquant le E-algorithme avec initialisations

$$\begin{cases} E_0^{(u)} &= \frac{\sum_{(i,j) \in N_u} h_{ij}^{k_1 l_1} P_i(x) P_j(y)}{P_{k_1(x)} P_{l_1(y)}} \quad u = 0, 1, \dots \\ g_{0,v}^{(u)} &= \frac{\sum_{(i,j) \in N_u} h_{ij}^{k_v+1 l_{v+1}} P_i(x) P_j(y)}{P_{k_v+1(x)} P_{l_{v+1}(y)}} - \frac{\sum_{(i,j) \in N_u} h_{ij}^{k_v l_v} P_i(x) P_j(y)}{P_{k_v(x)} P_{l_v(y)}} \quad u, v = 0, 1, \dots \end{cases}$$

on obtient

$$S_{n,m}(x, y) = E_m^{(n)} \text{ pour } m, n \geq 0.$$

On pourra bien sûr appliquer aussi les règles particulières du E-algorithme [13] pour éviter les instabilités numériques.

En disposant ces approximants dans une table à double entrée comme la table de Padé, on a obtenu un algorithme qui permet le calcul récursif de cette double suite de valeurs des approximants en un point donné.

### 3.4.3 Structure de la matrice de Frobenius-Padé

Intéressons-nous de nouveau à la résolution du second type de problèmes, c'est-à-dire, le calcul des coefficients du numérateur et du dénominateur d'une suite d'approximants. Nous avons déjà vu que pour certaines suites d'approximants on pouvait proposer une méthode récursive de calcul basée sur la méthode de bordage pour la résolution des systèmes linéaires. Nous allons maintenant déterminer, pour des choix particuliers des ensembles d'indices  $N, D$  et  $E$  (et donc pour certaines suites d'approximants) une "structure de déplacement" (en anglais, "displacement structure") dans la matrice du système permettant le calcul des coefficients du dénominateur (partie principale du calcul). Cette structure conduit au développement d'algorithmes rapides de calcul des coefficients de l'approximant.

On rappelle que, si l'on énumère les éléments dans les ensembles d'indices  $N, D$  et  $E$  correspondant à un approximant donné, de la façon suivante :

$$\begin{cases} D = \{(k_1, l_1), (k_2, l_2), \dots, (k_{n+1}, l_{n+1})\} \\ E \setminus N = \{(i_1, j_1), (i_2, j_2), \dots, (i_n, j_n)\} \end{cases}$$

et si l'on pose  $b_{k_{n+1} l_{n+1}} = 1$  (le système donnant les coefficients du dénominateur étant un système homogène de  $n$  équations à  $n+1$  inconnues), la matrice du système à résoudre pour le calcul des coefficients du dénominateur est donnée par

$$M = \begin{pmatrix} h_{i_1 j_1}^{k_1 l_1} & h_{i_1 j_1}^{k_2 l_2} & \dots & h_{i_1 j_1}^{k_n l_n} \\ h_{i_2 j_2}^{k_1 l_1} & h_{i_2 j_2}^{k_2 l_2} & \dots & h_{i_2 j_2}^{k_n l_n} \\ \dots & \dots & \dots & \dots \\ h_{i_n j_n}^{k_1 l_1} & h_{i_n j_n}^{k_2 l_2} & \dots & h_{i_n j_n}^{k_n l_n} \end{pmatrix}.$$

Comme les éléments de cette matrice vérifient certaines relations de récurrence, cela nous donne l'idée de chercher une structure de déplacement pour cette matrice.

Nous rappelons qu'une matrice  $M$  a une *structure de déplacement*  $\{Y, V\}$  ssi

$$VM - MY = GB \text{ où}$$

- $G \in \mathcal{M}_{n \times \alpha}, B \in \mathcal{M}_{\alpha \times n}$  les générateurs ;
- $\text{rang}(VM - MY) = \alpha$  petit par rapport à la dimension de la matrice  $n$ .

Des algorithmes rapides pour résoudre le système  $Mx = c$  ont été proposés pour différents types de matrices  $V$  et  $Y$  (voir, par exemple, [71] et les références citées). Tandis que l'élimination de Gauss appliquée à  $M$  nécessite  $\mathcal{O}(n^3)$  opérations, la structure de déplacement permet d'accélérer le procédé : en effet, comme les  $n^2$  éléments de  $M$  sont complètement déterminés par les éléments des générateurs  $\{G, B\}$ , on peut transférer le procédé d'élimination de Gauss et l'interpréter en termes d'opérations sur les générateurs, ce qui permet de définir des algorithmes "rapides" de résolution.

Revenons à notre problème et considérons le cas particulier suivant :

$$D = \{(i, j) : 0 \leq i, j \leq m-1\} \text{ et } E \setminus N = D \setminus \{(k_{n+1}, l_{n+1})\},$$

avec l'énumération

$$(0, 0), (0, 1), (0, 2), \dots, (0, m-1), (1, 0), (1, 1), \dots, (1, m-1), \dots, (m-1, 0), \\ \dots, (m-1, 1), \dots, (m-1, m-1).$$

Les relations de récurrence (3.20) et (3.21) obtenues ci-dessus pour le calcul récursif des quantités  $h_{ij}^{kl}$  fournissent une relation linéaire entre trois éléments consécutifs d'une ligne et trois éléments consécutifs d'une colonne de la matrice  $M$ , comme indiqué dans le schéma suivant :

$$\begin{array}{ccccc} & & h_{i-1,j}^{kl} & & h_{i,j-1}^{kl} \\ h_{ij}^{k-1,l} & & h_{ij}^{kl} & h_{ij}^{k+1,l} & \text{et } h_{ij}^{k,l-1} & h_{ij}^{kl} & h_{ij}^{k,l+1} \\ & & h_{i+1,j}^{kl} & & h_{i,j+1}^{kl} \end{array}$$

Cela suggère que, si l'on multiplie à gauche  $M$  par une matrice adéquate où l'on met les coefficients des relations (3.20) et (3.21) (qui correspondent aux coefficients de la relation de récurrence du système de polynômes orthogonaux  $\{P_k\}$ ), si l'on multiplie à droite  $M$  par la même matrice et si l'on soustrait les deux, on doit obtenir une matrice avec un grand nombre d'éléments nuls. Plus précisément, si l'on divise  $M$  en  $m \times m$  blocs chacun de dimension  $m \times m$ ,

$$M = \begin{pmatrix} M_{00} & M_{01} & \dots & M_{0,m-1} \\ M_{10} & M_{11} & \dots & M_{1,m-1} \\ \vdots & \vdots & \vdots & \vdots \\ M_{m-1,0} & M_{m-1,1} & \dots & M_{m-1,m-1} \end{pmatrix}, \quad M_{ij} = \begin{pmatrix} h_{i0}^{j0} & h_{i0}^{j1} & \dots & h_{i0}^{j,m-1} \\ h_{i1}^{j0} & h_{i1}^{j1} & \dots & h_{i1}^{j,m-1} \\ \vdots & \vdots & \vdots & \vdots \\ h_{i,m-1}^{j0} & h_{i,m-1}^{j1} & \dots & h_{i,m-1}^{j,m-1} \end{pmatrix},$$

avec  $0 \leq i, j \leq m-1$ , et si l'on pose

$$F = \begin{pmatrix} F_0 & 0 & \dots & \dots \\ 0 & F_0 & 0 & \dots \\ & & \ddots & \\ 0 & \dots & 0 & F_0 \end{pmatrix} \text{ avec } F_0 = \begin{pmatrix} \beta_0 & \gamma_1 & & & \\ \alpha_0 & \beta_1 & \ddots & & \\ & \alpha_1 & \ddots & \ddots & \\ & & \ddots & \ddots & \gamma_{m-1} \\ & & & \alpha_{m-2} & \beta_{m-1} \end{pmatrix},$$



Des résultats semblables peuvent être obtenus pour des ensembles plus généraux d'indices. En effet, on peut montrer que

**THÉORÈME 3.15.** [A13] *On considère l'approximant de Frobenius-Padé à deux variables de la fonction*

$$f(x, y) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} f_{ij} P_i(x) P_j(y)$$

défini par les ensembles d'indices

$$D = [i', i' + m - 1] \times [j', j' + m - 1] \text{ and } E \setminus N = [i^*, i^* + m - 1] \times [j^*, j^* + m - 1]$$

et soit  $M$  la matrice de dimension  $m^2 \times m^2$  du système à résoudre pour obtenir les coefficients du dénominateur. Alors  $M$  a une structure de déplacement de Hessenberg de rang  $4m$ .

Ce résultat permet de conclure que, pour une classe générale d'approximants de Frobenius-Padé correspondant à des ensembles carrés d'indices  $D$  et  $E \setminus N$ , la matrice  $M$  correspondant au système à résoudre pour le calcul des coefficients du dénominateur est une matrice "de type Hankel polynomial" [72], i.e., une matrice avec une structure de déplacement  $\{Y, V\}$  avec  $V$  et  $Y$  des matrices de Hessenberg (dans notre cas, tridiagonales). Des algorithmes rapides (de type Levinson, c'est-à-dire, qui fournissent une factorisation triangulaire de la matrice inverse) pour résoudre des systèmes avec une structure de déplacement de Hessenberg ont été proposés dans [54]. Dans [56], un algorithme de type Schur pour obtenir récursivement la factorisation triangulaire de matrices de type Hankel polynomial a été développé. Pour des matrices de dimension  $n \times n$  sa complexité est donnée par

$$C(n) = \mathcal{O}(M(n)n + n^2),$$

où  $M(n)$  est le coût de la multiplication de  $V, Y$  par un vecteur.

Dans notre cas, le coût de la résolution du système serait donc de  $\mathcal{O}(m^4)$  au lieu de  $\mathcal{O}(m^6)$  avec l'élimination de Gauss. Le développement de ce type d'algorithmes avec application au calcul des approximants de Frobenius-Padé à deux variables fait partie d'un travail en cours.

### 3.4.4 Approximants de Frobenius-Padé mixtes

#### (a) Définitions générales

Nous allons maintenant généraliser au cas de l'approximation de Frobenius-Padé la définition des "nested multivariate Padé approximants" proposée dans [52] pour des fonctions à deux variables données par leur développement en série entière. Cette approche consiste à appliquer l'approximation de Padé par rapport à la variable  $y$  aux coefficients de l'approximant de Padé par rapport à  $x$ , ce qui conduit à des algorithmes comportant des approximations à une variable (et donc des petits systèmes) et rapides en termes de calcul (on peut utiliser des méthodes de FFT), comparés à d'autres généralisations des approximants de Padé au cas de plusieurs variables. Ces approximants ont été étudiés dans [52] et des résultats de convergence donnés dans [53].

Soit  $f$  une fonction à deux variables donnée par son développement dans un système orthogonal  $\{P_k\}$  et on l'écrit sous la forme

$$f(x, y) = \sum_{j=0}^{\infty} \sum_{i=0}^{\infty} f_{ij} P_i(y) P_j(x) = \sum_{j=0}^{\infty} f_j(y) P_j(x).$$

(i) **Première étape** : on considère  $f$  comme fonction de  $x$  ( $y$  est considéré comme un paramètre) :

$$f_y(x) = \sum_{j=0}^{\infty} f_j(y) P_j(x).$$

On veut calculer deux polynômes  $Q_y(x)$  et  $P_y(x)$  de la forme

$$\begin{cases} Q_y(x) &= 1 + \sum_{i=1}^m b_i(y) x^i \\ P_y(x) &= \sum_{i=0}^n a_i(y) P_i(x) \end{cases}$$

tels que

$$Q_y(x) f_y(x) - P_y(x) = \mathcal{O}(P_{n+m+1}(x)), \quad (3.22)$$

ce qui signifie que les premiers  $n+m+1$  coefficients du développement dans le système orthogonal  $\{P_k\}$  s'annulent. C'est un type d'approximant de Frobenius-Padé pour la fonction d'une variable  $f_y(x)$  : la seule différence est que, pour simplifier les calculs, on exprime le dénominateur dans les puissances de  $x$  au lieu de considérer le système orthogonal  $\{P_k\}$ . Si l'on pose

$$x^i \sum_{j=0}^{\infty} f_j(y) P_j(x) = \sum_{j=0}^{\infty} f_j^i(y) P_j(x),$$

et si l'on incorpore l'expression de  $P_y$  et  $Q_y$  dans (3.22), on obtient

$$\sum_{j=0}^{\infty} \left( f_j(y) + \sum_{i=1}^m b_i(y) f_j^i(y) \right) P_j(x) - \sum_{i=0}^n a_i(y) P_i(x) = \mathcal{O}(P_{n+m+1}(x)).$$

Ceci conduit au système que doivent vérifier les  $b_i(y)$  et les  $a_j(y)$  pour que (3.22) soit satisfaite :

$$\sum_{i=1}^m b_i(y) f_j^i(y) + f_j(y) = 0 \quad j = n+1, \dots, n+m \quad (3.23)$$

$$\sum_{i=1}^m b_i(y) f_j^i(y) + f_j(y) = a_j(y) \quad j = 0, \dots, n. \quad (3.24)$$

(ii) **Deuxième étape** : on remplace les fonctions inconnues  $b_i(y)$ ,  $i = 0, \dots, m$  et  $a_j(y)$ ,  $j = 0, \dots, n$  (solution de notre problème) par des polynômes qui les approchent, de la façon suivante :

(a) pour  $i = 1, \dots, m$ , on remplace  $b_i(y)$  par un polynôme

$$b_i^*(y) = \sum_{k=0}^M b_{ik} y^k.$$

On a  $m(M+1)$  coefficients à calculer et l'on va les choisir de sorte que les premiers  $M+1$  coefficients du développement dans le système orthogonal  $\{P_k\}$  de chaque équation de (3.23) s'annulent. Les coefficients  $b_{kl}$  seront donc calculés à partir du système :

$$\begin{pmatrix} f_{n+1}^1(y) & f_{n+1}^2(y) & \cdots & f_{n+1}^m(y) \\ f_{n+2}^1(y) & f_{n+2}^2(y) & \cdots & f_{n+2}^m(y) \\ \vdots & \vdots & \cdots & \vdots \\ f_{n+m}^1(y) & f_{n+m}^2(y) & \cdots & f_{n+m}^m(y) \end{pmatrix} \begin{pmatrix} b_1^*(y) \\ b_2^*(y) \\ \vdots \\ b_m^*(y) \end{pmatrix} = - \begin{pmatrix} f_{n+1}(y) \\ f_{n+2}(y) \\ \vdots \\ f_{n+m}(y) \end{pmatrix},$$

en identifiant les coefficients des premiers  $M+1$  termes du développement dans la famille  $\{P_k\}$  de chaque équation. Pour cela on a besoin du résultat suivant :

**PROPOSITION 3.16. [A13]** *Soit  $f(x, y) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} f_{ij} P_i(x) P_j(y)$  une fonction de deux variables donnée par son développement dans un système de polynômes orthogonaux vérifiant la relation de récurrence à trois termes*

$$xP_k(x) = A_k P_{k+1}(x) + B_k P_k(x) + C_k P_{k-1}(x).$$

On pose comme ci-dessus

$$f_y(x) = \sum_{j=0}^{\infty} f_j(y) P_j(x), \quad f_j(y) = \sum_{k=0}^{\infty} f_{jk} P_k(y).$$

Alors

1. Les polynômes  $x^i P_k(x)$  s'expriment dans la base des  $\{P_k\}$  de la façon suivante

$$x^i P_k(x) = \alpha_{k, k+i}^i P_{k+i}(x) + \cdots + \alpha_{kk}^i P_k(x) + \cdots + \alpha_{k, k-i}^i P_{k-i}(x),$$

où les coefficients  $\alpha_{kj}^i$  se calculent par une récurrence sur l'indice supérieur  $i$  :

$$\begin{cases} \alpha_{k, k+i}^i &= A_{k+i-1} \alpha_{k, k+i-1}^{i-1} \\ \alpha_{k, k-i}^i &= C_{k-i+1} \alpha_{k, k-i+1}^{i-1} \\ \alpha_{kj}^i &= A_{j-1} \alpha_{k, j-1}^{i-1} + B_j \alpha_{kj}^{i-1} + C_{j+1} \alpha_{k, j+1}^{i-1}, \quad k-i+1 \leq j \leq k+i-1 \end{cases}.$$

2. On considère le développement de  $x^i f_y(x)$  dans le système orthogonal  $\{P_k\}$ ,

$$x^i f_y(x) = \sum_{j=0}^{\infty} f_j^i(y) P_j(x).$$

Alors le développement des fonctions  $f_j^i(y)$  dans le système  $\{P_k\}$  est donné par

$$f_j^i(y) = \sum_{k=0}^{\infty} f_{jk}^i P_k(y) \text{ avec } f_{jk}^i = \sum_{n=k-i}^{k+i} f_{nk} \alpha_{nj}^i$$



(b) on définit, pour  $j = 0, \dots, n$ , les polynômes

$$a_j^*(y) = \sum_{k=0}^N a_{jk} P_k(y),$$

en calculant les  $(n+1)(N+1)$  coefficients  $(a_{jk})$  de sorte que

$$a_j^*(y) - \sum_{i=1}^m b_i^*(y) f_j^i(y) = \mathcal{O}(P_{N+1}(y)).$$

(a) consiste dans la résolution d'un système linéaire de  $m(M+1)$  équations et de (b) on obtient immédiatement les coefficients  $a_{ij}$ .

DEFINITION 3.17. On appelle approximant de Frobenius-Padé mixte l'approximant rationnel

$$R(x, y) = \frac{P(x, y)}{Q(x, y)} \text{ avec } \begin{cases} P(x, y) = \sum_{i=0}^n \sum_{k=0}^N a_{ik} P_i(x) P_k(y) \\ Q(x, y) = 1 + \sum_{i=1}^m \sum_{k=0}^M b_{ik} x^i y^k \end{cases}$$

vérifiant

$$Q(x, y) f(x, y) - P(x, y) = \sum_{(i,j) \in (\mathbb{N}^2 \setminus E)} d_{ij} P_i(x) P_j(y), \text{ où}$$

$$E = \{(i, j) : (0 \leq i \leq n \text{ et } 0 \leq j \leq N) \text{ ou } (n+1 \leq i \leq n+m \text{ et } 0 \leq j \leq M)\}.$$

La forme et le cardinal de l'ensemble d'indices  $E$  dépendent du nombre de coefficients connus de la série  $f(x, y)$ . Si nous voulons une symétrie par rapport aux deux variables il est suffisant de prendre  $N = M = n + m$ . Mais dans ce cas on n'aura pas de symétrie de l'approximant (les mêmes puissances de  $x$  et de  $y$ ). Pour cela on doit prendre  $N = n$  et  $M = m$  et l'on privilégiera l'approximation en  $x$ . Ceci est une propriété très intéressante de ces approximants : on peut espérer d'obtenir de bonnes approximations quand la fonction a un ensemble non symétrique de singularités dû à la différente nature des variables. On remarque que l'on peut faire le même type de construction en commençant par la variable  $y$ .

Dans le cas général d'une famille quelconque de polynômes orthogonaux, on a besoin d'un nombre assez important de calculs pour construire la matrice du système donnant les coefficients du dénominateur et il semble difficile de trouver une structure pour cette matrice. Par contre, si l'on considère la famille des polynômes de Tchebychev, les calculs précédents deviennent très simples et l'on peut trouver une structure pour la matrice des coefficients. C'est ce que l'on va développer au paragraphe suivant.

### (b) Une classe d'approximants de Padé-Tchebychev à deux variables

Soit  $f(x, y)$  une fonction donnée par

$$f(x, y) = \sum_{i,j \geq 0} f_{ij} T_i(x) T_j(y) = \sum_{j=0}^{\infty} f_j(y) T_j(x)$$

où  $\{T_k\}$  est la famille des polynômes de Tchebychev de première espèce. On procède comme au paragraphe précédent pour construire les *approximants de Padé-Tchebychev mixtes* :

(i) **Première étape** : on construit un approximant linéarisé de Padé-Tchebychev de la fonction de  $x$ ,  $f_y(x) = f(x, y)$ , c'est-à-dire, deux polynômes  $Q_y(x)$  et  $P_y(x)$  :

$$\begin{cases} Q_y(x) &= \sum_{i=0}^m b_i(y)T_i(x) \\ P_y(x) &= \sum_{i=0}^n a_i(y)T_i(x) \end{cases}$$

vérifiant

$$Q_y(x)f_y(x) - P_y(x) = \mathcal{O}(T_{n+m+1}(x)).$$

On rappelle une propriété fondamentale des polynômes de Tchebychev

$$T_k(x)T_i(x) = \frac{1}{2}(T_{k+i}(x) + T_{k-i}(x)) \quad i \geq 0, k \geq 0, \quad (3.25)$$

avec la convention  $T_{-i}(x) = T_i(x)$  pour  $i \geq 0$ . En utilisant cette propriété et en choisissant  $m \leq n$ , on peut montrer que, si l'on fixe  $b_0(y) = 1$ , les  $b_j(y)$ ,  $j = 1, \dots, m$  sont solution du système :

$$H^m(y)B(y) = F(y) \text{ avec } F(y)^T = 2 \begin{pmatrix} f_{n+1}(y) & \cdots & f_{n+m}(y) \end{pmatrix},$$

$$H^m(y) = \begin{pmatrix} f_n(y) + f_{n+2}(y) & f_{n-1}(y) + f_{n+3}(y) & \cdots & f_{n+1-m}(y) + f_{n+1+m}(y) \\ \cdots & \cdots & \cdots & \cdots \\ f_{n+m-1}(y) + f_{n+m+1}(y) & f_{n+m-2}(y) + f_{n+m+2}(y) & \cdots & f_n(y) + f_{n+2m}(y) \end{pmatrix},$$

$$B(y) = \begin{pmatrix} b_1(y) \\ \vdots \\ b_m(y) \end{pmatrix} = \sum_{i=0}^{\infty} B_i T_i(y).$$

Les  $a_k(y)$  sont alors calculés à partir des  $b_j(y)$  par

$$\begin{cases} a_0(y) &= \frac{1}{2} \sum_{k=0}^m b_k(y) f_k(y) \\ a_i(y) &= \frac{1}{2} \left[ f_0(y) b_i(y) + \sum_{k=0}^m b_k(y) (f_{i+k}(y) + f_{i-k}(y)) \right] \quad i = 1, \dots, n. \end{cases}$$

(ii) **Deuxième étape** : on remplace les fonctions  $a_i(y)$  et  $b_i(y)$  par des polynômes qui les approchent (écrits dans la base de Tchebychev).

En utilisant encore une fois la relation (3.25) et en regroupant les termes avec le même indice dans le développement en série de Tchebychev, nous obtenons un système structuré pour le calcul des coefficients du dénominateur, et pouvons ainsi proposer un approximant rationnel facilement calculable qui vérifie une certaine condition d'ordre linéaire :  $Q(x, y)f(x, y) - P(x, y) = \mathcal{O}(T_{n+m}(x)T_m(y))$ . Ceci est résumé dans le résultat suivant :

**THÉORÈME 3.18. [A13]** *Soit  $f$  une fonction à deux variables donnée par son développement en série de Tchebychev*

$$f(x, y) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} f_{ij} T_i(x) T_j(y) = \sum_{j=0}^{\infty} f_j(y) T_j(x).$$

On définit l'approximant de Padé-Tchebychev mixte par

$$PTche_{m,n}(x, y) = P(x, y)/Q(x, y) \text{ avec}$$

$$\left\{ \begin{array}{l} P(x, y) = \sum_{j=0}^m b_j^*(y) T_j(x) \text{ avec } b_j^*(y) = \sum_{k=0}^m b_{jk}^* T_k(y) \\ Q(x, y) = \sum_{i=0}^n a_i^*(y) T_i(x) \text{ avec } a_i^*(y) = \sum_{k=0}^m a_{ik}^* T_k(y) \end{array} \right.$$

où les coefficients sont calculés de la façon suivante :

- les coefficients du dénominateur  $b_{ij}^*$ , regroupés dans un vecteur  $\mathcal{B} = (B_0^*, B_1^*, \dots, B_m^*)^T$ ,  $B_j^* = (b_{j1}^*, \dots, b_{jm}^*)^T$ ,  $j = 0, \dots, m$ , sont la solution du système de dimension  $(m(m+1)) \times (m(m+1))$  :

$$\sum_{j=0}^m (H_{j-k} + H_{j+k}) B_j^* = F_k \quad k = 0, \dots, m \quad \Leftrightarrow \quad \mathcal{H}\mathcal{B} = \mathcal{F} \quad (3.26)$$

avec

$$\mathcal{H} = \begin{pmatrix} H_0 & H_1 & \cdots & H_m \\ H_{-1} & H_0 & \cdots & H_{m-1} \\ \vdots & \vdots & & \vdots \\ H_{-m} & H_{-m+1} & \cdots & H_0 \end{pmatrix} + \begin{pmatrix} H_0 & H_1 & \cdots & H_m \\ H_1 & H_2 & \cdots & H_{m+1} \\ \vdots & \vdots & & \vdots \\ H_m & H_{m+1} & \cdots & H_{2m} \end{pmatrix} = \mathcal{H}_1 + \mathcal{H}_2$$

$\mathcal{H}_1$  une matrice Toeplitz par blocs et  $\mathcal{H}_2$  une matrice Hankel par blocs ;

- le numérateur est donné par

$$\left\{ \begin{array}{l} a_0^*(y) = \frac{1}{2} \sum_{k=0}^m b_k^*(y) f_k^{(m)}(y) \\ a_i^*(y) = \frac{1}{2} \left[ f_0^{(m)}(y) b_i^*(y) + \sum_{k=0}^m b_k^*(y) (f_{i+k}^{(m)}(y) + f_{i-k}^{(m)}(y)) \right] \end{array} \right. \quad i = 1, \dots, n,$$

où  $f_j^{(m)}(y) = \sum_{k=0}^m f_{jk} T_k(y)$  sont les sommes partielles d'ordre  $m$ .

Alors les polynômes  $P(x, y)$  et  $Q(x, y)$  vérifient

$$Q(x, y)f(x, y) - P(x, y) = R(x, y) = \sum_{i,j \geq 0} r_{ij} T_i(x) T_j(y)$$

avec

$$r_{ij} = 0 \text{ pour } i = 0, \dots, n+m \text{ et } j = 0, \dots, m.$$

Ce résultat nous permet de proposer des algorithmes rapides pour le calcul des coefficients du dénominateur. Ils seront basés sur les algorithmes d'inversion rapide de matrices de la forme Toeplitz-plus-Hankel développés dans [55]. Les formules et algorithmes donnés dans [55] peuvent être facilement généralisés au cas de matrices par blocs, et l'on peut montrer que la solution du système (3.26) peut être obtenue avec une complexité de  $\mathcal{O}(m^2(m+1)^2)$  (au lieu de  $\mathcal{O}(m^6)$  en utilisant l'élimination de Gauss). L'implémentation de ces algorithmes est en cours de développement.

### 3.4.5 Conclusion et perspectives

Nous nous sommes intéressés dans cette dernière partie à l'approximation rationnelle de séries orthogonales à deux variables. Nous avons proposé deux définitions d'approximants rationnels à deux variables pour des fonctions données par leur développement en série orthogonale : les *approximants de Frobenius-Padé à deux variables* et les *approximants de Frobenius-Padé mixtes*. Ces définitions ont été inspirées des différentes définitions des approximants de Padé (pour des séries entières) à plusieurs variables. Nous nous sommes essentiellement intéressés à la façon les calculer. Ainsi on a proposé différents algorithmes de calcul de suites de ces approximants (calcul explicite des coefficients) - algorithmes récursifs et algorithmes rapides basés sur la structure de la matrice - et de suites de valeurs de ces approximants en un point.

Mais beaucoup de problèmes restent ouverts et font actuellement l'objet de mes recherches. Plus particulièrement,

- les propriétés de convergence et d'accélération de convergence de suites de ces approximants. Dans le cas d'une variable, nous avons obtenu de très bons résultats numériques et de bonnes propriétés d'accélération de convergence pour certaines classes de fonctions, en particulier au voisinage des singularités (voir [86]). Nous espérons que ces bonnes propriétés de convergence puissent se généraliser au cas de plusieurs variables : nous essayons d'obtenir des propriétés d'accélération de convergence de ces approximants à partir de propriétés de la fonction  $f$  ou de la suite des coefficients du développement  $(f_{ij})$
- programmer et développer les différents algorithmes proposés dans cette partie, en étudiant leur stabilité numérique
- étude d'applications de ces approximants à différents domaines des mathématiques appliquées : recherche de problèmes d'analyse numérique où on dispose d'une approximation de la solution sous forme d'une somme partielle de série orthogonale, que l'on cherche à améliorer notamment en présence de singularités.

# Bibliographie

- [1] R.P. Agarwal, *Difference Equations and Inequalities - Theory, Methods and Applications*, Marcel Dekker, Inc., New York (1992).
- [2] A.C. Aitken, On Bernoulli's numerical solution of algebraic equations, Proc. R. Soc. Edinb., 46 (1926) 289–305.
- [3] G.A. Baker Jr., P.R. Graves–Morris, *Padé Approximants*, 2 vols., Addison–Wesley, Reading, 1981.
- [4] G.A. Baker Jr., P. Graves-Morris, *Padé Approximants*, Encyclopedia of Mathematics and its Applications, Vol.59, Cambridge University Press, 1997.
- [5] S. Banach, *Théorie des Opérations Linéaires*, Monografie Matematyczne, vol.1, Warszawa, 1932.
- [6] M.D. Benchiboun, *Étude de Certaines Généralisations du  $\Delta^2$  d'Aitken et Comparaison de Procédés d'Accélération de la Convergence*, Thèse 3ème cycle, Université des Sciences et Technologies de Lille, 1987.
- [7] C.M. Bender and S.A. Orszag, *Advanced Mathematical Methods for Scientists and Engineers*. McGraw-Hill International Editions, Mathematics Series, 1987.
- [8] C. Brezinski, Accélération de suites à convergence logarithmique, C.R. Acad. Sci. Paris, 273 A (1971) 727–730.
- [9] C. Brezinski, Etudes sur les  $\varepsilon$  et  $\rho$ –algorithmes, Numer. Math., 17 (1971) 153–162.
- [10] C. Brezinski, Conditions d'application et de convergence de procédés d'extrapolation, Numer. Math., 20 (1972) 64–79.
- [11] C. Brezinski, *Padé–Type Approximation and General Orthogonal Polynomials*, ISNM vol.50, Birkhäuser, Basel, 1980.
- [12] C. Brezinski, A general extrapolation algorithm, Numer. Math., 35 (1980) 175–187.
- [13] C. Brezinski, The Mulhbach-Neville-Aitken algorithm and some extensions, BIT 20 (1980) 444–451.
- [14] C. Brezinski, Some new convergence acceleration methods, Math. Comput., 39 (1982) 133–145.
- [15] C. Brezinski, Algorithm 585 : A subroutine for the general interpolation and extrapolation problems, ACM Trans. Math. Software, 8 (1982) 290–301.
- [16] C. Brezinski, Error control in convergence acceleration processes, IMA J. Numer. Anal., 3 (1983) 65–80.

- [17] C. Brezinski, Recursive interpolation, extrapolation and projection, *J. Comput. Appl. Math.*, 9 (1983) 369–376.
- [18] C. Brezinski, Composite sequence transformations, *Numer. Math.*, 46 (1985) 311–321.
- [19] C. Brezinski, Quasi-linear extrapolation processes, in *Numerical Mathematics. Singapore 1988*, R.P. Agarwal et al. eds., ISNM vol. 86, Birkhäuser, Basel, 1988, pp.61–78.
- [20] C. Brezinski, A new approach to convergence acceleration methods, in *Nonlinear Numerical Methods and Rational Approximation*, A. Cuyt ed., Reidel, Dordrecht, 1988, pp.373–405.
- [21] C. Brezinski, A survey of iterative extrapolation by the  $E$ -algorithm, *Det Kong. Norske Vid. Selsk. skr.*, 2 (1989) 1–26.
- [22] C. Brezinski, Algebraic properties of the  $E$ -transformation, in *Numerical Analysis and Mathematical Modelling*, Banach Center Publications, vol. 24, PWN, Warsaw, 1990, pp.85–90.
- [23] C. Brezinski, The asymptotic behavior of sequences and new series transformations based on the Cauchy product, *Rocky Mt. J. Math.*, 21 (1991) 71–84.
- [24] C. Brezinski, *Biorthogonality and its Applications to Numerical Analysis*, Marcel Dekker, New York, 1992.
- [25] C. Brezinski, A.C. Matos, A derivation of extrapolation algorithms based on error estimates, *J. Comput. Appl. Math.* 66 (1996) 5–26.
- [26] C. Brezinski, M. Redivo Zaglia, *Extrapolation Methods. Theory and Practice*, North-Holland, Amsterdam, 1991.
- [27] C. Brezinski, M. Redivo Zaglia, Construction of extrapolation processes, *Appl. Numer. Math.*, 8 (1991) 11–23.
- [28] C. Brezinski, M. Redivo Zaglia, H. Sadok, A breakdown-free Lanczos type algorithm for solving linear systems, *Numer. Math.* 63 (1992) 29–38.
- [29] C. Brezinski, M. Redivo Zaglia, On the kernel of sequence transformations, *Appl. Numer. Math.* 16 (1994) , 239–244.
- [30] C. Brezinski, M. Redivo Zaglia, A general extrapolation procedure revisited, *Advances Comput. Math.*2 (1994) 461–477.
- [31] C. Brezinski, M. Redivo-Zaglia, The PageRank vector : properties, computation, approximation, and acceleration, *SIAM J. Matrix Anal. Appl.*, to appear.
- [32] C. Brezinski, A. Salam, Matrix and vector sequence transformations revisited, *Proc. Edinb. Math. Soc.*, to appear.
- [33] C. Brezinski, J. Van Iseghem, Padé approximations, in *Handbook of Numerical Analysis, vol. III*, P.G. Ciarlet and J.L. Lions eds., North-Holland, Amsterdam, 1994, pp.47–222.
- [34] C. Brezinski, G. Walz, Sequences of transformations and triangular recursion schemes with applications in numerical analysis, *J. Comput. Appl. Math.*, 34 (1991) 361–383.
- [35] J.S. Chisholm, Rational Approximants defined from double power series, *Math. of Comput.* 27 (1973) 841–848.
- [36] Clenshaw, C.N., and Lord, K., Rational Approximations from Chebyshev Series, in *Studies in Numerical Analysis*, Academic Press, London (1974), 95–113.

- [37] F. Cordellier, *Interpolation Rationnelle et Autres Questions : Aspects Algorithmiques et Numériques*, Thèse d'Etat, Université des Sciences et Technologies de Lille, 1989.
- [38] A. Cuyt, Multivariate Padé approximants revisited, BIT 26 (1986) 71-79.
- [39] A. Cuyt, Multivariate Padé approximants, J. Math. Anal. Appl. 96 (1983) 283-293.
- [40] A. Cuyt, A comparison of some multivariate Padé-approximants, SIAM J. Math. Anal., Vol.14 (1983) 195-202.
- [41] A. Cuyt, A review of multivariate Padé Approximation theory, J. Comput. Appl. Math. 12 (1985), 221-232.
- [42] A. Cuyt, How well can the concept of Padé approximant be generalized to the multivariate case? , J. Comput. Appl. Math. 105 (1999) 25-50.
- [43] P.J. Davis, *Interpolation and Approximation*, Dover, New York, 1975.
- [44] M. G. de Bruin , Generalized Padé tables and Some Algorithms Therein, in *Proceedings of First French-Polish Meeting on Padé Approximation and Convergence Acceleration Techniques*, J. Gilewicz ed (CNRS), 1982, 1-10.
- [45] M. G. de Bruin, Simultaneous Padé approximants and orthogonality, in *Polynômes orthogonaux et applications*, C. Brezinski et al. Editors, Lecture Notes in Mathematics 1171, Springer Verlag (1984), 74-83.
- [46] M. G. de Bruin, Zeros of Polynomials generated by 4-term Recurrence Relations, in *Rational Approximation and Interpolation*, P.R. Graves-Morris, E.B. Saff and R.S. Varga eds, Springer (Heidelberg, 1984), 331-345.
- [47] J.P. Delahaye, *Sequence Transformations*, Springer Verlag, Berlin, 1988.
- [48] A. Draux, *Polynômes Orthogonaux Formels. Applications*, LNM 974, Springer-Verlag, Berlin, 1983.
- [49] J.E. Drummond, A formula for accelerating the convergence of a general series, Bull. Aust. Math. Soc., 6 (1972) 69-74.
- [50] L. Emmel, S. Kaber Y. Maday, Padé-Jacobi filtering for spectral approximations of discontinuous solutions, Numerical Algorithms, 33 (2003) 251-264.
- [51] V.N. Faddeeva, *Computational Methods of Linear Algebra*, Dover, New York, 1959.
- [52] P. Guillaume, Nested multivariate Padé approximants, Journ. of Comp. and Appl. Math., 82 (1997), 149-158.
- [53] P. Guillaume, Convergence of the Nested multivariate Padé approximants, J. of Approx. Theory, 94 (1998) 455-566.
- [54] G. Heinig, K. Rost, *Algebraic Methods for Toeplitz-like matrices and operators*, Akademie-Verlag Berlin and Birkhauser Basel, Boston, Stuttgart 1984.
- [55] G. Heinig, P. Jankowski, K. Rost, Fast Inversion Algorithms of Toeplitz-Plus-Hankel Matrices, Numer. Math. 52 (1988), 665-682.
- [56] G. Heinig, V. Olshevski, The Schur algorithm for matrices with Hessenberg displacement structure , in V. Olshevski, editor, *Structured Matrices in Mathematics, Computer Science and Engeneering II*, vol 281 of Contemporary Mathematics. AMS Publications (2001).
- [57] F.R. Gantmacher, *The Theory of Matrices*, Chelsea, New York, 1959.

- [58] J. Fleischer, Nonlinear Padé Approximants for Legendre Series, *J. of Math. and Physics*, 14 (1973), 246–248.
- [59] B. Germain–Bonne, *Estimation de la Limite de Suites et Formalisation de Procédés d’Accélération de la Convergence*, Thèse d’État, Université des Sciences et Technologies de Lille, 1978.
- [60] A.A. Gonchar, E.A. Rakhmanov, S.P. Suetin, On the Rate of Convergence for Padé Approximants of Orthogonal Expansions, in *Progress in Approximation Theory*, Gonchar, A.A. and Saff, E.B. eds, Springer–Verlag (1992) 169–190.
- [61] P.R. Graves-Morris, J.M. Wilkins, A fast algorithm to solve Kalman’s partial realisation problem for single input, multi-output systems, in *Approximation theory*, E.B. Saff (ed.), 1–20. Springer Lecture Notes in Mathematics, 1287, New York (1987).
- [62] T. Hávie, Generalized Neville type extrapolation schemes, *BIT*, 19 (1979) 204–213.
- [63] J.T. Holdeman Jr., A Method for the Approximation of Functions Defined by Formal Series Expansions in Orthogonal Polynomials, *Math. of Comput.*, 23 (1969), 275–287.
- [64] H.H.H. Homeier, Some applications of nonlinear convergence accelerators, *Int. J. Quantum Chem.*, 45 (1993) 545–562.
- [65] R. Hughes Jones, G.J. Makinson, The generation of Chisholm rational polynomial approximants to power series in two variables, *J. Inst. Maths. Applics.* (1974) 13, 299–310.
- [66] A. Iserles, S.P. Norsett, Bi-orthogonal polynomials, in *Orthogonal Polynomials and Their Applications*, C. Brezinski, A. Draux, A.P. Magnus, P. Maroni and A. Ronveaux eds., LNM 1171, Springer-Verlag, Berlin, 1985, pp.92–100.
- [67] A. Iserles, S.P. Norsett, On the theory of biorthogonal polynomials, *Trans. Amer. Math. Soc.*, 306 (1988) 455–474.
- [68] S. Kaber, Y. Maday, Analysis of some Padé-Chebyshev approximants, *SIAM Journal of Numerical Analysis*, vol. 43 (2005) , 437–454.
- [69] S. Kaber, J. Hesthaven, L. Lurati, Padé-Legendre interpolants for Gibbs reconstruction, *Journal of Scientific Computation*, à paraître.
- [70] S. Karlin, *Total Positivity*, vol. 1, Stanford University Press, Stanford (1962).
- [71] T. Kailath, *Fast reliable algorithms for matrices with structure*, in T.Kailath, A.H. Sayed, eds, SIAM Publications, Philadelphia 1999.
- [72] T. Kailath, V. Olshevski, Displacement structure approach to discrete transform based preconditioners of G.Strang type and of T. Chan type, *Calcolo*, 33 (1996), 191–208.
- [73] V. Lakshmikantham, D. Trigiante, *Theory of Difference Equations : Numerical Methods and Applications*. Mathematics in Science and Engineering, Vo.l 181. Academic Press (1988)
- [74] Lebedev, N.N., *Special Functions and their Applications*, Dover Publications, Inc, New York, 1972.
- [75] D. Levin, Development of non-linear transformations for improving convergence of sequences, *Int. J. Comput. Math.*, B3 (1973) 371–388.
- [76] D. Levin, General Order Padé type rational approximants defined from double power series, *J. Inst. Maths. Applics.* (1976) 18, 1–8.
- [77] H. Levy, F. Lessman, *Finite Difference Equations*, Macmillan, NewYork (1961).



- [78] J.C. Mason, Some applications and drawbacks of Padé Approximants, in *Approximation Theory and Applications*, Zvi Ziegler ed., Academic Press, New York, 1981, pp.207–223.
- [79] A.C. Matos, Acceleration methods based on convergence tests, *Numer. Math.*, 58 (1990) 329–340.
- [80] A.C. Matos, Construction of new transformations for lacunary power series based on the Cauchy–type approximants, *Appl. Numer. Math.*, 7 (1991) 493–507.
- [81] A.C. Matos, On the choice of the denominator in Padé and Cauchy–type approximation, in *Orthogonal Polynomials and their Applications*, C. Brezinski et al. eds., J.C. Baltzer, Basel, 1991, pp.347–352.
- [82] A.C. Matos, Some new acceleration methods for periodic–linearly convergent power series, *BIT*, 31 (1991) 686–696.
- [83] A.C. Matos, Acceleration results vor the vector E-algorithm, *Numer. Algorithms* 1, 237-260 (1991).
- [84] A.C. Matos, Linear difference operators and acceleration methods, *IMA Journ. of Numer. Analysis* (2000) 20, 359-388.
- [85] A.C. Matos, Some convergence results for the generalized Padé type approximants, *Numerical Algorithms* 11 (1996), 255–269.
- [86] A.C. Matos, Recursive computations of Padé-Legendre approximations and some acceleration properties, *Numer.Math.* 89 (2001), 535-560.
- [87] A.C. Matos, J. Van Iseghem, Simultaneous Frobenius-Padé approximants, *J. of Comput. Appl. Math.* 176 (2005) 231-258.
- [88] J. Matos, *Algoritmos de cálculo de Aproximantes de Frobenius-Padé e Generalizações*, PhD Thesis, University of Porto (Portugal ), July 2003.
- [89] L.M. Milne–Thomson, *The Calculus of Finite Differences*, Chelsea, New York, 1954.
- [90] M.Morandi Cecchi, M. Redivo Zaglia, A new recursive algorithm for a Gaussian quadrature formula via orthogonal polynomials ,in *Orthogonal polynomials and their Applications*, C. Brezinski, L. Gori and A. Ronveaux (editors), J.C. Baltzer AG, IMACS, 1991, pp.353–358.
- [91] G. Mühlbach, A recurrence formula for generalized divided differences and some applications, *J. Approx. Theory*, 9 (1973) 165–172.
- [92] K.J. Overholt, Extended Aitken acceleration, *BIT*, 5 (1965) 122–132.
- [93] G. Mühlbach, Neville-Aitken algorithms for interpolating by functions of Tchebychev-systems in the sense of Newton and in a generalized sense of Hermite. In A.G. Law and B. N. Sahney, eds., *Theory of Approximation with Applications*, PP. 200-212, New-York, 1976. Academic Press.
- [94] S. Paszkowski, Approximation uniforme des fonctions continues par des fonctions rationnelles, *Zastosow. Mat.*, 6 (1963) 441-458.
- [95] R. Powell, S.M. Shah, *Summability Theory and its Applications*, Van Nostrand Reinhold, London, 1972 ; 2nd revised edition, Prentice–Hall of India, New Delhi, 1988.
- [96] L.F. Richardson, The approximate arithmetic solution by finite differences of physical problems involving differential equations, with an application to the stress in a masonry dam, *Phil. Trans. Roy. Soc. London, ser. A*, 210 (1910) 307–357.

- [97] Z. Rocha, *Applications de la Théorie de la Biorthogonalité*, Thèse, Université des Sciences et Technologies de Lille, 1994.
- [98] W. Romberg, Vereinfachte numerische Integration, Kgl. Norske Vid. Selsk. Forsk., 28 (1955) 30–36.
- [99] A. Salam, On a generalization of the  $\varepsilon$ -algorithm, J. Comput. Appl. Math., 46 (1993) 455–464.
- [100] A. Salam, Non-commutative extrapolation algorithms, Numerical Algorithms, 7 (1994) 225–251.
- [101] D. Shanks, Non linear transformations of divergent and slowly convergent sequences, J. Math. Phys., 34 (1955) 1–42.
- [102] A. Sidi, An algorithm for a special case of a generalization of the Richardson extrapolation process, Numer. Math., 38 (1982) 299–307.
- [103] A. Sidi, On a generalization of the Richardson extrapolation process, Numer. Math., 57 (1990), 365–377.
- [104] A.C. Smith, A one-parameter method for accelerating the convergence of sequences and series, Comp. and Maths. with Appls., 4 (1978) 157–171.
- [105] D.A. Smith, W.F. Ford, Acceleration of linear and logarithmic convergence, SIAM J. Numer. Anal., 16 (1979) 223–240.
- [106] V.N. Sorokin, J. Van Iseghem, Algebraic aspects of matrix orthogonality for vector polynomials, J. of Approx. Theory 90 (1997), 97–116.
- [107] H. Stahl, V. Totik, *General Orthogonal Polynomials*, Encyclopedia of Mathematics and Its Applications, Cambridge University Press, 1992.
- [108] S.P. Suetin, On the Convergence of Rational Approximations to Polynomial Expansions in Domains of Meromorphy of a Given Function, Math. USSR Sbornik, 34 (1978), 367–381.
- [109] S.P. Suetin, On De Montessus de Ballore’s Theorem for Nonlinear Padé Approximants of Orthogonal Series and Faber Series, Soviet Math. Dokl., 22 (1980), 274–277.
- [110] S.P. Suetin, On Montessus De Ballore’s Theorem for Rational Approximants of Orthogonal Expansions, Math. USSR Sbornik 42 (1982), 399–411.
- [111] G. Szegő, *Orthogonal Polynomials*, American Mathematical Society, Colloquium Publications Volume XXIII, Providence, 1978.
- [112] W. Van Assche, *Asymptotics for Orthogonal Polynomials*, Lecture Notes in Mathematics, Vol. 1265, Springer-Verlag, NewYork, 1987.
- [113] I. Van den Berg, *Nonstandard Asymptotic Analysis*, Lecture Notes in Mathematics 1249 (Springer, Berlin, 1980).
- [114] J. Van Iseghem, Vector orthogonal relations. Vector Q.D algorithm, J. Comput. Appl. Math. 19 (1987), 141–150.
- [115] J.L. Walsh, *Interpolation and Approximation by Rational Functions in the Complex Domain*, American Mathematical Society, Colloquium Publications Volume XX, Providence, 1969.
- [116] G. Walz, *Asymptotics and Extrapolation*, AkademieVerlag, Berlin, 1996.

- [117] E.J. Weniger, Nonlinear sequence transformations for the acceleration of convergence and the summation of divergent series, *Computer Physics Reports*, 10 (1989)189–371.
- [118] E.J. Weniger, On the derivation of iterated sequence transformations for the acceleration of convergence and the summation of divergent series, *Comput. Phys. Commun.*, 64 (1991) 19–45.
- [119] J. Wimp, Toeplitz arrays, linear sequence transformations and orthogonal polynomials, *Numer. Math.*, 23 (1974) 1–17.
- [120] J. Wimp, *Sequence Transformations and Their Applications*, Academic Press, New York, 1981.
- [121] J. Wimp, *Computation with Recurrence Relations*, Pitman, Boston, 1984.
- [122] P. Wynn, On a device for computing the  $e_m(S_n)$  transformation, *MTAC*, 10 (1956) 91–96.
- [123] P. Wynn, On a procrustean technique for the numerical transformation of slowly convergent sequences and series, *Proc. Cambridge Phil. Soc.*, 52 (1956) 663–671.
- [124] P. Wynn, Acceleration techniques for iterated vector and matrix problems, *Math. Comput.*, 16 (1962) 301–322.

## Annexe A : Liste des publications correspondantes à ce document

- [A1] Construction of New Transformations for Lacunary Power Series Based on the Cauchy-Type Approximants, *Applied Numerical Mathematics* 7 (1991), 493–507.
- [A2] On the Choice of the Denominator in Padé and Cauchy-Type Approximation, in Orthogonal Polynomials and Their Applications. C. Brezinski, L. Gori and A. Ronveaux (editors) J.C. Baltzer AG, IMACS (1991), 347–352.
- [A3] Some New Acceleration Methods for Periodic-Linearly Convergent Power Series, *BIT* 31 (1991), 686–696.
- [A4] Acceleration Property for the E-Algorithm, (en collaboration avec Marc Prévost) *Numerical Algorithms* 2 (1992), 393–408.
- [A5] Least Squares Orthogonal Polynomials, (en collaboration avec Claude Brezinski), *Journal of Computational and Applied Mathematics* 46 (1993), 229–240.
- [A6] A Derivation of Extrapolation Algorithms Based on Error Estimates, (en collaboration avec Claude Brezinski), *Journal of Computational and Applied Mathematics* 66 (1996), 5–26.
- [A7] Some Convergence results for the generalized Padé-Type Approximants *Numerical Algorithms* 11 (1996), 255–269.
- [A8] Integral Representation of the Error and Asymptotic Error Bounds for the Generalized Padé Type Approximants, *Journal of Computational and Applied Mathematics* 77 (1997), 239–254.
- [A9] Linear Difference Operators and Acceleration Methods, *IMA Journal of Numerical Analysis* 20 (2000), 359–388.
- [A10] Least Squares Orthogonal Polynomials and Applications, (en collaboration avec C. Brezinski) in *Encyclopedia of Optimization*, C.A. Floudas and P.M. Pardalos Editors, Kluwer, Dordrecht (2001), 155–160.
- [A11] Recursive Computation of Padé–Legendre Approximants and Some Acceleration Properties, *Numerische Mathematik* 89 (2001), 535–560.
- [A12] Simultaneous Frobenius-Padé approximants, (en collaboration avec J. Van Iseghem), *Journal of Computational and Applied Mathematics*, 176 (2005), 231–258.
- [A13] Multivariate Frobenius Padé Approximants, *Journal of Computational and Applied Mathematics*, (à paraître) 2006.

**Annexe B : Autres publications**

- [B1] Acceleration Methods For Sequences such that  $\Delta S_n = \sum_{i=1}^{\infty} a_i g_i(n)$ , in Numerical and Applied Mathematics, C. Brezinski (editor), J.C. Baltzer, pp.447–451 (1989).
- [B2] Acceleration Methods Based On Convergence Tests, *Numerische Mathematik* 58 (1990), 329–340.
- [B3] A Convergence Acceleration Method Based On a Good Estimation of the Absolut Value of the Error, *IMA J. of Numerical Analysis*, Vol. 10 (1990), 243–251.
- [B4] Extrapolation Algorithms Based on the Asymptotic Expansion of the Inverse of the Error. Application to Continued Fractions, *Journal of Computational and Applied Mathematics* 32 (1990) , 179–190 .
- [B5] Acceleration Results for the Vector E-Algorithm, *Numerical Algorithms* 1 (1991) , 237–260.
- [B6] The Least Squares Problem and Orthogonal Polynomials, (en collaboration avec Filomena d’Almeida et Maria João Rodrigues), in Orthogonal Polynomials and Their Applications. C. Brezinski, L.Gori and A. Ronveaux (editors). J.C. Baltzer AG, IMACS (1991), 217–222.
- [B7] Convergence and Acceleration Properties for the Vector  $\epsilon$  - Algorithm, *Numerical Algorithms* 3 (1992), 313–320.
- [B8] Breakdown and Near-Breakdown control in the CGS algorithm using stochastic arithmetic (en collaboration avec J-M Chesneaux) *Numerical Algorithms* 11 (1996), 99–116.