A new formal approach to the rational interpolation problem

Marc Van Barel
Adhemar Bultheel

Report TW 134, July 1990

Department of Computer Science, K.U. Leuven

Abstract

An elegant and fast recursive algorithm is developed to solve the rational interpolation problem in a complementary way compared to existing methods. We allow confluent interpolation points, poles, and infinity as one of the interpolation points. Not only one specific solution is given but a nice parametrization of all solutions. We also give a linear algebra interpretation of the problem showing that our algorithm can also be used to handle a specific class of structured matrices.

Keywords : rational interpolation, structured matrices, recursive algorithms.
AMS(MOS) Classification : Primary : 65D15, Secondary : 41A20, 41A05, 41A21.

A new formal approach to the rational interpolation problem

Marc Van Barel and Adhemar Bultheel

Department of computer science

K.U. Leuven

July 27, 1990

Abstract

An elegant and fast recursive algorithm is developed to solve the rational interpolation problem in a complementary way compared to existing methods. We allow confluent interpolation points, poles, and infinity as one of the interpolation points. Not only one specific solution is given but a nice parametrization of all solutions. We also give a linear algebra interpretation of the problem showing that our algorithm can also be used to handle a specific class of structured matrices.

1 Introduction

In an earlier paper [26], we have introduced a new approach to solve the rational interpolation problem. This was done in a rather informal way. In this publication we want to describe this new approach in a rather formal manner.

Before we explain the advantages of the new approach, let us first introduce what rational interpolation is about. In the rational interpolation problem (**RIP**), one looks for a rational function $n(z)/d(z)$ satisfying some interpolation conditions. These interpolation conditions could be “proper”

$$n(z_i)/d(z_i) = l_i/m_i, \quad i = 1, 2, \dots, p,$$

or they could be “linearized”

$$l_i d(z_i) - m_i n(z_i) = 0, \quad i = 1, 2, \dots, p,$$

where $z_i, i = 1, 2, \dots, p$ are the interpolation points and l_i/m_i the corresponding function values. In the literature on rational interpolation, the Newton-Padé rational interpolation problem (**NPRIP**) plays a central role. The **NPRIP** works with linearized interpolation conditions and links the number of these conditions p to the maximum degrees of the numerator and denominator polynomials $n(z)$ and $d(z)$. More precisely, if δ_n and δ_d are the maximum allowable degrees of $n(z)$ and $d(z)$ respectively, then $p = \delta_n + \delta_d + 1$. In this way, the **NPRIP** always has a solution and the solutions are unique up to common factors in the numerator $n(z)$ and denominator polynomial $d(z)$. This unique solution is called the (δ_n, δ_d) Newton-Padé approximant (**NPA**). The rational function $n(z)/d(z)$, however, will not necessarily satisfy all of the proper rational interpolation conditions. The interpolation points where this problem occurs, are called unattainable points. A lot of attention has been

given to this problem in the literature [21]. Note that we have assumed that all interpolation points z_i are different, although the **NPRIP** also allows multiple conditions at the same interpolation point. When all the interpolation points are the same, we get the Padé approximation problem (**PAP**) described, solved and generalized in a lot of papers and books. We mention only the early work of Padé [22], the books of Baker and Graves-Morris [5, 6] and the connection with the minimal partial realization problem [19, 20, 17, 24]. As for the **PAP**, we can arrange all **NPA**s in a two-dimensional array, called the Newton-Padé table (**NPT**), where the (δ_n, δ_d) **NPA** is placed at row δ_d and column δ_n . We make a distinction between a normal and a nonnormal **NPT**. In a normal table, all the entries are different in contrast to a nonnormal table, where the same rational function can appear more than once. The set of positions, containing the same rational function, is called a block. Not considering the special cases of the 0 or ∞ approximant, a nonnormal **NPT** shows blocks that are unions of squares in general [9, 11, 15]. For the special case of a nonnormal Padé table, the blocks are contiguous and square. Assuming normality of the **NPT** (the Padé table), a lot of fast recursive algorithms have been developed to solve the **NPRIP** (the **PAP**). These algorithms follow a path in the table until they reach the final entry containing the rational interpolant (the Padé approximant). Life becomes a lot more difficult when the algorithm has to follow a path through a nonnormal table. When a block shows up on the way, special action has to be taken. Because the blocks in the Padé table are contiguous, this special action is much more easy in this case compared to a nonnormal **NPT**, where the blocks can even be non-contiguous. Therefore, almost all of the algorithms, we know to solve the **NPRIP** are reordering the interpolation points to transform the **NPT** into one containing blocks with a simpler structure. Note that it is even possible to reorder the interpolation points resulting in blocks that are all square as in the Padé table as observed by Wuytack [33]. Since these recursive algorithms possibly reorder the given interpolation data walking through the **NPT**, they do not always allow that further data points are added, even if we skip the pivoting strategy to enhance numerical stability. Furthermore, the practical algorithms, known today, are not dividing the computational process far enough. They group several interpolation points together to allow computation of divided differences. In this way, they destroy the equality of importance between the interpolation points. Let us mention some of the most important algorithms. In [31], Werner adds the pivoting strategy of Graves-Morris [12] to his earlier algorithm [30, 28, 29]. These methods use divided differences and a reordering of the data points. Graves-Morris and Hopkins, however, develop an algorithm which could be generalized to handle new data points [13]. Each step of this algorithm handles another data point. In [15], Gutknecht gives an algorithm to construct continued fractions along a diagonal or a staircase in the **NPT** without reordering the data points. He also shows that the algorithms of Werner and of Graves-Morris and Hopkins are special cases starting with a reordered set of data points. It is a pity that the algorithm of Gutknecht requires some conditions at the initialization equivalent to a reordering of the data points. A totally different approach is followed by Wuytack [32] for the normal and by Cordellier [10] for the nonnormal **NPT**. They generalize the classical QD-algorithm [23] solving the **PAP** to the **NPRIP**.

As indicated above, the Newton-Padé approach itself allows that there are unattainable points, i.e. the rational interpolant satisfies the “linearized” but not the “proper” conditions. Therefore, we have generalized the Newton-Padé problem setting such that our new algorithm always finds a solution satisfying the “proper” interpolation conditions. This elegant, fast and recursive algorithm does not require a reordering of the data points, thus allowing to add more interpolation data during the computations. Investigating the algorithm in detail, shows that the computation of divided differences of the older algorithms is split up in more

basic steps, each connected to adding one more data point. This is similar to the distinction between the Euclidean algorithm and the Berlekamp-Massey algorithm. The generalized approach essentially means that we disconnect the number of interpolation points and the maximum allowable degrees of numerator and denominator polynomial. Instead, we look for a rational function, having least complexity with a relative importance of the numerator and denominator degree, satisfying the rational interpolation conditions. Our algorithm not only allows to compute one solution but a parametrization of all solutions both for the “linearized” and the “proper” problem, containing as a special case the **NPA**s.

Our generalized problem setting can easily be transformed to a linear algebra problem. We essentially investigate the null spaces of matrices with a special structure [16, 3]. For the special case of the **PAP**, these are Toeplitz and Hankel matrices [24, 25, 34]. The new recursive algorithm, developed here, can be seen as a generalization of the basic algorithms to solve the connected linear algebra problems for the **PAP**. We mention the Levinson-Schur-Szegő algorithms for the Toeplitz case and the Euclidean-Berlekamp-Massey-Kronecker algorithms for the Hankel case [2]. Wherever the structured matrices appear, these algorithms and the corresponding new approach can be used, e.g. in the theory and application of orthogonal polynomials, digital filtering and signal processing, time series analysis, integral equations, stochastic processes, inverse scattering, speech analysis, BCH-decoding etc. In [3], Antoulas and Anderson develop a nonrecursive method to solve the “proper” **RIP**, even giving a parametrization of all solutions. Their tool is the Löwner matrix [18, 7]. In section 3 of the same article, they derive a recursive algorithm, going along the main diagonal of the **NPT**. This algorithm can be seen as a very crude version of the method, developed here.

We have tried to make the problem setting as formal as possible. Therefore, we need a lot of preliminary notations and definitions (section 2). Compared to [26], we pose the rational interpolation problem in a much more general way, inspired by [15, 14], allowing confluent interpolation points, poles and ∞ as interpolation point (section 3). In fact, our algorithm can be seen as a Berlekamp-Massey version of the Euclidean-like algorithm of Gutknecht [15]. In section 4, the “proper” problem is “linearized”. The connection with structured matrices is established in section 5 showing that not only Löwner matrices can lead to nice results concerning the **RIP**. Solving the rational interpolation problem in the next section, results in a nice and fast recursive algorithm essentially computing nested bases for the nullspaces of the corresponding structured matrices. A parametrization is given for all solutions not only of the “linearized” problem but also of the “proper” problem. Section 7 contains a small example to clarify the ideas. In the last section, we indicate some possibilities for future research.

2 Preliminary notations and definitions

To formulate the rational interpolation problem (**RIP**), we need some introductory notations, concepts and definitions.

We are working with elements of a field $\mathbf{F}$, $+$, $\times$. This field $\mathbf{F}$ can be finite or infinite. For example, in coding theory, some problems will lead to a formulation of the rational interpolation problem with a finite field $\mathbf{F}$. On the other hand, in linear system theory, the infinite field $\mathbf{F}$ of the real or complex numbers appears in several applications.

We now have to define precisely formal series, polynomials etc. and operations on these objects to avoid all possible confusion. In the sequel we let $\mathbf{Z}$ denote the set of all integers and $\mathbf{N}$ the set of all nonnegative integers.

We start by introducing the set of formal Laurent series at the point $\alpha \in \mathbf{F}$.

Definition 2.1 (formal Laurent series) The formal Laurent series (fLs) at a fixed point α are defined and denoted as follows:

$$\mathbf{F}_\alpha\langle z, z^{-1} \rangle = \left\{ \sum_{k=-\infty}^{+\infty} f_k(z - \alpha)^k \mid f_k \in \mathbf{F}, k \in \mathbf{Z} \right\}.$$

For this set of fLs we can define the sum operation.

Definition 2.2 (sum) Taking two elements $f_1(z) = \sum_{k \in \mathbf{Z}} f_{1,k}(z - \alpha)^k$ and $f_2 = \sum_{k \in \mathbf{Z}} f_{2,k}(z - \alpha)^k$ of $\mathbf{F}_\alpha\langle z, z^{-1} \rangle$, we define the fLs $f(z) = f_1(z) + f_2(z)$ as

$$f(z) = f_1(z) + f_2(z) = \sum_{k \in \mathbf{Z}} (f_{1,k} + f_{2,k})(z - \alpha)^k.$$

Unfortunately, the product of two fLs can not be defined without trouble since the coefficients of the product will involve infinite sums which need not converge. So to define these operations, we shall have to restrict ourselves to smaller sets, e.g. the set of positive formal Laurent series (pfLs) which have only finitely many nonzero coefficients with a negative index.

Definition 2.3 (positive formal Laurent series) The set of all formal Laurent series at a fixed point α with a finite number of nonzero terms with negative powers, is called the set of the positive formal Laurent series (pfLs) at α and is denoted as follows:

$$\mathbf{F}_\alpha\langle z \rangle = \left\{ \sum_{k=\sigma}^{+\infty} f_k(z - \alpha)^k \mid f_k \in \mathbf{F}, \mathbf{Z} \ni k \geq \sigma \in \mathbf{Z} \right\}.$$

For this set of pfLs we can define a product without any trouble as follows.

Definition 2.4 (product) Let $f_1(z)$ and $f_2(z)$ be in $\mathbf{F}_\alpha\langle z \rangle$: $f_i(z) = \sum_{\sigma_i}^{\infty} f_{i,k}(z - \alpha)^k$, $i = 1, 2$. Then the product pfLs $f_1(z) \times f_2(z)$, also denoted as $f_1(z)f_2(z)$, is given by

$$f_1(z) \times f_2(z) = \sum_{k=\sigma_1+\sigma_2}^{+\infty} f_k(z - \alpha)^k$$

with

$$f_k = \sum_{i=\sigma_1}^{k-\sigma_2} f_{1,i} f_{2,k-i} = \sum_{j=\sigma_2}^{k-\sigma_1} f_{1,k-j} f_{2,j}.$$

As you can see, each coefficient f_k involves only a finite number of terms. With the sum operation inherited from the fLs and the product operation as defined above, we note that $\mathbf{F}_\alpha\langle z \rangle, +, \times$ is a ring.

What has been done for the pfLs can be repeated for fLs with only finitely many nonzero coefficients with a positive index. These are the negative formal Laurent series (nfLs). They are defined as follows.

Definition 2.5 (negative formal Laurent series) The negative formal Laurent series (nfLs) at a fixed point α are defined and denoted as follows:

$$\mathbf{F}_\alpha\langle z^{-1} \rangle = \left\{ \sum_{k=-\infty}^{\sigma} f_k(z - \alpha)^k \mid f_k \in \mathbf{F}, \mathbf{Z} \ni k \leq \sigma \in \mathbf{Z} \right\}.$$

As with the **pfLs** we can define a product for the **nfLs** and we can show that $\mathbf{F}_\alpha\langle z^{-1} \rangle, +, \times$ is a ring too.

As a special case of the **pfLs** and the **nfLs** we define formal power series.

Definition 2.6 (formal power series) We define the positive and the negative formal power series (**pfps** and **nfps**) as the corresponding **pfLs** and the **nfLs** respectively with the additional condition that $\sigma = 0$. We denote these two sets as $\mathbf{F}_\alpha[[z]]$ and $\mathbf{F}_\alpha[[z^{-1}]]$ respectively.

Note that the set of the **pfLs** can be built up by shifted **pfps**, i.e.

$$\mathbf{F}_\alpha\langle z \rangle = \bigcup_{\sigma=-\infty}^{+\infty} (z - \alpha)^\sigma \mathbf{F}_\alpha[[z]]$$

with $(z - \alpha)^\sigma \mathbf{F}_\alpha[[z]] = \left\{ \sum_{k=0}^{+\infty} f_k(z - \alpha)^{k+\sigma} \right\}$. Note that for any **pfLs** $f(z)$, there are an infinite number of ways to represent $f(z)$ as a shifted **pfps** $(z - \alpha)^\sigma f'(z)$ with $f'(z) \in \mathbf{F}_\alpha[[z]]$.

Definition 2.7 (shift representation, order) Let $f(z) \in \mathbf{F}_\alpha\langle z \rangle$ be written as

$$f(z) = (z - \alpha)^\sigma f'(z) \text{ with } f'(z) \in \mathbf{F}_\alpha[[z]].$$

Then we call this a **pfLs** shift representation of the **pfLs** $f(z)$. We call σ the order of the **pfLs** shift representation. Similar definitions can be given for **nfLs**.

Finally we introduce the set of polynomial representations at a fixed point $\alpha \in \mathbf{F}$. Each element of this set belongs at the same time to $\mathbf{F}_\alpha\langle z^{-1} \rangle$ and to $\mathbf{F}_\alpha[[z]]$.

Definition 2.8 (polynomial representation) The set of all polynomial representations at α is defined as follows:

$$\mathbf{F}_\alpha[z] = \left\{ \sum_{k=0}^{\phi} f_k(z - \alpha)^k \mid f_k \in \mathbf{F}, k = 0, 1, \dots, \phi \in \mathbf{N} \right\}.$$

At this point, we have defined several sets based on the field $\mathbf{F}, +, \times$. All these sets can be embedded as follows:

$$\mathbf{F}_\alpha[z] \subset \mathbf{F}_\alpha[[z]] \subset \mathbf{F}_\alpha\langle z \rangle \subset \mathbf{F}_\alpha\langle z, z^{-1} \rangle \supset \mathbf{F}_\alpha\langle z^{-1} \rangle = (z - \alpha)^{-1} \mathbf{F}_\alpha[[z^{-1}]] \oplus \mathbf{F}_\alpha[z] \quad (2.1)$$

Note that the set of polynomial representations can be embedded in the **pfLs** as well as in the **nfLs**. Note that the sum and the product of two polynomial representations is also a polynomial representation and the result of the operations is independent of the embedding choice, i.e. we get the same sum- and product polynomial representation in the **pfLs** embedding as in the **nfLs** embedding. We have the following property.

Theorem 2.1 All the sets in (2.1) equipped with the operations $+$ and $\times$ form a ring except the set $\mathbf{F}_\alpha\langle z, z^{-1} \rangle$ of **fLs** for which the product is not defined.

It is well known that for the **pfLs** and the **nfLs** also the inverse and therefore also the quotient is defined.

Definition 2.9 (inverse, quotient) The inverse of a **pfLs** $f(z) \in \mathbf{F}_\alpha\langle z \rangle$, $f \neq 0$ is defined as the unique $g(z) \in \mathbf{F}_\alpha\langle z \rangle$ which satisfies

$$g(z) \times f(z) = 1.$$

We denote $g(z)$ as $1/f(z)$. The quotient $f(z) = f_1(z)/f_2(z)$ of the two **pfLs** $f_1(z)$ and $f_2(z)$, with $f_2 \neq 0$, is defined as $f(z) = f_1(z) \times 1/f_2(z)$ or, equivalently as the unique **pfLs** $f(z)$ satisfying

$$f_1(z) = f_2(z) \times f(z)$$

In the same way, we can define the divide operation for the set of **nfLs** $\mathbf{F}_\alpha\langle z^{-1} \rangle$. In practice the quotient series can be computed by using a technique called “synthetic division”. This technique is a generalization of the computation of the quotient of two reals.

We now have a field structure for some of the sets defined before.

Theorem 2.2 *The sets $\mathbf{F}_\alpha\langle z \rangle$ and $\mathbf{F}_\alpha\langle z^{-1} \rangle$ equipped with the operations $+$ and $\times$ form a field.*

The polynomial representations form a subset of the **pfLs** as well as of the **nfLs** and as such also their quotient can be defined as either a **pfLs** or a **nfLs**. However, the different embeddings give different quotients. So, if we write the quotient of two polynomial representations, we shall have to indicate which embedding we use. We do this by the notation $z \xrightarrow{+} \alpha$ if we consider the “positive” embedding and $z \xrightarrow{-} \alpha$ if we take the “negative” one. For example,

$$\frac{z-2}{z-1} = 2 + z + z^2 + z^3 + \dots, \quad z \xrightarrow{+} 0$$

and

$$\frac{z-2}{z-1} = 1 - z^{-1} - z^{-2} - z^{-3} - \dots, \quad z \xrightarrow{-} 0.$$

Note that the quotient of two polynomial representations is not necessarily a polynomial representation anymore. The notation $z \xrightarrow{+} \alpha$ or $z \xrightarrow{-} \alpha$ can seem a bit redundant right now because the notation also includes the point at which the polynomial representation is written. We shall need this redundancy in notation when we are going to work with polynomials instead of polynomial representations.

In the sequel we shall need equality not of all the terms of two series but only equality of the initial part of the two series. Therefore we give the following definition:

Definition 2.10 (equality, projection) *We say that the two series $f_1(z)$ and $f_2(z)$ at the same point α are equal from γ_1 to γ_2 ($\gamma_1 \leq \gamma_2$) if the terms in $(z - \alpha)^k$, $\gamma_1 \leq k \leq \gamma_2$ of the two series are the same. If we define the projection operator $\Pi_{\gamma_1}^{\gamma_2}$ applied to a series $f(z) = \sum f_k(z - \alpha)^k$ as*

$$\Pi_{\gamma_1}^{\gamma_2} f(z) = \sum_{k=\gamma_1}^{\gamma_2} f_k(z - \alpha)^k,$$

then we can denote the equality from γ_1 to γ_2 as follows:

$$\Pi_{\gamma_1}^{\gamma_2} f_1(z) = \Pi_{\gamma_1}^{\gamma_2} f_2(z).$$

We shall also use the following short hand notation

$$f_1(z) = f_2(z), \quad (z \xrightarrow{+} \alpha)_{\gamma_1}^{\gamma_2} \text{ or } f_1(z) = f_2(z), \quad (z \xrightarrow{-} \alpha)_{\gamma_1}^{\gamma_2}.$$

For example,

$$\Pi_1^2 \frac{z-2}{z-1} = \Pi_1^2 (4 + z + z^2 + 3z^3 - 17z^{12}), \quad z \xrightarrow{+} 0$$

or

$$\frac{z-2}{z-1} = 4 + z + z^2 + 3z^3 - 17z^{12}, \quad (z \xrightarrow{+} 0)_1^2.$$

In connection with the interpolation problem to be studied, the initial terms of a **pfLs** or a **nfLs** will be important. We shall suppose that we do not know all of the coefficients of the **pfLs** or **nfLs** in play but only the initial ones. They are defined as follows:

Definition 2.11 (initial coefficients, initial part) *The initial coefficients for a pFLs are the coefficients with the smallest index. For a nFLs, the initial coefficients are those with the greatest index. The initial part of a pFLs or a nFLs of length $\tau \in \mathbb{N}$ is the sum of the τ terms containing the first τ coefficients.*

Instead of working directly with the initial part of the quotient of positive or negative fFLs, we shall work with the initial parts of the dividend and divisor series. The question which initial coefficients of the dividend and divisor series determine the initial coefficients of the quotient series, is answered in the following theorem.

Theorem 2.3 *Let $f(z)$ and $g(z)$ be two pFLs at α which can be written, as shown above, as shifted pfps $f'(z)$ and $g'(z)$*

$$f(z) = (z - \alpha)^\phi f'(z) \text{ and } g(z) = (z - \alpha)^\psi g'(z).$$

If $g \neq 0$, we can always choose ψ such that $\Pi_0^0 g'(z) \neq 0$. Then, the first τ coefficients of the quotient pfps $q'(z) = f'(z)/g'(z)$ are determined by the first τ coefficients of the dividend $f'(z)$ and the first $\tau - \eta$ coefficients of the divisor $g'(z)$ with η the number of initial zero coefficients of $g'(z)$ (in fact η is the strict order of $g'(z)$ at α which will be defined in definition 2.14). This holds also for $q(z) = f(z)/g(z) = (z - \alpha)^{\phi-\psi} q'(z)$. Thus

$$q'(z) = \frac{\Pi_0^{\tau-1} f'(z)}{\Pi_0^{\tau-\eta-1} g'(z)}, \quad (z \xrightarrow{+} \alpha)_0^{\tau-1}$$

and

$$q(z) = \frac{\Pi_\phi^{\phi+\tau-1} f(z)}{\Pi_\psi^{\psi+\tau-\eta-1} g(z)}, \quad (z \xrightarrow{+} \alpha)_{\phi-\psi}^{\phi-\psi+\tau-1}.$$

A similar theorem can be stated for nFLs.

Let us now discuss the formal transformation of pFLs at α into pFLs at another point $\alpha' \neq \alpha$. First of all, we consider the transformation of terms with positive powers. We can always write $(z - \alpha)^k$ with $k \geq 0$ as

$$\begin{aligned} (z - \alpha)^k &= ((z - \alpha') + (\alpha' - \alpha))^k \\ &= \sum_{i=0}^k \binom{k}{i} (\alpha' - \alpha)^{k-i} (z - \alpha')^i. \end{aligned} \quad (2.2)$$

We observe that this transformation will *not* allow us to transform a pfps into another one since each term of the transformed series will depend on all the following coefficients in the original one. Therefore this would make us run into convergence problems again. However if there are only finitely many nonzero coefficients to follow, like in polynomial representations, then the transformation is perfectly possible. In this connection we give the following definition.

Definition 2.12 (equivalence) *We say that two polynomial representations (at different points) are equivalent if they can be transformed into one another.*

This equivalence defines a genuine equivalence relation and the equivalence classes are the sets of polynomial representations that correspond to the same polynomial function. This polynomial function is defined as the one which maps $\mathbf{F}$ into $\mathbf{F}$ by substituting the indeterminate z in the polynomial representation by an element of the field $\mathbf{F}$ and using the $+$ and $\times$ operations of the field. We shall call these equivalence classes polynomials.

Definition 2.13 (polynomial) A polynomial is an equivalence class of polynomial representations at different points. The set of all these polynomials is denoted by $\mathbf{F}[z]$.

A polynomial can be represented by any polynomial representation at any possible point $\alpha \neq \infty$. In this notation, we include the indeterminate z indicating that we shall use the notation z for the indeterminate when using a specific polynomial representation for a polynomial. So, if we write

$$n(z) = 3 + 7z - 4z^4 \text{ with } n(z) \in \mathbf{F}[z],$$

the polynomial $n(z)$ is represented with the indeterminate z , at 0, by $3 + 7z - 4z^4$. The addition and multiplication of polynomials is defined based in a straightforward way on the corresponding operations with the polynomial representations.

The transformation of terms in a **pfls** with a negative exponent is not as simple as the transformation of those with a positive exponent. Consider e.g., the transformation of the **pfls** $(z - \alpha)^{-1}$ at α into a **pfls** at α' . This gives

$$\begin{aligned} (z - \alpha)^{-1} &= 1/((\alpha' - \alpha) + (z - \alpha')) \\ &= \frac{1}{(\alpha' - \alpha)} - \frac{1}{(\alpha' - \alpha)^2}(z - \alpha') + \frac{1}{(\alpha' - \alpha)^3}(z - \alpha')^2 - \dots \\ &= \sum_{i=0}^{+\infty} (-1)^i \left(\frac{1}{\alpha' - \alpha} \right)^{i+1} (z - \alpha')^i. \end{aligned}$$

As you can see from this, every coefficient in the transformed series will involve infinitely many coefficients of the original one. This makes us run into convergence problems again. Therefore, we do not allow such a transformation.

On the other hand, if we transform the **nfls** $(z - \beta)^{-1}$ at β into a **nfls** at β' , we get

$$\begin{aligned} (z - \beta)^{-1} &= ((z - \beta') + (\beta' - \beta))^{-1} \\ &= \sum_{i=1}^{+\infty} (-1)^{i+1} (\beta' - \beta)^{-i+1} (z - \beta')^{-i} \end{aligned}$$

In this case, we don't have a convergence problem for the coefficients of the transformed series because each term of the new series depends only on a finite number of terms of the original series. More precisely, the first τ coefficients of the new series only depend on the first τ coefficients of the original series. Therefore, if we know only the initial part of a **nfls** at a point β , by formal transformation we know the initial part of the corresponding **nfls** at another point $\beta' \neq \beta$. Such two **nfls** which can be transformed into one another are called equivalent by definition. All **nfls** containing consistent information are equivalent.

This observation is an important one with respect to our interpolation problem to be studied. If we are given the initial terms of a **pfls** at the point α , then we are actually given interpolation data at that point. As we have seen for a **pfls** it is not possible to transform e.g., the first τ terms at α into the first τ terms at α' and retain the same information. However, it is possible as we have seen, that if we are given interpolation data as the first τ terms of a **nfls** at β to transform this into the first τ terms of an equivalent **nfls** at some other point β' and still have the same information. We shall then say that we have interpolation data at ∞ . Thus it makes sense to have different interpolation data at different finite points (given as initial parts of **pfls** at these points) but there is only one set of interpolation data at ∞ (which can be given as the initial part of any **nfls** at some arbitrary point β since these will all be equivalent).

The same exceptional position of the point ∞ appears in the definition of strict orders which we are going to discuss now. Consider e.g., a polynomial. As we know, there are several polynomial representations possible. The strict order, which is the index of the first initial nonzero term when we consider the polynomial representation as a **pfls**, will depend on the point α at which the polynomial representation is taken. However its degree, which is in fact the index of the first nonzero initial term when we consider the polynomial representation as a **nfls** will always be the same. For symmetry reasons, we shall call what is usually indicated as the degree the strict order at infinity. So we have the following definitions which generalize this idea also to **pfls** and **nfls**:

Definition 2.14 (strict order) *The strict order of a **pfls** at α indicates the index of the first initial coefficient different from zero, i.e. it is the maximum of the orders of all possible shift representations of the **pfls***

$$\text{ord}_\alpha(f(z)) = \max \{ \sigma | f(z) = (z - \alpha)^\sigma f'(z), f'(z) \in \mathbf{F}_\alpha[[z]] \}.$$

*The strict order of the zero series is defined to be $\text{ord}_\alpha(0) = +\infty$. Similar definitions can be given for **nfls**. The strict order of a **nfls** at β indicates in the same way the index of the first initial coefficient different from zero, i.e. the strict order of $f(z) \in \mathbf{F}_\beta(z^{-1})$ is*

$$\text{ord}_\beta(f(z)) = \min \{ \sigma | f(z) = (z - \beta)^\sigma f'(z), f'(z) \in \mathbf{F}_\beta[[z^{-1}]] \}$$

*with $\text{ord}_\beta(0) = -\infty$. However, since ord_β does not depend on β for a **nfls**, we shall indicate it as ord_∞ and call it the strict order at infinity.*

*The strict order of a polynomial $n(z)$ at a finite point α is defined as the strict order of its polynomial representation at that point considering it as a **pfls**. The strict order at infinity or the degree of a polynomial $n(z)$ is defined as the strict order of its polynomial representation at any finite point β considered as a **nfls**.*

Note that the strict order of a **pfls** or of a **nfls** whose initial part is specified and equal to zero can assume an infinity of values depending on the possible extensions of the zero initial part.

Before we pass to the rational interpolation problem in the next section, we still have to introduce the field of rationals associated with the ring of polynomials. We start by defining a rational expression and a rational form as follows:

Definition 2.15 (rational expression, rational form) *A rational expression is an ordered couple $(n(z), d(z)) \neq (0, 0)$ of two polynomials $n(z)$ and $d(z)$ belonging to $\mathbf{F}[z]$. When there is no confusion with the divide operation, we shall also use the notation $n(z)/d(z)$ for the rational expression. We call $n(z)$ the numerator and $d(z)$ the denominator of the rational expression. The set of all rational expressions is denoted by **RatExp**. Two rational expressions $(n_1(z), d_1(z))$ and $(n_2(z), d_2(z))$ are equivalent if $n_1 d_2 = n_2 d_1$. An equivalence class under this equivalence relation is called a rational form and can be represented by any of its rational expressions. The set of all rational forms is denoted by **RatForms**.*

The **RIP** to be defined will look for a rational expression of smallest possible complexity. Therefore, we need a measure for the complexity of a rational expression. A classical tool for this is the MacMillan degree, that is the maximum of the numerator and denominator degrees. However we then treat both numerator and denominator at the same level. We want to introduce two shift parameters $(s_1, s_2) \in \mathbf{Z} \times \mathbf{Z}$ which will assign a relative importance to numerator and denominator. We work with two shift parameters instead of one to get

symmetric problem settings in the sequel. Because only the difference between the two shift parameters s_1 and s_2 is essential, we can assume without loss of generality that $s_1 \geq 0$ and $s_2 \geq 0$ where at least one of the two shift parameters is equal to zero. The complexity of a rational expression or a polynomial couple and of a rational form will be measured by the (s_1, s_2) -degree defined as follows:

Definition 2.16 ((s_1, s_2) -degree) *For the shift parameters $(s_1, s_2) \in \mathbf{Z} \times \mathbf{Z}$, with $s_1 \geq 0$ and $s_2 \geq 0$ and at least one of them equal to zero, we define the (s_1, s_2) -degree of the rational expression $(n(z), d(z))$ as $\max \{\deg n(z) + s_1, \deg d(z) + s_2\}$. We recall that the degree of the zero polynomial is $-\infty$. The (s_1, s_2) -degree of a rational form is defined as the minimal (s_1, s_2) -degree of all the rational expressions representing this rational form.*

We have the following result.

Theorem 2.4 *Knowing that $s_1 \geq 0$ and $s_2 \geq 0$ with at least one of them equal to zero, the (s_1, s_2) -degree is always greater than or equal to zero and there are rational expressions and rational forms with (s_1, s_2) -degree equal to zero.*

Proof. Using the definition above, the proof is trivial. $\square$

3 Formulation of the rational interpolation problem

To formulate the rational interpolation problem **RIP**, we shall introduce a format for the interpolation data based on shift representations. The choice of the representations is however not essential. It will become clear after we have developed the algorithm to solve the **RIP**, that the crucial values to be computed from the interpolation data, are the residuals, which are defined later on. These residuals are independent from the representations used. If we can compute the residuals, the algorithm can pursue its way up to the solution of the **RIP**. At this point we shall not give this general formulation. We use a specific notation to represent the interpolation data to formulate the **RIP** and to make the solution method clear. Abstraction of the specific representation format will be given later.

Another thing to remember is that we shall construct a recursive algorithm. This means that the interpolation data become available one at a time. The order in which the data become available is irrelevant with respect to the order of the interpolation points but the next time a data becomes available for some point, it should be the next initial term of a formal Laurent series at that point. At no time we shall have to solve a lacunary Birkhoff interpolation problem.

For each interpolation point α , which could be ∞ , we assume the interpolation data are given as the first τ_α coefficients of a quotient series. These coefficients are not necessarily directly known. They may be determined by the first τ_α coefficients of a dividend and divisor series. We shall make a distinction now between finite interpolation points and the possible interpolation point at infinity. For each finite interpolation point α , we assume we know the initial part of two series written in shift representation $(z - \alpha)^{\lambda_\alpha} L_\alpha(z)$ and $(z - \alpha)^{\mu_\alpha} M_\alpha(z)$ with L_α and M_α belonging to $\mathbf{F}_\alpha[[z]]$. Without loss of generality we can always assume that:

- the orders $\lambda_\alpha, \mu_\alpha \geq 0$,
- at least one of the orders is equal to zero,
- $l_{0,\alpha} = \Pi_0^0 L_\alpha$ or $m_{0,\alpha} = \Pi_0^0 M_\alpha$ is different from zero, i.e. at least one of the constant terms is different from zero. Thus at least one of the orders is a strict order.

We denote the initial parts of length τ_α as:

$$L_\alpha^{\tau_\alpha} = \Pi_0^{\tau_\alpha-1} L_\alpha$$

and

$$M_\alpha^{\tau_\alpha} = \Pi_0^{\tau_\alpha-1} M_\alpha.$$

If α is the interpolation point at infinity, we are working with **nfLs** at a fixed point β . We assume the initial parts of two series $(z - \beta)^{-\lambda_\infty} L_\infty(z)$ and $(z - \beta)^{-\mu_\infty} M_\infty(z)$ are known with L_∞ and M_∞ belonging to $\mathbf{F}_\beta[[z^{-1}]]$. Without loss of generality, we can assume the same conditions as above for the orders and the constant terms. The initial parts are denoted by $L_\infty^{\tau_\infty}$ and $M_\infty^{\tau_\infty}$. We get the following formulation of the proper **RIP**, “proper” in contrast to the linearized **RIP** of the next section:

Definition 3.1 (proper rational interpolation problem (PRIP)) *Let the interpolation data be given in the format as described above. Let the shift parameters (s_1, s_2) be given, indicating the relative importance of numerator and denominator degree or complexity. The proper rational interpolation problem (PRIP) asks for the rational expression or the polynomial couple $(n(z), d(z))$ of minimal (s_1, s_2) -degree satisfying the following interpolation conditions:*

- if the interpolation point α is finite and $m_{0,\alpha} \neq 0$:

$$\frac{n(z)}{d(z)} = \frac{(z - \alpha)^{\lambda_\alpha} L_\alpha(z)}{(z - \alpha)^{\mu_\alpha} M_\alpha(z)}, \quad (z \xrightarrow{+} \alpha)^{\lambda_\alpha - \mu_\alpha + \tau_\alpha - 1}, \quad (3.1)$$

- if the interpolation point α is finite and $l_{0,\alpha} \neq 0$:

$$\frac{d(z)}{n(z)} = \frac{(z - \alpha)^{\mu_\alpha} M_\alpha(z)}{(z - \alpha)^{\lambda_\alpha} L_\alpha(z)}, \quad (z \xrightarrow{+} \alpha)^{\mu_\alpha - \lambda_\alpha + \tau_\alpha - 1}, \quad (3.2)$$

- if the interpolation point α is infinity and $m_{0,\infty} \neq 0$:

$$\frac{n(z)}{d(z)} = \frac{(z - \beta)^{-\lambda_\infty} L_\infty(z)}{(z - \beta)^{-\mu_\infty} M_\infty(z)}, \quad (z \xrightarrow{-} \beta)^{+\infty}_{-\lambda_\infty + \mu_\infty - \tau_\infty + 1}, \quad (3.3)$$

- if the interpolation point α is infinity and $l_{0,\infty} \neq 0$:

$$\frac{d(z)}{n(z)} = \frac{(z - \beta)^{-\mu_\infty} M_\infty(z)}{(z - \beta)^{-\lambda_\infty} L_\infty(z)}, \quad (z \xrightarrow{-} \beta)^{+\infty}_{-\mu_\infty + \lambda_\infty - \tau_\infty + 1}. \quad (3.4)$$

At this point, we shall give some examples to show that all usual interpolation conditions can be translated within our interpolation data format. We shall take the field $\mathbf{F}$ equal to $\mathbf{C}$, the field of complex numbers. Note that we lift the “formal” level to the “function” level. This is done for the simplicity of formulation. Let us consider the following examples:

Example 3.1 [regular point] The finite interpolation point α is a regular point with function value f_α and first derivative f'_α . These interpolation conditions can be written using the proposed format as

$$\frac{(z - \alpha)^{\lambda_\alpha} L_\alpha^{\tau_\alpha}(z)}{(z - \alpha)^{\mu_\alpha} M_\alpha^{\tau_\alpha}(z)}, \quad (z \xrightarrow{+} \alpha)^{\lambda_\alpha - \mu_\alpha + \tau_\alpha - 1},$$

with $\lambda_\alpha = \mu_\alpha = 0$, $\tau_\alpha = 2$, $L_\alpha^{\tau_\alpha}(z) = f_\alpha + f'_\alpha(z - \alpha)$ and $M_\alpha^{\tau_\alpha}(z) = 1 + 0(z - \alpha)$.

Example 3.2 [pole] When we want to express the fact that the finite interpolation point α is a pole of order $\leq \rho$, we can write this in our interpolation data format as follows:

$$\frac{(z - \alpha)^{\lambda_\alpha} L_\alpha(z)}{(z - \alpha)^{\mu_\alpha} M_\alpha(z)}, \quad (z \xrightarrow{+} \alpha)_{-\infty}^{\lambda_\alpha - \mu_\alpha + \tau_\alpha - 1},$$

with $\lambda_\alpha = 0$, $\mu_\alpha = \rho$, $\tau_\alpha = 0$ and $m_{0,\alpha} \neq 0$ but undetermined.

Example 3.3 [strictly proper] The rational interpolant has to be strictly proper, i.e. the degree of the numerator is smaller than the degree of the denominator. This means the point at infinity is a zero of order at least one. Writing this down, using our interpolation data format, we obtain:

$$\frac{(z - \beta)^{-\lambda_\infty} L_\infty(z)}{(z - \beta)^{-\mu_\infty} M_\infty(z)}, \quad (z \xrightarrow{-} \beta)_{-\lambda_\infty + \mu_\infty - \tau_\infty + 1}^{+\infty},$$

with $\lambda_\infty = 1$, $\mu_\infty = 0$, $\tau_\infty = 0$ and $m_{0,\infty} \neq 0$ but undetermined.

4 Transformation of the problem

In this section we shall show the equivalence of the **PRIP** and the linearized **RIP**, defined hereafter. The proper problem is stated using the divide operation for **pfls** and **nfls**. In the linearized problem we shall only use multiplication and addition of the series involved.

Definition 4.1 (linearized rational interpolation problem (**LRIP**)) *Let the interpolation data be given in the format as described in the previous section. Let the shift parameters (s_1, s_2) be given, indicating the relative importance of numerator and denominator degree. The linearized rational interpolation problem (**LRIP**) asks for the rational expression or the polynomial couple $(n(z), d(z))$ of minimal (s_1, s_2) -degree satisfying the following linearized interpolation conditions:*

- if the interpolation point α is finite:

$$(z - \alpha)^{\mu_\alpha} M_\alpha(z) n(z) - (z - \alpha)^{\lambda_\alpha} L_\alpha(z) d(z) = 0, \quad (z \xrightarrow{+} \alpha)_0^{\tau_\alpha - 1 + \sigma_\alpha} \quad (4.1)$$

with

$$\sigma_\alpha = \min \{ \lambda_\alpha + \text{ord}_\alpha(d), \mu_\alpha + \text{ord}_\alpha(n) \}, \quad (4.2)$$

- if the interpolation point α is infinite:

$$(z - \beta)^{-\mu_\infty - \delta_\infty} M_\infty(z) n(z) - (z - \beta)^{-\lambda_\infty - \delta_\infty} L_\infty(z) d(z) = 0, \quad (z \xrightarrow{-} \beta)_0^{\tau_\infty - 1 - \delta_\infty + \sigma_\infty} \quad (4.3)$$

with

$$\delta_\infty = \delta - \min \{ \mu_\infty + s_1, \lambda_\infty + s_2 \}, \quad (4.4)$$

$$\delta \geq (s_1, s_2)\text{-degree of } (n(z), d(z)) \quad (4.5)$$

and

$$\sigma_\infty = \max \{ \text{ord}_\infty(d) - \lambda_\infty, \text{ord}_\infty(n) - \mu_\infty \}. \quad (4.6)$$

Note that we could have chosen $\delta = (s_1, s_2)$ -degree of $(n(z), d(z))$. However, we need the flexibility of choosing a number greater than this (s_1, s_2) -degree as will become clear in section 6.

Definition 4.2 The entity δ chosen in equation (4.5) is called the *potential degree* of the polynomial couple $(n(z), d(z))$.

Before we explain the advantages of the linearized above the proper problem setting, we prove the equivalence of the two problems:

Theorem 4.1 Given the same interpolation data and the same shift parameters, the **PRIP** and the **LRIP** are equivalent, even in the more general case when the minimality condition on the (s_1, s_2) -degree is dropped.

Proof. To prove the general case, we shall not assume minimality of the (s_1, s_2) -degree during the proof. First of all, we shall show that the proper interpolation conditions (3.1) and (3.2) at a finite point α are equivalent to the linearized interpolation conditions (4.1). Let us consider one of the two cases of the **PRIP** conditions at a finite point α . The other case can be treated similarly. Let us investigate the case where $m_{0,\alpha} \neq 0$. In this case the interpolation conditions are written as (3.1). Multiplying the left and the right hand side of the equation by $(z - \alpha)^{\mu_\alpha} M_\alpha(z)$ and $d(z)$ and bringing everything to the left hand side, leads to the following equivalent interpolation conditions:

$$(z - \alpha)^{\mu_\alpha} M_\alpha(z) n(z) - (z - \alpha)^{\lambda_\alpha} L_\alpha(z) d(z) = 0, \quad (z \xrightarrow{+} \alpha)_{-\infty}^{\lambda_\alpha + \tau_\alpha - 1 + \text{ord}_\alpha(d)}. \quad (4.7)$$

Note that $d \neq 0$, because $d = 0$ implies $\text{ord}_\alpha(d) = +\infty$ and $n \neq 0$. Because $M_\alpha \neq 0$, $(n(z), d(z))$ can not satisfy the linearized interpolation condition. First of all, let us consider the lower bound $-\infty$. Because the left hand side of the equation only consists of **pfps**, we know that the result after multiplication and addition will be a **pfps** too. Hence, instead of the lower bound $-\infty$, we can use 0. To obtain the equivalence between the linearized interpolation conditions 4.7 and 4.1, it remains to be shown that the upper bounds are the same or

$$\lambda_\alpha + \text{ord}_\alpha(d) = \sigma_\alpha \quad (4.8)$$

with σ_α defined by equation (4.2). Let us consider the two cases $L_\alpha^{\tau_\alpha} \neq 0$ and $L_\alpha^{\tau_\alpha} = 0$:

- If $L_\alpha^{\tau_\alpha} \neq 0$, we get $\text{ord}_\alpha(L_\alpha) < \tau_\alpha$. Hence, from equation (3.1) or equation (4.7) we easily derive the following relationship between $\text{ord}_\alpha(d)$ and $\text{ord}_\alpha(n)$:

$$\mu_\alpha + \text{ord}_\alpha(M_\alpha) + \text{ord}_\alpha(n) = \lambda_\alpha + \text{ord}_\alpha(L_\alpha) + \text{ord}_\alpha(d).$$

Because $\text{ord}_\alpha(M_\alpha) = 0$ and $\text{ord}_\alpha(L_\alpha) \geq 0$, we obtain that

$$\mu_\alpha + \text{ord}_\alpha(n) \geq \lambda_\alpha + \text{ord}_\alpha(d).$$

and this proves equation (4.8).

- If $L_\alpha^{\tau_\alpha} = 0$, we get $\text{ord}_\alpha(L_\alpha) \geq \tau_\alpha$. Hence, from equation (3.1) or equation (4.7) we easily derive the following relationship between $\text{ord}_\alpha(d)$ and $\text{ord}_\alpha(n)$:

$$\mu_\alpha + \text{ord}_\alpha(M_\alpha) + \text{ord}_\alpha(n) \geq \lambda_\alpha + \tau_\alpha + \text{ord}_\alpha(d).$$

Because $\text{ord}_\alpha(M_\alpha) = 0$ and $\tau_\alpha \geq 0$, we obtain that

$$\mu_\alpha + \text{ord}_\alpha(n) \geq \lambda_\alpha + \text{ord}_\alpha(d).$$

and this proves equation (4.8).

Because the linearized interpolation conditions at a finite point α (4.1) together with (4.2) are symmetric, they are equivalent to the proper interpolation conditions also in the case $l_{0,\alpha} \neq 0$. This completes the proof when the interpolation is at a finite point α .

Let us now prove the equivalence between the proper interpolation conditions and the linearized interpolation conditions when the interpolation point α is infinity. As for a finite point, we shall prove the equivalence between (3.3) and (4.3). Because the linearized interpolation conditions (4.3) together with (4.5), (4.4) and (4.6) are symmetric, we can conclude then that the equivalence is also true for the proper interpolation conditions (3.4). Multiplying (3.3) by $(z-\beta)^{-\mu_\infty-\delta_\infty} M_\infty$ and $d(z)$, gives us the following equivalent conditions:

$$(z-\beta)^{-\mu_\infty-\delta_\infty} M_\infty(z)n(z) - (z-\beta)^{-\lambda_\infty-\delta_\infty} L_\infty(z)d(z) = 0, \quad (z \rightarrow \beta)^{+\infty}_{-\tau_\infty+1-\delta_\infty-\lambda_\infty+\text{ord}_\infty(d)}. \quad (4.9)$$

Because $\text{ord}_\infty((z-\beta)^{-\mu_\infty} M_\infty(z)n(z)) = -\mu_\infty + \text{ord}_\infty(n) \leq -\mu_\infty + \delta - s_1$ and $\text{ord}_\infty((z-\beta)^{-\lambda_\infty} L_\infty(z)d(z)) \leq -\lambda_\infty + \text{ord}_\infty(d) \leq -\lambda_\infty + \delta - s_2$, we can change the upper bound $+\infty$ from equation (4.9) into 0. It remains to be shown that $\sigma_\infty = -\lambda_\infty + \text{ord}_\infty(d)$. This can be easily proven similar to the finite point case.

At this point, we have proven the general case. Introducing the minimality condition on the (s_1, s_2) -degree of $(n(z), d(z))$ shows that the **PRIP** and the **LRIP** are equivalent. $\square$

From $\lambda_\alpha \geq 0$, $\mu_\alpha \geq 0$, $\text{ord}_\alpha(d) \geq 0$ and $\text{ord}_\alpha(n) \geq 0$, we easily see that $\sigma_\alpha \geq 0$. Hence, the number of linearized interpolation conditions $\tau_\alpha + \sigma_\alpha$ at a finite point α is greater than the number τ_α of (explicitly or implicitly) given initial coefficients of the quotient series. If the interpolation point is infinity, there are τ_∞ initial coefficients given equivalent to $\tau_\infty + \delta_\infty - \sigma_\infty$ linearized interpolation conditions. From $-\mu_\infty + \text{ord}_\infty(n) \leq -\mu_\infty + \delta - s_1$ and $-\lambda_\infty + \text{ord}_\infty(d) \leq -\lambda_\infty + \delta - s_2$, it is easy to show that $\lambda_\infty - \sigma_\infty \geq 0$. Hence, also for the interpolation point at infinity, there are more linearized interpolation conditions than given initial coefficients of the quotient series.

Going from the proper to the linearized rational interpolation problem setting has a very big advantage. Take for example the linearized interpolation conditions (4.1) for a finite point α . Instead of taking $(z \xrightarrow{+} \alpha)_0^{\tau_\alpha-1+\sigma_\alpha}$, we consider equality of the left hand side series and the zero series for $(z \xrightarrow{+} \alpha)_0^{l_\alpha-1}$ with $l_\alpha \in \mathbb{N}$. Otherwise the number of linearized interpolation conditions $\tau_\alpha + \sigma_\alpha$ would depend on the polynomial couple $(n(z), d(z))$ we are looking for. By making the number l_α of linearized interpolation conditions independent of $(n(z), d(z))$, we have the important property that all solutions having a fixed maximum (s_1, s_2) -degree δ form a vector space. Once we know this vector space, we can go back and look for those solutions satisfying the linearized interpolation conditions with upper bound $\tau_\alpha - 1 + \sigma_\alpha$. Before we elaborate on this in section 6, we shall investigate the equivalence between the **LRIP** and a certain linear algebra problem involving Hankel and Toeplitz matrices.

5 The equivalent linear algebra problem

In the previous section, we have proven that the **PRIP** and the **LRIP** are equivalent. Hence, instead of solving the **PRIP** we can solve the **LRIP**. Therefore, we investigate the **LRIP** in more detail. We are going to transform the **LRIP** into a linear algebra problem by writing the linearized interpolation conditions in terms of the coefficients of the **fLs** and polynomial representations into play. The initial coefficients of these **fLs** are denoted as follows:

- $L_\alpha(z) = \sum_{k=1}^{\tau_\alpha} l_{\alpha,k}(z-\alpha)^{k-1}, \quad (z \xrightarrow{+} \alpha)_0^{\tau_\alpha-1},$
- $M_\alpha(z) = \sum_{k=1}^{\tau_\alpha} m_{\alpha,k}(z-\alpha)^{k-1}, \quad (z \xrightarrow{+} \alpha)_0^{\tau_\alpha-1},$

- $L_\infty(z) = \sum_{k=1}^{\tau_\infty} l_{\infty,k}(z - \alpha)^{-k+1}, \quad (z \rightarrow \beta)_{-\tau_\infty+1}^0,$
- $M_\infty(z) = \sum_{k=1}^{\tau_\infty} m_{\infty,k}(z - \alpha)^{-k+1}, \quad (z \rightarrow \beta)_{-\tau_\infty+1}^0.$

When we consider the polynomial couple $(n(z), d(z))$ we are looking for in the **PRIP** or **LRIP**, we can not directly speak about the coefficients of the polynomials $n(z)$ and $d(z)$. We can only talk about coefficients of a specific polynomial representation for the two polynomials $n(z)$ and $d(z)$. We introduce the following notation for the polynomial representation coefficients:

Definition 5.1 (polynomial coefficient vector) *The coefficients of a polynomial representation at a point $\alpha \in \mathbf{F}$ of a polynomial $p(z)$ having degree $\leq \pi$ are denoted as follows:*

$$p(z) = \sum_{k=0}^{\pi} p_{\alpha,k}(z - \alpha)^k.$$

The polynomial coefficient vector at α , denoted by $\mathbf{p}_\alpha$, is defined as the vector built up by the polynomial coefficients $p_{\alpha,k}$ as follows:

$$\mathbf{p}_\alpha = \begin{bmatrix} p_{\alpha,0} & p_{\alpha,1} & \cdots & p_{\alpha,\pi} \end{bmatrix}^T$$

where T denotes the transpose operation.

When we know the coefficients of a polynomial representation at α of a polynomial $p(z)$, we can easily derive the coefficients of the polynomial representation at α' of the same polynomial $p(z)$ as stated by the following theorem:

Theorem 5.1 *The coefficients $p_{\alpha,k}$ and $p_{\alpha',k}$ of the polynomial representations at α and α' respectively of the same polynomial*

$$\begin{aligned} p(z) &= \sum_{k=0}^{\pi} p_{\alpha,k}(z - \alpha)^k \\ &= \sum_{k=0}^{\pi} p_{\alpha',k}(z - \alpha')^k \end{aligned}$$

of degree $\leq \pi$ are related by the following formula:

$$p_{\alpha',k} = \sum_{i=k}^{\pi} p_{\alpha,i} \binom{i}{k} (\alpha' - \alpha)^{i-k}, \quad k = 0, \dots, \pi.$$

This relation can also be written as a linear transformation between the two polynomial coefficient vectors $\mathbf{p}_\alpha$ and $\mathbf{p}_{\alpha'}$ as follows:

$$\mathbf{p}_{\alpha'} = T_{(\pi+1) \times (\pi+1)}^{\alpha \rightarrow \alpha'} \mathbf{p}_\alpha$$

with the $(\pi+1) \times (\pi+1)$ transformation matrix $T_{(\pi+1) \times (\pi+1)}^{\alpha \rightarrow \alpha'}$ defined as the following upper right triangular matrix

$$T_{(\pi+1) \times (\pi+1)}^{\alpha \rightarrow \alpha'} = \begin{bmatrix} t_{0,0} & t_{0,1} & \cdots & t_{0,\pi} \\ 0 & t_{1,0} & \cdots & t_{1,\pi-1} \\ & \ddots & \ddots & \\ 0 & \cdots & 0 & t_{\pi,0} \end{bmatrix}$$

with

$$t_{i,j} = \binom{i+j}{i} (\alpha - \alpha')^j, \quad i, j \geq 0 \text{ and } i + j \leq \pi.$$

Moreover, the transformation matrix satisfies the following relationship:

$$\left(T_{(\pi+1) \times (\pi+1)}^{\alpha \rightarrow \alpha'} \right)^{-1} = T_{(\pi+1) \times (\pi+1)}^{\alpha' \rightarrow \alpha}.$$

Proof. The relationship between the polynomial coefficients follows from the formal transformation of one polynomial representation into another based on equation (2.2). $\square$

Knowing the coefficients of a fLs, we can construct the following matrices:

Definition 5.2 (Toeplitz matrix, Hankel matrix) Given the coefficients l_k of a fLs $l(z) = \sum_{k=-\infty}^{+\infty} l_k(z - \alpha)^k$, the $r \times c$ Toeplitz matrix based on the fLs $l(z)$ whose $(1,1)$ -element is l_σ is denoted by $T_{r \times c}^{(\sigma)}(l)$ and defined as

$$T_{r \times c}^{(\sigma)}(l) = \begin{bmatrix} l_\sigma & l_{\sigma-1} & l_{\sigma-2} & \cdots & l_{\sigma-c+1} \\ l_{\sigma+1} & l_\sigma & l_{\sigma-1} & \ddots & l_{\sigma-c+2} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ l_{\sigma+r-1} & l_{\sigma+r-2} & l_{\sigma+r-3} & \cdots & l_{\sigma+r-c} \end{bmatrix}.$$

The Toeplitz matrix $T_{r \times c}^{(\sigma)}(l)$ uses the coefficients $l_{\sigma-c+1}, l_{\sigma-c+2}, \dots, l_{\sigma+r-1}$. The $r \times c$ Hankel matrix based on the fLs $l(z)$ whose $(1,1)$ -element is l_σ is denoted by $H_{r \times c}^{(\sigma)}(l)$ and defined as

$$H_{r \times c}^{(\sigma)}(l) = \begin{bmatrix} l_\sigma & l_{\sigma+1} & l_{\sigma+2} & \cdots & l_{\sigma+c-1} \\ l_{\sigma+1} & l_{\sigma+2} & l_{\sigma+3} & \ddots & l_{\sigma+c} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ l_{\sigma+r-1} & l_{\sigma+r} & l_{\sigma+r+1} & \cdots & l_{\sigma+r+c-2} \end{bmatrix}.$$

The Hankel matrix $H_{r \times c}^{(\sigma)}(l)$ uses the coefficients $l_\sigma, l_{\sigma+1}, \dots, l_{\sigma+r+c-2}$.

Using the concepts introduced above we state the following theorem:

Theorem 5.2 If we denote the (s_1, s_2) -degree of the polynomial couple $(n(z), d(z))$ by δ , the PRIP and the LRIP are equivalent to the following linear algebra problem. Look for the polynomial couple $(n(z), d(z))$, having minimal (s_1, s_2) -degree, represented by their polynomial representations at a fixed point γ , satisfying the following set of homogeneous equations:

- if the interpolation point α is finite, $\begin{bmatrix} n_\gamma^T & d_\gamma^T \end{bmatrix}^T$ have to satisfy the following linear homogeneous equations

$$\begin{bmatrix} T_{(\tau_\alpha + \sigma_\alpha) \times (\delta - s_1 + 1)}^{-\mu_\alpha + 1}(M_\alpha) & -T_{(\tau_\alpha + \sigma_\alpha) \times (\delta - s_2 + 1)}^{-\lambda_\alpha + 1}(L_\alpha) \\ T_{(\delta - s_1 + 1) \times (\delta - s_1 + 1)}^{\gamma \rightarrow \alpha} & 0 \\ 0 & T_{(\delta - s_2 + 1) \times (\delta - s_2 + 1)}^{\gamma \rightarrow \alpha} \end{bmatrix} \begin{bmatrix} n_\gamma \\ d_\gamma \end{bmatrix} = 0, \quad (5.1)$$

- if the interpolation point α is infinite, $\begin{bmatrix} n_\gamma^T & d_\gamma^T \end{bmatrix}^T$ have to satisfy the following linear homogeneous equations

$$\begin{bmatrix} H_{(\tau_\infty + \delta_\infty - \sigma_\infty) \times (\delta - s_1 + 1)}^{-\mu_\infty - \delta_\infty + 1}(M_\infty) & -H_{(\tau_\infty + \delta_\infty - \sigma_\infty) \times (\delta - s_2 + 1)}^{-\lambda_\infty - \delta_\infty + 1}(L_\infty) \\ T_{(\delta - s_1 + 1) \times (\delta - s_1 + 1)}^{\gamma \rightarrow \beta} & 0 \\ 0 & T_{(\delta - s_2 + 1) \times (\delta - s_2 + 1)}^{\gamma \rightarrow \beta} \end{bmatrix} \begin{bmatrix} n_\gamma \\ d_\gamma \end{bmatrix} = 0. \quad (5.2)$$

Proof. If the interpolation point α is finite, the linearized interpolation conditions (4.1) written in coefficient notation, together with theorem 5.1, lead to the linear equations (5.1). Similarly, if the interpolation point α is infinity, the linearized interpolation conditions (4.3) are equivalent to the linear equations (5.2). $\square$

If, for the moment, we forget the problem that the number of linear equations $\tau_\alpha + \sigma_\alpha$ and $\tau_\alpha + \delta_\alpha - \sigma_\alpha$ are dependent on the rational expression $(n(z), d(z))$, that we are looking for, we can apply the following very crude and inefficient algorithm to find a solution for the linear algebra problem and for the equivalent **LRIP** and **PRIP**. Assume first that the solution has minimal (s_1, s_2) -degree 0. Try to find a nontrivial element of the solution space of the set of linear homogeneous equations. If only the polynomial couple $(0, 0)$ is a solution, try to look for a nontrivial element of (s_1, s_2) -degree 1, and so on. Of course, this algorithm is very inefficient because it doesn't make use of the fact that the coefficient matrices of the linear algebra problem are highly structured. This structure guarantees a very efficient computational scheme for the solution of the **LRIP** and **PRIP** as will be shown in the next section.

6 Solution of the LRIP

We could use the linear algebra notation of the previous section to explain our solution method for the **LRIP** problem. However, we prefer the more compact “series” notation of section 4. It is clear that all the results we obtain in the “series” notation can be easily translated to corresponding linear algebra results.

In this section, we are going to develop an elegant and efficient recursive algorithm to solve the **LRIP** and equivalently the **PRIP**. At each recursive step of the algorithm, we are going from one level to another. The term “level” indicates that at that specific moment we have chosen a certain number l_α of linearized interpolation conditions for the finite interpolation points α and a specific number l_∞ of interpolation conditions at infinity. When the algorithm progresses from one level to the next, one more linearized interpolation condition is added, i.e. either l_α or l_∞ is increased by one. The question for which interpolation point α or ∞ , we can add the following linearized interpolation condition, will be answered later on in theorem 6.9. More precisely, we give the following definition.

Definition 6.1 (level) “At level l ” indicates that we consider the first l_α linearized interpolation conditions (4.1) for each finite interpolation point α , i.e.

$$(z - \alpha)^{\mu_\alpha} M_\alpha(z) n(z) - (z - \alpha)^{\lambda_\alpha} L_\alpha(z) d(z) = 0, \quad (z \xrightarrow{+} \alpha)_0^{l_\alpha-1} \quad (6.1)$$

and the first l_∞ linearized interpolation conditions (4.3) for the interpolation point at infinity, i.e.

$$(z - \beta)^{-\mu_\infty - \delta_\infty} M_\infty(z) n(z) - (z - \beta)^{-\lambda_\infty - \delta_\infty} L_\infty(z) d(z) = 0, \quad (z \xrightarrow{-} \beta)_{-l_\infty+1}^0 \quad (6.2)$$

with $l = \sum_\alpha l_\alpha + l_\infty$.

Note that in equation (6.2) the value of δ_∞ is dependent on the potential degree chosen for the polynomial couple $(n(z), d(z))$. Hence, “at level $l + 1$ ” indicates that one of the values l_α or l_∞ is increased by one, i.e. one more linearized interpolation condition is considered. This is, what we call, the $(l + 1)$ -th linearized interpolation condition. The corresponding interpolation point α or ∞ is called the $(l + 1)$ -th interpolation point z_{l+1} . Knowing the specific choices for l_α and l_∞ at level l , we can define the following vector spaces.

Definition 6.2 We denote by $S_{l,\delta}$, with level $l \geq 0$ and potential degree $\delta \geq 0$, the vector space of all polynomial couples $(n(z), d(z))$ having (s_1, s_2) -degree $\leq \delta$ and satisfying (6.1) for each α and (6.2) together with (4.5) for the interpolation point at infinity.

Because we have introduced the notion of (s_1, s_2) -degree in such a way that the (s_1, s_2) -degree of a nontrivial polynomial couple is always greater than or equal to zero, i.e. $S_{l,\delta} = \{(0, 0)\}$ for $\delta < 0$, we can concentrate our attention on the vector spaces $S_{l,\delta}$ with $\delta \geq 0$. In the sequel we are going to work extensively with polynomial couples. Therefore, we introduce the following notations. The components of a polynomial couple $x(z) \in \mathbf{F}[z] \times \mathbf{F}[z]$ are denoted as $x(z) = (n_x(z), d_x(z))$. Multiplication and division of a polynomial couple $x(z)$ by a (nonzero) polynomial $p(z) \in \mathbf{F}[z]$ are defined and denoted as follows: $p(z)x(z) = (p(z)n_x(z), p(z)d_x(z))$ and $x(z)/p(z) = (n_x(z)/p(z), d_x(z)/p(z))$. $x(z)/p(z)$ is not necessarily a polynomial couple. If both $n_x(z)$ and $d_x(z)$ are divisible by $p(z)$, we say that $x(z)$ is divisible by $p(z)$ and in this case $x(z)/p(z)$ reduces to a polynomial couple.

Definition 6.3 (highest (potential) degree coefficient) The highest (potential) degree coefficient of a polynomial couple $x(z) = (n_x(z), d_x(z))$ of (potential) degree δ is defined as the couple $Hx \in \mathbf{F}^2$ where the first component is the coefficient of the term of $n_x(z)$ with degree $\delta - s_1$ and the second component is the coefficient of the term of $d_x(z)$ with degree $\delta - s_2$.

In the definition, we have not explicitly mentioned the point α at which the polynomial representation is taken because the highest (potential) degree coefficient is independent of this choice. Note that a polynomial couple with a highest potential degree coefficient $\neq (0, 0)$ has (s_1, s_2) -degree equal to the potential degree and the highest potential degree coefficient is equal to the highest degree coefficient.

Definition 6.4 (constant coefficient) The constant coefficient of a polynomial couple $x(z)$ with respect to a finite point $\alpha \in \mathbf{F}$ is defined as the couple $C_\alpha x \in \mathbf{F}^2$ equal to

$$C_\alpha x = \begin{bmatrix} n_x(\alpha) & d_x(\alpha) \end{bmatrix}.$$

Using these definitions, it is easy to prove the following very important connections between various “neighbouring” vector spaces $S_{l,\delta}$.

Property 6.1 For subsequent potential degrees δ and $\delta + 1$ at the same level l , we have the following connections:

$$S_{l,\delta} \subset S_{l,\delta+1},$$

$$(z - \beta)S_{l,\delta} \subset S_{l,\delta+1}, \text{ for any } \beta \in \mathbf{F}.$$

For subsequent levels l and $l + 1$, we have the following relationships:

$$S_{l+1,\delta} \subset S_{l,\delta}$$

and

- if the $(l + 1)$ -th interpolation point is infinity

$$S_{l,\delta} \subset S_{l+1,\delta+1},$$

$$\{x(z) \in S_{l+1,\delta+1} | Hx = 0\} = S_{l,\delta},$$

- if the $(l + 1)$ -th interpolation point is the finite point $\alpha \in \mathbf{F}$

$$(z - \alpha)S_{l,\delta} \subset S_{l+1,\delta+1},$$

$$\{x(z) \in S_{l+1,\delta+1} | C_\alpha x = 0\} = S_{l,\delta}.$$

To prove theorem 6.6 later on, we introduce the notion of residual.

Definition 6.5 (residual) For $l > 0$, the l -th residual of the polynomial couple $v(z) = (n_v(z), d_v(z))$ is denoted as $R_l v$ and defined as

$$(z - \alpha)^{l_\alpha - 1} R_l v = (z - \alpha)^{\mu_\alpha} M_\alpha(z) n_v(z) - (z - \alpha)^{\lambda_\alpha} L_\alpha(z) d_v(z), \quad (z \xrightarrow{+} \alpha)_{l_\alpha - 1}^{l_\alpha - 1}$$

if the l -th linearized interpolation condition is the l_α -th condition for the finite interpolation point α , and if the l -th linearized interpolation condition is the l_∞ -th condition for the interpolation point ∞ where we have chosen the potential degree equal to δ , then it is denoted as $R_l^\delta v$ and defined by

$$(z - \beta)^{-l_\infty + 1} R_l^\delta v = (z - \beta)^{-\mu_\infty - \delta_\infty} M_\infty(z) n_v(z) - (z - \beta)^{-\lambda_\infty - \delta_\infty} L_\infty(z) d_v(z), \quad (z \xrightarrow{-} \beta)_{-l_\infty + 1}^{-l_\infty + 1}.$$

We have the following important property.

Property 6.2 For any two polynomials $a(z), b(z) \in \mathbf{F}[z]$ and any two polynomial couples $v(z) \in S_{l-1,\delta_v}$, $w(z) \in S_{l-1,\delta_w}$, we have the following equality

$$R_l(av + bw) = a(z_l)R_l v + b(z_l)R_l w$$

if z_l is a finite interpolation point and the equality

$$R_l^\delta(av + bw) = a_h R_l^{\delta_v} v + b_h R_l^{\delta_w} w$$

if z_l is the interpolation point ∞ and if the potential degree δ of $av + bw$ is equal to

$$\delta = \deg a + \text{potdeg } v = \deg b + \text{potdeg } w$$

with a_h and b_h the highest degree coefficients of the polynomials $a(z)$ and $b(z)$ respectively.

From the definitions 6.2 and 6.5, the following corollary follows immediately.

Corollary 6.3 The polynomial couple $v(z)$ is an element of $S_{l,\delta}$ if and only if the residuals $R_i v$ or $R_i^\delta v$ for $i = 1, 2, \dots, l$ are equal to zero and $v(z)$ has (s_1, s_2) -degree smaller than or equal to the potential degree δ .

Before we show in theorem 6.6 that a basis for each solution space $S_{l,\delta}$ with $\delta \geq 0$ can be constructed using two polynomial couples $v_l(z) = (n_{l,v}(z), d_{l,v}(z))$ and $w_l(z) = (n_{l,w}(z), d_{l,w}(z))$, we investigate a special case in the following lemma.

Lemma 6.4 If there are two polynomial couples with linearly independent highest potential degree coefficients satisfying the first l_∞ linearized interpolation conditions in the interpolation point ∞ , then each polynomial couple satisfies the first l_∞ linearized interpolation conditions in ∞ or, in other words, the first l_∞ linearized interpolation conditions in ∞ are trivial. This situation occurs if and only if

$$l_\infty \leq \mu_\infty + s_1 - \min \{\mu_\infty + s_1, \lambda_\infty + s_2\} - \text{ord}_\infty(M_\infty) \quad \text{and} \quad (6.3)$$

$$\leq \lambda_\infty + s_2 - \min \{\mu_\infty + s_1, \lambda_\infty + s_2\} - \text{ord}_\infty(L_\infty). \quad (6.4)$$

Proof. To avoid the difficulty that one of the components of the highest potential degree coefficient can be equal to zero, it is easy to see using property 6.2 that we can replace the original two polynomial couples $a(z)$ and $b(z)$ with two other polynomial couples $a'(z)$ and $b'(z)$ by taking linear polynomial combinations of $a(z)$ and $b(z)$ such that $a'(z)$ and $b'(z)$ with potential degree $\delta_{a'}$ and $\delta_{b'}$ respectively still have linearly independent highest potential degree coefficients, still have no zero highest potential degree coefficient components and still satisfy the first l_∞ linearized interpolation conditions (6.2) at the interpolation point ∞ . First of all, we suppose that both M_∞ and L_∞ are different from zero. In this case, we can write the highest degree terms of M_∞ and L_∞ as follows:

$$M_\infty(z) = m_h(z - \beta)^{\text{ord}_\infty(M_\infty)}, \quad (z \rightarrow \beta)_{\text{ord}_\infty(M_\infty)}^{+\infty},$$

$$L_\infty(z) = l_h(z - \beta)^{\text{ord}_\infty(L_\infty)}, \quad (z \rightarrow \beta)_{\text{ord}_\infty(L_\infty)}^{+\infty}.$$

If we have a polynomial couple $v(z) = (n(z), d(z))$ of potential degree δ with highest potential degree components different from zero satisfying the first l_∞ linearized interpolation conditions (6.2) at ∞ , we can write the highest degree δ_1 of the first term in the left-hand side of (6.2) as follows:

$$\begin{aligned} \delta_1 &= -\mu_\infty - \delta_\infty + \text{ord}_\infty(M_\infty) + \text{ord}_\infty(n) \\ &= -\mu_\infty - \delta + \min\{\mu_\infty + s_1, \lambda_\infty + s_2\} + \text{ord}_\infty(M_\infty) + \delta - s_1 \\ &= -\mu_\infty + \min\{\mu_\infty + s_1, \lambda_\infty + s_2\} + \text{ord}_\infty(M_\infty) - s_1. \end{aligned}$$

In the same way, it can be shown that the highest degree δ_2 of the second term in the left-hand side of (6.2) can be written as follows:

$$\delta_2 = -\lambda_\infty + \min\{\mu_\infty + s_1, \lambda_\infty + s_2\} + \text{ord}_\infty(L_\infty) - s_2.$$

It is very important to note that these highest degrees are independent of the specific $v(z)$ we have chosen and independent of the potential degree. In fact, they are only determined by the initial problem data. If one of the two highest degrees δ_1, δ_2 is greater than $-l_\infty$, δ_1 has to be equal to δ_2 and both are greater than $-l_\infty$. The highest potential degree coefficients of $a'(z)$ and $b'(z)$ seen as (2×1) vectors, have to satisfy

$$\begin{bmatrix} m_h & l_h \end{bmatrix} \begin{bmatrix} Ha' & Hb' \end{bmatrix} = \begin{bmatrix} 0 & 0 \end{bmatrix}. \quad (6.5)$$

Because $\begin{bmatrix} m_h & l_h \end{bmatrix}$ has rank one and the highest potential degree coefficients Ha' and Hb' are linearly independent, the right-hand side of equation (6.5) should have rank one. So, the highest potential degrees Ha' and Hb' have to be smaller than or equal to $-l_\infty$. In this case, it is clear that any polynomial couple satisfies the first l_∞ linearized interpolation conditions at ∞ . From above, this happens if and only if l_∞ satisfies

$$l_\infty \leq \min\{-\delta_1, -\delta_2\}$$

and this proves the lemma when M_∞ and L_∞ are not zero.

The case where M_∞ or L_∞ is equal to zero, can be treated in the same way. Following the problem setting only one of the two can be equal to zero, not both at the same time. $\square$

In the same way, we can prove the following lemma for a finite interpolation point α .

Lemma 6.5 *If*

$$l_\alpha \leq \min \{ \mu_\alpha + \text{ord}_\alpha(M_\alpha), \lambda_\alpha + \text{ord}_\alpha(L_\alpha) \}, \quad (6.6)$$

then all polynomial couples satisfy the first l_α linearized interpolation conditions at α . In other words, the first l_α interpolation conditions at α are trivial. If l_α is greater than the minimum of equation (6.6), the first l_α linearized interpolation conditions at α are not trivial. Hence, in this case it is not possible that there are two polynomial couples $a(z)$ and $b(z)$ having linearly independent constant coefficients with respect to the finite point α and satisfying the first l_α linearized interpolation conditions at α .

Given a number $\delta \in \mathbb{N}$ and a polynomial couple $x(z)$ having potential degree δ_x , we define the set of polynomial couples $\{x(z)\}^\delta$ as follows:

$$\begin{aligned} \{x(z)\}^\delta &= \emptyset, \text{ if } \delta < \delta_x \\ \{x(z)\}^\delta &= \{x(z), zx(z), \dots, z^{\delta-\delta_x}x(z)\}, \text{ if } \delta \geq \delta_x. \end{aligned}$$

Using this notation, we shall prove now the following theorem:

Theorem 6.6 *For each level $l \geq 0$, there exist two polynomial couples $v_l(z)$ and $w_l(z)$ with polynomial degrees $\delta_{l,v}$ and $\delta_{l,w}$ respectively, such that for each $\delta, |\delta| < \infty$, a basis $BS_{l,\delta}$ for $S_{l,\delta}$, is given by*

$$BS_{l,\delta} = V_{l,\delta} \cup W_{l,\delta} \text{ with } V_{l,\delta} = \{v_l(z)\}^\delta \text{ and } W_{l,\delta} = \{w_l(z)\}^\delta.$$

Proof. We shall prove the theorem by induction. The theorem is certainly true for $l = 0$. Choose, e.g., $v_0(z) = (1, 0)$ and $w_0(z) = (0, 1)$ with $\delta_{0,v} = (s_1, s_2)$ -degree of $v_0(z)$ and $\delta_{0,w} = (s_1, s_2)$ -degree of $w_0(z)$. We assume that the theorem is true for a certain value of l . Without loss of generality, we can assume that $v_l(z)$ and $w_l(z)$ are ordered such that the potential degree $\delta_{l,v}$ of $v_l(z)$ is smaller than or equal to the potential degree $\delta_{l,w}$ of $w_l(z)$.

In the sequel we shall give an algorithm to compute a possible choice for $v_{l+1}(z)$ and $w_{l+1}(z)$ together with their potential degrees, given $v_l(z)$ and $w_l(z)$ with their potential degrees. First of all, we are looking for a polynomial couple $a_{l+1}(z) \neq (0, 0) \in S_{l+1,\delta}$, having minimal potential degree δ with the additional condition that

- $a_{l+1}(z)$ has (s_1, s_2) -degree equal to δ if the $(l+1)$ -th interpolation condition is a condition in the point ∞ ,
- $a_{l+1}(z)$ is not divisible by $z - z_{l+1}$ if the $(l+1)$ -th interpolation condition is a condition in a finite interpolation point.

This polynomial couple $a_{l+1}(z)$ together with another polynomial couple $b_{l+1}(z)$ (up to re-ordering), will give us $v_{l+1}(z)$ and $w_{l+1}(z)$. Because the proof in case the $(l+1)$ -th interpolation condition is in a finite interpolation point, is very similar to the proof when the new interpolation condition is at infinity, we shall only consider the case where the new condition is at infinity. First of all, we consider the special case that $v_l(z)$ and $w_l(z)$ satisfy the $(l+1)$ -th linearized interpolation condition, i.e. both residuals $R_{l+1}^{\delta_{l,v}}v_l$ and $R_{l+1}^{\delta_{l,w}}w_l$ are equal to zero. In this case we can take $v_{l+1}(z) = v_l(z)$ with potential degree $\delta_{l+1,v} = \delta_{l,v}$ and $w_{l+1}(z) = w_l(z)$ with potential degree $\delta_{l+1,w} = \delta_{l,w}$. This last situation can only occur for the first interpolation conditions around ∞ , more precisely for l_∞ satisfying equations (6.3) and (6.4) from lemma 6.4. $v_l(z)$ and $w_l(z)$ are only determined by linearized conditions at finite interpolation points. Hence, it can be shown that their highest potential degree coefficients are linearly independent. It is clear that in this special case each $S_{l+1,\delta}$ is equal

to the corresponding $S_{l,\delta}$. Because, by induction, the theorem is true at level l , it will also be true for level $l+1$ when this special situation happens. When l_∞ does not satisfy (6.3) or (6.4) anymore, we get the following situation.

Because $a_{l+1}(z)$ has to be in $S_{l+1,\delta} \subset S_{l,\delta}$ (property 6.1), we can write $a_{l+1}(z)$ as a linear polynomial combination of $v_l(z)$ and $w_l(z)$. Because $S_{l,\delta} = \{(0,0)\}$ for $\delta < \delta_{l,v}$, we first try $\delta = \delta_{l,v}$.

- If $v_l(z)$ satisfies the $(l+1)$ -th interpolation condition, i.e., $R_{l+1}^{\delta_{l,v}} v_l = 0$, then $v_l(z) \neq (0,0)$ is an element of $S_{l+1,\delta}$ with minimal potential degree δ . We can choose $a_{l+1}(z)$ with potential degree $\delta_{l,v}$ equal to $v_l(z)$, because $v_l(z)$ has (s_1, s_2) -degree equal to $\delta_{l,v}$. If $v_l(z)$ would have a smaller (s_1, s_2) -degree, $v_l(z)$ would also be a solution of $S_{l,\delta_{l,v}-1}$ but $S_{l,\delta_{l,v}-1} = \{(0,0)\}$. In this case, $R_{l+1}^{\delta_{l,w}} w_l \neq 0$ and we choose $b_{l+1}(z) = w_l(z)$ with potential degree $\delta_{l,w} + 1$.
- If $v_l(z)$ does not satisfy the $(l+1)$ -th linearized interpolation condition, i.e. $R_{l+1}^{\delta_{l,v}} v_l \neq 0$, then $a_{l+1}(z) = p(z)v_l(z) + q(z)w_l(z)$ with $q \neq 0$. Otherwise $a_{l+1}(z)$ could be written as $a_{l+1}(z) = p(z)v_l(z)$ with $p(z)$ a polynomial. Since $a_{l+1}(z)$ should satisfy the $(l+1)$ -th interpolation condition, it follows that the potential degree of $a_{l+1}(z)$ has to be chosen greater than $\delta_{l,v} + \deg p$. Hence, the (s_1, s_2) -degree of $a_{l+1}(z)$ would be smaller than its potential degree. Because the (s_1, s_2) -degree of $a_{l+1}(z)$ has to be equal to its potential degree, $a_{l+1}(z)$ should have a component along $w_l(z)$. The smallest potential degree δ for which $a_{l+1}(z)$ can have a component $w_l(z)$ is $\delta_{l,w}$. We shall prove now that a possible choice for $a_{p+1}(z)$ is $w_l(z) - hz^{\delta_{l,w}-\delta_{l,v}}v_l(z)$ with $h \in \mathbf{F}$ computed as follows:

$$h = R_{l+1}^{\delta_{l,w}} w_l / R_{l+1}^{\delta_{l,v}} v_l \neq \infty$$

Using property 6.2, it is clear that $R_{l+1}^{\delta} a = 0$. If the (s_1, s_2) -degree of $a_{l+1}(z)$ would be smaller than the potential degree $\delta = \delta_{l,w}$, then, using property 6.1, $a_{l+1}(z) \in S_{l,\delta-1}$. Therefore, $a_{l+1}(z)$ could be written as a linear polynomial combination of the polynomial couple $v_l(z)$, without using $w_l(z)$ as a component. This leads to a contradiction because $a_l(z)$ has a component $w_l(z)$. Hence, $a_{l+1}(z)$ has (s_1, s_2) -degree equal to the potential degree $\delta_{l,w}$. In this second case, $b_{l+1}(z)$ is chosen as $b_{p+1}(z) = v_l(z)$ with potential degree $\delta_{l,v} + 1$.

$a_{l+1}(z)$ and $b_{l+1}(z)$ are a possible choice for $v_{l+1}(z)$ and $w_{l+1}(z)$ if we can prove that for each δ a basis for $S_{l+1,\delta}$ is given by $BS_{l+1,\delta} = A_{l+1,\delta} \cup B_{l+1,\delta} = \{a_{l+1}(z)\}^\delta \cup \{b_{l+1}(z)\}^\delta$. Take $a(z) \in S_{l+1,\delta}$, then $a(z)$ can be written as $a(z) = cz^{\delta-\delta_{l+1,a}}a_{l+1}(z) + a'(z)$ with the (s_1, s_2) -degree of $a'(z)$ smaller than its potential degree δ , i.e. with highest potential degree coefficient equal to zero. If $a(z)$ has potential degree smaller than the potential degree $\delta_{l+1,a}$ of $a_{l+1}(z)$, then the (s_1, s_2) -degree of $a(z)$ is smaller than its potential degree δ . Hence, $c = 0$. Otherwise, $a_{l+1}(z)$ would not have minimal potential degree. When the potential degree δ of $a(z)$ is greater than or equal to the potential degree $\delta_{l+1,a}$ of $a_{l+1}(z)$, we can always choose c such that $a(z) - cz^{\delta-\delta_{l+1,a}}a_{l+1}(z)$ has (s_1, s_2) -degree smaller than its potential degree δ . Indeed, lemma 6.4 indicates that when nontrivial interpolation conditions at ∞ are considered, it is not possible that there are two polynomial couples with linearly independent highest potential degree coefficients. Hence, Ha has to be a multiple of Ha_{l+1} . This determines the value of c in a unique way. Therefore, $a(z)$ can be written in a unique way as $a(z) = cz^{\delta-\delta_{l+1,a}}a_{l+1}(z) + a'(z)$ with the highest potential degree coefficient of $a'(z)$ equal to zero, i.e. $a'(z) \in S_{l,\delta-1}$ from property 6.1. Using the basis for $S_{l,\delta-1}$ based on $v_l(z)$ and $w_l(z)$, it is easy to see that $A_{l+1,\delta} \cup B_{l+1,\delta} = V_{l+1,\delta} \cup W_{l+1,\delta}$ forms a possible basis for $S_{l+1,\delta}$. $\square$

When proving the previous theorem, we gave a very efficient and elegant method to compute the polynomial couples $v_l(z)$ and $w_l(z)$ of each level $l \geq 0$. Let us summarize this method in the algorithm 1 below. From this algorithm or the previous theorem, it is easy to see the following property.

Algorithm 1: Computing $v_l(z)$ and $w_l(z)$

```

{Initialization}
  if  $s_1 \leq s_2$ 
    
$$\begin{bmatrix} v_0(z) & w_0(z) \\ \delta_{0,v} & \delta_{0,w} \end{bmatrix} = \begin{bmatrix} n_{0,v}(z) & n_{0,w}(z) \\ d_{0,v}(z) & d_{0,w}(z) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ s_1 & s_2 \end{bmatrix}$$

  else
    
$$\begin{bmatrix} v_0(z) & w_0(z) \\ \delta_{0,v} & \delta_{0,w} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ s_2 & s_1 \end{bmatrix}$$

  endif

  for  $l = 0, 1, 2, \dots$ 
    if  $z_{l+1} = \infty$ 
      Handling the interpolation point  $\infty$  (algorithm 2)
    else  $\{z_{l+1} \text{ is a finite interpolation point}\}$ 
      Handling a finite interpolation point  $\alpha$  (algorithm 3)
    endif
  endfor

```

Property 6.7 *The rank of the matrix formed by the constant coefficients $C_\alpha v_l$ and $C_\alpha w_l$ of the polynomial couples $v_l(z)$ and $w_l(z)$ defined by the previous theorem*

$$\begin{bmatrix} C_\alpha v_l & C_\alpha w_l \end{bmatrix} = \begin{bmatrix} n_{l,v}(\alpha) & n_{l,w}(\alpha) \\ d_{l,v}(\alpha) & d_{l,w}(\alpha) \end{bmatrix}$$

has full rank if α is not considered as an interpolation point so far or if α has been considered as an interpolation point but only with trivial linearized interpolation conditions. In the other case, i.e. when α has been taken as a finite interpolation point with nontrivial linearized interpolation conditions, this matrix has rank one. The same can be said about the highest potential degree coefficients Hv_l and Hw_l in connection with the interpolation point ∞ .

Proof. The property follows from the structure of the matrices $V_i(z), i = 1, 2, \dots, l$ of algorithms 2 and 3. $\square$

Until now, we implicitly assumed that not only the initial coefficients of $L_\alpha, M_\alpha, L_\infty$ and M_∞ are known but all the following coefficients as needed by the algorithm when computing certain residuals. We should make a connection between the upper bound $l_\alpha - 1$ of equation (6.1) for a finite interpolation point α and the upper bound $\tau_\alpha - 1 + \sigma_\alpha$ of (4.1) and between the lower bound $-l_\infty + 1$ of equation (6.2) and $-\tau_\infty + 1 - \delta_\infty + \sigma_\infty$ of (4.3). First of all, let

Algorithm 2: Handling the interpolation point ∞

```

if  $R_{l+1}^{\delta_{l,v}} v_l = 0$ 
  if  $R_{l+1}^{\delta_{l,w}} w_l = 0$ 
     $V_{l+1}(z) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ 
     $\begin{bmatrix} \delta_{l+1,v} & \delta_{l+1,w} \end{bmatrix} = \begin{bmatrix} \delta_{l,v} & \delta_{l,w} \end{bmatrix}$ 
  else
     $V_{l+1}(z) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ 
     $\begin{bmatrix} \delta_{l+1,v} & \delta_{l+1,w} \end{bmatrix} = \begin{bmatrix} \delta_{l,v} & \delta_{l,w} + 1 \end{bmatrix}$ 
  endif

else
   $V'_{l+1}(z) = \begin{bmatrix} -hz^{\delta_{l,w}-\delta_{l,v}} & 1 \\ 1 & 0 \end{bmatrix}$ 
  with  $h = R_{l+1}^{\delta_{l,w}} w_l / R_{l+1}^{\delta_{l,v}} v_l$ 
   $\begin{bmatrix} \delta_{l+1,a} & \delta_{l+1,b} \end{bmatrix} = \begin{bmatrix} \delta_{l,w} & \delta_{l,v} + 1 \end{bmatrix}$ 
  if  $\delta_{l+1,a} \leq \delta_{l+1,b}$ 
     $V_{l+1}(z) = V'_{l+1}(z)$ 
     $\begin{bmatrix} \delta_{l+1,v} & \delta_{l+1,w} \end{bmatrix} = \begin{bmatrix} \delta_{l+1,a} & \delta_{l+1,b} \end{bmatrix}$ 
  else
     $V_{l+1}(z) = V'_{l+1}(z) \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ 
     $\begin{bmatrix} \delta_{l+1,v} & \delta_{l+1,w} \end{bmatrix} = \begin{bmatrix} \delta_{l+1,a} & \delta_{l+1,b} \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ 
  endif
endif

```

Algorithm 3: Handling a finite interpolation point α

```

if  $R_{l+1}v_l = 0$ 
  if  $R_{l+1}w_l = 0$ 
    
$$V_{l+1}(z) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

    
$$\begin{bmatrix} \delta_{l+1,v} & \delta_{l+1,w} \end{bmatrix} = \begin{bmatrix} \delta_{l,v} & \delta_{l,w} \end{bmatrix}$$

  else
    
$$V_{l+1}(z) = \begin{bmatrix} 1 & 0 \\ 0 & z - z_{l+1} \end{bmatrix}$$

    
$$\begin{bmatrix} \delta_{l+1,v} & \delta_{l+1,w} \end{bmatrix} = \begin{bmatrix} \delta_{l,v} & \delta_{l,w} + 1 \end{bmatrix}$$

  endif

else
  
$$V'_{l+1}(z) = \begin{bmatrix} -h & z - z_{l+1} \\ 1 & 0 \end{bmatrix}$$

  with  $h = R_{l+1}w_l / R_{l+1}v_l$ 
  
$$\begin{bmatrix} \delta_{l+1,a} & \delta_{l+1,b} \end{bmatrix} = \begin{bmatrix} \delta_{l,w} & \delta_{l,v} + 1 \end{bmatrix}$$

  if  $\delta_{l+1,a} \leq \delta_{l+1,b}$ 
    
$$V_{l+1}(z) = V'_{l+1}(z)$$

    
$$\begin{bmatrix} \delta_{l+1,v} & \delta_{l+1,w} \end{bmatrix} = \begin{bmatrix} \delta_{l+1,a} & \delta_{l+1,b} \end{bmatrix}$$

  else
    
$$V_{l+1}(z) = V'_{l+1}(z) \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

    
$$\begin{bmatrix} \delta_{l+1,v} & \delta_{l+1,w} \end{bmatrix} = \begin{bmatrix} \delta_{l+1,a} & \delta_{l+1,b} \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

  endif

endif

```

us investigate which coefficients of M_α and L_α are used in the set of linearized interpolation conditions (4.1) at a finite interpolation point α and, similarly, which coefficients of M_∞ and L_∞ are used in the set of linearized interpolation conditions (4.3) when the interpolation point is equal to infinity. Equivalently, this theorem indicates which coefficients in the Hankel and Toeplitz matrices of theorem 5.2 are needed.

Theorem 6.8 *The number of initial coefficients of M_α and L_α needed in (4.1) is $\leq \tau_\alpha$. Similarly, the number of initial coefficients of M_∞ and L_∞ needed in (4.3) is $\leq \tau_\infty$.*

Proof. It is easy to see that the first $\tau_\alpha + \sigma_\alpha - (\mu_\alpha + \text{ord}_\alpha(n))$ coefficients of M_α are needed if $n \neq 0$ in equation (4.1). Similarly, the first $\tau_\alpha + \sigma_\alpha - (\lambda_\alpha + \text{ord}_\alpha(d))$ coefficients of L_α are needed. From (4.2), it follows that these two numbers are smaller than or equal to τ_α . In the same way, we can show that in equation (4.3), the first $\tau_\infty - \sigma_\infty + (\text{ord}_\infty(n) - \mu_\infty)$ coefficients of M_∞ are needed if $n \neq 0$ and the first $\tau_\infty - \sigma_\infty + (\text{ord}_\infty(d) - \lambda_\infty)$ coefficients of L_∞ are needed if $d \neq 0$. From (4.6), it follows that these two numbers are $< \tau_\infty$. $\square$

In fact, because the **PRIP** and the **LRIP** are equivalent and because the **PRIP** only knows the first τ_α initial coefficients of M_α and L_α and the first τ_∞ initial coefficients of M_∞ and L_∞ , the result of the previous theorem is intuitively clear. Let us now consider the problem of choosing the final values for l_α and l_∞ . We know these values if we can write σ_α and $\delta_\infty - \sigma_\infty$ not in function of the polynomial couple $(n(z), d(z))$ we are looking for, but in function of the given interpolation data. We have the following result.

Theorem 6.9 *The final solution of the **LRIP** or the corresponding **PRIP** is not divisible by $z - z_\alpha$ and satisfies the linearized interpolation conditions (6.1) with*

$$l_\alpha = \tau_\alpha + \max\{\lambda_\alpha - \text{ord}_\alpha(M_\alpha), \mu_\alpha - \text{ord}_\alpha(L_\alpha)\}. \quad (6.7)$$

If the initial coefficients of L_α or M_α are all zero, we can choose $\text{ord}_\alpha(L_\alpha)$ or $\text{ord}_\alpha(M_\alpha)$ respectively, such that l_α has the smallest value possible, i.e. $l_\alpha = \tau_\alpha + \lambda_\alpha$ or $l_\alpha = \tau_\alpha + \mu_\alpha$ respectively. This final solution has highest potential degree coefficient different from the zero vector and satisfies the linearized interpolation conditions (6.2) with

$$l_\infty = \tau_\infty + \max\{s_1 - \lambda_\infty + \text{ord}_\infty(L_\infty), s_2 - \mu_\infty + \text{ord}_\infty(M_\infty)\} - \min\{s_1 - \lambda_\infty, s_2 - \mu_\infty\}. \quad (6.8)$$

If the initial coefficients of L_∞ or M_∞ are all zero, we can choose $\text{ord}_\infty(L_\infty)$ or $\text{ord}_\infty(M_\infty)$ respectively, such that l_∞ has the smallest value possible.

Proof. Considering the proper, (3.1) and (3.2), or linearized interpolation conditions (4.1) for a finite interpolation point α , we can distinguish the following cases.

- If $\text{ord}_\alpha(L_\alpha) < \tau_\alpha$ and $\text{ord}_\alpha(M_\alpha) < \tau_\alpha$ then we have the following relationship between $\text{ord}_\alpha(n)$ and $\text{ord}_\alpha(d)$:

$$\mu_\alpha + \text{ord}_\alpha(M_\alpha) + \text{ord}_\alpha(n) = \lambda_\alpha + \text{ord}_\alpha(L_\alpha) + \text{ord}_\alpha(d).$$

Let us assume the case where $\text{ord}_\alpha(M_\alpha) = 0$. In this case $\sigma_\alpha = \lambda_\alpha + \text{ord}_\alpha(d)$. The other case $\text{ord}_\alpha(L_\alpha) = 0$ can be proven similarly. Because the final solution has to have minimal (s_1, s_2) -degree, it is not divisible by $z - z_\alpha$. Hence, either $\text{ord}_\alpha(n)$, either $\text{ord}_\alpha(d)$ has to be equal to zero. We have the following two cases.

- If $\lambda_\alpha \geq \mu_\alpha - \text{ord}_\alpha(L_\alpha)$, we get that $\text{ord}_\alpha(d) = 0$. Therefore, $\sigma_\alpha = \lambda_\alpha$ or $l_\alpha = \tau_\alpha + \sigma_\alpha = \tau_\alpha + \lambda_\alpha$. Hence, in this case, the theorem is true.

- If $\lambda_\alpha \leq \mu_\alpha - \text{ord}_\alpha(L_\alpha)$, we get that $\text{ord}_\alpha(n) = 0$. Therefore, $\sigma_\alpha = \mu_\alpha - \text{ord}_\alpha(L_\alpha)$. Hence, also in this case the theorem has been proven.

- If $\text{ord}_\alpha(L_\alpha) \geq \tau_\alpha$ and $\text{ord}_\alpha(M_\alpha) = 0$, we can write down the following relationship between $\text{ord}_\alpha(n)$ and $\text{ord}_\alpha(d)$:

$$\mu_\alpha + \text{ord}_\alpha(M_\alpha) + \text{ord}_\alpha(n) = \tau_\alpha + \lambda_\alpha + \text{ord}_\alpha(d) + p$$

with $p \in \mathbb{N}$. We want to choose this parameter p such that the number of additional linearized conditions σ_α is as small as possible. We can consider the following two cases.

- If $\mu_\alpha \geq \tau_\alpha + \lambda_\alpha + p$, we get that $\text{ord}_\alpha(n) = 0$. Therefore, $\sigma_\alpha = \lambda_\alpha + \text{ord}_\alpha = \mu_\alpha - \tau_\alpha - p$. We can make σ_α as small as possible by setting p equal to $p = \mu_\alpha - \tau_\alpha - \lambda_\alpha$. Hence, $\sigma_\alpha = \lambda_\alpha$.
- If $\mu_\alpha \leq \tau_\alpha + \lambda_\alpha + p$, we have that $\text{ord}_\alpha(d) = 0$. Hence, $\sigma_\alpha = \lambda_\alpha$.

This proves the theorem also in this case.

- The case $\text{ord}_\alpha(M_\alpha) \geq \tau_\alpha$ and $\text{ord}_\alpha(L_\alpha) = 0$ can be proven as in the previous case leading to $\sigma_\alpha = \mu_\alpha$.

For the interpolation point equal to infinity, we have in general that $l_\infty = \tau_\infty + \delta_\infty - \sigma_\infty$. Without loss of generality, we can look for the polynomial couple $(n(z), d(z))$ having (s_1, s_2) -degree equal to the potential degree δ of the linearized equations (4.3). In this case, we get that

$$\delta_\infty - \sigma_\infty = \delta - \min \{ \mu_\infty + s_1, \lambda_\infty + s_2 \} - \max \{ \text{ord}_\infty(d) - \lambda_\infty, \text{ord}_\infty(n) - \mu_\infty \} \quad (6.9)$$

with $\delta = \max \{ \text{ord}_\infty(n) + s_1, \text{ord}_\infty(d) + s_2 \}$. As for a finite interpolation point, we can distinguish between the following possibilities.

- If $\text{ord}_\infty(L_\infty) > -\tau_\infty$ and $\text{ord}_\infty(M_\infty) > -\tau_\infty$, we get the following relationship between $\text{ord}_\infty(n)$ and $\text{ord}_\infty(d)$:

$$-\mu_\infty + \text{ord}_\infty(M_\infty) + \text{ord}_\infty(n) = -\lambda_\infty + \text{ord}_\infty(L_\infty) + \text{ord}_\infty(d). \quad (6.10)$$

We consider the following two cases.

- If $\text{ord}_\infty(M_\infty) = 0$, we know that $\sigma_\infty = \text{ord}_\infty(d) - \lambda_\infty$. Hence (6.9) simplifies to

$$\delta_\infty - \sigma_\infty = \max \{ \text{ord}_\infty(n) - \text{ord}_\infty(d), s_2 - s_1 \} - \min \{ \mu_\infty - \lambda_\infty, s_2 - s_1 \}.$$

Using (6.10) and making the formula symmetric, we get

$$\delta_\infty - \sigma_\infty = \max \{ s_1 - \lambda_\infty + \text{ord}_\infty(L_\infty), s_2 - \mu_\infty + \text{ord}_\infty(M_\infty) \} - \min \{ s_1 - \lambda_\infty, s_2 - \mu_\infty \}.$$

This proves the theorem in this case.

- When $\text{ord}_\infty(L_\infty) = 0$, we can prove the theorem in a symmetric way.

- If $\text{ord}_\infty(L_\infty) \leq -\tau_\infty$ and $\text{ord}_\infty(M_\infty) = 0$, we have that $\sigma_\infty = \text{ord}_\infty(d) - \lambda_\infty$ and we can easily obtain the following relationship between the degrees $\text{ord}_\infty(n)$ and $\text{ord}_\infty(d)$

$$\text{ord}_\infty(n) - \text{ord}_\infty(d) = -\lambda_\infty - \tau_\infty + \mu_\infty - p \quad (6.11)$$

with $p \in \mathbb{N}$. We know that

$$\delta_\infty - \sigma_\infty = \max \{ \text{ord}_\infty(n) - \text{ord}_\infty(d), s_2 - s_1 \} - \min \{ \mu_\infty - \lambda_\infty, s_2 - s_1 \}. \quad (6.12)$$

Using (6.11) in (6.12) and choosing p such that we obtain the minimal value for $\delta_\infty - \sigma_\infty$, we derive the following result

$$\delta_\infty - \sigma_\infty = s_2 - \mu_\infty - \min \{ s_1 - \lambda_\infty, s_2 - \mu_\infty \}.$$

Hence, we have proven the theorem in this case.

- The remaining case, i.e. $\text{ord}_\infty(M_\infty) \leq -\tau_\infty$ and $\text{ord}_\infty(L_\infty) = 0$, can be handled in the same way as the previous case.

□

Because we know that $l_\alpha \geq \tau_\alpha$ and $l_\infty \geq \tau_\infty$, we sometimes need other coefficients than the given initial ones to compute the residuals of our algorithm but theorem 6.8 says that the final solution will not depend on these coefficients, whose value we do not know. Hence, we could choose the value zero for these coefficients or whatever value we like. This will become clear from the example we shall give in the next section. In the remainder of this section, we shall not only give a solution of the **LRIP** and the equivalent **PRIP** but also a parametrization of all solutions using a minimal number of free parameters. Before we describe these results, we need the following notation and definitions.

Definition 6.6 *The minimal potential degree $\delta_{l,\alpha}$ for a finite interpolation point α at level l is defined as follows*

$$\delta_{l,\alpha} = \min \{ \delta | v(z) \in S_{l,\delta} \text{ and } v(z) \text{ is not divisible by } z - \alpha \}.$$

The minimal potential degree $\delta_{l,\infty}$ for the interpolation point at infinity at level l is defined as follows

$$\delta_{l,\infty} = \min \{ \delta | v(z) \in S_{l,\delta} \text{ and } v(z) \text{ has } (s_1, s_2) - \text{degree equal to } \delta \}.$$

We can determine these minimal potential degrees in an efficient way using the following theorem.

Theorem 6.10 *Let $v_l(z)$ and $w_l(z)$ be defined by theorem 6.6. If $v_l(z)$ is not divisible by $z - \alpha$, the minimal potential degree $\delta_{l,\alpha} = \delta_{l,v}$. Otherwise, $\delta_{l,\alpha} = \delta_{l,w}$. If the (s_1, s_2) -degree of $v_l(z)$ is equal to the potential degree, $\delta_{l,\infty} = \delta_{l,v}$. In the other case, $\delta_{l,\infty} = \delta_{l,w}$.*

Proof. From theorem 6.6, we know that all polynomial couples $v(z) \in S_{l,\delta}$ can be written as a linear polynomial combination of the two polynomial couples $v_l(z)$ and $w_l(z)$. Hence, we can concentrate our attention on these two polynomial couples. If l_α satisfies equation (6.6) of lemma 6.5, we only consider trivial linearized interpolation conditions at the finite interpolation point α . In this case, we are certain that the matrix formed by the constant coefficients of the polynomial couples $v_l(z)$ and $w_l(z)$ with respect to the final interpolation

point α has full rank two. In this case $\delta_{l,\alpha}$ is equal to $\delta_{l,v}$. When l_α does not satisfy equation (6.6), we know that the matrix formed by the constant coefficients has rank one. Therefore, if $C_\alpha v_l \neq 0$, $\delta_{l,\alpha} = \delta_{l,v}$. If $C_\alpha v_l = 0$, we are certain that $C_\alpha w_l \neq 0$. Hence, $\delta_{l,\alpha} = \delta_{l,w}$. The same way of reasoning can be followed for the interpolation point equal to infinity. $\square$

Using the minimal potential degrees $\delta_{l,\alpha}$ and $\delta_{l,\infty}$, we can write down all solutions of the **LRIP** and the equivalent **PRIP** using the following parametrization.

Theorem 6.11 *Knowing the final values for the number of linearized interpolation conditions l_α and l_∞ , given by theorem 6.9, we can use algorithm 1 to compute the polynomial couples $v_l(z)$ and $w_l(z)$ at level $l = l_\infty + \sum_\alpha l_\alpha$ with potential degrees δ_v and δ_w respectively. The minimal (s_1, s_2) -degree δ of a solution of the **LRIP** and equivalent **PRIP** is equal to*

$$\delta = \min_\alpha \{\delta_{l,\alpha}, \delta_{l,\infty}\}.$$

A parametrization of all solutions is given by

$$v_{a,b}(z) = a(z)v_l(z) + b(z)w_l(z)$$

with the following restrictions on the two polynomial parameters $a(z)$ and $b(z)$:

- $\deg a(z) \leq \delta - \delta_{l,v}$,
- $\deg b(z) \leq \delta - \delta_{l,w}$,
- the (s_1, s_2) -degree of $v_{a,b}(z)$ is equal to δ , i.e. $a_h H v_l + b_h H w_l \neq 0$ with a_h the coefficient of the term $z^{\delta - \delta_{l,v}}$ of polynomial $a(z)$ and b_h the coefficient of the term $z^{\delta - \delta_{l,w}}$ of polynomial $b(z)$,
- $v_{a,b}(z)$ is not divisible by $z - \alpha$ for each finite interpolation point α we take into consideration, i.e. $a(\alpha)C_\alpha v_l + b(\alpha)C_\alpha w_l \neq 0$ with $C_\alpha v_l$ and $C_\alpha w_l$ the constant coefficients of $v_l(z)$ and $w_l(z)$ respectively with respect to the finite point α .

The two rational expressions $v_{a,b}(z)$ and $v_{a',b'}(z)$ are equivalent, i.e. represent the same rational form if and only if $(a(z), b(z)) = k(a'(z), b'(z))$ with $k \in \mathbb{F}$.

Proof. From the previous, it is clear that the matrix formed by the highest potential degree coefficients of $v_l(z)$ and $w_l(z)$ has at least rank one. The matrix formed by the constant coefficients of $v_l(z)$ and $w_l(z)$ with respect to a finite interpolation point α also has at least rank one. Together with theorem 6.9, this proves the first part of the theorem. The two rational expressions $v_{a,b}(z)$ and $v_{a',b'}(z)$ are equivalent if and only if by definition 2.15

$$\det \begin{bmatrix} n_{a,b}(z) & n_{a',b'}(z) \\ d_{a,b}(z) & d_{a',b'}(z) \end{bmatrix} = 0.$$

From the following relationship

$$\begin{bmatrix} n_{a,b}(z) & n_{a',b'}(z) \\ d_{a,b}(z) & d_{a',b'}(z) \end{bmatrix} = \begin{bmatrix} n_{l,v}(z) & n_{l,w}(z) \\ d_{l,v}(z) & d_{l,w}(z) \end{bmatrix} \begin{bmatrix} a(z) & a'(z) \\ b(z) & b'(z) \end{bmatrix}$$

and the fact that

$$\begin{bmatrix} n_{l,v} & n_{l,w} \\ d_{l,v} & d_{l,w} \end{bmatrix} \neq 0,$$

we can conclude that $v_{a,b}(z)$ and $v_{a',b'}(z)$ are equivalent if and only if

$$\begin{bmatrix} a & a' \\ b & b' \end{bmatrix} = 0,$$

i.e. $ab' - ba' = 0$. From the minimality of $v_{a,b}(z)$ and $v_{a',b'}(z)$, it follows that $a(z)$ and $b(z)$ have no nontrivial polynomial common factors and neither have $a'(z)$ and $b'(z)$. Because the polynomial couples $(a(z), b(z))$ and $(a'(z), b'(z))$ are nontrivial, the condition $ab' - ba' = 0$ is equivalent to $(a(z), b(z)) = k(a'(z), b'(z))$ with k a constant and not a polynomial. This proves the theorem completely. $\square$

For additional information concerning the following notes, we refer to [24] and [26].

Notes

- From the two by two polynomial matrices $V_j(z), j = 1, 2, \dots$, constructed in algorithm 1, it is easy to build up a continued fraction whose convergents, for the occasion defined as polynomial couples, are $v_j(z)$ or $w_j(z)$. Given l and $a(z), b(z)$ of theorem 1, we can even derive a continued fraction, whose last convergent is the polynomial couple $v_{a,b}(z)$. Contracting these continued fractions, i.e. grouping together consecutive $V_j(z)$ matrices, we can obtain continued fractions equivalent to those described by Gutknecht [15]. Therefore, our algorithm can be viewed as a Berlekamp-Massey version of the Euclidean-like algorithm of Gutknecht.
- Most of the computational effort, needed in algorithm 1, is to compute the residuals to get the value of h . Algorithm 1 is a layer adjoining algorithm, indicating that we use the polynomial couples $v_l(z)$ and $w_l(z)$ to compute the residuals and adding a next layer, i.e. using $V_{l+1}(z)$ to compute $v_{l+1}(z)$ and $w_{l+1}(z)$. A layer peeling version starts from initial residuals and updates these residuals at each step. The number of retained residuals decreases with each step. Therefore, it is called a layer peeling method. See also [8].
- Besides computing residuals, the basic polynomial operation is a multiplication by z , i.e. the shift operation. To implement the algorithm, one can use orthogonal polynomials instead of the classical basis $1, z, z^2, \dots$. The shift operation can be performed efficiently due to the three term recurrence relation for orthogonal polynomials.

7 A small example

We shall now work out a small example in full detail. Let us look for a parametrization of all polynomial couples $(n(z), d(z))$ having minimal $(s_1, s_2) = (0, 0)$ -degree satisfying some rational interpolation conditions. We shall consider interpolation conditions at the interpolation points $0, 1$ and ∞ with

- $\lambda_\infty = 1, \mu_\infty = 0, \tau_\infty = 2, \text{ord}_\infty(M_\infty) = 0, \beta = 0,$
- $\lambda_0 = 0, \mu_0 = 0, \tau_0 = 2, \text{ord}_0(M_0) = 0,$
- $\lambda_1 = 1, \mu_1 = 0, \tau_1 = 2, \text{ord}_1(L_1) = 0.$

We shall use algorithm 1 as a basis to explain what happens when solving the **LRIP** and equivalent **PRIP** for this example.

1. Algorithm 1 initializes the polynomial couples $v_0(z)$ and $w_0(z)$ and their potential degrees as follows

$$\begin{bmatrix} v_0(z) & w_0(z) \\ \delta_{0,v} & \delta_{0,w} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}.$$

2. Let us consider a first interpolation condition in the interpolation point 0. Given $L_0^1(z) = 1$, we also need the first coefficient of $M_0(z)$ to get the first coefficient of the quotient series. Let $M_0^1(z) = 1$. From the data it follows that $\text{ord}_0(L_0) = 0$. Hence, we can use (6.7) to determine the final value l_0 which turns out to be $l_0 = 2$. Therefore, we shall need the first two coefficients of L_0 and of M_0 . Using algorithm 3, we find $h = R_1 w_0 / R_1 v_0 = -1/1$ and

$$\begin{bmatrix} v_1(z) & w_1(z) \\ \delta_{1,v} & \delta_{1,w} \end{bmatrix} = \begin{bmatrix} 1 & z \\ 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Note that one of the two polynomial couples $v_1(z)$ and $w_1(z)$ will be divisible by $z - 0$ because we are not considering a trivial initial condition in the interpolation point 0 as indicated by lemma 6.5.

3. The next interpolation condition will be taken in the interpolation point 1. Let $M_1^1(z) = 0$. Hence, for the moment we only need to know that $L_1^1 \neq 0$. By lemma 6.5, this condition is a trivial initial condition in 1. Therefore,

$$\begin{bmatrix} v_2(z) & w_2(z) \\ \delta_{2,v} & \delta_{2,w} \end{bmatrix} = \begin{bmatrix} v_1(z) & w_1(z) \\ \delta_{1,v} & \delta_{1,w} \end{bmatrix}$$

and $v_2(z)$ as well as $w_2(z)$ are not divisible by $z - 1$. Although we only know that $\text{ord}_1(M_1) \geq 1$, we can use (6.7) to determine the final value for $l_1 = 2$. Hence, we shall only need the first initial coefficient of L_1 and the first two initial coefficients of M_1 .

4. The next interpolation condition is an interpolation condition at infinity. Given $L_\infty^1 = 0$, it is clear that we only need to know that $M_\infty^1 \neq 0$. The exact value of M_∞^1 can be set equal to m_1 . Following algorithm 2, we derive that $h = R_3^1 w_2 / R_3^0 v_2 = m_1 / m_1$. Therefore,

$$\begin{bmatrix} v_3(z) & w_3(z) \\ \delta_{3,v} & \delta_{3,w} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -z & 1 \\ 1 & 1 \end{bmatrix}.$$

This result is independent of the choice we can make for m_1 at this moment. Note that because the interpolation condition we have considered in this step is not a trivial initial condition as shown by lemma 6.4, we know that one of the highest potential degree coefficients of v_3 and w_3 is equal to the zero vector. Note that we don't need lemma 6.4 or lemma 6.5 to determine if the initial condition(s) are trivial or not. We can easily see this if both residuals into play are zero. The final value for l_∞ can be easily computed using (6.8), $l_\infty = 3$. Hence, we know that at level 7 the final solution can be written down using the polynomial couples $v_7(z)$ and $w_7(z)$.

5. We shall take the next interpolation condition at 0. We need the second coefficients of L_0 and M_0 , $L_0^2 = 1 + 2z$ and $M_0^2 = 1 + 1z$. Using algorithm 3, we compute

$h = R_4 w_3 / R_4 v_3 = -1/1$. Therefore,

$$\begin{bmatrix} v_4(z) & w_4(z) \\ \delta_{4,v} & \delta_{4,w} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -z+1 & -z^2 \\ 1 & 2 \end{bmatrix}.$$

Because we have considered already two linearized interpolation conditions at zero, we don't need to consider the interpolation point zero anymore in the sequel.

6. We shall take the next interpolation condition at infinity. Note that the final value for $l_\infty = 3$. Hence, after this step, we need another additional step. It can easily be seen that we don't need any additional data at infinity right now. Therefore, the following data at infinity will be used in one of the next steps (in fact, the following one). Following algorithm 2, we get the value of $h = R_5^2 w_4 / R_5^1 v_4 = 0/m_1$. Hence,

$$\begin{bmatrix} v_5(z) & w_5(z) \\ \delta_{5,v} & \delta_{5,w} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -z^2 & -z+1 \\ 2 & 2 \end{bmatrix}.$$

7. Now we consider the final linearized interpolation condition at infinity. Let $L_\infty^2 = 0 + 1z^{-1}$ and $M_\infty^1 = 1$, i.e. $m_1 = 1$. Following algorithm 2, we compute $h = R_6^2 w_5 / R_6^2 v_5 = 1/1$. So,

$$\begin{bmatrix} v_6(z) & w_6(z) \\ \delta_{6,v} & \delta_{6,w} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ z^2 - z + 1 & -z^2 \\ 2 & 3 \end{bmatrix}.$$

8. In this last step, we shall consider the final linearized interpolation condition at 1. Hence, we need the first coefficient of $L_1(z)$ and the first two coefficients of $M_1(z)$, i.e. $L_1^1(z) = 1$ and $M_1^2(z) = 0 + 2(z-1)$. From algorithm 3, we derive that $h = R_7 w_6 / R_7 v_6 = 1/1$. Therefore,

$$\begin{bmatrix} v_7(z) & w_7(z) \\ \delta_{7,v} & \delta_{7,w} \end{bmatrix} = \begin{bmatrix} -1 & z-1 \\ -2z^2 + z - 1 & z^3 - 2z^2 + 2z - 1 \\ 3 & 3 \end{bmatrix}.$$

Using theorem 6.11, we can express all solutions $v_{a,b}(z) = (n_{a,b}(z), d_{a,b}(z))$ of minimal (s_1, s_2) -degree 3 of the **LRIP** and the equivalent **PRIP** based on two free parameters $a, b \in \mathbf{F}$ as

$$\begin{bmatrix} n_{a,b}(z) \\ d_{a,b}(z) \end{bmatrix} = a \begin{bmatrix} -1 \\ -2z^2 + z - 1 \end{bmatrix} + b \begin{bmatrix} z-1 \\ z^3 - 2z^2 + 2z - 1 \end{bmatrix}$$

with the following conditions:

- $b \neq 0$, because the highest potential degree coefficient of $v_{a,b}(z)$ has to different from the zero vector,
- $a + b \neq 0$, because $v_{a,b}(z)$ has to be indivisible by $z - 0$,
- $a \neq 0$, because $v_{a,b}(z)$ has to be indivisible by $z - 1$.

The reader can check that $v_{a,b}(z)$, for each value of a and b with the conditions above, satisfies the proper and linearized interpolation conditions. Also $v_{a,b}(z)$ and $v_{a',b'}(z)$ with $(a,b) \neq k(a',b')$ with $k \in \mathbf{F}$ represent different rational forms or, by lifting this result from the "formal" to the "function" level, represent a different rational function.

8 Conclusion

To take away the disadvantages of the Newton-Padé rational interpolation approach, we have posed a more general rational interpolation problem. This general problem setting allows confluent interpolation points, pole information, ∞ as interpolation point. Also, it circumvents the problem of unattainable points. A nice and fast recursive algorithm solving this general problem was developed handling all interpolation data in a similar but distinct recursive step in contrast to most of the older algorithms grouping several interpolation points together to compute divided differences. The basic steps of our algorithm are very simple and can be seen as a generalization of the basic steps of the Berlekamp-Massey algorithm. The algorithms using divided differences follow the Euclidean approach computing in each step "polynomial" instead of "scalar" information. The generality of our problem setting allows the algorithm to compute not only a solution of the linearized but also of the proper rational interpolation problem. Moreover, all solutions can be written down using an elegant parametrization. Writing the problem in linear algebra terms, has shown the importance of the new algorithm with respect to several structured matrices.

There are several possibilities for future research. We shall mention some of them. In an informal way, we have already extended the results to the rational interpolation problem with vector data [27]. Using the polynomial matrix framework, which we used in a minimal partial realization context, the matrix case could be handled in a similar way. In [1], Anderson and Antoulas generalize their scalar results of [4] to the matrix case, solving the matrix **RIP** in a nonrecursive way using not only a polynomial but also a state-space approach. Can this state-space description for the solution also be adopted in a recursive setting? For the special case of Padé approximants, we have used the vector results to define minimal vector Padé approximants. How do these results extend to the matrix and the Newton-Padé case? Can we construct vector and matrix tables? The new algorithm constructed in this paper can be seen as a way to walk through the Newton-Padé table along a diagonal. How does the algorithm change when following other paths such as rows, columns, etc? In [2], Antoulas handles the antidiagonal walk using the Euclidean algorithm. What is the full significance of our new algorithm on handling structured matrices [7]?

References

- [1] B.D.O. Anderson and A.C. Antoulas. Rational interpolation and state variable realizations. *Lin. Alg. Appl., Special Issue on Matrix Problems*, 1989. Submitted.
- [2] A.C. Antoulas. Rational interpolation and the Euclidean algorithm. *Lin. Alg. Appl.*, 108:157-171, 1988.
- [3] A.C. Antoulas and B.D.O. Anderson. On the scalar rational interpolation problem. *IMA J. Math. Control and Information*, 3:61-88, 1986.
- [4] A.C. Antoulas and B.D.O. Anderson. State-space and polynomial approaches to rational interpolation. In *Proceedings MTNS89*, 1989.
- [5] Jr. G.A. Baker and P. Graves-Morris. *Padé Approximants. Part I : Basic Theory*. Addison-Wesley, Reading, Mass., 1981.
- [6] Jr. G.A. Baker and P. Graves-Morris. *Padé Approximants. Part II : Extensions and Applications*. Addison-Wesley, Reading, Mass., 1981.

- [7] V. Belevitch. Interpolation matrices. *Philips Res. Rep.*, 25:337–369, 1970.
- [8] A. Bruckstein and T. Kailath. An inverse scattering framework for several problems in signal processing. *IEEE Acoust. Speech Signal Process. magazine*, pages 6–20, 1987.
- [9] G. Claessens. On the structure of the Newton-Padé table. *J. Approx. Theory*, 22:304–319, 1978.
- [10] F. Cordellier. Utilisation de l'invariance homographique dans les algorithmes de losanges. In H.J. Bünger H. Werner, editor, *Padé Approximation and its Applications*, pages 62–94, Bad Honnef, 1983. Lecture Notes in Math. 1071.
- [11] W.B. Gragg. The Padé table and its relation to certain algorithms of numerical analysis. *SIAM Rev.*, 14:1–62, 1972.
- [12] P.R. Graves-Morris. Practical, reliable, rational interpolation. *J. Inst. Maths. Applics.*, 25:267–286, 1980.
- [13] P.R. Graves-Morris and T.R. Hopkins. Reliable rational interpolation. *Numer. Math.*, 36:111–128, 1981.
- [14] M.H. Gutknecht. The rational interpolation problem revisited. Technical Report 88-01, ETH Zürich, 1988. Appeared in *Rocky Mountain J. Math.*
- [15] M.H. Gutknecht. Continued fractions associated with the Newton-Padé table. *Numer. Math.*, 56:547–589, 1989.
- [16] G. Heinig, W. Hoppe, and K. Rost. Structured matrices in interpolation and approximation problems. *Wiss. Z. d. TU Karl-Marx-Stadt*, 31:196–202, 1989.
- [17] R.E. Kalman. On partial realizations, transfer functions, and canonical forms. *Acta Polytech. Scand.*, 31:9–32, 1979.
- [18] K. Löwner. Über monotone Matrixfunktionen. *Math. Z.*, pages 177–216, 1934.
- [19] A. Magnus. Certain continued fractions associated with the Padé table. *Math. Z.*, 78:361–374, 1962.
- [20] A. Magnus. Expansion of power series into P-fractions. *Math. Z.*, 80:209–216, 1962.
- [21] J. Meinguet. On the solubility of the Cauchy interpolation problem. In A. Talbot, editor, *Approximation theory*, pages 137–163, London and New York, 1970. Academic Press.
- [22] H. Padé. Sur la représentation approchée d'une fonction par des fractions rationnelles. *Ann. de l'Ecole Normale Sup. 3ieme serie*, 9:3–93, 1892.
- [23] H. Rutishauser. Der Quotienten-Differenzen-Algorithmus. *Zeit. für Angewandte Math.*, 5:233–251, 1954.
- [24] M. Van Barel. *Nested Minimal Partial Realizations and Related Matrix Rational Approximants*. PhD thesis, K.U. Leuven, January 1989.
- [25] M. Van Barel and A. Bultheel. Minimal Padé sense matrix approximations around $s = 0$ and $s = \infty$. In A. Cuyt, editor, *Nonlinear numerical methods and rational approximation*, pages 143–154, Dordrecht, 1989. Reidel Publ. Comp.

- [26] M. Van Barel and A. Bultheel. A new approach to the rational interpolation problem. *J. Comp. Appl. Math.*, September 1990. Accepted.
- [27] M. Van Barel and A. Bultheel. A new approach to the rational interpolation problem: the vector case. Technical Report TW130, Department of Computer Science, K.U. Leuven, July 1990.
- [28] H. Werner. A reliable method for rational interpolation. In L. Wuytack, editor, *Padé approximation and its applications*, pages 257–277, Berlin-Heidelberg-New York, 1979. Springer Verlag.
- [29] H. Werner. Ein Algorithmus zur rationalen Interpolation. In H. Werner L. Collatz, G. Meinardus, editor, *Numerische Methoden der Approximationstheorie*, 5, pages 319–337, Basel, 1980. Birkhäuser-Verlag. ISNM 52.
- [30] H. Werner. Altes und Neues zur rationalen Interpolation, September 1981. Vortrag 10 Österreichischen Mathematiker Kongress, Innsbruck.
- [31] H. Werner. Algorithm 51 : A reliable and numerically stable program for rational interpolation of Lagrange data. *Computing*, 31:269–286, 1983.
- [32] L. Wuytack. An algorithm for rational interpolation similar to the QD-algorithm. *Numer. Math.*, 20:418–424, 1973.
- [33] L. Wuytack. Extrapolation to the limit by using continued fraction interpolation. *Rocky Mountain J. Math.*, 4:395–397, 1974.
- [34] G.-L. Xu and A. Bultheel. The problem of matrix Padé approximation. Technical report, Department of Computer Science, K.U. Leuven, November 1988.