The University of Alberta

Linear Systems with Integral Domain Polynomial Coefficients

by

Gordon Harold Atwood

A thesis
submitted to the Faculty of Graduate Studies and Research
in partial fulfillment of the requirements for the degree
of Master of Science

Department of Computing Science

Edmonton, Alberta
Spring, 1987

ISBN 0-315-37732-1

THE UNIVERSITY OF ALBERTA

RELEASE FORM

NAME OF AUTHOR: Gordon Harold Atwood

TITLE OF THESIS: Linear Systems with Integral Domain Polynomial Coefficients

DEGREE FOR WHICH THIS THESIS WAS PRESENTED: Master of Science

YEAR THIS DEGREE GRANTED: 1987

(Signed) Gordon H. Atwood

Permanent Address:
1903 6th Avenue South
Lethbridge, Alberta
Canada T1J 1B8

Date: April 13/1987

THE UNIVERSITY OF ALBERTA

FACULTY OF GRADUATE STUDIES AND RESEARCH

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research, for acceptance, a thesis entitled **Linear Systems with Integral Domain Polynomial Coefficients** submitted by **Gordon Harold Atwood** in partial fulfillment of the requirements for the degree of **Master of Science**.

..........Stan Cabay..........

Supervisor

..........Gerald Cliff..........

..........[illegible]..........

..........Robert A. Reckhow..........

Date: April 9, 1987

to Irenka

ABSTRACT

A new method for solving systems of linear equations with univariate polynomial components over an integral domain is developed and analyzed. The method first constructs a truncated power series representation of the solution and, if required, the solution is then converted into rational form.

The conversion is accomplished by constructing a sequence of Padé approximants for the components of the solution vector of power series. The algorithm developed for this purpose is two orders of magnitude faster than competitive methods.

It is shown that the proposed algorithm is asymptotically superior to competitive methods whenever either the solution is required in power series form, or in reduced rational form. On the other hand, if the solution is required in rational form, but not necessarily in reduced rational form, it is shown that fraction-free methods are asymptotically superior.

Acknowledgements

I wish to thank my supervisor, Dr. Stan Cabay for the time, patience, knowledge, and valuable insight that he provided during the the development of this thesis†. His expertise in Padé theory was the driving force in many of the results herein and his wisdom and guidance were very much appreciated.

I credit Dr. Piotr Kossowski, since many of the results in Chapter 2 would not have been possible had he not laid the groundwork in his own doctoral dissertation.

I thank the members of my examining committee, Dr. B. Joe, Dr. R. Reckhow, and Dr. G. Cliff, for their time and effort in reviewing this research, and Dr. W. Davis for chairing the committee.

I also thank Ted Bentley for his help and expertise with document preparation, and Chung Hee Hwang for her comments on Chapter 1.

I dedicate this thesis to Dr. Stan Devitt, Dr. Don Ferguson, Dr. John Hiscocks, and Prof. Frank Schaffer, and I thank them for their encouragement and help during my undergraduate studies.

Finally, I further dedicate this thesis to my parents and to my wife Irene.

† Funded, in part, by a Postgraduate NSERC scholarship.

Table of Contents

List of Tables

Chapter 1

Introduction

Let $\mathbf{D}$ be an integral domain, and let $\mathbf{D}[x]$ denote the domain of polynomials in the indeterminate x with coefficients from $\mathbf{D}$. Furthermore, let $\mathbf{D}[[x]]$ denote the domain of formal power series in the indeterminate x with coefficients from $\mathbf{D}$. Then, $\mathbf{D}[x]$ and $\mathbf{D}[[x]]$ are integral domains, and $\mathbf{D} \subset \mathbf{D}[x] \subset \mathbf{D}[[x]]$.

In the development of this thesis we make reference to the field of quotients $\mathbf{F}_D$ for $\mathbf{D}$ ($\frac{a}{b} \in \mathbf{F}_D$ *iff* $a,b \in \mathbf{D}$ and $b \neq 0$). We then refer to the corresponding polynomial domain $\mathbf{F}_D[x]$, and power series domain $\mathbf{F}_D[[x]]$. Finally, the field of rational functions over $\mathbf{D}$ in the indeterminate x (i.e., the field of quotients for $\mathbf{D}[x]$) is denoted by $\mathbf{D}(x)$.

In this thesis we consider the problem of solving for $F(x) \in \mathbf{D}_n(x)$ the system of linear equations

$$M(x) \cdot F(x) = G(x) \quad (1.1)$$

where $M(x)$ is a matrix of order n and $G(x)$ is a vector of length n, both with components in D[x]. Notationally, $M(x) \in \mathbf{D}_{n,n}[x]$ and $G(x) \in \mathbf{D}_n[x]$. To describe the effectiveness of various algorithms discussed for solving (1.1), we assume that the degrees of the polynomials in $M(x)$ and $G(x)$ are all bounded by δ. In addition, we assume that the sizes (in D) of the coefficients of the polynomials in $M(x)$ and $G(x)$ are bounded by ∂. The size of an element $d \in \mathbf{D}$ is (loosely speaking) the amount of storage required to represent d on a computer, in terms of bits or words.

The problem of solving (1.1) when D is a field has been investigated extensively. For such a domain the usual assumption in the development and analysis of algorithms is that all operations in D require a constant amount of time. Thus, it is adequate to assume that $\partial = 1$. In a computing environment this assumption is justified for some fields (e.g., the field of integers modulo p, or the field of real numbers if floating point numbers provide an adequate approximation), but not for others (e.g., the field of rational numbers for which the cost of an operation depends on the size of the rational numbers involved). When D is a field, the algorithms commonly used for solving (1.1) can be classified into three general categories.

The first is the category of fraction-free methods developed during the 1960's by Bareiss [1] and Lipson [11], which are based on Gaussian elimination. When performing a row operation during the triangular decomposition of the matrix $M(x)$ by Gaussian elimination, the pivot row and each target row are multiplied by appropriate elements from $D[x]$ to ensure that the resulting row is contained in $D[x]$ rather than in $D(x)$ (hence the name fraction-free). However, the effectiveness of the various fraction-free methods is realized only when factors known to exist (and requiring no *GCD* computations) are systematically removed from the resulting rows. As a consequence, the degrees of the components grow linearly rather than exponentially during the elimination. Higginson [9] shows that the fraction-free algorithms require $O(\delta^2 n^5)$ operations in D.

In the second category are the modular (or congruential) and evaluation-interpolation methods developed and studied by McClellan [14], Lipson [12], and others in the 1970's. Modular techniques are applicable when the domain ($D[x]$ in our case) is a Principal Ideal Domain. First the linear system (1.1) is solved modulo a set of ideals (i.e., in our case, a set of relatively prime polynomials). The solution in $D_n[x]$ is then obtained by applying the Chinese Remainder Theorem to these results. When the ideals all correspond to relatively prime polynomials of degree one (i.e., when the generators of the ideals are polynomials of degree one), a modulo operation corresponds to evaluation and the Chinese Remainder Theorem is equivalent to interpolation (c.f., Lipson [13]). Note, however, that evaluation-interpolation methods can be applied to problems outside Principal Ideal Domains (see McClellan [14], who applies this method to multivariate polynomials over a field). For our problem, where $D[x]$ is the domain of polynomials over a field, the evaluation-interpolation method requires the solution of (1.1) at $\delta n+1$ points, with a cost of $O(\delta n^4)$ operations in D. The interpolation step to reconstruct the solution in $D_n[x]$ requires an additional $O(\delta^2 n^3)$ operations in D. Hence, the complexity of evaluation-interpolation methods (and therefore, also, the complexity of modular methods) is $O(\delta^2 n^3+\delta n^4)$ (see McClellan [14] for a detailed analysis).

The third and final category contains the power series methods developed mainly in the 1980's. They were first proposed by Moenck and Carter [15] and involve the construction of successive terms in a power series representation $\overline{F}(x)$ of $F(x)$ in (1.1). After $2\delta n+1$ such terms have been computed, Padé theory allows the construction of the exact solution $F(x) \in D_n(x)$ from $\overline{F}(x)$ (we postpone any detailed discussion of Padé theory until Chapter 2). Cabay and Domzy [3] improve the power series method and show that its complexity is $O(\delta^2 n^3)$.

In addition to the computational superiority of power series methods over fraction-free and modular (evaluation-interpolation) methods, they provide a truncated power series solution to (1.1) as an intermediate result. Such a solution is useful in Markov chains, flowgraphs, coding theory, and circuit theory (c.f. Lipson [11], [12] for further discussion). Another advantage of power series methods is that the final rational solution obtained is in reduced form, which is not the case with either the fraction-free or modular (evaluation-interpolation) methods.

When the linear system (1.1) has sparse polynomial components none of the above three categories of methods is particularly suitable and several other techniques provide significantly better complexity results. This is the central issue of Horowitz and Sahni [10], who review and compare several methods under three general models of sparsity. Gentleman and Johnson [6], Griss [8], and Smit [16] also present variations of these algorithms and analyze their complexities under sparse conditions.

The problem of solving (1.1) when D is not a field has also received attention, but mainly for integral domains of a special nature, such as when D is the domain of integers or when D is itself the domain of polynomials. McClellan [14] and Lipson [12] explore modular methods to reduce the problem (with coefficients over the integral domain D) to a suitable number of similar problems with coefficients over a field, rather than over D. The reduced problem over a field can be solved by any of the methods already described, and the solution over D can then be reconstructed with the Chinese Remainder Theorem.

In our research, we make no underlying assumptions about the structure of D. As a consequence, the methods available for solving (1.1) are the fraction-free methods of Bareiss [1] and Lipson [11]. Generalizing the cost analysis performed by Higginson [9] from the integers to an arbitrary integral domain, it is easy to determine that the cost of fraction-free methods for solving (1.1) over an arbitrary integral domain D is $O(\partial^2\delta^2 n^7)$ unit operations in D. By unit operation we mean the time required to perform an operation in D relative to how the operands are stored. If storage is measured in terms of bits, then a unit operation is the cost of a bit operation. If storage is measured in terms of words, then a unit operation is the cost of a word operation.

In Chapter 3 we generalize the power series method for solving (1.1) from a field to an arbitrary integral domain **D**. This method has three stages. First, a truncated power series solution $\overline{F}(z)$ in $\mathbf{D}_n[z]$ is computed, where $x = dz$ is a substitution defined in Chapter 3. We note that this is not, directly, a solution to (1.1), but with a simple substitution we can obtain the truncated power series solution $\overline{F}(x)$ in $\mathbf{F}_{D_n}[x]$. Using $\overline{F}(z)$, a rational solution $F(z)$ in $\mathbf{D}_n(z)$ is calculated by making use of Padé theory. Finally, the inverse substitution yields the exact solution $F(x)$ in $\mathbf{D}_n(x)$ to (1.1).

The cost of computing $2\delta n+1$ terms of the power series solution is $O(\partial^2\delta^4 n^5 + \partial^2\delta^2 n^6)$. This, we shall show, is sufficient to completely characterize the corresponding rational functions. As noted earlier, such a solution is of use in flowgraphs and other problem areas. In applications where fewer than $2\delta n+1$ terms of the power series solution are required, our method is therefore a clear winner over fraction-free methods, except, perhaps, for large δ and small n.

When the solution to (1.1) is required in rational form, the conversion of the truncated power series solution to rational form is required. Conversion algorithms are the subject of Chapter 2, and the new results on Padé theory obtained therein are contributions in their own right.

Padé fractions over a field have a long established history in mathematics with applications in a variety of fields (Cragg [7] provides an excellent survey). Cabay and Choi [2] develop a fast algorithm for their computation and provide a comparative study with other algorithms.

For the computation of Padé fractions over an arbitrary integral domain, the algorithms available are those of Geddes [5] and Cabay and Kossowski [4]. For the conversion of the solution vector of power series to rational form, the algorithm JPADE of Cabay and Kossowski (which they show to be superior to Geddes') requires $O(\partial^2\delta^6 n^9)$ unit operations in **D** when the power series are normal, and may require up to $O(\partial^2\delta^7 n^{10})$ unit operations in the abnormal case.

The cost of conversion using JPADE is therefore not satisfactory, and in Chapter 2 we take advantage of the special property of the power series in the solution vector (namely, that they are expansions of rational functions) to reduce the cost of JPADE. The resulting algorithm, using $\overline{Jpade}$, has a complexity of $O(\partial^2\delta^4 n^7)$ when all of the power series are normal, and can reach a complexity of $O(\partial^2\delta^5 n^8)$ in the abnormal case. It should be noted that severely abnormal problems are highly exceptional, and for practical purposes the complexity of the conversion, using $\overline{Jpade}$, is realistically $O(\partial^2\delta^4 n^7)$.

It is shown in Chapter 2 that when the integral domain is a *GCD* domain (c.f. Lipson [13], pp. 132), the rational functions obtained by $\overline{Jpade}$ are in reduced form, except, perhaps, for some elements from **D**. On the other hand, the solution obtained by fraction-free methods is not in reduced form. Because of the duality of JPADE with Euclid's extended algorithm (c.f., Cabay and Kossowski [4]), the best available algorithm for reducing the rational solution obtained from the fraction-free methods is JPADE. The cost of this reduction is exactly the cost of converting the power series solutions to rational form using $\overline{Jpade}$, that is, $O(\partial^2\delta^4 n^7)$ unit operations in **D** in the normal case, and $O(\partial^2\delta^5 n^8)$ otherwise. Since it is less costly to compute $2\delta n+1$ terms in the power series representation of the solution than to compute an unreduced rational representation by the fraction-free methods, the power series method is asymptotically superior.

For problems where the solution is only required in rational form, but not necessarily in reduced form, the $O(\partial^2\delta^2 n^7)$ cost of the fraction-free methods is asymptotically smaller than the total cost of the power series method using $\overline{J}pade$, which is $O(\partial^2\delta^4 n^7 + \partial^2\delta^4 n^5 + \partial^2\delta^2 n^6)$ in the normal case, and $O(\partial^2\delta^5 n^8 + \partial^2\delta^4 n^5 + \partial^2\delta^2 n^6)$ otherwise. For practical purposes, an implementation is required to provide a true test of superiority between the two methods.

The following notation is adopted throughout this thesis. Complex objects such as matrices and vectors are given capitalized names. Their components are similarly named and delineated by subscripts. Thus, if $G(x) \in \mathbf{D}_n[x]$, then the ith component of the $G(x)$ is denoted $G_i(x)$. Simple elements such as scalars, polynomials and power series are given lower-case names. Their components are similarly named and delineated by parenthesized superscripts. For example, if $A(x) \in \mathbf{D}_n[[x]]$, then

$$A(z) = \sum_{i=0}^{\infty} A^{(i)} x^i \quad .$$

where $A^{(i)} \in \mathbf{D}_n$ for all $i \geq 0$. In Chapter 2 it is necessary to make use of sequences. We find it convenient to use subscripts to delineate the elements of the sequence. Due to the order of presentation, it is not possible to confuse elements of a sequence with components of a vector or a matrix as only simple (lower-case) objects are used in sequences. Moreover, the use of sequences is confined to Chapter 2, whereas the use of vectors and matrices is confined to Chapter 3.

Chapter 2

Padé Fractions Over an Integral Domain

In this chapter we examine the problem of constructing Padé fractions for a univariate power series with coefficients over an integral domain **D** when the power series is known to satisfy certain constraints. The algorithm developed for this construction is used in Chapter 3 to solve linear systems of polynomials over this same integral domain **D**.

As noted in Chapter 1, substantial work has been done in Padé theory for power series over a field. Until recently, however, solutions to problems in an integral domain $D[[z]]$ were constructed by first embedding them in $\mathbf{F}_D[[z]]$ and then applying conventional algorithms to obtain results in $\mathbf{F}_D(z)$. These could then be converted into results in $\mathbf{D}(z)$ in a systematic manner (c.f., Cabay and Kossowski [5]).

Unfortunately, operations in $\mathbf{F}_D$ experience large growth in the intermediate coefficients used in the computation of the solution. *GCD* algorithms may be used to remove common factors at each step, but this does not provide a reasonable time complexity. Geddes [6] presents a method for computing Padé fractions where all operations are performed in **D**. His method involves specifying a linear system (a Hankel system) whose solution is the desired Padé fraction and then solving this system using fraction-free methods as described by Bareiss [1].

Cabay and Kossowski [5] unite the algorithm of Cabay and Choi [3] for the computation of scaled Padé fractions with algorithms for computing greatest common divisors of polynomials (c.f., Brown [2]) to obtain their algorithm JPADE. JPADE also performs all its operations in **D**, but is computationally superior to Geddes' algorithm.

In this chapter we present a new algorithm $\overline{Jpade}$ which, for a pair of power series $\left(a(z),b(z)\right)$ satisfying certain constraints, is able to solve the problem substantially faster than JPADE.

In Sections 2.1 and 2.2 we present some of the central notions of power series remainder sequences of Cabay and Kossowski required to understand JPADE. In Sections 2.3 and 2.4 we present the special constraints on $a(z)$ and $b(z)$ that we have hinted at, and relate Cabay and Kossowski's results to our own. In Sections 2.5 and 2.6 we give the algorithm $\overline{Jpade}$ and a cost analysis. Finally, in Section 2.7 we explain how $\overline{Jpade}$ may be of use in solving linear systems given by (1.1).

We introduce the two formal power series

$$a(z) = \sum_{i=0}^{\infty} a^{(i)} z^i \in \mathbf{D}[[z]] \tag{2.1}$$

and

$$b(z) = \sum_{i=0}^{\infty} b^{(i)} z^i \in \mathbf{D}[[z]] \ , \tag{2.2}$$

where $b^{(0)} \neq 0$. These definitions will be used throughout this chapter.

2.1. Power Series Remainder Sequences

In attempting to deal with the problem of computing the GCD of two polynomials, the concept of a polynomial remainder sequence is introduced (c.f., Brown [2]). This depends on the notion of polynomial pseudo-division (c.f., Knuth [12], pp. 364-377). In this section we detail Cabay and Kossowski's extension of this concept to power series remainder sequences and power series pseudo-division.

Definition 2.1. Given the power series $a(z)$ and $b(z)$ defined by (2.1) and (2.2), and a non-negative integer s_0, let

$$r_{-1}(z) = a(z) , \quad (2.3)$$
$$r_0(z) = b(z) , \quad (2.4)$$

and

$$s_{-1} = 1 . \quad (2.5)$$

The sequence

$$\{s_i, r_i(z)\}_{i=-1,0,\ldots} \quad (2.6)$$

is said to be a **power series remainder sequence (PSRS)** for $\left(a(z), b(z)\right)$ with respect to s_0 provided that

$$\alpha_{i+1}\, r_{i-1}(z) + \omega_{i+1}(z) \cdot r_i(z) = z^{s_i + s_{i+1}}\, \beta_{i+1}\, r_{i+1}(z), \; i=0,\ldots , \quad (2.7)$$

where for all $i \geq 0$,

(1) $\alpha_{i+1}, \beta_{i+1} \in \mathbf{D}$,

(2) $\omega_{i+1}(z) = \sum_{j=0}^{s_i} \omega_{i+1}^{(j)} z^j \in \mathbf{D}[z]$,

(3) $r_{i+1}(z) \in \mathbf{D}[[z]]$, and

(4) s_{i+1} is either a positive integer or $s_{i+1} = +\infty$. By convention, $s_{i+1} = +\infty$ if and only if $r_{i+1}(z) = 0$. Otherwise, s_{i+1} is maximal such that $r_{i+1}(0) \neq 0$.

In (2.7), $\omega_{i+1}(z)$ and $r_{i+1}(z)$ are called the **power series pseudo-quotient** and **power series pseudo-remainder**, respectively, on pseudo-division of $r_{i-1}(z)$ by $r_i(z)$ relative to s_i.

■

Corresponding to the PSRS (2.6), Cabay and Kossowski [5] also give

Definition 2.2. Given the PSRS (2.6), the sequence

$$\{u_i(z), v_i(z)\}_{i=-1,0,\ldots} \tag{2.8}$$

is the cofactor sequence for $\left(a(z), b(z)\right)$ if $u_i(z), v_i(z) \in D[z]$, $i = 0, \ldots$,

$$u_{-1}(z) = 0, \qquad v_{-1}(z) = z^{-s_0-1},$$
$$u_0(z) = 1, \qquad v_0(z) = 0 \tag{2.9}$$

and for $i = 0, 1, \ldots$, $u_{i+1}(z)$ and $v_{i+1}(z)$ are determined by

$$\beta_{i+1}\, u_{i+1}(z) = z^{s_i + s_{i-1}}\, \alpha_{i+1} u_{i-1}(z) + \omega_{i+1}(z) \cdot u_i(z)\,, \tag{2.10}$$
$$\beta_{i+1}\, v_{i+1}(z) = z^{s_i + s_{i-1}}\, \alpha_{i+1} v_{i-1}(z) + \omega_{i+1}(z) \cdot v_i(z)\,. \tag{2.11}$$

■

The power series pseudo-quotient $\omega_{i+1}(z)$ is calculated in $D[z]$ by solving

$$\begin{bmatrix} r_i^{(0)} & & \\ \vdots & \ddots & \\ r_i^{(s_i)} & \cdots & r_i^{(0)} \end{bmatrix} \begin{bmatrix} \omega_{i+1}^{(0)} \\ \vdots \\ \omega_{i+1}^{(s_i)} \end{bmatrix} = -\alpha_{i+1} \begin{bmatrix} r_{i-1}^{(0)} \\ \vdots \\ r_{i-1}^{(s_i)} \end{bmatrix}. \tag{2.12}$$

A PSRS can be determined by selecting $\alpha_{i+1} = c_i^{s_i+1}$ (where $c_i = r_i(0)$ for all $i \geq -1$) and $\beta_{i+1} = 1$ in (2.7). Unfortunately, this choice of α_{i+1} and β_{i+1} allows exponential growth in the coefficients of $u_i(z)$, $v_i(z)$, and $r_i(z)$, which is illustrated by

Example 2.3. Let

$$\begin{aligned} r_{-1}(z) = {} & 2 + 3z - 538z^2 + 8216z^3 - 71676z^4 + 656130z^5 - 8872897z^6 \\ & + 117238355z^7 - 1218912918z^8 + 12946871992z^9 - 158114090732z^{10} \\ & + 1865932164067z^{11} - 20694552958140z^{12} + 233361966173612z^{13} \\ & - 2715347521336995z^{14} + 31324801647722896z^{15} + \cdots \end{aligned} \tag{2.13}$$
$$r_0(z) = -1\,,$$

where D is the domain of integers. With $s_0 = 0$, (2.7) yields

$$r_1(z) = 3 - 538\,z + 8216\,z^2 - 71676\,z^3 + 656130\,z^4 + \cdots$$
$$r_2(z) = -264796 + 4205180\,z - 36593298\,z^2 + 326379249\,z^3 + \cdots$$
$$r_3(z) = 990981997992 - 14982621326820\,z + 114490402001640\,z^2 + \cdots$$
$$r_4(z) = -28977538459555156111803134017792 + 932391413638500308407811624473024\,z + \cdots$$
$$r_5(z) = 500431674770589227977002150959835625128210441979381344876671207276954198016 0 + \cdots$$

■

Cabay and Kossowski provide a definition of β_{i+1} that keeps the growth of the coefficients in $u_i(z)$, $v_i(z)$, and $r_i(z)$ linear with respect to the degree of $u_i(z)$. Specifically,

$$\alpha_{i+1} = c_i^{\,\epsilon_i+1}, \quad i \geq 0 \tag{2.14}$$

and

$$\beta_{i+1} = \begin{cases} (-1)^{\epsilon_0+1}, & i=0, \\ (-1)^{\epsilon_i+1}\, c_{i-1}\, h_{i-1}^{\,\epsilon_i}, & i>0, \end{cases} \tag{2.15}$$

where

$$h_i = \begin{cases} c_0^{\,\epsilon_0}, & i=0, \\ c_i^{\,\epsilon_i}\, h_{i-1}^{\,1-\epsilon_i}, & i>0. \end{cases} \tag{2.16}$$

Repeating the calculations in Example 2.3 with β_{i+1} given now by (2.15), the decrease in growth is demonstrated by

Example 2.4.

$$\begin{aligned} r_{-1}(z) = 2 + 3\,z - 538\,z^2 + 8216\,z^3 - 71676\,z^4 + 656130\,z^5 - 8872897\,z^6 \\ + 117238355\,z^7 - 1218912918\,z^8 + 12946871992\,z^9 - 158114090732\,z^{10} \\ + 1865932164067\,z^{11} - 20694552958140\,z^{12} + 233361966173612\,z^{13} \\ - 2715347521336995\,z^{14} + 31324801647722896\,z^{15} + \cdots \end{aligned}$$
$$r_0(z) = -1$$
$$r_1(z) = 3 - 538\,z + 8216\,z^2 - 71676\,z^3 + 656130\,z^4 - 8872897\,z^5 + \cdots$$
$$r_2(z) = 264796 - 4205180\,z + 36593298\,z^2 - 326379249\,z^3 + 4421903521\,z^4 + \cdots$$

$$r_3(z) = 110109110888 - 1664735702980\, z + 12721155777960\, z^2 - 86545107500124\, z^3 + \ldots$$

$$r_4(z) = 510215480292756252 - 1641687176412127664\, z + 32136890636472495328\, z^2 + \ldots$$

$$r_5(z) = 14218050427141126608670526 85 - 302060394595010888129247651 9\, z + \ldots .$$

The removed factors β_{i+1} are

$$\beta_1 = -1$$
$$\beta_2 = -1$$
$$\beta_3 = 9$$
$$\beta_4 = 70116921616$$
$$\beta_5 = 12124016300545889148544 .$$

■

Let m_i and n_i be defined as follows

$$m_{-1} = -1, \qquad n_{-1} = -s_0 - 1, \tag{2.17}$$

and

$$m_{i+1} = m_i + s_i, \qquad n_{i+1} = n_i + s_i, \; i = -1, 0, \ldots . \tag{2.18}$$

Then we have the following theorem due to Cabay and Kossowski.

Theorem 2.5. For $i \geq 1$ the PSRS (2.6) and the corresponding cofactor sequence (2.8) satisfy

(1) $v_i(0) \neq 0$,

(2) $a(z) \cdot v_i(z) + b(z) \cdot u_i(z) = z^{m_i + n_i + s_i}\, r_i(z)$, (2.19)

(3) $min\,\{m_i - deg(u_i(z))\,,\, n_i - deg(v_i(z))\} = 0$, and

(4) $GCD\,\{u_i(z), v_i(z)\} = h$, for some $h \in \mathbf{D}$.

■

Thus, $u_i(z) / v_i(z)$ is a Padé fraction of type (m_i, n_i) for the pair $\big(a(z), b(z)\big)$, where

Definition 2.6. The expression $u(z)/v(z) \in \mathbf{D}(z)$, is a Padé fraction of type (m,n) for the pair $\left(a(z), b(z)\right)$ if

(1) $v(0) \neq 0$,

(2) $a(z) \cdot v(z) + b(z) \cdot u(z) = O(z^{m+n+1})$ †,

(3) $deg(u(z)) \leq m$, $deg(v(z)) \leq n$, and

(4) $GCD\,\{u(z), v(z)\} = h$, for some $h \in \mathbf{D}$.

2.2. Resultants for Power Series

One fundamental result on Padé fractions is their relationship to resultants (determinants) (c.f. Gragg [8]). In this section, we highlight the work of Cabay and Kossowski [5] which relates resultants to power series remainder sequences. Thus, the iterations (2.7), (2.10), and (2.11) for constructing a PSRS and the corresponding cofactor sequence provides an algorithm for computing resultants as well as Padé fractions.

Definition 2.7. Given formal power series $a(z)$ and $b(z)$ defined by (2.1) and (2.2) and non-negative integers j and k, define the following resultants ‡

† The (algebraic) O-symbol indicates that the right side is a power series beginning with the power $z^{m+n+\kappa+1}$, $0 \leq \kappa \leq \infty$, $\kappa = +\infty$ means that $a(z) \cdot v(z) + b(z) \cdot u(z) = 0$.

‡ $|M|$ is the determinant of the matrix M.

$$\hat{U}_{j,k}(a,b,z) = \begin{vmatrix} a^{(0)} & & b^{(0)} & \\ \vdots & \ddots & \vdots & \ddots \\ a^{(k)} & a^{(0)} & b^{(j)} & b^{(0)} \\ \vdots & \vdots & \vdots & \vdots \\ a^{(j+k)} & \cdots\ a^{(j)} & b^{(j+k)} & \cdots\ b^{(k)} \\ 0 & \cdots\ 0 & 1 & \cdots\ z^{j} \end{vmatrix}, \tag{2.20}$$

$$\hat{V}_{j,k}(a;b,z) = \begin{vmatrix} a^{(0)} & & b^{(0)} & \\ \vdots & \ddots & \vdots & \ddots \\ a^{(k)} & a^{(0)} & b^{(j)} & b^{(0)} \\ \vdots & \vdots & \vdots & \vdots \\ a^{(j+k)} & \cdots\ a^{(j)} & b^{(j+k)} & \cdots\ b^{(k)} \\ 1 & \cdots\ z^{k} & 0 & \cdots\ 0 \end{vmatrix}, \tag{2.21}$$

and

$$\hat{R}_{j,k}(a,b,z) = \begin{vmatrix} a^{(0)} & & b^{(0)} & \\ \vdots & \ddots & \vdots & \ddots \\ a^{(k)} & a^{(0)} & b^{(j)} & b^{(0)} \\ \vdots & \vdots & \vdots & \vdots \\ a^{(j+k)} & \cdots\ a^{(j)} & b^{(j+k)} & \cdots\ b^{(k)} \\ a(z) & \cdots\ z^{k}\, a(z) & b(z) & \cdots\ z^{j}\, b(z) \end{vmatrix}. \tag{2.22}$$

Clearly, $\hat{U}_{j,k}(a,b,z)$, $\hat{V}_{j,k}(a,b,z) \in \mathbf{D}[z]$ and $\hat{R}_{j,k}(a,b,z) \in \mathbf{D}[[z]]$. One relationship (c.f. Gragg [8]) on resultants is

$$a(z) \cdot \hat{V}_{j,k}(a,b,z) + b(z) \cdot \hat{U}_{j,k}(a,b,z) = \hat{R}_{j,k}(a,b,z)$$

and

$$\hat{R}_{j,k}(a,b,z) = z^{j+k}\sum_{l=1}^{\infty} z^{l} \begin{vmatrix} a^{(0)} & & & b^{(0)} & & \\ \vdots & \ddots & a^{(0)} & \vdots & \ddots & b^{(0)} \\ \vdots & & \vdots & \vdots & & \vdots \\ a^{(j+k)} & \cdots & a^{(j)} & b^{(j+k)} & \cdots & b^{(k)} \\ a^{(j+k+l)} & \cdots & a^{(j+l)} & b^{(j+k+l)} & \cdots & b^{(k+l)} \end{vmatrix} \qquad (2.23)$$

Cabay and Kossowski [5] show that for $j,k \geq 0$, the resultants $\hat{U}_{j,k}(a,b,z)$, $\hat{V}_{j,k}(a,b,z)$, and $\hat{R}_{j,k}(a,b,z)$ are either all identically zero, or with α_{i+1} and β_{i+1} defined by (2.14) and (2.15) they correspond, respectively, to $u_i(z)$, $v_i(z)$, and $r_i(z)$ for some $i \geq 0$. Formally, they give

Theorem 2.8. The PSRS (2.6) and the cofactor sequence (2.8) with α_{i+1} and β_{i+1} defined by (2.14) and (2.15) respectively, are related to the resultants defined by (2.20), (2.21), and (2.22), for $i > 0$, according to

$$\hat{R}_{m_i,n_i}(a,b,z) = z^{m_i+n_i+s_i} r_i(z)$$
$$\hat{R}_{m_i+l,n_i+l}(a,b,z) = 0, \qquad 0 < l < s_i - 1$$
$$c_i \hat{R}_{m_{i+1}-1,n_{i+1}-1}(a,b,z) = z^{m_{i+1}+n_{i+1}-1} h_i r_i(z)$$

$$\hat{U}_{m_i,n_i}(a,b,z) = u_i(z)$$
$$\hat{U}_{m_i+l,n_i+l}(a,b,z) = 0, \qquad 0 < l < s_i - 1$$
$$c_i \hat{U}_{m_{i+1}-1,n_{i+1}-1}(a,b,z) = z^{s_i-1} h_i u_i(z)$$

$$\hat{V}_{m_i,n_i}(a,b,z) = v_i(z)$$
$$\hat{V}_{m_i+l,n_i+l}(a,b,z) = 0, \qquad 0 < l < s_i - 1$$
$$c_i \hat{V}_{m_{i+1}-1,n_{i+1}-1}(a,b,z) = z^{s_i-1} h_i v_i(z)$$

Proof: See Cabay and Kossowski [5]. ∎

As a consequence of Theorem 2.8, the coefficients of $u_i(z)$, $v_i(z)$, and $r_i(z)$ grow linearly with respect to m_i and n_i.

Making use of these results, Cabay and Kossowski present their algorithm JPADE. In the next section we present a constraint on the power series which provides for additional factors in the intermediate Padé fractions to become known.

2.3. Resultants for Rational Functions

We wish to consider the rational expression $p(z) / q(z) \in D(z)$, where

$$p(z) = \sum_{j=0}^{m} p^{(j)} z^j \qquad (2.24)$$

and

$$q(z) = \sum_{j=0}^{n} q^{(j)} z^j \ , \ q^{(0)} \neq 0 \ . \qquad (2.25)$$

For notational convenience, let

$$d = q^{(0)} \qquad (2.26)$$

and define

$$a(z) = \sum_{i=0}^{\infty} a^{(i)} z^i \in D[[z]] \ , \qquad (2.27)$$

where

$$a(z) = \frac{\sum_{j=0}^{m} p^{(j)} d^j z^j}{1 + \sum_{j=1}^{n} q^{(j)} d^{j-1} z^j} \ . \qquad (2.28)$$

Thus,

$$a(z) = d \, \frac{p(dz)}{q(dz)} \ , \qquad (2.29)$$

and $a(z)$ may be regarded as the power series representation of $d \, p(z) / q(z)$ with $z = dz$.

In Chapter 3, we show that the solution to a linear system over $\mathbf{D}[z]$ can be characterized by a vector of power series satisfying (2.28). For the purposes of the remaining sections of this chapter, we require only that $a(z)$ in (2.27) and d in (2.26) be known, and it is assumed that $p(x)$ and $q(x)$ are not explicitly known.

Theorem 2.9 below relates the resultants of $a(z)$ to those of $p(x)$ and $q(x)$. Using these relationships, for the power series $a(z)$ given by (2.28) (again, it is assumed that $a(z)$, and not $p(x)$ and $q(x)$, is explicitly known), we will show in Section 2.4 that we are able to determine a PSRS for which additional factors are removed.

Theorem 2.9. Let $p(x)$, $q(x)$ and $a(z)$ be given by (2.24), (2.25) and (2.28), and define

$$b(z) = -1 \quad . \tag{2.30}$$

Then, for non-negative integers j and k and for $x = dz$:

$$\begin{aligned} q(x)\; R_{j,k}(a,b,z) &= (-1)^{j+1}\, d^{j(k-1)}\; R_{j,k}(p,q,x) \;, \\ C_{j,k}(a,b,z) &= (-1)^{j}\, d^{j(k-1)}\; C_{j,k}(p,q,x) \;, \end{aligned} \tag{2.31}$$

and

$$V_{j,k}(a,b,z) = (-1)^{j+1}\, d^{j(k-1)-1}\; V_{j,k}(p,q,x) \;.$$

Proof: From (2.29), it follows that

$$\begin{bmatrix} d & & & \\ q^{(1)}\, d & d & & \\ \vdots & & \ddots & \\ q^{(j+k)}\, d^{j+k} & \cdots & & d \end{bmatrix} \begin{bmatrix} a^{(0)} \\ \vdots \\ a^{(j+k)} \end{bmatrix} = d \begin{bmatrix} p^{(0)} \\ \vdots \\ p^{(j+k)}\, d^{j+k} \end{bmatrix} . \tag{2.32}$$

Let

$$\Xi = \begin{bmatrix} d & & & & \\ q^{(1)}d & d & & & \\ \vdots & & \ddots & & \\ q^{(j+k)}d^{j+k} & \cdots & \cdots & d & \\ 0 & \cdots & \cdots & \cdots & 1 \end{bmatrix} \tag{2.33}$$

Clearly, $|\Xi| = d^{j+k+1}$. Thus, from (2.20), (2.30), (2.32), and (2.33), it follows that

$$C'_{j,k}(a,b,z) = \frac{|\Xi|}{d^{j+k+1}} \cdot C'_{j,k}(a,b,z)$$

$$= \frac{1}{d^{j+k+1}} \begin{vmatrix} p^{(0)}d & & & -d & & \\ & \ddots & & & \ddots & \\ \vdots & & p^{(0)}d & \vdots & & -d \\ \vdots & & \vdots & \vdots & & \vdots \\ p^{(j+k)}d^{j+k+1} & \cdots & p^{(j)}d^{j+1} & -q^{(j+k)}d^{j+k} & \cdots & -q^{(k)}d^{k} \\ 0 & \cdots & 0 & 1 & \cdots & z^{j} \end{vmatrix}$$

$$= (-1)^{j} \begin{vmatrix} p^{(0)} & & & 1 & & \\ & \ddots & & & \ddots & \\ \vdots & & p^{(0)} & \vdots & & 1 \\ \vdots & & \vdots & \vdots & & \vdots \\ p^{(j+k)}d^{j+k} & \cdots & p^{(j)}d^{j} & q^{(j+k)}d^{j+k-1} & \cdots & q^{(k)}d^{k-1} \\ 0 & \cdots & 0 & 1 & \cdots & z^{j} \end{vmatrix}$$

$$= \frac{(-1)^{j}}{(d^{1}\cdots d^{k})(d^{1}\cdots d^{j+1})} \begin{vmatrix} p^{(0)} & & d & \\ \vdots & & \vdots & \\ p^{(k)}d^{k} & p^{(0)}d^{k} & q^{(j)}d^{j} & d^{j+1} \\ \vdots & \vdots & \vdots & \vdots \\ p^{(j+k)}d^{j+k} & \cdots \; p^{(j)}d^{j+k} & q^{(j+k)}d^{j+k} & \cdots \; q^{(k)}d^{j+k} \\ 0 & \cdots \; 0 & d & \cdots \; d^{j+1}z^{j} \end{vmatrix}$$

$$= \frac{(-1)^j (d^1 \cdots d^{j+k}) d}{(d^1 \cdots d^k)(d^1 \cdots d^{j+1})} \begin{vmatrix} p^{(0)} & & d & \\ \vdots & \ddots & \vdots & \ddots \\ p^{(k)} & p^{(0)} & q^{(j)} & d \\ \vdots & \vdots & \vdots & \vdots \\ p^{(j+k)} \; \cdots & p^{(j)} & q^{(j+k)} \; \cdots & q^{(k)} \\ 0 \; \cdots & 0 & 1 \; \cdots & d^j z^j \end{vmatrix}$$

$$= (-1)^j d^{j(k-1)} \dot{U}_{j,k}(p,q,dz)$$

$$= (-1)^j d^{j(k-1)} \dot{U}_{j,k}(p,q,z) \; .$$

Similarly, from (2.21), (2.30), (2.32), and (2.33)

$$\dot{V}_{j,k}(a,b,z) = \frac{|\Xi|}{d^{j+k+1}} \cdot V_{j,k}(a,b,z)$$

$$= \frac{(-1)^{j+1} (d^1 \cdots d^{j+k})}{(d^1 \cdots d^k)(d^1 \cdots d^{j+1})} \begin{vmatrix} p^{(0)} & & d & \\ \vdots & \ddots & \vdots & \ddots \\ p^{(k)} & p^{(0)} & q^{(j)} & d \\ \vdots & \vdots & \vdots & \vdots \\ p^{(j+k)} \; \cdots & p^{(j)} & q^{(j+k)} \; \cdots & q^{(k)} \\ 1 \; \cdots & d^k z^k & 0 \; \cdots & 0 \end{vmatrix}$$

$$= (-1)^{j+1} d^{j(k-1)-1} \dot{V}_{j,k}(p,q,dz)$$

$$= (-1)^{j+1} d^{j(k-1)-1} \dot{V}_{j,k}(p,q,z) \; . \qquad (2.34)$$

Finally, to prove (2.31), from (2.22) and (2.29), first observe that

$$d\,\hat{R}_{j,k}(p,q,dz) = d\begin{vmatrix} p^{(0)} & & & d & & \\ \vdots & \ddots & & \vdots & \ddots & \\ p^{(k)} & \cdots & p^{(0)} & q^{(j)} & & d \\ \vdots & & \vdots & \vdots & & \vdots \\ p^{(j+k)} & \cdots & p^{(j)} & q^{(j+k)} & \cdots & q^{(k)} \\ p(dz) & \cdots & d^k z^k p(dz) & q(dz) & \cdots & d^j z^j q(dz) \end{vmatrix}$$

$$= q(dz)\begin{vmatrix} p^{(0)} & & & d & & \\ \vdots & \ddots & & \vdots & \ddots & \\ p^{(k)} & \cdots & p^{(0)} & q^{(j)} & & d \\ \vdots & & \vdots & \vdots & & \vdots \\ p^{(j+k)} & \cdots & p^{(j)} & q^{(j+k)} & \cdots & q^{(k)} \\ a(z) & \cdots & d^k z^k a(z) & d & \cdots & d^{j+1} z^j \end{vmatrix} . \qquad (2.35)$$

Thus, from (2.30), (2.32), (2.33), and (2.35), it follows that

$$q(dz)\ \hat{R}^{-}_{j,k}(a,b,z) = \frac{q(dz)\,|\Xi|}{d^{j+k+1}}\cdot \hat{R}_{j,k}(a,b,z)$$

$$= \frac{(-1)^{j+1}(d^1\cdots d^{j+k})}{(d^1\cdots d^k)(d^1\cdots d^{j+1})}\, q(dz)\begin{vmatrix} p^{(0)} & & & d & & \\ \vdots & \ddots & & \vdots & \ddots & \\ p^{(k)} & \cdots & p^{(0)} & q^{(j)} & & d \\ \vdots & & \vdots & \vdots & & \vdots \\ p^{(j+k)} & \cdots & p^{(j)} & q^{(j+k)} & \cdots & q^{(k)} \\ a(z) & \cdots & d^k z^k a(z) & d & \cdots & d^{j+1} z^j \end{vmatrix}$$

$$= (-1)^{j+1} d^{j(k-1)}\ \hat{R}_{j,k}(p,q,dz)$$

$$= (-1)^{j+1} d^{j(k-1)}\ \hat{R}_{j,k}(p,q,z) .$$

Therefore, as with (2.23),

$$q(dz)\ \hat{R}_{j,k}(a,b,z)$$

$$= (-1)^{j+1} d^{j(k-1)} x^{j+k} \sum_{l=1}^{\infty} x^{l} \begin{vmatrix} p^{(0)} & & & d & & \\ & \ddots & & & \ddots & \\ & & p^{(0)} & & & d \\ \vdots & & \vdots & \vdots & & \vdots \\ p^{(j+k)} & \dots & p^{(j)} & q^{(j+k)} & \dots & q^{(k)} \\ p^{(j+k+l)} & \dots & p^{(j+l)} & q^{(j+k+l)} & \dots & q^{(k+l)} \end{vmatrix}$$

■

Corollary 2.10.--For the power series $a(z)$ and $b(z)$ defined by (2.28) and (2.30), respectively, the PSRS (2.6) and the cofactor sequence (2.8) with α_{i+1} and β_{i+1} defined by (2.14) and (2.15), respectively, satisfy, for $i>0$,

$$(-1)^{m_i+1}\ d^{m_i(n_i-1)}\ \hat{R}_{m_i,n_i}(p,q,x) = z^{m_i+n_i+s_i}\ q(x)\ r_i(z)\ ,$$

$$(-1)^{m_i}\ d^{m_i(n_i-1)}\ \hat{U}_{m_i,n_i}(p,q,x) = u_i(z)\ , \tag{2.36}$$

and

$$(-1)^{m_i+1}\ d^{m_i(n_i-1)-1}\ \hat{V}_{m_i,n_i}(p,q,x) = v_i(z)\ , \tag{2.37}$$

where $x = dz$.

Proof: The Corollary is the immediate consequence of Theorems 2.8 and 2.9.

■

It can be observed from (2.34) that $\hat{V}_{m_i,n_i}(p,q,x)$ is divisible by d since the expansion about the constant term is divisible by d. Thus, from (2.36) and (2.37) it follows that both $u_i(z)$ and $v_i(z)$ in the cofactor sequence (2.8) with α_{i+1} and β_{i+1} defined by (2.14) and (2.15), respectively, are divisible by $d^{m_i(n_i-1)}$. In addition, from (2.19), the factor $d^{m_i(n_i-1)}$ also divides $r_i(z)$. In the next section, by the appropriate selection of α_{i+1} and β_{i+1}, we determine a PSRS and a corresponding cofactor sequence where the factor $d^{m_i(n_i-1)}$ is systematically removed.

2.4. A Reduced Power Series Remainder Sequence

For the power series $a(z)$ and $b(z)$ given by (2.28) and (2.30), in this section the factors identified in Corollary 2.10 are systematically removed from the PSRS (2.6) and from the cofactor sequence (2.8). This is accomplished simply by redefining α_{i+1} and β_{i+1} in (2.14) and (2.15).

We denote a reduced PSRS for $\left(a(z), b(z)\right)$ with respect to s_0 by

$$\{s_i, \overline{r}_i(z)\}_{i=-1,0,} \tag{2.38}$$

and the corresponding reduced cofactor sequence by

$$\{\overline{u}_i(z), \overline{v}_i(z)\}_{i=-1,0,} \tag{2.39}$$

defined as follows. Initially, set

$$\overline{u}_{-1}(z) = 0, \qquad \overline{v}_{-1}(z) = z^{-s_0-1}, \qquad \overline{r}_{-1}(z) = a(z),$$

$$\overline{u}_0(z) = 1, \qquad \overline{v}_0(z) = 0, \qquad \overline{r}_0(z) = b(z) = -1\ ,$$

$$s_{-1} = 1\ .$$

Then, for $i=0,1,\ldots$, determine $\overline{u}_{i+1}(z)$, $\overline{v}_{i+1}(z)$, $\overline{\omega}_{i+1}(z)$, and $\overline{r}_{i+1}(z)$ from

$$\overline{\alpha}_{i+1}\ \overline{r}_{i-1}(z) + \overline{\omega}_{i+1}(z)\ \overline{r}_i(z) = z^{s_i+s_{i+1}}\overline{\beta}_{i+1}\ \overline{r}_{i+1}(z)\ , \tag{2.40}$$

$$\overline{\beta}_{i+1}\ \overline{u}_{i+1}(z) = z^{s_i+s_{i-1}}\ \overline{\alpha}_{i+1}\ \overline{u}_{i-1}(z) + \overline{\omega}_{i+1}(z)\ \overline{u}_i(z)\ , \tag{2.41}$$

and

$$\overline{\beta}_{i+1}\ \overline{v}_{i+1}(z) = z^{s_i+s_{i-1}}\ \overline{\alpha}_{i+1}\ \overline{v}_{i-1}(z) + \overline{\omega}_{i+1}(z)\ \overline{v}_i(z)\ . \tag{2.42}$$

Let

$$\overline{c}_i = \overline{r}_i(0)\ ,\ i=-1,0,\ldots\ . \tag{2.43}$$

Then, in (2.40), (2.41), and (2.42), set

$$\overline{\alpha}_{i+1} = \overline{c}_i^{\,s_i+1} \quad , i \geq 0 \tag{2.44}$$

and

$$\overline{\beta}_{i+1} = \begin{cases} (-1)^{s_0+1} d^{-s_0}, & i=0 \\ (-1)^{s_i+1} \overline{c}_{i-1} \, \overline{h}_{i-1}^{\,s_i}, & i>0 \end{cases} \tag{2.45}$$

where

$$\overline{h}_i = \begin{cases} \overline{c}_0^{\,s_0} d^{2s_0+s_1-1}, & i=0 \\ \overline{c}_i^{\,s_i} \overline{h}_{i-1}^{\,1-s_i} d^{s_i+s_{i+1}}, & i>0 \,. \end{cases} \tag{2.46}$$

In addition, in (2.40),

$$\overline{\omega}_{i+1}(z) = \sum_{j=0}^{s_i} \overline{\omega}_{i+1}^{(j)} z^j$$

is the power series pseudo-quotient on power series pseudo-division of $\overline{r}_{i-1}(z)$ by $\overline{r}_i(z)$ with respect to s_i; and, either $s_{i+1} = \infty$ (in which case, by convention, $\overline{r}_{i+1}(z) = 0$), or s_{i+1} is the smallest integer such that $\overline{r}_{i+1}(0) \neq 0$.

As in (2.17) and (2.18), define

$$m_{i+1} = m_i + s_i \,, \quad n_{i+1} = n_i + s_i \,, \quad i = -1, 0, \ldots \,, \tag{2.47}$$

where $m_{-1} = -1$ and $n_{-1} = -s_0 - 1$. Using (2.47), we derive the following identity which is used extensively below:

$$\begin{aligned} m_{i+1}(n_{i+1} - 1) &= (m_i + s_i)(n_i + s_i - 1) \\ &= m_i(n_i - 1) + (m_i + n_i + s_i - 1)s_i \,, \quad i \geq -1 \,. \end{aligned} \tag{2.48}$$

Theorem 2.11. The PSRS (2.6) and the cofactor sequence (2.8) with α_{i+1} and β_{i+1} given by (2.14) and (2.15) is related to the reduced PSRS (2.38) and the reduced cofactor sequence (2.39) with $\overline{\alpha}_{i+1}$ and $\overline{\beta}_{i+1}$ given by (2.44) and (2.45) according to

$$r_i(z) = \overline{r}_i(z)\, d^{m_i(n_i-1)} \tag{2.49}$$

$$u_i(z) = \overline{u}_i(z)\, d^{m_i(n_i-1)} \tag{2.50}$$

and

$$v_i(z) = \overline{v}_i(z)\, d^{m_i(n_i-1)} \tag{2.51}$$

for all $i \geq 0$.

Proof: We prove by induction on i that (2.50), (2.51), and (2.49) hold and, in addition, for $i \geq 1$

$$h_{i-1} = \overline{h}_{i-1}\, d^{m_i(n_i-1)-(m_i+n_i+s_i-1)} \tag{2.52}$$

and

$$\beta_{i+1} = \overline{\beta}_{i+1}\, d^{[m_i(n_i-1)-(m_i+n_i+s_i-1)]s_i + m_{i-1}(n_{i-1}-1)} . \tag{2.53}$$

Basis ($i=1$): From the initialization of the sequences involved, and since $m_0 = 0$,

$$\begin{aligned} &u_{-1}(z) = \overline{u}_{-1}(z), \qquad v_{-1}(z) = \overline{v}_{-1}(z), \qquad r_{-1}(z) = \overline{r}_{-1}(z), \\ &u_0(z) = \overline{u}_0(z)\, d^{m_0(n_0-1)},\ v_0(z) = \overline{v}_0(z)\, d^{m_0(n_0-1)},\ r_0(z) = \overline{r}_0(z)\, d^{m_0(n_0-1)} . \end{aligned} \tag{2.54}$$

Consequently, $c_0 = \overline{c}_0$, and together with (2.14) and (2.44),

$$\alpha_1 = \overline{\alpha}_1 . \tag{2.55}$$

From (2.15), (2.45), and (2.47),

$$\begin{aligned} \beta_1 &= (-1)^{s_0+1} \\ &= \overline{\beta}_1\, d^{s_0} \\ &= \overline{\beta}_1\, d^{-m_1(n_1-1)} . \end{aligned} \tag{2.56}$$

Thus, from (2.7), (2.40), (2.54), (2.55), and (2.56), we have that

$$\begin{aligned} z^{s_0+s_1}\beta_1\, r_1(z) &= \alpha_1\, r_{-1}(z) + \omega_1(z)\, r_0(z) \\ &= \overline{\alpha}_1\, \overline{r}_{-1}(z) + \overline{\omega}_1(z)\, \overline{r}_0(z) \end{aligned} \tag{2.57}$$

$$= z^{s_0+s_1} \overline{\beta}_1 \, \overline{r}_1(z)$$
$$= z^{s_0+s_1} \beta_1 \, \overline{r}_1(z) \, d^{m_1(n_1-1)} .$$

In (2.57), using the uniqueness of the pseudo-quotient in the power series pseudo-division (2.12) of $\overline{r}_{-1}(z)$ by $\overline{r}_0(z)$, we have inferred that

$$\omega_1(z) = \overline{\omega}_1(z) \tag{2.58}$$

and

$$r_1(z) = \overline{r}_1(z) \, d^{m_1(n_1-1)} . \tag{2.59}$$

From (2.10), (2.41), (2.54), (2.55), (2.58), and (2.56) we have

$$\begin{aligned} \beta_1 \, u_1(z) &= z^{s_0+s-1} \alpha_1 \, u_{-1}(z) + \omega_1(z) \, u_0(z) \\ &= z^{s_0+s-1} \overline{\alpha}_1 \, \overline{u}_{-1}(z) + \overline{\omega}_1(z) \, \overline{u}_0(z) \\ &= \overline{\beta}_1 \, \overline{u}_1(z) \\ &= \beta_1 \, \overline{u}_1(z) \, d^{m_1(n_1-1)} . \end{aligned}$$

Therefore,

$$u_1(z) = \overline{u}_1(z) \, d^{m_1(n_1-1)} . \tag{2.60}$$

Similarly, with (2.11), (2.42), (2.54), (2.55), (2.58), and (2.56) we have

$$v_1(z) = \overline{v}_1(z) \, d^{m_1(n_1-1)} . \tag{2.61}$$

By (2.16), (2.54), (2.46), and (2.47)

$$\begin{aligned} h_0 &= c_0^{s_0} \\ &= \overline{c}_0^{s_0} \, d^{m_0(n_0-1)s_0} \\ &= \overline{h}_0 \, d^{-(2s_0+s_1-1)} \\ &= \overline{h}_0 \, d^{m_1(n_1-1)-(m_1+n_1+s_1-1)} . \end{aligned} \tag{2.62}$$

Finally, from (2.15), (2.54), (2.62), and (2.45)

$$\beta_2 = \overline{\beta}_2 \, d^{\left[m_1(n_1-1)-(m_1+n_1+s_1-1)\right]s_1+m_0(n_0-1)} . \tag{2.63}$$

Therefore, from (2.59), (2.60), (2.61), (2.62), and (2.63), it follows that (2.49), (2.50), (2,51), (2.52), and (2.53) are valid for $i=1$.

Inductive Step ($i \geq 1$): Assume for some arbitrary integer $i \geq 1$ that (2.49), (2.50), (2,51), (2.52), and (2.53) are valid for all j where $1 \leq j \leq i$. We now show that they are also valid at $i+1$.

From (2.43) and (2.49), $c_i = \bar{c}_i \, d^{m_i(n_i-1)}$, and therefore, using (2.14) and (2.44),

$$\alpha_{i+1} = \bar{\alpha}_{i+1} \, d^{m_i(n_i-1)(s_i+1)} \quad . \tag{2.64}$$

Now, from (2.7), (2.49), (2.64), (2.40), (2.53), and (2.48)

$$\begin{aligned} z^{s_i+s_{i-1}} \beta_{i+1} r_{i+1}(z) &= \alpha_{i+1} \, r_{i-1}(z) + \omega_{i+1}(z) \, r_i(z) \\ &= \bar{\alpha}_{i+1} \, \bar{r}_{i-1}(z) d^{m_i(n_i-1)(s_i+1)+m_{i-1}(n_{i-1}-1)} + \omega_{i+1}(z) \, \bar{r}_i(z) \, d^{m_i(n_i-1)} \\ &= \left[\bar{\alpha}_{i+1} \, \bar{r}_{i-1}(z) + \bar{\omega}_{i+1}(z) \, \bar{r}_i(z) \right] d^{m_i(n_i-1)(s_i+1)+m_{i-1}(n_{i-1}-1)} \\ &= z^{s_i+s_{i-1}} \, \bar{\beta}_{i+1} \, \bar{r}_{i+1}(z) \, d^{m_i(n_i-1)(s_i+1)+m_{i-1}(n_{i-1}-1)} \\ &= z^{s_i+s_{i-1}} \beta_{i+1} \, \bar{r}_{i+1}(z) \, d^{m_i(n_i-1)+(m_i+n_i+s_i-1)s_i} \\ &= z^{s_i+s_{i-1}} \beta_{i+1} \, \bar{r}_{i+1}(z) \, d^{m_{i+1}(n_{i+1}-1)} \quad . \end{aligned} \tag{2.65}$$

In (2.65), using the uniqueness of the pseudo-quotient in the power series pseudo-division (2.12) of $\bar{r}_{i-1}(z)$ by $\bar{r}_i(z)$, we have inferred that

$$\omega_{i+1}(z) = \bar{\omega}_{i+1}(z) \, d^{m_i(n_i-1)s_i+m_{i-1}(n_{i-1}-1)} \tag{2.66}$$

and

$$r_{i+1}(z) = \bar{r}_{i+1}(z) \, d^{m_{i+1}(n_{i+1}-1)} \, .$$

From (2.10), (2.64), (2.50), (2.66), (2.41), (2.53), and (2.48)

$$\begin{aligned} \beta_{i+1} \, u_{i+1}(z) &= z^{s_i+s_{i-1}} \alpha_{i+1} \, u_{i-1}(z) + \omega_{i+1}(z) \, u_i(z) \\ &= \left[z^{s_i+s_{i-1}} \bar{\alpha}_{i+1} \, \bar{u}_{i-1}(z) + \bar{\omega}_{i+1}(z) \, \bar{u}_i(z) \right] d^{m_i(n_i-1)(s_i+1)+m_{i-1}(n_{i-1}-1)} \\ &= \bar{\beta}_{i+1} \, \bar{u}_{i+1}(z) \, d^{m_i(n_i-1)(s_i+1)+m_{i-1}(n_{i-1}-1)} \end{aligned}$$

$$= \beta_{i+1}\ \overline{u}_{i+1}(z)\ d^{m_i(n_i-1)+(m_i+n_i+s_i-1)s_i}$$
$$= \beta_{i+1}\ \overline{u}_{i+1}(z)\ d^{m_{i+1}(n_{i+1}-1)} .$$

),

Therefore,

$$u_{i+1}(z) = \overline{u}_{i+1}(z)\ d^{m_{i+1}(n_{i+1}-1)} .$$

Similarly, with (2.42), (2.51), (2.66), (2.53), and (2.48)

$$v_{i+1}(z) = \overline{v}_{i+1}(z)\ d^{m_{i+1}(n_{i+1}-1)} .$$

From (2.16), (2.52), (2.43), (2.49), (2.48), (2.46), and (2.47)

$$\begin{aligned} h_i &= c_i^{s_i}\ h_{i-1}^{1-s_i} \\ &= \overline{c}_i^{s_i}\ \overline{h}_{i-1}^{1-s_i}\ d^{m_i(n_i-1)-(m_i+n_i+s_i-1)(1-s_i)} \\ &= \overline{c}_i^{s_i}\ \overline{h}_{i-1}^{1-s_i}\ d^{m_{i+1}(n_{i+1}-1)-(m_i+n_i+s_i-1)} \\ &= \overline{h}_i\ d^{m_{i+1}(n_{i+1}-1)-(m_{i+1}+n_{i+1}+s_{i+1}-1)} . \end{aligned} \tag{2.67}$$

Finally, from (2.15), (2.43), (2.49), (2.67), and (2.45) we have that

$$\begin{aligned} \beta_{i+2} &= (-1)^{s_{i+1}+1}\ c_i\ h_i^{s_{i+1}} \\ &= (-1)^{s_{i+1}+1}\ \overline{c}_i\ \overline{h}_i^{s_{i+1}}\ d^{\left[m_{i+1}(n_{i+1}-1)-(m_{i+1}+n_{i+1}+s_{i+1}-1)\right]s_{i+1}+m_i(n_i-1)} \\ &= \overline{\beta}_{i+2}\ d^{\left[m_{i+1}(n_{i+1}-1)-(m_{i+1}+n_{i+1}+s_{i+1}-1)\right]s_{i+1}+m_i(n_i-1)} . \end{aligned}$$

But we have now shown that (2.49), (2.50), (2.51), (2.52), and (2.53) are valid for $i+1$. Therefore, by induction on i, they are valid for all $i \geq 1$. Hence, with (2.54), the theorem follows. ■

Corollary 2.12. For all $i>0$, using $x=dz$, we have

$$(-1)^{m_i+1}\, \hat{R}_{m_i,n_i}(p,q,x) = z^{m_i+n_i+s_i}\, q(x)\, \overline{r}_i(z) \tag{2.68}$$

$$\hat{R}_{m_i+l,n_i+l}(p,q,x) = 0 \qquad 0<l<s_i-1$$

$$(-1)^{m_{i+1}}\, d^{s_{i+1}+1}\, \overline{c}_i\, \hat{R}_{m_{i+1}-1,n_{i+1}-1}(p,q,x) = z^{m_{i+1}+n_{i+1}-1} q(x)\, \overline{h}_i\, \overline{r}_i(z) \tag{2.69}$$

$$(-1)^{m_i}\, U_{m_i,n_i}(p,q,x) = \overline{u}_i(z) \tag{2.70}$$

$$U_{m_i+l,n_i+l}(p,q,x) = 0 \qquad 0<l<s_i-1$$

$$(-1)^{m_{i+1}-1}\, d^{s_{i+1}+1}\, \overline{c}_i\, U_{m_{i+1}-1,n_{i+1}-1}(p,q,x) = z^{s_i-1}\, \overline{h}_i\, \overline{u}_i(z)$$

$$(-1)^{m_i+1}\, V_{m_i,n_i}(p,q,x) = d\, \overline{v}_i(z) \tag{2.71}$$

$$V_{m_i+l,n_i+l}(p,q,x) = 0 \qquad 0<l<s_i-1$$

$$(-1)^{m_{i+1}}\, d^{s_{i+1}}\, \overline{c}_i\, V_{m_{i+1}-1,n_{i+1}-1}(p,q,x) = z^{s_i-1}\, \overline{h}_i\, \overline{v}_i(z)\ .$$

Proof: The results follow directly from Theorem 2.8, Theorem 2.9, Corollary 2.10, and Theorem 2.11.

■

Corollary 2.13. $\overline{u}_i(z)$, $\overline{v}_i(z) \in \mathbf{D}[z]$, $\overline{r}_i(z) \in \mathbf{D}[[z]]$, $\overline{h}_i \in \mathbf{D}$ for all $i \geq 0$, and $\overline{\beta}_i \in \mathbf{D}$ for all $i>1$, where $x=dz$.

Proof: From (2.54) and (2.70) it is clear that $\overline{u}_i(z) \in \mathbf{D}[z]$ for all $i \geq 0$. Next, observe that $V_{m_i,n_i}(p,q,x)$ in (2.34) is divisible by d in the integral domain **D** since it's zero term coefficient has a factor of d, and all higher term coefficients do, too. Hence, from (2.54) and (2.71), $\overline{v}_i(z) \in \mathbf{D}[z]$ for all $i \geq 0$. In addition, from (2.19) and Theorem 2.11, it now follows that $\overline{r}_i(z) \in \mathbf{D}[[z]]$ for all $i \geq 0$.

Furthermore, from (2.43) and (2.69) we note that $\overline{h}_i$ is the leading non-zero coefficient of $(-1)^{m_i+1} d^{s_i+1}\, \hat{R}_{m_{i+1}-1,n_{i+1}-1}(p,q,x)$. Consequently, $\overline{h}_i \in \mathbf{D}$ for all $i \geq 0$.

Finally, by (2.45), using the knowledge that $\overline{h}_i \in \mathbf{D}$ for all $i \geq 0$ and that $\overline{c}_i = \overline{r}_i(0) \in \mathbf{D}$, it follows that $\overline{\beta}_{i+1} \in \mathbf{D}$ for $i > 0$. In addition, if $s_0 = 0$, then $\overline{\beta}_1 \in \mathbf{D}$; otherwise, $\overline{\beta}_1$ is not in the integral domain $\mathbf{D}$.

■

2.5. Algorithm $\overline{J}pade$

The algorithm $\overline{J}pade$ presented below is a specialization of the algorithm JPADE given by Cabay and Kossowski [5]. The specialization consists of making use of an additional factor in β_{i+1} which has been shown to exist in Theorem 2.11. The factor arises from the special nature of the power series pair $\left(a(z), b(z)\right)$ used for computation. Given integers $m \geq n \geq 0$, the existence of $p(z)$ and $q(z)$ given by (2.24) and (2.25), $d = q^{(0)}$ from (2.26), $a(z) = \dfrac{\sum_{j=0}^{m} p^{(j)} d^j z^j}{1 + \sum_{j=1}^{n} q^{(j)} d^j z^j}$ as specified by (2.28), and $b(z) = -1$ from (2.30), the algorithm computes all the Padé fractions $\overline{u}_i(z) / \overline{v}_i(z)$ of type (m_i, n_i) for the pair $\left(a(z), -1\right)$ along the off-diagonal path $m_i - n_i = m - n$, $i = 1, \ldots$, where $m_i \leq m$ and $n_i \leq n$ for all $i \geq -1$.

In the general situation, only the relationship (2.28), $a(z)$, and d are known. Consequently we are not aware of the actual values of m and n for $p(z)$ and $q(z)$. Later, in Section 2.7, we show that additional results may be inferred when m and n are known.

Algorithm $\overline{Jpade}\left(a(z), m, n, d\right)$:

Input: $a(z)$, m, n, d, where

(1) m and n are non-negative integers such that $m \geq n$, and

(2) the truncated power series

$$a(z) = \sum_{j=0}^{m+n} a^{(j)} z^j$$

with coefficients $a^{(j)} \in \mathbf{D}$, an integral domain.

Output: $\left(\begin{bmatrix} \overline{u}_i(z) \\ \overline{v}_i(z) \end{bmatrix}, \begin{bmatrix} m_i \\ n_i \end{bmatrix}\right)$, where

$\overline{u}_i(z) / \overline{v}_i(z) \in \mathbf{D}(z)$ is the largest Padé fraction of type (m_i, n_i) for the pair $\left(a(z), -1\right)$.

Step 1: **$** Initialization, c.f. (2.54) **$**

$ $\overline{c}_{-1}$ is needed in Step 8, $(i=0)$, note $\overline{c}_{-1} \leftarrow \overline{r}_{-1}$ **$**

$ $\overline{h}_{-1}$ is needed in Step 7, $(i=1)$, to satisfy (2.46) **$**

$$\overline{c}_{-1} \leftarrow 1$$
$$s_{-1} \leftarrow 1$$
$$\overline{h}_{-1} \leftarrow \frac{1}{d}$$
$$i \leftarrow 0$$
$$\begin{bmatrix} \overline{u}_0(z) & \overline{u}_{-1}(z) \\ \overline{v}_0(z) & \overline{v}_{-1}(z) \end{bmatrix} \leftarrow \begin{bmatrix} 1 & 0 \\ 0 & z^{n-m-1} \end{bmatrix}$$
$$\begin{bmatrix} m_0 & m_{-1} \\ n_0 & n_{-1} \end{bmatrix} \leftarrow \begin{bmatrix} 0 & -1 \\ n-m & n-m-1 \end{bmatrix}$$

Step 2: \$ Compute the residual for $\overline{u}_i(z) / \overline{v}_i(z)$ c.f. (2.40) \$

Find s_i and $\overline{r}_i(z)$, $deg(\overline{r}_i(z)) \le s_i$ such that

$$(a(z)\ \overline{v}_i(z) + (-1)\ \overline{u}_i(z)) \bmod z^{m_i+n_i+2s_i+1} = z^{m_i+n_i+s_i}\ \overline{r}_i(z)\ ,$$

where $s_i \le n - n_i$ and $\overline{r}_i(0) \ne 0$, otherwise set $s_i = n - n_i + 1$.

Set $\overline{c}_i = \overline{r}_i(0)$.

Step 3: \$ Calculation of degrees of next Padé fraction \$

$$m_{i+1} = m_i + s_i$$
$$n_{i+1} = n_i + s_i\ .$$

Step 4: \$ Termination \$

If $n_{i+1} > n$ then

$$\text{Return}\left(\begin{bmatrix} \overline{u}_i(z) \\ \overline{v}_i(z) \end{bmatrix}, \begin{bmatrix} m_i \\ n_i \end{bmatrix} \right).$$

Step 5: \$ Compute residual for $\overline{u}_{i-1}(z)/\overline{v}_{i-1}(z)$, c.f. (2.40) \$

Compute $\overline{r}_{i-1}(z)$, $deg(\overline{r}_{i-1}(z)) \le s_i$, such that

$$(a(z)\ \overline{v}_{i-1}(z) + (-1)\ \overline{u}_{i-1}(z)) \bmod z^{m_{i-1}+n_{i-1}+s_{i-1}+s_i+1} = z^{m_{i-1}+n_{i-1}+s_{i-1}}\ \overline{r}_{i-1}(z)\ ,$$

where $\overline{r}_{i-1}(0) \ne 0$.

\$ Note, for $i=0$, $(a(z)\ z^{n-m-1})$ mod $z^0 = z^{n-m-1}\ \overline{r}_{-1}(z)$, therefore $a(z) = \overline{r}_{-1}(z)$ \$

Step 6: \$ Pseudo-division of $\overline{r}_{-1}(z)$ by $\overline{r}_i(z)$ relative to s_i, c.f. (2.12) \$

Compute the pseudo-quotient $\overline{\omega}_{i+1}(z)$, $deg(\overline{\omega}_{i+1}(z)) \le s_i$, such that

$$(\overline{c}_i^{\,e_i+1}\ \overline{r}_{i-1}(z) + \overline{\omega}_{i+1}(z)\ \overline{r}_i(z))\ mod\ z^{e_i+1} = 0\ .$$

\$ Only the first s_i terms of $\overline{r}_{-1}(z)$ and $\overline{r}_0(z)$ are computed or needed in Step 6 \$

Step 7: \$ Begin computing a common factor, c.f. (2.45) \$

If $i > 0$ then

$$\overline{h}_{i-1} \leftarrow \overline{c}_{i-1}^{\,e_{i-1}}\ \overline{h}_{i-2}^{\,1-e_{i-1}}\ d^{e_{i-1}+e_i}\ .$$

Step 8: \$ Finish computing a common factor, c.f. (2.46) \$

$$\overline{\beta}_{i+1} \leftarrow (-1)^{e_i+1}\ \overline{c}_{i-1}\ \overline{h}_{i-1}^{\,e_i}\ .$$

Step 9: \$ Advance the computation, c.f. (2.41) and (2.42) \$

Compute the Padé fraction $\overline{u}_{i+1}(z)\ /\ \overline{v}_{i+1}(z)$ of type (m_{i+1}, n_{i+1}) by means of

$$\begin{bmatrix}\overline{u}_{i+1}(z)\\ \overline{v}_{i+1}(z)\end{bmatrix} \leftarrow \left(\begin{bmatrix}\overline{u}_i(z) & \overline{u}_{i-1}(z)\\ \overline{v}_i(z) & \overline{v}_{i-1}(z)\end{bmatrix}\begin{bmatrix}\overline{\omega}_{i+1}(z)\\ \overline{c}_i^{\,e_i+1} z^{e_i+e_{i-1}}\end{bmatrix}\right)\ /\ \overline{\beta}_{i+1}\ .$$

Set $i \leftarrow i+1$ and go to Step 2.

■

Although the initialization is not within the integral domain $\mathbf{D}[z]$, it is algorithmically correct, as can be seen by examining the $\overline{u}_i(z)$, $\overline{v}_i(z)$ and $\overline{r}_i(z)$ computed for each $i \geq 0$.

We consider again Example 2.3, and the reduced PSRS which results from using $\overline{Jpade}$. We note that $\overline{r}_{-1}(z)$ is given by (2.13), from Example 2.3, with $d = 7$ and $x = dz$, satisfies

$$\overline{r}_{-1}(z) = \frac{p(dz)}{q(dz)}$$

where

$$p(x) = 2 + 3\,x - 9\,x^2 + 15\,x^3 - 5\,x^4 - 7\,x^5 + 11\,x^6 + 3\,x^7$$

and

$$q(x) = 1 + 9\,x + 5\,x^2 + 17\,x^3 + 3\,x^4 + 6\,x^5 + 13\,x^6 + 2\,x^7 \ .$$

The savings are clearly demonstrated, by using, as in in Example 2.3, $r_0(z) = -1$, $s_0 = 0$ and $d = 7$.

Example 2.14

$$r_1(z) = 3 - 538\,z + 8216\,z^2 - 71676\,z^3 + 656130\,z^4 - 8872897\,z^5 + 117238355\,z^6 + \cdots$$
$$r_2(z) = 264796 - 4205180\,z + 36593298\,z^2 - 326379249\,z^3 + 4421903521\,z^4 + \cdots$$
$$r_3(z) = 2247124712 - 33974198020\,z + 259615424040\,z^2 - 1766226683676\,z^3 + \cdots$$
$$r_4(z) = 4336760025948 - 139541107566756\,z + 273159063285472\,z^2 + \cdots$$
$$r_5(z) = 102722024481320685 - 218231433398179719\,z + 593626399451390592\,z^2 + \cdots \ .$$

The removed factors β_{i+1} are

$$\beta_1 = -1$$
$$\beta_2 = -1$$
$$\beta_3 = 441$$
$$\beta_4 = 3435729159184$$
$$\beta_5 = 247428904092773064256 \ .$$

■

2.6. Cost Analysis of $\overline{J}pade$

In assessing the costs of $\overline{J}pade$, it is assumed that classical algorithms (c.f. Knuth [12]) are used for multiplication and exact division in the integral domain **D**. Thus, as in Brown [2], for $a, b \in \mathbf{D}$, we assume that the function **C** is the cost of a multiplication or division operation

$$\mathbf{C}(a \cdot b) = \mathbf{S}(a) \cdot \mathbf{S}(b) \ ,$$

where the function **S** measures the total storage space that is required for its argument. For example, if storage is measured in bits, then cost refers to the number of bit operations, whereas if storage is measured in single precision words, then cost is a measure of the number of single precision operations. In the following, we assume that the number of multiplications dominates the number of additions, and consequently, we do not count additions. Finally, we also assume that

$$S(a \cdot b) = S(a) + S(b) \tag{2.72}$$

and

$$S(a + b) = max\{S(a), S(b)\} \ . \tag{2.73}$$

These assumptions are valid for some integral domains, such as the univariate polynomials over a field, but not for others. Assumption (2.73) is an approximate bound for the integral domain of the integers which might, in the worst case, grow in size by one additional unit (in this case we might replace equality in (2.73) with approximate equality). However, to simplify the cost analysis, we use (2.73) as given. When **a** is a polynomial, or vector (or matrix) of polynomials, or even a power series, $S(a)$ is the size of the largest coefficient from D found in **a**.

To determine the cost of *Jpade*, we begin with estimates of the sizes of all the variables in the algorithm at the ith pass. For our purposes, suppose that the coefficients of $p(x)$, $q(x)$, and d given by (2.24), (2.25), and (2.26), are bounded by some integer $\kappa > 0$, where $x = dz$.

To bound the size of $a(z)$, from (2.28), it follows that

$$a^{(i)} = d^i p^{(i)} - \sum_{j=1}^{i} d^{j-1} q^{(j)} a^{(i-j)} \ , \ i = 0,1,\ldots \ . \tag{2.74}$$

Thus, $S(a^{(0)}) \leq \kappa$, and using (2.74), (2.72), and (2.73), it follows by induction on i that

$$S(a^{(i)}) \leq (i+1)\kappa \ , \ i = 0,1,\ldots \ . \tag{2.75}$$

To bound the size of $\overline{u}_i(z)$, from (2.20) and (2.70), we first observe that $\overline{u}_i(z)$ has the form

$$\overline{u}_i(z) = \sum_{j=0}^{m_i} \overline{u}_i^{(j)} (dz)^j ,$$

where $S(\overline{u}_i^{(j)}) \leq (m_i + n_i + 1)\kappa$. Thus,

$$S(\overline{u}_i(z)) \leq (2m_i + n_i + 1)\kappa .$$

Similarly, using (2.21) and (2.71), it follows that

$$S(\overline{v}_i(z)) \leq (m_i + 2n_i)\kappa .$$

To bound the size of $\overline{r}_i(z)$, using (2.68), (2.26), and (2.35), we see that the size of the lth coefficient is bounded by

$$S(a^{(m_i + n_i + s_i + l)}) + (m_i + n_i + 1)\kappa - \kappa .$$

To see this we note that the lth coefficient of $\overline{r}_i(z)$ is actually the $(m_i + n_i + s_i + l)$-th coefficient of the r.h.s. of (2.35) divided by d. Thus the bound on the size is found by examining the size of that coefficient in (2.35). But, the size of the coefficients of $a(z)$ are bounded by (2.75). Hence, the first $s_i + 1$ coefficients of $\overline{r}_i(z)$, namely, those required in Step 2 of *Jpade*, are bounded by

$$\begin{aligned} S(\overline{r}_i(z)) &\leq [(m_i + n_i + 2s_i + 1) + (m_i + n_i)]\kappa \\ &\leq 2(m_i + n_i + s_i + 1)\kappa . \end{aligned} \tag{2.76}$$

Furthermore, since $\overline{c}_i = \overline{r}_i(0)$, it follows that

$$\begin{aligned} S(\overline{c}_i) &\leq [(m_i + n_i + s_i) + (m_i + n_i)]\kappa \\ &\leq [2(m_i + n_i) + s_i]\kappa . \end{aligned} \tag{2.77}$$

From (2.35), (2.69), and (2.26) we know that $\overline{h}_i$ is the first non-zero coefficient of $(-1)^{m_i+1} d^{s_i+1} \hat{R}_{m_{i+1}-1, n_{i+1}-1}(p,q,z)$. Further, exactly $z^{m_{i+1}+n_{i+1}-1}$ can be factored out. Hence, from an analysis similar to the one used to find bounds for $\overline{r}_i(z)$, it follows that

$$S(\overline{h_i}) \le [2(m_{i+1} + n_{i+1}) + s_{i+1}]\kappa \quad . \tag{2.78}$$

Next, from (2.45), (2.77), and (2.78), the size of $\overline{\beta_{i+1}}$ is bounded by

$$S(\overline{\beta_{i+1}}) \le [(2(m_{i-1} + n_{i-1}) + s_{i-1}) + (2(m_i + n_i) + s_i)s_i]\kappa$$
$$\le [2(m_i + n_i) + s_i](1 + s_i)\kappa \quad .$$

Finally, since $\overline{\omega_{i+1}}(z)$ must satisfy (2.12), it follows from (2.76) and (2.77) that

$$S(\overline{\omega_{i+1}}(z)) = S(\overline{c_i}^{s_i+1}\ \overline{r_{i-1}}(z)) - S(\overline{r_i}(z)) \quad .$$

Therefore,

$$S(\overline{\omega_{i+1}}(z)) \le [2(m_i + n_i) + s_i](s_i + 1)\kappa + 2(m_{i-1} + n_{i-1} + s_{i-1} + 1)\kappa$$
$$- 2(m_i + n_i + s_i + 1)\kappa$$
$$\le [2(m_i + n_i) + s_i](1 + s_i)\kappa \quad .$$

We summarize the bounds on the size of the coefficients of the variables in Table 2.1 below. For convenience, we increase the bounds to provide similar formulas for each variable.

Variable	Bound on size of coefficients
$\overline{u_i}(z)$	$2\kappa(m_i + n_i + s_i + 1)$
$\overline{v_i}(z)$	$2\kappa(m_i + n_i + s_i + 1)$
$\overline{r_i}(z)$	$2\kappa(m_i + n_i + s_i + 1)$
$\overline{c_i}$	$2\kappa(m_i + n_i + s_i + 1)$
$\overline{h_i}$	$2\kappa(m_{i+1} + n_{i+1} + s_{i+1} + 1)$
$\overline{\beta_{i+1}}$	$2\kappa(m_i + n_i + s_i + 1)(s_i + 1)$
$\overline{\omega_{i+1}}(z)$	$2\kappa(m_i + n_i + s_i + 1)(s_i + 1)$

Table 2.1 Bounds on Variable Size in $\overline{J}pade$

A count of the number of unit multiplications required for each step at the ith pass through the algorithm $\overline{J}pade$ can now obtained.

Step 2 requires the computation of $2s_i$ terms of the power series $a(z)\,\overline{v}_i(z) + b(z)\,\overline{u}_i(z)$, namely, the coefficients of z^l, where $l = m_i + n_i + 1, \ldots, m_i + n_i + 2s_i$. The cost of computing one term of $a(z)\,\overline{v}_i(z)$ is

$$[deg(\overline{v}_i(z)) + 1]\; S(a(z))\; S(\overline{v}_i(z)) \;.$$

But only the first $m_i + n_i + 2s_i$ terms of $a(z)$ are needed for the ith iteration. Thus, $S(a(z)) \le (m_i + n_i + 2s_i + 1)\kappa = (m_{i+1} + n_{i+1} + 1)\kappa$. Only $2s_i$ terms of of $a(z)\,\overline{v}_i(z)$ need to be calculated before $\overline{r}_i(z)$ is determined. Hence

$$C(a(z)\,\overline{v}_i(z)) \le 4s_i(n_i + 1)(m_i + n_i + 2s_i + 1)(m_i + n_i + s_i + 1)\kappa^2 \;.$$

Similarly, since $b(z) = -1$, the cost of computing $(-1)\,\overline{u}_i(z)$ is bounded by $(m_i + 1)$. The total cost of Step 2 is not larger than

$$\begin{aligned} &4s_i(n_i + 1)(m_i + n_i + 2s_i + 1)(m_i + n_i + s_i + 1)\kappa^2 + (m_i + 1) \\ &\qquad \le 9s_i(m_i + n_i + s_i + 1)^3\kappa^2 \;. \end{aligned}$$

A detailed cost analysis of the other steps in $\overline{Jpade}$ can be conducted in a similar fashion, and the results are summarized in Table 2.2 below.

Step	# of Unit Multiplications	Bound
2.	$4s_i(n_i+1)(m_i+n_i+2s_i+1)(m_i+n_i+s_i+1)\kappa^2+(m_i+1)$	$9s_i(m_i+n_i+s_i+1)^3\kappa^2$
5.	$2(s_i+1)(n_{i-1}+1)(m_{i-1}+n_{i-1}+s_{i-1}+s_i+1)(m_{i-1}+n_{i-1}+s_{i-1}+1)\kappa^2+(m_{i-1}+1)$	$3(s_i+1)(m_i+n_i+s_i+1)^3\kappa^2$
6.	$[10s_i+4(s_i+1)+16(s_i+1)s_i](s_i+1)(m_i+n_i+s_i+1)^2\kappa^2$	$16(s_i+1)^3(m_i+n_i+s_i+1)^2\kappa^2$
7.	$4(s_i^3+s_i^2)(m_i+n_i+s_i+1)^2\kappa^2$	$8s_i^3(m_i+n_i+s_i+1)^2\kappa^2$
8.	$4(2s_i+1)(m_i+n_i+s_i+1)^2\kappa^2$	$8(s_i+1)(m_i+n_i+s_i+1)^2\kappa^2$
9.	$[8+10(s_i+1)](s_i+1)(m_i+n_i+s_i+1)^3\kappa^2$	$18(s_i+1)^2(m_i+n_i+s_i+1)^3\kappa^2$

Table 2.2 Bounds on Number of Unit Multiplications in $\overline{Jpade}$

The total number of unit multiplications in Steps 2, 5, and 9 is bounded by

$$\begin{aligned} &(18s_i^2 + 48s_i + 3)(m_i + n_i + s_i + 1)^3\kappa^2 \\ &\qquad \le 18(s_i + 2)^2(m_i + n_i + s_i + 1)^3\kappa^2 \;, \end{aligned} \tag{2.79}$$

and the number of unit multiplications in Steps 6, 7, and 8 is bounded by

$$(24s_i^3+48s_i^2+56s_i+24)(m_i+n_i+s_i+1)^2\kappa^2$$
$$\le 24(s_i+1)^3(m_i+n_i+s_i+1)^2\kappa^2 \ . \qquad (2.80)$$

To estimate the total cost of *Jpade*, two cases are taken into consideration. In the case when the step size is bounded by some constant **c** (which is the case of normal power series, with $c=1=s_i$), the number of passes through the algorithm is bounded by m. Then, the total cost is bounded by

$$\sum_{i=0}^{m}[24(c+1)^3(m_i+n_i+c+1)^2\kappa^2 + 18(c+2)^2(m_i+n_i+c+1)^3\kappa^2]$$
$$\le 24(c+1)^3(m+1)(m+n+2)^2\kappa^2 + 18(c+2)^2(m+1)(m+n+2)^3\kappa^2 \ . \qquad (2.81)$$

In this case, the algorithm has a time complexity of $O(\kappa^2(m+n)^4)$ unit operations in **D**.

When the step size is not bounded, the cost of the algorithm is estimated by using the inequality

$$\sum_{i=0}^{l}(m_{i+1})^p \, s_i^q \le m^{p+q} \ , \qquad (2.82)$$

where $p,q>0$ and l is such that $\sum_{i=0}^{l} s_i = m_{l+1} \le m$. Since

$$2m_{i+1} \ge m_{i+1}+n_{i+1} = m_i+n_i+2s_i \ge m_i+n_i+s_i+1 \ ,$$

the bounds (2.79) and (2.80), respectively, can be rewritten as

$$(18s_i^2+48s_i+3)(2m_{i+1})^3\kappa^2 = (144s_i^2+384s_i+24)\, m_{i+1}^3\kappa^2 \qquad (2.83)$$

and

$$(96s_i^3+192s_i^2+224s_i+96)\, m_{i+1}^2\kappa^2 \ . \qquad (2.84)$$

Then, by applying inequality (2.82), a bound for the total cost of *Jpade* is given by

$$\sum_{i=0}^{l}(144s_i^2+384s_i+24)\, m_{i+1}^3\kappa^2 + (96s_i^3+192s_i^2+224s_i+96)\, m_{i+1}^2\kappa^2 \qquad (2.85)$$

$$\leq (96 s_i^3 m_{i+1}^2 + 144 s_i^2 m_{i+1}^3 + 192 s_i^2 m_{i+1}^2 + 224 s_i m_{i+1}^2 + 96 m_{i+1}^2 + 384 s_i m_{i+1}^3 + 24 m_{i+1}^3)\,\kappa^2$$

$$\leq (240 m^5 + 576 m^4 + 248 m^3 + 96 m^2)\,\kappa^2 \ .$$

In the abnormal case, *Jpade* may have a time complexity as large as $O(\kappa^2 m^5)$ unit operations in **D**.

2.7. Content Reduction

Let $p(x)$ and $q(x)$ be polynomials of degree m and n given by (2.24) and (2.25), respectively, and define d and $a(z)$ by (2.26) and (2.27). With m, n, $a(z)$, and d as input, according to Theorem 2.11, Corollary 2.12, and Corollary 2.13, *Jpade* computes

$$\overline{u}_i(z) = \sum_{j=0}^{m_i} \overline{u}_i^{(j)} d^j z^j \in \mathbf{D}[z] \qquad (2.86)$$

and

$$\overline{v}_i(z) = \overline{v}_i^{(0)} + \sum_{j=1}^{n_i} \overline{v}_i^{(j)} d^{j-1} z^j \in \mathbf{D}[z] \qquad (2.87)$$

where m_i and n_i are maximal such that $m_i \leq m$ and $n_i \leq n$, and

$$m_i - n_i = m - n \ .$$

In addition, from Theorem 2.5, there exists $\lambda \in \mathbf{D}$ such that

$$\lambda = GCD\,\{\overline{u}_i(z),\ \overline{v}_i(z)\}\ , \qquad (2.88)$$

and, because of (2.28),

$$a(z) = \frac{\overline{u}_i(z)}{\overline{v}_i(z)}\ .$$

Theorem 2.14. The constant term $\overline{v}_1^{(0)}$ divides $\overline{u}_1(z)$ and $\overline{v}_1(z)$.

Proof: From (2.28), (2.87), and since $x = dz$, it follows that

$$\frac{\sum_{j=0}^{m} p^{(j)} x^j}{d + \sum_{j=1}^{n} q^{(j)} x^j} = \frac{\sum_{j=0}^{m_1} \overline{u}_1^{(j)} x^j}{d\ \overline{v}_1^{(0)} + \sum_{j=1}^{n_1} \overline{v}_1^{(j)} x^j} . \tag{2.89}$$

Acknowledging (2.88), let $\hat{u}(x)$ and $\hat{v}(x)$ be such that

$$\sum_{j=0}^{m_1} \overline{u}_1^{(j)} x^j = \lambda\ \hat{u}(x)$$

and

$$d\ \overline{v}_1^{(0)} + \sum_{j=1}^{n_1} \overline{v}_1^{(j)} x^j = \lambda\ \hat{v}(x) ,$$

where $\lambda \in \mathbf{D}$ and $GCD\{\hat{u}(x), \hat{v}(x)\} = 1$. hence,

$$d\ \overline{v}_1^{(0)} = \lambda\ \hat{v}^{(0)} . \tag{2.90}$$

But, from (2.89), there also exists $\hat{r}(x) \in \mathbf{D}[x]$ such that

$$\sum_{j=0}^{m} p^{(j)} x^j = \hat{u}(x)\ \hat{r}(x)$$

and

$$d + \sum_{j=1}^{n} q^{(j)} x^j = \hat{v}(x)\ \hat{r}(x) .$$

Thus,

$$d = \hat{v}^{(0)}\ \hat{r}^{(0)} . \tag{2.91}$$

Therefore, from (2.90) and (2.91), we conclude that

$$\overline{v}_1^{(0)}\ \hat{r}^{(0)} = \lambda .$$

Hence, $\overline{v}_1^{(0)}$ divides λ and the theorem follows.

Thus, the size of the coefficients in the output (2.86) and (2.87) can be reduced by removing the factor $\overline{v}_i^{(0)}$. In the next chapter, we compute

$$\tilde{u}(x) = \sum_{j=0}^{m_i} \frac{\overline{u}_i^{(j)}}{\overline{v}_i^{(0)}} x^j$$

and

$$\tilde{v}(x) = d + \sum_{j=1}^{n_i} \frac{\overline{v}_i^{(j)}}{\overline{v}_i^{(0)}} x^j .$$

Thus $\tilde{u}(x) / \tilde{v}(x)$ satisfies

$$\frac{\tilde{u}(x)}{\tilde{v}(x)} = \frac{p(x)}{q(x)} .$$

Note that there may still be some common factor from **D** contained in $\tilde{u}(x)$ and $\tilde{v}(x)$.

Chapter 3

Linear Systems with Integral Domain Polynomial Coefficients

In this chapter, the power series method of Moenck and Carter [17] and Cabay and Domzy [4] for solving the linear system (1.1) over a field is generalized to encompass problems over an arbitrary integral domain $\mathbf{D}$. An iterative formula for constructing successive terms in the power series solution is given in Section 3.1. Each power series component in the solution vector is shown to satisfy the input constraints imposed on the algorithm $\overline{Jpade}$ described in Chapter 2. Thus, in Section 3.2 we are able to use the results of Chapter 2 and Section 3.1 to develop the algorithm *Solve* which computes the solution to (1.1) in rational form. The complexity of *Solve* is analyzed in Section 3.3.

3.1. A Power Series Solution

Let

$$M(x) = \sum_{i=0}^{\delta} M^{(i)} x^i \in \mathbf{D}_{n,n}[x] \tag{3.1}$$

and

$$G(x) = \sum_{i=0}^{\delta} G^{(i)} x^i \in \mathbf{D}_n[x] \ . \tag{3.2}$$

In this chapter, we consider the problem of solving for $F(x)$ the system of linear equations of order n

$$M(x)\, F(x) = G(x) \ . \tag{3.3}$$

To ensure the existence of a solution to (3.3), we assume that $|M(x)| \neq 0$. As a consequence, there exists a point x_0 such that $|M(x_0)| \neq 0$. Without loss of generality, we assume that $x_0 = 0$ is such a point; that is, we assume

$$|M^{(0)}| \neq 0 \ .$$

Let $M^{adj}(x)$ denote the adjoint matrix of $M(x)$. As a consequence, $M^{adj}(x) = |M(x)| \cdot M(x)^{-1}$. Thus,

$$P(x) = M^{adj}(x) \cdot G(x) \tag{3.4}$$

where $P(x) \in \mathbf{D}_n[x]$ and

$$q(x) = |M(x)| \; . \tag{3.5}$$

Formally, for some non-negative integer Δ,

$$P(x) = \sum_{i=0}^{\Delta} P^{(i)} x^i$$

and

$$q(x) = \sum_{i=0}^{\Delta} q^{(i)} x^i \; .$$

Let $d = q(0)$. Then

$$d = |M^{(0)}| \tag{3.6}$$

and, consequently, $d \neq 0$.

By Cramer's rule, a solution to (3.3) is given by

$$F(x) = \frac{P(x)}{q(x)} \in \mathbf{D}_n(x) \; . \tag{3.7}$$

In addition, Hadamard's inequality applied to (3.4) and (3.5) yields the bound

$$\Delta \leq \delta n$$

for the degrees of $P(x)$ and $q(x)$. More precise bounds may be obtained, but they do not provide a significant decrease in the bound, (c.f. McClellan [16], for example).

Similar to the methods of Moenck and Carter [17] and Cabay and Domzy [4] for solving (3.3) when $\mathbf{D}$ is a field, we wish to express the solution $F(x)$ as a vector of formal power series. We will see that the power series components of the vector calculated will satisfy the constraints for $\overline{Jpade}$. Consequently, $\overline{Jpade}$ may be used to obtain a rational solution that is in reduced form, except for a factor from $\mathbf{D}$ in each component of the solution vector.

Let

$$M' = M^{(0)adj} \in \mathbf{D}_{n,n} \tag{3.8}$$

and, for $i = 0,1,\ldots$, determine the sequence of vectors

$$A^{(i)} = M' \left[d^i\, G^{(i)} - \sum_{j=1}^{\delta} d^{j-1}\, M^{(j)} A^{(i-j)} \right] \in \mathbf{D}_n \tag{3.9}$$

where d is given by (3.6) †.

Theorem 3.1. Let

$$A(z) = \sum_{i=0}^{\infty} A^{(i)} z^i \in \mathbf{D}_n[[z]] \tag{3.10}$$

Then, together with $M(x)$, $G(x)$, and d given by (3.1), (3.2), and (3.6), respectively,

$$M(x) \cdot A(z) = d \cdot G(x) \tag{3.11}$$

where $x = dz$.

Proof: From (3.6) and (3.8)

$$M' \cdot M^{(0)} = d \cdot \mathbf{I}_n .$$

Then, multiplying both sides of (3.9) by M' and re-arranging the terms, we obtain ‡

$$\sum_{j=0}^{i} d^j\, M^{(j)}\, A^{(i-j)} = d^{i+1}\, G^{(i)} .$$

† Note that $A^{(i-j)} = 0$ if $i - j < 0$.

‡ Also, $M^{(j)} = 0$ if $j > \delta$, and $G^{(i)} = 0$ if $i > \delta$.

Hence, with $x = dz$,

$$\begin{aligned} M(x) \cdot A(z) &= \sum_{j=0}^{\delta} M^{(j)} x^j \cdot \sum_{i=0}^{\infty} A^{(i)} z^i \\ &= \sum_{i=0}^{\infty} z^i \sum_{j=0}^{i} d^j M^{(j)} A^{(i-j)} \\ &= \sum_{i=0}^{\infty} d^{i+1} G^{(i)} z^i \\ &= d\, G(x) . \end{aligned}$$

■

At this point we note that by finding the determinant and adjoint matrix of $M(0)$ we have, in affect, found $M(0)^{-1}$. As we will see in Section 3.3, the cost of this calculation is insignificant when compared with the rest of the algorithm which will be specified in Section 3.2. We have chosen to express the problem in terms of d and M', rather than in terms of the LU-decomposition of $M(0)$, say, primarily for the sake of simplicity in the presentation.

Corollary 3.2. The vector $A(z)$ of power series (3.10) is related to the vector of rational expressions $\dfrac{P(x)}{q(x)}$ given by (3.4) and (3.5) according to

$$q(x) \cdot A(z) = d\, P(x) . \qquad (3.12)$$

Proof: From (3.7), it follows that

$$M(x) \cdot P(x) = q(x)\, G(x) . \qquad (3.13)$$

The relationship (3.12) follows from (3.11) and (3.13), since $|M(x)| \neq 0$.

■

3.2. Algorithm *Solve*

The algorithm *Solve* presented below is designed to handle linear systems of univariate polynomials over an arbitrary integral domain **D**. It first determines whether $M(x)$ is singular or not. If it is singular we simply return this fact and halt. If it is non-singular we construct a vector of truncated power series and then apply the algorithm $\overline{Jpade}$, which is presented in Section 2.5, to each component of the vector. Finally we make use of Theorem 2.14 to obtain the exact rational solution to the linear system.

Algorithm *Solve* $\left(M(x)\ ,\ G(x)\ ,\ \delta\ ,\ n\right)$:

Input: M, G, δ, and n, where

(1) $M(x) \in \mathbf{D}_{n\ n}[x]$, $deg(M(x)) \leq \delta$, and

(2) $G(x) \in \mathbf{D}_n[x]$, $deg(G(x)) \leq \delta$.

Output: $\left(F(x)\ ,\ Singular\right)$, where $F(x) \in \mathbf{D}_n(x)$ and either

(1) $M(x)$ is invertible in which case $M(x)\,F(x) - G(x) = \mathbf{0}_n$ and $Singular$ = **FALSE**, or

(2) $M(x)$ is not invertible in which case $F(x) = \mathbf{0}_n$ and $Singular$ = **TRUE**.

Step 1: $ Find a point x_l such that $|M(x_l)| \neq 0$ $.

$l \leftarrow -1$

$d \leftarrow 0$

While $l < \delta n$ and $d = 0$ do

$l \leftarrow l+1$

$d \leftarrow |M(x_l)|$.

$ We assume that the user will provide a method of find the points to examine when he select the arbitrary integral domain D to work in $

Step 2: $ Test for singularity $

If $d = 0$ then Return $\left(F(x) \leftarrow \mathbf{0}_n , \text{TRUE}\right)$.

Step 3: $ Obtain new representation $

$$M(x) \leftarrow \sum_{i=0}^{\delta} M^{(i)} (x - x_l)^i$$

$$G(x) \leftarrow \sum_{i=0}^{\delta} G^{(i)} (x - x_l)^i ;$$

Step 4: $ Calculate the adjoint matrix of $M^{(0)}$ $

$$M' \leftarrow M^{(0)adj} .$$

Step 5: $ Calculation of Power Series Solution $A(x)$, c.f. (3.9) $

$i \leftarrow 0$

While $i \leq 2\delta n$ do

$$A^{(i)} \leftarrow M' \left[d^i G^{(i)} - \sum_{j=1}^{\delta} d^{j-1} M^{(j)} A^{(i-j)} \right]$$

$i \leftarrow i + 1$.

Step 6: $ Find Padé Fractions, c.f. Chapter 2 $

$i \leftarrow 1$

While $i \leq n$ do

$$\begin{bmatrix} \overline{U}_i(z) \\ \overline{V}_i(z) \end{bmatrix} = \overline{J}pade(A_i(z), \delta n, \delta n, d)$$

$i \leftarrow i+1$

Step 7: \$ Transform Padé fractions in z to Padé fractions in x, c.f. Theorem 2.14 \$

$i \leftarrow 1$

While $i \leq n$ do

$$U_i(x) \leftarrow d \, \frac{\overline{U}_i(\frac{x+x_l}{d})}{\overline{V}_i(0)}$$

$$V_i(x) \leftarrow d \, \frac{\overline{V}_i(\frac{x+x_l}{d})}{\overline{V}_i(0)}$$

$i \leftarrow i+1$

Step 8: \$ Return Solution \$

$$\text{Return}\left(F(x) \leftarrow \left[\frac{U_1(x)}{V_1(x)}, \ldots, \frac{U_n(x)}{V_n(x)}\right]^T, \textbf{FALSE}\right).$$

■

We present the following example to show the operation of algorithm *Solve*. For convenience we use the integral domain **D** of the integers.

Example 3.3. We wish to solve the following linear system for $F(x)$.

$$\begin{bmatrix} x+2x^2 & 2 & 4-x \\ -1-x^2 & 3x+x^2 & 0 \\ 1-2x & 2 & x^2 \end{bmatrix} \cdot \begin{bmatrix} F_1(x) \\ F_2(x) \\ F_3(x) \end{bmatrix} = \begin{bmatrix} 1+2x \\ 0 \\ -1 \end{bmatrix}.$$

In Step 1 we determine that the point $x_l = 0$ is suitable and that $M(x)$ is invertible. In Steps 3 and 4 we find the determinant of $M(0)$, $d = -8$, and the adjoint matrix

$$M' = \begin{bmatrix} 0 & 8 & 0 \\ 0 & -4 & -4 \\ -2 & 2 & 2 \end{bmatrix}.$$

In Step 5 we calculate the vector of power series $A(z)$ to 13 terms:

$$A(z) = \begin{bmatrix} 0 \\ 4 \\ -4 \end{bmatrix} + \begin{bmatrix} -96 \\ 48 \\ 16 \end{bmatrix} z + \begin{bmatrix} -896 \\ 1344 \\ -896 \end{bmatrix} z^2 + \begin{bmatrix} -23040 \\ 18176 \\ -6016 \end{bmatrix} z^3 + \begin{bmatrix} -292864 \\ 359424 \\ -185088 \end{bmatrix} z^4$$

$$+ \begin{bmatrix} -5988352 \\ 5529600 \\ -2243072 \end{bmatrix} z^5 + \begin{bmatrix} -90963968 \\ 99311616 \\ -47774720 \end{bmatrix} z^6 + \begin{bmatrix} -1646329856 \\ 1622654976 \\ -706078720 \end{bmatrix} z^7$$

$$+ \begin{bmatrix} -26766082048 \\ 28082470912 \\ -13010890752 \end{bmatrix} z^8 + \begin{bmatrix} -464764272640 \\ 469105311744 \\ -209380483072 \end{bmatrix} z^9 + \begin{bmatrix} -7748220092416 \\ 8008572731392 \\ -3658539319296 \end{bmatrix} z^{10}$$

$$+ \begin{bmatrix} -132438092152832 \\ 134904982274048 \\ -60759395958784 \end{bmatrix} z^{11} + \begin{bmatrix} -2229284833853440 \\ 2291220412366848 \\ -1041024555614208 \end{bmatrix} z^{12} .$$

In Step 6 we call $\overline{J}pade$ to obtain the Padé fraction of type (6,6):

To illustrate the effect of $\overline{J}pade$ we print out the intermediate steps:

$$s_{-1} = 1$$

$$\overline{r}_{-1}(z) = -96z - 896z^2 - 23040z^3 - 292864z^4 - 5988352z^5 - 90963968z^6$$
$$- 1646329856z^7 - 26766082048z^8 - 464764272640z^9 - 7748220092416z^{10}$$
$$- 132438092152832z^{11} - 2229284833853440z^{12}$$

$$\overline{u}_0(z) = 1$$

$$\overline{u}_{-1}(z) = 0$$

$$\overline{v}_0(z) = 0$$

$$\overline{v}_{-1}(z) = \frac{1}{z}$$

$i = 0$ iteration:

$$\overline{r}_0(z) = -1$$

$$s_0 = 0$$

$$\overline{\beta}_1 = -1$$

$$\overline{u}_1(z) = 0$$

$$\overline{v}_1(z) = 1$$

$i = 1$ iteration:

$$\overline{r}_1(z) = -96z - 896z^2 - 23040z^3 - 292864z^4 - 5988352z^5 - 90963968z^6$$
$$-1646329856z^7 - 26766082048z^8 - 464764272640z^9 - 7748220092416z^{10}$$
$$-132438092152832z^{11} - 2229284833853440z^{12}$$

$s_1 = 1$

$\overline{\beta}_2 = -1$

$\overline{u}_2(z) = -9216z$

$\overline{v}_2(z) = 96 - 896z$

$i = 2$ iteration:

$$\overline{r}_2(z) = -1409024z^3 - 7471104z^4 - 312475648z^5 - 3366977536z^6 - 765439508048z^7$$
$$-1094432325632z^8 - 20634960658132z^9 - 3274003405864 96z^{10}$$
$$-5771651643867136z^{11} - 95346813480992768z^{12} + 1997439211132682240z^{13}$$

$s_2 = 1$

$\overline{\beta}_3 = 589824$

$\overline{u}_3(z) = 2113536z + 8519680z^2$

$\overline{v}_3(z) = -22016 + 116736z + 4194304z^2$

$i = 3$ iteration:

$$\overline{r}_3(z) = 1015021568z^5 + 752458137600z^6 + 509859594240z^7 + 15565565460480z^8$$
$$+ 202476972146688z^9 + 4065000425473024z^{10} + 618981683374325 76z^{11}$$
$$+ 1121251250063540224z^{12} - 8157234140337070407 68z^{13} - 9350298295770818805760z^{14}$$

$s_3 = 1$

$\overline{\beta}_4 = 127062312484864$

$\overline{u}_4 = 23789568z - 1541537792z^2 - 6675234816z^3$

$\overline{v}_4(z) = -247808 + 18370560z - 42450944z^2 - 3256745984z^3$

$i = 4$ iteration:

$$\overline{r}_4(z) = -55090471763968z^7 - 247104500924416z^8 - 1040081735804518 4z^9$$
$$-11998543474170265 6z^{10} - 2619911032444813312z^{11}$$
$$-37988866401748647936z^{12} - 1009710409482402319564 8z^{13}$$
$$\pm 52595247038931921587404 8z^{14} + 72602144298442979645849 60z^{16}$$

$s_4 = 1$

$\overline{\beta}_5 = 659372021443314431936$

$\overline{u}_5(z) = 201747333312z + 97805139968z^2 + 571689926656z^3 - 2121512517632z^4$

$\overline{v}_5(z) = -210153472 + 942628864z + 35683860480z^2 + 103926988800z^3 + 699335180288z^4$

$i = 5$ iteration:

$$\overline{r}_5(z) = 52462647563517952z^9 + 385591031279648768z^{10} + 1099083038143492 9152z^{11}$$
$$+ 14556063684805577932 8z^{12} - 795756582224930499723 2640z^{13}$$
$$- 987319839965077202618286 08z^{14} - 32430147701259584449124 7616z^{15}$$
$$- 155901731119619958816899 0720z^{16}$$

$s_5 = 1$

$\overline{\beta}_6 = 1942374450672995508161987 21536$

$\overline{u}_6(z) = 300193677312z + 595438075904z^2 + 4675088678912z^3 - 7777025892352z^4$
$- 4166118277120z^5$

$\overline{v}_6(z) = -3127017472 + 229830164 48z + 487277199360z^2 + 285820846080z^3$
$+ 5374866358272z^4 - 7595434914 6112z^5$

$i = 6$ iteration:

$$\overline{r}_6(z) = -27303072740933 6320z^{11} - 2730307274093363200z^{12}$$
$$- 1184494009713647449003786 24z^{13} - 1130478017644647187251986432z^{14}$$
$$- 7605021092710010684828549 12z^{15} - 192285896486368204637706 6496z^{16}$$
$$+ 16932387861663646225455382 5280z^{17}$$

$s_6 = 1$

$\overline{\beta}_7 = 176149080919929350747543694481555456$

$\overline{u}_7(z) = 244108849280z - 16273899520z^2 + 260382392320z^3 - 7290706984960z^4$
$+ 16664473108480z^5$

$\overline{v}_7(z) = -254279680 + 2542796800z + 34582036480z^2 - 81369497600z^3$
$+ 390573588480z^4 - 7290706984960z^5 + 16664473108480z^6$

$i = 7$ iteration:

$$\overline{r}_7(z) = -963194470779571694206 9760z^{13} - 6640062107849167661432 8320z^{14}$$
$$+ 178413916659970562903244800z^{15} - 3425245913740107334025 2160z^{16}$$
$$+ 140460514844214031066490 47040z^{17} - 3714985716489295561354117 1200z^{18}$$

$s_7 = 1$

$n_8 > n = 6$, so terminate, returning $\overline{u}_7(z)$ and $\overline{v}_7(z)$.

Thus,

$$\overline{U}_1(z) = 24410849280z - 16273899520z^2 + 260382392320z^3 - 7290706984960z^4 + 16664473108480z^5$$

and

$$\overline{V}_1(z) = -254279680 + 2542796800z + 34582036480z^2 - 81369497600z^3 + 390573588480z^4 - 7290706984960z^5 + 16664473108480z^6$$

Similarly, $A_2(z)$ and $A_3(z)$ also yield Padé fractions of type (6,6) via $\overline{J}pade$:

$$\overline{U}_2(z) = 2034237440 + 4068474880z + 162738995200z^2 - 260382392320z^3 + 2083059138560z^4 - 33328946216960z^5$$

$$\overline{V}_2(z) = 508559360 - 5085593600z - 69164072960z^2 + 162738995200z^3 - 781147176960z^4 + 14581413969920z^5 - 33328946216960z^6$$

and

$$\overline{U}_3(z) = 508559360 - 7119831040z + 65095598080z^2 + 65095598080z^3 - 1041529569280z^4$$

$$\overline{V}_3(z) = -127139840 + 1271398400z + 17291018240z^2 - 40684748800z^3 + 195286794240z^4 - 3645353492480z^5 + 8332236554240z^6 .$$

In Step 7 we calculate $U_i(x) = \dfrac{\overline{U}_i(\frac{x}{d})}{\overline{V}_i(0)}$, and $V_i(x) = d\,\dfrac{\overline{V}_i(\frac{x}{d})}{\overline{V}_i(0)}$, for each i:

$$U_1(x) = 12x + x^2 + 2x^3 + 7x^4 + 2x^5$$

and

$$V_1(x) = -8 - 10x + 17x^2 + 5x^3 + 3x^4 + 7x^5 + 2x^6 ,$$

$$U_2(x) = 4 - x + 5x^2 + x^3 + x^4 + 2x^5$$

and

$$V_2(x) = -8 - 10x + 17x^2 + 5x^3 + 3x^4 + 7x^5 + 2x^6 ,$$

and finally

$$U_3(x) = -4 - 7x - 8x^2 + x^3 + 2x^4$$

and

$$V_3(x) = -8 - 10x + 17x^2 + 5x^3 + 3x^4 + 7x^5 + 2x^6 .$$

Finally, in Step 8 we return the solution

$$F(x) = \left[\frac{U_1(x)}{V_1(x)} , \frac{U_2(x)}{V_2(x)} , \frac{U_3(x)}{V_3(x)} \right]^T .$$

■

3.3. Cost Analysis

Step 4 in the algorithm *Solve* requires that the user specify $\delta n+1$ values x_l in the integral domain **D**. Expanding the input variables $M(x)$ and $G(x)$ about any such point x_l yields the representation

$$M(x) = \sum_{i=0}^{\delta} M_l^{(i)} (x - x_l)^i \ , \ M_l^{(i)} \in \mathbf{D}_{n,n} \ ,$$

and

$$G(x) = \sum_{i=0}^{\delta} G_l^{(i)} (x - x_l)^i \ , \ G_l^{(i)} \in \mathbf{D}_n \ .$$

For all values of x_l, $l=0,\ldots,\delta n$, let ∂ be a bound on the size of the coefficients of $M_l^{(i)}$ and $G_l^{(i)}$. That is, for $l=0,\ldots,\delta n$ and $i=0,\ldots,\delta$

$$\mathbf{S}(M_l^{(i)}), \mathbf{S}(G_l^{(i)}) \leq \partial \ . \qquad (3.14)$$

To minimize the size of ∂, it is important to select the values x_l to be of small size. Indeed, it is recommended that $x_0 = 0$ is a desirable choice. Then, if $M(0)$ is non-singular (a likely possibility for practical problems), the algorithm exits from Step 1 in one iteration, and no growth of coefficients is encountered in the expansion of $M(x)$ and $G(x)$ about the point $x_0 = 0$ in Step 3 of *Solve*. Based on the assumption (3.14), we now obtain bounds for the cost of each step of *Solve*.

In Step 1, the cost of evaluating $M(x_l)$ is insignificant. To find the determinant $|M(x_l)|$, we propose that the fraction-free methods of Bareiss [1] be used to obtain a triangular decomposition of the matrix $M(x_l) \in \mathbf{D}_{n,n}$ which yields $|M(x_l)|$ as a byproduct. Generalizing Higginson's [10] analysis of the one-step fraction-free method over the domain of integers to an arbitrary integral domain $\mathbf{D}$, we find that

$$\mathbf{C}(|M(x_l)|) \leq \partial^2 n^5 \tag{3.15}$$

unit multiplications. Thus, if $M(x_0)$ is non-singular, only one iteration in Step 1 is required and (3.15) represents an estimate for the total cost of Step 1. On the other hand, in the worst case, when $M(x)$ is singular, $\delta n+1$ iterations in Step 1 are required and an estimate for the upper bound of the cost of Step 1 is therefore

$$\leq 2\partial^2 \delta n^6 .$$

Steps 2 and 3 are insignificant, affecting only the low order terms of the cost analysis, so we now consider Step 4. To find a bound on the cost of Step 4 we propose, first, to solve

$$M^{(0)} \cdot H = \mathbf{I}_{n,n}$$

using fraction-free methods to obtain

$$H = |M^{(0)}|^{-1} .$$

Since the triangular decomposition of $M(x_l)$ is already available from Step 1, only forward and back substitutions are required to obtain H. This requires at most

$$\leq \partial^2 n^5 \tag{3.16}$$

additional unit multiplications in $\mathbf{D}$ (c.f. Higginson [10]).

To obtain $M^{(0)^{adj}}$, it is only necessary (at insignificant cost) to compute

$$M' = M^{(0)^{adj}} = d\,H \;, \tag{3.17}$$

where $d = |M^{(0)}|$, and normalize the result to ensure that $M' \in \mathbf{D}_{n,n}$.

Thus, from (3.16) and (3.17) it follows that Step 4 requires

$$\leq 2\partial^2 n^5$$

unit multiplications.

To obtain an estimate for a bound on the cost of Step 5, we first use Hadamard's inequality to obtain

$$\mathbf{S}(d) \leq \partial n$$

and

$$\mathbf{S}(M') \leq \partial(n-1) \;.$$

Then, using (2.72), (2.73), (3.14), and the fact that

$$\mathbf{S}(A^{(i)}) \leq (i+1)\partial n$$

(see (2.75)), it follows that †

$$\mathbf{S}(M^{(j)}A^{(i-j)}) \leq (i-j+1)\partial n + \partial \;.$$

Finally, if d^{i-1} has been computed, for $i \geq 1$, then the cost of computing d^i is just

$$\mathbf{C}(d^i) \leq i\partial^2 n^2 \;.$$

Thus, assuming d^{j-1} has been computed for $1 \leq j \leq \delta$,

$$\begin{aligned}\mathbf{C}\Big(\sum_{j=1}^{\delta} d^{j-1}M^{(j)}A^{(i-j)}\Big) &= n\sum_{j=1}^{\delta}\mathbf{S}(d^{j-1})\,\mathbf{S}(M^{(j)}A^{(i-j)}) + \sum_{j=1}^{\delta}\mathbf{C}(M^{(j)}A^{(i-j)}) \\ &\leq n\sum_{j=1}^{\delta}[(j-1)\partial n][(i-j+1)\partial n+\partial] + n^2\sum_{j=1}^{\delta}\partial[(i-j+1)\partial n] \\ &\leq \partial^2 n^2\sum_{j=1}^{\delta}(j-1)[(i-j+1)n+1] + \partial^2 n^3\sum_{j=1}^{\delta}(i-j+1) \;.\end{aligned}$$

† Note that $(i-j+1) = 0$ if $i-j+1 < 0$.

Consequently, assuming d^{i-1} has already been computed, for $i>0$,

$$\begin{aligned}
C(A^{(i)}) &= C(d^i M' G^{(i)} - M' \sum_{j=1}^{\delta} d^{j-1} M^{(j)} A^{(i-j)}) \\
&\le C(d^i) + n\, S(d^i)\, S(M' G^{(i)}) + C(M' G^{(i)}) \\
&\quad + n^2 S(M')\, S(\sum_{j=1}^{\delta} d^{j-1} M^{(j)} A^{(i-j)}) + C(\sum_{j=1}^{\delta} d^{j-1} M^{(j)} A^{(i-j)}) \\
&\le (i-1)\partial^2 n^2 + i\partial^2 n^3 + \partial^2 n^2 (n-1) + \partial^2 n^2 (n-1)(in+1) \\
&\quad + \partial^2 n^2 \sum_{j=1}^{\delta} (j-1)[(i-j+1)n+1] + \partial^2 n^3 \sum_{j=1}^{\delta} (i-j+1) \\
&= i\partial^2 n^2 + i\partial^2 n^4 + 2\partial^2 n^3 - 3\partial^2 n^2 \\
&\quad + \partial^2 n^2 \sum_{j=1}^{\delta} (j-1)[(i-j+1)n+1] + \partial^2 n^3 \sum_{j=1}^{\delta} (i-j+1) .
\end{aligned}$$

Since the first $2\delta n+1$ terms only of $A(z)$ are required,

$$\begin{aligned}
C(A(z)) &\le \sum_{i=0}^{2\delta n} (C(A^{(i)}) \\
&\le \sum_{i=0}^{2\delta n} [i\partial^2 n^2 + i\partial^2 n^4 + 2\partial^2 n^3 - 3\partial^2 n^2] + \partial^2 n^2 \sum_{i=0}^{2\delta n} \sum_{j=1}^{\delta} (j-1)[(i-j+1)n+1] \\
&\quad + \partial^2 n^3 \sum_{i=0}^{2\delta n} \sum_{j=1}^{\delta} (i-j+1) \\
&\le 3\partial^2 \delta^2 n^6 + 9\partial^2 \delta^4 n^5 .
\end{aligned}$$

Thus, the cost of Step 5 is

$$\le 9\partial^2\delta^4 n^5 + 3\partial^2\delta^2 n^6 .$$

In Step 6 we obtain the Padé fraction of type $(\delta n, \delta n)$ for each component $A_i(z)$ of $A(z)$. From (3.12),

$$A_i(z) = d\,\frac{P_i(dz)}{q(dz)}$$

where the coefficients of $P_i(z)$ are bounded by ∂n. Thus, when the power series is normal, from (2.81), it follows that the complexity of applying $\overline{Jpade}$ to $A_i(z)$ to yield

$\frac{\overline{U}_i(z)}{\overline{V}_i(z)}$ such that $\frac{\overline{U}_i(z)}{\overline{V}_i(z)} = d\,\frac{P_i(dz)}{q(dz)}$ is bounded by

$$C\left(\frac{\overline{U}_i(z)}{\overline{V}_i(z)}\right) \leq 768(c+1)^3\partial^2\delta^3 n^5 + 2204(c+2)^2\partial^2\delta^4 n^6$$

unit multiplications, where c is a constant associated with the step-size within the *Jpade*. Depending on the abnormality of the power series, the bound, using (2.85), could become

$$C\left(\frac{\overline{U}_i(z)}{\overline{V}_i(z)}\right) \leq 920\partial^2\delta^4 n^6 + 230\partial^2\delta^5 n^7$$

unit multiplications.

Thus, the cost in Step 6 to compute all n such Padé fractions is bounded by

$$\leq 768(c+1)^2\partial^2\delta^3 n^6 + 2204(c+2)^2\partial^2\delta^4 n^7$$

when all n power series are normal, and, depending on the abnormality,

$$\leq 920\partial^2\delta^4 n^7 + 230\partial^2\delta^5 n^8$$

when all n power series are not normal.

In Step 7, with Theorem 2.14, since we know that the Padé fractions calculated in Step 6 are exact solutions, it follows that we may divide $\overline{V}_i(0)$ through $\overline{V}_i(z)$ and $\overline{U}_i(z)$ for each $1 \leq i \leq n$.

From Table 2.1 and (2.82) we know that the size of the coefficients of $\overline{U}_i(z)$ are bounded by

$$S(\overline{U}_i(z)) \leq 2(\delta n + \delta n + 1)(\partial n) \leq 6\partial\delta n^2$$

and that the size of the coefficients of $\overline{V}_i(z)$ are bounded by

$$S(\overline{V}_i(z)) \leq 2(\delta n + \delta n + 1)(\partial n) \leq 6\partial\delta n^2 .$$

Also, the δnth degree term of $\overline{V}_i(z)$ must have a factor of size ∂n (Hadamard's rule), a factor of size $(\delta n-1)\partial n$ (from $d^{\delta n-1}$), and the factor $\overline{V}_i(0)$. Consequently, it follows that

$$S(\overline{V}_i(0)) \le 6\partial\delta n^2 - \partial n - (\delta n-1)\partial n = 5\partial\delta n^2 .$$

Dividing through $\overline{V}_i(z)$ with $\overline{V}_i(0)$ costs

$$\begin{aligned} C\left(\frac{\overline{V}_i(z)}{\overline{V}_i(0)}\right) &\le \delta n \; 6\partial\delta n^2 \; 5\partial\delta n^2 \\ &= 30\partial^2\delta^3 n^5 \end{aligned}$$

unit multiplications. Similarly,

$$\begin{aligned} C\left(\frac{\overline{U}_i(z)}{\overline{V}_i(0)}\right) &\le \delta n \; 6\partial\delta n^2 \; 5\partial\delta n^2 \\ &= 30\partial^2\delta^3 n^5 \end{aligned}$$

unit multiplications. Thus, over all $2n$ components it costs

$$\le 60\partial^2\delta^3 n^6 \qquad (3.18)$$

unit multiplications. The resulting polynomials will be referred to as $U_i(z)$ and $V_i(z)$. Also, by Hadamard's inequality and (3.12), $S(U_i(z)) \le (\delta n+1)\partial n$ and $S(V_i(z)) \le (\delta n)\partial n$. Therefore, the cost of making the substitution $x = dz$, for all $2n$ polynomials is

$$\begin{aligned} &\le (2n)(\delta n)((\delta n+1)\partial n)(\delta n\partial n) \\ &\le 4\partial^2\delta^3 n^6 . \end{aligned} \qquad (3.19)$$

Thus, the cost of Step 7 is, from (3.18) and (3.19)

$$\le 64\partial^2\delta^3 n^6 .$$

We summarize the cost of each step in Table 3.1 below.

Step		# of Unit Multiplications	Normal Case Complexity	Abnormal Case Complexity
1.		$2\partial^2\delta n^6$	$\partial^2\delta n^6$	$\partial^2\delta n^6$
4.		$2\partial^2 n^5$	$\partial^2 n^5$	$\partial^2 n^5$
5.		$9\partial^2\delta^4 n^5 + 3\partial^2\delta^2 n^6$	$\partial^2\delta^4 n^5 + \partial^2\delta^2 n^6$	$\partial^2\delta^4 n^5 + \partial^2\delta^2 n^6$
6.	Normal	$768(c+1)^3\partial^2\delta^3 n^6 + 2204(c+2)^2\partial^2\delta^4 n^7$	$\partial^2\delta^3 n^6 + \partial^2\delta^4 n^7$	---
	Abnormal	$920\partial^2\delta^4 n^7 + 230\partial^2\delta^5 n^8$	---	$\partial^2\delta^4 n^7 + \partial^2\delta^5 n^8$
7.		$64\partial^2\delta^3 n^6$	$\partial^2\delta^3 n^6$	$\partial^2\delta^3 n^6$

Table 3.1 Bounds on Number of Unit Multiplications in *Solve*

From Table 3.1 it is clear that the cost of obtaining the truncated power series solution to (3.3) after Step 5 is only $O(\partial^2\delta^2 n^6 + \partial^2\delta^4 n^5)$. The cost of fraction-free method is, as we have seen earlier, $O(\partial^2\delta^2 n^7)$. Clearly, except when $\delta >> n$, the power series method is superior. In addition, this is the cost of obtaining the first $2\delta n + 1$ terms of the power series. The cost will be even smaller if fewer terms are required. Finally, $2\delta n + 1$ terms completely characterizes the corresponding rational functions for each power series in the solution.

When the rational solution is desired, *Jpade* dominates the complexity of *Solve*. When all n of the power series in $A(z)$ are normal, the complexity of *Solve* becomes $O(\partial^2\delta^4 n^7 + \partial^2\delta^4 n^5 + \partial^2\delta^2 n^6)$. This compares favorably to the fraction-free methods which, to get a similarly reduced solution, would cost $O(\partial^2\delta^4 n^7 + \partial^2\delta^2 n^7)$. Depending on the abnormality of the power series, the time complexity of *Solve* could become $O(\partial^2\delta^5 n^8 + \partial^2\delta^4 n^5 + \partial^2\delta^2 n^6)$. This also compares favorably with Bareiss' [1] method which would then become $O(\partial^2\delta^5 n^8 + \partial^2\delta^4 n^7)$. In order to obtain these reduced solutions from the fraction-free methods it is necessary to use Cabay and Kossowski's JPADE [5] algorithm. It, as with *Jpade* in our algorithm *Solve*, is the dominating cost in the reduction.

Finally, when any rational solution is sufficient (which certainly makes sense when the integral domain **D** is not a *GCD* domain), the fraction-free method is asymptotically smaller, being of complexity $O(\partial^2\delta^2 n^7)$.

Chapter 4

Future Considerations

In Chapter 3 we generalize and develop the power series method for solving linear systems of univariate polynomials over an arbitrary integral domain **D**. We also compare it with the fraction-free methods of Bareiss [1]. When the power series solution is required, the power series method is superior to the fraction-free method except, perhaps, for large δ and small n. Such a power series solution (specifically, the first $2\delta n+1$ terms) is sufficient to completely characterize both the associated rational functions and the entire (infinite) power series solution. The cost of obtaining this solution is only $O(\partial^2\delta^2 n^6+\partial^2\delta^4 n^5)$, whereas the fraction-free method is $O(\partial^2\delta^2 n^7)$. Furthermore, the fraction-free solution is in unreduced rational form and a further division-step is required to obtain the power series form. As mentioned in Chapter 1, the solution in power series form is required in several problem areas.

When the solution is desired in reduced rational form there are two possible ways to proceed. Either the truncated power series solution can be converted into reduced rational form using *Jpade*, or the rational solution obtained by the fraction-free methods can be reduced using Cabay and Kossowski's [5] JPADE algorithm. JPADE is the best method available for reducing the rational functions obtained by the fraction-free method due to the duality of JPADE with Euclid's extended algorithm. However, computing the reduced rational form using JPADE requires the same time complexity as *Jpade* does for converting the truncated power series solution. Since computing the truncated power series solution is less costly than the unreduced rational solution, the power series method is again superior to the fraction-free method when the reduced rational form is desired. Here, reduced means the *GCD* of a rational function will be an element of the integral domain **D**.

The only time the cost of the fraction-free methods is asymptotically smaller than the cost of the the power series method is when the solution is desired in rational form, but not necessarily in reduced rational form. This is a reasonable form when the integral domain **D** is not a *GCD* domain, but not generally useful otherwise.

The algorithm $\overline{Jpade}$ by itself deserves recognition as a contribution to Padé theory. With the same *a priori* information, $\overline{Jpade}$ is two orders of magnitude better than JPADE. In all situations where the power series $A(x)$ in $\mathbf{D}[x]$ is known to have an exact rational solution, $\overline{Jpade}$ is clearly superior. Further study into possible applications for $\overline{Jpade}$ is warranted; one possibility would be the use of $\overline{Jpade}$ as a homomorphism between the rational polynomial algebra $\mathbf{D}(x)$ and the power series algebra $\mathbf{D}[[z]]$ for certain algebraic operations such as addition or multiplication of rational polynomial functions.

One issue that is important to recognize is the validity of the basic assumptions made in the cost analysis (see (2.72) and (2.73)) of $\overline{Jpade}$ and *Solve* concerning the behavior of the integral domain **D** under addition and multiplication. Are these realistic assumptions? It is entirely possible that there exists an integral domain that cannot be represented in our model, just as there exist fields that do not satisfy the assumption that field operations are constant. For practical analysis the assumptions made in this thesis for the cost analysis are certainly reasonable. One possible alternate model would be that elements of **D** may be represented by multivariate polynomials. A comparison of the power series method with fraction-free methods could prove interesting under this assumption, and is left open for future research.

It is important to note that the conversion routine $\overline{Jpade}$, and also JPADE for *GCD* computations, dominates the time complexity of both the power series and fraction free methods, respectively. Any improvement in either of these routines would strongly affect the time complexity of the corresponding method. If we consider special integral domains, various improvements become possible. When **D** is the domain of integers, the Schönhage-Strassen fast integer multiplication routines may be used. If **D** is the set of multivariate polynomials over a field, we may use *FFT* techniques to accelerate polynomial multiplication. When **D** is an arbitrary integral domain, we may be able to combine Cabay and Choi's [3] quadratic Padé fraction algorithm with *Jpade* to obtain a divide and conquer algorithm. Another area to examine is the possible improvement of the construction of the power series solution in *Solve*. These areas merit future research consideration.

While the analyses of the power series method and the fraction-free methods provide accurate theoretical information, they do not determine the true practicality of the power series method. Since the complexities of both the power series method and the fraction-free method are so high (as are the constants associated with the number of multiplicative operations for the power series method), it is not clear which method is more efficient for practical problems. To determine their practical usefulness would require the implementation of both methods, and detailed timing comparisons would have to be made. This is left for future research.

A problem inherent in all the methods for solving linear systems of equations with dense components is that of large time complexities. As mentioned in Chapter 1, various sparse models have been proposed and studied (see Horowitz and Sahni [11] for a comprehensive survey). There is some preliminary indication that it may be possible to take advantage of certain types of sparsity using the power series method, but further investigation is needed.

Another interesting problem for future research is the investigation of generalizing evaluation-interpolation methods from linear systems with field polynomial coefficients to those with integral domain polynomial coefficients. This would complete the generalization of the three categories of solution methods mentioned in Chapter 1.

References

[1] E. H. Bareiss, "Computational Solutions of Matrix Problems Over an Integral Domain", *J. Inst. Maths. Applics.*, Vol. 10, 1972, pp. 68-104.

[2] W. S. Brown, "The Subresultant PRS Algorithm", *ACM Trans. Math. Software*, Vol. 4, No. 3, Sep 1978, pp. 237-249.

[3] S. Cabay and D. K. Choi, "Algebraic Computations of Scaled Padé Fractions", *SIAM J. Comput.*, Vol. 15, No. 1, Feb 1986, pp. 243-270.

[4] S. Cabay and B. Domzy, Systems of Linear Equations with Dense Univariate Polynomial Coefficients, *J. ACM*, Jul 1987.

[5] S. Cabay and P. Kossowski, Power Series Remainder Sequences and Padé Fractions Over an Integral Domain, *J. Symbolic Computation*, to appear.

[6] K. O. Geddes, "Symbolic Computation of Padé Approximants", *ACM Trans. Math. Software*, Vol. 5, No. 2, Jun 1979, pp. 218-233.

[7] W. M. Gentleman and S. C. Johnson, "Analysis of Algorithms, A Case Study: Determinants of Matrices with Polynomial Entries", *ACM Trans. Math. Software*, Vol. 2, No. 3, Sep 1976, pp. 232-241.

[8] W. B. Gragg, "The Padé Table and Its Relation to Certain Algorithms of Numerical Analysis", *SIAM Rev.*, Vol. 14, 1972, pp. 1-61.

[9] M. L. Griss, "The Algebraic Solution of Sparse Linear Systems via Minor Expansion", *ACM Trans. Math. Software*, Vol. 2, No. 1, Mar 1976, pp. 31-49.

[10] M. M. Higginson, *Methods for Systems of Polynomial Equations*, Department of Computing Science, The University of Alberta, 1974. M.Sc. Thesis.

[11] E. Horowitz and S. Sahni, "On Computing the Exact Determinant of Matrices with Polynomial Entries", *J. ACM*, Vol. 22, No. 1, Jan 1975, pp. 38-50.

[12] D. E. Knuth, *The Art of Computer Programming*, Addison Wesley, Reading, MA, 1969. Vol. 2.

[13] J. D. Lipson, Symbolic Methods for the Computer Solution of Linear Equations with Applications to Flowgraphs, *Proceedings of the 1968 Summer Institute on Symbolic Mathematical Computation*, June 1969. IBM Programming Lab. Rep. FSC69-0312.

[14] J. D. Lipson, Flowgraphs and Their Generating Functions, *Proceedings of the IFIP Congress 1974*, North Holland Publishing Co., Amsterdam, 1974, pp. 489-493.

[15] J. D. Lipson, *Elements of Algebra and Algebraic Computing*, Addison Wesley, Reading, MA, 1981.

[16] M. T. McClellan, "The Exact Solution of Systems of Linear Equations with Polynomial Coefficients", *J. ACM*, Vol. 20, No. 4, Oct 1973, pp. 563-588.

[17] R. T. Moenck and J. H. Carter, Approximate Algorithms to Derive Exact Solutions to Systems of Linear Equations, *Proceedings of Eurosam*, 1979, pp. 65-73.

[18] J. Smit, New Recursive Minor Expansion Algorithms, A Presentation in a Comparative Context, in *Lecture Notes in Computing Science*, G. Goos and J. Hartamis (ed.), Springer-Verlag, Berlin-Heidelberg-New York, 1977, pp. 74-87.