ROYAUME DU MAROC
Université Mohammed V - RABAT -
جامعة محمد الخامس – الرباط –
Faculté des sciences
كلية العلوم
CENTRE D'ETUDES DOCTORALES - SCIENCES ET TECHNOLOGIES

N° d'ordre 3681

THESE

En vue de l'obtention du : ***DOCTORAT***

Structure de Recherche : Laboratoire Mathématiques, Informatiques et Applications-Sécurité de l'Information .
Discipline : Mathématiques Appliquées.
Spécialité : Analyse Numérique.

Présentée et soutenue le 28/09/2022 par :

SIDKI Sara

Padé approximation and quasiorthogonality :

Recursive computation of orthogonal polynomials and quasiorthogonal polynomials

JURY

Abouir Jilali	PES, Faculté des sciences et technologies Mohammedia	Président/Rapporteur
Rhanizar Bouchta	PES, Ecole Normale Supérieure de Rabat.	Rapporteur/Examinateur
Esghir Mustapha	PH, Faculté des Sciences de Rabat.	Rapporteur/Examinateur
Errachid Mohammed	PH,Centre régional des métiers de l'éducation et de la formation de Rabat	Examinateur
Sadaka Rachid	PES, Ecole Normale Supérieure de Rabat	Directeur de thèse

Année universitaire : 2021/ 2022

Faculté des Sciences, avenue Ibn Battouta, BP. 1014 RP, Rabat –Maroc
☎ 00212 (05) 37 77 18 76 00212(05) 37 77 42 61; http://www. fsr.um5.ac.ma

Acknowledgements

First of all, I thank God for his gifts and mercy, offering me the will and the strength to go ahead and finish this thesis. I would like to thank all the people who accompanied me throughout the preparation period of this thesis. The latter was a new experience, full of testing concerning not only my training and my scientific knowledge but also my patience and especially my passion for mathematics.

I would like to deeply thank the rest of my thesis committee Abouir Jilali, Professor at Faculty of Sciences and technology in Mohammedia, Hassan II University. It is an honor for me that he chairs the examining board. I would like to thank them very much or her availability.

I would like to thank Rhanizar Bouchta , Professor at Ecole Normale Supérieure of Rabat, Mohammed V University, Esghir Mustapha Professor at Faculty of Sciences in Rabat, Mohammed V University, and Mohammed Errachid Professor at Centre régional des métiers de l'éducation et de la formation for agreeing to report my thesis, and being part of my jury, It's an honor.

I would like to express my gratitude to my supervisors, Professor Sadaka Rachid, professor at Ecole Normale Supérieure of Rabat (ENS), Mohammed V University and Professor Benazzouz abderrahim, professor at Ecole Normale Supérieure of Rabat (ENS), Mohammed V University for the confidence they had shown me.

After proposing the subject and accepting to direct my thesis, they had always been present during this period to guide me and support me in the difficult moments of this trip. Due to their listening, their knowledge, their advice and their presence, I was able to contribute in several axes of the theory which has enriched my works of research and my knowledge.

It is important to me to express my gratitude to my parents who have constantly supported me in my efforts and taught me to always try to do the best I can. No doubt that all this has contributed to my continued commitment to ever longer studies that this thesis finally concludes.

I am indebted to many of my friends and student colleagues for providing a stimulating and fun filled environment, and also for the discussion and the support. I would like to especially thank Yousra BOUIBRINE, Soukaina Sabir and Nabil Moukafih for all the fun time, for the support and for helping in revising some parts of this manuscript.

The order of my thanks does not matter. All those who I named brought me. at one time or another, decisive support.

Dédicaces

To my parents

To my sisters

To my friends...

Résumé

L'approximation de Padé permet de résoudre plusieurs problèmes dans des domaines différents, notamment dans l'analyse numérique l'approximation de Padé donne de bon résultats dans les problèmes d'accélération de la convergence. Le calcul de ces approximants a fait l'objectif de plusieurs travaux, dans cette thèse nous allons proposer des algorithmes de calcul des polynômes orthogonaux et des polynômes quasiorthogonaux afin de déterminer les approximants de Padé dans le cas scalaire et vectoriel. Pour y faire, nous allons commencer par rappeler l'approximation de Padé et les polynômes orthogonaux, nous introduisons la définition des polynômes quasiorthogonaux ayant une structure de matrice triangulaire inférieure. Nous allons rappeler la définition du complément de Schur et donner quelques propriétés qui seront utiles dans ce travail.

Dans le troisième chapitre, nous donnons une relation de récurrence entre deux familles de polynômes orthogonaux. Nous dérivons deux algorithmes récursifs pour calculer les polynômes orthogonaux en utilisant le qd algorithme et la relation de récurrence bien connue des déterminants de Hankel. Nous donnons également une relation de récurrence utile entre les polynômes orthogonaux de la même famille et les polynômes auxiliaires que nous allons définir, ce qui nous permet de construire un troisième algorithme pour calculer récursivement les polynômes orthogonaux.

Dans le quatrième chapitre, nous étendons le premier travail au cas vectoriel. La théorie de l'approximation vectorielle de Padé étudiée dans notre travail est liée à la théorie des polynômes quasi-orthogonaux.
Le calcul récursif des approximants vectoriels de Padé revient à calculer récursivement les polynômes quasiorthogonaux. Nous introduisons une procédure efficace pour calculer les familles adjacentes de polynômes quasi-orthogonaux. L'algorithme dérivé utilise des relations récursives dont les coefficients sont écrits comme rapport de deux déterminants de matrices triangulaires inférieurs par blocs. La stratégie de

calcul de tels déterminants est donnée et analysée.

Dans le dernier chapitre, nous allons appliquer les algorithmes proposés dans ce travail, sur un problème de réduction des modèles.

Abstract

The Padé approximation solves several problems in domains different, in particular in the numerical analysis the approximation of Padé gives good results in the problems of acceleration of convergence. The calculation of these approximants has been the objective of several works, in this thesis we will propose algorithms for calculating orthogonal polynomials and quasi-orthogonal polynomials in order to determine the Padé approximants in the scalar and vector case. To do this, we will start by recalling the Padé approximation and the orthogonal polynomials, we let us introduce the definition of quasiorthogonal polynomials having a lower triangular matrix structure. We will recall the definition of Schur's complement and give some properties that will be useful in this work.

In the third chapter, we give a recurrence relation between two families of orthogonal polynomials. We derive two recursive algorithms for computing the orthogonal polynomials by using the qd algorithm and the well known recurrence relation of Hankel determinants. We also give a useful recurrence relation between the orthogonal polynomials of the same family and auxiliary polynomials, which allows us to build a third algorithm for computing recursively the orthogonal polynomials.
In the fourth chapter, we extend the first work to the vector case. The theory of vector Padé approximation studied in our work is related to the theory of quasiorthogonal polynomials. The recursive calculation of vector Padé approximants is equivalent to calculate recursively the quasiorthogonal polynomials. introduces an efficient procedure to compute adjacent families of quasi-orthogonal polynomials. The derived algorithm uses short recursive relations whose coefficients are written in terms of lower triangular block determinants. The strategy of computing such determinants is given and analyzed.

In the last chapter, we will apply the algorithms proposed in this work, on a model reduction problem.

List of related publications

1. ***Computing recursive orthogonal polynomial with Schur complements. Journal of Computational and Applied Mathematics(2019).***

2. ***The recursive quasi-orthogonal polynomial algorithm. Journal of Numerical algorithms(2022).***

Table des matières

1 Introduction

1.1 Motivation

Let f be a function that is infinity differentiable at a complex or reel variable. We say that f admits a Taylor expansion of order n if it exists a polynomial P of degree n such that

$$f(x) - P(x) = o(x^n) \tag{1.1}$$

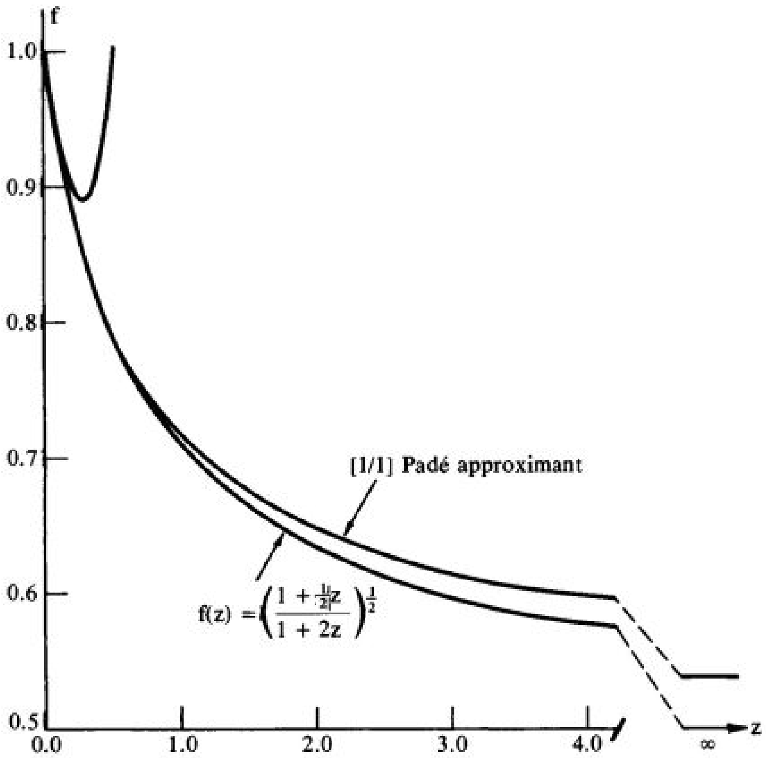
P is called Taylor polynomial of f with order n, the Taylor expansion gives a polynomial approximation of f , increasing the order of approximation we obtain a well approximation.

If we replace the polynomial P by a rational fraction $\dfrac{P}{Q}$, such that

$$f(x) - \frac{P(x)}{Q(x)} = o(x^n), \tag{1.2}$$

we obtain rational approximation or Padé approximation which more accurate than polynomial approximation.

Padé approximation consists of building a rational function so that its power series expansion matches the power series of the function to be approximated. These approximants are applied in physics, chemistry and mathematics , they are especially used in numerical analysis to accelerate the convergence of sequences and iterative algorithms. The theory of Padé approximation was generalized to the vector case and matrix case, approximating respectively series with vector coefficients and matrix coefficients. The computation of a sequence of Padé approximants was the subject of many works. We find recurrence schemes proposed to calculate the Padé approximants using the methods of acceleration of convergence, also recurrence relations were proposed to calculate recursively scalar and nonscalar Padé approximants using the theory of orthogonal polynomials thanks to their connection with denominators of Padé approximants. which makes the theory of orthogonal polynomials a main ingredient in the construction of Padé approximants.



1.2 History of Padé approximation

Before talking about Padé approximants, we must first introduce the notion of continued fractions. The theory of continued fractions began with Leonhard Euler(1707-1783) who showed that every rational number can be developed into a continued fraction and each irrational number gives rise to an infinite continued fraction.

In 1772, Joseph Louis Lagrange(1707-1783)[40] published a paper on the use of continued fractions in integral calculus where he developed a general method to obtain the continued fraction expansion of the solution of a differential equation. He then gave some examples and reduced the continued fractions thus obtained to ordinary fractions (by computing their convergents) whose power series expansions in ascending powers of the variable agree with those of the functions up to the term whose power is the sum of the degrees of the numerator and the denominator. This gives rise to Padé approximation theory.

In 1845, Carl Gustav Jacobi Jacob (1804,1851)[11] made many contributions to the theory of continued fractions, the well known of these being the writing of Padé approximants as a ratio of determinants.

In 1873, Charles Hermite (1822-1901) proved the transcendence of the number e using a new type of Padé approximants which is currently known by the name of Padé-Hermite. In 1882, using the same method as Hermite Carl Louis Ferdinand Lindeman (1852-1939) showed that the number π is transcendent.

In 1892, Henri Eugène Padé (1863-1953) proposed to arrange these approximants in a well known double-entry table called **Padé table**. Padé marked a great step in the study of these approximants, he approached the problem of convergence by formulating his famous conjecture and he even suggested certain generalizations of these approximants, such as for example the quadratic approximants to which we only have just recently gained prominence. He also showed that if the same rational function appears more than once in the table this function occupies a square block of cells. This result is known as : Block structure.

In 1894, continued fractions were used in the construction of an integral with respect to a general measure by Thomas Jan Stielthes (1856-1894)[52]. These results led Stielthes to lay down the basic notions of the theory of orthogonal polynomials .

For their part, mathematicians are interested by problems of convergence acceleration of sequences. In 1955, Shanks [26] proposed an interesting transformation of sequences and in 1956 Peter Wynn gave an algorithm for computing Shanks transforms recursively. This convergence acceleration process will be called "ε-algorithm" and Shanks will show that the numbers calculated by the ε-algorithm are none other than the values of Padé approximants at some points.

The problem that arises in the study of Padé approximants is that of their calculation. A direct computation of each approximant is expensive, it is more optimal to find recursively a sequence of approximants. There are recursive computation methods in the case of a normal table but what made possible to link the adjacent families of Padé approximants of Padé table is the systematic use of formal orthogonal polynomials. In 1980, Claude Brezinski [15]. For the case of a non-normal Padé table, we find the work of André Draux [2] who extended the recurrence relations between the elements of the Padé table to non-normal case.

Padé approximants have been exploited by physicists, mathematicians and chemists, in particular, in numerical analysis these approximants were used to accelerate the convergence of certain sequences and recursive algorithms.

Theory of Padé approximation was extended by mathematicians to the case of series with vector coefficients and series with matrix coefficients, among which we cite Jeannette's vector approximants [41] for the approximation of several functions by rational approximants with the same denominator. We also cite the Padé-Hermite approximation which consists in simultaneously approximating a vector function by a vector of polynomials.

The generalization of the Padé approximation was also treated by R.Sadaka [48] who introduced a new type of vector Padé approximants with a vector numerator of polynomials and a triangular matrix lower than the denominator. The interest of this work is in the use of vector Padé approximants for accelerating the convergence of vector sequences .

In 1990, Guo-liang Xu and Adhemar Bultheel [31], gave a general definition of matrix Padé approximants and introduced Padé approximants type I and type II, Depending on the normalization of the denominator.

In 1999, Guo-liang Xu and Adhemar Bultheel [32] provided a recursive procedure for computing the Matrix Padé approximants defined in their previous work.

1.3 Continued fractions

Euler was the first mathematician not only to give a clear exposition of continued fractions but also to use them extensively to solve various problems.

He also proposed to develop a function of variable into a continued fraction. The idea is to develop a function whose power series expansions in ascending power of the variable into a continued fraction. For the function

$$f(x) = \sum_{i=0}^{\infty} c_i x^i \tag{1.3}$$

then the continued fraction obtained by Euler

$$\frac{c_0 \mid}{\mid 1} - \frac{\frac{c_1}{c_0}x \mid}{\mid 1 + \frac{c_1}{c_0}x} - \frac{\frac{c_2}{c_1}x \mid}{\mid 1 + \frac{c_2}{c_1}x} - \frac{\frac{c_3}{c_2}x \mid}{\mid 1 + \frac{c_3}{c_2}x} \;\cdots\cdots$$

Lagrange proposed two formulas as solution of Euler's problem

1. $$a_0 + \frac{\alpha_0 \mid}{\mid 1} + \frac{\alpha_1 \mid}{\mid 1} + \frac{\alpha_2 \mid}{\mid 1} + \dots$$

   where α_i , $i \geqslant 1$, are monomials of the first degree, α_0 is a monomial or $\alpha_0 = 1$, and a_0 equal 1 or zero.

2.
$$a_0 + \frac{\alpha_0 \mid}{\mid a_1} + \frac{\alpha_1 \mid}{\mid a_2} + \frac{\alpha_2 \mid}{\mid a_3} + \dots$$
where $\alpha_i, i \geqslant 2$ are monomials of second degree and $a_i, i \geqslant 1$ are binomials of first degree.
a_0, a_1 are polynomials and α_0 is a monomial.

The theory of continued fractions arises from consideration of expressions given in the form
$$a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{... + \cfrac{1}{a_n \ddots}}}} \tag{1.4}$$
The quantities $a_0, a_1, \dots, a_n$ are called te partial quotients and may be taken to be integers , real or complex numbers , or functions of variables.
The expression (1.4) can be given as a rational fraction of the partial quotients, let's write
$$C_0 = a_0,$$
$$C_1 = a_0 + \frac{1}{a_1} = \frac{a_0a_1 + 1}{a_1},$$
$$C_2 = a_0 + \frac{1}{a_1 + \frac{1}{a_2}} = \frac{a_0(a_1a_2 + 1) + a_2}{a_1a_2 + 1}$$
and so on, the C_n are called the nth convergents of the continued fraction.
Consider a simple example, the fraction number $\frac{415}{93}$, let's find the continued fraction , we have $\frac{415}{93} = 4.4624...$, so we can write $\frac{415}{93} = 4 + \frac{43}{93}$, we take the reciprocal $\frac{93}{43}$ which approximately 2.1628 and $\frac{93}{43} = 2 + \frac{7}{43}$, and we have $\frac{43}{7} = 6.1429...$, then $\frac{43}{7} = 6 + \frac{1}{7}$ and $\frac{7}{1} = 7 + \frac{0}{1}$. Consequently, we have $a_0 = 4, a_1 = 2, a_2 = 6, a_3 = 7$ and we write
$$\frac{415}{93} = 4 + \cfrac{1}{2 + \cfrac{1}{6 + \frac{1}{7}}}$$
which we call the continued fraction of $\frac{415}{93}$.
Generally, a continued fraction is writing as follows
$$C = a_0 + \cfrac{b_1}{a_1 + \cfrac{b_2}{a_2 + \cfrac{b_3}{a_3 + \cfrac{b_4}{\dots + \cfrac{b_n}{a_n + \ddots}}}}}$$

and

$$C_n = a_0 + \frac{b_1|}{|a_1} + \ldots\ldots + \frac{b_n|}{|a_n}$$

where $n = 0, 1, 2, \ldots$, b_i and a_i are integers, real or complex numbers. If $b_i = 1$ we return to simple form (1.4). When the expression contains finitely many terms, it is called a finite continued fraction. When the expression contains infinitely many terms, it is called an infinite continued fraction.

Consider the function $f(x)$ in (1.3), and let

$$C(x) = c_0 + \frac{a_1 x|}{|1} + \frac{a_2 x|}{|1} + \ldots\ldots$$

If the coefficients a_i are chosen so that the convergent $C_k(x)$ has an expansion of power series identical to that of $f(x)$, so C_k is the approximant of $f(x)$, which is equivalent to

$$C_k(x) - f(x) = O(x^{v_k})$$

where v_k is highest possible. We can prove that C_k is a ratio of two polynomials, and consequently we can say that C_k is identical to the Padé approximant of $f(x)$.

Padé approximants are also closely related to some methods which are used in numerical analysis to accelerate the convergence of sequences and iterative processes. We present here the ε-algorithm.

1.4 ε-algorithm

In numerical analysis, many methods produce sequences, in particular methods for solving linear equations, but sometimes the convergence of these sequences is very slow, for this reason we use the convergence acceleration methods, these methods consist of transforming a sequence $(S_n)_{n\in\mathbb{N}}$ to a sequence $(T_n)_{n\in\mathbb{N}}$ such that they converge to the same limit but the convergence of $(T_n)_{n\in\mathbb{N}}$ is faster than the original sequence, the known methods of acceleration of convergence we find Δ^2-d'Aiteken , Shank's transformation and Richardson's method. The expression of Shank's transformation is giving by

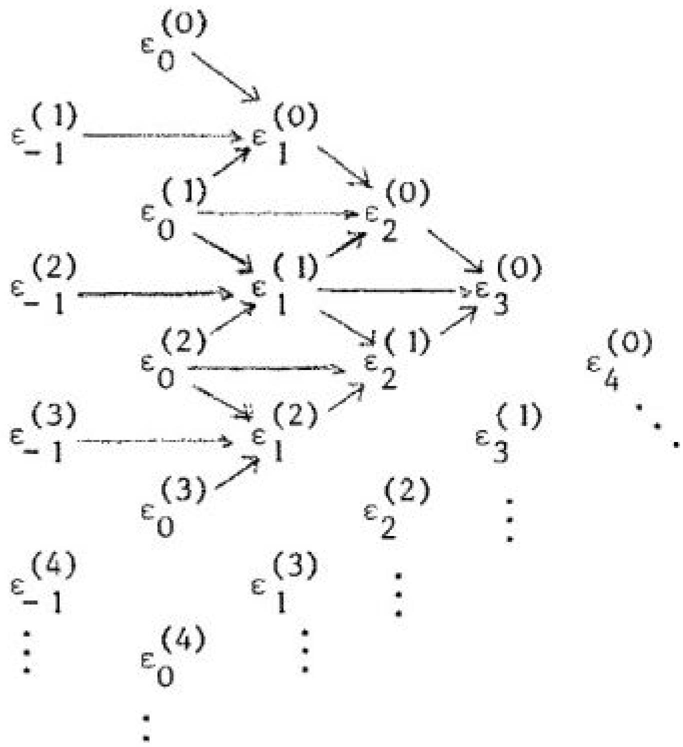
$$e_k(S_n) = \frac{\begin{vmatrix} S_n & \ldots & \ldots & S_{n+k} \\ \Delta S_n & \ldots & \ldots & \Delta S_{n+k} \\ \vdots & \vdots & \vdots & \vdots \\ \Delta S_{n+k-1} & \ldots & \ldots & \Delta S_{n+2k-1} \end{vmatrix}}{\begin{vmatrix} 1 & 1 & \ldots & 1 \\ \Delta S_n & \ldots & \ldots & \Delta S_{n+k} \\ \vdots & \vdots & \vdots & \vdots \\ \Delta S_{n+k-1} & \ldots & \ldots & \Delta S_{n+2k-1} \end{vmatrix}}, \tag{1.5}$$

where $\Delta S_n = S_{n+1} - S_n$

P. Wynn gives a recursive process to compute recursively such ratio of determinants denoted ε-algorithm, this algorithm is given as follows

$$\begin{cases} \varepsilon_{-1}^{(n)} = 1, \varepsilon_0^{(n)} = S_n, \forall n = 0, 1, \dots \\ \varepsilon_{k+1}^{(n)} = \varepsilon_{k-1}^{(n+1)} + \dfrac{1}{\varepsilon_k^{(n+1)} - \varepsilon_k^{(n)}} \\ \varepsilon_{2k}^{(n)} = e_k(S_n), \forall n, k = 0, 1, \dots. \end{cases} \tag{1.6}$$

The quantities $\varepsilon_k^{(n)}$ are placed in the following table



Applying the Shank's transformation to the sequence $S_n = \sum_{i=0}^{n} c_i x^i$, we can prove that $e_k(S_n)$ is the Padé approximant of f, with numerator of degree $n+k$ and denominator of degree k. Using the relations of ε-algorithm, then we can give a recursive algorithm for computing the elements of Padé table.

The construction of Padé approximants is also possible by using the theory of orthogonal polynomials, the relation existing between orthogonal polynomials and the denominator of Padé approximants allows us to give a recursive procedure to compute the Padé approximants, this will be the subject of our thesis.

We will present new recursive algorithms for building a sequence of orthogonal polynomials and adjacent systems of orthogonal polynomials, also we will extend our work to vector Padé approximation defined by R.Sadaka[48], which are related to quasi-orthogonal polynomials.
We will give a new expression of orthogonal polynomials, this expression will allows us to find a three-terms recurrence relation using th Schur complements and their properties, the qd-algorithm will be used to obtain a recursive algorithm for computing the quantities denoted $\beta_k^{(i)}$ appearing in the recurrence relation. Using an auxiliary polynomials and orthogonal polynomials, we will propose another algorithm for computing orthogonal polynomial of the same family. An extension of this work will be treated, we will introduce the definition of quasiorthogonal polynomials, give an expression of these polynomials as a Schur complement, then we will obtain a four-term recurrence relation of the quasiorthogonal polynomials. The derived algorithm uses short recursive relations whose coefficients are written in terms of lower triangular block determinants. The strategy of computing such determinants is given and analyzed.
This thesis is organized as follows : In the second chapter, we will present the essential results used throughout this work starting with the definition of scalar and vector approximants and passing to the introduction of the properties of Schur's complements by showing the link with Padé approximants. In the third chapter, the Schur complements, the Sylvester identity and qd algorithm will be used to derive a recursive algorithm for computing the orthogonal polynomials without referring to the relations of orthogonality. We will recover the recurrence relation between the Hankel determinants when we apply the Sylvester identity to the expression of Hankel determinant. This relation will help us to give another recursive algorithm to calculate the orthogonal polynomials. We will also give an algorithm to calculate the orthogonal polynomials of the same family by using the Sylvester and auxiliary polynomials. In the fourth chapter, we will present a recursive algorithm to compute adjacent families of quasi-orthogonal polynomials of the form (4.3). New short-term recurrence relations involving multiple families of such polynomials with lower block determinants are introduced. An illustrative example with applications to vector Padé approximations will be given.The fifth chapter is an application of the algorithms proposed in this thesis to a single-machine infinite-bus(SMIB) power system.

2 Preliminary

This chapter is devoted to recall the definitions and proprieties of the notions which will be used in this thesis. We first recall the Padé approximation of scalar function, we give the expression of a Padé appproximant and we present the Padé table, an other notion is the Padé-type approximants, in these approximants it is possible to choose some of the poles and then to define the denominator and the numerator so that the expansion of the approximant matches the series to be approximated as far as possible. We recall vector Padé approximants introduced by R.Sadaka [48] and a set of Padé tables are proposed as a extension to the scalar case. In the last section the definition of Schur complements and some their properties are recalled and we show the link with Padé approximation.

2.1 Scalar Padé approximation

Let us consider a scalar function f expressed as a formal power series

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

The goal of Padé approximation is to find a rational function $\dfrac{P(x)}{Q(x)}$, with degree of $P \leqslant p$ and degree of $Q \leqslant q$ such that its power series expansion agree the power series expansion of the function to be approximated as far as possible. Let

$$\frac{P(x)}{Q(x)} = \frac{a_0 + a_1 x + \ldots + a_p x^p}{b_0 + b_1 x + \ldots + b_q x^q} \tag{2.1}$$

verifying the following relation

$$Q(x)f(x) - P(x) = O(x^{p+q+1})$$

which is equivalent to the following system

$$\begin{cases} c_0 b_0 = a_0 \\ c_0 b_1 + c_1 b_0 = a_1 \\ c_0 b_2 + c_1 b_1 + c_2 b_0 = a_2 \\ \vdots \\ c_{p-q} b_q + c_{p-q+1} b_{q-1} + \ldots + c_p b_0 = a_p \\ c_{p+1} b_0 + c_p b_1 + \ldots + c_{p-q+1} b_q = 0 \\ \vdots \\ c_{p+q} b_0 + c_{p+q-1} b_1 + \ldots + c_p b_q = 0 \end{cases}$$

with the convention $c_i = 0, \forall i < 0$ and we put $b_0 = 1$. So we have two systems with solutions are the coefficients $a_0, a_1, \ldots, a_p$ and $b_1, b_2, \ldots, b_q$, we obtain

$$\begin{bmatrix} c_{p-q+1} & c_{p-q+2} & \cdots & \cdots & c_p \\ c_{p-q+2} & c_{p-q+3} & \cdots & \cdots & c_{p+1} \\ \vdots & \vdots & \cdots & \cdots & \vdots \\ c_p & c_{p+1} & \cdots & \cdots & c_{p+q-1} \end{bmatrix} \begin{bmatrix} b_q \\ b_{q-1} \\ \vdots \\ b_1 \end{bmatrix} = - \begin{bmatrix} c_{p+1} \\ c_{p+2} \\ \vdots \\ c_{p+q} \end{bmatrix} \tag{2.2}$$

and

$$\begin{bmatrix} c_0 & 0 & \ldots & & & 0 \\ c_1 & c_0 & 0 & \ldots & & 0 \\ c_2 & c_1 & c_0 & 0 & \ldots & 0 \\ \vdots & & & & & \vdots \\ c_p & \ldots & & & \ldots & c_{p-q} \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_q \end{bmatrix} = \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_p \end{bmatrix} \tag{2.3}$$

we remark that coefficients of denominator are calculating from system (2.2), and we replace the them in the system (2.3), so the problem of Padé approximation is based on finding the coefficients of the denominator $Q(x)$. The condition of existence and uniqueness of Padé approximant is that matrix $H_q^{(p-q+1)}$ is of maximal rank

$$H_q^{(p-q+1)} = \begin{vmatrix} c_{p-q+1} & c_{p-q+2} & \cdots & \cdots & c_p \\ c_{p-q+2} & c_{p-q+3} & \cdots & \cdots & c_{p+1} \\ \vdots & \vdots & \cdots & \cdots & \vdots \\ c_p & c_{p+1} & \cdots & \cdots & c_{p+q-1} \end{vmatrix} \tag{2.4}$$

and the explicit expression of Padé approximant is given by

$$\frac{P(x)}{Q(x)} = \frac{\begin{vmatrix} c_{p-q+1} & c_{p-q+2} & \cdots & c_p & c_{p+1} \\ c_{p-q+2} & c_{p-q+3} & \cdots & c_{p+1} & c_{p+2} \\ \vdots & & & & \vdots \\ c_p & c_{p+1} & \cdots & c_{p+q-1} & c_{p+q} \\ x^q & x^{q-1} & \cdots & x & 1 \end{vmatrix}}{\begin{vmatrix} c_{p-q+1} & c_{p-q+2} & \cdots & c_{p+1} \\ c_{p-q+2} & c_{p-q+3} & \cdots & c_{p+2} \\ \vdots & & & \vdots \\ \sum_{i=0}^{p-q} c_i x^{q+i} & \sum_{i=0}^{p-q+1} c_i x^{q+i-1} & \cdots & \sum_{i=0}^{p} c_i x^i \end{vmatrix}} \tag{2.5}$$

These Padé approximants are denoted by $[p/q]$, Padé proposed to place them in a double array table called Padé table. The Padé approximants $[p/p]$ are placed in the diagonal of the table,

p/q	0	1	2	3	$\ldots$
0	$[0/0]$	$[1/0]$	$[2/0]$	$[3/0]$	$\ldots$
1	$[0/1]$	$[1/1]$	$[2/1]$	$[3/1]$	$\ldots$
2	$[0/2]$	$[1/2]$	$[2/2]$	$[3/2]$	$\ldots$
3	$[0/3]$	$[1/3]$	$[2/3]$	$[3/3]$	$\ldots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$

TABLE 1 – Padé Table

For example, the Padé approximant [2/2] of the function $f(x) = sin(x) + 1$ is

$$[2/2]_f(x) = \frac{1 + x + \frac{1}{6}x^2}{1 + \frac{1}{6}x^2},$$

and graphically, we have

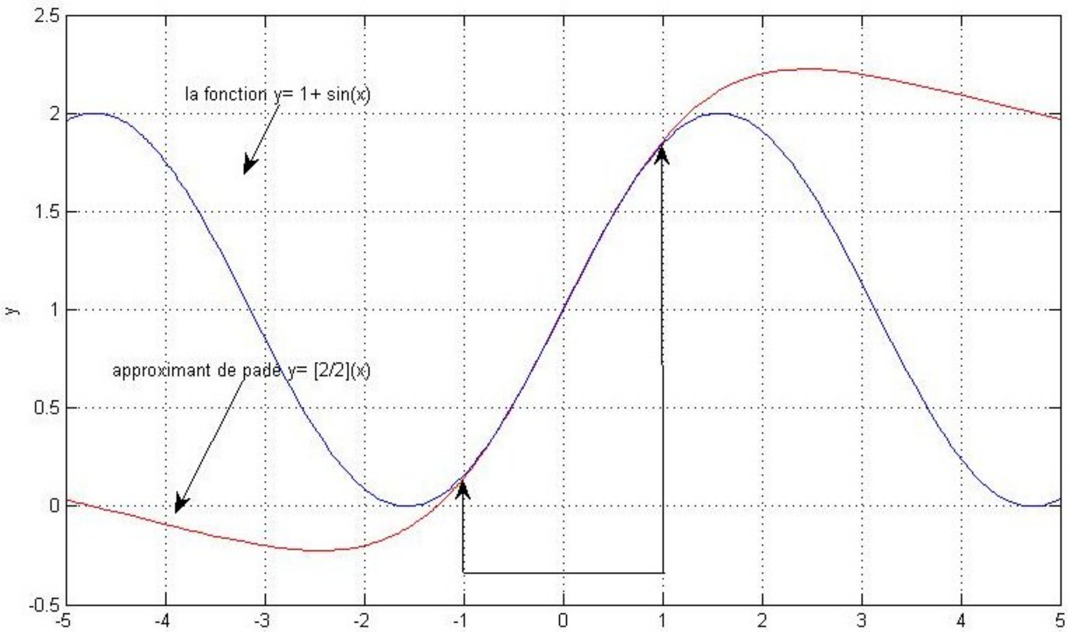


FIGURE 1 – Padé approximant [2/2] and f(x)=sin(x)+1

In general , the Padé approximants in diagonal $[p/p]$ and the approximants $[p+1/p]$ give best acceleration of convergence of the power series f.
Padé table is called normal if we have

$$H_q^{p-q+1} \neq 0, \forall p, q \geqslant 0,$$

otherwise, the Padé table is called non normal. The recursive computation of the elements of the normal and non normal Padé table was the subject of many works. One the methods used is the ε-algorithm , let consider the sequence $\{S_n\}$

$$S_n = \sum_{i=0}^{n} c_i x^i,$$

with $S_0 = c_0$, $\Delta S_i = c_{i+1} x^{i+1}$,
applying Shank's transformation in (1.5)

$$e_k(S_n)) = \frac{\begin{vmatrix} \sum_{i=0}^{n} c_i x^i & \sum_{i=0}^{n+1} c_i x^i & \cdots & \sum_{i=0}^{n+k} c_i x^i \\ x^{n+1} c_{n+1} & x^{n+2} c_{n+2} & \cdots & x^{n+k+1} c_{n+k+1} \\ \vdots & \vdots & \vdots & \vdots \\ x^{n+k} c_{n+k} & x^{n+k+1} c_{n+k+1} & \cdots & x^{n+2k} c_{n+2k} \end{vmatrix}}{\begin{vmatrix} 1 & \cdots & \cdots & 1 \\ x^{n+1} c_{n+1} & x^{n+2} c_{n+2} & \cdots & x^{n+k+1} c_{n+k+1} \\ \vdots & \vdots & \vdots & \vdots \\ x^{n+k} c_{n+k} & x^{n+k+1} c_{n+k+1} & \cdots & x^{n+2k} c_{n+2k} \end{vmatrix}}$$

We multiply first column of denominator and numerator by x^k , the second by x^{k-1}

$$e_k(S_n)) = \frac{\begin{vmatrix} \sum_{i=0}^{n} c_i x^{i+k} & \sum_{i=0}^{n+1} c_i x^{i+k-1} & \cdots & \sum_{i=0}^{n+k} c_i x^i \\ x^{n+k+1} c_{n+1} & x^{n+k+1} c_{n+2} & \cdots & x^{n+k+1} c_{n+k+1} \\ \vdots & \vdots & \vdots & \vdots \\ x^{n+2k} c_{n+k} & x^{n+k+1} c_{n+k+1} & \cdots & x^{n+2k} c_{n+2k} \end{vmatrix}}{\begin{vmatrix} x^k & \cdots & \cdots & 1 \\ x^{n+k+1} c_{n+1} & x^{n+k+1} c_{n+2} & \cdots & x^{n+k+1} c_{n+k+1} \\ \vdots & \vdots & \vdots & \vdots \\ x^{n+2k} c_{n+k} & x^{n+2k} c_{n+k+1} & \cdots & x^{n+2k} c_{n+2k} \end{vmatrix}}$$

finally , we divide the second rows of denominator and numerator by x^{n+k+1}, the third by x^{n+k+2}

$$e_k(S_n)) = \frac{\begin{vmatrix} \sum_{i=0}^{n} c_i x^{i+k} & \sum_{i=0}^{n+1} c_i x^{i+k-1} & \dots & \sum_{i=0}^{n+k} c_i x^i \\ c_{n+1} & c_{n+2} & \dots & c_{n+k+1} \\ \vdots & \vdots & \vdots & \vdots \\ c_{n+k} & c_{n+k+1} & \dots & c_{n+2k} \end{vmatrix}}{\begin{vmatrix} x^k & \dots & \dots & 1 \\ c_{n+1} & c_{n+2} & \dots & c_{n+k+1} \\ \vdots & \vdots & \vdots & \vdots \\ c_{n+k} & c_{n+k+1} & \dots & c_{n+2k} \end{vmatrix}}$$

Consequently , we obtain the expression of the Padé approximant $[n+k/k]$.

$$e_k(S_n) = \varepsilon_{2k}^{(n)} = [n+k/k]$$

and we apply the relations (1.6), the elements of the Padé table are computed, for more details see [15] .

In the next section, we present another way to find the Padé approximants using Padé-type approximation thanks to a wise choice of the of denominator of the Padé approximant.

2.2 Padé-type approximation

Let's consider the formal power series $f(x)$ in (1.3). The goal of this approximation is to construct a rational function having a denominator of degree k and a numerator of degree $k-1$.

Let's start with the definition of a functional c which acts on the real space of the polynomials $\mathcal{P}$:

$$c(x^i) = c_i, \quad \text{pour tout } i = 0, 1, \dots$$

We define the generating function of f :

$$G(x,t) = \frac{1}{1-xt},$$

we can also see that :

$$f(t) = c(G(x,t)),$$

indeed, we expand $\dfrac{1}{1-xt}$ into a power series, then we apply the functional c we get :

$$\begin{aligned} c((1-xt)^{-1}) &= c(1+xt+x^2t^2+\ldots) \\ &= c(1)+c(x)t+c(x^2)t^2+\ldots \\ &= c_0+c_1t+c_2t^2+\ldots = f(t). \end{aligned}$$

Let v be an arbitrary polynomial of degree k belonging to $\mathcal{P}$:

$$v(x) = \sum_{i=0}^{k} b_i x^i,$$

and define w by :

$$w(t) = c(\frac{v(x)-v(t)}{x-t}),$$

this polynomial is of degree $k-1$ and it is written :

$$w(t) = \sum_{i=0}^{k-1} \left(\sum_{j=0}^{k-1-i} c_j b_{j+i+1} \right) t^i. \tag{2.6}$$

Indeed, we write :

$$\frac{(x^i-t^i)}{(x-t)} = x^{i-1}+x^{i-2}t+x^{i-3}t^2+\ldots+xt^{i-2}+t^{i-1}$$

and applying c we get the expression (2.6).

Then, let $\tilde{v}$ and $\tilde{w}$ be the transformed polynomials of v and w

$$\tilde{v}(t) = t^k v(t^{-1}),$$

$$\tilde{w}(t) = t^{k-1} w(t^{-1}).$$

We keep the same numbering of the coefficients of v and w to express $\tilde{v}$ and $\tilde{w}$:

$$\tilde{v}(t) = b_0t^k + b_1t^{k-1} + \ldots + b_{k-1}t + b_k,$$

$$\tilde{w}(t) = a_0t^{k-1} + a_1t^{k-2} + \ldots + a_{k-2}t + a_{k-1},$$

with $a_i = \sum\limits_{j=0}^{k-1-i} c_j b_{j+i+1}$ Finally, the main result is the following :

Théorème 1 *: [16]*

$$\frac{\tilde{w}(t)}{\tilde{v}(t)} - f(t) = O(t^k) \tag{2.7}$$

The approximant $\dfrac{\tilde{w}(t)}{\tilde{v}(t)}$ will be called of Padé type denoted by $[k-1/k]_f$.

2.3 Vector Padé approximants

Various generalizations of scalar Padé approximants have been proposed and analyzed to develop Padé approximation for the non-scalar functions.

For the first time Hermite decided to approximate simultaneously a sequence of functions $e^{\lambda_i x}$ using the rational approximation.

De Bruin introduces the simultaneous Padé approximation, his idea is to approximate a sequence of functions f_i with a rational function P_i/Q.

Vector Padé approximations are based on the approximation of a given vector function $F = (f_1, \ldots, f_n)$ from $\mathbb{R}$ to $\mathbb{R}^n$ by a pairwise (P, Q) so that the order of approximation of the expression $QF - P$ is high as possible. The pair (P, Q) is said to be a Padé approximant of F, where P is a vector of polynomials, and Q is a matrix of polynomials

$$P = \begin{pmatrix} P_1 \\ P_2 \\ \vdots \\ P_k \end{pmatrix}, \textit{with } \deg\,(P_i) = p_i, \quad i = 1, \ldots, k,$$

and

$$Q = \begin{pmatrix} Q_{11} & Q_{12} & \ldots & Q_{1n} \\ Q_{21} & Q_{22} & \ldots & Q_{2n} \\ \vdots & \vdots & \ldots & \vdots \\ Q_{k1} & Q_{k2} & \ldots & Q_{kn} \end{pmatrix}$$

with $deg(Q_{ij}) = q_{ij}, \quad i = 1, \ldots, n,$ and $\quad j = 1, \ldots, k.$

The choice of the form of the matrix Q leads to different formulations of the vector Padé approximation.
For instance, when $n = k = 1$, the approximation (P, Q) simplifies to the scalar Padé approximation [34].
For $k = 1$ and $n \in \mathbb{N}$, (P, Q) gives the Padé-Hermite approximation [35].
The situation when Q is a diagonal matrix that contains the same polynomial on the diagonal and the polynomials P_i have the same degree is provided by Van Iseghem [41].
Sadaka [48] explored the case when the vector Padé approximants $(P, Q_{(m,n)})$ have the form

$$Q_{m,n} = \begin{bmatrix} Q_{(m,n)_1} & 0 \\ Q_{(m,n)_2} & Q_{(m,n)_1} \end{bmatrix}, \quad P = \begin{bmatrix} P_1 \\ P_2 \end{bmatrix}, \tag{2.8}$$

where $deg(Q_{(m,n)_1}) = m$, $deg(Q_{(m,n)_2}) = n$, and $deg(P_1) = deg(P_2) = l$, with $m-n = 2p+1$, for an integer p [48].
The function F is defined from $\mathbb{R}$ to $\mathbb{R}^2$ as

$$F(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \end{bmatrix},$$

where the components f_1 and f_2 have the following expansions

$$f_1(x) = \sum_{i=0}^{\infty} c_i^1 x^i, \quad f_2(x) = \sum_{i=0}^{\infty} c_i^2 x^i,$$

with c_i^1 and $c_i^2 \in \mathbb{R}$.
The function F can be approximated by the vector Padé approximant $(P, Q_{m,n})$, such that the order of $Q_{m,n}F - P$ is $(r+l+1)$, with $r = \frac{m+n+1}{2}$, i.e.,

$$\begin{cases} Q_{(m,n)_1} f_1 - P_1 = O(x^{r+l+1}) \\ Q_{(m,n)_1} f_2 + Q_{(m,n)_2} f_2 - P_2 = O(x^{r+l+1}). \end{cases} \tag{2.9}$$

for $l = m-1$, the system (2.9) is equivalent to

$$\begin{cases} c_m^1 q_0^1 + c_{m-1}^1 q_1^1 + \ldots + c_0^1 q_m^1 = 0 \\ c_m^2 q_0^1 + c_{m-1}^2 q_1^1 + \ldots + c_0^2 q_m^1 + c_m^1 q_0^2 + \ldots + c_{m-n}^1 q_n^2 = 0 \\ c_{m+1}^1 q_0^1 + c_m^1 q_1^1 + \ldots + c_1^1 q_m^1 = 0 \\ c_{m+1}^2 q_0^1 + c_m^2 q_1^1 + \ldots + c_1^2 q_m^1 + c_{m+1}^1 q_0^2 + \ldots + c_{m-n+1}^1 q_n^2 = 0 \\ \vdots \\ c_{m+r-1}^1 q_0^1 + c_{m+r-2}^1 q_1^1 + \ldots + c_{r-1}^1 q_m^1 = 0 \\ c_{m+r-1}^2 q_0^1 + c_{m+r-2}^2 q_1^1 + \ldots + c_{r-1}^2 q_m^1 + c_{m+r-1}^1 q_0^2 + c_{m+r}^1 q_1^2 + \ldots + c_{m-n+r-1}^1 q_n^2 = 0 \end{cases}$$

with $q_0^1 = 1$, the necessary and sufficient condition for the existence and uniqueness of $Q_{m,n}$ and P such that the system (2.9) holds is $\Delta_{m,n} \neq 0$ where

$$\Delta_{m,n} = \begin{vmatrix} c_0^1 & c_1^1 & \ldots & c_{m-1}^1 & 0 & \ldots & \ldots & 0 \\ c_1^1 & c_2^1 & \ldots & c_m^1 & 0 & \ldots & \ldots & 0 \\ \vdots & \vdots & & \vdots & \vdots & & & \vdots \\ c_{r-1}^1 & c_r^1 & \ldots & c_{m+r-2}^1 & 0 & \ldots & \ldots & 0 \\ c_0^2 & c_1^2 & \ldots & c_{m-1}^2 & c_m^1 & \ldots & \ldots & c_{m-n}^1 \\ c_1^2 & c_2^2 & \ldots & c_m^2 & c_{m+1}^1 & \ldots & \ldots & c_{m-n+1}^1 \\ \vdots & & & \vdots\vdots & & & & \vdots \\ c_{r-1}^2 & c_r^2 & \ldots & c_{m+r-2}^2 & c_{m+r-1}^1 & \ldots & \ldots & c_{m+r-n-1}^1 \end{vmatrix}.$$

Under this condition, the polynomials $Q_{(m,n)_1}$ and $Q_{(m,n)_2}$ are expressed as follows

$$Q_{(m,n)_1}(x) = \frac{1}{\Delta_{m,n}} \begin{vmatrix} 1 & x & x^2 & \dots & x^m & 0 & \dots & 0 \\ c^1_m & c^1_{m-1} & c^1_{m-2} & \dots & c^1_0 & 0 & \dots & 0 \\ c^1_{m+1} & c^1_m & c^1_{m-1} & \dots & c^1_1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ c^1_{r+m-1} & c^1_{r+m-2} & c^1_{r+m-2} & \dots & c^1_{r-1} & 0 & \dots & 0 \\ c^2_m & c^2_{m-1} & c^2_{m-2} & \dots & c^2_0 & c^1_m & \dots & c^1_{m-n} \\ c^2_{m+1} & c^2_m & c^2_{m-1} & \dots & c^2_1 & c^1_{m-1} & \dots & c^1_{m-n+1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\ c^1_{r+m-1} & c^2_{r+m-2} & c^2_{m+r-2} & \dots & c^2_{r-1} & c^1_{m+r-1} & \dots & c^1_{m-n+r-1} \end{vmatrix}$$

and

$$Q_{(m,n)_2}(x) = \frac{1}{\Delta_{m,n}} \begin{vmatrix} 0 & 0 & 0 & \dots & 0 & 1 & \dots & x^n \\ c^1_m & c^1_{m-1} & c^1_{m-2} & \dots & c^1_0 & 0 & \dots & 0 \\ c^1_{m+1} & c^1_m & c^1_{m-1} & \dots & c^1_1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ c^1_{r+m-1} & c^1_{r+m-2} & c^1_{r+m-2} & \dots & c^1_{r-1} & 0 & \dots & 0 \\ c^2_m & c^2_{m-1} & c^2_{m-2} & \dots & c^2_0 & c^1_m & \dots & c^1_{m-n} \\ c^2_{m+1} & c^2_m & c^2_{m-1} & \dots & c^2_1 & c^1_{m-1} & \dots & c^1_{m-n+1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\ c^1_{r+m-1} & c^2_{r+m-2} & c^2_{m+r-2} & \dots & c^2_{r-1} & c^1_{m+r-1} & \dots & c^1_{m-n+r-1} \end{vmatrix}$$

By analogy of scalar case Sadaka [48] proposed to organize these vector Padé approximants in a table, he denotes the vector Padé approximants by $[l/n]$ for each fixed integer p , using the relation $m - n = 2p + 1$, then l represents the degree of P_1, P_2, n the degree of $Q_{(m,n)_1}$ and $n + 2p + 1$ degree of $Q_{(m,n)_2}$,

T_p					
	[0/0]	[1/0]	[2/0]	[3/0]	[4/0]
	[0/1]	[1/1]	[2/1]	[3/1]	...
	[0/2]	[1/2]	[2/2]	...	...
	[0/3]	[1/3]	[2/3]	...	...
	[0/4]	[1/4]	[2/4]	...	.....
	$\vdots$	$\vdots$	$\vdots$		

for each integer p, we can construct a table T_p which gives a family of Padé approximants of the vector function F.

2.4 Orthogonal polynomials

In our thesis we will give recursive algorithms for computing orthogonal polynomials and quasiorthogonal polyomials in order to exploit the relation between these polynomials and Padé approximants in scalar and vector case.

Orthogonal polynomials are special case of orthogonal functions, say that two functions are orthogonal on an interval [a,b] if :

$$\int_a^b f_i(x)f_j(x)dx = 0, i \neq j$$

In particular, a family of orthogonal polynomials is a sequence of polynomials $q_0, q_1, \ldots\ldots,$ with $deg(q_i) = n_i$ and for all $i, j = 0, 1, \ldots$ we have

$$\int_a^b q_i(x)q_j(x)dx = 0, \quad i \neq j$$

In our work we use the following definition of orthogonal polynomials :

Définition 1 *[15] Let $\mathcal{P}$ be the vector space of real polynomials and let $\{c_k\}_{k\in\mathbb{N}}$ be a given infinite sequence of real numbers. We define a linear functional c on $\mathcal{P}$ by*

$$\begin{array}{rcl} c & : & \mathcal{P} \quad \rightarrow \mathbb{R} \\ & & x^i \quad \mapsto \quad c_i \qquad\qquad i \geqslant 0. \end{array}$$

The family of polynomials $\{Q_k\}$ such that each Q_k satisfies

$$\begin{cases} deg(Q_k) = k \\ c(x^iQ_k(x)) = 0 & 0 \leqslant i \leqslant k-1, \\ c(x^kQ_k(x)) \neq 0 \end{cases}$$

$\{Q_k\}$ is called a family of orthogonal polynomials with respect to c (or with respect to the sequence $\{c_k\}$).

The following theorem gives the recurrence relation between the orthogonal polynomials.

Théorème 2 *[15] The family $\{Q_k\}$ of orthogonal polynomials satisfies the following three-term recurrence relationship :*

$$Q_{k+1}(x) = (A_{k+1}x + B_{k+1})Q_k(x) - C_{k+1}Q_{k-1}(x),$$

with $Q_{-1}(x) = 0$ and $Q_0(x)$ is an arbitrary non zero constant. The constants are given by

$$A_{k+1} = \frac{h_{k+1}}{\beta_k}, B_{k+1} = -h_{k+1}\frac{\alpha_k}{(h_k\beta_k)} \text{ and } C_{k+1} = \frac{\beta_{k-1}h_{k+1}}{\beta_k h_{k-1}},$$

with $\alpha_k = c(xQ_k^2(x))$, $\beta_k = c(xQ_k(x)Q_{k+1}(x))$ and $h_k = c(Q_k^2(x))$.

We can define the orthogonal polynomials with respect to the linear functional $c^{(i)}$, where $c^{(i)}$ is defined by

$$c^{(i)}(x^j) = c_{i+j}, \quad \forall j \in \mathbb{N}.$$

Définition 2 *[4] The family of polynomials $\{Q_k^{(i)}\}$, such that each $Q_k^{(i)}$ satisfies*

$$\begin{cases} deg(Q_k) = k \\ c^{(i)}(x^j Q_k(x)) = 0 & \quad 0 \leqslant j \leqslant k-1 \\ c^{(i)}(x^K Q_k(x)) \neq 0 \end{cases} \tag{2.10}$$

is called the adjacent system of orthogonal polynomials.

The existence and uniqueness of $Q_k^{(i)}$ is given by the following proposition :

Proposition 1 *[15] For $i \geqslant 0$ and $k \geqslant 0$ the solution of* (2.10)*, $Q_k^{(i)}$, exists and is unique if and only if the Hankel determinant $H_k^{(i)} \neq 0$, for $k \geqslant 1$*

$$H_k^{(i)} = \begin{vmatrix} c_i & c_{i+1} & \dots & \dots & c_{i+k-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ c_{i+k-1} & c_{i+k} & \dots & \dots & c_{i+2k-2} \end{vmatrix}.$$

Remarque 3 *The orthogonal polynomials and the adjacent orthogonal polynomials can be placed in a two dimensional array as follows :*

$$\begin{array}{ccccccc} Q_0 & Q_0^{(1)} & Q_0^{(2)} & \dots & \dots & Q_0^{(i)} & \dots \\ & Q_1 & Q_1^{(1)} & \dots & \dots & Q_1^{(i-1)} & \dots \\ & & Q_2 & \ddots & \dots & & \\ & & & \ddots & \ddots & \dots & \\ & & & & \ddots & Q_{k-1}^{(1)} & \dots \\ & & & & & Q_k & \ddots \\ & & & & & & \ddots \end{array} \tag{2.11}$$

2.5 Quasi-orthogonal polynomials

Theory of quasiorthogonal polynomials was introduced by Riesz [46] for the case of order 1, with Chihara [53] quasiorthogonal polynomials are extended to order $r \geqslant 1$, he considered the polynomials

$$q_n = \sum_{j=0}^{k} a_{n,j} p_{n-jr}(x), \tag{2.12}$$

where $k \geqslant 0, r \geqslant 1$ are fixed integers and $a_{n,j}$ constants, define $p_{-m} = 0, m = 1, 2, \ldots, kr$.
Satisfying the relation

$$\int_a^b q_m(x)q_n(x)d\alpha(x) = 0, \quad \text{for} m \neq n \pm jr \quad (j = 0, 1, \ldots, k) \tag{2.13}$$

These polynomials are called quasiorthogonal polynomials of index (k,r), which are an extension of quasiorthogonal polynomials of index (1,1) defined by Riesz[46] of the form

$$Ap_n(x) + Bp_{n+1}(x)$$

Quasiorthogonal polynomials are defined respect to a quasi-definite moment functional c

Définition 3 *[54] The moment functional c ia quasi-definite if and only if $H_n \neq 0, \forall n \geqslant 1$.*

Définition 4 *[53] A monic polynomial q_n of degree $n \geqslant r > 0$ is quasiorthogonal of order r with respect to c if and only if*

$$c(x^j q_n(x)) = 0, \quad j = 0, \ldots, n-r-1. \tag{2.14}$$

It is quasiorthogonal with respect to c , strictly of order r, if and only if

$$\begin{cases} c(x^j q_n(x)) = 0, & j = 0, \ldots, n-r-1, \\ c(x^j q_n(x)) \neq 0, & j = n-r. \end{cases} \tag{2.15}$$

Maroni [47] gave definition of sequence of quasi orthogonal polynomials

Définition 5 *[47] A sequence $\{q_n\}_{n\geqslant 0}$ is a sequence of quasiorthogonal polynomials of order r if and only if*

- *every polynomial q_n, for $n \geqslant r$, satisfies* (2.14)*,*
- *there exists at least $l \in \mathbb{N}, l \geqslant r$, such that $c(x^{l-r}q_l(x)) \neq 0$,*
- *if $n < r$, q_n is subject to no condition.*

A sequence $\{q_n\}_{n\geqslant 0}$ is sequence of quasiorthogonal polynomials strictly of order r if and only if any polynomial q_n , for $n \geqslant r$, satisfies (2.15)*.*

Chihara [53] proved that quasiorthogonal polynomials of order r verify a three term recurrence relation

Théorème 4 *[53] Let $\{q_n\}_{n\geqslant 0}$ be a sequence of quasiorthogonal polynomials of order r with respect to a quasi-definite moment functional c. These polynomials satisfy the following three term recurrence relation*

$$E_n(x)q_n(x) = F_n(x)q_{n-1}(x) + G_n(x)q_{n-2}(x), \tag{2.16}$$

where $E_n(x)$ and $F_n(x)$ are monic polynomials of degree at most equal to r and $r+1$ respectively. $G_n(x)$ is a polynomial of degree at most equal to r. If the polynomials $q_{n-1}(x)$ and $q_{n-2}(x)$ are quasi orthogonal strictly of order r, the degrees of $E_n(x)$, $F_n(x)$ and $G_n(x)$ are exactly respectively r, $r+1$ and r.

In his paper [49] Sadaka proposed an algorithm for constructing the Padé approximant of a vector function defined in [48].

Our goal in this thesis is to give recurrence relations of quasiorthogonal polynomials. In the second part of our work, we will study a special case of quasiorthogonal polynomials related to vector Padé approximants defined by Sadaka [48], and we will give a four term recurrence relation of the quasiorthogonal polynomials of order p.

The paper in [48] extends results for the theory of orthogonal polynomials to analogous quasi-orthogonal polynomials and discusses their connection with the Padé approximants given by (2.8). It is shown that the polynomial, denoted by $Q_{(m,n)}$ can be expressed as

$$\begin{aligned} Q_{(m,n)_1}(x) &= x^m \widetilde{Q}_{(m,n)_1}(x^{-1}), \\ Q_{(m,n)_2}(x) &= x^n \widetilde{Q}_{(m,n)_2}(x^{-1}), \end{aligned}$$

where

$$Q_{m,n} = \begin{bmatrix} Q_{(m,n)_1} & 0 \\ Q_{(m,n)_2} & Q_{(m,n)_1} \end{bmatrix}.$$

Moreover, these polynomials satisfy the following quasi-orthogonality condition

$$C(x^j Q_{m,n}(x)) = 0_{\mathbb{R}^2}, \quad 0 \leqslant j \leqslant n+p \tag{2.17}$$

with $Q_{(0,n)_1} = 1$ and $Q_{0,-1} = I$.
where the functional C is defined by

$$\begin{cases} C(Ix^i) = \begin{pmatrix} c_i^1 \\ c_i^2 \end{pmatrix}, \\ C(Jx^i) = \begin{pmatrix} 0 \\ c_{2p+1+i}^1 \end{pmatrix} \end{cases}$$

and

$$C(Mx^I) = aC(Ix^i) + bC(Jx^i), \text{ for } M = aI + bJ, \tag{2.18}$$

with

$$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \text{ and } J = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}.$$

Proposition 5 *For $i \geqslant 0$, $p \in \mathbb{N}$, $m > n \geqslant 0$, such that $m - n = 2p + 1$ the polynomials $Q_{m,n}^{(i)}$ exists and is unique if and only if the following matrices are non singular*

$$D_{m,n}^{(i)} = \begin{bmatrix} c_i^1 & \dots & c_{i+m-2}^1 & c_{i+m-1}^1 & 0 & \dots & 0 \\ c_i^2 & \dots & c_{i+m-2}^2 & c_{i+m-1}^2 & c_{i+m-1}^1 & \dots & c_{i+m-n-1}^1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ c_{i+r-1}^1 & \dots & c_{i+m+r-3}^1 & c_{i+m+r-2}^1 & 0 & \dots & 0 \\ c_{i+r-1}^2 & \dots & c_{i+m+r-3}^2 & c_{i+m+r-2}^2 & c_{i+m+r-2}^1 & \dots & c_{i+m-n+r-2}^1 \end{bmatrix} \tag{2.19}$$

Generally, in the determinant $D_{m,n}^{(i)}$, i represents the lower index of the first element, m represents the number of columns in the first block and $n+1$ the numbers of columns in the second block.

According to (2.17), the quasi-orthogonal polynomials $Q_{(m,n)_1}$ and $Q_{(m,n)_2}$ can be written as

$$Q_{(m,n)_1}(x) = \frac{\begin{vmatrix} 1 & x & \dots & \dots & x^{m-1} & x^m & 0 & \dots & 0 & 0 \\ c_0^1 & c_1^1 & c_2^1 & \dots & c_{m-1}^1 & c_m^1 & 0 & \dots & 0 & 0 \\ c_0^2 & c_1^2 & c_2^2 & \dots & c_{m-1}^2 & c_m^2 & c_m^1 & \dots & c_{m-n+1}^1 & c_{mn}^1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\ c_{r-1}^1 & c_r^1 & c_{r+1}^1 & \dots & c_{m+r-2}^1 & c_{m+r-1}^1 & 0 & \dots & 0 & 0 \\ c_{r-1}^2 & c_r^2 & c_{r+1}^2 & \dots & c_{m+r-2}^2 & c_{m+r-1}^2 & c_{m+r-1}^1 & \dots & c_{m-n+r}^1 & c_{m-n+r-1}^1 \end{vmatrix}}{|D_{m,n}^{(1)}|}$$

and

$$Q_{(m,n)_2}(x) = \frac{\begin{vmatrix} 0 & 0 & 0 & \dots & \dots & 0 & x^n & \dots & x & 1 \\ c_0^1 & c_1^1 & c_2^1 & \dots & c_{m-1}^1 & c_m^1 & 0 & \dots & 0 & 0 \\ c_0^2 & c_1^2 & c_2^2 & \dots & c_{m-1}^2 & c_m^2 & c_m^1 & \dots & c_{m-n+1}^1 & c_{m-n}^1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\ c_{r-1}^1 & c_r^1 & c_{r+1}^1 & \dots & c_{m+r-2}^1 & c_{m+r-1}^1 & 0 & \dots & 0 & 0 \\ c_{r-1}^2 & c_r^2 & c_{r+1}^2 & \dots & c_{m+r-2}^2 & c_{m+r-1}^2 & c_{m+r-1}^1 & \dots & c_{m-n+r}^1 & c_{m-n+r-1}^1 \end{vmatrix}}{|D_{m,n}^{(1)})|}$$

where $Q_{0,-1} = I$.

Let $Q_{m,n}$ quasiorthogonal polynomials defined in (2.25) and $\tilde{Q}_{m,n}$ the vector Padé approximants then we have a relation between these polynomials

$$\tilde{Q}_{(m,n)_1}(x) = x^m Q_{(m,n)_1}(x^{-1}) \times \frac{\Delta_{m,n}}{|D_{m,n}^{(1)}|} \tag{2.20}$$

and

$$\tilde{Q}_{(m,n)_2}(x) = x^n Q_{(m,n)_2}(x^{-1}) \times \frac{\Delta_{m,n}}{|D_{m,n}^{(1)}|}. \tag{2.21}$$

Adjacent systems of quasi-orthogonal polynomials denoted by $Q^{(i)}_{(m,n)}$ of the form

$$Q^{(i)}_{m,n} = \left[\begin{array}{cc} Q^{(i)}_{(m,n)_1} & 0 \\ Q^{(i)}_{(m,n)_2} & Q^{(i)}_{(m,n)_1} \end{array} \right], \tag{2.22}$$

are defined such that

$$C^{(i)}(x^j Q^{(i)}_{m,n}(x)) = 0_{\mathbb{R}^2}, \quad 0 \leqslant j \leqslant n+p \tag{2.23}$$

with $Q^{(k)}_{(0,-1)_1} = I$ and where the linear function $C^{(i)}$ is defined by

$$C^{(k)}(Ix^i) = \begin{pmatrix} c^1_{i+k} \\ c^2_{i+k} \end{pmatrix}, \quad C^{(k)}(Jx^i) = \begin{pmatrix} 0 \\ c^1_{k+2p+1+i} \end{pmatrix}$$

and

$$C^{(k)}(Mx^I) = aC^{(k)}(Ix^i) + bC^{(k)}(Jx^i), \text{ for } M = aI + bJ, \tag{2.24}$$

with I and J are the same in (2.17).

The results given in this work are obtained thanks to the use of Schur complements and their properties, we will recall some of these properties in the next section and we will show the link with Padé approximation.

2.6 Schur complement

The Schur complement [39] has many applications in numerical analysis, it has been used in extrapolation methods and projection algorithms.
In [7], the Schur complement and its properties are applied to give the recursive polynomial interpolation algorithm (RPIA) for computing the polynomial interpolation.
In [3], the Schur complement identities are used to give the matrix recursive polynomial interpolation algorithm (MRPIA) for computing the Hermite polynomial interpolation.
In [19], the Schur complement is applied to build two algorithms, called the recursive interpolation algorithm (RIA) and the recursive projection algorithm (RPA).

We first recall the definition of Schur complements [39] and we give some of their properties.

Définition 6 *[17] Let M_1 be a matrix partitioned into 4 blocks of the form*

$$M_1 = \left[\begin{array}{cc} A & B \\ C & D \end{array} \right], \tag{2.25}$$

where the submatrix D is assumed to be square and nonsingular. The Schur complement of D in M_1 denoted by (M_1/D), is defined by

$$(M_1/D) = A - BD^{-1}C. \tag{2.26}$$

If D is not a square matrix then a pseudo-Schur complement of D in M_1 can still be defined as in [51] and [16].
Let us remark that having the nonsingular submatrix D in the lower right-hand side corner of M_1 is a matter of convention. We can similarly define the following Schur complements :

$$\begin{aligned}(M_1/A) &= D - CA^{-1}B,\\ (M_1/B) &= C - DB^{-1}A,\\ (M_1/C) &= B - AC^{-1}D.\end{aligned}$$

Now we will give some properties of the Schur complement.

Proposition 2 *[17] We assume that the submatrix D is nonsigular and the matrix M_1 is a square matrix defined by* (2.25)*, so we have*

$$|(M_1/D)| = \frac{|M_1|}{|D|},$$

where $|X|$ denote the determinant of the square matrix X.

Proposition 3 *[18] If $A = x \in \mathbb{R}^p$ and B is the vector row $(z_1, \ldots, z_k)$ with $z_i \in \mathbb{R}^p$, then,*

$$(M_1/D) = \frac{|M_1|}{|D|}. \tag{2.27}$$

Proposition 4 *[42] Let us assume that the submatrix D is nonsingular, then*

$$\left(\begin{bmatrix} A & B \\ C & D \end{bmatrix}/D\right) = \left(\begin{bmatrix} C & D \\ A & B \end{bmatrix}/D\right) = \left(\begin{bmatrix} D & C \\ B & A \end{bmatrix}/D\right) = \left(\begin{bmatrix} B & A \\ D & C \end{bmatrix}/D\right). \tag{2.28}$$

Proposition 5 *[42] Assuming that the matrix D is nonsingular and E is a matrix such that the product EA is well defined, then*

$$\left(\begin{bmatrix} EA & EB \\ C & D \end{bmatrix}/D\right) = E\left(\begin{bmatrix} A & B \\ C & D \end{bmatrix}/D\right). \tag{2.29}$$

We recall the Sylvester's identity [42] which is a particular case of the quotient property [27]. Consider the matrices K, M_3, M_2 and M_4 partitioned as follows :

$$K = \begin{bmatrix} A & B & E \\ C & D & F \\ G & H & L \end{bmatrix}, \quad M_1 = \begin{bmatrix} A & B \\ C & D \end{bmatrix},$$

$$M_2 = \begin{bmatrix} B & E \\ D & F \end{bmatrix}, \quad M_3 = \begin{bmatrix} D & F \\ H & L \end{bmatrix} \text{and} \quad M_4 = \begin{bmatrix} C & D \\ G & H \end{bmatrix}.$$

Proposition 6 *(The Sylvester identity)[42] If the matrices M_3 and D are nonsingular, then*

$$(K/M_3) = ((K/D)/(M_3/D)) = (M_1/D) - (M_2/D)(M_3/D)^{-1}(M_4/D). \qquad (2.30)$$

Proposition 7 *(The quotient property). If the matrices M_2 and B are nonsingular, then*

$$(K/M_2) = ((K/B)/(M_2/B)) = (\begin{bmatrix} A & B \\ G & H \end{bmatrix}/B) - (\begin{bmatrix} B & E \\ H & L \end{bmatrix}/B)(M_2/B)^{-1}(M_1/B).$$

Schur complements are related to Padé approximants, as is showed by C.Brezinski [17].

Proposition 8 *[17] Let f be the formal function in (1.3), the Padé approximant $[p/q]$ in (2.1), can be written as*

$$[p/q] = \frac{\begin{vmatrix} \sum_{i=0}^{p-q} c_i x^i & c_{p-q+1} & \dots & c_p \\ -x^{p-q+1} c_{p-q+1} & c_{p-q+1} - x c_{p-q+2} & \dots & c_p + x c_{p+1} \\ \vdots & \vdots & \vdots & \vdots \\ -x^{p-q+1} c_p & c_p - x c_{p+1} & \dots & c_p + x c_{p+1} \end{vmatrix}}{|W|}$$

where

$$W = \begin{bmatrix} c_{p-q+1} - x c_{p-q+2} & \dots & c_p + x c_{p+1} \\ \vdots & \vdots & \vdots \\ c_p - x c_{p+1} & \dots & c_p + x c_{p+1} \end{bmatrix}$$

Taking the Schur complement of W in the numerator of $[p/q]$ and applying Schur's formula(2.26), we obtain

$$[p/q] = \sum_{i=0}^{p-q} c_i x^i + x^{p-q+1} \left\langle c, W^{-1} c \right\rangle \qquad (2.31)$$

where $c = \begin{bmatrix} c_{p-q+1} & \dots & c_p \end{bmatrix}^T$.

3

Computing recursive orthogonal polynomial with Schur complements

In this chapter, we will give three recursive algorithms for computing the orthogonal polynomials. The first algorithm is derived from the recurrence relation, obtained by the Schur complement, and the qd algorithm. The second is given by using the recurrence relationship of Hankel determinants, which we recover by applying the Sylvester identity and the Schur complement's properties. These two algorithms allow us to fill the table of orthogonal polynomials and adjacent systems. The third algorithm is obtained by applying the Sylvester identity and using an auxiliary polynomial. It will be used for computing recursively the orthogonal polynomials of the same family.

3.1 The recurrence relation of orthogonal polynomials

In this section, the orthogonal polynomials are expressed as Schur complements, we will give a recurrence relation satisfied by the orthogonal polynomials.

3.1.1 Explicit expression of orthogonal polynomials

Let $Q_k \in \mathcal{P}$, written as

$$Q_k(x) = \sum_{i=0}^{k} a_i x^i,$$

then

$$c(x^i Q_k(x)) = 0 \quad for \;\; 0 \leqslant i \leqslant k-1,$$

by the orthogonality relations and the linearity of c, we get

$$\begin{cases} c_0 + a_1 c_1 + \ldots + a_k c_k = 0 \\ c_1 + a_1 c_2 + \ldots + a_k c_{k+1} = 0 \\ \quad \vdots \\ c_{k-1} + a_1 c_k + \ldots + a_k c_{2k-1} = 0 \end{cases} \tag{3.1}$$

with $a_0 = 1$.

The system (3.1) can be written as follows :

$$\begin{bmatrix} c_1 & c_2 & \ldots & c_{k-1} & c_k \\ c_2 & c_3 & \ldots & c_k & c_{k+1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ c_{k-1} & c_k & \ldots & c_{2k-3} & c_{2k-2} \\ c_k & c_{k+1} & \ldots & \ldots & c_{2k-1} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ \vdots \\ a_k \end{bmatrix} = - \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ \vdots \\ c_{k-1} \end{bmatrix}. \tag{3.2}$$

The solution of (3.2) exists and is unique if and only if the coefficient matrix denoted by $D_k^{(1)}$ and called Hankel matrix, is nonsingular. We mention that $|D_k^{(i)}| = H_k^{(i)}$.

The expression of $Q_k(x)$ is given by the following theorem :

Théorème 6 *[15] Assuming that $H_k^{(1)} \neq 0$, then Q_k exists and is unique. Q_k can* *be computed by developing the following determinant with respect to the last row*

$$Q_k(x) = \frac{1}{H_k^{(1)}} \begin{vmatrix} c_0 & c_1 & \dots & \dots & c_k \\ c_1 & c_2 & \dots & \dots & c_{k+1} \\ \vdots & \vdots & & & \vdots \\ c_{k-1} & c_k & \dots & \dots & c_{2k-1} \\ 1 & x & \dots & \dots & x^k \end{vmatrix}, \quad \text{for } k = 1, 2, \dots, \tag{3.3}$$

and $Q_0(x) = 1$.

The polynomial Q_k can be also expressed as a Schur complement. We will prove this in the following theorem :

Théorème 7 *Assuming that the matrix* $D_k^{(1)}$ *is square and nonsingular, then we have*

$$Q_k(x) = \left(\left[\begin{array}{c|ccccc} 1 & x & \dots & \dots & x^{k-1} & x^k \\ \hline c_0 & c_1 & \dots & \dots & c_{k-1} & c_k \\ c_1 & c_2 & \dots & \dots & c_k & c_{k+1} \\ \vdots & \vdots & & & \vdots & \vdots \\ c_{k-1} & c_k & \dots & \dots & c_{2k-2} & c_{2k-1} \end{array} \right] / D_k^{(1)} \right), \tag{3.4}$$

with $Q_0(x) = 1$.

Proof.

From (3.2), the coefficients $a_1, \dots\dots, a_k$ are given by

$$\begin{bmatrix} a_1 \\ \vdots \\ \vdots \\ a_k \end{bmatrix} = -\left(D_k^{(1)}\right)^{-1} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{k-1} \end{bmatrix},$$

with the condition $a_0 = 1$, then we have

$$Q_k(x) = 1 - \begin{bmatrix} x & x^2 & \dots & x^k \end{bmatrix} (D_k^{(1)})^{-1} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ \vdots \\ c_{k-1} \end{bmatrix}. \tag{3.5}$$

From (2.26) and (3.5), we see that Q_k can be written as a Schur complement.

$$Q_k(x) = \left(\left[\begin{array}{c|cccc} 1 & x & \dots & \dots & x^k \\ \hline c_0 & c_1 & \dots & \dots & c_k \\ c_1 & c_2 & \dots & \dots & c_{k+1} \\ \vdots & \vdots & & & \vdots \\ c_{k-1} & c_k & \dots & \dots & c_{2k-1} \end{array} \right] / D_k^{(1)} \right). \tag{3.6}$$

We can similarly prove that the polynomial $Q_k^{(i)}$ can be expressed as a Schur complement.

Proposition 9 *For $i \geqslant 0$ and $k \geqslant 1$, assuming that $D_k^{(i+1)}$ is square and nonsingular. The polynomial $Q_k^{(i)}$ can be written as a Schur complement*

$$Q_k^{(i)}(x) = \left(\left[\begin{array}{c|cccc} 1 & x & \dots & \dots & x^k \\ \hline c_i & c_{i+1} & \dots & \dots & c_{i+k} \\ c_{i+1} & c_{i+2} & \dots & \dots & c_{i+k+1} \\ \vdots & \vdots & & & \vdots \\ c_{i+k-1} & c_{i+k} & \dots & \dots & c_{i+2k-1} \end{array} \right] / D_k^{(i+1)} \right), \tag{3.7}$$

with $Q_0^{(i)}(x) = 1$.

3.1.2 The recurrence relation

For $k \geqslant 2$, the polynomial Q_k is partitioned into nine blocks

$$Q_k(x) = \left(\left[\begin{array}{c|cccc|c} 1 & x & \dots & \dots & x^{k-1} & x^k \\ \hline c_0 & c_1 & \dots & \dots & c_{k-1} & c_k \\ c_1 & c_2 & \dots & \dots & c_k & c_{k+1} \\ \vdots & \vdots & & & \vdots & \vdots \\ c_{k-2} & c_{k-1} & \dots & \dots & c_{2k-3} & c_{2k-2} \\ \hline c_{k-1} & c_k & \dots & \dots & c_{2k-2} & c_{2k-1} \end{array} \right] / D_k^{(1)} \right), \tag{3.8}$$

with $Q_0(x) = 1$ and $Q_1(x) = 1 - \dfrac{c_0}{c_1}x$.

For $k \geqslant 1$ and $i \geqslant 0$, we define the polynomial

$$G_k^{(i)}(x) = \left(\left[\begin{array}{ccccc|c} 1 & x & \dots & \dots & x^{k-1} & x^k \\ \hline c_i & c_{i+1} & \dots & \dots & c_{i+k-1} & c_{i+k} \\ c_{i+1} & c_{i+2} & \dots & \dots & c_{i+k} & c_{i+k+1} \\ \vdots & \vdots & \vdots & \vdots & \dots & \vdots \\ c_{i+k-1} & c_{i+k} & \dots & \dots & c_{i+2k-2} & c_{i+2k-1} \end{array} \right] / D_k^{(i)} \right). \tag{3.9}$$

We can prove that the polynomial $G_k^{(1)}$ verifies the relation given by the following proposition :

Proposition 10 *Let $G_k^{(1)}$ be the polynomial given by (3.9) and $Q_k^{(1)}$ is given by (3.7), applying some properties of the Schur complement, we get the following relation :*

$$G_{k-1}^{(1)}(x) = Q_{k-1}^{(1)}(x) \times (D_k^{(1)}/D_{k-1}^{(2)})^{-1} \times (D_k^{(1)}/D_{k-1}^{(1)}), \quad \textit{for } k \geqslant 1,$$

with $Q_0^{(1)}(x) = G_0^{(1)}(x) = 1$.

Proof.

Using Proposition 2 , we have

$$|G_{k-1}^{(1)}(x)| = G_{k-1}^{(1)}(x),$$

then

$$G_{k-1}^{(1)}(x) = \frac{\begin{vmatrix} 1 & x & \dots & x^{k-2} & x^{k-1} \\ c_1 & c_2 & \dots & c_{k-1} & c_k \\ c_2 & c_3 & \dots & c_k & c_{k+1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ c_{k-1} & c_k & \dots & c_{2k-3} & c_{2k-2} \end{vmatrix}}{H_{k-1}^{(1)}} \tag{3.10}$$

$$= \frac{\begin{vmatrix} 1 & x & \dots & x^{k-2} & x^{k-1} \\ c_1 & c_2 & \dots & c_{k-1} & c_k \\ c_2 & c_3 & \dots & c_k & c_{k+1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ c_{k-1} & c_k & \dots & c_{2k-3} & c_{2k-2} \end{vmatrix}}{H_{k-1}^{(2)}} \times \frac{H_{k-1}^{(2)}}{H_{k-1}^{(1)}} \tag{3.11}$$

$$= \frac{\begin{vmatrix} 1 & x & \dots & x^{k-2} & x^{k-1} \\ c_1 & c_2 & \dots & c_{k-1} & c_k \\ c_2 & c_3 & \dots & c_k & c_{k+1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ c_{k-1} & c_k & \dots & c_{2k-3} & c_{2k-2} \end{vmatrix}}{H_{k-1}^{(2)}} \times \frac{H_{k-1}^{(2)}}{H_k^{(1)}} \frac{H_k^{(1)}}{H_{k-1}^{(1)}} \tag{3.12}$$

$$= \left(\left[\begin{array}{c|cccc} 1 & x & \dots & x^{k-2} & x^{k-1} \\ \hline c_1 & c_2 & \dots & c_{k-1} & c_k \\ c_2 & c_3 & \dots & c_k & c_{k+1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ c_{k-1} & c_k & \dots & c_{2k-3} & c_{2k-2} \end{array} \right] / D_{k-1}^{(2)} \right) \times \left(D_k^{(1)}/D_{k-1}^{(2)} \right)^{-1} \left(D_k^{(1)}/D_{k-1}^{(1)} \right). \tag{3.13}$$

Hence, from the expression (3.7) and using Proposition 2, we get

$$G_{k-1}^{(1)}(x) = Q_{k-1}^{(1)}(x) \times (D_k^{(1)}/D_{k-1}^{(2)})^{-1} \times (D_k^{(1)}/D_{k-1}^{(1)}). \tag{3.14}$$

Using (2.27), (2.29) and (3.14) , we will prove that the orthogonal polynomials and the adjacent orthogonal polynomials satisfy a recurrence relation given by the following theorem :

Théorème 8 *For $k \geqslant 1$, the polynomials Q_k, $Q_{k-1}^{(1)}$ and Q_{k-1} verify the following recurrence relation :*

$$Q_k(x) = Q_{k-1}(x) - xQ_{k-1}^{(1)}(x) \times (D_k^{(1)}/D_{k-1}^{(2)})^{-1}(D_k^{(0)}/D_{k-1}^{(1)}), \tag{3.15}$$

where $Q_0(x) = 1$.

Proof.

We apply the Sylvester identity to the polynomial Q_k partitioned as follows :

$$Q_k(x) = \left(\left[\begin{array}{c|cccc|c} 1 & x & \dots & \dots & x^{k-1} & x^k \\ \hline c_0 & c_1 & \dots & \dots & c_{k-1} & c_k \\ c_1 & c_2 & \dots & \dots & c_k & c_{k+1} \\ \vdots & \vdots & & & \vdots & \vdots \\ c_{k-2} & c_{k-1} & \dots & \dots & c_{2k-3} & c_{2k-2} \\ \hline c_{k-1} & c_k & \dots & \dots & c_{2k-2} & c_{2k-1} \end{array} \right] / D_k^{(1)} \right)$$

$$= \left(\left[\begin{array}{c|cccc} 1 & x & \dots & \dots & x^{k-1} \\ \hline c_0 & c_1 & \dots & \dots & c_{k-1} \\ c_1 & c_2 & \dots & \dots & c_k \\ \vdots & \vdots & & & \vdots \\ c_{k-2} & c_{k-1} & \dots & \dots & c_{2k-3} \end{array} \right] / D_{k-1}^{(1)} \right) - \left(\left[\begin{array}{cccc|c} x & \dots & \dots & x^{k-1} & x^k \\ \hline c_1 & \dots & \dots & c_{k-1} & c_k \\ c_2 & \dots & \dots & c_k & c_{k+1} \\ \vdots & & & \vdots & \vdots \\ c_{k-1} & \dots & \dots & c_{2k-3} & c_{2k-2} \end{array} \right] / D_{k-1}^{(1)} \right)$$

$$\times (D_k^{(1)}/D_{k-1}^{(1)})^{-1}(D_k^{(0)}/D_{k-1}^{(1)}),$$

then

$$Q_k(x) = Q_{k-1}(x) - xG_{k-1}^{(1)}(x) \times \left(D_k^{(1)}/D_{k-1}^{(1)}\right)^{-1} \times \left(D_k^{(0)}/D_{k-1}^{(1)}\right),$$

we substitute the expression of $G_{k-1}^{(1)}$ given by (3.14) in the previous relation and we obtain

$$Q_k(x) = Q_{k-1}(x) - xQ_{k-1}^{(1)}(x) \times (D_k^{(1)}/D_{k-1}^{(2)})^{-1} \times (D_k^{(0)}/D_{k-1}^{(1)}).$$

Remarque 9 *If we use Proposition 2, we have*

$$(D_k^{(1)}/D_{k-1}^{(2)})^{-1} = \frac{H_{k-1}^{(2)}}{H_k^{(1)}} \;\; and \;\; \left(D_k^{(0)}/D_{k-1}^{(1)}\right) = \frac{H_k^{(0)}}{H_{k-1}^{(0)}}.$$

The relation given by (3.15) *will be written as follows :*

$$Q_k(x) = Q_{k-1}(x) - x\frac{H_k^{(0)}H_{k-1}^{(2)}}{H_k^{(1)}H_{k-1}^{(1)}}Q_{k-1}^{(1)}(x).$$

In the general case where $i \geqslant 0$, we have the recurrence relationship between two families of orthogonal polynomials, $Q_k^{(i)}, Q_{k-1}^{(i+1)}$ and $Q_{k-1}^{(i)}$.

Théorème 10 *For $i \geqslant 0$ and $k \geqslant 1$, the polynomials $Q_k^{(i)}, Q_{k-1}^{(i)}, Q_{k-1}^{(i+1)}$ verify the following recurrence relation :*

$$Q_k^{(i)}(x) = Q_{k-1}^{(i)}(x) - x\beta_k^{(i)}Q_{k-1}^{(i+1)}(x), \tag{3.16}$$

with

$$\beta_k^{(i)} = \frac{H_k^{(i)}H_{k-1}^{(i+2)}}{H_k^{(i+1)}H_{k-1}^{(i+1)}}, \tag{3.17}$$

and $Q_0^{(i)}(x) = 1$.

Proof.
The proof of this theorem can be done in a similar way of the previous theorem and we use (2.27) to replace the Schur complement by the ratio of Hankel determinants.

The recurrence relation given by (3.16) was introduced by C.Brezinski and al.[22], with a parameter λ defined from the orthogonality relations and the three-terms recurrence relationship, but in our case we get directly the coefficients $\beta_k^{(i)}, \;\; i = 0, \ldots, \;\; k = 1, 2, \ldots$ given as ratio of Hankel determinants.

3.2 Formulation of the recursive orthogonal polynomial algorithm

In this section, we will give two recursive algorithms for computing the orthogonal polynomials.

3.2.1 The qd algorithm

In the recurrence relation given by (3.16), we propose to compute recursively the coefficients $\beta_k^{(i)}$ given by (3.17) using the qd algorithm [37].

Définition 7 *[36] Let the expressions of $q_k^{(i)}$ and $e_k^{(i)}$ given by*

$$q_k^{(i)} = \frac{H_k^{(i+1)} H_{k-1}^{(i)}}{H_k^{(i)} H_{k-1}^{(i+1)}} \tag{3.18}$$

and

$$e_k^{(i)} = \frac{H_{k+1}^{(i)} H_{k-1}^{(i+1)}}{H_k^{(i)} H_k^{(i+1)}}, \quad \text{for } i \geqslant 0, k \geqslant 1. \tag{3.19}$$

The qd algorithm is a recursive algorithm which allows us to calculate the expressions $e_k^{(i)}$ and $q_k^{(i)}$ without computing all Hankel determinants in the expressions given by (3.18) *and* (3.19).

The following relations define the qd algorithm

$$\begin{cases} q_1^{(i)} = \dfrac{c_{i+1}}{c_i}, e_0^{(i)} = 0 \quad i = 0, 1, \dots \\ q_{k+1}^{(i)} e_k^{(i)} = q_k^{(i+1)} e_k^{(i+1)} \quad k = 1, 2, \dots i = 0, 1, \dots \\ q_k^{(i)} + e_k^{(i)} = q_k^{(i+1)} + e_{k-1}^{(i+1)}. \end{cases} \tag{3.20}$$

Using the relations given by (3.18) and (3.19) we find the second relation in (3.20) and the third relation in (3.20) is obtained from (3.18), (3.19) and from the recurrence relation between the Hankel determinants given by (3.23).

3.2.2 The computation of the coefficients $\beta_k^{(i)}$

The following theorem gives the algorithm for computing the $\beta_k^{(i)}$ using the qd algorithm.

Théorème 11 *For $k = 1, \dots,$ and $i = 0, \dots,$ the quantities $\beta_k^{(i)}, q_k^{(i)}$ and $e_k^{(i)}$ are given by* (3.17) , (3.18) *and* (3.19) *respectively. The following relations define a recursive algorithm for computing the coefficients $\beta_k^{(i)}$*

$$\begin{cases} q_1^{(i)} = \dfrac{c_{i+1}}{c_i}, e_0^{(i)} = 0, \beta_1^{(i)} = \dfrac{c_i}{c_{i+1}}, \quad i = 0, 1, \dots \\ q_{k+1}^{(i)} e_k^{(i)} = q_k^{(i+1)} e_k^{(i+1)}, \quad k = 1, 2, \dots, i = 0, 1, \dots \\ q_k^{(i)} + e_k^{(i)} = q_k^{(i+1)} + e_{k-1}^{(i+1)}, \quad k = 1, 2, \dots, i = 0, 1, \dots \\ q_{k+1}^{(i)} \beta_{k+1}^{(i)} = q_k^{(i+1)} \beta_k^{(i)}, \quad k = 1, 2, \dots, i = 0, 1, \dots . \end{cases} \tag{3.21}$$

Proof.

The proof will be done for $i = 0$ and can be generalized for all $i > 0$.

For $k \geqslant 1$ and $i \geqslant 0$, the $q_k^{(i)}$ and $e_k^{(i)}$ are computed recursively from the qd algorithm.

We have

$$\beta_k^{(0)} = \frac{H_k^{(0)} H_{k-1}^{(2)}}{H_k^{(1)} H_{k-1}^{(1)}},$$

and

$$q_k^{(0)} = \frac{H_k^{(1)} H_{k-1}^{(0)}}{H_k^{(0)} H_{k-1}^{(1)}},$$

then

$$q_k^{(0)} \beta_k^{(0)} = \frac{H_k^{(0)} H_{k-1}^{(2)}}{H_k^{(1)} H_{k-1}^{(1)}} \frac{H_k^{(1)} H_{k-1}^{(0)}}{H_k^{(0)} H_{k-1}^{(1)}} = \frac{H_{k-1}^{(0)} H_{k-1}^{(2)}}{(H_{k-1}^{(1)})^2}.$$

We see that

$$q_{k-1}^{(1)} \beta_{k-1}^{(0)} = \frac{H_{k-1}^{(0)} H_{k-1}^{(2)}}{(H_{k-1}^{(1)})^2},$$

then we obtain

$$q_k^{(0)} \beta_k^{(0)} = q_{k-1}^{(1)} \beta_{k-1}^{(0)}, \tag{3.22}$$

with $q_1^{(0)} \beta_1^{(0)} = 1$.

Similarly, we can prove that $q_k^{(i)} \beta_k^{(i)} = q_{k-1}^{(i+1)} \beta_{k-1}^{(i)}$.

Remarque 12 *From the relation given by* (3.22)*, the* $\beta_k^{(0)}$ *are obtained recursively without computing each Hankel determinant in the expression given by* (3.17)*.*

Using the relations given by (3.21), then applying the relation given by (3.16), we get the Recursive Orthogonal Polynomial Algorithm (ROPA) as follows :

Algorithm 1 *(ROPA)*

1. *Compute* $e_1^{(i)} = 0, q_2^{(i)} = \dfrac{c_{i+1}}{c_i}, \beta_2^{(i)} = \dfrac{c_i}{c_{i+1}}, Q_0^{(i)}(x) = 1.$
2. *For* $k = 2, \ldots, m-1$
   - *For* $i = 1, \ldots, m-2$
     - *Compute* $e_k^{(i)} = (q_k^{(i+1)} + e_{k-1}^{(i+1)}) - q_k^{(i)}$.
   - *For* $i = 1, \ldots, m-2$
     - *Compute* $q_{k+1}^{(i)} = \frac{q_k^{(i+1)} e_k^{(i+1)}}{e_k^{(i)}}$.
   - *Compute* q, e.
   - *For* $i = 1, \ldots, m-2$
     - *Compute* $\beta_{k+1}^{(i)} = \dfrac{q_k^{(i+1)}}{q_{k+1}^{(i)}} \beta_k^{(i)}$.
3. *For* $i = 1, \ldots, m-2$
   $l = l + 1$
   - *For* $k = 2, \ldots, min(l, m)$
     - *Compute* $Q_k^{(i)}(x) = Q_{k-1}^{(i)}(x) - x\beta_k^{(i)} Q_{k-1}^{(i+1)}(x)$.

Example 1 *Let the function $f(x) = e^x$, expressed as*

$$f(x) = \sum_{i \geqslant 0} \frac{1}{i!} x^i,$$

where $c_i = \dfrac{1}{i!}$.

The Padé approximation consists to find a rational fraction $\dfrac{\tilde{Q}}{\tilde{P}}$, to approach the function f such that the following relation is verified

$$\tilde{P}(x)e^x - \tilde{Q}(x) = O(x^{p+q+1}),$$

with $deg(\tilde{P}) = p, deg(\tilde{Q}) = q$.

In this example, we propose to compute recursively the orthogonal polynomial P of degree p, such that

$$P(x) = x^p \tilde{P}(x^{-1}).$$

We will use the ROPA to compute $Q_k^{(i)}$, for $k = 1, \ldots, 10$ and $i = 0, \ldots, 9$.

We have $i \geqslant 0$ and $x \in \mathbb{R}$

$$Q_0^{(i)}(x) = 1, q_1^{(i)} = \frac{c_{i+1}}{c_i}, e_0^{(i)} = 0, \beta_1^{(i)} = \frac{c_i}{c_{i+1}},$$

then

$$Q_1(x) = Q_0(x) - x\beta_1^{(0)} Q_0^{(1)}(x) = 1 - x \times \frac{c_0}{c_1} = 1 - x,$$

$$Q_1^{(1)}(x) = Q_0^{(1)}(x) - x\beta_1^{(1)} Q_0^{(2)}(x) = 1 - 2x.$$

The polynomial Q_2 is obtained by using the recurrence relationship given by (3.16).

$$Q_2(x) = Q_1(x) - x\beta_2^{(0)} Q_1^{(1)}(x).$$

The coefficient $\beta_2^{(0)}$ will be computed by the qd algorithm

$$\begin{cases} e_1^{(0)} = q_1^{(1)} - q_1^{(0)} + e_0^{(1)} = \dfrac{c_2}{c_1} - \dfrac{c_1}{c_0} = -\frac{1}{2} \\ e_1^{(1)} = q_1^{(2)} - q_1^{(1)} + e_0^{(2)} = \dfrac{c_3}{c_2} - \dfrac{c_2}{c_1} = \frac{-1}{6} \\ q_1^{(1)} = \dfrac{c_2}{c_1} = \frac{1}{2} \\ q_2^{(0)} = \dfrac{e_1^{(1)}}{e_1^{(0)}} q_1^{(1)} = \frac{1}{6} \\ \beta_2^{(0)} = \dfrac{q_1^1}{q_2^{(0)}} \beta_1^{(0)} = 3. \end{cases}$$

We substitute the expression of the polynomials Q_1 *and* $Q_1^{(1)}$ *in the expression of* Q_2 *and we get*

$$Q_2(x) = 1 - 4x + 6x^2.$$

Thus, we obtain all the polynomials in the following table :

$$\begin{array}{cccccc}
Q_0 & Q_0^{(1)} & Q_0^{(2)} & \dots & \dots & Q_0^{(10)} \\
 & Q_1 & Q_1^{(1)} & \dots & \dots & Q_1^{(9)} \\
 & & Q_2 & \ddots & \ddots & \vdots \\
 & & & & \ddots & \ddots \\
 & & & & Q_9 & Q_9^{(1)} \\
 & & & & & Q_{10}
\end{array}$$

The expressions of the polynomials $Q_1, Q_2, \dots, Q_{10}$ *are given by*

$$Q_0(x) = 1$$

$$Q_1(x) = 1 - x$$

$$Q_2(x) = 1 - 4x + 6x^2$$

$$Q_3(x) = 1 - 9x + 36x^2 + 60x^3$$

$$\vdots$$

$$\vdots$$

$$\begin{aligned}
Q_{10}(x) = 1 &-(1759218625670887x)/17592186044416+(340161417486977x^2)/68719476736 \\
&-(2721291364475551x^3)/17179869184 + (7738672378815101x^4)/2147483648 \\
&-(8125606054160473x^5)/134217728 + (793516221098513x^6)/1048576 \\
&-(1813751372413587x^7)/262144 + (5781332527780123x^8)/131072 \\
&-(5781332553114357x^9)/32768 + (5492265947166785x^{10})/16384.
\end{aligned}$$

3.3 The Recurrence relation of Hankel determinants

The coefficients $\beta_k^{(i)}$ can be computed if we calculate every Hankel determinant in the expression given by (3.17).
In the following theorem, we will prove that the well known recurrence relation between the Hankel determinants can be found by using the Sylvester identity.

Théorème 13 *[15] For $k \geqslant 2$ and $i \geqslant 0$, the Hankel determinants obey the recurrence relation*

$$H_k^{(i)} H_{k-2}^{(i+2)} = H_{k-1}^{(i+2)} H_{k-1}^{(i)} - (H_{k-1}^{(i+1)})^2, \tag{3.23}$$

which initialized with $H_0^{(i)} = 1$ and $H_1^{(i)} = c_i$.

Proof.

We apply the Sylvester identity to $(D_k^{(i)}/D_{k-1}^{(i+2)})$

$$\left(D_k^{(i)}/D_{k-1}^{(i+2)}\right) = \left(\left[\begin{array}{c|cccc|c} c_i & c_{i+1} & c_{i+2} & \cdots & c_{i+k-2} & c_{i+k-1} \\ \hline c_{i+1} & c_{i+2} & c_{i+3} & \cdots & c_{i+k-1} & c_{i+k} \\ \vdots & \vdots & & & \vdots & \vdots \\ c_{i+k-2} & c_{i+k-1} & \cdots & \cdots & c_{i+2k-4} & c_{i+2k-3} \\ \hline c_{i+k-1} & c_{i+k} & c_{i+k+1} & \cdots & c_{i+2k-3} & c_{i+2k-2} \end{array} \right] / D_{k-1}^{(i+2)} \right),$$

then we have

$$(D_k^{(i)}/D_{k-1}^{(i+2)}) = \left(\left[\begin{array}{c|cccc} c_i & c_{i+1} & c_{i+2} & \cdots & c_{i+k-2} \\ \hline c_{i+1} & c_{i+2} & c_{i+3} & \cdots & c_{i+k-1} \\ \vdots & \vdots & & & \vdots \\ c_{i+k-2} & c_{i+k-1} & \cdots & c_{i+2k-5} & c_{i+2k-4} \end{array} \right] / D_{k-2}^{(i+2)} \right)$$

$$- \left(\left[\begin{array}{c|cccc} c_{i+1} & c_{i+2} & c_{i+3} & \cdots & c_{i+k-1} \\ c_{i+2} & c_{i+3} & c_{i+4} & \cdots & c_{i+k} \\ \vdots & \vdots & \vdots & & \vdots \\ c_{i+k-2} & c_{i+k-1} & \cdots & c_{i+2k-5} & c_{i+2k-4} \\ \hline c_{i+k-1} & c_{i+k} & \cdots & c_{i+2k-4} & c_{i+2k-3} \end{array} \right] / D_{k-2}^{(i+2)} \right) \times (D_{k-1}^{(i+2)}/D_{k-2}^{(i+2)})^{-1}$$

$$\times \left(\left[\begin{array}{c|cccc} c_{i+1} & c_{i+2} & c_{i+3} & \cdots & c_{i+k-1} \\ c_{i+2} & c_{i+3} & c_{i+4} & \cdots & c_{i+k} \\ \vdots & \vdots & \vdots & & \vdots \\ c_{i+k-2} & c_{i+k-1} & \cdots & c_{i+2k-5} & c_{i+2k-4} \\ \hline c_{i+k-1} & c_{i+k} & c_{i+k+1} & \cdots & c_{i+2k-3} \end{array} \right] / D_{k-2}^{(i+2)} \right).$$

Hence,

$$(D_k^{(i)}/D_{k-1}^{(i+2)}) = (D_{k-1}^{(i)}/D_{k-2}^{(i+2)}) - (D_{k-1}^{(i+1)}/D_{k-2}^{(i+2)})^2 (D_{k-1}^{(i+2)}/D_{k-2}^{(i+2)})^{-1},$$

so using Proposition 2, we get

$$\frac{H_k^{(i)}}{H_{k-1}^{(i+2)}} = \frac{H_{k-1}^{(i)}}{H_{k-2}^{(i+2)}} - \left(\frac{H_{k-1}^{(i+1)}}{H_{k-2}^{(i+2)}} \right)^2 \times \frac{H_{k-2}^{(i+2)}}{H_{k-1}^{(i+2)}}.$$

Therefore

$$H_k^{(i)}H_{k-2}^{(i+2)} = H_{k-1}^{(i+2)}H_{k-1}^{(i)} - (H_{k-1}^{(i+1)})^2.$$

We recall that the relation given by (3.23) was obtained by Brezinski in [15] using the orthogonal polynomials, in our work we used the Sylvester identity which allows us to find (3.23) from the Hankel determinant's expression.
Using the relation (3.23), then applying the relation (3.16), we give Algorithm 2.

Algorithm 2 *(The computation of $Q_k^{(i)}$)*

1. *Initialisation* $Q_0^{(i)}(x) = 1, H_0^{(i)} = 1,\ H_1^{(i)} = c_i$
2. *For* $k = 2, \ldots, m$ *do*
   - *For* $i = 1, \ldots, p-2(k-1)$
     - *Compute* $H_k^{(i)} = \dfrac{1}{H_{k-2}^{(i+2)}}\left(H_{k-1}^{(i+2)}H_{k-1}^{(i)} - (H_{k-1}^{(i+1)})^2\right).$
   - *end*
   - *Put* $l = 1$
3. *For* $i = 1, \ldots, p+m-2$
   $l = l+1$
   - *For* $k = 2, \ldots, min(l, m)$
     - *Compute* $Q_k^{(i)}(x) = Q_{k-1}^{(i)}(x) - x\dfrac{H_k^{(i)}H_{k-1}^{(i+2)}}{H_k^{(i+1)}H_{k-1}^{(i+1)}}Q_{k-1}^{(i+1)}(x).$
   - *end*
4. *end*

Remarque 14 *Algorithm 2 allows us to calculate recursively the orthogonal polynomials by computing each Hankel determinant in* $\beta_k^{(i)}$ (3.17)*, but if we use the ROPA we get the* $\beta_k^{(i)}$ *recursively with the qd algorithm.*

3.4 Some Properties

In this section we will give some properties of the ROPA given in the previous section.

Proposition 11 *From the expressions* (3.9) *and* (3.17)*, we have*

1. $c(x^jG_k^{(i)}(0)) = 0,$ *for* $j = i, \ldots, i+k-1.$
2. $\beta_k^{(i)} = \dfrac{c(x^{i+k-1}Q_{k-1}^{(i)}(x))}{c(x^{i+k}Q_{k-1}^{(i+1)}(x))},$ *for* $i \geqslant 0$ *and* $k \geqslant 1.$
3. $\beta_k^{(i)} = \dfrac{e_k^{(i)}}{e_k^{(i+1)}}\beta_{k-1}^{(i)},$ *for* $i \geqslant 0$ *and* $k \geqslant 1.$

Proof.

1. We have

$$c(x^j G_k^{(i)}(x)) = c\left(\left(\left[\begin{array}{cccc|c} x^j & \dots & \dots & x^{j+k-1} & x^{j+k} \\ \hline c_i & c_{i+1} & \dots & c_{i+k-1} & c_{i+k} \\ c_{i+1} & c_{i+2} & \dots & c_{i+k} & c_{i+k+1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ c_{i+k-1} & c_{i+k} & \dots & c_{i+2k-2} & c_{i+2k-1} \end{array} \right] / D_k^{(i)} \right) \right),$$

for $j = i, i+1, \dots, i+k-1$ we will get two rows similar, then $c(x^j G_k^{(i)}(x))$ will be null.

2. We have $\beta_k^{(i)} = \dfrac{H_k^{(i)}}{H_{k-1}^{(i+1)}} \times \dfrac{H_{k-1}^{(i+2)}}{H_k^{(i+1)}}$, then

$$\frac{H_k^{(i)}}{H_{k-1}^{(i+1)}} = \left(\left[\begin{array}{c|cccc} c_i & c_{i+1} & \dots & \dots & c_{i+k-1} \\ c_{i+1} & c_{i+2} & \dots & \dots & c_{i+k} \\ c_{i+2} & c_{i+3} & \dots & \dots & c_{i+k+1} \\ \vdots & \vdots & & & \vdots \\ c_{i+k-2} & c_{i+k-1} & \dots & \dots & c_{i+2k-3} \\ \hline c_{i+k-1} & c_{i+k} & \dots & \dots & c_{i+2k-2} \end{array} \right] / D_{k-1}^{(i+1)} \right),$$

so using (2.28), we get

$$\frac{H_k^{(i)}}{H_{k-1}^{(i+1)}} = \left(\left[\begin{array}{c|cccc} c_{i+k-1} & c_{i+k} & \dots & \dots & c_{i+2k-2} \\ \hline c_i & c_{i+1} & \dots & \dots & c_{i+k-1} \\ c_{i+1} & c_{i+2} & \dots & \dots & c_{i+k} \\ c_{i+2} & c_{i+3} & \dots & \dots & c_{i+k+1} \\ \vdots & \vdots & & & \vdots \\ c_{i+k-2} & c_{i+k-1} & \dots & \dots & c_{i+2k-3} \end{array} \right] / D_{k-1}^{(i+1)} \right),$$

if we apply the functional c we obtain

$$\frac{H_k^{(i)}}{H_{k-1}^{(i+1)}} = c\left(\left[\begin{array}{c|cccc} x^{i+k-1} & x^{i+k} & \dots & \dots & x^{i+2k-2} \\ \hline c_i & c_{i+1} & \dots & \dots & c_{i+k-1} \\ c_{i+1} & c_{i+2} & \dots & \dots & c_{i+k} \\ c_{i+2} & c_{i+3} & \dots & \dots & c_{i+k+1} \\ \vdots & \vdots & & & \vdots \\ c_{i+k-2} & c_{i+k-1} & \dots & \dots & c_{i+2k-3} \end{array} \right] / D_{k-1}^{(i+1)} \right)$$

$$= c\left(x^{i+k-1} \left[\begin{array}{c|cccc} 1 & x & \dots & \dots & x^{k-1} \\ \hline c_i & c_{i+1} & \dots & \dots & c_{i+k-1} \\ c_{i+1} & c_{i+2} & \dots & \dots & c_{i+k} \\ c_{i+2} & c_{i+3} & \dots & \dots & c_{i+k+1} \\ \vdots & \vdots & & & \vdots \\ c_{i+k-2} & c_{i+k-1} & \dots & \dots & c_{i+2k-3} \end{array} \right] / D_{k-1}^{(i+1)} \right).$$

Therefore, we have $\dfrac{H_k^{(i)}}{H_{k-1}^{(i+1)}} = c(x^{i+k-1}Q_{k-1}^{(i)}(x))$, likewise we find that $\dfrac{H_{k-1}^{(i+2)}}{H_k^{(i+1)}} = (c(x^{i+k}Q_{k-1}^{(i+1)}(x)))^{-1}$.

3. We have

$$\beta_{k+1}^{(i)} = \frac{q_k^{(i+1)}}{q_{k+1}^{(i)}}\beta_k^{(i)},$$

and from the qd algorithm, we have

$$q_{k+1}^{(i)} = \frac{e_k^{(i+1)}}{e_k^{(i)}}q_k^{(i+1)},$$

we substitute the expression of $q_{k+1}^{(i)}$ in the expression of $\beta_{k+1}^{(i)}$, we obtain

$$\beta_k^{(i)} = \frac{e_k^{(i)}}{e_k^{(i+1)}}\beta_{k-1}^{(i)}.$$

3.5 The recursive orthogonal polynomial algorithm with auxiliary polynomial

In this section, we will use an auxiliary polynomial $g_k^{(i)}$ to find a recursive algorithm for computing the polynomials Q_k given by (3.4).
We recall the expression of Q_k

$$Q_k(x) = \left(\left[\begin{array}{c|cccc} 1 & x & \dots & \dots & x^k \\ \hline c_0 & c_1 & \dots & \dots & c_k \\ c_1 & c_2 & \dots & \dots & c_{k+1} \\ \vdots & \vdots & & & \vdots \\ c_{k-1} & c_k & \dots & \dots & c_{2k-1} \end{array}\right] / D_k^{(1)}\right).$$

If we apply the Sylvester identity, we get

$$Q_k(x) = Q_{k-1}(x) - \left(\left[\begin{array}{cccc|c} x & \dots & \dots & x^{k-1} & x^k \\ \hline c_1 & \dots & \dots & c_{k-1} & c_k \\ c_2 & \dots & \dots & c_k & c_{k+1} \\ \vdots & & & \vdots & \vdots \\ c_{k-1} & \dots & \dots & c_{2k-3} & c_{2k-2} \end{array}\right] / D_{k-1}^{(1)}\right)(D_k^{(1)}/D_{k-1}^{(1)})^{-1}(D_k^{(0)}/D_{k-1}^{(1)}). \tag{3.24}$$

For $k \geqslant 1$ and $i > k$, we set the following auxiliary polynomial :

$$g_k^{(i)}(x) = \left(\left[\begin{array}{c|cccccc} x^i & x & \dots & \dots & x^{k-1} & x^k \\ \hline c_i & c_1 & \dots & \dots & c_{k-1} & c_k \\ c_{i+1} & c_2 & \dots & \dots & c_k & c_{k+1} \\ \vdots & \vdots & & & & \vdots \\ c_{i+k-1} & c_k & \dots & \dots & c_{2k-2} & c_{2k-1} \end{array} \right] / D_k^{(1)} \right), \tag{3.25}$$

where $g_0^{(i)}(x) = x^i$.
We need the following lemma to compute recursively the polynomials Q_k.

Lemma 1 *Let $g_k^{(i)}$ be the polynomial defined by* (3.25)*, applying some properties of the Schur complement, we have*

$$(D_k^{(1)}/D_{k-1}^{(1)}) = c(x^{k-1} g_{k-1}^{(k)}(x)), \quad \textit{for} \quad k \geqslant 1. \tag{3.26}$$

Proof. Using Proposition 3, we have

$$(D_k^{(1)}/D_{k-1}^{(1)}) = \left(\left[\begin{array}{ccccc|c} c_1 & c_2 & c_3 & \dots & c_{k-1} & c_k \\ c_2 & c_3 & c_4 & \dots & c_k & c_{k+1} \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ c_{k-1} & c_k & c_{k+1} & \dots & c_{2k-3} & c_{2k-2} \\ \hline c_k & c_{k+1} & c_{k+2} & \dots & c_{2k-2} & c_{2k-1} \end{array} \right] / D_{k-1}^{(1)} \right)$$

$$= \left(\left[\begin{array}{c|cccc} c_{2k-1} & c_k & c_{k+1} & \dots & c_{2k-2} \\ \hline c_k & c_1 & c_2 & \dots & c_{k-1} \\ c_{k+1} & c_2 & c_3 & \dots & c_k \\ \vdots & \vdots & \vdots & & \vdots \\ c_{2k-2} & c_{k-1} & c_k & \dots & c_{2k-3} \end{array} \right] / D_{k-1}^{(1)} \right).$$

Using Proposition 4, we get

$$(D_k^{(1)}/D_{k-1}^{(1)}) = c(x^{k-1} \left(\left[\begin{array}{c|cccc} x^k & x & x^2 & \dots & x^{k-1} \\ \hline c_k & c_1 & c_2 & \dots & c_{k-1} \\ c_{k+1} & c_2 & c_3 & \dots & c_k \\ \vdots & \vdots & & & \vdots \\ c_{2k-2} & c_{k-1} & c_k & \dots & c_{2k-3} \end{array} \right] / D_{k-1}^{(1)} \right)) = c(x^{k-1} g_{k-1}^{(k)}(x)).$$

We substitute the expression given by (3.26) in (3.24), we get the following theorem :

Théorème 15 *For $k \geqslant 1$, the polynomials Q_k and Q_{k-1} satisfy the following recurrence relationship :*

$$Q_k(x) = Q_{k-1}(x) - c(x^{k-1} Q_{k-1}(x))(c(x^{k-1} g_{k-1}^{(k)}(x))^{-1} g_{k-1}^{(k)}(x),$$

with $Q_0(x) = 1$.

Proof.

We have

$$Q_k(x) = Q_{k-1}(x) - \left(\left[\begin{array}{cccc|c} x & \dots & \dots & x^{k-1} & x^k \\ \hline c_1 & \dots & \dots & c_{k-1} & c_k \\ c_2 & \dots & \dots & c_k & c_{k+1} \\ \vdots & & & & \vdots \\ c_{k-1} & \dots & \dots & c_{2k-3} & c_{2k-2} \end{array} \right] / D_{k-1}^{(1)} \right) (D_k^{(1)}/D_{k-1}^{(1)})^{-1} (D_k^{(0)}/D_{k-1}^{(1)}).$$

So using Proposition 3 and the relations given by (3.25) and (3.26), we obtain

$$Q_k(x) = Q_{k-1}(x) - c(x^{k-1}Q_{k-1}(x))(c(x^{k-1}g_{k-1}^{(k)}(x))^{-1}g_{k-1}^{(k)}(x). \tag{3.27}$$

Proposition 12 *For $k \geqslant 1$ and $i > k$, the polynomials $g_k^{(i)}$ verify the following recurrence relationship :*

$$g_k^{(i)}(x) = g_{k-1}^{(i)}(x) - c(x^{k-1}g_{k-1}^{(k)}(x))^{-1}c(x^{k-1}g_{k-1}^{(i)}(x))g_{k-1}^{(k)}(x), \tag{3.28}$$

with $g_0^{(i)} = x^i$.

Proof. The polynomial $g_k^{(i)}$ is partitioned as follows :

$$g_k^{(i)}(x) = \left(\left[\begin{array}{c|cccc|c} x^i & x & \dots & \dots & x^{k-1} & x^k \\ \hline c_i & c_1 & \dots & \dots & c_{k-1} & c_k \\ c_{i+1} & c_2 & \dots & \dots & c_k & c_{k+1} \\ \vdots & \vdots & & & \vdots & \vdots \\ c_{i+k-2} & c_{k-1} & \dots & \dots & c_{2k-3} & c_{2k-2} \\ \hline c_{i+k-1} & c_k & \dots & \dots & c_{2k-2} & c_{2k-1} \end{array} \right] / D_k^{(1)} \right),$$

we apply the Sylvester identity

$$
\begin{aligned}
g_k^{(i)}(x) = & \left(\left[\begin{array}{c|ccccc} x^i & x & \dots & \dots & x^{k-2} & x^{k-1} \\ \hline c_i & c_1 & \dots & \dots & c_{k-2} & c_{k-1} \\ c_{i+1} & c_2 & \dots & \dots & c_{k-1} & c_k \\ \vdots & \vdots & & & \vdots & \vdots \\ c_{i+k-1} & c_{k-1} & \dots & \dots & c_{2k-4} & c_{2k-3} \end{array} \right] / D_{k-1}^{(1)} \right) \\
& - \left(\left[\begin{array}{cccc|c} x & \dots & \dots & x^{k-1} & x^k \\ \hline c_1 & \dots & \dots & c_{k-1} & c_k \\ c_2 & \dots & \dots & c_k & c_{k+1} \\ \vdots & & & \vdots & \vdots \\ c_{k-1} & \dots & \dots & c_{2k-3} & c_{2k-2} \end{array} \right] / D_{k-1}^{(1)} \right) \\
& \times (D_k^{(1)}/D_{k-1}^{(1)})^{-1} \times \left(\left[\begin{array}{c|cccc} c_i & c_1 & \dots & \dots & c_{k-1} \\ c_{i+1} & c_2 & \dots & \dots & c_k \\ \vdots & \vdots & & & \vdots \\ c_{i+k-2} & c_{k-1} & \dots & \dots & c_{2k-3} \\ \hline c_{i+k-1} & c_k & \dots & \dots & c_{2k-2} \end{array} \right] / D_{k-1}^{(1)} \right).
\end{aligned}
$$

Using Proposition 3, Proposition 4 and the relations given by (3.25), (3.26) we get

$$g_k^{(i)}(x) = g_{k-1}^{(i)}(x) - c(x^{k-1} g_{k-1}^{(k)}(x))^{-1} c(x^{k-1} g_{k-1}^{(i)}(x)) g_{k-1}^{(k)}(x).$$

From the relation given by (3.28), we compute recursively the polynomials $g_k^{(i)}$ as follows :

Algorithm 3 *(The computation of $g_k^{(i)}$)*

1. *For* $i = 1, \dots, n$
2. *Compute* $g_0^{(i)}(x) = x^i$.
   - *For* $k = 1, \dots, i-1$
   - *Compute* $g_k^{(i)}(x) = g_{k-1}^{(i)}(x) - c(x^{k-1}g_{k-1}^{(k)}(x))^{-1}c(x^{k-1}g_{k-1}^{(i)}(x))g_{k-1}^{(k)}(x)$.
   - *end*
3. *end*

Then, applying Algorithm 3 and the relation given by (**??**), we get Algorithm 4.

Algorithm 4 *(The computation of Q_k)*

1. *Choose* $Q_0(x) = 1$.
2. *For* $i = 1, \dots, n$
   - *Compute* $g_0^{(i)}(x) = x^i$.
3. *end*
4. *For* $k = 1, \dots, i-1$
   - *Compute* $g_k^{(i)}(x) = g_{k-1}^{(i)}(x) - c(x^{k-1}g_{k-1}^{(k)}(x))^{-1}c(x^{k-1}g_{k-1}^{(i)}(x))g_{k-1}^{(k)}(x)$.
   - *Compute* $Q_k(x) = Q_{k-1}(x) - c(x^{k-1}Q_{k-1}(x))(c(x^{k-1}g_{k-1}^{(k)}(x))^{-1}g_{k-1}^{(k)}(x)$.
5. *end*

Remarque 16 *Algorithm 4 allows us to calculate the orthogonal polynomials Q_m of the same family of orthogonal polynomials $\{Q_m\}_{m\in\mathbb{N}}$, but the ROPA give a relation between two families of orthogonal polynomials $\{Q_m^{(i)}\}_{m\in\mathbb{N}}$ and $\{Q_m^{(i+1)}\}_{m\in\mathbb{N}}$.*

Example 2 *We choose in this example to calculate the polynomials Q_k, for $k = 1, 2$.*

Let the function

$$f(x) = \sum_{i=0}^{\infty} \frac{1}{i!} x^i,$$

with the initial settings $g_0^{(i)}(x) = x^i$ and $Q_0(x) = 1$.

For computing the polynomial Q_2, we should calculate the polynomial $g_1^{(2)}$ recursively from Algorithm 3.

$$g_1^{(i)}(x) = g_0^{(i)}(x) - c(x^0 g_0^{(1)}(x))^{-1} c(x^0 g_0^{(i)}(x)) g_0^{(1)}(x) = x^i - \frac{c_i}{c_1} x.$$

We finally get

$$Q_1(x) = 1 - \frac{c_0}{c_1} x = 1 - x,$$

and

$$Q_2(x) = 1 - \frac{c_0 c_3 - c_1 c_2}{c_1 c_3 - c_2^2} x - \frac{c_0 c_2 - c_1^2}{c_1 c_3 - c_2^2} x^2 = 1 - 4x + 6x^2.$$

4 Computing recursive quasiorthogonal polynomials

This chapter, is interested in exploring efficient recursive algorithms to compute the quasi-orthogonal polynomials as defined in [48]. We provide new algebraic properties of the adjacent families of quasi-orthogonal polynomials. We show that they can be generated by four-term recurrence relations whose coefficients are written in terms of lower triangular block determinants. The algorithms presented generalize algorithms discussed in [50], which compute Hankel determinants, to allow the computation of lower triangular block determinants.The derived algorithms allow us to connect various quasi-orthogonal polynomials in the Padé table.

4.1 Recurrence relations

In this section, we give a short-term recurrence relation of quasi-orthogonal polynomials. Based on this short-term relation, we define the Padé table with different values of p.

4.1.1 Explicit expression of quasi-orthogonal polynomials

Consider the polynomials

$$Q_{(m,n)_1}(x) = \sum_{i=0}^{m} q_i^1 x^i \text{ and } Q_{(m,n)_2}(x) = \sum_{i=0}^{n} q_i^2 x^i, \text{ with } q_0^1 = 0.$$

These polynomials can be expressed as a Schur complement. We will prove this in the following Proposition.

Proposition 17 *Assume that the matrix $D_{m,n}^{(1)}$ is nonsingular, then the polynomials $Q_{(m,n)_1}$ and $Q_{(m,n)_2}$ can be expressed as a Schur complement as follows*

$$Q_{(m,n)_1}(x) = (\left[\begin{array}{c|cccccccc} 1 & x & \dots & \dots & x^{m-1} & x^m & 0 & \dots & 0 & 0 \\ \hline c_0^1 & c_1^1 & c_2^1 & \dots & c_{m-1}^1 & c_m^1 & 0 & \dots & 0 & 0 \\ c_0^2 & c_1^2 & c_2^2 & \dots & c_{m-1}^2 & c_m^2 & c_m^1 & \dots & c_{m-n+1}^1 & c_{m-n}^1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\ c_{r-1}^1 & c_r^1 & c_{r+1}^1 & \dots & c_{m+r-2}^1 & c_{m+r-1}^1 & 0 & \dots & 0 & 0 \\ c_{r-1}^2 & c_r^2 & c_{r+1}^2 & \dots & c_{m+r-2}^2 & c_{m+r-1}^2 & c_{m+r-1}^1 & \dots & c_{m-n+r}^1 & c_{m-n+r-1}^1 \end{array}\right] / D_{m,n}^{(1)})$$

and

$$Q_{(m,n)_2}(x) = (\left[\begin{array}{c|cccccccc} 0 & 0 & 0 & \dots & \dots & 0 & x^n & \dots & x & 1 \\ \hline c_0^1 & c_1^1 & c_2^1 & \dots & c_{m-1}^1 & c_m^1 & 0 & \dots & 0 & 0 \\ c_0^2 & c_1^2 & c_2^2 & \dots & c_{m-1}^2 & c_m^2 & c_m^1 & \dots & c_{m-n+1}^1 & c_{m-n}^1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\ c_{r-1}^1 & c_r^1 & c_{r+1}^1 & \dots & c_{m+r-2}^1 & c_{m+r-1}^1 & 0 & \dots & 0 & 0 \\ c_{r-1}^2 & c_r^2 & c_{r+1}^2 & \dots & c_{m+r-2}^2 & c_{m+r-1}^2 & c_{m+r-1}^1 & \dots & c_{m-n+r}^1 & c_{m-n+r-1}^1 \end{array}\right] / D_{m,n}^{(1)}).$$

Proof.

The quasi-orthogonality condition in (2.17) yields the following system

$$\begin{bmatrix} c_1^1 & c_2^1 & \dots & c_{m-1}^1 & c_m^1 & 0 & \dots & 0 \\ c_1^2 & c_2^2 & \dots & c_{m-1}^2 & c_m^2 & c_m^1 & \dots & c_{m-n}^1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ c_r^1 & c_{r+1}^1 & \dots & c_{m+r-2}^1 & c_{m+r-1}^1 & 0 & \dots & 0 \\ c_r^2 & c_{r+1}^2 & \dots & c_{m+r-2}^2 & c_{m+r-1}^2 & c_{m+r-1}^1 & \dots & c_{m-n+r-1}^1 \end{bmatrix} \begin{bmatrix} q_1^1 \\ \vdots \\ q_{m-1}^1 \\ q_m^1 \\ q_n^2 \\ \vdots \\ q_0^2 \end{bmatrix} = \begin{bmatrix} -c_0^1 \\ -c_0^2 \\ \vdots \\ \vdots \\ -c_{r-1}^1 \\ -c_{r-1}^2 \end{bmatrix}. \tag{4.1}$$

Under the condition $|D_{m,n}^{(1)}| \neq 0$ and $q_0^{(1)} = 1$, the coefficients $q_1^{(1)}, \dots, q_m^{(1)}, q_0^{(2)}, \dots, q_n^{(2)}$ are given by

$$\begin{bmatrix} q_1^1 \\ \vdots \\ q_{m-1}^1 \\ q_m^1 \\ q_n^2 \\ \vdots \\ q_0^2 \end{bmatrix} = -(D_{m,n}^{(1)})^{-1} \begin{bmatrix} -c_0^1 \\ -c_0^2 \\ \vdots \\ \vdots \\ -c_{r-1}^1 \\ -c_{r-1}^2 \end{bmatrix}. \tag{4.2}$$

then

$$Q_{(m,n)_1}(x) = 1 - \begin{bmatrix} x & \dots & x^{m-1} & x^m & 0 & \dots & 0 \end{bmatrix} (D_{m,n}^{(1)})^{-1} \begin{bmatrix} c_0^1 \\ c_0^2 \\ \vdots \\ \vdots \\ c_{r-1}^1 \\ c_{r-1}^2 \end{bmatrix},$$

and

$$Q_{(m,n)_2}(x) = 0 - \begin{bmatrix} 0 & 0 & \dots & 0 & x^n & \dots & x & 1 \end{bmatrix} (D_{m,n}^{(1)})^{-1} \begin{bmatrix} c_0^1 \\ c_0^2 \\ \vdots \\ \vdots \\ c_{r-1}^1 \\ c_{r-1}^2 \end{bmatrix}.$$

From (2.26) and (4.2), the polynomials $Q_{(m,n)_1}, Q_{(m,n)_2}$ are written as a Schur complement.

4.1.2 Formulation of recurrence relations

Before describing the short-term recurrence relation of the quasi-orthogonal polynomials of the form (2.8), we need to define some auxiliary polynomials that will be used in this section. We define the matrix polynomial $Q^{(i)}_{m',n'}$ of this form

$$Q^{(i)}_{m',n'} = \begin{bmatrix} Q^{(i)}_{(m',n')_1} & 0 \\ Q^{(i)}_{(m',n')_2} & Q^{(i)}_{(m',n')_1} \end{bmatrix}, \tag{4.3}$$

where $Q_{(m',n')_1}(x) = \sum_{k=0}^{m'} q^1_k x^k$ and $Q_{(m',n')_2}(x) = \sum_{k=0}^{n'} q^2_k x^k$, with $q^1_0 = 1$, $\deg(Q^{(i)}_{(m',n')_1}) = m'$, and $\deg(Q^{(i)}_{(m',n')_2}) = n'$. Here m' and n' are two integers such that $m' - n' = 2p'$, for $p' \in \mathbb{N}$.

The matrix polynomial $Q^{(i)}_{m',n'}$ is defined to be quasi-orthogonal with respect to the function $C^{(i)}$ as given by (2.24). This leads to the following equations

$$\begin{cases} C^{(i)}(x^j Q_{m',n'}(x)) = 0_{\mathbb{R}^2}, \quad 0 \leqslant j \leqslant r-2, \\ c^{(i)}(x^{r-1} Q_{m',n'}(x)) = 0 \end{cases} \tag{4.4}$$

with $r = \dfrac{m'+n'}{2}$. Using the linearity of the function $C^{(i)}$, equations (4.4) can formulated be as the linear system

$$\begin{bmatrix} c^1_{i+1} & \cdots & c^1_{i+m'-1} & c^1_{i+m'} & 0 & \cdots & 0 \\ c^2_{i+1} & \cdots & c^2_{i+m'-1} & c^2_{i+m'} & c^1_{i+m'} & \cdots & c^1_{i+m'-n'} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ c^1_{i+r-2} & \cdots & c^1_{i+m'+r-3} & c^1_{i+m'+r-2} & 0 & \cdots & 0 \\ c^2_{i+r-2} & \cdots & c^2_{i+m'+r-3} & c^2_{i+m'+r-2} & c^1_{i+m'+r-2} & \cdots & c^1_{i+m'-n'+r-2} \\ c^1_{i+r-1} & \cdots & c^1_{i+m'+r-2} & c^1_{i+m'+r-1} & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} q^1_1 \\ \vdots \\ q^1_{m'-1} \\ q^1_{m'} \\ q^2_{n'} \\ \vdots \\ q^2_0 \end{bmatrix} = \begin{bmatrix} -c^1_i \\ -c^2_i \\ \vdots \\ -c^1_{i+r-2} \\ -c^2_{i+r-2} \\ -c^1_{i+r-1} \end{bmatrix}. \tag{4.5}$$

This shows that the existence and uniqueness of $Q^{(i)}_{m',n'}$ holds when $D^{(i)}_{m',n'}$ is nonsingular, where

$$D^{(i)}_{m',n'} = \begin{bmatrix} c^1_i & c^1_{i+1} & \cdots & c^1_{i+m'-1} & 0 & \cdots & 0 \\ c^2_i & c^2_{i+1} & \cdots & c^2_{i+m'-1} & c^2_{i+m'-1} & \cdots & c_{i+m'-n'-1} \\ \vdots & & & \vdots & & & \vdots \\ c^1_{i+r-1} & c^1_{i+r} & \cdots & c^1_{i+m'+r-2} & 0 & \cdots & 0 \\ c^2_{i+r-1} & c^2_{i+r} & \cdots & c^2_{i+m'+r-2} & c^1_{i+m'+r-2} & \cdots & c_{i+m'-n'+r-2} \\ c^1_{i+r} & c^1_{i+r+1} & \cdots & c^1_{i+m'+r-1} & 0 & \cdots & 0 \end{bmatrix}.$$

Proposition 18 *Assume that the $D^{(i+1)}_{m',n'}$ is nonsingular, then the polynomials $Q^{(i)}_{(m',n')_1}(x)$ and $Q^{(i)}_{(m',n')_2}(x)$ exist, and admit the following expressions*

$$Q^{(i)}_{(m',n')_1}(x) =$$

$$\left(\left[\begin{array}{c|cccccccc}
1 & x & \dots & \dots & x^{m'-1} & x^{m'} & 0 & \dots & 0 \\
\hline
c^1_i & c^1_{i+1} & c^1_{i+2} & \dots & c^1_{i+m'-1} & c^1_{i+m'} & 0 & \dots & 0 \\
c^2_i & c^2_{i+1} & c^2_{i+2} & \dots & c^2_{i+m'-1} & c^2_{i+m'} & c^1_{i+m'} & \dots & c^1_{i+m'-n'+1} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\
c^1_{i+r-2} & c^1_{i+r-1} & c^1_{i+r} & \dots & c^1_{i+m'+r-3} & c^1_{i+m'+r-2} & 0 & \dots & 0 \\
c^2_{i+r-2} & c^2_{i+r-1} & c^2_{i+r} & \dots & c^2_{i+m'+r-3} & c^2_{i+m'+r-2} & c^1_{i+m'+r-2} & \dots & c^1_{i+m'-n'+r-1} \\
c^1_{i+r-1} & c^1_{i+r} & c^1_{r+1} & \dots & c^1_{i+m'+r-2} & c^1_{i+m'+r-1} & 0 & \dots & 0
\end{array}\right] / D^{(i+1)}_{m',n'}\right)$$

and

$$Q^{(i)}_{(m',n')_2}(x) =$$

$$\left(\left[\begin{array}{c|cccccccc}
0 & 0 & \dots & \dots & 0 & 0 & x^{n'-1} & \dots & 1 \\
\hline
c^1_i & c^1_{i+1} & c^1_{i+2} & \dots & c^1_{i+m'-1} & c^1_{i+m'} & 0 & \dots & 0 \\
c^2_i & c^2_{i+1} & c^2_{i+2} & \dots & c^2_{i+m'-1} & c^2_{i+m'} & c^1_{i+m'} & \dots & c^1_{i+m'-n'+1} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\
c^1_{i+r-2} & c^1_{i+r-1} & c^1_{i+r} & \dots & c^1_{i+m'+r-3} & c^1_{i+m'+r-2} & 0 & \dots & 0 \\
c^2_{i+r-2} & c^2_{i+r-1} & c^2_{i+r} & \dots & c^2_{i+m'+r-3} & c^2_{i+m'+r-2} & c^1_{i+m'+r-2} & \dots & c^1_{i+m'-n'+r-1} \\
c^1_{i+r-1} & c^1_{i+r} & c^1_{i+r+1} & \dots & c^1_{i+m'+r-2} & c^1_{i+m'+r-1} & 0 & \dots & 0
\end{array}\right] / D^{(i+1)}_{m',n'}\right)$$

Proof.

The proof is analogous to the proof of Proposition 17. The system (4.5) is used.

In the following lemma, we show that the quasi-orthogonal polynomials of the form (2.8) can be computed with short-term recurrence relation in terms of the auxiliary polynomials as defined by (4.3).

Lemme 19 *For $m \geqslant 1, n \geqslant 1$ such that $m - n = 2p + 1$, we have the following relation*

$$Q_{(m,n)_1}(x) = Q_{(m,n-1)_1}(x) - x\beta_{m,n} Q^{(1)}_{(m-1,n)_1}(x). \tag{4.6}$$

and

$$Q_{(m,n)_2}(x) = xQ_{(m,n-1)_2}(x) - \beta_{m,n} Q^{(1)}_{(m-1,n)_2}(x). \tag{4.7}$$

with $Q_{(m,n)_1}(0) = 1$ and $\beta_{m,n} = \dfrac{|D^{(0)}_{m+1,n-1}||D^{(2)}_{m-1,n}|}{|D^{(1)}_{m,n}||D^{(1)}_{m,n-1}|}$.

Proof.

We first show (4.6), the polynomial $Q_{(m,n)_1}$ can be decomposed as

$$Q_{(m,n)_1}(x) =$$
$$\left(\left[\begin{array}{c|cccccccc|c}
1 & x & \dots & \dots & x^{m-1} & x^m & 0 & \dots & 0 & 0 \\
\hline
c_0^1 & c_1^1 & c_2^1 & \dots & c_{m-1}^1 & c_m^1 & 0 & \dots & 0 & 0 \\
c_0^2 & c_1^2 & c_2^2 & \dots & c_{m-1}^2 & c_m^2 & c_m^1 & \dots & c_{m-n+1}^1 & c_{m-n}^1 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\
c_{r-2}^1 & c_{r-1}^1 & c_r^1 & \dots & c_{m+r-3}^1 & c_{m+r-2}^1 & 0 & \dots & 0 & 0 \\
c_{r-2}^2 & c_{r-1}^2 & c_r^2 & \dots & c_{m+r-3}^2 & c_{m+r-2}^2 & c_{m+r-2}^1 & \dots & c_{m-n+r-1}^1 & c_{m-n+r-2}^1 \\
c_{r-1}^1 & c_r^1 & c_{r+1}^1 & \dots & c_{m+r-2}^1 & c_{m+r-1}^1 & 0 & \dots & 0 & 0 \\
\hline
c_{r-1}^2 & c_r^2 & c_{r+1}^2 & \dots & c_{m+r-2}^2 & c_{m+r-1}^2 & c_{m+r-1}^1 & \dots & c_{m-n+r}^1 & c_{m-n+r-1}^1
\end{array}\right] / D_{m,n}^{(1)}\right).$$

We apply the Sylvester identity (2.30) to $Q_{(m,n)_1}$, we obtain

$$Q_{(m,n)_1}(x) = \left(\left[\begin{array}{c|cccccccc}
1 & x & \dots & \dots & x^{m-1} & x^m & 0 & \dots & 0 \\
\hline
c_0^1 & c_1^1 & c_2^1 & \dots & c_{m-1}^1 & c_m^1 & 0 & \dots & 0 \\
c_0^2 & c_1^2 & c_2^2 & \dots & c_{m-1}^2 & c_m^2 & c_m^1 & \dots & c_{m-n+1}^1 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\
c_{r-2}^1 & c_{r-1}^1 & c_r^1 & \dots & c_{m+r-3}^1 & c_{m+r-2}^1 & 0 & \dots & 0 \\
c_{r-2}^2 & c_{r-1}^2 & c_r^2 & \dots & c_{m+r-3}^2 & c_{m+r-2}^2 & c_{m+r-2}^1 & \dots & c_{m-n+r-1}^1 \\
c_{r-1}^1 & c_r^1 & c_{r+1}^1 & \dots & c_{m+r-2}^1 & c_{m+r-1}^1 & 0 & \dots & 0
\end{array}\right] / D_{m,n-1}^{(1)}\right)$$
$$-\left(\left[\begin{array}{cccccccc|c}
x & \dots & \dots & x^{m-1} & x^m & 0 & \dots & 0 & 0 \\
\hline
c_1^1 & c_2^1 & \dots & c_{m-1}^1 & c_m^1 & 0 & \dots & 0 & 0 \\
c_1^2 & c_2^2 & \dots & c_{m-1}^2 & c_m^2 & c_m^1 & \dots & c_{m-n+1}^1 & c_{m-n}^1 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\
c_{r-1}^1 & c_r^1 & \dots & c_{m+r-3}^1 & c_{m+r-2}^1 & 0 & \dots & 0 & 0 \\
c_{r-1}^2 & c_r^2 & \dots & c_{m+r-3}^2 & c_{m+r-2}^2 & c_{m+r-2}^1 & \dots & c_{m-n+r-1}^1 & c_{m-n+r-2}^1 \\
c_r^1 & c_{r+1}^1 & \dots & c_{m+r-2}^1 & c_{m+r-1}^1 & 0 & \dots & 0 & 0
\end{array}\right] / D_{m,n-1}^{(1)}\right)$$
$$\times (D_{m,n}^{(1)}/D_{m,n-1}^{(1)})^{-1}(D_{m+1,n-1}^{(0)}/D_{m,n-1}^{(1)}).$$

An application of Proposition 3 gives

$$\left(\left[\begin{array}{cccccccc|c}
x & \dots & \dots & x^{m-1} & x^m & 0 & \dots & 0 & 0 \\
\hline
c_1^1 & c_2^1 & \dots & c_{m-1}^1 & c_m^1 & 0 & \dots & 0 & 0 \\
c_1^2 & c_2^2 & \dots & c_{m-1}^2 & c_m^2 & c_m^1 & \dots & c_{m-n+1}^1 & c_{m-n}^1 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\
c_{r-1}^1 & c_r^1 & \dots & c_{m+r-3}^1 & c_{m+r-2}^1 & 0 & \dots & 0 & 0 \\
c_{r-1}^2 & c_r^2 & \dots & c_{m+r-3}^2 & c_{m+r-2}^2 & c_{m+r-2}^1 & \dots & c_{m-n+r-1}^1 & c_{m-n+r-2}^1 \\
c_r^1 & c_{r+1}^1 & \dots & c_{m+r-2}^1 & c_{m+r-1}^1 & 0 & \dots & 0 & 0
\end{array}\right] / D_{m,n-1}^{(1)}\right) =$$

$$\left(\left[\begin{array}{c|cccccccc} x & \dots & \dots & x^{m-1} & x^m & 0 & \dots & 0 & 0 \\ \hline c_1^1 & c_2^1 & \dots & c_{m-1}^1 & c_m^1 & 0 & \dots & 0 & 0 \\ c_1^2 & c_2^2 & \dots & c_{m-1}^2 & c_m^2 & c_m^1 & \dots & c_{m-n+1}^1 & c_{m-n}^1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\ c_{r-1}^1 & c_r^1 & \dots & c_{m+r-3}^1 & c_{m+r-2}^1 & 0 & \dots & 0 & 0 \\ c_{r-1}^2 & c_r^2 & \dots & c_{m+r-3}^2 & c_{m+r-2}^2 & c_{m+r-2}^1 & \dots & c_{m-n+r-1}^1 & c_{m-n+r-2}^1 \\ c_r^1 & c_{r+1}^1 & \dots & c_{m+r-2}^1 & c_{m+r-1}^1 & 0 & \dots & 0 & 0 \end{array}\right] /D_{m-1,n}^{(2)}\right)\times$$

$$\frac{|D_{m-1,n}^{(2)}|}{|D_{m,n-1}^{(1)}|}.$$

It follows that

$$Q_{(m,n)_1}(x) = \left(\left[\begin{array}{c|cccccccc} 1 & x & \dots & \dots & x^{m-1} & x^m & 0 & \dots & 0 \\ \hline c_0^1 & c_1^1 & c_2^1 & \dots & c_{m-1}^1 & c_m^1 & 0 & \dots & 0 \\ c_0^2 & c_1^2 & c_2^2 & \dots & c_{m-1}^2 & c_m^2 & c_m^1 & \dots & c_{m-n+1}^1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\ c_{r-2}^1 & c_{r-1}^1 & c_r^1 & \dots & c_{m+r-3}^1 & c_{m+r-2}^1 & 0 & \dots & 0 \\ c_{r-2}^2 & c_{r-1}^2 & c_r^2 & \dots & c_{m+r-3}^2 & c_{m+r-2}^2 & c_{m+r-2}^1 & \dots & c_{m-n+r-1}^1 \\ c_{r-1}^1 & c_r^1 & c_{r+1}^1 & \dots & c_{m+r-2}^1 & c_{m+r-1}^1 & 0 & \dots & 0 \end{array}\right] /D_{m,n-1}^{(1)}\right)$$

$$- \left(\left[\begin{array}{cccccccc|c} x & \dots & \dots & x^{m-1} & x^m & 0 & \dots & 0 & 0 \\ \hline c_1^1 & c_2^1 & \dots & c_{m-1}^1 & c_m^1 & 0 & \dots & 0 & 0 \\ c_1^2 & c_2^2 & \dots & c_{m-1}^2 & c_m^2 & c_m^1 & \dots & c_{m-n+1}^1 & c_{m-n}^1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\ c_{r-1}^1 & c_r^1 & \dots & c_{m+r-3}^1 & c_{m+r-2}^1 & 0 & \dots & 0 & 0 \\ c_{r-1}^2 & c_r^2 & \dots & c_{m+r-3}^2 & c_{m+r-2}^2 & c_{m+r-2}^1 & \dots & c_{m-n+r-1}^1 & c_{m-n+r-2}^1 \\ c_r^1 & c_{r+1}^1 & \dots & c_{m+r-2}^1 & c_{m+r-1}^1 & 0 & \dots & 0 & 0 \end{array}\right] /D_{m,n-1}^{(1)}\right)$$

$$\times \left(D_{m,n}^{(1)}/D_{m,n-1}^{(1)}\right)^{-1}\left(D_{m+1,n-1}^{(0)}/D_{m,n-1}^{(1)}\right).$$

Finally, we get

$$Q_{(m,n)_1}(x) = Q_{(m,n-1)_1}(x) - x\frac{|D_{m+1,n-1}^{(0)}||D_{m-1,n}^{(2)}|}{|D_{m,n}^{(1)}||D_{m,n-1}^{(1)}|}Q_{(m-1,n)_1}^{(1)}(x).$$

The relation (4.7) can be shown in a similar way.
We have the following theorem

Théorème 20 *For $m > n \geqslant 1$, $p > 0$ such that $m - n = 2p + 1$, the polynomials $Q_{(m,n)_1}(x)$ and $Q_{(m,n)_2}(x)$ satisfy the following relations*

$$Q_{(m,n)_1}(x) = Q_{(m,n-2)_1}(x) - x(\beta_{m,n} + \beta_{m,n-1})Q_{(m-1,n-1)_1}(x) + x^2\beta_{m,n}\beta_{m-1,n}^{(1)})Q_{(m-2,n)_1}(x) \tag{4.8}$$

and

$$Q_{(m,n)_2}(x) = x^2 Q_{(m,n-2)_2}(x) - x(\beta_{m,n}+\beta_{m,n-1}) Q^{(1)}_{(m-1,n-1)_2}(x) + \beta_{m,n}\beta^{(1)}_{m-1,n} Q^{(2)}_{(m-2,n)_2}(x) \tag{4.9}$$

with $Q_{(m,0)_1}, Q_{(m,-1)_1}$ *are two polynomials of degree* m*,*

$$Q_{(0,-1)_1}(x) = 1, \quad Q_{(m,-1)_2} = 0, \quad Q_{(m,0)_2} = \frac{|D_{m+1,-1}|}{|D^{(1)}_{m,0}|},$$

$$\beta_{m,n} = \frac{|D^{(0)}_{m+1,n-1}||D^{(2)}_{m-1,n}|}{|D^{(1)}_{m,n}||D^{(1)}_{m,n-1}|}, \quad \beta^{(1)}_{m,n} = \frac{|D^{(1)}_{m+1,n-1}||D^{(3)}_{m-1,n}|}{|D^{(2)}_{m,n}||D^{(2)}_{m,n-1}|}.$$

$D^{(i)}_{m,n}$ *is given by* (2.19)*. Moreover, the polynomials* $Q_{(m,0)_1}, Q_{(m,-1)_1}$ *are computed using the following relations*

$$Q_{(m,-1)_1}(x) = Q_{(m-2,-1)_1}(x) - x(\beta_{m,-1}+\beta_{m-1,-1}) Q^{(1)}_{(m-2,-1)_1}(x) + \beta_{m,-1}\beta^{(1)}_{m-1,-1} Q^{(2)}_{(m-2,-1)_1},$$

$$Q_{(m,0)_1}(x) = Q_{(m-1,-1)_1}(x) - x(\beta_{m,0}+\beta_{m,-1}) Q^{(1)}_{(m-1,-1)_1}(x) + x^2 \beta^{(1)}_{m-1,0}\beta_{m,0} Q^{(2)}_{(m-2,0)_1}(x),$$

where $Q^{(i)}_{(m,0)_2} = \dfrac{|D_{m+1,-1}|}{|D^{(1)}_{m,0}|}$, $\beta_{m,-1} = \dfrac{|D_{m,-1}||D^{(2)}_{m-1,-1}|}{|D^{(1)}_{m,-1}||D^{(1)}_{m-1,-1}|}$ *and* $\beta^{(1)}_{m,-1} = \dfrac{|D^{(1)}_{m,-1}||D^{(3)}_{m-1,-1}|}{|D^{(2)}_{m,-1}||D^{(2)}_{m-1,-1}|}$.

Proof.

We apply the Sylvester identity to the polynomials $Q_{m,n-1}$ and $Q^{(1)}_{m-1,n}$ and the same techniques as the proof of the lemma 19. Then

$$Q_{(m,n-1)_1}(x) = Q_{(m,n-2)_1}(x) - x\beta_{m,n-1} Q^{(1)}_{(m-1,n-1)_1}(x) \tag{4.10}$$

$$Q_{(m,n-1)_2}(x) = xQ_{(m,n-2)_2}(x) - \beta_{m,n-1} Q^{(1)}_{(m-1,n-1)_2}(x) \tag{4.11}$$

and

$$Q^{(1)}_{(m-1,n)_1}(x) = Q^{(1)}_{(m-1,n-1)_1} - x\beta_{m-1,n} Q^{(2)}_{(m-2,n)_1}(x) \tag{4.12}$$

$$Q^{(1)}_{(m-1,n)_2}(x) = xQ^{(1)}_{(m-1,n-1)_2} - \beta_{m-1,n} Q^{(2)}_{(m-2,n)_2}(x) \tag{4.13}$$

(4.8) and (4.9) relations can be established by substituting (4.10) and (3.16) with (4.6) and also (4.11), (4.13) with the expression of $Q_{(m,n)_2}$ given by (4.7).
We now state the following results, which give the short-term recurrence relations to compute the polynomials $Q^{(i)}_{(m,n)_1}(x)$ and $Q^{(i)}_{(m,n)_2}(x)$, for $i \geqslant 0$.

Théorème 21 *Let* $Q^{(i)}_{(m,n)_1}(x)$ *and* $Q^{(i)}_{(m,n)_2}(x)$ *be two polynomials as given by* (2.22)*. For* $i \geqslant 0$*,* $m > n \geqslant 1$ *,* $p > 0$ *such that* $m - n = 2p + 1$*.* $Q^{(i)}_{(m,n)_1}(x)$ *and* $Q^{(i)}_{(m,n)_2}(x)$

are computed as follows

$$\begin{aligned} Q^{(i)}_{(m,n)_1}(x) &= Q^{(i)}_{(m,n-2)_1}(x) - x(\beta^{(i)}_{m,n} + \beta^{(i)}_{m,n-1})Q^{(i+1)}_{(m-1,n-1)_1}(x) \\ &+ x^2\beta^{(i)}_{m,n}\beta^{(i+1)}_{m-1,n}Q^{(i+2)}_{(m-2,n)_1}(x) \end{aligned} \tag{4.14}$$

$$\begin{aligned} Q^{(i)}_{(m,n)_2}(x) &= x^2 Q^{(i)}_{(m,n-2)_2}(x) - x(\beta^{(i)}_{m,n} + \beta^{(i)}_{m,n-1})Q^{(i+1)}_{(m-1,n-1)_2}(x) \\ &+ \beta^{(i)}_{m,n}\beta^{(i+1)}_{m-1,n}Q^{(i+2)}_{(m-2,n)_2}(x) \end{aligned} \tag{4.15}$$

with $Q^{(i)}_{(m,0)_1}, Q^{(i)}_{(m,-1)_1}$ *are two polynomials of degree* m*,*

$$Q^{(i)}_{(0,-1)_1}(x) = 1, \quad Q^{(i)}_{(m,-1)_2} = 0, \quad Q^{(i)}_{(m,0)_2} = \frac{|D^{(i)}_{m+1,-1}|}{|D^{(i+1)}_{m,0}|},$$

$$\beta^{(i)}_{m,n} = \frac{|D^{(i)}_{m+1,n-1}||D^{(i+2)}_{m-1,n}|}{|D^{(i+1)}_{m,n}||D^{(i+1)}_{m,n-1}|}.$$

Furthermore, the polynomials $Q^{(i)}_{(m,-1)_1}(x)$ *and* $Q^{(i)}_{(m,0)_2}$ *are computed as follows*

$$\begin{aligned} Q^{(i)}_{(m,-1)_1}(x) &= Q^{(i)}_{(m-2,-1)_1}(x) - x(\beta^{(i)}_{m,-1} + \beta^{(i)}_{m-1,-1})Q^{(i+1)}_{(m-2,-1)_1}(x) \\ &+ \beta^{(i)}_{m,-1}\beta^{(i+1)}_{m-1,-1}Q^{(i+2)}_{(m-2,-1)_1}, \end{aligned} \tag{4.16}$$

$$\begin{aligned} Q^{(i)}_{(m,0)_1}(x) &= Q^{(i)}_{(m-1,-1)_1}(x) - x(\beta^{(i)}_{m,0} + \beta^{(i)}_{m,-1})Q^{(i+1)}_{(m-1,-1)_1}(x) \\ &+ x^2\beta^{(i+1)}_{m-1,0}\beta^{(i)}_{m,0}Q^{(i+2)}_{(m-2,0)_1}(x), \end{aligned} \tag{4.17}$$

with $\beta^{(i)}_{m,-1} = \dfrac{|D^{(i)}_{m,-1}||D^{(i+2)}_{m-1,-1}|}{|D^{(i+1)}_{m,-1}||D^{(i+1)}_{m-1,-1}|}$,

Proof.
The relations (4.14) and (4.15) are proved using the same techniques as in Theorem 20.

In the following, we define the Padé table, including the adjacent families of quasi-orthogonal polynomials $Q^{(i)}_{(m,n)}$ of the form (2.8), where $m - n = 2p + 1$, for some $p \in \mathbb{N}$ and $i = 0, 1, \ldots$. We display these polynomials in a table denoted by T_i for each integer i. Denote by (m, n) the i-th adjacent quasi-orthogonal polynomials $Q^{(i)}_{(m,n)}$ in this table.

$$\begin{array}{llllll} T_i = & (1,0) & & & & \\ & (3,0) & (2,1) & & & \\ & (5,0) & (4,1) & (3,2) & & \\ & (7,0) & (6,1) & (5,2) & (4,3) & \\ & (9,0) & (8,1) & (7,2) & (6,3) & (5,4) \\ & \cdots & \cdots & \cdots & \cdots & \cdots \end{array}$$

This table is made to fill the first diagonal with the polynomials $Q^{(i)}_{(n+1,n)}$ when $p = 0$. The second diagonal handles the polynomials $Q^{(i)}_{(n+3,n)}$, when $p = 1$, etc.

4.2 Formulation of the recursive quasi-orthogonal polynomial algorithm

This section introduces the proposed algorithm to compute the quasi-orthogonal polynomials of the form (4.3) recursively using (4.14) and (4.15) formulas. The proposed algorithm provides an efficient procedure to generate the coefficients $\beta^{(i)}_{m,n}$ without computing the lower block determinants. Note that the computation of the short-term recurrence relations of quasi-orthogonal polynomials in (4.14) and (4.15) require evaluating $\beta^{(i)}_{m,n}$, $\beta^{(i)}_{m,n-1}$ and $\beta^{(i+1)}_{m-1,n}$, where $m-n$ is assumed to be odd. Two procedures are described to compute these quantities : the first one for computing $\beta^{(i)}_{m,n}$ when $m-n$ is odd, and the second one for $m-n$ is even. First, we define some notations that will be needed.

In the following, we set $\beta^{(i)}_{m,n}$ to be $\beta^{(i)}_{m,n} = \dfrac{\alpha^{(i)}_{m+1,n-1}}{\alpha^{(i+1)}_{m,n}}$ with

$$\alpha^{(i)}_{m,n} = \frac{|D^{(i)}_{m,n}|}{|D^{(i+1)}_{m-1,n}|}. \tag{4.18}$$

For $m-n=2p$, i.e., $r=\frac{m+n}{2}$, we define $\gamma^{(i)}_{m,n}$ as
$\gamma^{(i)}_{m,n} =$

$$\left(\left[\begin{array}{c|cccccc} c^1_i & c^1_{i+1} & \cdots & c^1_{i+m-1} & 0 & \cdots & 0 \\ c^2_i & c^2_{i+1} & \cdots & c^2_{i+m-1} & c^1_{i+m-1} & \cdots & c^1_{i+m-n-1} \\ \vdots & & & & & & \vdots \\ c^1_{i+r-2} & c^1_{i+r-1} & \cdots & c^1_{i+m+r-3} & 0 & \cdots & 0 \\ c^2_{i+r-2} & c^2_{i+r-1} & \cdots & c^2_{i+m+r-3} & c^1_{i+m+r-3} & \cdots & c^1_{i+m-n+r-3} \\ \hline c^2_{i+r-1} & c^2_{i+r} & \cdots & c^2_{i+m+r-2} & c^1_{i+m+r-2} & \cdots & c^1_{i+m-n+r-2} \end{array}\right] / D^{(i+1)}_{m-1,n}\right) \tag{4.19}$$

and for $m-n=2p+1$, i.e., $r=\dfrac{m+n+1}{2}$, we define $\delta^{(i)}_{m,n}$ as
$\delta^{(i)}_{m,n} =$

$$\left(\left[\begin{array}{c|cccccc} c^1_i & c^1_{i+1} & \cdots & c^1_{i+m-1} & 0 & \cdots & 0 \\ c^2_i & c^2_{i+1} & \cdots & c^2_{i+m-1} & c^1_{i+m-1} & \cdots & c_{i+m-n-1} \\ \vdots & & & \vdots & \vdots & & \vdots \\ c^1_{i+r-3} & c^1_{i+r-2} & \cdots & c^1_{i+m+r-4} & 0 & \cdots & 0 \\ c^2_{i+r-3} & c^2_{i+r-2} & \cdots & c^2_{i+m+r-4} & c^1_{i+m+r-4} & \cdots & c^1_{i+m-n+r-4} \\ c^1_{i+r-2} & c^1_{i+r-1} & \cdots & c^1_{i+m+r-3} & 0 & \cdots & 0 \\ \hline c^1_{i+r-1} & c^1_{i+r} & \cdots & c^1_{i+m+r-2} & 0 & \cdots & 0 \end{array}\right] / D^{(i+1)}_{m-1,n}\right) \tag{4.20}$$

4.2.1 Computation of coefficients $\beta_{m,n}$ with $m-n=2p+1$

The following theorem gives the procedure to compute the coefficients $\beta^{(i)}_{m,n}$ when $m-n$ is odd.

Théorème 22 *For all $m>n>0$, with $m-n=2p+1$, and $r=\dfrac{m+n+1}{2}$ the quantities $\alpha^{(i)}_{m,n},\gamma^{(i)}_{m,n}$ are giving in (4.18) and (4.19) respectively. The following relations define a recursive algorithm for computing the coefficients $\beta^{(i)}_{m,n}$. The coefficients $\beta^{(i)}_{m,n}$ are computed via the following equations*

$$\begin{cases} \beta^{(i)}_{m,n} = \dfrac{\alpha^{(i)}_{m+1,n-1}}{\alpha^{(i+1)}_{m,n}}, \\ \alpha^{(i)}_{m,n} = \gamma^{(i)}_{m,n-1} - \gamma^{(i+1)}_{m-1,n}\beta^{(i)}_{m-1,n}, \\ \gamma^{(i)}_{m,n-1} = c^2_{i+r-1} + \left\langle C^{(i)}_{m-1,n-1}, q^{(i)}_{m-1,n-1} \right\rangle, \\ \gamma^{(i+1)}_{m-1,n} = c^2_{i+r} + \left\langle C^{(i+1)}_{m-2,n}, q^{(i+1)}_{m-2,n} \right\rangle, \end{cases} \tag{4.21}$$

where $\langle\cdot,\cdot\rangle$ is the 2-inner product for two vectors,

$$C^{(i)}_{m-1,n-1} = \begin{bmatrix} c^2_{i+r} \\ \vdots \\ c^2_{i+r+m-2} \\ c^1_{i+r+m-2} \\ \vdots \\ c^1_{i+r+m-n-1} \end{bmatrix}, \quad q^{(i)}_{m-1,n-1} = \begin{bmatrix} q^{1,i}_1 \\ q^{1,i}_2 \\ \vdots \\ q^{1,i}_{m-1} \\ q^{2,i}_{n-1} \\ \vdots \\ q^{2,i}_0 \end{bmatrix}$$

and $\{q^{1,i}_1,\dots,q^{1,i}_{m-1}\}$, $\{q^{2,i}_0,\dots,q^{2,i}_{n-1}\}$ are the coefficients of $Q^{(i)}_{(m-1,n-1)_1}$, $Q^{(i)}_{(m-1,n-1)_2}$; respectively.

Proof.

According to (4.18), $\alpha^{(i)}_{m,n}$ can be written as Schur complement as

$$\alpha^{(i)}_{m,n} = \left(\begin{bmatrix} c^1_i & c^1_{i+1} & \cdots & c^1_{i+m-1} & 0 & \cdots & 0 \\ c^2_i & c^2_{i+1} & \cdots & c^2_{i+m-1} & c^1_{i+m-1} & \cdots & c^1_{i+m-n-1} \\ \vdots & & & \vdots & & & \vdots \\ c^1_{i+r-1} & c^1_{i+r} & \cdots & c^1_{i+m+r-2} & 0 & \cdots & 0 \\ c^2_{i+r-1} & c^2_{i+r} & \cdots & c^2_{i+m+r-2} & c^1_{i+m+r-2} & \cdots & c^1_{i+m-n+r-1} \end{bmatrix} / D^{(i+1)}_{m-1,n} \right).$$

We apply the Sylvester identity to

$$\alpha_{m,n}^{(i)} = \left(\left[\begin{array}{c|cccccc|c} c_i^1 & c_{i+1}^1 & \cdots & c_{i+m-1}^1 & 0 & \cdots & 0 & 0 \\ c_i^2 & c_{i+1}^2 & \cdots & c_{i+m-1}^2 & c_{i+m-1}^1 & \cdots & c_{i+m-n}^1 & c_{i+m-n-1}^1 \\ \vdots & & & \vdots & & & & \vdots \\ c_{i+r-2}^1 & c_{i+r-1}^1 & \cdots 1 & c_{i+m+r-3}^1 & 0 & \cdots & 0 & 0 \\ c_{i+r-2}^2 & c_{i+r-1}^2 & \cdots & c_{i+m+r-3}^2 & c_{i+m+r-3}^1 & \cdots & c_{i+m-n+r-2}^1 & c_{i+m-n+r-3}^1 \\ \hline c_{i+r-1}^1 & c_{i+r}^1 & \cdots & c_{i+m+r-2}^1 & 0 & \cdots & 0 & 0 \\ \hline c_{i+r-1}^2 & c_{i+r}^2 & \cdots & c_{i+m+r-2}^2 & c_{i+m+r-2}^1 & \cdots & c_{i+m-n+r-1}^1 & c_{i+m-n+r-2}^1 \end{array} \right] / D_{m-1,n}^{(i+1)} \right).$$

We obtain

$$\alpha_{m,n}^{(i)} = \left(\left[\begin{array}{c|cccccc} c_i^1 & c_{i+1}^1 & \cdots & c_{i+m-1}^1 & 0 & \cdots & 0 \\ c_i^2 & c_{i+1}^2 & \cdots & c_{i+m-1}^2 & c_{i+m-1}^1 & \cdots & c_{i+m-n}^1 \\ \vdots & & & \vdots & & & \vdots \\ c_{i+r-2}^1 & c_{i+r-1}^1 & \cdots & c_{i+m+r-3}^1 & 0 & \cdots & 0 \\ c_{i+r-2}^2 & c_{i+r-1}^2 & \cdots & c_{i+m+r-3}^2 & c_{i+m+r-3}^1 & \cdots & c_{i+m-n+r-2}^1 \\ \hline c_{i+r-1}^2 & c_{i+r}^2 & \cdots & c_{i+m+r-2}^2 & c_{i+m+r-2}^1 & \cdots & c_{i+m-n+r-1}^1 \end{array} \right] / D_{m-1,n-1}^{(i+1)} \right)$$

$$- \left(\left[\begin{array}{cccccc|c} c_{i+1}^1 & \cdots & c_{i+m-1}^1 & 0 & \cdots & 0 & 0 \\ c_{i+1}^2 & \cdots & c_{i+m-1}^2 & c_{i+m-1}^1 & \cdots & c_{i+m-n}^1 & c_{i+m-n-1}^1 \\ \vdots & & \vdots & & & & \vdots \\ c_{i+r-1}^1 & \cdots & c_{i+m+r-3}^1 & 0 & \cdots & 0 & 0 \\ c_{i+r-1}^2 & \cdots & c_{i+m+r-3}^2 & c_{i+m+r-3}^1 & \cdots & c_{i+m-n+r-2}^1 & c_{i+m-n+r-3}^1 \\ \hline c_{i+r}^2 & \cdots & c_{i+m+r-2}^2 & c_{i+m+r-2}^1 & \cdots & c_{i+m-n+r-1}^1 & c_{i+m-n+r-2}^1 \end{array} \right] / D_{m-1,n-1}^{(i+1)} \right)$$

$$\times \left(D_{m-1,n}^{(i+1)} / D_{m-1,n-1}^{i+1} \right)^{-1} \times \left(D_{m,n-1}^{(i)} / D_{m-1,n-1}^{(i+1)} \right).$$

By using the Schur complement property : $(M/D) = \dfrac{|M|}{|D|}$, and multiplying by $\dfrac{|D_{m-2,n}^{(i+2)}|}{|D_{m-2,n}^{(i+2)}|}$, we get

$$\alpha_{m,n}^{(i)} = \gamma_{m,n-1}^{(i)} - \frac{\left| \begin{array}{c|cccccc} c_{i+1}^1 & c_{i+2}^1 & \cdots & c_{i+m-1}^1 & 0 & \cdots & 0 & 0 \\ c_{i+1}^2 & c_{i+2}^2 & \cdots & c_{i+m-1}^2 & c_{i+m-1}^1 & \cdots & c_{i+m-n}^1 & c_{i+m-n-1}^1 \\ \vdots & & \vdots & & & \vdots & & \\ c_{i+r-1}^1 & c_{i+r}^1 & \cdots & c_{i+m+r-3}^1 & 0 & \cdots & 0 & 0 \\ c_{i+r-1}^2 & c_{i+r}^2 & \cdots & c_{i+m+r-3}^2 & c_{i+m+r-3}^1 & \cdots & c_{i+m-n+r-2}^1 & c_{i+m-n+r-3}^1 \\ \hline c_{i+r}^2 & c_{i+r+1}^2 & \cdots & c_{i+m+r-2}^2 & c_{i+m+r-2}^1 & \cdots & c_{i+m-n+r-1}^1 & c_{i+m-n+r-2}^1 \end{array} \right|}{\left| D_{m-2,n}^{(i+2)} \right|}$$

$$\times \frac{\left| D_{m-2,n}^{(i+2)} \right| \left| D_{m,n-1}^{(i)} \right|}{\left| D_{m-1,n-1}^{(i+1)} \right| \left| D_{m-1,n}^{(i+1)} \right|}.$$

Therefore
$$\alpha^{(i)}_{m,n} = \gamma^{(i)}_{m,n-1} - \gamma^{(i+1)}_{m-1,n}\beta^{(i)}_{m-1,n}.$$
$\gamma^{(i)}_{m,n-1}$ and $\gamma^{(i+1)}_{m-1,n}$ in (4.21) can be written as follows :

$$\gamma^{(i)}_{m,n-1} = c^2_{i+r-1} + \begin{bmatrix} c^2_{i+r} & \dots & c^2_{i+r+m-2} & c^1_{i+r+m-2} & \dots & c^1_{i+r+mn-1} \end{bmatrix} (D^{(i+1)}_{m-1,n-1})^{-1} \begin{bmatrix} -c^1_i \\ -c^2_i \\ \vdots \\ -c^1_{i+r-2} \\ -c^2_{i+r-2} \end{bmatrix}$$

, and

$$\gamma^{(i+1)}_{m-1,n} = c^2_{i+r} + \begin{bmatrix} c^2_{i+r+1} & \dots & c^2_{i+r+m-2} & c^1_{i+r+m-2} & \dots & c^1_{i+r+m-n-2} \end{bmatrix} (D^{(i+2)}_{m-2,n})^{-1} \begin{bmatrix} -c^1_{i+1} \\ -c^2_{i+1} \\ \vdots \\ -c^1_{i+r-1} \\ -c^2_{i+r-1} \end{bmatrix}.$$

Set

$$C^{(i)}_{m-1,n-1} = \begin{bmatrix} c^2_{i+r} \\ \dots \\ c^2_{i+r+m-2} \\ c^1_{i+r+m-2} \\ \dots \\ c^1_{i+r+m-n-1} \end{bmatrix} \quad \text{and} \quad q^{(i)}_{m-1,n-1} = \begin{bmatrix} q^{1,i}_1 \\ q^{1,i}_2 \\ \vdots \\ q^{1,i}_{m-1} \\ q^{2,i}_{n-1} \\ \vdots \\ q^{2,i}_0 \end{bmatrix}$$

such as,

$$q^{(i)}_{m-1,n-1} = (D^{(i+1)}_{m-1,n-1})^{-1} \begin{bmatrix} -c^1_i \\ -c^2_i \\ \vdots \\ -c^1_{i+r-2} \\ -c^2_{i+r-2} \end{bmatrix},$$

with $q^{(i)}_{m-1,n-1}$ are the coefficients of the polynomial $Q^{(i)}_{m-1,n-1}$ given by (4.1). It follows that
$$\gamma^{(i)}_{m,n-1} = c^2_{i+r-1} + \left\langle C^{(i)}_{m-1,n-1}, q^{(i)}_{m-1,n-1} \right\rangle,$$
and
$$\gamma^{(i+1)}_{m-1,n} = c^2_{i+r} + \left\langle C^{(i+1)}_{m-2,n}, q^{(i+1)}_{m-2,n} \right\rangle$$
Which completes the proof.

4.2.2 Computation of coefficients $\beta_{m,n}$ with $m - n = 2p$

The computation of the quantities $\beta^{(i)}_{m,n}$, when $m - n = 2p$ is described in the next theorem.

Théorème 23 *For all $m > n > 0$ and $m - n = 2p$, for some p and $r = \frac{m+n}{2}$.the quantities $\alpha^{(i)}_{m,n}, \delta^{(i)}_{m,n}$ are giving in* (4.18) *and* (4.20) *respectively. The following relations define a recursive algorithm for computing the coefficients $\beta^{(i)}_{m,n}$. The coefficients $\beta^{(i)}_{m,n}$ are computed as*

$$\begin{cases} \beta^{(i)}_{m,n} = \dfrac{\alpha^{(i)}_{m+1,n-1}}{\alpha^{i+1}_{m,n}} \\ \alpha^{(i)}_{m,n} = \delta^{(i)}_{m,n-1} - \delta^{(i+1)}_{m-1,n}\beta^{(i)}_{m-1,n} \\ \delta^{(i)}_{m,n-1} = c^1_{i+r} + \left\langle C^{(1,i)}_{m-1,n-1}; q^{(i)}_{m-1,n-1} \right\rangle \\ \delta^{(i+1)}_{m-1,n} = c^1_{i+r+1} + \left\langle C^{(1,i+1)}_{m-2,n'}; q^{(i+1)}_{m-2,n} \right\rangle . \end{cases}$$

where

$$C^{(1,i)}_{m'-1,n'-1} = \begin{bmatrix} c^1_{i+r'+1} \\ c^1_{i+r'+2} \\ \vdots \\ c^1_{i+m'+r'-1} \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad C^{(1,i+1)}_{m'-2,n'} = \begin{bmatrix} c^1_{i+r'+2} \\ c^1_{i+r'+3} \\ \vdots \\ c^1_{i+m'+r'-1} \\ 0 \\ \vdots \\ 0 \end{bmatrix},$$

and $q^{(i)}_{m-1,n-1}, q^{(i+1)}_{m-2,n}$ are the coefficients of the polynomials $Q^{(i)}_{m-1,n-1}$, and $Q^{(i+1)}_{m-2,n}$; respectively.

Proof.

The proof is similar to the ones given in Theorem 22.

4.2.3 Computation of coefficients $\beta_{m,-1}$ with m=2p

Now, we present a recursive procedure to compute $\beta_{m,-1}$ as defined below (4.17). Consider

$$\alpha^{(i)}_{m,-1} = \frac{|D^{(i)}_{m,-1}|}{|D^{(i+1)}_{m-1,-1}|} \tag{4.22}$$

When m is even, and $r = \dfrac{m}{2}$, we define the matrices $D^{(i)}_{m,-1}$,$\gamma^{(i)}_{m-1,-1}$

$$D^{(i)}_{m,-1} = \begin{bmatrix} c^1_i & c^1_{i+1} & \cdots & c^1_{i+m-1} \\ c^2_i & c^2_{i+1} & \cdots & c^2_{i+m-1} \\ \vdots & \vdots & & \vdots \\ c^1_{i+r-1} & c^1_{i+r} & \cdots & c^1_{i+m+r-2} \\ c^2_{i+r-1} & c^2_{i+r} & \cdots & c^2_{i+m+r-2} \end{bmatrix},$$

$$\gamma^{(i)}_{m-1,-1} = \left(\left[\begin{array}{c|ccc} c^1_i & c^1_{i+1} & \cdots & c^1_{i+m-2} \\ c^2_i & c^2_{i+1} & \cdots & c^2_{i+m-2} \\ \vdots & \vdots & & \vdots \\ c^2_{i+r-2} & c^2_{i+r-1} & \cdots & c^2_{i+m+r-4} \\ \hline c^2_{i+r-1} & c^2_{i+r} & \cdots & c^2_{i+m+r-3} \end{array} \right] / D^{(i+1)}_{m-2,-1} \right). \tag{4.23}$$

When m is odd, and $r = \dfrac{m-1}{2}$, we define

$$D^{(i)}_{m,-1} = \begin{bmatrix} c^1_i & c^1_{i+1} & \cdots & c^1_{i+m-1} \\ c^2_i & c^2_{i+1} & \cdots & c^2_{i+m-1} \\ \vdots & \vdots & & \vdots \\ c^1_{i+r-1} & c^1_{i+r} & \cdots & c^1_{i+m+r-2} \\ c^2_{i+r-1} & c^2_{i+r} & \cdots & c^2_{i+m+r-2} \\ c^1_{i+r} & c^1_{i+r+1} & \cdots & c^1_{i+m+r-1} \end{bmatrix}$$

$$\delta^{(i)}_{m-1,-1} = \left(\left[\begin{array}{c|ccc} c^1_i & c^1_{i+1} & \cdots & c^1_{i+m-2} \\ c^2_i & c^2_{i+1} & \cdots & c^2_{i+m-2} \\ \vdots & \vdots & & \vdots \\ c^1_{i+r-1} & c^1_{i+r} & \cdots & c^1_{i+m+r-3} \\ \hline c^1_{i+r} & c^1_{i+r+1} & \cdots & c^1_{i+m+r-2} \end{array} \right] / D^{(i+1)}_{m-1,-1} \right) \tag{4.24}$$

The following lemmas give the procedure to compute the coefficient $\beta^{(i)}_{m,-1}$ used in (4.17) recursively.

Lemme 24 *Let $\alpha^{(i)}_{m,-1}, \gamma^{(i)}_{m,-1}$ be defined by* (4.22) *and* (4.23) *; respectively. When m is even, and $r = \frac{m}{2}$. Then $\beta^{(i)}_{m,-1}$ can be computed by using the following formulas.*

$$\begin{cases} \beta^{(i)}_{m,-1} = \dfrac{\alpha^{(i)}_{m,-1}}{\alpha^{(i+1)}_{m,-1}}, \beta_{2,-1} = \dfrac{c^1_2(c^1_0 c^2_1 - c^2_0 c^1_1)}{c^1_1(c^1_1 c^2_2 - c^2_1 c^1_2)}, \\ \alpha^{(i)}_{m,-1} = \gamma^{(i)}_{m-1,-1} - \gamma^{(i+1)}_{m-1,-1}\beta^{(i)}_{m-1,-1} \\ \gamma^{(i)}_{m-1,-1} = c^2_{i+r-1} + \left\langle C^{(i)}_{m-2,-1}; q^{(i)}_{m-2,-1} \right\rangle \end{cases}$$

where

$$C^{(i)}_{m-2,-1} = \begin{bmatrix} c^2_{i+r} \\ \vdots \\ c^2_{i+m+r-3} \end{bmatrix},$$

and $q^{(i)}_{m-2,-1}$ *is the vector column of the coefficients of the polynomial* $Q^{(i)}_{m-2,-1}$.

Lemme 25 *Let* $\alpha^{(i)}_{m,-1}, \delta^{(i)}_{m,-1}$ *be defined by (4.22) and (4.24) respectively. When* m *is odd and* $r = \dfrac{m-1}{2}$. *Then* $\beta^{(i)}_{m,n}$ *are computed as*

$$\begin{cases} \beta^{(i)}_{m,-1} = \dfrac{\alpha^{(i)}_{m,-1}}{\alpha^{(i+1)}_{m,-1}}, \beta_{1,-1} = \dfrac{c^1_0}{c^1_1}, \\ \alpha^{(i)}_{m,-1} = \delta^{(i)}_{m-1,-1} - \delta^{(i+1)}_{m-1,-1}\beta^{(i)}_{m-1,-1} \\ \delta^{(i)}_{m-1,-1} = c^1_{i+r} + \left\langle C^{1(i)}_{m-2,-1}; q^{(i)}_{m-2,-1} \right\rangle \end{cases} \tag{4.25}$$

with

$$C^{1(i)}_{m-2,-1} = \begin{bmatrix} c^1_{i+r+1} \\ \vdots \\ c^1_{i+m+r-2} \end{bmatrix},$$

and $q^{(i)}_{m-2,-1}$ *are the coefficients of the polynomial* $Q^{(i)}_{m-2,-1}$.

Algorithm 5 describes how the matrix polynomials $Q^{(i)}_{m,n}$ of the form (4.3) are computed by the short-term recurrence relations (4.14) and (4.15).

Algorithm 5 *(Recursive Algorithm for computation of quasiorthogonal polynomials)*

1. *Inputs :* $Q_0^{(i)}(x) = 1$, $Q_{m_2}^{(i)}(x) = 0$, $Q_{(1,0)_1}^{(i)}(x) = 1 - \frac{c_i^1}{c_{i+1}^1}$, $|D_0^{(i)}| = 1$, $Q_{(1,0)_2}^{(i)}(x) = \dfrac{|D_{2,-1}^{(i)}|}{(c_{i+1}^1)^2}$, $|D_1^{(i)}| = c_i^1$.
2. *For* $k = 0, \ldots\ldots, h$
   - *Set* $m = 2k$
   - *Compute* $Q_m^{(i)}(x) = Q_{m-2}^{(i)}(x) - x(\beta_m^{(i)} + \beta_{m-1}^{(i)})Q_{m-2}^{(i+1)}(x) + \beta_m^{(i)}\beta_{m-1}^{(i+1)}Q_{m-2}^{(i+2)}(x)$.
   - *end*
3. *For* $k' = 1, \ldots\ldots, h'$
   - *Set* $m = 2k' + 1$
   - *Compute* $Q_{(m,0)_1}^{(i)}(x) = Q_{(m-1,-1)_1}^{(i)}(x) - x(\beta_{m,0}^{(i)} + \beta_m^{(i)})Q_{(m-1,-1)_1}^{(i+1)}(x) + x^2\beta_{m-1,0}^{i+1}\beta_{m,0}^{(i)}Q_{(m-2,0)_1}^{(i+2)}(x)$
   - *end*
4. *For* $p = 1, \ldots\ldots, l$
   - *For* $m = 1, \ldots\ldots, s$
   - *For* $n = 1, \ldots\ldots, s'$
   - *If* $m - n = 2p + 1$ *do*
   - *Compute* $Q_{(m,n)_1}^{(i)}(x) = Q_{(m,n-2)_1}^{(i)}(x) - x(\beta_{m,n}^{(i)} + \beta_{m,n-1}^{(i)})Q_{(m-1,n-1)_1}^{(i+1)}(x) + x^2\beta_{m,n}^{(i)}\beta_{m-1,n}^{(i+1)}Q_{(m-2,n)_1}^{(i+2)}(x)$
   - *Compute* $Q_{(m,n)_2}^{(i)}(x) = x^2 Q_{(m,n-2)_2}^{(i)}(x) - x(\beta_{m,n}^{(i)} + \beta_{m,n-1}^{(i)})Q_{(m-1,n-1)_2}^{(i+1)}(x) + \beta_{m,n}^{(i)}\beta_{m-1,n}^{(i+1)}Q_{(m-2,n)_2}^{(i+2)}(x)$
   - *end*

Outputs : polynomials $Q_{(m,n)_1}^{(i)}(x), Q_{(m,n)_2}^{(i)}(x)$.

We give the following example to illustrate the different steps of the proposed algorithm to compute the quasi-orthogonal polynomials with applications to the vector Padé approximant of the form (2.8).

4.3 Example

Consider the function

$$F(x) = \begin{bmatrix} e^x \\ ln(1+x) \end{bmatrix} = \begin{bmatrix} 1 + x + \frac{x^2}{2} + \frac{x^3}{6} + \frac{x^4}{24} + \frac{x^5}{120} + \dots \\ x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \frac{x^5}{5} - \frac{x^6}{6} + \dots \end{bmatrix}.$$

The main of this example is to compute the vector Padé approximant $\widetilde{Q}_{m,n}$, for $m = 5$ and $n = 2$. First, we apply the Algorithm 5 to compute the quasi-orthogonal polynomial $Q_{m,n}$.
We start to compute $Q_{1,-1}$ and $Q_{0,-1}$.

$$Q_{1,-1} = \begin{bmatrix} 1 - x\dfrac{c_i^1}{c_{i+1}^{(1)}} & 0 \\ 0 & 1 - x\dfrac{c_i^1}{c_{i+1}^{(1)}} \end{bmatrix}, Q_{0,-1} = 1, Q_{1,0}^{(i)} = \begin{bmatrix} 1 - x\dfrac{c_i^1}{c_{i+1}^{(1)}} & \\ \dfrac{|D_2^{(i)}}{D_{1,0}^{(i+1)}} & 1 - x\dfrac{c_i^1}{c_{i+1}^{(1)}} \end{bmatrix}.$$

$$\beta_{2,-1} = \frac{c_2^1(c_0^1c_1^2 - c_0^2c_1^1)}{c_1^1(c_1^1c_2^2 - c_1^2c_2^1)}, \quad \beta_{1,-1} = \frac{c_0^1}{c_1^1}, \quad \beta_1^{(1)} = \frac{c_1^1}{c_2^1}$$

From (4.16), we have

$$Q_{(2,-1)_1}(x) = 1 - \frac{1}{2}x - x^2$$

$$Q_{(2,-1)_1}^{(1)}(x) = 1 - (2/3)x - 4x^2$$

$$Q_{(2,-1)_1}^{(2)}(x) = 1 - (9 * x^2)/2 - (15 * x)/8$$

$$Q_{(2,-1)_1}^{(3)}(x) = 1 - (16 * x^2)/3 - (44 * x)/15.$$

According to (4.25), we get

$$\begin{cases} \beta_{3,-1}^{(0)} = \dfrac{\alpha_{3,-1}^{(0)}}{\alpha_{3,-1}^{(1)}} \\ \alpha_{3,-1}^{(0)} = \delta_{2,-1}^{(0)} - \delta_{2,-1}^{(1)}\beta_{2,-1}^{(0)} = \frac{7}{12} \\ \delta_{2,-1}^{(0)} = c_1^1 + c_2^1\dfrac{-c_0^1}{c_1^1} = 1/2 \\ \delta_{2,-1}^{(1)} = c_2^1 + c_3^1\dfrac{-c_1^1}{c_2^1} = 1/6. \end{cases}$$

such that $\dfrac{-c_0^1}{c_1^1}$ and $\dfrac{-c_1^1}{c_2^1}$ are respectively the coefficients of $Q_1^{(0)}$ and $Q_1^{(1)}$.
Similarly,

$$\begin{cases} \alpha_{3,-1}^{(1)} = \delta_2^{(1)} - \delta_2^{(2)}\beta_{2,-1}^{(1)} = \frac{2}{9} \\ \beta_{2,-1}^{(1)} = -4/3 \\ \delta_{2,-1}^{(2)} = c_3^1 + c_4^1\dfrac{-c_2^1}{c_3^1} = 1/24 \\ \beta_{3,-1}^{(0)} = 2.625. \end{cases}$$

and

$$\begin{cases} \beta_{2,-1}^{(2)} = -9/8 \\ \delta_{2,-1}^{(3)} = 1/120 \\ \beta_{3,-1}^{(1)} = 4.3573 \end{cases}$$

and

$$\begin{cases} \beta_{4,-1}^{(0)} = \dfrac{\alpha_{4,-1}^{(0)}}{\alpha_{4,-1}^{(1)}} = 0.2660 \\ \alpha_{4,-1}^{(0)} = \gamma_3^{(0)} - \gamma_3^{(1)}\beta_{3,-1}^{(0)} = 0.1874 \\ \alpha_{4,-1}^{(1)} = \gamma_3^{(1)} - \gamma_3^{(2)}\beta_{3,-1}^{(1)} = 0.7044 \\ \gamma_{3,-1}^{(0)} = c_1^2 + [c_2^2 \;\; c_3^2]\begin{bmatrix} q_1^{1,0} \\ q_2^{1,0} \end{bmatrix} = 11/12 \\ \gamma_{3,-1}^{(1)} = c_2^2 + [c_3^2 \;\; c_4^2]\begin{bmatrix} q_1^{1,1} \\ q_2^{1,1} \end{bmatrix} = 0.2778 \\ \gamma_{3,-1}^{(2)} = c_3^2 + [c_4^2 \;\; c_5^2]\begin{bmatrix} q_1^{1,2} \\ q_2^{1,2} \end{bmatrix} = -0.0979 \end{cases}$$

with the $q_1^{1,i}$ and $q_2^{1,i}$ are the coefficients of $Q_{2,-1}^{(i)}$.
$Q_{4,-1}(x)$ and $Q_{4,-1}^{(1)}(x)$ are now given by
$Q_{4,-1}(x) =$

$$\begin{pmatrix} 1 - 3.3913x + 2.087x^2 + 9.3913x^3 - 5.2174x^4 & 0 \\ 0 & 1 - 3.3913x + 2.087x^2 + 9.3913x^3 - 5.2174x^4 \end{pmatrix}$$

and
$Q_{4,-1}^{(1)}(x) =$

$$\begin{pmatrix} 52.865x^4 + 40.0997x^3 - 9.3191x^2 - 3.11635x + 1 & 0 \\ 0 & 52.865x^4 + 40.0997x^3 - 9.3191x^2 - 3.11635x + 1 \end{pmatrix},$$

with $\begin{cases} \beta^{(1)}_{4,-1} = -1.907613, \\ \beta^{(2)}_{3,-1} = 5.1962 \\ \delta^{(4)}_{2,-1} = 1/720, \\ \alpha^{(2)}_{4,-1} = -0.369587, \\ \gamma^{(3)}_{3,-1} = 47/900 \end{cases}$

Now, we compute the coefficients $\beta^{(3)}_{2,0} = \dfrac{\alpha^{(3)}_{3,-1}}{\alpha^{(4)}_{2,0}}$ and $\beta^{(2)}_{3,0} = \dfrac{\alpha^{(2)}_{4,-1}}{\alpha^{(3)}_{3,0}}$ via

$$\begin{cases} \beta^{(3)}_{2,0} = 7.0667, \\ \beta^{(2)}_{3,0} = 0.0423, \\ \alpha^{(4)}_{2,0} = \delta^{(4)}_{2,-1} = c^1_5 + c^1_6 \dfrac{-c^1_4}{c^1_5} = 1/720 \\ \alpha^{(3)}_{3,0} = \gamma^{(3)}_{3,-1} - \gamma^{(4)}_{2,0}\beta^{(3)}_{2,0} = -8.7223 \\ \gamma^{(3)}_{3,-1} = c^{(2)}_4 + [c^2_5 \ \ c^2_6] \begin{bmatrix} q^{1,3}_1 \\ q^{1,3}_2 \end{bmatrix} = 0.0522 \\ \gamma^{(4)}_{2,0} = c^2_5 + [\ c^2_6 \quad c^2_6\ ][\ \begin{matrix} q^{1,4}_1 \\ q^{2,4}_0 \end{matrix}\ ] = 149/120 \end{cases}$$

with $q^{1,3}_1, q^{1,3}_2$ are the coefficients of $Q^{(3)}_{2,-1}$, and $q^{1,4}_1, q^{2,4}_0$ are the coefficients of $Q^{(4)}_{1,0}$, we obtain

$$Q^{(2)}_{(3,0)_1} = 26.4441x^3 + 11.1652x^2 - 7.1135x + 1,$$

$$Q^{(2)}_{(3,0)_2} = 44.904458598726023.$$

$Q^{(3)}_{(2,1)_1}$ and $Q^{(3)}_{(2,1)_2}$ are given by

$$Q^{(3)}_{(2,1)_1} = 30x^2 - 10x + 1,$$

$$Q^{(3)}_{(2,1)_2} = 6280 - 32460x$$

Similarly, we have

$$Q^{(2)}_{(3,2)_1} = -210x^3 + 90.000000000000227x^2 - 15.000000000000034x + 1$$

$$Q^{(2)}_{(3,2)_2}(x) = 5262599.999833005x^2 - 1718399.999945471x + 166889.9999947042$$

The computation of $Q^{(1)}_{4,1}$ leads to

$$Q^{(1)}_{(4,1)_1}(x) = Q^{(1)}_{4,-1}(x) - x(\beta^{(1)}_{4,1} + \beta^{(1)}_{4,0})Q^{(2)}_{(3,0)_1}(x) + x^2\beta^{(1)}_{4,1}\beta^{(2)}_{3,1}Q^{(3)}_{(2,1)_1}(x)$$

$$Q^{(1)}_{(4,1)_2}(x) = -x(\beta^{(1)}_{4,1} + \beta^{(1)}_{4,0})Q^{(2)}_{(3,0)_2}(x) + x^2\beta^{(1)}_{4,1}\beta^{(2)}_{3,1}Q^{(3)}_{(2,1)_2}(x)$$

with

$$\begin{cases} \beta_{3,1}^{(2)} = \dfrac{\alpha_{4,0}^{(2)}}{\alpha_{3,1}^{(3)}} = 7.91688; \\ \beta_{4,0}^{(1)} = \dfrac{\alpha_{5,-1}^{(1)}}{\alpha_{4,0}^{(2)}} = 8.0667322 \\ \beta_{4,1}^{(1)} = \dfrac{\alpha_{5,0}^{(1)}}{\alpha_{4,1}^{(2)}} = 0.02813 \\ \alpha_{5,0}^{(1)} = \gamma_{5,-1}^{(1)} - \gamma_{4,0}^{(2)}\beta_{4,0}^{(1)} = -1.9239 \\ \alpha_{5,-1}^{(1)} = \delta_{4,-1}^{(1)} - \delta_{4,-1}^{(2)}\beta_{4,-1}^{(1)} = 0.02534 \\ \delta_{4,-1}^{(1)} = 0.019317; \delta_{4,-1}^{(2)} = 0.00315831 \end{cases}$$

Then, we find

$$Q_{(4,1)_1}^{(1)} = -154.475702567x^4 - 52.5455755315x^3 + 48.4901667679x^2 - 11.2100094x + 1$$

$$Q_{(4,1)_2}^{(2)} = -6865.406140101707x + 1398.567520032000$$

We continue with the same steps and compute $Q_{5,0}^{(0)}$

$$Q_{(5,0)_1}^{(0)}(x) = Q_{(4,-1)_1}^{(0)} - x(\beta_{5,0}^{(0)} + \beta_{5,-1}^{(0)})Q_{(4,-1)_1}^{(1)}(x) + x^2\beta_{5,0}^{(0)}\beta_{4,0}^{(1)}Q_{(3,0)_1}^{(2)}(x)$$

$$Q_{(5,0)_2}^{(0)}(x) = \beta_{5,0}^{(0)}\beta_{4,0}^{(1)}Q_{(3,0)_2}^{(2)}(x)$$

the coefficients are given by

$$\begin{cases} \beta_{5,-1}^{(0)} = \dfrac{\alpha_{5,-1}^{(0)}}{\alpha_{5,-1}^{(1)}} = 3.661339568 \\ \alpha_{5,-1}^{(0)} = \delta_{4,-1}^{(0)} - \delta_{4,-1}^{(1)}\beta_{4,-1}^{(0)} = 0.0927783 \end{cases}, \begin{cases} \beta_{5,0}^{(0)} = \dfrac{\alpha_{6,-1}^{(0)}}{\alpha_{5,0}^{(1)}} = -0.085831395 \\ \alpha_{6,-1}^{(0)} = \gamma_{5,-1}^{(0)} - \gamma_{5,-1}^{(1)}\beta_{5,-1}^{(0)} = 0.06235819 \end{cases}$$

then, it holds

$$Q_{(5,0)_1}^{(0)} = -207.278746995x^5 - 156.272574x^4 + 47.63693x^3 + 12.5319487x^2 - 6.96680817x + 1$$

$$Q_{(5,0)_2}^{(2)}(x) = -31.0908986.$$

It remains to compute the coefficients $\beta_{4,2}^{(1)}, \beta_{5,1}^{(0)}$ and $\beta_{5,2}^{(0)}$, then we compute the auxiliary polynomials $Q_{3,1}^{(2)}, Q_5^{(0)}$, $Q_{4,0}^{(1)}$ as follows

$$Q_{(3,1)_1}^{(2)} = Q_{(3,0)_1}^{(2)}(x) - x\beta_{3,1}^{(2)}Q_{(2,1)_1}^{(3)} = -211.0623x^3 + 90.3340x^2 - 15.0304x + 1$$

$$Q_{(5,-1)_1}^{(0)} = Q_{4,-1}^{(0)}(x) - x\beta_{5,-1}^{(0)}Q_{4,-1}^{(1)}(x)$$

$$Q_{(5,-1)_1}^{(0)} = -193.5057x^5 - 151.9826x^4 + 43.5119x^3 + 13.4917x^2 - 7.0526x + 1$$

$Q^{(1)}_{(4,0)_1} = Q^{(1)}_{4,-1}(x) - x\beta^{(1)}_{4,0}Q^{(2)}_{(3,0)_1}(x) = -160.4664x^4 - 49.9816x^3 + 48.0636x^2 - 11.1816x + 1.$

According to these relations, we compute $\beta^{(i)}_{m,n}$

$$\begin{cases} \beta^{(1)}_{4,2} = \dfrac{\alpha^{(1)}_{5,1}}{\alpha^{(2)}_{4,2}} = 9.147103 \\ \alpha^{(2)}_{4,2} = \delta^{(2)}_{4,1} - \delta^{(3)}_{3,2}\beta^{(2)}_{3,2} = 0.00014654 \\ \alpha^{(1)}_{5,1} = \delta^{(1)}_{5,0} - \delta^{(2)}_{4,1}\beta^{(1)}_{4,1} = 0.02813. \end{cases} , \begin{cases} \beta^{(0)}_{5,1} = \dfrac{\alpha^{(0)}_{6,0}}{\alpha^{(1)}_{5,1}} = 8.08036 \\ \alpha^{(0)}_{6,0} = \delta^{(0)}_{6,-1} - \delta^{(1)}_{5,0}\beta^{(0)}_{5,0} = 0.0108314 \end{cases}$$

and

$$\begin{cases} \beta^{(0)}_{5,2} = \dfrac{\alpha^{(0)}_{6,1}}{\alpha^{(1)}_{5,2}} = 0.018490599 \\ \alpha^{(0)}_{6,1} = \gamma^{(0)}_{6,1} - \gamma^{(1)}_{5,1}\beta^{(0)}_{5,1} = -10.658912 \\ \alpha^{(1)}_{5,2} = \gamma^{(1)}_{5,1} - \gamma^{(2)}_{4,2}\beta^{(1)}_{4,2} = -0.00057645034 \end{cases}$$

Finally, the desired quasi-orthogonal polynomials $Q_{(5,2)_1}(x)$ and $Q_{(5,2)_1}(x)$ are

$$Q_{(5,2)_1}(x) = 1008.944894x^5 + 284.53801x^4 - 347.614x^3 + 103.46207x^2 - 15.05892x + 1$$

$$Q^{(0)}_{(5,2)_2}(x) = 945662.918x^2 - 301969.105003x + 28227.0101$$

According to (2.20) and (2.21), the vector Padé approximants $\widetilde{Q}_{(5,2)_1}$ and $\widetilde{Q}_{(5,2)_2}$ are shown as

$$\widetilde{Q}_{(5,2)_1}(x) = 1 + 0.2820x - 0.3445x^2 + 0.1025x^3 - 0.0149x^4 + 9.9113 \times 10^{-4}x^5$$

$$\widetilde{Q}^{(0)}_{(5,2)_2}(x) = 937.2791 - 299.2920x + 27.9768x^2.$$

5 Reduction model using Padé approximation

The modeling of phenomena or physical systems often requires representations by mathematical models that will be simulated, analyzed or controlled. Mathematical models are represented by dynamical systems that can be with high-order, which makes the task of mathematicians more complicated given the capacity computer storage space and execution time. A number of techniques for the reduction of large-scale dynamical systems have been proposed, among them we find Proper Orthogonal Decomposition [10, 10], Balanced Truncation [44, 38], Interpolatory methods including methods referred to as Rational Krylov Methods [25]. There exist also methods based on the Routh stability criterion and the Padé approximation technique[56].

In this chapter, we propose to apply the Padé approximation technique for reduction in order of high-order systems. Let consider the single-machine infinite-bus (SMIB) power system as an application of the algorithms proposed in this thesis and we will see how to reduce the order of this system.

5.1 The problem setting

The principle of the reduction of linear models using the Padé approximation can be defined as follows :

Let consider the linear system LTI :

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t), \end{cases} \tag{5.1}$$

where $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$ and $D \in \mathbb{R}^{p\times m}$, $x \in \mathbb{R}^n$, $y \in \mathbb{R}^p$ and $u \in \mathbb{R}^m$. If $n = m = 1$, the dynamic system is called a single-input, single-output (SISO) system, otherwise it is called a multi-input, multi-output (MIMO) system.

This representation is not unique, one can use the representation by the transfer function given by :

$$H(s) = C(sI_n - A)^{-1}B + D \tag{5.2}$$

The goal is to determine a simpler, lower-order LTI dynamical system than the original system :

$$\begin{cases} \dot{x}_r(t) &= A_r x_r(t) + B_r u(t) \\ y_r(t) &= C_r x_r(t) + D_r u(t) \end{cases}$$

where $A_r \in \mathbb{R}^{r\times r}$, $B_r \in \mathbb{R}^{r\times m}$, $C_r \in \mathbb{R}^{p\times r}$ and $D_r \in \mathbb{R}^{p\times m}$, $r \ll n$ having the reduced transfer function is written :

$$H_r(s) = C_r(sI_r - A_r)^{-1}B_r + D_r,$$

the Padé approximation makes it possible to approximate the transfer function of the high order system by a lower order transfer function equalizing their moments in s_0.

The transfer function of a linear system is a rational function :

$$H(s) = \frac{\sum_{i=0}^{m} a_i s^i}{\sum_{j=0}^{n} b_j s^j}, \tag{5.3}$$

where

$$H(s) = m_0 + m_1(s - s_0) + m_2(s - s_0)^2 \dots \tag{5.4}$$

such that the m_i are the moments of the system (5.1) at s_0, which are written :

$$m_i = \frac{d^i H}{ds^i}(s_0) \quad \forall i = 0, \dots,$$

a lower order model represented by the reduced transfer function which is also a rational function of the following form :

$$H_r(s) = \frac{\sum\limits_{i=0}^{k} d_i s^i}{\sum\limits_{j=0}^{r} e_j s^j}, \quad \forall j < r \text{ et } r < n$$

where

$$H_r(s) = \widehat{m}_0 + \widehat{m}_1(s - s_0) + m_2(s - s_0)^2 \ldots + m_r(s - s_0)^r, \tag{5.5}$$

and

$$\frac{d^i H_r}{ds^i}(s_0) \quad \forall i = 0, \ldots, r,$$

so that the reduced order model corresponds to a certain number of moments

$$m_i = \widehat{m}_i, \quad \forall i = 1, \ldots, r$$

we put $e_0 = 1$:

$$\begin{cases} d_0 = & c_0 \\ d_1 = & c_1 + c_0 e_1 \\ d_2 = & c_2 + c_0 e_1 + c_1 e_2 \\ \vdots = & \vdots \\ d_k = & c_k + c_{k-1} e_1 + \ldots + c_1 e_k + c_0 e_{k+1} \\ 0 = & c_{k+1} + c_k e_1 + \ldots + c_1 e_k + c_0 e_{k+1} \\ \vdots = & \vdots \\ 0 = & c_{r-1} + c_{r-2} e_1 + \ldots + c_1 e_{r+1} + c_0 e_r \end{cases} \tag{5.6}$$

this system of equations , allows us to calculate the polynomial values in numerator and denominator of transfer function $H_r(s)$.

These e quations remind us of linear systems used to find the Padé approximant, so, Using the Padé approximation we will try to construct an approximation of the function $H(s)$, the transfer function with lower order than n, which allows us to obtain a simpler system, following the principle of this approximation we can approach $H(s)$ by a fractional function noted $(p/q)_H$ whose power series expansion around s_0 coincides with that of $H(s)$ up to the order $p + q$. This approximant is none other than the reduced order transfer function.

5.2 Presentation of Power system

Let consider the single-machine infinite-bus (SMIB) power system illustrated in the schema in Figure 2 as an illustrative application. The machine is supplying power through a step-up transformer and a high-voltage transmission line to an infinite grid. X_T and X_L represent the reactance of the transformer and the transmission line respectively ; V_T and V_B are the generator terminal and infinite bus voltage, respectively.[33]

The system consists of three-phase 160-MVA synchronous machine with automatic excitation control system. The simple synchronous machine model was developed by Heffron and Phillips. [55]

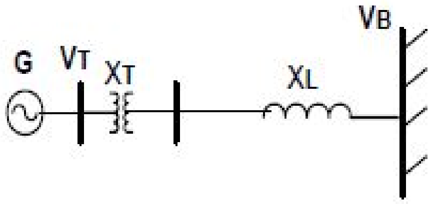


FIGURE 2 – The single-machine infinite-bus (SMIB) power system

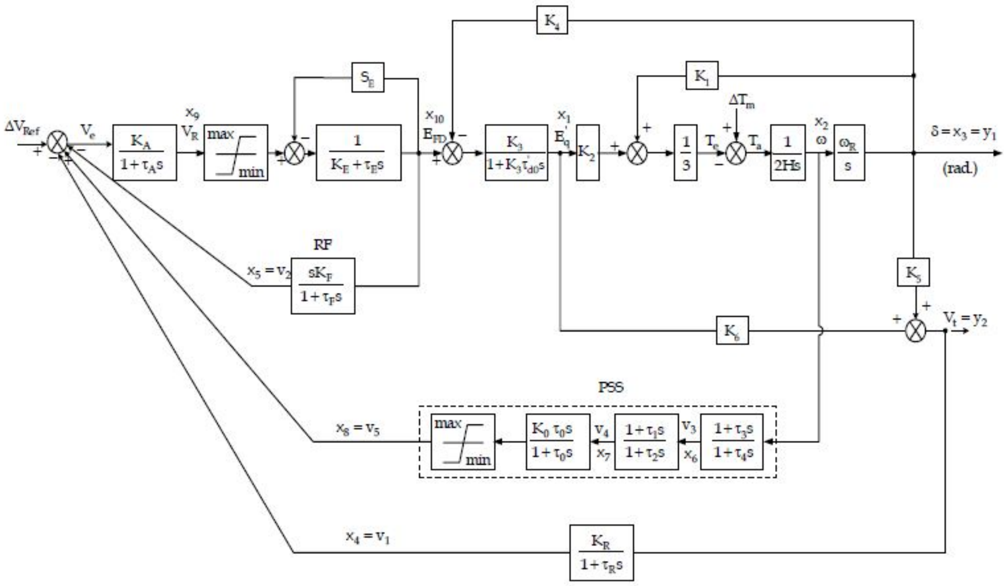


FIGURE 3 – Block diagram representation of Philips-Heffron model of single-machine infinite bus (SMIB)power system)

The system of Fig.3 may be described in the state-space form as :

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx \end{cases}$$

with, $x^T = \begin{bmatrix} E_q & \omega & \delta & v_1 & v_2 & v_3 & v_4 & v_5 & v_R & E_{FD} \end{bmatrix}$; $u^T = [\Delta V_{Ref} \ \ \Delta T_m]$; $y^T = [\delta \ \ V_t]$.
For more details see [28, 29] with

$A =$

$$\begin{bmatrix} -0.5517 & 0 & -0.3091 & 0 & 0 & 0 & 0 & 0 & 0 & 0.195 \\ -0.041 & 0 & -0.035 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 314.1593 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 9.554 & 0 & -0.866 & -20 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0.0421 & -0.0328 \\ -0.1962 & 10.8696 & -0.1672 & 0 & 0 & -10.8696 & 0 & 0 & 0 & 0 \\ -0.9386 & 51.9849 & -0.7999 & 0 & 0 & -41.1153 & -10.869 & 0 & 0 & 0 \\ -0.9386 & 51.9849 & -0.7999 & 0 & 0 & -41.1153 & -10.869 & -0.1 & 0 & 0 \\ 0 & 0 & 0 & -1000 & -1000 & 0 & 0 & 1000 & -20 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0526 & -0.8211 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 \\ 0 & 0.0926 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0.4428 \\ 0 & 2.1179 \\ 0 & 2.1179 \\ 1000 & 0 \\ 0 & 0 \end{bmatrix}, \ C = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.4777 & 0 & -0.0433 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

According to (5.2), the transfer function matrix (based on the numerical values of the matrices A, B and C) of the 10^{th} order two-input two-output linear time invariant model of practical power system under study is given by :

$$G(s) = \frac{1}{D} \begin{bmatrix} a_1^1(s) & a_1^2(s) \\ a_2^1(s) & a_2^2(s) \end{bmatrix}$$

where : $D(s) = s^{10} + 64.21s^9 + 1596s^8 + 1.947 \times 10^4 s^7 + 1.268 \times 10^5 s^6 + 5.036 \times 10^5 s^5 + 1.569 \times 10^6 s^4 + 3.24 \times 10^6 s^3 + 4.061 \times 10^6 s^2 + 2.905 \times 10^6 s + 2.531 \times 10^5$

$a_1^1(s) = -2298s^5 - 9.845 \times 10^5 s^4 - 1.376 \times 10^6 s^3 - 6.838 \times 10^6 s^2 - 6.1 \times 10^6 s - 5.43 \times 10^5$,

$a_2^1(s) = 85.23s^7 + 3651s^6 + 5.208 \times 10^4 s^5 + 2.98 \times 10^5 s^4 + 8.471 \times 10^5 s^3 + 3.105 \times 10^6 s^2 + 2.752 \times 10^6 s + 2.45 \times 10^5$

$a_1^2 = 29.09s^8 + 1868s^7 + 4.61 \times 10^4 s^6 + 5.459 \times 10^5 s^5 + 3.185 \times 10^6 s^4 + 8.702 \times 10^6 s^3 + 1.206 \times 10^7 s^2 + 7.606 \times 10^6 s + 6.483 \times 10^5$,

$a_2^2(s) = -1.26s^8 - 85.18s^7 - 2089s^6 - 2.568 \times 10^4 s^5 - 1.909 \times 10^5 s^4 - 7.123 \times 10^5 s^3 - 1.084 \times 10^6 s^2 - 2.972 \times 10^5 s - 1.942 \times 10^4$

5.3 Numerical simulation

First, we will apply the Algorithm ROPA(algorithm 1) for computing the orthogonal polynomials which are the denominator of scalar Padé approximant and we calculate the numerator of this approximant. The transfer function G is a matrix then we calculate the approximant of each component.

We propose to give the 4^{th} order reduced transfer function

$$G_4(s) = \begin{bmatrix} g_{11}(s) & g_{12}(s) \\ g_{21}(s) & g_{22}(s) \end{bmatrix}, \tag{5.7}$$

let compute the scalar Padé approximants of $g_{11}(s), g_{12}(s), g_{21}(s)$, and $g_{22}(s)$, using the algorithme 1, we obtain the following orthogonal polynomials

$$\tilde{Q}_{41} = 0.391s^4 + 4.4375s^3 + 5.391s^2 + 3.5378s + 1$$

$$\tilde{Q}_{42}(s) = 0.4952s^4 + 5.5332s^3 + 6.1985s^2 + 3.7191s + 1$$
$$\tilde{Q}_{43}(s) = 0.1692s^4 + 2.1706s^3 + 4.9501s^2 + 1.6460s + 1$$
$$\tilde{Q}_{44}(s) = -2.0105s^4 - 21.032s^3 - 9.9087s^2 - 5.2527s + 1$$

Then the denominators of scalar Padé approximants [3/4] are

$$Q_{41}(s) = 2.557s^4 + 11.3491s + 14.422s^2 + 9.04808 + 1$$

$$Q_{42}(s) = 2.0202s^4 + 7.5102s^3 + 12.517s^2 + 1.173s + 1$$
$$Q_{43}(s) = 5.91016s^4 + 9.7281s^3 + 29.255s^2 + 12.828s + 1$$
$$Q_{43}(s) = -0.49s^4 + 2.612s^3 + 4.928s^2 + 10.461s + 1$$

From these polynomials polynomials numerator are computed

$$P_1(s) = -2.1454 - 18.8887s - 24.8057s^2 - 6.5784s^3$$

$$P_2(s) = 0.968 + 0.8983s + 11.2968s^2 + 4.6406s^3$$
$$P_3(s) = 2.5614 + 23.8281s + 41.9077s^2 + 31.8699s^3$$
$$P_4(s) = -0.0767 - 0.9878s - 3.4451s^2 - 2.9984s^3$$

Consequently , we have reduced the order of the original system to the 4^{th} order.

Let's consider the following form of the original transfer function

$$G(s) = [G_1(s), G_2(s)] \tag{5.8}$$

with $G_1(s) = \begin{bmatrix} g_{11}(s) \\ g_{12}(s) \end{bmatrix}$, and $G_2(s) = \begin{bmatrix} g_{21}(s) \\ g_{22}(s) \end{bmatrix}$

we propose to give vector Padé approximants [1/2], which give a reduced transfer function of order 4, of $G_1$and G_2. Using the proposed algorithm Algorithm 5, we compute the quasiorthogonal polynomials associated to the vector Padé approximants studied in this work

For the vector G_1 we have

$$\tilde{Q}^1_{2,1}(x) = \begin{bmatrix} 1.0078s^2 + 1.3988s + 1 & 0 \\ -0.041s - 0.0633 & 1.0078s^2 + 1.3988s + 1 \end{bmatrix}$$

and for G_2, we obtain

$$\tilde{Q}^2_{2,1}(x) = \begin{bmatrix} 1 - 8.6025s^2 - 16.2348s & 0 \\ -7.8921s - 14.2896 & 1 - 8.6025s^2 - 16.2348s \end{bmatrix}$$

The reduced system of order 4 is computed from the approximant with denominator of degree 2 , so the denominator matrix is giving

$$Q^1_{2,1}(s) = \begin{bmatrix} 1 + 1.3879s + 0.99226s^2 & 0 \\ -0.04068 - 0.06281s & 1 + 1.3879s + 0.99226s^2 \end{bmatrix}$$

with vector numerator

$$P^1_1(s) = \begin{bmatrix} -2.1454 - 2.454s \\ 1.0553 + 1.2198s \end{bmatrix}.$$

and the approximant of vector $G_2(s)$

$$Q^2_{2,1}(s) = \begin{bmatrix} -0.1162s^2 - 1.8872s + 1 & 0 \\ 1.6611 + 0.9174s & -0.1162s^2 - 1.8872s + 1 \end{bmatrix},$$

with

$$P^2_1(s) = \begin{bmatrix} 2.5614 - 4.1819s \\ -0.3744 + 2.1253s \end{bmatrix}.$$

The computation of Padé approximant with denominator degree is 2 is due to the calculation of inverse of the matrix polynomial $Q_{2,1}$ which give a transfer function of order 4 .

Conclusion and Perspectives

In order to compute recursively a sequence of Padé approximant of scalar or non scalar function, we can use the link with the theory of orthogonal polynomials and quasi orthogonal polynomials.

We are focused in this thesis on the recursive computation of orthogonal polynomials and quasiorthogonal polynomials we proposed the first algorithm called ROPA : Recursive Orthogonal Polynomial Algorithm, as an extension of this work we obtain a recurrence schemes for computing quasiorthogonal polynomials.

In the second chapter, we present the essential results used throughout this work starting with the definition of scalar and vector approximants and passing to the introduction of the properties of Schur's complements by showing the link with Padé approximants.

In the third chapter, the Schur complements, the Sylvester identity and qd algorithm to derive a recursive algorithm for computing the orthogonal polynomials without referring to the relations of orthogonality. We recovered the recurrence relation between the Hankel determinants when we apply the Sylvester identity to the expression of Hankel determinant. This relation helped us to give another recursive algorithm to calculate the orthogonal polynomials. We have also gave an algorithm to calculate the orthogonal polynomials of the same family by using the Sylvester and auxiliary polynomials.

In the fourth chapter, we present a recursive algorithm to compute adjacent families of quasi-orthogonal polynomials of the form (4.3). New short-term recurrence relations involving multiple families of such polynomials with lower block determinants are introduced. An illustrative example with applications to vector Padé approximations is given.

The fifth chapter is an application of the algorithms proposed in this thesis to a single-machine infinite-bus(SMIB) power system.

In the next works, we wish to improve our algorithm for calculating quasi polynomials and we are going to give an extension of the Qd-algorithm in our case.

In another axis, we will focus our attention to the theory of tensors, exploiting the results obtained in this work since we find that in the case of quasiorthogonal polynomials, the elements of each adjacent family can be organized in a table and related to two other different table of families of quasiorthogonal polynomials.

As an extension of Sadaka's work [48], we propose in future works to define a Padé approximants of matrix function of the form :

$$\mathbf{F} = \begin{bmatrix} F_1 & 0_{\mathbb{R}^2} \\ F_2 & F_1 \end{bmatrix} \tag{5.9}$$

with $F_1 = \begin{bmatrix} f_1^1 \\ f_2^1 \end{bmatrix}$, $F_2 = \begin{bmatrix} f_1^2 \\ f_2^2 \end{bmatrix}$.
Then, we consider the block matrix polynomials :

$$\mathbf{Q} = \begin{bmatrix} Q_1(t) & 0_{\mathbb{R}^4} \\ Q_2(t) & Q_1(t) \end{bmatrix}, \tag{5.10}$$

with $Q_1 = \begin{bmatrix} Q_1^1 & 0 \\ Q_2^1 & Q_1^1 \end{bmatrix}$, $Q_2 = \begin{bmatrix} Q_1^1 & 0 \\ Q_2^2 & Q_1^2 \end{bmatrix}$,

we have also the numerator

$$\mathbf{P} = \begin{bmatrix} P_1 & 0_{\mathbb{R}^2} \\ P_2 & P_1 \end{bmatrix} \tag{5.11}$$

$P_1 = \begin{bmatrix} P_1^1 \\ P_2^1 \end{bmatrix}$, $P_2 = \begin{bmatrix} P_1^2 \\ P_2^2 \end{bmatrix}$

Let

$deg(Q_1) = u_1 = \begin{bmatrix} u_1^1 \\ u_2^1 \end{bmatrix}$, $deg(Q_2) = u_2 = \begin{bmatrix} u_1^2 \\ u_2^2 \end{bmatrix}$ and $deg(P_1) = \begin{bmatrix} v_1^1 \\ v_2^1 \end{bmatrix}$, $deg(P_2) = \begin{bmatrix} v_1^2 \\ v_2^2 \end{bmatrix}$.

Then we have :

$$\mathbf{Q}(t)\mathbf{F}(t) - \mathbf{P}(t) = O(t^{W+1}), \tag{5.12}$$

with $W = \begin{bmatrix} w^1 \\ w^2 \end{bmatrix} = \begin{bmatrix} w_1^1 & w_2^1 \\ w_1^2 & w_2^2 \end{bmatrix}$, and $Q(0) = I_{4\times 4}$.

$$\begin{cases} Q_1(t)F_1(t) - P_1 = O(t^{w_1^1+1}) \\ Q_1(t)F_2(t) + Q_2(t)F_1(t) - P_2(t) = O(t^{w_2+1}). \end{cases}$$

Bibliographie

[1] A.Salam, : Vector Padé-type Approximants and Vector Padé Approximants. Journal of Approximation Theory Volume 97, Issue 1. 92–112(1999)

[2] A.Draux, Polynômes orthogonaux formels, applications, Springer-Verlag, Heidelberg,1983.

[3] A.Messaoudi, , H.Sadok, , R. Sadaka, New algorithm for computing the Hermite interpolation polynomial. Numerical Algorithms Journal. 77,1069–1092(2018)

[4] A. Draux, Polynômes orthogonaux formels-Application. Springer-Verlag Berlin Heidelberg New York, 1983.

[5] A. Messaoudi, Matrix recursive interpolation algorithm : a formalism for linear equations-II : iterative methods. Journal of Computational and Applied Mathematics, 76(1996), 31–53 .

[6] A. Messaoudi, Recursive interpolation algorithm : a formalism for linear equations-I : Direct methods. Journal of Computational and Applied Mathematics, 76(1996), 13–30 .

[7] A. Messaoudi, H. Sadok, Recursive polynomial interpolation algorithm (RPIA). Numerical Algorithms Journal, 76(2017), 675–694.

[8] A. Messaoudi, Some properties of the recursive projection and interpolation algorithms. IMA Journal of Numerical Analysis, 15(1995), 307-318.

[9] A.Draux, On quasi-orthogonal polynomials of order r. Taylor and Francis online, Integral Transforms and special functions (2016).

[10] A.Har, J.T.Borggaard, and D.Pelletier, Local improvements to reducedorder models using sensitivity analysis of the proper orthogonal decomposition. Journal of Fluid Mechanics, 629 :41–72(2009).

[11] C.G.Jacobi, uber die Dastellung einer Reite.e.gegebener Werthe durch eine gebrochen rationale Funktion, J.Reine.Angew. Math,30(1845), 127-156.

[12] C.Brezinski and M.Redivo.Zagalia and H.Sadok, Avoiding breakdown and near-breakdown in Lanczos type algorithm, Numerical Algor.1(1991) 261-274.

[13] C.Brezinski and M.Redivo.Zagalia and H.Sadok, A breakdown-free Lanczos type algorithm for solving linear system, Numer.Math.63(1992) 29-38.

[14] C.Brezinski and M.Redivo.Zagalia, Treatment of near- breakdown in the GS algorithm, Numerical algorithm,7(1994) 33-73.

[15] C. Brezinski, : Padé-type approximation and general orthogonal polynomials. Birkhauser-Verlag Basel AG(1980)

[16] C.Brezinski, : A general extrapolation algorithm. Numerische Mathematik. 35, 175–187(1980)

[17] C.Brezinski, : Other manifestations of the Schur complement. Linear Algebra and Its Applications. 111, 231–247(1988)

[18] C.Brezinski, : Vector transformations. methodology and applications to linear systems. Journal of Computational and Applied Mathematics. 98, 149-175(1998)

[19] C. Brezinski, Recursive interpolation, extrapolation and projection. Journal of Computational and Applied Mathematics, 9(1983), 369–376.

[20] C. Brezinski, Some determinantal identities in a vector space, with applications. In Werner, H. Bunger, H.J. (eds.), Padé Approximation and its Applications, Bad-Honnef, 1983, Lecture Notes in Mathematics, Springer, Berlin, volume 1071(1984), 1–11.

[21] C. Brezinski, M. Redivo Zaglia, A Schur complement approach to a general extrapolation algorithm. Linear Algebra and Its Applications, 368(2003), 279–301.

[22] C. Brezinski, H. Sadok and M. Redivo Zaglia, Orthogonal polynomials and the Lanczos method. Numerical Analysis and Mathematical Modelling Banach Center Publications, volume 29, 1994.

[23] C.Brezinski, M. Redivo–Zaglia, The genesis and early developments of Aitken's process, Shanks' transformation, the ε–algorithm, and related fixed point methods. Journal of Numerical Algorithms(2018).

[24] C. Brezinski, K.A. Driver, M. Redivo-Zaglia, Quasi-orthogonality with applications to some families of classical orthogonal polynomials. Applied Numerical Mathematics 48,157–168 (2004).

[25] C.Beattie and S.Gugercin, Model Reduction by Rational Interpolation.arXiv :1409.2140v1(2014).

[26] D.SHANKS, "Nonlinear transformations of divergent and slowly convergent sequences", J, Math. and Phys. (Cambridge Mass.), 34, 1-42 (1955).

[27] D.V. Ouellette, Schur complements and statistics. Linear Algebra and Its Applications, 36(1981), 187–295.

[28] D.Papadoplolos, order reduction of linear system models with a time-frequency domain method. Journal of the Franklin Institute , Vol 322, No. 4, pp. 209-220(1986).

[29] D.Papadoplolos , Test and simulations in connection with large generator dynamics. Proceedings of the ICEM'82 Conference, Budapest, Part 3, 881-884(1982).

[30] E.Arian, M.Fahl, E.Sachs, Trust-region proper orthogonal decomposition models by optimization methods. In Proceedings of the 41st IEEE Conference on Decision and Control, pp 3300–3305, Las Vegas, NV(2002).

[31] Guo-liangXu, AdhemarBultheel, Matrix padé approximation : definitions and properties. Linear Algebra and its Applications 137/138 :67-136(1990).

[32] Guo-liangXu, AdhemarBultheel, Matrix Padé Approximation : Recursive Computations. Journal of Computational and Applied Mathematics Vol.10, No.3,254-262(1992).

[33] G.Parmar, S.Mkherjee, R.Prasad, Reduced order modelling of linear MIMO system using genetic algorithm. International Journal of Simulation Modelling 6(3),173-184 (2007).

[34] H.Padé, : Sur la représentation approchée d'une fonction par des fractions rationnelles. Ann. Sci. Ecole Norm. Sup. 9(3), 1-93(1892)

[35] H.Padé, : Sur la géneralisation des fractions continues algébriques. J. Math. Pures Appl. (4) 10, 291-329(1894)

[36] H. Gutknecht Martin, N. Parlett Beresford, From qd to LR, or, How were the qd and LR algorithms discovered ?. IMA Journal of Numerical Analysis, 13(2009), 1-13.

[37] H. Rutishauser, Ein kontinuierliches Analogon zum Quotienten Differenzen Algorithmus. Archiv der Mathematik(Basel), 5(1954), 132-137.

[38] H.Antil, M.Heinkenschloss, and R.H.W. Hoppe. Domain decomposition and balanced truncation model reduction for shape optimization of the Stokes system. Optimization Methods and Software, 26(4–5) :643–669, (2011).

[39] I. Schur, Potenzreihn im Innern des Einheitskreises. Journal für die reine und angewandte Mathematik, 147(1917), 205–232.

[40] J.L.Lagrange, Sur l'usage des fractions continues dans le calcul intégral, Nouveaux Mém. Acad.Sci. Berlin, 7(1776), 236-264, Œuvre 4(1869),301-322.

[41] J.V.Iseghem-Le Luyer, : Approximants de Padé Vectoriels. Thèse présenté à l'université des sciences et techniques de Lilles Flandres Artois(1987)

[42] K.Jbilou, , A.Messaoudi, , Tabaâ, K. : Some Schur complement identities and applications to matrix extrapolation methods. Linear Algebra and Its Applications. 392, 195–210(2004)

[43] K. Jbilou, A. Messaoudi, Matrix recursive interpolation algorithm for block linear systems Direct methods. Linear Algebra and Its Applications, 294(1999), 137–154.

[44] M.Heinkenschloss, D.C.Sorensen, and K. Sun. Balanced truncation model reduction for a class of descriptor systems with application to the oseen equations. SIAM J. Sci. Comput., 30(2) :1038–1063(2008).

[45] M.H.Ggutknecht, The unsymmetric Lanczos algorithm and their relations to Padé approximation, continued fractions,and the qd-algorithm : Proceeding of the Copper Mountain Conference on Iterative Methods(1990).

[46] M.Riesz, Sur le problème des moments. Troisième note, Arkiv för Matematik, Astronomi och Fysik.17 :1–52(1923) .

[47] P. Maroni, Prolégomènes à l'étude des polynômes orthogonaux semi-classiques. Annali di Matematica Pura ed Applicata , 149,165–184 (1987).

[48] R.Sadaka, : Padé approximation of vector functions. Applied Numerical Mathematics. 21, 57-70 (1996)

[49] R.Sadaka, : An algorithm for constructing a class of Padé approximants of vector functions. Applied Numerical Mathematics 24, 483-499(1997)

[50] S.Sidki, , Sadaka, R., Benazzouz, A. : Computing recursive orthogonal polynomial with Schur complements. Journal of Computational and Applied Mathematics(2019).

[51] T.Ando, : Generalized Schur complements. Linear Algebra and Its Applications. 27, 173–186(1979)

[52] T.J.Stieltjes, Recherches sur les fractions continues Ann.Fac.Toulouse 8(1894) 1-122.

[53] T.S. Chihara, On Quasi-Orthogonal Polynomials. American Mathematical Society, Vol.8, No.4, pp 765-767 (Aug., 1957).

[54] T.S.Chihara , An Introduction to Orthogonal Polynomials. New York (NY) : Gordon and Breach,1978.

[55] W.G.Heffron, R.A.Philips, Effect of modern amplidyne voltage regulator on underexcited operation of large turbine generators, AIEE Transactions , Vol.PAS 1, pp 692-697(1952).

[56] Y.Shamash, Model reduction using the Routh stability criterion and the Padé approximation technique. International Journal of Control, 1975, VOL. 21, No.3, 475-484(2013).

ROYAUME DU MAROC
Université Mohammed V - RABAT -
جامعة محمد الخامس – الرباط –
Faculté des sciences
كلية العلوم

CENTRE D'ETUDES DOCTORALES - SCIENCES ET TECHNOLOGIES

Résumé

L'approximation de Padé permet de résoudre plusieurs problèmes dans des domaines différents, notamment dans l'analyse numérique l'approximation de Padé donne de bon résultats dans les problèmes d'accélération de la convergence. Le calcul de ces approximants a fait l'objectif de plusieurs travaux, dans cette thèse nous allons proposer des algorithmes de calcul des polynômes orthogonaux et des polynômes quasiorthogonaux afin de déterminer les approximants de Padé dans le cas scalaire et vectoriel. Dans un premier temps, nous donnons une relation de récurrence entre deux familles de polynômes orthogonaux. Nous dérivons deux algorithmes récursifs pour calculer les polynômes orthogonaux en utilisant l'algorithme qd et la relation de récurrence bien connue des déterminants de Hankel. Nous donnons également une relation de récurrence utile entre les polynômes orthogonaux de la même famille et auxiliaires polynômes, ce qui nous permet de construire un troisième algorithme pour calculer récursivement les polynômes orthogonaux. Dans le cas vectoriel, nous étendons le premier travail au cas vectoriel. La théorie de l'approximation vectorielle de Padé étudiée dans notre travail est liée à la théorie des polynômes quasi-orthogonaux. Le calcul récursif des approximants vectoriels de Padé revient à calculer récursivement les polynômes quasiorthogonaux. Nous introduisons une procédure efficace pour calculer des familles adjacentes de polynômes quasi-orthogonaux. L'algorithme dérivé utilise des relations récursives courtes dont les coefficients sont écrits en termes de déterminants de blocs triangulaires inférieurs. La stratégie de calcul de tels déterminants est donnée et analysée.

Mots-clefs (5): Approximants de Padé, Polynômes Orthogonaux, Approximants de Padé vectoriels, Polynômes quasi-orthogonaux, Compléments de Schur.

Abstract

The Padé approximation solves several problems in domains different, in particular in the numerical analysis the approximation of Padé gives good results in the problems of acceleration of convergence. The calculation of these approximants has been the objective of several works, in this thesis we will propose algorithms for calculating orthogonal polynomials and quasi-orthogonal polynomials in order to determine the Padé approximants in the scalar and vector case. To do this, we will start by recalling the Padé approximation and the orthogonal polynomials. We give a recurrence a recurrence relation between two families of orthogonal polynomials. We derive two recursive algorithms for computing the orthogonal polynomials by using the qd algorithm and the well known recurrence relation of Hankel determinants. We also give a useful recurrence relation between the orthogonal polynomials of the same family and auxiliary polynomials, which allows us to build a third algorithm for computing recursively the orthogonal polynomials. In the second part, we extend the first work to the vector case. The theory of vector Padé approximation studied in our work is related to the theory of quasiorthogonal polynomials. The recursive calculation of vector Padé approximants is equivalent to calculate recursively the quasiorthogonal polynomials. introduces an efficient procedure to compute adjacent families of quasi-orthogonal polynomials. The derived algorithm uses short recursive relations whose coefficients are written in terms of lower triangular block determinants. The strategy of computing such determinants is given and analyzed.

Key Words (5): Padé approximants, Orthogonal polynomials, vector Padé approximants, quasi-orthogonal polynomials, Schur complements.

✉ Faculté des Sciences, avenue Ibn Battouta, BP. 1014 RP, Rabat –Maroc
☎ 00212 (05) 37 77 18 76 🖷 00212(05) 37 77 42 61; http://www. fsr.um5.ac.ma