University of Alberta

Stable Rational Interpolation

by

R. Kacheong Yeung ©

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of **Doctor of Philosophy**.

Department of Computing Science

Edmonton, Alberta
Fall 1999

0-612-46952-2

University of Alberta

Library Release Form

Name of Author: R. Kacheong Yeung

Title of Thesis: Stable Rational Interpolation

Degree: Doctor of Philosophy

Year this Degree Granted: 1999

R. Kacheong Yeung
2204, 8210–111 Street
Edmonton, Alberta
Canada, T6G 2C7

Date: Sept. 29, 1999

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 **Stable Rational Interpolation** submitted by R. Kacheong Yeung in partial fulfillment of the requirements for the degree of **Doctor of Philosophy**.

S. Cabay (Supervisor)

W.W. Armstrong

J. Buchanan

R. Meleshko

Y.S. Wong

A. Bultheel (External Examiner)

Date: Sept. 17, 1996

To the memory of my mother,
Mack Wai Fong

Abstract

A new algorithm is given for interpolating data at $N + 1$ distinct points by a rational function. The algorithm is *fast*, requiring $O(N^2)$ operations, except for certain pathological cases. A floating-point error analysis is provided and is used to show that the algorithm is weakly stable. The algorithm is reliable in that it gives accurate results when the problem is well-conditioned and does not contain unattainable points and it identifies *a posteriori* all unattainable points in the data and alerts the user when the problem is ill-conditioned due to factors such as clustering of close points.

The performance of the algorithm is controlled by a user-specified stability tolerance τ. Experimental results are provided which support the given error analysis and which demonstrate that practically the point-wise error is bounded by $O(\tau\mu)$, where μ is the unit error.

Acknowledgements

I wish to express my sincere gratitude to my supervisor, Dr. Stan Cabay, for his supervision, guidance, and encouragement throughout the course of this study. I am grateful for all the lessons he taught me both in academics and life in general.

I wish to thank the members of my examining committee, Drs. W.W. Armstrong, J. Buchanan, A. Bultheel (Katholieke Universiteit Leuven, Belgium), R. Meleshko, and Y.S. Wong (Mathematical Sciences), for their constructive criticisms of this work. The thorough review of my thesis by Dr. A. Bultheel, Dr. R. Meleshko and the insightful comments from Dr. Y.S. Wong helped improve the quality of the thesis tremendously.

I would also like to thank my M.Sc. supervisor, Dr. S.M. Farouq Ali (Petroleum Engineering), who has always been an inspiration for me in the pursuit of excellence.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Let $\mathcal{D}$ denote the field of real numbers. For non-negative integers L and M, define $\mathcal{P}_L$ to be the set of polynomials of degree at most L with coefficients in $\mathcal{D}$ and $\mathcal{R}(L, M)$ to be the set of rational functions $r_{L,M}(z)$ that can be written in the form

$$r_{L,M}(z) = \frac{U(z)}{V(z)}, \tag{1.1}$$

with

$$U(z) = \sum_{i=0}^{L} U_i z^i \in \mathcal{P}_L, \quad V(z) = \sum_{i=0}^{M} V_i z^i \in \mathcal{P}_M, \tag{1.2}$$

where the denominator may not be the constant zero polynomial. By removing common factors from $U(z)$ and $V(z)$ in (1.1), we obtain the reduced form denoted by $r_{L,M}(z) = U''(z)/V''(z)$, where $U''(z)$ and $V''(z)$ are relatively prime. It is understood that the values of rational functions are always computed using the reduced form.

The rational interpolation problem is defined as follows.

Problem 1.1 (Nonlinear rational interpolation) *Let L and M be non-negative integers, and let $N = L+M$. Given $\{(z_j, f_j, g_j)\} \in \mathcal{D}^3$ for $j = 0, \ldots, N$, with* $\max\{|f_j|, |g_j|\} = 1$ *for all j and $z_i \neq z_j$ for all $i \neq j$, the problem of nonlinear rational interpolation is to determine an $r_{L,M}(z) \in \mathcal{R}(L, M)$ such that for all j*

$$\frac{U''(z_j)}{V''(z_j)} + \frac{f_j}{g_j} = 0. \tag{1.3}$$

Note that the normalization $\max\{|f_j|, |g_j|\} = 1$ in Problem 1.1 precludes the data $(f_j, g_j) = (0, 0)$, and it prescribes that the large interpolation values f_j/g_j be represented by correspondingly small values of g_j. Later, we introduce a similar normalization of $(U''(z), V''(z))$.

The problem of rational interpolation is an old one; it has its origin with Cauchy [19] when in 1821 he extended the Lagrangian formulation of polynomial interpolation at distinct points to rational functions. For this historical reason, the problem defined by Definition 1.1 is called the *Cauchy Interpolation* problem.

Rational functions are superior to polynomials for interpolating data because they can achieve more accurate approximations with the same amount of computation [46]. In addition, rational interpolants have a natural way of interpolating poles whereas polynomial interpolants do not.

Applications of rational interpolation range from simple root-finding of nonlinear equations [40, 44, 48] to the field of engineering where linear control systems are encountered [1, 13]. Rational interpolation also has a very rich theory. It is in close relation with a whole body of mathematical theory including the theory of orthogonal polynomials [24, 26], Padé approximants [5], continued fractions [33, 29, 60, 63], determinants [39], and the calculus of finite differences [33]. It also gives rise to many specialized matrices such as the Hankel and Toeplitz matrices [32, 39], the divided-difference matrix [34], Löwner matrix [11, 49. 59], and the generalized Vandermonde matrix [11].

The multiplication of (1.3) by the denominators g_j and $V''(z_j)$ leads to a different formulation of the problem.

Problem 1.2 (Linear rational interpolation) *Let L and M be non-negative integers, and let $N = L + M$. Given $\{(z_j, f_j, g_j)\} \in \mathcal{D}^3$ for $j = 0, \ldots, N$, with $\max\{|f_j|, |g_j|\} = 1$ for all j and $z_i \neq z_j$ for all $i \neq j$, the problem of linear rational interpolation is to determine $U(z) \in \mathcal{P}_L$ and $V(z) \in \mathcal{P}_M$ ($V(z) \neq 0$) such that for all j*

$$g_jU(z_j) + f_jV(z_j) = 0. \tag{1.4}$$

A solution $(U(z), V(z))$ satisfying (1.4) is said to be a **linear rational interpolant** *of type $[L, M]$. The reduced form $U''(z)/V''(z)$ obtained from $(U(z), V(z))$ is unique up to a scalar and is called the* **rational interpolant** *of type $[L, M]$.*

The value of $g_j = 0$ in (1.4) is permissible. In this case, $U''(z)/V''(z)$ would need to satisfy $V''(z_j) = 0$, $U''(z_j) \neq 0$. That is $U''(z)/V''(z)$ has a pole at $z = z_j$.

With (1.2), (1.4) becomes

$$\begin{pmatrix} g_0z_0^0 & \cdots & g_0z_0^L & f_0z_0^0 & \cdots & f_0z_0^M \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ g_Nz_N^0 & \cdots & g_Nz_N^L & f_Nz_N^0 & \cdots & f_Nz_N^M \end{pmatrix} \begin{pmatrix} U_0 \\ \vdots \\ U_L \\ V_0 \\ \vdots \\ V_M \end{pmatrix} = 0, \tag{1.5}$$

where the coefficient matrix is called the generalized Vandermonde matrix. Equation (1.5) is an under-determined linear homogeneous system of $N+1$ equations with $N+2$ unknowns. Thus, a nontrivial linear rational interpolant always exists.

It is quite clear that every solution of (1.3) is also a solution of (1.4). But the converse is not true. A solution of the linear rational interpolation problem may not be a solution of the nonlinear rational interpolation problem. If $(U(z), V(z))$ in (1.4) both have a factor of $(z - z_\sigma)$, where z_σ is one of the interpolation points, then upon forming the nonlinear rational function, such a factor cancels out and the reduced rational form may no longer interpolate at z_σ. This point is referred to as an unattainable point. In chapter 2, we illustrate that if one solution of Problem 1.2 has this property, then all solutions do. Thus, no (nonlinear) rational function can be found which interpolates at such point z_σ; the best we can do is to identify it.

The straightforward way to solve the linear rational interpolation problem is to directly solve the system of equations in (1.5). This system of equations can be solved numerically by the Gaussian elimination method which is known to be stable. Gaussian elimination requires $O(N^3)$ operations over $\mathcal{D}$.

Due to the special structure of (1.5), it is not surprising that faster $O(N^2)$ algorithms have been discovered. These fast algorithms [41, 42, 43, 53, 63] rely on constructing a solution recursively from its lower-degree type solutions until the final rational interpolant is attained. These lower-degree type solutions when arranged in order of degree form a table of solutions called the rational interpolation table. Each $[L, M]$ entry in the table corresponds to the rational interpolant of type $[L, M]$. This table is unique when the rational interpolant is normalized, traditionally by setting the leading coefficient of the denominator to 1.

These algorithms [41, 42, 43, 53, 63] construct the solution of type $[L, M]$ from its neighboring rational interpolants, which are in turn constructed from their neighboring rational interpolants. However, due to unattainable points mentioned above, some of these entries in the rational interpolation table can be identical. These identical entries appear in blocks which are called singular blocks, corresponding to the rank deficiency of the matrix in (1.5). When constructing the intermediate solutions involving identical rational interpolants, these algorithms breakdown.

To rectify the problem, relationships have been observed among entries at the border of singular blocks [9, 33], called singular rules. These singular rules methods require exact arithmetic because one needs to detect the exact size of the singular blocks. Detailed treatments of the singular blocks in rational interpolation (in an exact arithmetic setting) are given in Gutknecht [36, 34], van Barel and Bultheel [54, 55], Antoulas and Anderson [3], and Beckermann [6]. These recent studies [3, 6, 34, 36, 54, 55] are aimed at obtaining a theoretical algorithm to handle singular blocks.

When $\mathcal{D}$ is not the field of real numbers but rather some integral domain (e.g., rational numbers, integers) where exact arithmetic is possible, the injudicious use of these fast algorithms gives exponential growth of the size of intermediate results (see however Beckermann and Labahn [10]). In such cases the cost of these algorithm in terms of Boolean operations is exponential rather than $O(N^2)$. Consequently, the practicality of these algorithms is limited to domains $\mathcal{D}$ where operands do not grow and the cost of each operation is fixed (for example, finite fields).

On the other hand, with floating point numbers, the size of operations and therefore the cost of each operation are fixed. In this domain, counting the number of operations as the complexity of the algorithm is therefore appropriate. However, since floating point numbers are not exact, all of the above-mentioned algorithms suffer the consequences of roundoff errors. When there are singular blocks (numerically, ill-conditioned blocks) along the path of the computation, the algorithm becomes unstable [34]. Ill-conditioned blocks arise when the coefficient matrix in (1.5) becomes ill-conditioned.

Werner [60, 61] and Graves-Morris [31, 29, 30] addressed the problem of near-singular blocks by proposing a certain reordering of the data. But reordering is not considered to be inductive because one cannot add further data and proceed to higher degrees since the interpolation points may be reordered by the algorithm [33]. More importantly, there is no proof that such a reordering scheme leads to numerical stability. Indeed, Grave-Morris shows that even with the proposed reordering, the error bound still grows exponentially [30].

Although *fast* algorithms have been developed algebraically, their numerical counterparts have not yet been developed. Currently, there are no *fast* algorithms which are known to be numerically stable [36]. Cabay *et al.* [16], however, showed some preliminary experimental results that suggest the possibility of a *fast* numerically stable algorithm with the look-ahead approach. This look-ahead approach gave the early directions to this research and eventually led to a numerically stable algorithm.

In this thesis, we present the first numerically weakly stable algorithm for nonlinear rational interpolation. We show that the algorithm yields a good solution when the problem is well-conditioned (the stability of the problem will be made precise in Chapter 5). Whenever the problem is ill-conditioned–due to an ill-conditioned linear system (1.5) or the existence of nearly duplicate data—any solution is sensitive to small perturbations in the data. The algorithm recognizes these situations, as well as any unattainability in the data, by providing appropriate quantitative parameters. Hence the new algorithm is reliable.

The outline of this thesis is as follows: The characterization of the solutions of rational interpolation is given in Chapter 2. Conditions for the unique solution are given in this chapter. We also describe the nature of the multiple solutions where the solution is not unique. A set of insightful examples is designed to convey some of the important concepts of rational interpolation. In Chapter 3, we introduce the Linear Rational Interpolation System (LRIS), which is an important construct that enables us to apply the divide-and-conquer strategy. As a result, it leads to an $O(N^2)$ algorithm. This $O(N^2)$ algorithm is described in Chapter 4, first in algebraic and then in numerical form. Chapter 5 gives precise definitions of problem and algorithm stability. Chapter 6 gives a detailed error analysis of the numerical algorithm. We give the necessary error bounds of the crucial expressions that enable us in Chapters 7 and 8 to show that a continued-fraction representation of the solution of Problem 1.1, which is immediately obtainable from the output of the algorithm, gives small errors at the interpolation points whenever the interpolation problem is well-conditioned and does not contain unattainable points. We further show that the algorithm is weakly stable: that is whenever the problem is well-conditioned, the algorithm computes a solution that is close to the true solution. To provide evidence to support the error analysis, we report a variety of numerical experiments in Chapter 9. These experiments show that given a user-specified stability tolerance τ, in practice, the point-wise error is bounded by $O(\tau\mu)$, where μ is the unit error, as opposed to $O(\tau^2\mu)$ obtained theoretically. Finally, we make some concluding remarks in Chapter 10.

Chapter 2

Characterization of Solutions

As we already observed in Chapter 1, a nontrivial solution of Problem 1.2 always exists. In this chapter, we give conditions for its uniqueness (up to a scalar)[1]. In addition, when the solution is not unique, we describe the nature of the multiple solutions.

The solution of the nonlinear problem (Problem 1.1) is strongly connected to the linear one according to the following theorem (see [64] for a detailed proof).

Theorem 2.1 *There exists a rational function $r_{L,M} \in \mathcal{R}(L, M)$ satisfying (1.3) if and only if the reduced form of the $[L, M]$ linear rational solution $U''(z)/V''(z)$ satisfies (1.4).*

The theorem says that the solution of the nonlinear rational problem, if it exists, is the reduced pair $(U''(z), V''(z))$ which is obtained from the linear rational interpolant $(U(z), V(z))$. In this chapter, this equivalence is made more explicit.

The results in this chapter are not new; except for certain notions of singularity and their relationship to linear solutions, all the concepts presented here can be found in [33, 64]. The main contributions of this chapter are a series of insightful examples illustrating these concepts. The examples involve the interpolation of all or part of the data in Table 2.1 below.

j	0	1	2	3	4	5	6	7
z_j	-3	-2	-1	0	1	2	3	4
f_j	-3	-2	-3	0	1	2	1	2
g_j	1	1	1	1	1	1	1	1

Table 2.1: Data used to illustrate rational interpolation concepts.

Let

$$M_{L,M} = \begin{pmatrix} g_0 z_0^0 & \cdots & g_0 z_0^{L-1} & f_0 z_0^0 & \cdots & f_0 z_0^{M-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ g_{N-1} z_{N-1}^0 & \cdots & g_{N-1} z_{N-1}^{L-1} & f_{N-1} z_{N-1}^0 & \cdots & f_{N-1} z_{N-1}^{M-1} \end{pmatrix} \qquad (2.1)$$

[1] Henceforth, by unique solution we mean unique up to a scalar.

be the generalized Vandermonde matrix of type $[L, M]$ for the data $\{(z_j, f_j, g_j)\}_{j=0,\ldots,N-1}$.

Theorem 2.2 *A solution to the linear rational interpolation problem of type* $[L, M]$ *is unique if* $M_{L,M}$ *is nonsingular.*

Proof: Since $M_{L,M}$ is assumed to be nonsingular, a non-trivial solution $(P^{(1)}(z), Q^{(1)}(z))$ of type $[L-1, M]$ satisfying

$$M_{L,M} \begin{pmatrix} P_0^{(1)} \\ \vdots \\ P_{L-1}^{(1)} \\ Q_0^{(1)} \\ \vdots \\ Q_{M-1}^{(1)} \end{pmatrix} = -Q_M^{(1)} \begin{pmatrix} f_0 z_0^M \\ \vdots \\ f_{N-1} z_{N-1}^M \end{pmatrix}, \tag{2.2}$$

exists uniquely (up to a scalar), with $Q_m^{(1)} \neq 0$. Similarly, a non-trivial solution $(P^{(2)}(z), Q^{(2)}(z))$ of type $[L, M-1]$ satisfying

$$M_{L,M} \begin{pmatrix} P_0^{(2)} \\ \vdots \\ P_{L-1}^{(2)} \\ Q_0^{(2)} \\ \vdots \\ Q_{M-1}^{(2)} \end{pmatrix} = -P_L^{(2)} \begin{pmatrix} f_0 z_0^L \\ \vdots \\ f_{N-1} z_{N-1}^L \end{pmatrix}, \tag{2.3}$$

exists, with $P_L^{(2)} \neq 0$. The addition of an extra point, z_N, to (2.2) and (2.3) gives

$$\begin{pmatrix} g_0 z_0^0 & \cdots & g_0 z_0^{L-1} & f_0 z_0^0 & \cdots & f_0 z_0^M \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ g_N z_N^0 & \cdots & g_N z_N^{L-1} & f_N z_N^0 & \cdots & f_N z_N^M \end{pmatrix} \begin{pmatrix} P_0^{(1)} \\ \vdots \\ P_{L-1}^{(1)} \\ Q_0^{(1)} \\ \vdots \\ Q_M^{(1)} \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ g_N P^{(1)}(z_N) + f_N Q^{(1)}(z_N) \end{pmatrix}, \tag{2.4}$$

and

$$\begin{pmatrix} g_0 z_0^0 & \cdots & g_0 z_0^L & f_0 z_0^0 & \cdots & f_0 z_0^{M-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ g_N z_N^0 & \cdots & g_N z_N^L & f_N z_N^0 & \cdots & f_N z_N^{M-1} \end{pmatrix} \begin{pmatrix} P_0^{(2)} \\ \vdots \\ P_L^{(2)} \\ Q_0^{(2)} \\ \vdots \\ Q_{M-1}^{(2)} \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ g_N P^{(2)}(z_N) + f_N Q^{(2)}(z_N) \end{pmatrix}. \tag{2.5}$$

We now show that $(P^{(1)}(z), Q^{(1)}(z))$ and $(P^{(2)}(z), Q^{(2)}(z))$ cannot both interpolate at the point z_N. Otherwise,

$$g_j P^{(1)}(z_j) + f_j Q^{(1)}(z_j) = 0, \quad j = 0, \ldots, N \tag{2.6}$$

$$g_j P^{(2)}(z_j) + f_j Q^{(2)}(z_j) = 0, \quad j = 0, \ldots, N, \tag{2.7}$$

which gives

$$g_j(Q^{(2)}(z_j)P^{(1)}(z_j) - P^{(2)}(z_j)Q^{(1)}(z_j)) = 0, \quad j = 0, \ldots, N \tag{2.8}$$

$$f_j(Q^{(2)}(z_j)P^{(1)}(z_j) - P^{(2)}(z_j)Q^{(1)}(z_j)) = 0, \quad j = 0, \ldots, N. \tag{2.9}$$

Because $|f_j| + |g_j| \neq 0$, this means $Q^{(2)}(z)P^{(1)}(z) - P^{(2)}(z)Q^{(1)}(z)$, a polynomial of degree at most N, has $N+1$ zeroes. So, $Q^{(2)}(z)P^{(1)}(z) = P^{(2)}(z)Q^{(1)}(z)$. Since $\deg(Q^{(2)}(z)P^{(1)}(z)) \leq L + M - 2$, this can happen only if $P^{(2)}_L = Q^{(1)}_M = 0$, which is a contradiction.

Assume then that $(P^{(1)}(z), Q^{(1)}(z))$ does not interpolate at the point z_N (otherwise, $(P^{(2)}(z), Q^{(2)}(z))$ does not and we proceed in a similar fashion); i.e., assume $g_N P^{(1)}(z_N) + f_N Q^{(1)}(z_N) \neq 0$. Then the matrix on the left-hand-side of (2.4) must be nonsingular otherwise the solution $(P^{(1)}(z), Q^{(1)}(z))$ would not be unique. Consequently,

$$\begin{pmatrix} g_0 z_0^0 & \cdots & g_0 z_0^{L-1} & f_0 z_0^0 & \cdots & f_0 z_0^M \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ g_N z_N^0 & \cdots & g_N z_N^{L-1} & f_N z_N^0 & \cdots & f_N z_N^M \end{pmatrix} \begin{pmatrix} U_0 \\ \vdots \\ U_{L-1} \\ V_0 \\ \vdots \\ V_M \end{pmatrix} = -U_L \begin{pmatrix} f_0 z_0^L \\ \vdots \\ f_N z_N^L \end{pmatrix}. \tag{2.10}$$

has a unique solution $(U(z), V(z))$. □

We now illustrate Theorem 2.2 with an example.

Example 2.1 *Consider the problem of finding the linear rational interpolant of type* $[1, 1]$ *which interpolates the first three points of the data given in Table 2.1. The associated generalized Vandermonde matrix is*

$$M_{1,1} = \begin{pmatrix} 1 & -3 \\ 1 & -2 \end{pmatrix}. \tag{2.11}$$

Since this matrix is nonsingular, Theorem 2.2 tells us that the solution is unique. To verify this, all the solutions of (1.5), which in this case is

$$\begin{pmatrix} 1 & -3 & -3 & 9 \\ 1 & -2 & -2 & 4 \\ 1 & -1 & -3 & 3 \end{pmatrix} \begin{pmatrix} U_0 \\ U_1 \\ V_0 \\ V_1 \end{pmatrix} = 0, \tag{2.12}$$

are given by the one parameter family $(U_0, U_1, V_0, V_1) = (6\beta, 3\beta, 2\beta, \beta)$. *The solution, therefore, is unique (up to the scalar* β*). In polynomial form, the solution is* $(U(z), V(z)) = (3\beta z + 6\beta, \beta z + 2\beta) = (3\beta(z+2), \beta(z+2))$.

Note that $(z+2)$ is a common factor in the linear solution and that the reduced form of the linear solutions is always $U''(z)/V''(z) = 3$. By Theorem 2.1, this is the only candidate that can solve the nonlinear interpolation problem (1.1). Observe that this reduced form does not interpolate at $z = -2$. So, we can conclude that no nonlinear rational function of type $[1, 1]$ exists which interpolates at $z = -2$. Such a point is appropriately called an *unattainable point.*

Definition 2.1 *A point* z_σ *is an unattainable point with respect to* $[L, M]$ *if the reduced rational interpolant obtained from any possible solution of type* $[L, M]$ *to the linear Problem 1.2 does not interpolate at* z_σ.

Theorem 2.3 *If* $M_{L,M}$ *is nonsingular, then* z_σ*, with* $0 \leq \sigma \leq N$*, is an unattainable point with respect to* $[L, M]$ *if and only if* $|U(z_\sigma)| + |V(z_\sigma)| = 0$.

Proof: First, suppose that $|U(z_\sigma)|+|V(z_\sigma)| = 0$. It follows that both $U(z)$ and $V(z)$ contain a factor of $(z - z_\sigma)$ and so we can write $(U(z), V(z)) = (z - z_\sigma)(U^*(z), V^*(z))$. But then $U^*(z_\sigma)/V^*(z_\sigma)$ cannot interpolate at z_σ. Otherwise $((\alpha z+\beta)U^*(z), (\alpha z+\beta)V^*(z))$ is also a solution of the Problem 1.2 which contradicts the uniqueness of the result of Theorem 2.2.

Conversely, if $|U(z_\sigma)| + |V(z_\sigma)| \neq 0$, then at least one of $U(z)$ or $V(z)$ does not contain a factor of $(z - z_\sigma)$. So assume $V(z_\sigma) \neq 0$ (if not, we can proceed with $U(z_\sigma)$). It follows from $g_\sigma U(z_\sigma) + f_\sigma V(z_\sigma) = 0$ that

$$\frac{U(z_\sigma)}{V(z_\sigma)} + \frac{f_\sigma}{g_\sigma} = 0.$$

□

The next example illustrates the elusiveness of unattainability.

Example 2.2 *The problem is to obtain the linear rational interpolant of type* $[2, 1]$ *which interpolates the first four points of the data in Table 2.1. The associated generalized Vandermonde matrix is*

$$M_{2,1} = \begin{pmatrix} 1 & -3 & -3 \\ 1 & -2 & -2 \\ 1 & -1 & -3 \end{pmatrix} \qquad (2.13)$$

which is once again nonsingular. This tells us that (1.5), which for this problem is

$$\begin{pmatrix} 1 & -3 & 9 & -3 & 9 \\ 1 & -2 & 4 & -2 & 4 \\ 1 & -1 & 1 & -3 & 3 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} U_0 \\ U_1 \\ U_2 \\ V_0 \\ V_1 \end{pmatrix} = 0, \tag{2.14}$$

has a one-parameter family of solution. This family is $(U_0, U_1, U_2, V_0, V_1) = (0, -\beta, -\beta, \beta, \beta)$, *which gives* $(U(z), V(z)) = (-\beta z^2 - \beta z, \beta z + \beta) = (-\beta(z+1)z, \beta(z+1))$.

In the example above, the linear solutions have the common factor $(z+1)$. The reduced form is $U''(z)/V''(z) = -z$, which does not interpolate at $z = -1$. This point is therefore an unattainable point. Note that the unattainable point $z = -2$ of Example 2.1 disappears in Example 2.2 even though the data is the same except for the inclusion of an extra interpolation point. More importantly, the point $z = -1$ which is attainable in Example 2.1 has become unattainable in Example 2.2. These two examples illustrate the transient nature of unattainable points; that is, they may come and go with changes in the type of the rational interpolant of the same data. Furthermore, they are inherent to the problem. When computing a nonlinear rational interpolant, the best we can do is identify such points. This can be done only *a posteriori*.

Example 2.3 *The problem is to find the linear solution of type* $[3, 2]$ *which interpolates the first six points of the data in Table 2.1. The associated generalized Vandermonde matrix*

$$M_{3,2} = \begin{pmatrix} 1 & -3 & 9 & -3 & 9 \\ 1 & -2 & 4 & -2 & 4 \\ 1 & -1 & 1 & -3 & 3 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix} \tag{2.15}$$

is singular, and so the family of solutions will have at least two degrees of freedom. Solving

$$\begin{pmatrix} 1 & -3 & 9 & -27 & -3 & 9 & -27 \\ 1 & -2 & 4 & -8 & -2 & 4 & -8 \\ 1 & -1 & 1 & -1 & -3 & 3 & -3 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 8 & 2 & 4 & 8 \end{pmatrix} \begin{pmatrix} U_0 \\ U_1 \\ U_2 \\ U_3 \\ V_0 \\ V_1 \\ V_2 \end{pmatrix} = 0 \tag{2.16}$$

yields the general solution $(U_0, U_1, U_2, U_3, V_0, V_1, V_2) = (0, -\beta, -(\alpha+\beta), -\alpha, \beta, (\alpha+\beta), \alpha)$, *where* α, β *are arbitrary parameters, not both zero. In polynomial form, the linear solution is* $(U(z), V(z)) = (-\alpha z^3 - (\alpha+\beta)z^2 - \beta z, \alpha z^2 + (\alpha+\beta)z + \beta) = (-(\alpha z+\beta)(z+1)z, (\alpha z+\beta)(z+1))$.

In Example 2.3, there are two common factors $c(z) = (z+1)$ and $d(z) = (\alpha z + \beta)$ in the general solution which play very different roles. The polynomial $d(z)$ prescribes all the parameters in the family of linear rational interpolants satisfying (2.16). It is not essential to the solution; its removal (i.e., assign $d(z)=1$) still gives a polynomial pair $(U'(z), V'(z)) = (-zc(z), c(z)) = (-z(z+1), (z+1))$ which solves the linear interpolation Problem 1.2 (i.e., (2.16)). On the other hand, the polynomial $c(z)$ is essential to the solution; its removal from $(U'(z), V'(z))$ yields $(U''(z), V''(z)) = (U'(z)/c(z), V'(z)/c(z)) = (-z, 1)$ which no longer solves the linear interpolation Problem 1.2. In addition, the reduced rational function $U''(z)/V''(z)$ does not solve the nonlinear Problem 1.1 because it does not interpolate precisely at the zero of $c(z)$. The zero of $c(z)$ is an unattainable point.

The above observations hold in general and are summarized in the following theorem (see [33, 64, 45]).

Theorem 2.4 *The general solution $(U(z), V(z)) \in \mathcal{P}_L \times \mathcal{P}_M$ of Problem 1.2 can be expressed as*

$$(U(z), V(z)) = (U''(z)c(z)d(z), V''(z)c(z)d(z)) \tag{2.17}$$

where $U''(z)$ and $V''(z)$ are relatively prime polynomials, $c(z)$ is some unique polynomial that divides $\prod_{j=0}^{N}(z - z_j)$ and $d(z)$ is an arbitrary polynomial.

Proof: Let $(U^{(1)}(z), V^{(1)}(z))$ and $(U^{(2)}(z), V^{(2)}(z))$ be solutions of Problem 1.2. Then, for $j = 0, \ldots, N$,

$$g_j U^{(1)}(z_j) + f_j V^{(1)}(z_j) = 0, \tag{2.18}$$

$$g_j U^{(2)}(z_j) + f_j V^{(2)}(z_j) = 0. \tag{2.19}$$

Consequently,

$$U^{(1)}(z_j)V^{(2)}(z_j) - V^{(1)}(z_j)U^{(2)}(z_j) = 0, \quad j = 0, \ldots, N. \tag{2.20}$$

because $|f_j| + |g_j| \neq 0$. There are $N+1$ zeroes, but the degree of this combined polynomial is at most $L + M = N$. Therefore, $U^{(1)}(z)V^{(2)}(z) - V^{(1)}(z)U^{(2)}(z) = 0$. Now, assume $V^{(1,2)}(z) \neq 0$ (if not, we can proceed with $U^{(1,2)}(z)$). Thus, there exists a pair $(U''(z), V''(z))$ such that

$$\frac{U''(z)}{V''(z)} = \frac{U^{(1)}(z)}{V^{(1)}(z)} = \frac{U^{(2)}(z)}{V^{(2)}(z)}. \tag{2.21}$$

Therefore, every solution has the same reduced form $U''(z)/V''(z)$. If this reduced pair $(U''(z), V''(z))$ does not interpolate all the given points, there must be a polynomial $c(z)$

of minimal degree for which $(U''(z)c(z),\ V''(z)c(z))$ satisfies the interpolation conditions

$$c(z_j)(g_jU''(z_j) + f_jV''(z_j)) = 0, \quad j = 0, \ldots, N. \tag{2.22}$$

So, every solution pair must contain the *minimal (unique)* pair $(U'(z),\ V'(z)) = (U''(z)c(z),\ V''(z)c(z))$ which solves Problem 1.2. Hence, the general solution to the linear rational interpolation problem will have the form of $(U''(z)c(z)d(z), V''(z)c(z)d(z))$, where $d(z)$ is an arbitrary polynomial with $\deg(d(z)) \leq \min\{L - \deg(U'(z)), M - \deg(V'(z))\}$[2]. □

Remark 2.1 *A solution $(U(z), V(z))$ of type $[L, M]$ for Problem 1.2 may satisfy $|U(z_\sigma)| + |V(z_\sigma)| = 0$, for some σ, $0 \leq \sigma \leq N$, and yet z_σ can be an attainable point. This will happen if $z - z_\sigma$ divides $d(z)$ in Theorem 2.4. So the requirement that $M_{L,M}$ be nonsingular in Theorem 2.3 is necessary.*

Definition 2.2 *The minimal solution of type $[L, M]$ of Problem 1.2 is the linear pair $(U'(z), V'(z)) = (U''(z)c(z),\ V''(z)c(z))$ defined in Theorem 2.4.*

The *minimal solution* is the lowest degree pair which solves the linear interpolation problem. This solution is special in the family of solutions of Problem 1.2 in that it provides the basis for all members in that family. It consists of the reduced part $(U''(z), V''(z))$ combined with the polynomial $c(z)$ whose zeroes are the unattainable points.

Recall in Problem 1.2 that the reduced rational function $r_{L,M} = U''(z)/V''(z)$ of type $[L, M]$ obtained from its linear solution is called the rational interpolant of type $[L, M]$. The elements $r_{L,M}$ for different values of L and M can be arranged in a table called the *rational interpolation table.* (Note that $U''(z)/V''(z)$ exists uniquely, but when $\deg(c(z)) > 0$ it does not solve Problem 1.1 and $(U''(z), V''(z))$ does not solve Problem 1.2). The rational interpolation table is unique[3].

Continuing with Example 2.3, we see that the nonunique linear solution is the same as the solution of Example 2.2 except that it has one extra free parameter α. The reduced form is identical in both examples. In fact, all the rational interpolants neighboring the one of type [3, 2] (see Table 2.2) have the same reduced form $-z$. In the rational interpolation table, these identical entries appear in a square block as illustrated in Fig 2.1. This structure is referred to as a block [22, 34].

[2] The degree of $c(z)d(z)$ is called the *defect* ∂ by Gutknecht [33]; $\partial = \min\{L - \deg(U''(z)), M - \deg(V''(z))\}$. Gutknecht then used the term *degenerate* to describe a rational interpolant $U''(z)/V''(z)$ that has a positive *defect* (i.e., $\deg(c(z)d(z)) > 0$). But this terminology does not have universal acceptance. For example, Claessens [22] uses the term *degenerate* to refer to a rational interpolant $U''(z)/V''(z)$ that has unattainable points (i.e., $\deg(c(z)) > 0$).

[3] Recall that by uniqueness, we mean uniqueness up to a scalar.

Entry	$U(z)$	$V(z)$
$(2,1)$	$-\beta(z+1)z$	$\beta(z+1)$
$(2,2)$	$-\beta(z+1)z$	$\beta(0z^2+z+1)$
$(2,3)$	$-\beta(z+1)z$	$\beta(0z^3+0z^2+z+1)$
$(3,1)$	$-\beta(0z^2+z+1)z$	$\beta(z+1)$
$(3,2)$	$-(\alpha z+\beta)(z+1)z$	$(\alpha z+\beta)(z+1)$
$(3,3)$	$-\beta(z-3)(z+1)z$	$(0z+\beta)(z-3)(z+1)$
$(4,1)$	$-\beta(0z^3+0z^2+z+1)z$	$\beta(z+1)$
$(4,2)$	$-(0z+\beta)(z-3)(z+1)z$	$\beta(z-3)(z+1)$
$(4,3)$	$-\beta(z-3)(z-4)(z+1)z$	$\beta(z-3)(z-4)(z+1)$

Table 2.2: Solutions of entries in a block.

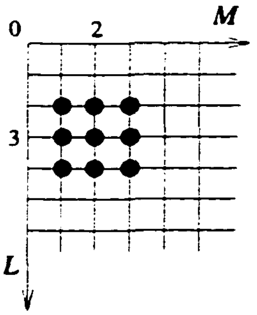

Figure 2.1: An illustration of a singular block.

Note that the solutions for type [3, 3], [4, 2] and [4, 3] are unique (up to a scalar). However, their corresponding generalized Vandermonde matrices are singular. For example,

$$M_{3,3} = \begin{pmatrix} 1 & -3 & 9 & -3 & 9 & -27 \\ 1 & -2 & 4 & -2 & 4 & -8 \\ 1 & -1 & 1 & -3 & 3 & -3 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 2 & 4 & 8 \end{pmatrix}, \tag{2.23}$$

which is singular. Thus, Theorem 2.2 goes only one way (i.e., the matrix $M_{L,M}$ may be singular and yet the solution will be unique). Now, if we were to highlight these singular $M_{L,M}$ within a block structure (in Fig. 2.2), we would notice that they are located together in the lower right hand corner. The commonality of the location of a singular $M_{L,M}$ in the rational interpolation table is that for each entry $[i,j]$ its three immediate neighbors to the left, top and also the upper left (i.e., $[i,j-1]$, $[i-1,j]$ and $[i-1,j-1]$, respectively) are also within the same block. With this observation, a block structure then contains at least one singular $M_{L,M}$ (i.e., the block structure of a 2×2 square block). For a detailed description of the structure of singular blocks in the rational interpolation table, see Claessens [22] and

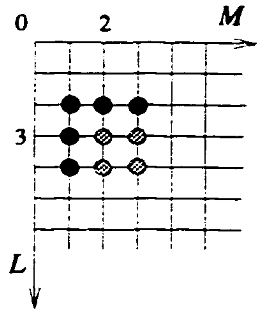

Figure 2.2: An illustration of singular $M_{L,M}$ within a block structure.

Gutknecht [33, 34] (see also Theorem 9.1 in Chapter 9). Since a block contains at least one singular $M_{L,M}$, the use of the modifier "singular" is appropriate[4].

Remark 2.2 *Note that the reduced form of type* $[1,0]$ *is also* $-z$*, but it does not belong to the singular block in Fig. 2.1 (or Fig. 2.2). This is unlike the case of Padé table, where all identical entries are always arranged in square blocks [28].*

In the final example below, we will show that a block can be formed by two square blocks overlapping one another (i.e., a union of square blocks).

Example 2.4 (Union of blocks) *The block structure of the data in Table 2.3 is illustrated in Fig. 2.3. Note that the only difference in the data in Table 2.1 and Table 2.3 is that* $f_7 = 2$ *and* $f_7 = 4$*, respectively. The solutions in a block structure are tabulated in Table 2.4. Because of this change in the data, the solution of type* $[4,3]$ *is no longer unique and has the general form* $(-(\alpha z+\beta)(z-3)(z+1)z, (\alpha z+\beta)(z-3)(z+1))$*. The block structure in this example is a union of two blocks as depicted in Fig. 2.3. In the figure, the union of two blocks is highlighted in the right. All of the elements in the block structure have a rational interpolant of* $-z$*. The entries with singular* $M_{L,M}$ *are shaded as opposed to the nonsingular ones which are highlighted in black. Notice that we do not know the function*

j	0	1	2	3	4	5	6	7
z_j	-3	-2	-1	0	1	2	3	4
f_j	-3	-2	-3	0	1	2	1	4
g_j	1	1	1	1	1	1	1	1

Table 2.3: Data used to illustrate a union of two square blocks.

[4] Prior to the definition of $M_{L,M}$ in this study, the relationship between the block structure and its corresponding singularity of the matrices was not clear. For example, Gutknecht [34] refers to blocks as "so-called singular blocks" without clarification.

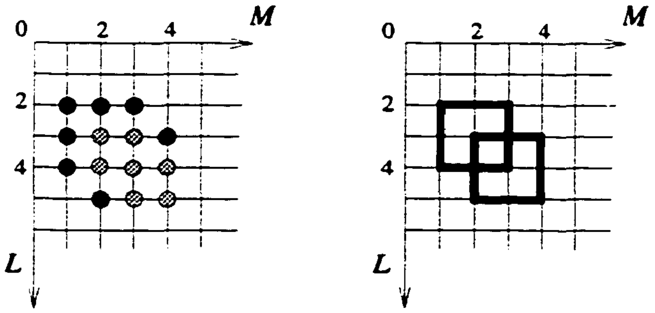

Figure 2.3: A rational table with a union of two squares.

Entry	$U(z)$	$V(z)$
(2, 1)	$-\beta(z+1)z$	$\beta(z+1)$
(2, 2)	$-\beta(z+1)z$	$\beta(0z^2+z+1)$
(2, 3)	$-\beta(z+1)z$	$\beta(0z^3+0z^2+z+1)$
(3, 1)	$-\beta(0z^2+z+1)z$	$\beta(z+1)$
(3, 2)	$-(\alpha z+\beta)(z+1)z$	$(\alpha z+\beta)(z+1)$
(3, 3)	$-\beta(z-3)(z+1)z$	$(0z+\beta)(z-3)(z+1)$
(3, 4)	$-\beta(z-3)(z+1)z$	$(0z^2+0z+\beta)(z-3)(z+1)$
(4, 1)	$-\beta(0z^3+0z^2+z+1)z$	$\beta(z+1)$
(4, 2)	$-(0z+\beta)(z-3)(z+1)z$	$\beta(z-3)(z+1)$
(4, 3)	$-(\alpha z+\beta)(z-3)(z+1)z$	$(\alpha z+\beta)(z-3)(z+1)$
(4, 4)	$-\beta(z-z_8)(z-3)(z+1)z$	$(0z+\beta)(z-z_8)(z-3)(z+1)$
(5, 2)	$-(0z^2+0z+\beta)(z-3)(z+1)z$	$\beta(z-3)(z+1)$
(5, 3)	$-(0z+\beta)(z-z_8)(z-3)(z+1)z$	$\beta(z-z_8)(z-3)(z+1)$
(5, 4)	$-\beta(z-z_9)(z-z_8)(z-3)(z+1)z$	$\beta(z-z_9)(z-z_8)(z-3)(z+1)$

Table 2.4: Solutions of entries in a block structure.

values of the last two points (z_8 and z_9). So, the solutions given in Table 2.4 for entries [4, 4], [5, 3] *and* [5, 4] *are possible linear solutions. But these solutions interpolate the data regardless of function values. Since all possible linear solutions must have the same reduced form (Theorem 2.4), such possible solutions are sufficient to obtain their reduced forms.*

For a more general treatment of the block structure of the rational interpolation table, see Claessens [22] and Gutknecht [33, 34].

Chapter 3

The Linear Rational Interpolation System

A linear rational interpolant can be obtained by solving the system of linear equations (1.5) directly. For example, one can use the Gaussian elimination method which requires $O(N^3)$ operations in $\mathcal{D}$. Furthermore, when the interpolation problem is treated as a least squares problem, a class of least sqares methods can be used (see for example [14, 55, 56, 57]).

Because of the special structure of the matrix in (1.5), however, a number of recursive algorithms have been developed, which by taking advantage of this structure require only $O(N^2)$ operations. These recursive algorithms follow a path in the rational interpolation table connected with a sequence of data points, possibly reordered. They can be classified according to how they respond to singular blocks along the path.

The first class of algorithms gives no consideration to singular blocks [41, 43, 42, 47, 52, 63]. A good summary of these can be found in [31] and [5, Chap. 7]. The development of these algorithms implicitly assumes a normal rational interpolation table (i.e., a rational interpolation table in which all entries are distinct) in order to proceed from one entry on the path to the next. Thus, when a singular block is encountered, they break down.

The second class of algorithms accommodates singular blocks in one of two ways. They either reorder the interpolation points so as to remove singular blocks (except possibly at the end of the path) [29, 30, 31, 60], or they provide a mechanism (called singular rules [34]) for detecting singular blocks and a procedure for skipping over them [9, 16, 33, 34, 36].

The more recent third class of algorithms proceeds directly through singular blocks [2, 54, 55]. This class differs from the other two in that these algorithms iterate by successively increasing not the degree of the interpolants but the number of interpolation points and for each such increase by recursively constructing a basis for all interpolants (along a path) independent of degrees. The interpolant of the appropriate minimal degrees satisfying

$L + M \leq N$ is easily determined from this basis.

The main strength of the algorithms in the third class is that they not only recognize singular blocks but also give the entire family of solutions for an entry within singular blocks. Their greatest weakness is the sensitivity of this basis within singular blocks on small perturbations of the interpolation data. That is, the algorithms in this third class are numerically unstable.

This lack of numerical stability prevails with the algorithms in the other two classes as well. There are experimental results [8, 16, 61] which lead to the discussion of numerical stability. But these studies use the term stability loosely and do not deal with it formally. Until this thesis, no algorithms have been shown to be numerically stable [36].

Our algorithm is from the second class; a mechanism is provided for skipping over singular blocks. We do this by collecting two adjacent linear rational interpolants in a matrix called a Linear Rational Interpolation System (LRIS). A simple rule applied to a LRIS detects singular entries.

A precise definition of the LRIS, together with some of its properties, are given in §3.1. The relationship that we use in Chapter 4 to build a fast algorithm is described in terms of two LRIS's. This description is given in §3.2, where we show that a particular LRIS can be constructed from two smaller ones. The idea here is to deploy the power of the divide-and-conquer strategy. Lastly, in §3.3, we show that the successive application of the divide-and-conquer strategy results in a fast iterative algorithm. The algorithm proceeds along a staircase in the rational interpolation table bypassing singular blocks in its path. If there should be no singular blocks encountered, this method reduces to Werner's Algorithm [60].

3.1 Linear Rational Interpolation System (LRIS)

Assume without loss of generality that $L \geq M$. If $L < M$, we interpolate instead the points $\{(z_j, g_j, f_j)\}_{j=0,\dots,N}$.

Definition 3.1 *Given $N+1$ points $\{(z_j, f_j, g_j)\}_{j=0,\dots,N}$ and two nonnegative integers L and M such that $L + M = N$, the Linear Rational Interpolation System (LRIS) of degree type $[L, M]$ is*

$$S(z) = \begin{pmatrix} U(z) & P^*(z) \\ V(z) & Q^*(z) \end{pmatrix}, \tag{3.1}$$

with

$$\begin{pmatrix} P^*(z) \\ Q^*(z) \end{pmatrix} = \begin{pmatrix} (z - z_N)P(z) \\ (z - z_N)Q(z) \end{pmatrix}, \tag{3.2}$$

if it satisfies the following conditions:

- *Degree Condition:* $U(z) \in \mathcal{P}_L$, $V(z) \in \mathcal{P}_M$ *and either*

  a) $P(z) \in \mathcal{P}_{L-1}$, $Q(z) \in \mathcal{P}_M$, *or*
  b) $P(z) \in \mathcal{P}_L$, $Q(z) \in \mathcal{P}_{M-1}$;

- *Nonsingularity Condition:* $\det(S(z)) \neq 0$;

- *Interpolation Condition:* $(\,g_j \quad f_j\,)\,S(z_j) = (\,0 \quad 0\,)$, $\quad j = 0, \ldots, N.$

Remark 3.1 *In Definition 3.1,* $(U(z), V(z))$ *is the linear rational interpolant for the points* z_j, $j = 0, \ldots, N$, *whereas* $(P(z), Q(z))$ *is the linear rational interpolants for the points* z_j, $j = 0, \ldots, N-1$.

Define

$$t_{i,j}(z) = \prod_{l=i}^{j}(z - z_l). \tag{3.3}$$

The determinant of the LRIS $S(z)$, which is extremely important in the subsequent development, can be expressed in terms of (3.3).

Lemma 3.1

$$\det(S(z)) = \Gamma\, t_{0,N}(z), \tag{3.4}$$

where

$$t_{0,N}(z) = \prod_{l=0}^{N}(z - z_l)$$

and Γ *is a constant.*

Proof: We give a proof for degree condition (a); the proof for degree condition (b) is similar. From (3.1)

$$\det(S((z)) = (z - z_N)(U(z)Q(z) - V(z)P(z)), \tag{3.5}$$

where the degrees of $(U(z), V(z))$ are at most L and M, respectively, and the degrees of $(P(z), Q(z))$ are at most $L - 1$ and M, respectively. So, $\deg(\det(S(z))) \leq N + 1$.

But, from the interpolation condition,

$$g_j U(z_j) + f_j V(z_j) = 0, \quad j = 0, \ldots, N, \tag{3.6}$$

$$(z_j - z_N)(g_j P(z_j) + f_j Q(z_j)) = 0, \quad j = 0, \ldots, N. \tag{3.7}$$

Therefore,

$$\det(S(z_j)) = (z - z_N)(U(z_j)Q(z_j) - V(z_j)P(z_j)) = 0, \quad j = 0, \ldots, N, \tag{3.8}$$

because $|f_j| + |g_j| \neq 0$. Since $\deg(\det(S(z))) \leq N + 1$, then

$$\det(S(z)) = \Gamma\, t_{0,N}(z), \tag{3.9}$$

where Γ is the leading coefficient of $U(z)Q(z) - V(z)P(z)$. □

Theorem 3.1 *A LRIS of type $[L, M]$ exists if and only if $M_{L,M}$ is nonsingular.*

Proof: First suppose that $M_{L,M}$ is singular. It then follows that there exists a solution $(X(z), Y(z))$ of type $[L-1, M-1]$ interpolating not just $N-1$ but the first N points. So $(X(z), Y(z))$ is a linear rational interpolant of types $[L-1, M]$ and $[L, M-1]$ as well. But $(z - z_N)(X(z), Y(z))$ of type $[L, M]$ interpolates the first $N+1$ points. From Theorem 2.4, all solutions of Problem 1.2 of type $[L-1, M-1]$, $[L, M-1]$, $[L-1, M]$ and $[L, M]$ must therefore have the same reduced form. Consequently, no LRIS $S(z)$ of type $[L, M]$ (with $\det(S(z)) \neq 0$) can be formed.

Conversely, if $M_{L,M}$ is nonsingular, from the proof of Theorem 2.2, we know that there exist solutions $(P^{(1)}(z), Q^{(1)}(z))$ with $Q^{(1)}_M \neq 0$ of type $[L-1, M]$ and $(P^{(2)}(z), Q^{(2)}(z))$ with $P^{(2)}_L \neq 0$ of type $[L, M-1]$ with $Q^{(2)}(z)P^{(1)}(z) \neq P^{(2)}(z)Q^{(1)}(z)$. In addition, from Theorem 2.2 that there exists a unique solution $(U(z), V(z))$ of type $[L, M]$ with $\deg(U(z)) = L$, or $\deg(V(z)) = M$, or both. Assume without loss of generality that $\deg(U(z)) = L$. Then for

$$S(z) = \begin{pmatrix} U(z) & (z - z_N)P^{(1)}(z) \\ V(z) & (z - z_N)Q^{(1)}(z) \end{pmatrix} \tag{3.10}$$

the leading coefficient of $\det(S(z)) = (z - z_N)(U(z)Q^{(1)}(z) - V(z)P^{(1)}(z))$ is $U_L Q^{(1)}_M \neq 0$. So, the Nonsingularity Condition for LRIS is satisfied. It is obvious that $S(z)$ also satisfies the Degree and Interpolation Conditions. □

Definition 3.2 *An entry $[L, M]$ in the rational interpolation table is a nonsingular entry if the corresponding $M_{L,M}$ is nonsingular, otherwise it is a singular entry.*

Remark 3.2 *When $M_{L,M}$ is nonsingular, Theorem 3.1 tells us that at least one LRIS $S(z)$ of type $[L, M]$ satisfying one of the degree conditions in Definition 3.1 exists, but not necessarily both. When a LRIS $S(z)$ exists, from Theorem 2.2 and its proof, it is unique (up to a scalar multiple of its columns).*

The existence of a LRIS for a given data set does not imply that a rational interpolant satisfying (1.3) exists, since there could be unattainable points in the system. We illustrate this with an example.

Example 3.1 *The LRIS of type* $[1,1]$ *from the data points* $\{(0,-1,1),(1,-2,1),(2,-1,1)\}$ *is*

$$S(z) = \begin{pmatrix} U(z) & (z-z_2)P(z) \\ V(z) & (z-z_2)Q(z) \end{pmatrix} = \begin{pmatrix} z-1 & (z-2)(z+1) \\ z-1 & z-2 \end{pmatrix} \tag{3.11}$$

because

$$(\,g_j \quad f_j\,)\,S(z_j) = 0, \qquad j = 0,\ldots,2. \tag{3.12}$$

and $\det(S(z)) = -z(z-1)(z-2) \neq 0$. *But from Theorem 2.3 and Theorem 3.1,* $z=1$ *is unattainable for* $(U(z),V(z))$ *since* $|U(1)| + |V(1)| = 0$.

It is also true that $(P(z),Q(z))$ can have unattainable points. But, $(U(z),V(z))$ and $(P(z),Q(z))$ cannot both have the same unattainable point z_σ, since $\det(S(z))$ would have a factor $(z-z_\sigma)^2$, contradicting Lemma 3.1.

To obtain the rational interpolant $U(z)/V(z)$ of the type $[L,M]$, only the first column of $S(z)$ is required. Thus, solving an additional system to obtain $(P(z),Q(z))$ is extra computation and does not seem to contribute to the overall solution. However, as we will see in the next section, this structure of a LRIS allows us to deploy the divide-and-conquer strategy, which then leads to an efficient algorithm for its computation.

3.2 Divide-and-Conquer

Now that a LRIS is defined, we would like to show that it can be written as a product $S(z) = s(z)\hat{s}(z)$ of LRIS's of lower degree types. This is the essence of the basic step in a divide-and-conquer strategy. Once we have established this basic step, we can then further split these into yet smaller LRIS's, and so on.

Let us study this basic step in detail. Given $[L,M]$, choose one of two diagonal staircases through $[L,M]$ along which computation will proceed as illustrated in Fig. 3.1. Note that because $M \leq L$ so $[L,M]$ is in the lower triangular region. Once a path is chosen, say Path A, then the linear rational interpolant $(u(z),v(z))$ of type $[l,m]$ for the subproblem must lie on Path A.

It should be noted that using a staircase to arrive at a solution is not novel. Indeed, there are numerous algorithms [9, 31, 43, 61, 64] that use a staircase or a diagonal path [16, 33] to arrive at a solution.

Let $[l,m]$ with $l \leq L$ and $m \leq M$ be a nonsingular entry along one of the two staircases that pass through $[L,M]$. Let

$$s(z) = \begin{pmatrix} u(z) & p^*(z) \\ v(z) & q^*(z) \end{pmatrix} \tag{3.13}$$

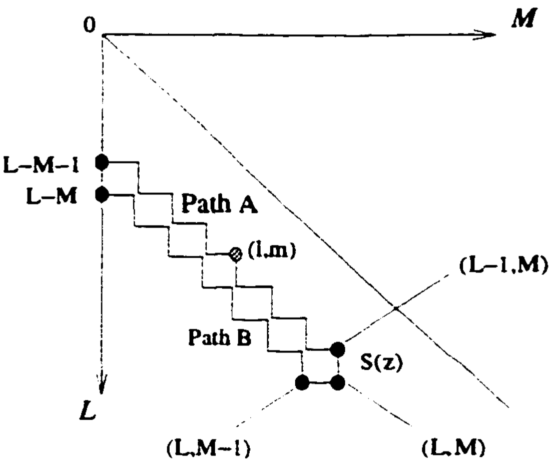

Figure 3.1: A rational table of a given set of data.

be the LRIS of type $[l, m]$, which interpolates not all the $N+1$ points but rather only the first $n+1$ points, i.e.,

$$(\, g_j \quad f_j \,)\, s(z_j) = (\, 0 \quad 0 \,), \quad j = 0, \ldots, n, \tag{3.14}$$

where $n = l + m$. Associated with $s(z)$, define the ***residual*** to be the pair $(\, w_j \quad r_j \,)$, $j = n+1, \ldots, N$, where

$$(\, w_j \quad r_j \,) = (\, g_j \quad f_j \,)\, s(z_j), \quad j = n+1, \ldots, N. \tag{3.15}$$

Similar to Lemma 3.1, we have

$$\begin{aligned} \det(s(z)) &= \gamma\, t_{0,n}(z) \\ &= \gamma \prod_{l=0}^{n} (z - z_l), \end{aligned} \tag{3.16}$$

where γ is the leading coefficient of $u(z)q(z) - v(z)p(z)$.

Remark 3.3 *Note that the residual $(\, w_j \quad r_j \,)$, $j = n+1, \ldots, N$, resembles the original data in that $|w_j| + |r_j| \neq 0$, for $j = n+1, \ldots, N$. This fact is obvious when we multiply both sides of (3.15) by $s^{adj}(z_j)$, resulting in*

$$(\, w_j \quad r_j \,)\, s^{adj}(z_j) = (\, g_j \quad f_j \,)\, \gamma\, t_{0,n}(z_j), \quad j = n+1, \ldots, N. \tag{3.17}$$

Since $|g_j| + |f_j| \neq 0$ and $t_{0,n}(z_j) \neq 0$, for $j = n+1, \ldots, N$, then $|w_j| + |r_j| \neq 0$ for $j = n+1, \ldots, N$.

Let

$$\hat{s}(z) = \begin{pmatrix} \hat{u}(z) & \hat{p}^*(z) \\ \hat{v}(z) & \hat{q}^*(z) \end{pmatrix} \tag{3.18}$$

be the LRIS of type $[\hat{l}, \hat{m}]$, which interpolates the residual $(w_j \quad r_j)$, $j = n+1, \ldots, N$, i.e.,

$$(w_j \quad r_j)\, \hat{s}(z_j) = (0 \quad 0), \quad j = n+1, \ldots, N, \tag{3.19}$$

where $N - n - 1 = \hat{l} + \hat{m}$ because of the degree condition.

Similarly, we have

$$\begin{aligned} \det(\hat{s}(z)) &= \hat{\gamma}\, t_{n+1,N}(z) \\ &= \hat{\gamma} \prod_{l=n+1}^{N} (z - z_l), \end{aligned} \tag{3.20}$$

where $\hat{\gamma}$ is the leading coefficient of $\hat{u}(z)\hat{q}(z) - \hat{v}(z)\hat{p}(z)$.

To express $S(z)$ in terms of $s(z)$ and $\hat{s}(z)$, four cases need to be considered. These cases are governed by the orientations of $S(z)$ and $s(z)$ as in Figure 3.2. Let

$$b = L - l. \tag{3.21}$$

In Fig. 3.2, these are the three possibilities, viz., $M - m = b - 1$, $M - m = b$ and $M - m = b + 1$.

Theorem 3.2 *Let $s(z)$ of type $[l, m]$ be a LRIS for the data points $\{(z_j, f_j, g_j)\}_{j=0,\ldots,l+m=n}$ with residuals $\{(z_j, r_j, w_j)\}_{j=n+1,\ldots,N}$. If $\hat{s}(z)$ is a LRIS of type $[\hat{l}, \hat{m}]$ for the residuals $\{(z_j, r_j, w_j)\}_{j=n+1,\ldots,N}$, then $S(z) = s(z)\hat{s}(z)$ is a LRIS of type $[L, M]$ for the data points $\{(z_j, f_j, g_j)\}_{j=0,\ldots,N}$, where*

$$[\hat{l}, \hat{m}] = \begin{cases} [b-1, b-1] & \text{if } M - m = b - 1, \\ [b, b-1] & \text{if } M - m = b, \\ [b, b] & \text{if } M - m = b + 1, \end{cases} \tag{3.22}$$

and lies along the staircase on and immediately below the diagonal.

Proof: We will prove the case $M - m = b - 1$ only; the proofs of the other two cases are similar.

We are given that $\hat{s}(z)$ is a LRIS of type $[b-1, b-1]$ interpolating the residuals $\{(z_j, r_j, w_j)\}_{j=n+1,\ldots,N}$. Then component-wise we have

$$\deg(\hat{s}(z)) \leq \begin{pmatrix} b-1 & b \\ b-1 & b-1 \end{pmatrix}, \tag{3.23}$$

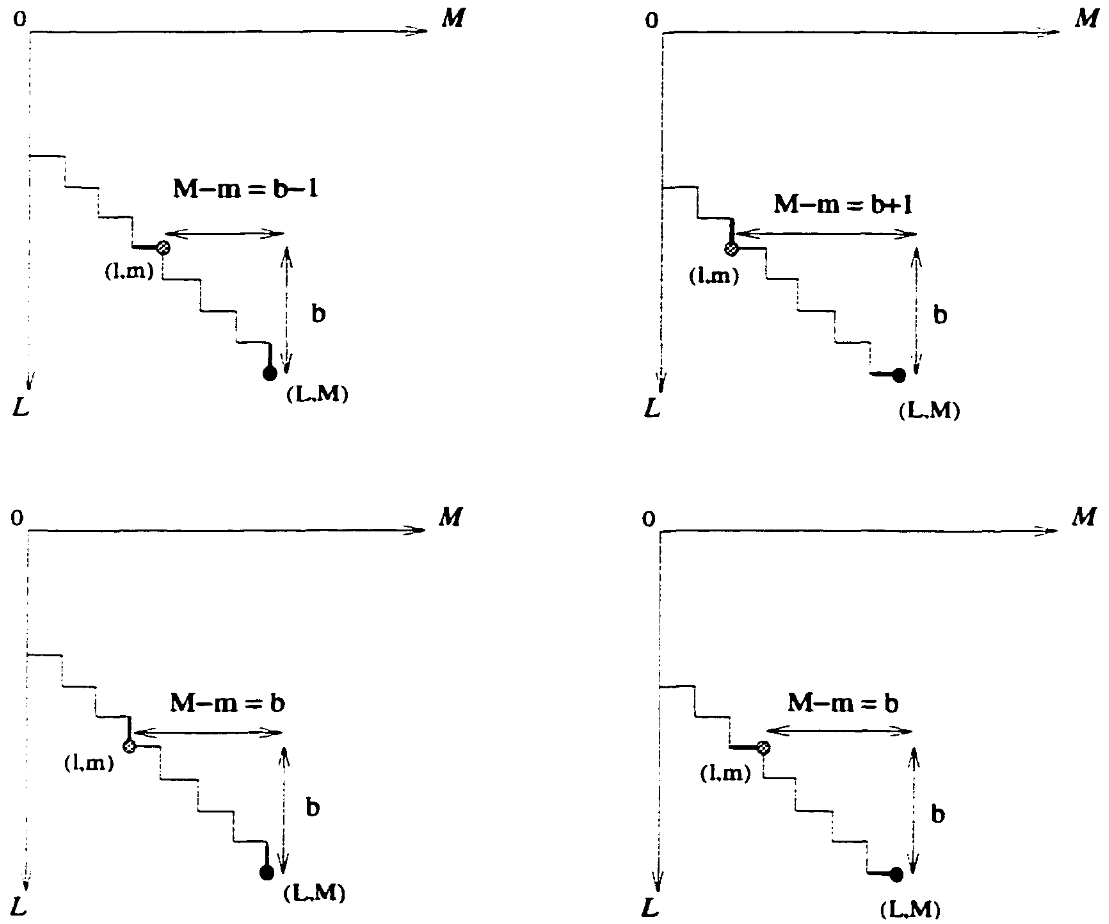

Figure 3.2: Orientations of $s(z)$ and $S(z)$

since $(\hat{p}(z), \hat{q}(z))$ interpolates one fewer residual and must be of type $[b-1, b-2]$ (lies on the staircase below the diagonal). Furthermore, with $M-m = b-1$ in Fig. 3.2, the degree condition for $s(z)$ is

$$\deg(s(z)) \leq \begin{pmatrix} l & l+1 \\ m & m \end{pmatrix}. \tag{3.24}$$

Because $l = L - b$ and $m = M - b + 1$, it then follows from (3.23) and (3.24) that

$$\deg(s(z)\hat{s}(z)) \leq \begin{pmatrix} L & L \\ M & M+1 \end{pmatrix}, \tag{3.25}$$

and so $S(z) = s(z)\hat{s}(z)$ satisfies the degree conditions in Definition 3.1.

For the interpolation condition, it is given that $s(z)$ interpolates z_j, for $j = 0, \ldots, n$, i.e.,

$$(\, g_j \quad f_j \,)\, s(z_j) = (\, 0 \quad 0 \,), \quad j = 0, \ldots, n. \tag{3.26}$$

and $\hat{s}(z)$ interpolates the residuals of $s(z)$ for $j = n+1, \ldots, N$, i.e,

$$\begin{pmatrix} w_j & r_j \end{pmatrix} \hat{s}(z_j) = \begin{pmatrix} 0 & 0 \end{pmatrix}, \quad j = n+1, \ldots, N. \tag{3.27}$$

Combining (3.26) and (3.27) gives

$$\begin{pmatrix} g_j & f_j \end{pmatrix} s(z_j)\hat{s}(z_j) = \begin{pmatrix} g_j & f_j \end{pmatrix} S(z_j) = \begin{pmatrix} 0 & 0 \end{pmatrix}, \quad j = 0, \ldots, N, \tag{3.28}$$

Thus, $S(z) = s(z)\hat{s}(z)$ satisfies the interpolation condition in Definition 3.1. Finally, since

$$\begin{aligned} \det(S(z)) &= \det(s(z)) \det(\hat{s}(z)) & (3.29)\\ &= \gamma t_{0;n}(z) \cdot \hat{\gamma} t_{n+1,N}(z) & (3.30)\\ &= \Gamma t_{0,N}(z), & (3.31) \end{aligned}$$

where $\Gamma = \gamma\hat{\gamma}$, then the nonsingularity condition in Definition 3.1 is also satisfied. □

We now examine how unattainable points of $S(z)$ are related to those of $s(z)$ and $\hat{s}(z)$. From Theorem 3.2, observe that

$$\begin{pmatrix} U(z) \\ V(z) \end{pmatrix} = s(z) \begin{pmatrix} \hat{u}(z) \\ \hat{v}(z) \end{pmatrix}. \tag{3.32}$$

Nothing can be said for points z_σ, $0 \leq \sigma \leq n$; z_σ may be an unattainable point for $(U(z), V(z))$ but not for $(\hat{u}(z), \hat{v}(z))$. For points z_σ, $n+1 \leq \sigma \leq N$, however, we have the following results.

Theorem 3.3 *Given LRIS's $s(z)$ and $\hat{s}(z)$, then for $n+1 \leq \sigma \leq N$, $|U(z_\sigma)| + |V(z_\sigma)| = 0$ if and only if $|\hat{u}(z_\sigma)| + |\hat{v}(z_\sigma)| = 0$.*

Proof: From (3.32), it follows that if $|\hat{u}(z_\sigma)| + |\hat{v}(z_\sigma)| = 0$ then $|U(z_\sigma)| + |V(z_\sigma)| = 0$. Conversely, observe that $s(z_j)$ is not singular for $n+1 \leq \sigma \leq N$, since

$$\det(s(z)) = \gamma\, t_{0,n}(z) \tag{3.33}$$

where

$$t_{0,n}(z) = \prod_{l=0}^{n} (z - z_l).$$

Therefore, from (3.32), it follows that

$$s(z_\sigma)^{adj} \begin{pmatrix} U(z_\sigma) \\ V(z_\sigma) \end{pmatrix} = det(s(z_\sigma)) \begin{pmatrix} \hat{u}(z_\sigma) \\ \hat{v}(z_\sigma) \end{pmatrix}. \tag{3.34}$$

Since $\det(s(z_\sigma)) \neq 0$ for $n+1 \leq \sigma \leq N$, it follows that $|\hat{u}(z_\sigma)| + |\hat{v}(z_\sigma)| = 0$ if $|U(z_\sigma)| + |V(z_\sigma)| = 0$. □

Theorem 3.3 tells us that in the range of $n+1 \leq \sigma \leq N$, the unattainable point of z_σ for $(U(z), V(z))$ is independent of $s(z)$; it can be determined from $\hat{s}(z)$ alone. Thus, Theorem 3.3 gives us an efficient way to test whether z_σ is an unattainable point in the range $n+1 \leq \sigma \leq N$; we need to test only $\hat{s}(z)$ rather than $S(z)$ which has larger degree polynomials.

3.3 The Recursive Case

Theorem 3.2 says that we can divide a large LRIS system $S(z)$ into two smaller LRIS systems $s(z)$ and $\hat{s}(z)$ as $S(z) = s(z)\hat{s}(z)$. But a more constructive way of interpreting Theorem 3.2 is that given a small LRIS $s(z)$ with its residuals, a larger LRIS $S(z)$ can be constructed if we can construct a LRIS $\hat{s}(z)$ that interpolates the residuals of $s(z)$. Thus, we can apply this idea to accommodate more and more points by recursively applying this theorem. It is this idea that leads to a recursive $O(N^2)$ algorithm and which is described in the next section.

Let $\{(l_i, m_i)\}_{i=0,\dots,k+1}$ be a sequence of nonsingular entries along one of the two staircases through $[L, M]$. We have $l_{i+1} \geq l_i$ and $m_{i+1} \geq m_i$, with a strict inequality for one of these, and k is such that $[L, M] = (l_{k+1}, m_{k+1})$. Define

$$n_i = l_i + m_i, \quad i = 0, \dots, k+1, \tag{3.35}$$

and

$$t_i = n_{i+1} - n_i, \quad i = 0, \dots, k. \tag{3.36}$$

Let $S^{(i)}(z)$ be the LRIS of type (l_i, m_i) for $\{(f_j, g_j)\}_{j=0,\dots,n_i}$ and define

$$(\, w_j \quad r_j \,)^{(i)} = (\, g_j \quad f_j \,) S^{(i)}(z_j), \quad j = n_i + 1, \dots, n_i + t_i. \tag{3.37}$$

If $s^{(i)}(z)$ is a LRIS of the appropriate type interpolating the residual $(\, w_j \quad r_j \,)^{(i)}$, $j = n_i + 1, \dots, n_i + t_i$, from Theorem 3.2,

$$S^{(i+1)}(z) = S^{(i)} s^{(i)}(z), \tag{3.38}$$

is a LRIS of type $[l_{i+1}, m_{i+1}]$. We will show in Chapter 4 that if t_i is chosen such that it is the smallest step size for advancing from one nonsingular entry to the next, an $O(N^2)$ operations algorithm is devised.

Thus, the recursive theorems allow us to advance from one nonsingular entry to another along a staircase path. Here, with $i = 0, \dots, k$, we have $S^{(k+1)}(z)$ in $k+1$ steps as

$$S^{(k+1)}(z) = s^{(0)}(z) s^{(1)}(z) \cdots s^{(k)}(z), \tag{3.39}$$

where $S^{(k+1)}(z)$ interpolates z_j, $j = 0, \ldots, n_k + t_k = N$.

Similar to Lemma 3.1, we have the following lemma for $\det(s^{(i)}(z))$.

Lemma 3.2

$$\det(s^{(i)}(z)) = \gamma^{(i)}\, t_{n_i+1, n_i+t_i}(z), \tag{3.40}$$

where

$$t_{n_i+1, n_i+t_i}(z) = \prod_{l=n_i+1}^{n_i+t_i} (z - z_l)$$

and $\gamma^{(i)}$ is the leading coefficient of $u^{(i)}(z)p^{(i)}(z) - v^{(i)}(z)q^{(i)}(z)$.

Proof: Since $s^{(i)}(z)$ interpolates the residuals $\{(\, w_j \quad r_j\,)\}_{j=n_i+1,\ldots,n_i+t_i}$, the proof is similar to that of Lemma 3.1. □

The concept of unattainability in the two-step case carries over to the multi-step case, and it is summarized in the following corollary.

Corollary 3.1 *Given LRIS's $s^{(0)}(z) \cdots s^{(k)}(z)$, the point z_σ, $n_i + 1 \leq \sigma \leq n_i + t_i$ is unattainable with respect to $[L, M]$ for the interpolant $(U(z), V(z))$ if and only if*

$$s^{(i)}(z_\sigma) \cdots s^{(k-1)}(z_\sigma) \begin{pmatrix} u^{(k)}(z_\sigma) \\ v^{(k)}(z_\sigma) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{3.41}$$

Proof: In Theorem 3.3, set $s(z) = s^{(0)}(z) \cdots s^{(i-1)}(z)$ which interpolates z_j, $j = 0, \ldots, n_i$, and

$$\begin{pmatrix} \hat{u}(z) \\ \hat{v}(z) \end{pmatrix} = s^{(i)}(z) \cdots s^{(k-1)}(z) \begin{pmatrix} u^{(k)}(z) \\ v^{(k)}(z) \end{pmatrix} \tag{3.42}$$

which interpolates z_j, $j = n_i + 1, \ldots, N$. □

The above Corollary gives us an efficient way to test whether a point z_σ is an unattainable point.

Chapter 4

Interpolation Algorithm

In this chapter, we present a *fast* algorithm that produces a sequence of LRIS $s^{(i)}(z)$, $i = 0, \ldots, k$, described in §3.3. We first present this algorithm in algebraic form and then, through appropriate modifications, in numerical form. A stability analysis of the numerical algorithm is given in Chapter 6.

4.1 Algebraic Form

When computing $s^{(i)}(z)$, we take advantage of a certain observation that reduces cost. More importantly, an extension of this observation is crucial to the development of a stable evaluation formula described in Chapters 7 and 8. Given $S^{(i)}(z)$ and the associated residuals

$$(\, w_j \quad r_j \,)^{(i)} = (\, g_j \quad f_j \,)\, S^{(i)}(z_j), \quad j = n_i + 1, \ldots, n_i + t_i,$$

let

$$C^{(i)} = \{z_j : w_j^{(i)} = 0, j = n_i + 1, \ldots, n_i + t_i\} \tag{4.1}$$

and

$$\theta^{(i)}(z) = \prod_{z_l \in C^{(i)}} (z - z_l). \tag{4.2}$$

Theorem 4.1 *The i^{th} LRIS can be represented as*

$$s^{(i)}(z) = \begin{pmatrix} 1 & 0 \\ 0 & \theta^{(i)}(z) \end{pmatrix} s'^{(i)}(z), \tag{4.3}$$

where

$$s'^{(i)}(z) = \begin{pmatrix} u'(z) & p^{*'}(z) \\ v'(z) & q^{*'}(z) \end{pmatrix}^{(i)} \tag{4.4}$$

is an LRIS of appropriate type that interpolates at the points z_j, $j = n_i + 1, \ldots, n_i + t_i$, but excluding the points $z_j \in C^{(i)}$.

Proof: Let $z_j \in C^{(i)}$; that is, consider those z_j for which $w_j^{(i)} = 0$. Because $|r_j^{(i)}| + |w_j^{(i)}| \neq 0$, then $r_j^{(i)} \neq 0$ and it follows from

$$(\,w_j^{(i)} \quad r_j^{(i)}\,)\begin{pmatrix} u(z_j) & p^*(z_j) \\ v(z_j) & q^*(z_j) \end{pmatrix}^{(i)} = (\,0 \quad 0\,) \tag{4.5}$$

that $v^{(i)}(z_j) = q^{*^{(i)}}(z_j) = 0$. Thus, $\theta^{(i)}(z)$ is a factor of both $v^{(i)}(z)$ and $q^{*^{(i)}}(z)$ and the result follows. □

Note that with Theorem 4.1, the interpolation condition $(\,w_j \quad r_j\,)^{(i)} s^{(i)}(z_j) = (\,0 \quad 0\,)$, $j = n_i + 1, \ldots, n_i + t_i$, can now be written as

$$(\,w_j \quad r_j\theta(z_j)\,)^{(i)} s'^{(i)}(z_j) = (\,0 \quad 0\,), \quad j = n_i + 1, \ldots, n_i + t_i. \tag{4.6}$$

Pseudo-code for the algebraic case is given in Algorithm 4.1 below.

Algorithm 4.1 (Algebraic Interpolation Algorithm)

```
**Input**: $N$, $L$, $\{(z_j, f_j, g_j)\}_{j=0,\ldots,N}$.
**Output**: $k$, $s'^{(0)}(z), \cdots, s'^{(k)}(z)$ and $C^{(0)}, \cdots, C^{(k)}$.
**Initialization**:
    $M \leftarrow N - L$, $i \leftarrow 0$, $n_i \leftarrow -1$,
    $t_i \leftarrow \max\{L - M - 1, 0\} + 1$, $s'^{(-1)}(z) \leftarrow I$, $\theta^{(-1)}(z) \leftarrow 1$, Done $\leftarrow$ FALSE.
do { Compute $(w_j, r_j)^{(i)}$ for $j = n_i + 1, \ldots, n_i + t_i$ from (3.37)
    $(\,w_j \quad r_j\,)^{(i)} = (\,g_j \quad f_j\,)\begin{pmatrix}1 & 0\\ 0 & \theta^{(-1)}(z_j)\end{pmatrix} s'^{(-1)}(z_j) \cdots \begin{pmatrix}1 & 0\\ 0 & \theta^{(i-1)}(z_j)\end{pmatrix} s'^{(i-1)}(z_j).$
    Determine $C^{(i)}$, $\theta^{(i)}(z)$ according to (4.1) and (4.2).
    Determine $s'^{(i)}(z)$ such that (see (4.6))
        $(\,w_j \quad r_j\theta(z_j)\,)^{(i)} s'^{(i)}(z_j) = (\,0 \quad 0\,), \quad j = n_i + 1, \ldots, n_i + t_i, z_j \notin C^{(i)}$
    **If** $n_i + t_i = N$ **then**
        Done $\leftarrow$ TRUE; $k \leftarrow i + 1$.
    **elseif** $s'^{(i)}(z)$ is nonsingular **then**
        $n_{i+1} \leftarrow n_i + t_i$; $i \leftarrow i + 1$; $t_i \leftarrow 1$.
    **else**
        $t_i \leftarrow t_i + 1$.
    **end**{if}
}**Until** (Done=TRUE)
**Output**: $k$, $s'^{(0)}(z), \cdots, s'^{(k)}(z)$ and $C^{(0)}, \cdots, C^{(k)}$.
```

Algorithm 4.1 computes a general linear rational interpolant of type $[L, M]$, for $M \leq L$. However, it is the *basic* type [31] for which $L = M$ or $L = M + 1$ that is of primary interest (i.e., this type lies along the staircase on or immediately below the diagonal of the rational interpolation table). Each $s^{(i)}(z)$ in the algorithm, except possibly for the first LRIS $s^{(0)}(z)$, is a LRIS of *basic* type. Thus, the degree of $s^{(i)}(z)$, for $i > 0$ is

$$\deg(s^{(i)}(z)) \leq \begin{pmatrix} \lfloor \frac{t_i}{2} \rfloor & \lfloor \frac{t_i+1}{2} \rfloor \\ \lfloor \frac{t_i-1}{2} \rfloor & \lfloor \frac{t_i}{2} \rfloor \end{pmatrix}. \tag{4.7}$$

Because the first step serves only to provide a general degree type, we shall focus our attention on the *basic* type in the following discussion.

We now examine some aspects of Algorithm 4.1. In particular, we give some insights about the size of t_i and the properties of $s'^{(i)}(z)$.

Lemma 4.1 *If $C^{(i)}$ contains $\lfloor (t_i + 1)/2 \rfloor$ or more members from $\{(z_j)\}_{j=n_i+1,\dots,n_i+t_i}$, then a linear rational interpolant of type $[\lfloor t_i/2 \rfloor, \lfloor (t_i - 1)/2 \rfloor]$ for the residuals is*

$$(u(z),\ v(z))^{(i)} = \left(\prod_{\substack{l=n_i+1 \\ z_l \notin C^{(i)}}}^{n_i+t_i} (z - z_l),\ 0 \right) \tag{4.8}$$

Proof: Since $t_i = \lfloor (t_i + 1)/2 \rfloor + \lfloor t_i/2 \rfloor$, then in (4.8) $\deg(u(z)) \leq t_i - \lfloor (t_i + 1)/2 \rfloor = \lfloor t_i/2 \rfloor$. In addition, with $(u(z), v(z))^{(i)}$ given by (4.8)

$$w_j^{(i)} u^{(i)}(z_j) + r_j^{(i)} v^{(i)}(z_j) = 0, \quad j = n_i + 1, \dots, n_i + t_i, \tag{4.9}$$

because either $w_j^{(i)} = 0$ if $z_j \in C^{(i)}$ or $u^{(i)}(z_j) = 0$ if $z_j \notin C^{(i)}$. □

A consequence of Lemma 4.1 is that the do-loop in Algorithm 4.1 continues to cycle (increasing t_i by one for each cycle) as long as at least half of the $w_j^{(i)}$, $j = n_i + 1, \dots, n_i + t_i$ are zero. This is so because in these cases according to Lemma 4.1 and Theorem 2.4, the only choice of $s'^{(i)}(z)$ satisfying (4.6) must have 0 components in the second row (i.e., $s'(i)(z)$ must be singular). Thus, the first opportunity for the termination of the do-loop occurs when there is exactly one more $w_j^{(i)}$ which is nonzero rather than zero. Note that when this happens t_i will be odd and $w_{n_i+t_i}^{(i)} \neq 0$. Theorem 4.2 below describes this occurrence; $s'^{(i)}(z)$ satisfying (4.6) in this case is nonsingular and so the do-loop does indeed terminate.

Theorem 4.2 *Given the residuals* $\{(z_j, r_j^{(i)}, w_j^{(i)})\}_{j=n_i+1,\ldots,n_i+t_i}$ *with* $w_{n_i+t_i}^{(i)} \neq 0$ *and* t_i *odd, suppose* $C^{(i)}$ *contains* $(t_i-1)/2$ *members. Then the LRIS* $s^{(i)}(z)$ *of type* $[(t_i-1)/2, (t_i-1)/2]$ *is given by*

$$s^{(i)}(z) = \begin{pmatrix} u^{(i)}(z) & (z - z_{n_i+t_i})p^{(i)}(z) \\ v^{(i)}(z) & 0 \end{pmatrix}, \tag{4.10}$$

where

$$v^{(i)}(z) = \prod_{\substack{l=n_i+1 \\ z_l \in C^{(i)}}}^{n_i+t_i} (z - z_l) \tag{4.11}$$

$$p^{(i)}(z) = \prod_{\substack{l=n_i+1 \\ z_l \notin C^{(i)}}}^{n_i+t_i-1} (z - z_l) \tag{4.12}$$

and $u^{(i)}(z) \in \mathcal{P}_{(t_i-1)/2}$ *is a polynomial interpolating the* $(t_i+1)/2$ *points*

$$u(z_j) = -\frac{r_j^{(i)}}{w_j^{(i)}} v(z_j), \quad j = n_i+1, \ldots, n_i+t_i, \quad z_j \notin C^{(i)}. \tag{4.13}$$

Proof: The pair $(u^{(i)}(z), v^{(i)}(z))$ is a linear rational interpolant of type $[(t_i-1)/2, (t_i-1)/2]$ because the degree conditions are clearly satisfied and because (4.13) holds with $z_j \notin C^{(i)}$ and

$$w_j^{(i)} u^{(i)}(z_j) + r_j^{(i)} v^{(i)}(z_j) = 0 \cdot u^{(i)}(z_j) + r_j^{(i)} \cdot 0 = 0, \quad z_j \in C^{(i)}. \tag{4.14}$$

On the other hand, from Lemma 4.1, $(p^{(i)}(z), 0)$ is a linear rational interpolant of type[1] $[(t_i-1)/2, (t_i-3)/2]$ for the residual $\{(z_j, r_j^{(i)}, w_j^{(i)})\}_{j=n_i+1,\ldots,n_i+t_i-1}$. Finally, $s^{(i)}(z)$ is nonsingular because $\det(s^{(i)}(z)) = \gamma^{(i)} t_{n_i+1,n_i+t_i}(z) = \gamma^{(i)} \prod_{l=n_i+1}^{n_i+t_i} (z - z_l)$. □

In summary, the i^{th} iteration consists of two parts. The first part requires the successive computations of $(\, r_j^{(i)} \quad w_j^{(i)} \,)$, $j = n_i+1, \ldots$, until $(t_i+1)/2$ nonzero and $(t_i-1)/2$ zero $w_j^{(i)}$'s are computed (with t_i odd). This requires $O(n_i t_i)$ operations in $\mathcal{D}$. The second part requires the computation of the polynomial $u^{(i)}(z)$ of $(t_i-1)/2$ which interpolates the points specified by (4.13), as well as the expansion of $p^{(i)}(z)$ specified by (4.12) ($v^{(i)}(z)$ corresponds to $\theta^{(i)}(z)$ and therefore requires no expansion). This requires an additional $O(t_i^2)$ operations in $\mathcal{D}$. (Here, we assume an $O(t_i^2)$ polynomial interpolation algorithm is used such as the one given in [37, Chap. 5].) Thus, the total cost of the algorithm is

$$\sum_{i=0}^{k} \left[O(n_i t_i) + O(t_i^2) \right] = O(N^2) \tag{4.15}$$

since $\sum_{i=0}^{k} t_i = N + 1$.

[1] By convention, a polynomial of negative degree is the zero polynomial.

4.2 Numerical Form

Before we consider the numerical version of Algorithm 4.1, the notions of norm and condition number of an $n \times n$ matrix A are needed for the discussion. Throughout this study, the 1-norm is used, e.g.,

$$\| (g_j \quad f_j) \| = \max\{|g_j|, |f_j|\}, \tag{4.16}$$

and

$$\|A\| = \max_{1 \le j \le n} \sum_{i=1}^{n} |A_{i,j}|. \tag{4.17}$$

For a polynomial $P(z) \in \mathcal{P}_n$, we use the norm,

$$\|P(z)\| = \sum_{i=0}^{n} |p_i|, \tag{4.18}$$

and for a 2×2 polynomial matrix $s(z)$, where

$$s(z) = \begin{pmatrix} u(z) & p^*(z) \\ v(z) & q^*(z) \end{pmatrix}, \tag{4.19}$$

we use the norm

$$\|s(z)\| = \max\{\|u(z)\| + \|v(z)\|, \|p^*(z)\| + \|q^*(z)\|\}. \tag{4.20}$$

The condition number of A is defined as

$$\kappa(A) = \|A\| \, \|A^{-1}\|. \tag{4.21}$$

More detailed descriptions of norm and condition number of a matrix are presented in Chapter 5.

Let

$$S_l^{(i)}(z_j) = s^{(l)}(z_j) \cdots s^{(i-1)}(z_j), \quad 0 \le l < i, \tag{4.22}$$

and $S_i^{(i)}(z_j) = I$. Note that $S_0^{(i)}(z_j) = S^{(i)}(z_j)$.

The numerical version of Algorithm 4.1 has two major modifications. First, the definition of $C^{(i)}$ is replaced by

$$C^{(i)} = \{z_j : |\alpha_j^{(i)} w_j^{(i)}| < \tau\mu, \; j = n_i + 1, \ldots, n_i + t_i\}, \tag{4.23}$$

where

$$\alpha_j^{(i)} = \frac{1}{\| (w_j^{(i)} \quad r_j^{(i)}) \|}, \tag{4.24}$$

τ is a stability parameter tolerance specified by the user, and μ is the unit error on the machine on which the algorithm is implemented. Note that $\|\,(\, w_j^{(i)} \quad r_j^{(i)}\,)\,\| \neq 0$ (see Remark 3.3). Second, we replace the nonsingularity test of $s'^{(i)}(z)$ by the *stability criterion*

$$\tau^{(i)}(z_{n_{i+1}+1}) \leq \tau, \tag{4.25}$$

where

$$\tau^{(i)}(z_j) = \max_{0 \leq l \leq i} \kappa(S_{l+1}^{(i+1)}(z_j)) \cdot \|s^{(l)^{-1}}(z_j)\|. \tag{4.26}$$

We call $\tau^{(i)}(z_j)$ the stability parameter at z_j. In (4.26), by convention we set $\tau^{(i)}(z_{n_{i+1}+1})$ to ∞ if any one of $s^{(l)}(z)$, $l = 0, \ldots, i$, is singular. The relationship between the nonsingularity test of $s'^{(i)}(z)$ in Algorithm 4.1 and the stability criterion (4.25) is given by Remark 4.1 below.

Remark 4.1 *The LRIS $s'^{(i)}(z)$ is nonsingular if and only if $\tau^{(i)}(z_{n_i+1}) \leq \tau$ for some finite τ.*

Proof: Suppose $s'^{(i)}(z)$ is nonsingular. By assumption at the i^{th} step of the do-loop in Algorithm 4.1, $s'^{(l)}(z)$ is also nonsingular for $0 \leq l \leq i-1$. So

$$\begin{aligned} \det(S_{l+1}^{(i+1)}(z)) &= \det(s^{(l+1)}(z)) \cdots \det(s^{(i)}(z)) \\ &= \gamma^{(l+1)} \cdots \gamma^{(i)} t_{n_{l+1}+1,n_{i+1}}(z). \end{aligned} \tag{4.27}$$

Since $t_{n_{l+1}+1,n_{i+1}}(z_{n_{i+1}+1}) = \prod_{a=n_{l+1}+1}^{n_{i+1}} (z_{n_{i+1}+1} - z_a) \neq 0$ for $0 \leq l \leq i$, then

$$\begin{aligned} \tau^{(i)}(z_{n_{i+1}+1}) &= \max_{0 \leq l \leq i} \kappa(S_{l+1}^{(i+1)}(z_{n_{i+1}+1})) \cdot \|s^{(l)^{-1}}(z_{n_{i+1}+1})\| \\ &= \max_{0 \leq l \leq i} \frac{\|S_{l+1}^{(i+1)}(z_{n_{i+1}+1})\| \, \|S_{l+1}^{(i+1)^{adj}}(z_{n_{i+1}+1})\| \, \|s^{(l)^{adj}}(z_{n_{i+1}+1})\|}{|\det(S_{l+1}^{(i+1)}(z_{n_{i+1}+1}))| \, |\det(s^{(l)}(z_{n_{i+1}+1}))|} < \infty. \end{aligned} \tag{4.28}$$

Conversely, if $s'^{(i)}(z)$ is singular, then $\tau^{(i)}(z_{n_{i+1}+1}) = \infty$ by convention. □

The specification of the tolerance parameter τ provides control over the conditioning of all the LRIS's $S^{(l)}(z)$ and $s^{(l)}(z)$, $0 \leq l \leq i$, evaluated at the point $z_{n_{i+1}+1}$. Whereas, only minimal well-conditioning (finite τ) of $s^{(i)}(z_{n_{i+1}+1})$ is sufficient to ensure the nonsingular of $s'^{(i)}(z)$ (i.e., to satisfy the nonsingularity test in Algorithm 4.1), the well-conditioning of all the LRIS's is vital to the numerical stability of the algorithm. For example, in Chapter 6 we show that if $\tau^{(i)}(z_{n_{i+1}+1}) \leq \tau$, then the residual error is bounded by $O(\mu\tau)$.

The objective of the stability criterion $\tau^{(i)}(z_{n_{i+1}+1}) \leq \tau$ in (4.25) is to ensure that the i^{th} LRIS $s^{(i)}(z)$ does not result in a solution $S^{(i+1)}(z) = S^{(i)}(z)s^{(i)}(z)$ which corresponds to an ill-conditioned problem as we shall see in Theorem 5.2 of Chapter 5. In particular, it follows from (4.26) that the condition number of $S^{(i+1)}(z)$ at the point $z_{n_{i+1}+1}$ satisfies

$$\kappa(S^{(i+1)}(z_{n_{i+1}+1})) \quad \leq \quad \tau^{(i)}(z_{n_{i+1}+1})\|s^{(0)}(z_{n_{i+1}+1})\|. \tag{4.29}$$

If the stability criterion (4.25) is satisfied, we know at least $\kappa(S^{(i+1)}(z_{n_{i+1}+1}))$ is bounded by $\tau\ \|s^{(0)}(z_{n_{i+1}+1})\|$. However, nothing can be said about the other points in the same step.

Note that for a general rational interpolant where $L - M > 1$, the first LRIS $s^{(0)}(z)$ involves a polynomial interpolation of degree $n_1 = L - M - 1$, and has the form of

$$s^{(0)}(z) = \begin{pmatrix} u^{(0)}(z) & t_{0,n_1}(z) \\ 1 & 0 \end{pmatrix}, \tag{4.30}$$

where $u^{(0)}(z)$ is a polynomial interpolating the first $n_1 + 1$ points with $\deg(u^{(0)}(z)) = n_1$. We refer to the analysis of the polynomial interpolation in [37]. For this case, the stability parameter is

$$\begin{aligned} \tau^{(0)}(z_{n_1+1}) &= \|s^{(0)^{-1}}(z_{n_1+1})\| \\ &= \frac{\|s^{(0)^{adj}}(z_{n_1+1})\|}{|\gamma^{(0)} t_{0,n_1}(z_{n_1+1})|} \\ &= \frac{\|s^{(0)^{adj}}(z_{n_1+1})\|}{|\gamma^{(0)} \prod_{a=0}^{n_1}(z_a - z_{n_1+1})|}. \end{aligned} \tag{4.31}$$

Since the magnitude of $1/|t_{0,n_1}(z_{n_1+1})|$ depends proportionally on the size of $n_1 = L - M - 1$, in general, for small $L - M$, $\tau^{(0)}(z_{n_1+1})$ is small and therefore (4.25) is satisfied. Hence, the first LRIS $s^{(0)}(z)$ is generally of the form (4.30). However, as $L - M$ increases, $\tau^{(0)}(z_{n_1+1})$ increases proportionally, in which case, a general $s^{(0)}(z)$ on the staircase may be required so that the stability criterion is satisfied (i.e., $\tau^{(0)}(z_{n_1+1}) \ < \tau$). Although the design of the algorithm is to compute a general rational interpolant of type $[L, M]$, we implicitly exclude the cases where $M \ll L$, since these cases (including the case where $M = 0$, which reduces to a polynomial interpolation problem) are not the focus of this study.

As we shall see in Chapter 6, the residual error bound that we shall obtain is a pointwise error bound. But since for every step, only one point is used to determine the stability of the solution obtained, the other points in the same step may not be bounded by τ. At the beginning of the i^{th} step, we only know that

$$\tau^{(i-1)}(z_{n_i+1}) \leq \tau, \tag{4.32}$$

but we may or may not have the same bound at the points $\{z_j\}_{j=n_i+2,\ldots,n_i+t_i}$. Thus, we introduce the parameter

$$\psi_j = \frac{\tau^{(i-1)}(z_j)}{\tau^{(i-1)}(z_{n_i+1})}, \quad j = n_i + 1, \ldots, n_i + t_i, \tag{4.33}$$

which is computed by Algorithm 4.2 below. It then follows from (4.25) that

$$\tau^{(i-1)}(z_j) \leq \tau \cdot \psi_j, \quad j = n_i + 1, \ldots, n_i + t_i, \tag{4.34}$$

an inequality which is used in the stability proofs of Chapter 6. Notice that the magnitude of ψ_j, $j = n_i + 2, \ldots, n_i + t_i$, in (4.33) can be arbitrarily large. However, as we shall see in §9.4, should ψ_j corresponding to z_j be large in the range $n_i + 2 \leq j \leq n_i + t_i$, say at the point z^*, this could only mean that z^* is close to one of the points z_j, $j = 0, \ldots, n_i$. Indeed, if z^* is one of point z_j, $j = 0, \ldots, n_i$, then $\psi_j = \infty$ since $\tau^{(i-1)}(z_j) = \infty$ in (4.26). This follows because if in particular $z^* \in \{z_j, j = n_l + 1, \ldots, n_{l+1}\}$, $0 \leq l < i$, then $\det(s^{(l)}(z^*)) = \gamma^{(l)} t_{n_l+1,n_l+t_l}(z^*) = 0$ in (4.26). Thus, a large ψ_j serves to indicate problems in the data set.

Algorithm 4.2 (Numerical Interpolation Algorithm)

```
Input: $N$, $L$, $\tau$, $\{(z_j, f_j, g_j)\}_{j=0,\ldots,N}$.
Output: $k$, $s'^{(0)}(z), \cdots, s'^{(k)}(z)$, $C^{(0)}, \cdots, C^{(k)}$, and $\{\psi_j\}_{j=0,\ldots,N}$.
Initialization:
    $M \leftarrow N - L$, $i \leftarrow 0$, $n_i \leftarrow -1$,
    $t_i \leftarrow \max\{L - M - 1, 0\} + 1$, $s^{(-1)}(z) \leftarrow I$, $\theta^{(-1)}(z) = 1$, Done $\leftarrow$ FALSE.
do{
    Compute $(w_j, r_j)^{(i)}$ for $j = n_i + 1, \ldots, n_i + t_i$ from (3.37)
        $(\, w_j \quad r_j \,)^{(i)} = (\, g_j \quad f_j \,) \begin{pmatrix} 1 & 0 \\ 0 & \theta^{(-1)}(z) \end{pmatrix} s'^{(-1)}(z_j) \cdots \begin{pmatrix} 1 & 0 \\ 0 & \theta^{(i-1)}(z) \end{pmatrix} s'^{(i-1)}(z_j).$
    Determine $C^{(i)}$ according to (4.23).
    Normalize $\theta^{(i)}(z)$ so that $\|\theta^{(i)}(z)\| = 1$.
    Using Gaussian elimination with complete pivoting, determine $s'^{(i)}(z)$ such that
        $\alpha_j^{(i)} (\, w_j \quad r_j \,)^{(i)} s'^{(i)}(z_j) = (\, 0 \quad 0 \,)$,  $j = n_i + 1, \ldots, n_i + t_i, z_j \notin C^{(i)}$,
      where $\alpha_j^{(i)} = 1/\| (\, w_j^{(i)} \quad r_j^{(i)} |\,) \|$.
    Normalize $s'^{(i)}(z)$ so that each column of $\begin{pmatrix} 1 & 0 \\ 0 & \theta^{(i)}(z) \end{pmatrix} s'^{(i)}(z)$ has norm equal to 1.
    If $n_i + t_i = N$ then
        Compute $\{\psi_j\}_{j=n_i+1,\ldots,n_i+t_i}$; $k \leftarrow i + 1$; Done $\leftarrow$ TRUE.
    elseif $\tau^{(i)}(z_{n_{i+1}+1}) \leq \tau$ then
        Compute $\{\psi_j\}_{j=n_i+1,\ldots,n_i+t_i}$;

        $n_{i+1} \leftarrow n_i + t_i$; $i \leftarrow i + 1$; $t_i \leftarrow 1$.
        else
            $t_i \leftarrow t_i + 1$.
        end{if}
    } Until (Done=TRUE)
    Output: $k$, $s'^{(0)}(z), \cdots, s'^{(k)}(z)$, $C^{(0)}, \cdots, C^{(k)}$, $\kappa^{(0)}, \cdots, \kappa^{(k)}$ and $\{\psi_j\}_{j=0,\ldots,N}$.
```

The output $\kappa^{(0)}, \cdots, \kappa^{(k)}$ is the condition numbers of the subproblems of (4.6), the significance of which will be made clear in Chapters 5 and 6.

We use Gaussian elimination with complete pivoting to solve the system of equations in (4.6). The system of equations is solved first by reducing the corresponding matrix to an upper triangular form with complete pivoting, next by assigning the last variable to one (or should a zero pivot be encountered, by assigning one to the variable corresponding to the zero pivot at row l and one to the subsequent variables in the solution vector from $l+1$ to $t_i + 1$), and finally, by back substituting for the remaining variables. This procedure guarantees a solution for the i^{th} iteration even for a singular system.

Note that Algorithm 4.2 terminates when $n_i + t_i = N$ regardless of the size of the stability criterion. So, the last $s^{(k)}(z)$ in the solution may cause the final solution to be ill-conditioned, in which case $\kappa^{(k)}$ is used to alert the user.

We now discuss the complexity of Algorithm 4.2. There are k iterations of the do-loop in the algorithm. Each iteration consists of computing t_i residuals which requires $O(n_i t_i)$ operations, solving a system of t_i equations with Gaussian elimination which requires $O(t_i^3)$ operations, and computing $r^{(i)}(z_{n_{i+1}+1})$ which requires $O(n_i t_i)$ operations. However, since t_i is not known beforehand, the algorithm accepts a t_i only if the system solved using Gaussian elimination satisfies the stability criterion. Thus, the computation of the i^{th} system requires the solution of t_i systems each requiring at most $O(t_i^3)$ operations and the computation of t_i stability parameters $r^{(i)}(z_{n_{i+1}+1})$ each requiring at most $O(n_i t_i)$ operations. Since $\sum_{i=0}^{k} t_i = N + 1$, the complexity of the algorithm is

$$\sum_{i=0}^{k} \left[O(n_i t_i) + O(t_i^3) t_i + O(n_i t_i) t_i \right] = O(N^2) + O(t^3 N) + O(t N^2), \tag{4.35}$$

where $t = \max_{0 \le i \le k}\{t_i\}$.

For small t, the cost of Gaussian elimination is small and hence, like the algebraic algorithm, the complexity of Algorithm 4.2 is $O(N^2)$. Thus, Algorithm 4.2 is most efficient when t is minimal (or when the number of steps is maximal). Ideally $t = t_i = 1$, in which case our algorithm returns the same linear rational interpolant as does Werner's [60, 61]. A discussion of Werner's algorithm is given in Chapter 9.

For the extreme case, however, it is conceivable that for a given set of data, the singularity test is never satisfied, and therefore, Gaussian elimination is ultimately used to compute the linear rational interpolant. In such a case, N systems of equations are solved by Gaussian elimination each requiring as much as $O(N^3)$ operations, hence giving a complexity of $O(N^4)$. But such a case is rarely encountered in practice.

Note that in Algorithm 4.2, with the exception of the initial step, each subsequent step size t_i is initialized to one. This step size t_i in the i^{th} iteration is indeed one if the stability criterion in (4.25) is immediately satisfied; otherwise it is incremented by one recursively until (4.25) is satisfied or until the condition $n_i + t_i = N$ is reached, in which case the program terminates. This strategy, while not optimal in efficiency, guarantees a minimal step size t_i and hence the lowest degree type $s^{(i)}(z)$ at each iteration. This is the best we can do numerically since unlike the algebraic case (c.f. Theorem 4.2) there is no simple way *a priori* to determine the minimal step size that gives a well-conditioned $S^{(i+1)}(z)$.

Chapter 5

Problem Conditioning and Algorithm Stability

Since the goal of this research is to develop an efficient algorithm that is stable, the notion of stability must be clearly defined. The notions of stability and condition number are introduced by mathematicians to describe the sensitivity of solutions to mathematical problems when there are small perturbations in the input. Similar notions are given for describing numerical algorithms which compute solutions to these problems. In this chapter, the precise definitions of these concepts are given.

5.1 Problem Conditioning

The linear rational interpolation problem is equivalent to solving the set of $N \times N$ linear equations (1.5) (see also [3, 11, 16, 24, 26, 32, 39]); so we look at the stability issues based on such a system, namely,

$$Ax = b. \tag{5.1}$$

(Note that one can arrive at (5.1) from (1.5) by moving one column of the coefficient matrix to the right hand side.) The variable vector x is the unknown being sought, and A and b are the data on which the solution depends. We say that the problem (5.1) is *well-conditioned* if the solution x depends in a continuous way on A and b; a small change in A and b would lead to a correspondingly small change in x. If a problem is ill-conditioned, it is usually difficult to solve without first attempting to understand more about the problem itself, usually by returning to the context in which the mathematical problem is formulated. To measure the degree of conditioning of the problem, the *condition number* is introduced.

The condition number of a problem attempts to measure the worst possible effect on the solution of x of (5.1) when the inputs A and b are perturbed by a small amount. Let

δA and δb be perturbations of A and b, and $x + \delta x$ be the solution (if it exists) of the perturbed equation

$$(A + \delta A)(x + \delta x) = b + \delta b. \tag{5.2}$$

The condition number of (5.1) is defined to be

$$\kappa_p = \sup_{\delta A, \delta b} \frac{\|\delta x\|/\|x\|}{\|\delta A\|/\|A\| + \|\delta b\|/\|b\|}, \tag{5.3}$$

This condition number κ_p is a measure of the sensitivity of the solution x to small changes in the data A and b. If κ_p is large, then small relative changes in A and b can lead to large relative changes in x and the problem is said to be ill-conditioned. But if κ_p is small, then small relative changes in A and b always lead to correspondingly small relative changes in x. Since numerical calculations almost always involve a variety of small computational errors, problems with large condition numbers are difficult to solve accurately. Such problems are called *ill-conditioned.*

Let $\kappa(A) = \|A\|\,\|A^{-1}\|$. We have the following relationship (see Golub and van Loan [27]):

Theorem 5.1 *If $Ax = b$, where A is nonsingular, and if*

$$\frac{\|\delta A\|}{\|A\|} < \frac{1}{\kappa(A)}, \tag{5.4}$$

then $(A + \delta A)$ is nonsingular. And if we define δx by (5.2) then

$$\frac{\|\delta x\|}{\|x\|} \le \frac{\kappa(A)}{1 - \kappa(A)\frac{\|\delta A\|}{\|A\|}} \left\{ \frac{\|\delta A\|}{\|A\|} + \frac{\|\delta b\|}{\|b\|} \right\}. \tag{5.5}$$

In practice, we say that A is well-conditioned if $\kappa(A)$ is small. If this is the case, then we know that if we obtain a solution $\bar{x}$ to $Ax = b$ where $\|b - A\bar{x}\|$ is small, the $\|\bar{x} - x\|$ is small as well. On the other hand, if $\kappa(A)$ is large, then no conclusion can be drawn about the size of $\|\bar{x} - x\|$ from the size of $\|b - A\bar{x}\|$. In the case where $\kappa(A)$ is large, we say that A is ill-conditioned.

In linear rational interpolation problems, the matrix A is the generalized Vandermonde system with one column removed. While $\kappa(A)$ can be small for problems with small N, in general, $\kappa(A)$ is large where N is large. Hence, even for well-conditioned problems, the formulation of the linear rational interpolation problem in (1.5) is limited to problems with small N.

On the other hand, the formulation of the linear rational interpolation problem in §3.3 is different from (1.5), namely,

$$\begin{pmatrix} g_j & f_j \end{pmatrix} s^{(0)}(z_j) \ldots s^{(k)}(z_j) = \begin{pmatrix} 0 & 0 \end{pmatrix}, \quad j = 0, \ldots, N. \tag{5.6}$$

The unknowns here are a sequence of LRIS's $s^{(i)}(z)$, $i = 0, \ldots, k$, as compared to x (in (5.1) or $(U(z), V(z))$ in (1.5)). This formulation of the problem leads to a different conditioning of the problem.

Note that there are many sequences on the solution path that satisfy (5.6). But once a particular sequence $s^{(i)}(z)$, $i = 0, \ldots, k$, is selected, we can examine the conditioning of the problem. Without loss of generality, we assume a fixed sequence of $s^{(i)}(z)$, $i = 0, \ldots, k$, in the following discussion.

Given the input $\{(z_j, f_j, g_j)\}_{j=0,\ldots,N}$, the solution of the interpolation satisfying (5.6) is $S^{(k+1)}(z) = s^{(0)}(z) \cdots s^{(k)}(z)$. Let the perturbed input be $\{(z_j, f_j + \delta f_j, g_j + \delta g_j)\}_{j=0,\ldots,N}$ and the corresponding solution (if it exists) be $S^{(k+1)}(z) + \delta S^{(k+1)}(z) = (s^{(0)}(z) + \delta s^{(0)}(z)) \cdots (s^{(k)}(z) + \delta s^{(k)}(z))$, i.e.,

$$(g_j + \delta g_j \quad f_j + \delta f_j)\,(s^{(0)}(z_j) + \delta s^{(0)}(z_j)) \cdots (s^{(k)}(z_j) + \delta s^{(k)}(z_j)) = (0 \quad 0), \quad j = 0, \ldots, N. \tag{5.7}$$

For the original problem the residual satisfies

$$(w_j \quad r_j)^{(i)} = (g_j \quad f_j)\, S^{(i)}(z_j), \quad j = n_i + 1, \ldots, n_i + t_i. \tag{5.8}$$

and for the perturbed problem the residual satisfies

$$(w_j + \delta w_j \quad r_j + \delta r_j)^{(i)} = (g_j + \delta g_j \quad f_j + \delta f_j)\,(S^{(i)}(z_j) + \delta S^{(i)}(z_j)), \tag{5.9}$$

for $j = n_i + 1, \ldots, n_i + t_i$.

The conditioning of the problem is presented in two steps for i, $i = 0, \ldots, k$. We first give the sensitivity of the residuals $(w_j \quad r_j)^{(i)}$, $j = n_i + 1, \ldots, n_i + t_i$ to its perturbed input in Lemma 5.1 below. We then give the sensitivity of the solution $s^{(i)}(z)$ to the perturbed residuals in Lemma 5.2. Finally, we combine the two lemmas in Theorem 5.2 to give the conditioning of the model problem (5.6).

Lemma 5.1 *For $j = n_i + 1, \ldots, n_i + t_i$, $(\delta w_j \quad \delta r_j)^{(i)}$ in (5.9) satisfies*

$$\frac{\|(\delta w_j \quad \delta r_j)^{(i)}\|}{\|(w_j \quad r_j)^{(i)}\|} \le \kappa(S^{(i)}(z_j)) \left(\frac{\|(\delta g_j \quad \delta f_j)\|}{\|(g_j \quad f_j)\|} + \frac{\|\delta S^{(i)}(z_j)\|}{\|S^{(i)}(z_j)\|} + \frac{\|(\delta g_j \quad \delta f_j)\|}{\|(g_j \quad f_j)\|} \frac{\|\delta S^{(i)}(z_j)\|}{\|S^{(i)}(z_j)\|} \right). \tag{5.10}$$

Proof: With (5.8), (5.9) becomes

$$(\delta w_j \quad \delta r_j)^{(i)} = (\delta g_j \quad \delta f_j)\, S^{(i)}(z_j) + ((g_j \quad f_j) + (\delta g_j \quad \delta f_j))\delta S^{(i)}(z_j) \tag{5.11}$$

so that

$$\|\,(\,\delta w_j \quad \delta r_j\,)^{(i)}\,\| \le \|\,(\,\delta g_j \quad \delta f_j\,)\,\|\,\|S^{(i)}(z_j)\| + (\|\,(\,g_j \quad f_j\,)\,\| + \|\,(\,\delta g_j \quad \delta f_j\,)\,\|)\|\delta S^{(i)}(z_j)\|. \tag{5.12}$$

From (5.8),

$$(\,w_j \quad r_j\,)^{(i)}\,S^{(i)^{-1}}(z_j) = (\,g_j \quad f_j\,) \tag{5.13}$$

so that

$$\|\,(\,w_j \quad r_j\,)^{(i)}\,\|\,\|S^{(i)^{-1}}(z_j)\| \ge \|\,(\,g_j \quad f_j\,)\,\|. \tag{5.14}$$

From (5.12) and (5.14), the result follows. □

Before we give the relationship between the residual and $s^{(i)}(z)$ in Lemma 5.2, we introduce a new notation below.

Given the residual of the original problem $(\,w_j \quad r_j\,)^{(i)}$, $j = n_i + 1, \ldots, n_i + t_i$, from (5.8), we have

$$w_j^{(i)} u^{(i)}(z_j) + r_j^{(i)} v^{(i)}(z_j) = 0, \quad j = n_i + 1, \ldots, n_i + t_i, \tag{5.15}$$

and

$$w_j^{(i)} p^{(i)}(z_j) + r_j^{(i)} q^{(i)}(z_j) = 0, \quad j = n_i + 1, \ldots, n_i + t_i - 1. \tag{5.16}$$

In matrix form, (5.15) becomes

$$M^{(i)} x^{(i)} = 0 \tag{5.17}$$

where

$$M^{(i)} = \begin{pmatrix} w^{(i)}_{n_i+1} z^0_{n_i+1} & \cdots & w^{(i)}_{n_i+1} z^l_{n_i+1} & r^{(i)}_{n_i+1} z^0_{n_i+1} & \cdots & r^{(i)}_{n_i+1} z^m_{n_i+1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ w^{(i)}_{n_i+t_i} z^0_{n_i+t_i} & \cdots & w^{(i)}_{n_i+t_i} z^l_{n_i+t_i} & r^{(i)}_{n_i+t_i} z^0_{n_i+t_i} & \cdots & r^{(i)}_{n_i+t_i} z^m_{n_i+t_i} \end{pmatrix}, \tag{5.18}$$

$x^{(i)} = (u_0^{(i)}, \ldots, u_l^{(i)}, v_0^{(i)}, \ldots, v_m^{(i)})^t$, and except possibly for the first step, $l = \lfloor \frac{t_i}{2} \rfloor$ and $m = \lfloor \frac{t_i - 1}{2} \rfloor$.

If $l = m = 0$ (i.e, $t_i = 1$), then $M^{(i)} = (\,w_j^{(i)} \quad r_j^{(i)}\,)$. In this case, we let $A^{(i)} = \max\{|w_j^{(i)}|, |r_j^{(i)}|\}$. Note that on a staircase path, with the exception of $l = m = 0$, $(p^{(i)}(z), q^{(i)}(z))$ is of degree type $[l-1, m]$ if $l > m$ or $[l, m-1]$ if $l = m$. Without loss of generality, we assume that $l > m$ (if not, we proceed with the m^{th} column). We remove the $(l+1)^{th}$ column from $M^{(i)}$ to form

$$A^{(i)} = \begin{pmatrix} w^{(i)}_{n_i+1} z^0_{n_i+1} & \cdots & w^{(i)}_{n_i+1} z^{l-1}_{n_i+1} & r^{(i)}_{n_i+1} z^0_{n_i+1} & \cdots & r^{(i)}_{n_i+1} z^m_{n_i+1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ w^{(i)}_{n_i+t_i} z^0_{n_i+t_i} & \cdots & w^{(i)}_{n_i+t_i} z^{l-1}_{n_i+t_i} & r^{(i)}_{n_i+t_i} z^0_{n_i+t_i} & \cdots & r^{(i)}_{n_i+t_i} z^m_{n_i+t_i} \end{pmatrix}. \tag{5.19}$$

The $(l+1)^{th}$ column of $M^{(i)}$ is denoted by $M_l^{(i)}$. Hence with $|z_j| \leq 1$, $j = n_i + 1, \ldots, n_i + t_i$, we have the relationship

$$\|A^{(i)}\| \geq \| \, (\, w_j^{(i)} \quad r_j^{(i)} \,) \, \|, \quad j = n_i + 1, \ldots, n_i + t_i. \tag{5.20}$$

Let $x^{(i)^l}$ be the vector formed by removing the $l+1$ element $x_l^{(i)}$ from $x^{(i)}$.

With this notation, (5.17) becomes

$$A^{(i)} x^{(i)^l} = -x_l^{(i)} M_l^{(i)} \tag{5.21}$$

and (5.16) becomes

$$A^{(i)} y^{(i)} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ c^{(i)} \end{pmatrix} \tag{5.22}$$

where $y^{(i)} = (p_0^{(i)}, \ldots, p_{l-1}^{(i)}, q_0^{(i)}, \ldots, q_m^{(i)})^t$, and $c^{(i)} \neq 0$.

Similarly, given the residual of the perturbed problem $(\, w_j + \delta w_j \quad r_j + \delta r_j \,)^{(i)}$, $j = n_i + 1, \ldots, n_i + t_i$, from (5.9), we have for $j = n_i + 1, \ldots, n_i + t_i$

$$(w_j^{(i)} + \delta w_j^{(i)})(u^{(i)}(z_j) + \delta u^{(i)}(z_j)) + (r_j^{(i)} + \delta r_j^{(i)})(v^{(i)}(z_j) + \delta v^{(i)}(z_j)) = 0, \tag{5.23}$$

and for $j = n_i + 1, \ldots, n_i + t_i - 1$

$$(w_j^{(i)} + \delta w_j^{(i)})(p^{(i)}(z_j) + \delta p^{(i)}(z_j)) + (r_j^{(i)} + \delta r_j^{(i)})(q^{(i)}(z_j) + \delta q^{(i)}(z_j)) = 0. \tag{5.24}$$

In matrix form, (5.23) becomes

$$(M^{(i)} + \delta M^{(i)})(x^{(i)} + \delta x^{(i)}) = 0 \tag{5.25}$$

where

$$\delta M^{(i)} = \begin{pmatrix} \delta w_{n_i+1}^{(i)} z_{n_i+1}^0 & \cdots & \delta w_{n_i+1}^{(i)} z_{n_i+1}^l & \delta r_{n_i+1}^{(i)} z_{n_i+1}^0 & \cdots & \delta r_{n_i+1}^{(i)} z_{n_i+1}^m \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \delta w_{n_i+t_i}^{(i)} z_{n_i+t_i}^0 & \cdots & \delta w_{n_i+t_i}^{(i)} z_{n_i+t_i}^l & \delta r_{n_i+t_i}^{(i)} z_{n_i+t_i}^0 & \cdots & \delta r_{n_i+t_i}^{(i)} z_{n_i+t_i}^m \end{pmatrix} \tag{5.26}$$

and $\delta x^{(i)} = (\delta u_0^{(i)}, \ldots, \delta u_l^{(i)}, \delta v_0^{(i)}, \ldots, \delta v_m^{(i)})^t$.

Corresponding to the relationship between the matrices $M^{(i)}$ and $A^{(i)}$, we let $\delta A^{(i)}$ be the square matrix by removing the $(l+1)^{th}$ column from $\delta M^{(i)}$, and the $(l+1)^{th}$ column of $\delta M^{(i)}$ is denoted by $\delta M_l^{(i)}$. Let also $\delta x^{(i)^l}$ be the vector formed by removing the $(l+1)^{th}$ element $\delta x_l^{(i)}$ from $\delta x^{(i)}$.

With this notation, (5.25) becomes

$$(A^{(i)} + \delta A^{(i)})(x^{(i)^l} + \delta x^{(i)^l}) = -(x_l^{(i)} + \delta x_l^{(i)})(M_l^{(i)} + \delta M_l^{(i)}), \tag{5.27}$$

and (5.24) becomes

$$(A^{(i)} + \delta A^{(i)})(y^{(i)} + \delta y^{(i)}) = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \bar{c}^{(i)} \end{pmatrix} \tag{5.28}$$

where $\delta y^{(i)} = (\delta p_0^{(i)}, \ldots, \delta p_{l-1}^{(i)}, \delta q_0^{(i)}, \ldots, \delta q_m^{(i)})^t$, and $\bar{c}^{(i)} \neq 0$.

To solve $x^{(i)}$ and $y^{(i)}$, we let $x_l^{(i)} = \bar{x}_l^{(i)}$ and $c^{(i)} = \bar{c}^{(i)}$.

Lemma 5.2 *If $A^{(i)}$ is nonsingular and if*

$$\frac{\|\delta A^{(i)}\|}{\|A^{(i)}\|} < \frac{1}{\kappa(A^{(i)})}, \tag{5.29}$$

then

$$\frac{\|\delta x^{(i)}\|}{\|x^{(i)}\|}, \ \frac{\|\delta y^{(i)}\|}{\|y^{(i)}\|} \leq \frac{\kappa(A^{(i)})}{1 - \kappa(A^{(i)})\frac{\|\delta A^{(i)}\|}{\|A^{(i)}\|}} \frac{\|\delta M^{(i)}\|}{\|A^{(i)}\|}. \tag{5.30}$$

Proof: Since $x_l^{(i)} = \bar{x}_l^{(i)}$, we have $\delta x_l^{(i)} = 0$. From (5.21) and (5.27), we have

$$(A^{(i)} + \delta A^{(i)})\delta x^{(i)'} + \delta A^{(i)} x^{(i)'} = -x_l^{(i)} \delta M_l^{(i)} \tag{5.31}$$

$$\begin{aligned} (A^{(i)} + \delta A^{(i)})\delta x^{(i)'} &= -\delta A^{(i)} x^{(i)'} - x_l^{(i)} \delta M_l^{(i)} \\ &= -\delta M^{(i)} x^{(i)}. \end{aligned} \tag{5.32}$$

So,

$$\delta x^{(i)'} = -(A^{(i)} + \delta A^{(i)})^{-1} \delta M^{(i)} x^{(i)}. \tag{5.33}$$

From the nonsingularity of $A^{(i)}$, and (5.29), (5.33) and $\delta x_l^{(i)} = 0$, it follows that

$$\|\delta x^{(i)'}\| = \|\delta x^{(i)}\| \leq \frac{\|A^{(i)^{-1}}\|}{1 - \|A^{(i)^{-1}}\| \, \|\delta A^{(i)}\|} \|\delta M^{(i)}\| \, \|x^{(i)}\|. \tag{5.34}$$

(Note that the proof of the inequality

$$\|A^{(i)} + \delta A^{(i)})^{-1}\| \leq \frac{\|A^{(i)^{-1}}\|}{1 - \|A^{(i)^{-1}}\| \, \|\delta A^{(i)}\|} \tag{5.35}$$

can be found in [4, Chapter 7].) By multiplying both sides of (5.34) by $1/\|x^{(i)}\|$ and the right hand side by $\|A^{(i)}\|/\|A^{(i)}\|$, the result follows for $\|\delta x^{(i)}\|/\|x^{(i)}\|$.

Similarly, since $c^{(i)} = \bar{c}^{(i)}$, from (5.22) and (5.28), we have

$$(A^{(i)} + \delta A^{(i)})\delta y^{(i)} + \delta A^{(i)} y^{(i)} = 0 \tag{5.36}$$

$$(A^{(i)} + \delta A^{(i)})\delta y^{(i)} \;=\; -\delta A^{(i)} y^{(i)}. \tag{5.37}$$

So,

$$\delta y^{(i)} = -(A^{(i)} + \delta A^{(i)})^{-1}\delta A^{(i)} y^{(i)} \tag{5.38}$$

From the nonsingularity of $A^{(i)}$, and (5.29) and (5.38), it follows that

$$\|\delta y^{(i)}\| \le \frac{\|A^{(i)^{-1}}\|}{1 - \|A^{(i)^{-1}}\|\,\|\delta A^{(i)}\|}\|\delta A^{(i)}\|\,\|y^{(i)}\|. \tag{5.39}$$

By multiplying both sides of (5.34) by $1/\|y^{(i)}\|$ and the right hand side by $\|A^{(i)}\|/\|A^{(i)}\|$, the result follows for $\|\delta y^{(i)}\|/\|y^{(i)}\|$, because $\|\delta A^{(i)}\| \le \|\delta M^{(i)}\|$. □

We now give the relationship between the input and output in the theorem below.

Theorem 5.2 *If*

$$\frac{\|\delta A^{(i)}\|}{\|A^{(i)}\|} < \frac{1}{\kappa(A^{(i)})}, \tag{5.40}$$

then

$$\begin{aligned}\frac{\|\delta s^{(i)}(z)\|}{\|s^{(i)}(z)\|} \le \; & \frac{t_i\,(t_i+1)\,\kappa(A^{(i)})}{1 - \kappa(A^{(i)})\dfrac{\|\delta A^{(i)}\|}{\|A^{(i)}\|}} \max_{n_i+1\le j\le n_i+t_i}\left\{\kappa(S^{(i)}(z_j))\left(\frac{\|\,(\,\delta g_j \quad \delta f_j\,)\,\|}{\|\,(\,g_j \quad f_j\,)\,\|} + \right.\right.\\ & \left.\left.\frac{\|\delta S^{(i)}(z_j)\|}{\|S^{(i)}(z_j)\|} + \frac{\|\,(\,\delta g_j \quad \delta f_j\,)\,\|}{\|\,(\,g_j \quad f_j\,)\,\|}\cdot\frac{\|\delta S^{(i)}(z_j)\|}{\|S^{(i)}(z_j)\|}\right)\right\}.\end{aligned} \tag{5.41}$$

Proof: Let J_i be the index of the largest perturbed residual which is defined by

$$\|\,(\,\delta w^{(i)}_{J_i} \quad \delta r^{(i)}_{J_i}\,)\,\| = \max_{n_i+1\le j\le n_i+t_i}\|\,(\,\delta w_j \quad \delta r_j\,)\,\| \tag{5.42}$$

For the normalization $|z_j| \le 1$, $j = n_i+1,\ldots,n_i+t_i$, we have

$$\begin{aligned}\|\delta M^{(i)}\| &= \max\left\{\sum_{j=n_i+1}^{n_i+t_i}|\delta w^{(i)}_j|,\; \sum_{j=n_i+1}^{n_i+t_i}|\delta r^{(i)}_j|\right\}\\ &\le \sum_{j=n_i+1}^{n_i+t_i}\max\{|\delta w^{(i)}_j|, |\delta r^{(i)}_j|\}\\ &= \sum_{j=n_i+1}^{n_i+t_i}\|\,(\,\delta w^{(i)}_j \quad \delta r^{(i)}_j\,)\,\|\\ &\le t_i\cdot\|\,(\,\delta w^{(i)}_{J_i} \quad \delta r^{(i)}_{J_i}\,)\,\|.\end{aligned} \tag{5.43}$$

From (5.43) and (5.20), it follows that

$$\frac{\|\delta A^{(i)}\|}{\|A^{(i)}\|} \le t_i\frac{\|\,(\,\delta w_{J_i} \quad \delta r_{J_i}\,)\,\|}{\|\,(\,w_{J_i} \quad r_{J_i}\,)\,\|}. \tag{5.44}$$

Thus, $\exists J_i$, $n_i + 1 \le J_i \le n_i + t_i$ such that (5.44) is true. Therefore,

$$\frac{\|\delta A^{(i)}\|}{\|A^{(i)}\|} \le t_i \max_{n_i+1\le j\le n_i+t_i} \frac{\|\,(\,\delta w_j \quad \delta r_j\,)^{(i)}\,\|}{\|\,(\,w_j \quad r_j\,)^{(i)}\,\|}. \tag{5.45}$$

With (5.45), it follows from Lemmas 5.1 and 5.2 that

$$\begin{aligned} \frac{\|\delta x^{(i)}\|}{\|x^{(i)}\|}, \frac{\|\delta y^{(i)}\|}{\|y^{(i)}\|} \le & \frac{t_i\,\kappa(A^{(i)})}{1-\kappa(A^{(i)})\frac{\|\delta A^{(i)}\|}{\|A^{(i)}\|}} \max_{n_i+1\le j\le n_i+t_i} \left\{\kappa(S^{(i)}(z_j)) \left(\frac{\|\,(\,\delta g_j \quad \delta f_j\,)\,\|}{\|\,(\,g_j \quad f_j\,)\,\|}\right.\right. \\ & \left.\left. + \frac{\|\delta S^{(i)}(z_j)\|}{\|S^{(i)}(z_j)\|} + \frac{\|\,(\,\delta g_j \quad \delta f_j\,)\,\|}{\|\,(\,g_j \quad f_j\,)\,\|}\cdot\frac{\|\delta S^{(i)}(z_j)\|}{\|S^{(i)}(z_j)\|}\right)\right\}. \end{aligned} \tag{5.46}$$

Note that

$$\frac{\|\delta x^{(i)}\|}{\|x^{(i)}\|} = \frac{\|\delta u^{(i)}(z)\| + \|\delta v^{(i)}(z)\|}{\|u^{(i)}(z)\| + \|v^{(i)}(z)\|} \tag{5.47}$$

and

$$\frac{\|\delta y^{(i)}\|}{\|y^{(i)}\|} = \frac{\|\delta p^{(i)}(z)\| + \|\delta q^{(i)}(z)\|}{\|p^{(i)}(z)\| + \|q^{(i)}(z)\|} \tag{5.48}$$

However, when we form $\delta s^{(i)}(z)$ and $s^{(i)}(z)$, the second column is multiplied by $(z - z_{n_i+t_i})$. Thus, we need to obtain a bound for

$$\frac{\|\delta p^{*^{(i)}}(z)\| + \|\delta q^{*^{(i)}}(z)\|}{\|p^{*^{(i)}}(z)\| + \|q^{*^{(i)}}(z)\|} = \frac{\|(z - z_{n_i+t_i})\delta p^{(i)}(z)\| + \|(z - z_{n_i+t_i})\delta q^{(i)}(z)\|}{\|(z - z_{n_i+t_i})p^{(i)}(z)\| + \|(z - z_{n_i+t_i})q^{(i)}(z)\|}. \tag{5.49}$$

With $|z_j| \le 1$, we can see that the expression in the numerator is bounded by

$$\begin{aligned} \|\delta p^{*^{(i)}}(z)\| + \|\delta q^{*^{(i)}}(z)\| &\le \|(z - z_{n_i+t_i})\|\,(\|\delta p^{(i)}(z)\| + \|\delta q^{(i)}(z)\|) \\ &\le 2\,(\|\delta p^{(i)}(z)\| + \|\delta q^{(i)}(z)\|). \end{aligned} \tag{5.50}$$

To obtain a lower bound for the denominator, we first show that

$$\|a(z)\| \le (\beta + 1)\,\|a^*(z)\|, \tag{5.51}$$

where β is a non-negative integer, $a^*(z) = (z - z_{n_i+t_i})a(z)$ and $a(z) \in \mathcal{P}_\beta$. To see how (5.51) is true, we first note that

$$a(z) = \sum_{\alpha=0}^{\beta} a_\alpha z^\alpha = a_0 + a_1 z + \cdots + a_\beta z^\beta. \tag{5.52}$$

Now let the reciprocal of $a(z)$ be

$$\begin{aligned} \hat{a}(z) = z^\beta a(\frac{1}{z}) &= z^\beta(a_0 + a_1\frac{1}{z} + \cdots + a_\beta\frac{1}{z^\beta}) \\ &= a_0 z^\beta + a_1 z^{\beta-1} + \cdots + a_\beta. \end{aligned} \tag{5.53}$$

Clearly, $\|a(z)\| = \|\hat{a}(z)\|$. With the reciprocal form, then

$$\begin{aligned} (z - z_{n_i+t_i})a(z) &= a^*(z) & (5.54) \\ (1 - z_{n_i+t_i}z)\hat{a}(z) &= \hat{a}^*(z). & (5.55) \end{aligned}$$

Thus,

$$\begin{aligned} \hat{a}(z) &= (1 - z_{n_i+t_i}z)^{-1}\hat{a}^*(z)(\text{mod } z^{\beta+1}) \\ &= (1 + z_{n_i+t_i}z + z^2_{n_i+t_i}z^2 + \cdots + z^{\beta}_{n_i+t_i}z^{\beta})\hat{a}^*(z)(\text{mod } z^{\beta+1}). \quad (5.56) \end{aligned}$$

Hence with $|z_j| \leq 1$, it follows from (5.56) that

$$\|\hat{a}(z)\| \leq (\beta + 1)\|\hat{a}^*(z)\| \quad (5.57)$$

and (5.51) follows.

From (5.51) we can now write

$$\begin{aligned} \|p^{(i)}(z)\| + \|q^{(i)}(z)\| &\leq (\deg(p^{(i)}(z)) + 1)\|p^{*(i)}(z)\| + (\deg(q^{(i)}(z)) + 1)\|q^{*(i)}(z)\| \\ &\leq (\deg(p^{(i)}(z)) + 1)(\|p^{*(i)}(z)\| + \|q^{*(i)}(z)\|) \quad (5.58) \end{aligned}$$

Hence with (5.50) and (5.58), we have

$$\begin{aligned} \frac{\|\delta p^{*(i)}(z)\| + \|\delta q^{*(i)}(z)\|}{\|p^{*(i)}(z)\| + \|q^{*(i)}(z)\|} &\leq 2(\deg(p^{(i)}(z)) + 1)\frac{\|\delta p^{(i)}(z)\| + \|\delta q^{(i)}(z)\|}{\|p^{(i)}(z)\| + \|q^{(i)}(z)\|} \\ &= (t_i + 1)\frac{\|\delta y^{(i)}(z)\|}{\|y^{(i)}(z)\|}, \quad (5.59) \end{aligned}$$

where $\deg(p^{(i)}(z)) = \lfloor (t_i - 1)/2 \rfloor$. Because

$$\begin{aligned} \frac{\|\delta p^{*(i)}(z)\| + \|\delta q^{*(i)}(z)\|}{\|s^{(i)}(z)\|} &= \frac{\|\delta p^{*(i)}(z)\| + \|\delta q^{*(i)}(z)\|}{\max\{\|u^{(i)}(z)\| + \|v^{(i)}(z)\|, \|p^{*(i)}(z)\| + \|q^{*(i)}(z)\|\}} \\ &\leq \frac{\|\delta p^{*(i)}(z)\| + \|\delta q^{*(i)}(z)\|}{\|p^{*(i)}(z)\| + \|q^{*(i)}(z)\|}, \quad (5.60) \end{aligned}$$

with (5.46) and (5.59), the result follows. □

From Theorem 5.2, we now define the condition number of (5.6) to be

$$\kappa_S = \max_{0 \leq i \leq k} \left\{ \frac{t_i\,(t_i + 1)\,\kappa(A^{(i)})}{1 - \kappa(A^{(i)})\frac{\|\delta A^{(i)}\|}{\|A^{(i)}\|}} \max_{n_i+1 \leq j \leq n_i+t_i} \kappa(S^{(i)}(z_j)) \right\}. \quad (5.61)$$

Note that the condition of the problem is expressed in terms of the solution $S^{(i)}(z)$, $i = 0, \ldots, k$, to the problem. This is similar to expressing the condition number of the problem (5.1) of solving $Ax = b$ in terms of solution A^{-1} since $\kappa(A) = \|A\|\,\|A^{-1}\|$.

One can observe that if $t_0 = N+1$, (i.e., we take only one step to arrive at the solution), we have $S^{(0)}(z) = I$ and therefore, $\delta S^{(0)}(z_j) = 0$, $j = 0, \dots, N$. Thus, we have

$$\frac{\|\delta s^{(0)}(z)\|}{\|s^{(0)}(z)\|} \leq \frac{t_0\,(t_0+1)\,\kappa(A^{(0)})}{1-\kappa(A^{(0)})\frac{\|\delta A^{(0)}\|}{\|A^{(0)}\|}} \left(\frac{\|\,(\,\delta g_j \quad \delta f_j\,)\,\|}{\|\,(\,g_j \quad f_j\,)\,\|}\right). \tag{5.62}$$

Hence, the conditioning of the problem reduces to the one similar to that of (1.5). In this case, $\kappa(A^{(0)})$ would be large for problems with large N. In general, we would like to solve small systems so that all $\kappa(A^{(i)})$, $i = 0, \dots, k$ are small. (Note that for $t_i = 1$, $\kappa(A^{(i)}) = 1$.) In this case, if all $\kappa(S^{(i)}(z_j))$, $i = 0, \dots, k$, $j = 0, \dots, N$ are also small, then the problem is well-conditioned.

With this condition number (5.61) of the problem, we define a well-conditioned problem below.

Definition 5.1 *The interpolation problem (5.6) is well-conditioned if κ_S is not too large.*

(Note that how large the condition number may be before we consider a problem to be ill-conditioned depends on the accuracy of the data and the accuracy desired in the solution. See Bunch [15] for a detailed discussion.)

5.2 Algorithm Stability

We now discuss the precise definitions of numerical stability using first the model problem (5.1) and then the model problem (5.6). The following is a well-accepted definition of stability for numerical algorithms introduced by Bunch [15].

Definition 5.2 *An algorithm for solving linear equations is* **strongly stable** *for a class of matrices $\mathcal{M}$ if for each A in $\mathcal{M}$ and for each b the computed solution $\bar{x}$ to $Ax = b$ satisfies $\hat{A}\bar{x} = \hat{b}$, where $\hat{A}$ is also in $\mathcal{M}$, for some $\hat{A}$ that is close to A and $\hat{b}$ is close to b.*

Definition 5.2 simply states that the computed solution is the exact solution of a slightly perturbed problem which is in the same class as the original problem. However, in many situations, we are only interested in whether or not solutions are close to the true solution; we do not need to know whether their solutions are the true solutions of nearby problems as it is given in Definition 5.2. In this case, a weaker type of stability such as the one also introduced by Bunch [15] suffices.

Definition 5.3 *An algorithm for solving linear equations is* **weakly stable** *for a class of matrices $\mathcal{M}$ if for each well-conditioned A in $\mathcal{M}$ and for each b, the computed solution $\bar{x}$ to $Ax = b$ is such that $\|x - \bar{x}\|/\|x\|$ is small.*

From (5.5), it follows that a (strongly) stable algorithm is also weakly stable but not the converse is not true.

Let $r = b - A\bar{x}$. Then $\bar{x}$ satisfies the perturbed system $A\bar{x} = b - r$. If we know that A is well-conditioned, then it follows that to prove weak stability it is sufficient to show that the residual[1] r is relatively small in comparison to b.

Corresponding to the notion of weak stability of an algorithm for solving $Ax = b$, we present a similar definition of weak stability of Algorithm 4.2 below.

Definition 5.4 *Algorithm 4.2 for solving Problem 1.1 is weakly stable if for all well-conditioned problems, the computed solution $\bar{s}^{(i)}(z)$, $i = 0, \ldots, k$ is such that $\|s^{(i)}(z) - \bar{s}^{(i)}(z)\|/\|s^{(i)}(z)\|$, $i = 0, \ldots, k$ is small.*

With the notation of $\bar{s}^{(i)}(z)$ (also $\bar{S}^{(i)}(z)$) representing the computed solution, we let $\{(\,\bar{g}_j \quad \bar{f}_j\,)\}_{j=0,\ldots,N}$ be the input (if it exists) such that the computed solution interpolates it exactly, i.e.,

$$(\,\bar{g}_j \quad \bar{f}_j\,)\,\bar{s}^{(0)}(z_j)\ldots\bar{s}^{(k)}(z_j) = (\,0 \quad 0\,), \quad j = 0, \ldots, N. \tag{5.63}$$

Thus, with the condition number in (5.61) and Theorem 5.2, we have, for $i = 0, \ldots, k$,

$$\frac{\|s^{(i)}(z) - \bar{s}^{(i)}(z)\|}{\|s^{(i)}(z)\|} \le \kappa_S \max_{n_i+1\le j\le n_{i+1}} \left\{ \frac{\|\,(\,g_j - \bar{g}_j \quad f_j - \bar{f}_j\,)\,\|}{\|\,(\,g_j \quad f_j\,)\,\|} + \frac{\|S^{(i)}(z_j) - \bar{S}^{(i)}(z_j)\|}{\|S^{(i)}(z_j)\|} \right\} + O(\delta^2), \tag{5.64}$$

where $O(\delta^2)$ is of order

$$\frac{\|\,(\,g_j - \bar{g}_j \quad f_j - \bar{f}_j\,)\,\|}{\|\,(\,g_j \quad f_j\,)\,\|}\,\frac{\|S^{(i)}(z_j) - \bar{S}^{(i)}(z_j)\|}{\|S^{(i)}(z_j)\|}.$$

From (5.64), it follows that in order to prove Algorithm 4.2 is weakly stable, it is sufficient to show that

$$\left\{ \frac{\|\,(\,g_j - \bar{g}_j \quad f_j - \bar{f}_j\,)\,\|}{\|\,(\,g_j \quad f_j\,)\,\|} + \frac{\|S^{(i)}(z_j) - \bar{S}^{(i)}(z_j)\|}{\|S^{(i)}(z_j)\|} \right\}, \quad j = 0, \ldots, N, \tag{5.65}$$

is small for all well-conditioned problems.

[1] This residual r should not be confused with the residual r_j used elsewhere in the thesis.

Chapter 6

Error Analysis of the Algorithm

The objective of an error analysis is to show the existence of an *a priori* bound for some appropriate measure of the effects of round off errors on an algorithm [37]. Obtaining a bound for the errors is the most important task. Ideally, the bound is small for all choices of problem data. If not, it should at least reveal features of the algorithm that characterize any potential instabilities, and thereby suggest how the instability can be cured or avoided.

In this chapter, we give error bounds for computing the residuals $(\, w_j \quad r_j \,)^{(i)}\,(z_j)$, $j = n_i + 1, \ldots, n_i + t_i$ and for solving the associated generalized Vandermonde system to obtain $s^{(i)}(z)$. These errors subsequently allow us to prove that Algorithm 4.2 is weakly stable later in Chapter 8.

We first give the preliminaries that are needed for the error analysis.

6.1 Preliminaries

In this study, the conventional μ is used to denote the unit-roundoff of the floating point computations, and $fl(\cdot)$ is used to denote the floating point operation of an expression. The over-score bar ($\bar{\ }$) is also used to denote floating point expressions, e.g., $fl(x) = \bar{x}$.

Theorem 6.1 *If x is a real number within the range of floating-point numbers, then*

$$fl(x) = x(1+\delta), \quad \textit{where} \;\; |\delta| < \mu \tag{6.1}$$

or

$$fl(x) = \frac{x}{(1+\delta)}, \quad \textit{where} \;\; |\delta| \le \mu. \tag{6.2}$$

Proof: See either Forsythe and Moler [25] or Higham [37]. □

We assume the following basic arithmetic rounding operations: for any two floating-point numbers x and y, the exact real number x op y, where op $= +, -, *, /$, is obtained and then rounded. Thus,

$$fl(x \text{ op } y) = (x \text{ op } y)(1+\delta), \quad \text{where } |\delta| < \mu, \tag{6.3}$$

or

$$fl(x \text{ op } y) = \frac{(x \text{ op } y)}{(1+\delta)}, \quad \text{where } |\delta| < \mu. \tag{6.4}$$

The term $(1+\delta)^{\pm 1}$ appears every time a floating point operation is performed. So it is important to keep track of its effect after a series of floating point operations. The following lemma allows us to do just that.

Lemma 6.1 *If $|\delta_i| \le \mu$ and $k_i = \pm 1$ for $i = 1, \ldots, n$ and $n\mu < 1$, then*

$$\prod_{i=1}^{n}(1+\delta_i)^{k_i} = 1 + \phi_n, \tag{6.5}$$

where

$$|\phi_n| \le \frac{n\mu}{1-n\mu} = \lambda_n.$$

Proof: See [37] for a detailed proof. □

A convenient notation [51] for keeping track of the powers of $(1+\delta_i)$ is the following:

$$<n> = \prod_{i=1}^{n}(1+\delta_i)^{k_i}. \tag{6.6}$$

With this notation, one can readily see that

$$<\alpha><\beta> \;=\; <\alpha+\beta>, \tag{6.7}$$

$$\frac{<\alpha>}{<\beta>} \;=\; <\alpha+\beta>, \tag{6.8}$$

We now examine the effect of rounding errors for Horner's rule of evaluating a polynomial. For evaluating

$$P(z) = \sum_{i=0}^{n} p_i z^i, \tag{6.9}$$

a backward error analysis [37] shows that Horner's rule gives, not $P(z)$, but rather the exact evaluation at z of some perturbed polynomial (see Lemma 6.2 below)

$$\bar{P}(z) = \sum_{i=0}^{n} \bar{p}_i z^i. \tag{6.10}$$

We write

$$fl(P(z_j)) = \bar{P}(z_j), \tag{6.11}$$

where it is understood that $\bar{P}(z)$ is different for different z_j. Corresponding to $\bar{P}(z)$, we also define

$$\tilde{P}(z) = \sum_{i=0}^{n} |\bar{p}_i| z^i. \tag{6.12}$$

Lemma 6.2 *Horner's method for evaluating $P(z) \in \mathcal{P}_n$ at z_j yields $\bar{P}(z_j) = fl(P(z_j))$, where*

$$\bar{P}(z_j) = P(z_j) - \Phi, \tag{6.13}$$

and $|\Phi| \leq \tilde{P}(|z_j|)\ \lambda_{2n}$.

Proof: Higham [37, pp. 104–105] has shown that using Horner's method,

$$\begin{aligned} \bar{P}(z_j) &= p_0 \langle 1 \rangle + p_1 z_j \langle 3 \rangle + \cdots + p_{n-1} z_j^{n-1} \langle 2n-1 \rangle + p_n z_j^n \langle 2n \rangle & (6.14)\\ &= \bar{p}_0 + \bar{p}_1 z_j + \cdots + \bar{p}_{n-1} z_j^{n-1} + \bar{p}_n z_j^n, & (6.15) \end{aligned}$$

where

$$\begin{aligned} p_0 &= \frac{\bar{p}_0}{\langle 1 \rangle} = \bar{p}_0 \langle 1 \rangle & (6.16)\\ p_1 &= \frac{\bar{p}_1}{\langle 3 \rangle} = \bar{p}_1 \langle 3 \rangle & (6.17)\\ &\vdots \\ p_n &= \frac{\bar{p}_n}{\langle 2n \rangle} = \bar{p}_n \langle 2n \rangle. & (6.18) \end{aligned}$$

So,

$$\begin{aligned} P(z_j) &= p_0 + p_1 z_j + \cdots + p_{n-1} z_j^{n-1} + p_n z_j^n \\ &= \bar{p}_0 \langle 1 \rangle + \bar{p}_1 z_j \langle 3 \rangle + \cdots + \bar{p}_{n-1} z_j^{n-1} \langle 2n-1 \rangle + \bar{p}_n z_j^n \langle 2n \rangle \\ &= \bar{p}_0 (1 + \phi_1) + \bar{p}_1 z_j (1 + \phi_3) + \cdots + \bar{p}_n z_j^n (1 + \phi_{2n}) \\ &= \bar{P}(z_j) + \Phi, & (6.19) \end{aligned}$$

where $\Phi \leq \lambda_{2n} \tilde{P}(|z_j|)$, and the result follows. □

Corollary 6.1 *If Horner's method for evaluation of $P(z) \in \mathcal{P}_n$ at z_j yields $\bar{P}(z_j)$, then*

$$\bar{P}(z_j) \langle \alpha \rangle = P(z_j) - \Phi', \tag{6.20}$$

where $|\Phi'| \leq \tilde{P}(|z_j|)\ \lambda_{2n+\alpha}$.

Proof: From the proof of Lemma 6.2, we get

$$\begin{aligned} \bar{P}(z_j)<\alpha> &= P(z_j)<\alpha> - \Phi<\alpha> \\ &= P(z_j) + \Phi'. \end{aligned} \tag{6.21}$$

where

$$\begin{aligned} |\Phi'| &= |P(z_j)\lambda_\alpha + \Phi(1+\lambda_\alpha)| \\ &\leq |P(z_j)|\lambda_\alpha + |\Phi|(1+\lambda_\alpha) \\ &\leq |P(z_j)|\lambda_\alpha + \bar{P}(|z_j|)(\lambda_{2n} + \lambda_{2n}\lambda_\alpha) \\ &\leq \bar{P}(|z_j|)(1+\lambda_{2n})\lambda_\alpha + \bar{P}(|z_j|)(\lambda_{2n} + \lambda_{2n}\lambda_\alpha) \\ &= \bar{P}(|z_j|)(\lambda_{2n} + \lambda_\alpha + 2\,\lambda_{2n}\lambda_\alpha) \\ &\leq \bar{P}(|z_j|)\lambda_{2n+\alpha}. \end{aligned} \tag{6.22}$$

In the above, we have used Lemma 6.1 to show

$$\begin{aligned} \lambda_{2n} + \lambda_\alpha + 2\,\lambda_{2n}\lambda_\alpha &= \frac{2n\mu(1-\alpha\mu) + \alpha\mu(1-2n\mu) + 4n\alpha\mu^2}{(1-2n\mu)(1-\alpha\mu)} \\ &= \frac{(2n+\alpha)\mu}{1-(2n+\alpha)\mu + 2n\alpha\mu^2} \\ &\leq \frac{(2n+\alpha)\mu}{1-(2n+\alpha)\mu} = \lambda_{2n+\alpha}. \end{aligned} \tag{6.23}$$

□

Let $P(z) = \Theta P'(z)$, where

$$\Theta(z) = \prod_{i=1}^{a}(z - z_i) \tag{6.24}$$

and $P'(z) \in \mathcal{P}_b$.

Corollary 6.2 *If Horner's method is used to evaluate $P'(z) \in \mathcal{P}_b$ at z_j, and $\theta(z) \in \mathcal{P}_a$ at z_j is evaluated from its roots product, then*

$$\bar{P}(z_j) = P(z_j) - \Phi'', \tag{6.25}$$

where $|\Phi''| \leq |\theta(z_j)|\,\bar{P}'(|z_j|)\ \lambda_{2n}$ and $n = a + b$.

Proof:

$$\begin{aligned} \bar{P}(z_j) &= fl(\theta(z_j))fl(P'(z_j))<1> \\ &= \theta(z_j)fl(P'(z_j))<2a>, \end{aligned} \tag{6.26}$$

since $fl(\theta(z_j)) = \theta(z_j)\langle 2a-1 \rangle$. Applying Corollary 6.1,

$$\bar{P}(z_j) = \theta(z_j)P'(z_j) - \Phi'', \tag{6.27}$$

where $|\Phi''| \leq |\theta(z_j)|\,\tilde{P}'(|z_j|)\,\lambda_{2b+2a}$, and the result follows. □

6.2 Error Analysis of the Algorithm

We can now turn to the error analysis of Algorithm 4.2. We examine the i^{th} iteration in detail. At the start of the i^{th} iteration, Algorithm 4.2 has available $s^{(0)}(z)\cdots s^{(i-1)}(z)$ which approximately interpolates z_j, $j = 0, \ldots, n_i$. Algorithm 4.2 next finds $s^{(i)}(z)$ so that $s^{(0)}(z)\cdots s^{(i)}(z)$ interpolates at z_j, $j = n_i + 1, \ldots, n_i + t_i$. In this section we analyze the errors made in the computation of $s^{(i)}(z)$.

There are two sources of computational errors, viz.,

1. errors in calculating the residuals $(\, w_j \quad r_j\,)^{(i)}\,(z_j)$ for $j = n_i + 1, \ldots, n_i + t_i$ and
2. errors in solving the associated generalized Vandermonde system to obtain $s^{(i)}(z)$.

In the following, we present bounds for these two errors. Note that the error analysis is carried out for the *basic* degree type $[L, L]$ or $[L+1, L]$, the general case $[L, M]$ $(L > M+1)$ is the same except for the first step where we interpolate by a polynomial of degree $L-M-1$, in which case we refer to the error analysis for the polynomial interpolation [37].

6.2.1 Computation of the Residuals

In this section, a relative error bound for computing the residuals is given. The equation used to compute the residual $(\, w_j \quad r_j\,)^{(i)}$ is

$$(\, w_j \quad r_j\,)^{(i)} = (\, g_j \quad f_j\,)\, S^{(i)}(z_j), \quad j = n_i + 1, \ldots, n_i + t_i, \tag{6.28}$$

where

$$S^{(i)}(z) = \begin{pmatrix} 1 & 0 \\ 0 & \theta^{(0)}(z) \end{pmatrix} s'^{(0)}(z) \begin{pmatrix} 1 & 0 \\ 0 & \theta^{(1)}(z) \end{pmatrix} s'^{(1)}(z) \cdots \begin{pmatrix} 1 & 0 \\ 0 & \theta^{(i-1)}(z) \end{pmatrix} s'^{(i-1)}(z). \tag{6.29}$$

The computed $(\, \bar{w}_j \quad \bar{r}_j\,)^{(i)}$, does not satisfy (6.28) exactly. Rather, it satisfies

$$(\, \bar{w}_j \quad \bar{r}_j\,)^{(i)} = fl\left((\, g_j \quad f_j\,)\, \bar{S}^{(i)}(z_j)\right), \quad j = n_i + 1, \ldots, n_i + t_i, \tag{6.30}$$

where $\bar{S}^{(i)}(z_j)$ is computed iteratively according to

$$\bar{S}^{(i)}(z_j) = fl(\bar{S}^{(i-1)}(z_j) fl(s^{(i-1)}(z_j))) \tag{6.31}$$

with $\bar{S}^{(0)}(z_j) = S^{(0)}(z_j) = I$. Let

$$\begin{aligned} \delta S^{(i)}(z_j) &= \bar{S}^{(i)}(z_j) - S^{(i)}(z_j), \quad j = n_i + 1, \ldots, n_i + t_i, & (6.32) \\ (\,\delta w_j \quad \delta r_j\,)^{(i)} &= (\,\bar{w}_j \quad \bar{r}_j\,)^{(i)} - (\,w_j \quad r_j\,)^{(i)}, \quad j = n_i + 1, \ldots, n_i + t_i. & (6.33) \end{aligned}$$

In the following, we find bounds for $\delta S^{(i)}(z_j)$ and then for $(\,\delta w_j \quad \delta r_j\,)^{(i)}$.

Lemma 6.3 *The evaluation error $\delta S^{(i)}(z_j)$ in (6.32) satisfies*

$$\delta S^{(i)}(z_j) = \sum_{l=0}^{i-1} \bar{S}^{(l)}(z_j) \delta s^{\dagger^{(l)}}(z_j) \bar{s}^{(l+1)}(z_j) \cdots \bar{s}^{(i-1)}(z_j) + O(\mu^2), \qquad (6.34)$$

where $\|\delta s^{\dagger^{(l)}}(z_j)\| \leq \lambda_{t_l+3}$.

Proof: The proof is by induction on i. The result (6.34) is true for the initial step $i = 0$ because $\bar{S}^{(0)}(z) = S^{(0)}(z) = I$ and consequently $\delta S^{(0)}(z_j) = 0$ for $z_j, j = n_0 + 1, \ldots, n_0 + t_0$.

Assuming that the result is true for i; we show that it must then be true for $i + 1$. From (6.31),

$$\begin{aligned} \bar{S}^{(i+1)}(z_j) &= fl(\bar{S}^{(i)}(z_j) fl(s^{(i)}(z_j))) \\ &= \bar{S}^{(i)}(z_j) \bar{s}^{(i)}(z_j) {<}2{>}, & (6.35) \end{aligned}$$

where $<2>$ accounts for the error made when multiplying matrices of order 2. Using Lemma 6.2 and Corollary 6.2,

$$\bar{s}^{(i)}(z_j) = fl(s^{(i)}(z_j)) = s^{(i)}(z_j) + \delta s^{(i)}(z_j), \qquad (6.36)$$

where $|\delta s^{(i)}(z_j)| \leq \begin{pmatrix} 1 & 0 \\ 0 & |\theta^{(i)}(z_j)| \end{pmatrix} \bar{s}'^{(i)}(|z_j|) \lambda_{t_i+1}$, since the maximum degree of the components in $s^{(i)}(z)$ is $\lfloor \frac{t_i+1}{2} \rfloor$. Because $\| \begin{pmatrix} 1 & 0 \\ 0 & \theta^{(i)}(z) \end{pmatrix} s'^{(i)}(z) \| = 1$, it follows from Corollary 6.2 that $\|\delta s^{(i)}(z_j)\| \leq \lambda_{t_i+1}$. Next, applying Corollary 6.1 to $\bar{s}^{(i)}(z)$, it follows that

$$\bar{s}^{(i)}(z_j) {<}2{>} = s^{(i)}(z_j) + \delta s^{\dagger^{(i)}}(z_j), \qquad (6.37)$$

where

$$\|\delta s^{\dagger^{(i)}}(z_j)\| \leq \lambda_{t_i+3}. \qquad (6.38)$$

Thus, from (6.35) and (6.37), we have

$$\begin{aligned} \bar{S}^{(i+1)}(z_j) &= (S^{(i)}(z_j) + \delta S^{(i)}(z_j))(s^{(i)}(z_j) + \delta s^{\dagger^{(i)}}(z_j)) \\ &= S^{(i)}(z_j) s^{(i)}(z_j) + \delta S^{(i+1)}(z_j), & (6.39) \end{aligned}$$

where

$$\delta S^{(i+1)}(z_j) = S^{(i)}(z_j)\delta s^{\dagger^{(i)}}(z_j) + \delta S^{(i)}(z_j)s^{(i)}(z_j) + \delta S^{(i)}(z_j)\delta s^{\dagger^{(i)}}(z_j). \tag{6.40}$$

Using (6.32) and (6.36), (6.40) becomes

$$\begin{aligned}\delta S^{(i+1)}(z_j) &= \bar{S}^{(i)}(z_j)\delta s^{\dagger^{(i)}}(z_j) + \delta S^{(i)}(z_j)\bar{s}^{(i)}(z_j) - \delta S^{(i)}(z_j)\delta s^{(i)}(z_j) \\ &= \bar{S}^{(i)}(z_j)\delta s^{\dagger^{(i)}}(z_j) + \delta S^{(i)}(z_j)\bar{s}^{(i)}(z_j) + O(\mu^2). \qquad (6.41)\end{aligned}$$

(Note that in (6.41) we have replaced terms $\delta S^{(i)}(z_j)\delta s^{(i)}(z_j)$ with $O(\mu^2)$ for simplicity. This replacement is valid since at the start of the induction on $l+1$, we have assumed that the result is true for $\delta S^{(l)}(z_j)$, and $\delta s^{\dagger^{(l)}}(z_j)$ is bounded according to (6.38)). Now, expanding recursively, (6.41) becomes

$$\begin{aligned}\delta S^{(i+1)}(z_j) &= \bar{S}^{(i)}(z_j)\delta s^{\dagger^{(i)}}(z_j) + \\ &\quad \bar{S}^{(i-1)}(z_j)\delta s^{\dagger^{(i-1)}}(z_j)\bar{s}^{(i)}(z_j) + \\ &\quad \bar{S}^{(i-2)}(z_j)\delta s^{\dagger^{(i-2)}}(z_j)\bar{s}^{(i-1)}(z_j)\bar{s}^{(i)}(z_j) + \\ &\quad \cdots \\ &\quad \bar{S}^{(1)}(z_j)\delta s^{\dagger^{(1)}}(z_j)\bar{s}^{(2)}(z_j)\cdots\bar{s}^{(i)}(z_j) + \\ &\quad \bar{S}^{(0)}(z_j)\delta s^{\dagger^{(0)}}(z_j)\bar{s}^{(1)}(z_j)\cdots\bar{s}^{(i)}(z_j) + O(\mu^2) \\ &= \sum_{l=0}^{i} \bar{S}^{(l)}(z_j)\delta s^{\dagger^{(l)}}(z_j)\bar{s}^{(l+1)}(z_j)\cdots\bar{s}^{(i)}(z_j) + O(\mu^2). \qquad (6.42)\end{aligned}$$

□

With the expression of an error of evaluating $S^{(i)}(z_j)$ given in Lemma 6.3, we now give its relative error bound and a relative error bound for the residual error in Theorems 6.2 and 6.3 below. In these theorems, we use the numerical counterpart of $\tau^{(i-1)}(z_j)$ from (4.26)

$$\bar{\tau}^{(i-1)}(z_j) = \max_{0\le l\le i-1} \kappa(\bar{S}^{(i)}_{l+1}(z_j)) \cdot \|\bar{s}^{(l)^{-1}}(z_j)\|, \quad j = n_i+1,\ldots,n_i+t_i, \tag{6.43}$$

as part of the running error bound. This expression of the computed $\tau^{(i-1)}(z_j)$ differs from the actual $\bar{\tau}^{(i-1)}(z_j)$ by several floating point operations and is used here for simplicity. As pointed out by Higham [37] there are always rounding errors in the computation of the running error bound, but their effects are negligible for $N\mu \ll 1$; therefore we do not need many correct significant digits in an error bound. Similarly, we use

$$\bar{\psi}_j = \frac{\bar{\tau}^{(i-1)}(z_j)}{\bar{\tau}^{(i-1)}(z_{n_i+1})}, \quad j = n_i+1,\ldots,n_i+t_i, \tag{6.44}$$

as the computed ψ_j.

Theorem 6.2 *The evaluation of $S^{(i)}(z_j)$ for $j = n_i + 1, \ldots, n_i + t_i$ in (6.31) yields $\bar{S}^{(i)}(z_j)$ such that*

$$\bar{S}^{(i)}(z_j) = S^{(i)}(z_j) + \delta S^{(i)}(z_j), \quad j = n_i + 1, \ldots, n_i + t_i, \tag{6.45}$$

where

$$\frac{\|\delta S^{(i)}(z_j)\|}{\|\bar{S}^{(i)}(z_j)\|} \leq i \cdot \tau \cdot \bar{\psi}_j \cdot \max_{0 \leq l \leq i-1} \lambda_{t_l+3} + O(\mu^2). \tag{6.46}$$

Proof: From (4.22), we can write

$$\bar{S}_a^{(i)}(z_j) = \bar{s}^{(a)}(z_j) \cdots \bar{s}^{(i-1)}(z_j) + O(\mu) \tag{6.47}$$

so that for $l < a$,

$$\bar{S}_l^{(i)}(z_j) = \bar{s}^{(l)}(z_j) \cdots \bar{s}^{(a-1)}(z_j) \bar{S}_a^{(i)}(z_j) + O(\mu), \tag{6.48}$$

where $O(\mu)$ accounts for at most i multiplications of matrices of order 2. Then

$$\begin{aligned}
&\sum_{l=0}^{i-1} \|\bar{S}^{(l)}(z_j)\| \, \|\bar{s}^{(l+1)}(z_j) \cdots \bar{s}^{(i-1)}(z_j)\| \\
&\quad = \sum_{l=0}^{i-1} \|\bar{S}^{(l)}(z_j)[\bar{s}^{(l)}(z_j)\bar{S}_{l+1}^{(i)}(z_j)] \cdot [\bar{s}^{(l)}(z_j)\bar{S}_{l+1}^{(i)}(z_j)]^{-1}\| \|\bar{s}^{(l+1)}(z_j) \cdots \bar{s}^{(i-1)}(z_j)\| \\
&\quad \leq \sum_{l=0}^{i-1} \|\bar{S}^{(i)}(z_j) + O(\mu)\| \|\bar{S}_{l+1}^{(i)^{-1}}(z_j)\| \|\bar{s}^{(l)^{-1}}(z_j)\| \, \|\bar{S}_{l+1}^{(i)}(z_j) + O(\mu)\|, \\
&\quad \leq \sum_{l=0}^{i-1} \|\bar{S}^{(i)}(z_j))\| \kappa(\bar{S}_{l+1}^{(i)}(z_j)) \|\bar{s}^{(l)^{-1}}(z_j)\| + O(\mu)\kappa(\bar{S}_{l+1}^{(i)}(z_j)) \|\bar{s}^{(l)^{-1}}(z_j)\|, \\
&\quad \leq i \cdot \bar{\tau}^{(i-1)}(z_j) \|\bar{S}^{(i)}(z_j)\| + O(\mu)\, i\, \bar{\tau}^{(i-1)}(z_j),
\end{aligned} \tag{6.49}$$

where we have used $i \cdot \bar{\tau}^{(i-1)}(z_j) = \max_{0 \leq l < i} \kappa(\bar{S}_{l+1}^{(i)}(z_j)) \, \|\bar{s}^{(l)^{-1}}(z_j)\|$ in (6.43) as an upper bound for $\sum_{l=0}^{i-1} \kappa(\bar{S}_{l+1}^{(i)}(z_j)) \|\bar{s}^{(l)^{-1}}(z_j)\|$. From Lemma 6.3, we have

$$\|\delta S^{(i)}(z_j)\| \leq \sum_{l=0}^{i-1} \|\bar{S}^{(l)}(z_j)\| \, \|\bar{s}^{(l+1)}(z_j) \cdots \bar{s}^{(i-1)}(z_j)\| \, \|\delta s^{\dagger^{(l)}}(z_j)\| + O(\mu^2), \tag{6.50}$$

where $\|\delta s^{\dagger^{(l)}}(z_j)\| \leq \lambda_{t_l+3}$. It follows from (6.49) and (6.50) that

$$\frac{\|\delta S^{(i)}(z_j)\|}{\|\bar{S}^{(i)}(z_j)\|} \leq i \cdot \bar{\tau}^{(i-1)}(z_j) \cdot \max_{0 \leq l \leq i-1} \lambda_{t_l+3} + O(\mu^2), \quad j = n_i + 1, \ldots, n_i + t_i. \tag{6.51}$$

Since

$$\bar{\tau}^{(i-1)}(z_{n_i+1}) \leq \tau, \tag{6.52}$$

then the bound (6.51) at z_{n_i+1} becomes

$$\frac{\|\delta S^{(i)}(z_j)\|}{\|\bar{S}^{(i)}(z_j)\|} \leq i \cdot \tau \cdot \max_{0 \leq l \leq i-1} \lambda_{t_l+3} + O(\mu^2). \tag{6.53}$$

For the remaining points $z_{n_i+2}, \ldots, z_{n_i+t_i}$, from (6.44) and (6.52), (6.51) becomes

$$\frac{\|\delta S^{(i)}(z_j)\|}{\|\bar{S}^{(i)}(z_j)\|} \leq i \cdot \tau \cdot \bar{\psi}_j \cdot \max_{0 \leq l \leq i-1} \lambda_{t_l+3} + O(\mu^2), \quad j = n_i + 1, \ldots, n_i + t_i. \tag{6.54}$$

□

Theorem 6.3 *The residual error* $(\, \bar{w}_j \quad \bar{r}_j \,)^{(i)}$ *in (6.33) satisfies*

$$\frac{\|\,(\, \delta r_j^{(i)} \quad \delta w_j^{(i)} \,)\,\|}{\|\,(\, \bar{r}_j^{(i)} \quad \bar{w}_j^{(i)} \,)\,\|} \leq i \cdot \tau \cdot \bar{\psi}_j \cdot \max_{0 \leq l \leq i-1} \lambda_{t_l+5} + O(\mu^2) \quad j = n_i + 1, \ldots, n_i + t_i. \tag{6.55}$$

Proof: From (6.30)

$$\begin{aligned} (\, \bar{w}_j \quad \bar{r}_j \,)^{(i)} &= fl\left((\, g_j \quad f_j \,)\, \bar{S}^{(i)}(z_j) \right), \quad j = n_i + 1, \ldots, n_i + t_i, \\ &= (\, g_j \quad f_j \,)\, \bar{S}^{(i)}(z_j) {<}2{>}, \quad j = n_i + 1, \ldots, n_i + t_i, \end{aligned} \tag{6.56}$$

where $<2>$ accounts for the error made when multiplying $(\, g_j \quad f_j \,)$ by $\bar{S}^{(i)}(z_j)$ of order 2. Similar to the proof of Lemma 6.3, with (6.7), we apply Corollary 6.1 to $\bar{S}^{(i)}(z_j)$ which in turn applies to each $\bar{s}^{(l)}(z_j)$ in the summation of the proof in Lemma 6.3, i.e.,

$$\bar{s}^{(l)}(z_j){<}2{>}{<}2{>} = \bar{s}^{(l)}(z_j){<}4{>} = s^{(l)}(z_j) + \delta s^{\ddagger^{(l)}}(z_j), \tag{6.57}$$

where $\|\delta s^{\ddagger^{(l)}}(z_j)\| \leq \lambda_{t_l+5}$. So that

$$\bar{S}^{(i)}(z_j){<}2{>} = S^{(i)}(z_j) + \delta S^{*^{(i)}}(z_j) \tag{6.58}$$

where

$$\delta S^{*^{(i)}}(z_j) = \sum_{l=0}^{i-1} \bar{S}^{(l)}(z_j) \delta s^{\ddagger^{(l)}}(z_j) \bar{s}^{(l+1)}(z_j) \cdots \bar{s}^{(i-1)}(z_j) + O(\mu^2). \tag{6.59}$$

With (6.58), (6.56) becomes

$$\begin{aligned} (\, \bar{w}_j \quad \bar{r}_j \,)^{(i)} &= (\, g_j \quad f_j \,)\, (S^{(i)}(z_j) + \delta S^{*^{(i)}}(z_j)), \quad j = n_i + 1, \ldots, n_i + t_i, \\ &= (\, w_j \quad r_j \,)^{(i)} + (\, \delta w_j \quad \delta r_j \,)^{(i)} \quad j = n_i + 1, \ldots, n_i + t_i, \end{aligned} \tag{6.60}$$

where

$$(\, \delta w_j \quad \delta r_j \,)^{(i)} = \sum_{l=0}^{i-1} (\, g_j \quad f_j \,)\, \bar{S}^{(l)}(z_j) \delta s^{\ddagger^{(l)}}(z_j) \bar{s}^{(l+1)}(z_j) \cdots \bar{s}^{(i-1)}(z_j) + O(\mu^2). \tag{6.61}$$

So that

$$\|\,(\, \delta w_j \quad \delta r_j \,)^{(i)}\,\| \leq \sum_{l=0}^{i-1} \|\,(\, g_j \quad f_j \,)\, \bar{S}^{(l)}(z_j)\| \, \|\delta s^{\ddagger^{(l)}}(z_j)\| \, \|\bar{s}^{(l+1)}(z_j) \cdots \bar{s}^{(i-1)}(z_j)\| + O(\mu^2). \tag{6.62}$$

From (4.22), we can write

$$\bar{S}_a^{(i)}(z_j) = \bar{s}^{(a)}(z_j) \cdots \bar{s}^{(i-1)}(z_j) + O(\mu) \tag{6.63}$$

so that for $l < a$,

$$\bar{S}_l^{(i)}(z_j) = \bar{s}^{(l)}(z_j) \cdots \bar{s}^{(a-1)}(z_j)\bar{S}_a^{(i)}(z_j) + O(\mu), \tag{6.64}$$

where $O(\mu)$ account for at most i multiplications of matrices of order 2. Then (6.62) becomes

$$\begin{aligned}
\| (\, \delta w_j \quad \delta r_j \,)^{(i)} \| &\leq \sum_{l=0}^{i-1} \| (\, g_j \quad f_j \,)\, \bar{S}^{(l)}(z_j) \| \, \| \bar{s}^{(l+1)}(z_j) \cdots \bar{s}^{(i-1)}(z_j) \| \, \| \delta s^{\ddagger^{(l)}}(z_j) \| + O(\mu^2) \\
&= \sum_{l=0}^{i-1} \| (\, g_j \quad f_j \,)\, \bar{S}^{(l)}(z_j) [\bar{s}^{(l)}(z_j) \bar{S}_{l+1}^{(i)}(z_j)] \cdot [\bar{s}^{(l)}(z_j) \bar{S}_{l+1}^{(i)}(z_j)]^{-1} \| \cdot \\
&\qquad \cdot \| \bar{s}^{(l+1)}(z_j) \cdots \bar{s}^{(i-1)}(z_j) \| \, \| \delta s^{\ddagger^{(l)}}(z_j) \| + O(\mu^2) \\
&\leq \sum_{l=0}^{i-1} \| (\, \bar{w}_j \quad \bar{r}_j \,)^{(i)} + O(\mu) \| \, \| \bar{s}^{(l)^{-1}}(z_j) \| \, \| \bar{S}_{l+1}^{(i)^{-1}}(z_j) \| \cdot \\
&\qquad \cdot (\| \bar{S}_{l+1}^{(i)}(z_j) \| + O(\mu) \|) \, \| \delta s^{\ddagger^{(l)}}(z_j) \| + O(\mu^2) \\
&\leq \sum_{l=0}^{i-1} \| (\, \bar{w}_j \quad \bar{r}_j \,)^{(i)} \| \, \| \delta s^{\ddagger^{(l)}}(z_j) \| \, \kappa(\bar{S}_{l+1}^{(i)}(z_j)) \, \| \bar{s}^{(l)^{-1}}(z_j) \| + O(\mu^2).
\end{aligned} \tag{6.65}$$

With $\bar{\tau}^{(i-1)}(z_j) = \max_{0 \leq l < i} \kappa(\bar{S}_{l+1}^{(i)}(z_j)) \, \| \bar{s}^{(l)^{-1}}(z_j) \|$, we have

$$\frac{\| (\, \delta w_j \quad \delta r_j \,)^{(i)} \|}{\| (\, \bar{w}_j \quad \bar{r}_j \,)^{(i)} \|} \leq i \cdot \bar{\tau}^{(i-1)}(z_j) \, \| \delta s^{\ddagger^{(l)}}(z_j) \| + O(\mu^2). \tag{6.66}$$

Since

$$\bar{\tau}^{(i-1)}(z_{n_i+1}) \leq \tau, \tag{6.67}$$

then the bound (6.66) at z_{n_i+1} becomes

$$\frac{\| (\, \delta w_j \quad \delta r_j \,)^{(i)} \|}{\| (\, \bar{w}_j \quad \bar{r}_j \,)^{(i)} \|} \leq i \cdot \tau \cdot \max_{0 \leq l \leq i-1} \lambda_{t_l+5} + O(\mu^2). \tag{6.68}$$

For the remaining points $z_{n_i+2}, \ldots, z_{n_i+t_i}$, from (6.44) and (6.67), (6.66) becomes

$$\frac{\| (\, \delta w_j \quad \delta r_j \,)^{(i)} \|}{\| (\, \bar{w}_j \quad \bar{r}_j \,)^{(i)} \|} \leq i \cdot \tau \cdot \bar{\psi}_j \cdot \max_{0 \leq l \leq i-1} \lambda_{t_l+5} + O(\mu^2), \quad j = n_i + 1, \ldots, n_i + t_i. \tag{6.69}$$

□

6.2.2 Interpolation of the Residuals

In this section, we present the computation of $s^{(i)}(z) = \begin{pmatrix} 1 & 0 \\ 0 & \theta^{(i)}(z) \end{pmatrix} s'^{(i)}(z)$ which interpolates the residual $(\, \bar{w}_j^{(i)} \quad \bar{r}_j^{(i)} \,)$ for $j = n_i + 1, \ldots, n_i + t_i$. We first normalize the residual by

$$\bar{\alpha}_j^{(i)} = \frac{1}{\| (\, \bar{w}_j^{(i)} \quad \bar{r}_j^{(i)} \,) \|}, \tag{6.70}$$

so that $\|\bar{\alpha}_j^{(i)} (\, \bar{w}_j^{(i)} \quad \bar{r}_j^{(i)} \,) \| = 1$. (If $\| (\, \bar{w}_j^{(i)} \quad \bar{r}_j^{(i)} \,) \| = 0$, we set $\bar{\alpha}_j^{(i)} = 1$). We then proceed by considering two cases: $z_j \in C^{(i)}$ where $|\bar{\alpha}_j^{(i)} \bar{w}_j^{(i)}| \leq \tau\mu$ and $z_j \notin C^{(i)}$ where $|\bar{\alpha}_j^{(i)} \bar{w}_j^{(i)}| > \tau\mu$.

For the first case where $|\bar{\alpha}_j^{(i)} \bar{w}_j^{(i)}| \leq \tau\mu$, we include z_j in $C^{(i)}$, and hence the $\theta^{(i)}(z)$ function is constructed.

For the second case where $z_j \notin C^{(i)}$, we use Gaussian elimination to solve the two linear systems of equations

$$\theta^{(i)}(z_j)\bar{r}_j^{(i)} v'^{(i)}(z_j) + \bar{w}_j^{(i)} u'^{(i)}(z_j) = 0, \quad j = n_i + 1, \ldots, n_i + t_i, z_j \notin C^{(i)}, \tag{6.71}$$

and

$$\theta^{(i)}(z_j)\bar{r}_j^{(i)} q'^{(i)}(z_j) + \bar{w}_j^{(i)} p'^{(i)}(z_j) = 0, \quad j = n_i + 1, \ldots, n_i + t_i - 1, z_j \notin C^{(i)}, \tag{6.72}$$

to obtain $s'^{(i)}(z)$, where

$$s'^{(i)}(z) = \begin{pmatrix} u'^{(i)}(z) & (z - z_{n_{i+1}})p'^{(i)}(z) \\ v'^{(i)}(z) & (z - z_{n_{i+1}})q'^{(i)}(z) \end{pmatrix}.$$

Note that in (6.72), if $t_i = 1$, $p'^{(i)}(z) = 1$ and $q'^{(i)}(z) = 0$.

Theorem 6.4 *If the computed $s'^{(i)}(z)$ is obtained by solving (6.71) and (6.72) using Gaussian elimination with complete pivoting, then*

$$\frac{\bar{\alpha}_j^{(i)} (\, \bar{w}_j \quad \theta^{(i)}(z_j)\bar{r}_j \,)^{(i)} s'^{(i)}(z_j)}{\|s'^{(i)}(z)\|} = (\, E_{w_j} \quad E_{r_j} \,), \quad j = n_i + 1, \ldots, n_i + t_i, z_j \notin C^{(i)}, \tag{6.73}$$

where $\| (\, E_{w_j} \quad E_{r_j} \,) \| \leq 2(t_i^4 + 3t_i^3)\rho_i\mu$, and ρ_i is a constant of order unity in practice.

Proof: Since $(p'^{(i)}(z), q'^{(i)}(z))$ interpolates one point fewer than $(u'^{(i)}(z), v'^{(i)}(z))$ so that the corresponding system of equations is a subset of that for solving $(u'^{(i)}(z), v'^{(i)}(z))$, we only show the analysis of $(u'^{(i)}(z), v'^{(i)}(z))$.

There are two possibilities of degree conditions: $\deg(u'^{(i)}) = \deg(p'^{(i)}(z))+1$, $\deg(v'^{(i)}(z)) = \deg(q'^{(i)}(z))$ or $\deg(u'^{(i)}(z)) = \deg(p'^{(i)}(z))$, $\deg(v'^{(i)}(z)) = \deg(q'^{(i)}(z))+1$. We only show the first one; the other one is similar.

The interpolant $(u'^{(i)}(z), v'^{(i)}(z))$ interpolates the point z_j, for $j = n_i + 1, \dots, n_i + t_i$, $z_j \notin C^{(i)}$, and they are obtained by solving the system:

$$M^{(i)} \cdot x^{(i)} = 0 \tag{6.74}$$

where $M^{(i)} = \mathrm{diag}(\bar{\alpha}^{(i)}_{n_i+1}, \bar{\alpha}^{(i)}_{n_i+2}, \dots, \bar{\alpha}^{(i)}_{n_i+t_i})\cdot$

$$\cdot \begin{pmatrix} \bar{w}_{n_i+1} & \cdots & \bar{w}_{n_i+1} z^{l}_{n_i+1} {<l>} & \theta(z_{n_i+1})\bar{r}_{n_i+1} {<2a>} & \cdots & \theta(z_{n_i+1})\bar{r}_{n_i+1} z^{l-1}_{n_i+1} {<2a+l-1>} \\ \vdots & & \vdots & \vdots & & \vdots \\ \bar{w}_{n_i+t_i} & \cdots & \bar{w}_{n_i+t_i} z^{l}_{n_i+t_i} {<l>} & \theta(z_{n_i+t_i})\bar{r}_{n_i+t_i} {<2a>} & \cdots & \theta(z_{n_i+t_i})\bar{r}_{n_i+t_i} z^{l-1}_{n_i+t_i} {<2a+l-1>} \end{pmatrix}^{(i)}, \tag{6.75}$$

$x^{(i)} = (\, u'^{(i)}_0 \dots u'^{(i)}_l \; v'^{(i)}_0 \dots v'^{(i)}_{l-1}\,)^t$, $l = \lfloor \frac{t_i}{2} \rfloor$, a is the degree of $\theta^{(i)}(z)$ and the $<\cdot>$ accounts for the error made in constructing $M^{(i)}$.

The system (6.74) is solved first by reducing $M^{(i)}$ to an upper triangular form with complete pivoting, next by assigning the last variable to one (or should a zero pivot be encountered, by assigning one to the variable corresponding to the zero pivot and one to the rest of the variables in the solution vector that are below the index of the pivot), and finally, by back substituting for the remaining variables to obtain $x^{(i)}$. This procedure guarantees a solution for the i^{th} iteration even for a singular system. It yields $x^{(i)}$ satisfying exactly

$$(M^{(i)} + \delta M^{(i)})x^{(i)} = 0, \tag{6.76}$$

where

$$\|\delta M^{(i)}\| \le 1.01(t_i^3 + 3t_i^2)\rho_i \mu \|M^{(i)}\|^1$$

and ρ_i is the growth factor[2] associated with LU-decomposition of $M^{(i)}$ [25]. From the above equation, we have

$$M^{(i)} \cdot x^{(i)} = -\delta M^{(i)} \cdot x^{(i)}, \tag{6.77}$$

where

$$\|\delta M^{(i)} \cdot x^{(i)}\| \le 2(t_i^3 + 3t_i^2)\mu \, \|M^{(i)}\| \, (\|u'^{(i)}(z)\| + \|v'^{(i)}(z)\|).$$

Since $z_j \in [-1, 1]$, $max\{|\bar{\alpha}^{(i)}_j \bar{r}^{(i)}_j|, |\bar{\alpha}^{(i)}_j \bar{w}^{(i)}_j|\} = 1$ and $\|\theta(z)^{(i)}\| = 1$, it is easy to see that $\|M^{(i)}\| \le t_i \max(|\theta(z_j)|) \max\{|\bar{\alpha}^{(i)}_j \bar{r}^{(i)}_j|, \, |\bar{\alpha}^{(i)}_j \bar{w}^{(i)}_j|\} \le t_i$. (We have implicitly assumed that $|z_j {<\cdot>}| \le 1$ for this result. It is not true for $z = \pm 1$, but we can restrict (i.e., by scaling) z such that this assumption is always true.) So,

$$\|\delta M^{(i)} \cdot x^{(i)}\| \le 2\mu(t_i^4 + 3t_i^3)(\|u'^{(i)}(z)\| + \|v'^{(i)}(z)\|) \tag{6.78}$$

[1] The result in [25] uses the ∞-norm, but the same result also applies to 1-norm.

[2] In practice, the magnitude of ρ_i is of unity.

or,

$$\frac{\|\delta M^{(i)} \cdot x^{(i)}\|}{(\|u'^{(i)}(z)\| + \|v'^{(i)}(z)\|)} \leq 2\mu(t_i^4 + 3t_i^3). \tag{6.79}$$

Thus, we can write the individual equation to be

$$\frac{\bar{\alpha}_j^{(i)}(\theta^{(i)}(z_j)\bar{r}_j^{(i)}v'^{(i)}(z_j) + \bar{w}_j^{(i)}u'^{(i)}(z_j))}{(\|u'^{(i)}(z)\| + \|v'^{(i)}(z)\|)} = E_{w_j}, \tag{6.80}$$

where E_{w_j} is the error introduced by using Gaussian elimination that is bounded by $|E_{w_j}| \leq 2(t_i^4 + 3t_i^3)\mu$. And the result follows. □

Note that we do not obtain the matrix $A^{(i)}$ from the matrix $M^{(i)}$ when solving the system of equations (6.71) as described in Chapter 5. Because we use Gaussian elimination with complete pivoting to solve the system of equations, in practice, using the matrix $M^{(i)}$ gives a more accurate solution than using the matrix $A^{(i)}$.

The condition number for the system (6.71), which we output as $\kappa^{(i)}$ in place of $\kappa(A^{(i)})$ is still $M^{(i)}$ with one column removed; the removed column corresponds to the last column of the matrix after complete pivoting. This column is selected because it corresponds to the column we move to the right hand side when solving for $(u'^{(i)}(z), v'^{(i)}(z))$.

Chapter 7

Continued-Fraction Representation

The use of a continued-fraction to represent a rational number has its root in the Thiele fraction [38, Chap. 9]. Many authors [33, 31, 60, 9, 63, 20] have used similar continued-fractions to represent rational functions. In this chapter, we introduce another continued-fraction representation. One reason for representing a rational interpolant in the continued-fraction form rather than the classical form (two polynomials, one over another, as in (1.3)) is that it gives a smaller condition number as we shall see in §9.6. Also, as we shall see in Chapter 8, by using this new representation, we can prove that Algorithm 4.2 is weakly stable.

In this chapter, as in Chapter 3, we first present the two-step divide-and-conquer representation of a continued-fraction form and discuss the condition under which unattainability occurs, and then we extend those results recursively.

Recall from Theorems 3.2 and 4.1 that the linear rational interpolant of type $[L, M]$ is given by

$$\begin{pmatrix} U(z) \\ V(z) \end{pmatrix} = \begin{pmatrix} u'(z) & p^{*'}(z) \\ \theta(z)v'(z) & \theta(z)q^{*'}(z) \end{pmatrix} \begin{pmatrix} \hat{u}'(z) \\ \hat{\theta}(z)\hat{v}'(z) \end{pmatrix}. \tag{7.1}$$

Lemma 7.1 *For the linear rational interpolant of type $[L, M]$ in (7.1), a continued fraction form is given by*

$$\frac{U(z)}{V(z)} = \frac{1}{\theta(z)v'(z)} \left(u'(z) + \frac{-\gamma t'_{0,n}(z)}{q^{*'}(z) + v'(z)\dfrac{\hat{u}'(z)}{\hat{\theta}(z)\hat{v}'(z)}} \right), \tag{7.2}$$

where $t'_{0,n}(z) = t_{0,n}(z)/\theta(z)$.

Proof: From (7.1), we have

$$\frac{U(z)}{V(z)} = \frac{u'(z)\hat{u}'(z) + p^{*'}(z)\hat{\theta}(z)\hat{v}'(z)}{\theta(z)(v'(z)\hat{u}'(z) + q^{*'}(z)\hat{\theta}(z)\hat{v}'(z))}$$

$$= \frac{u'(z)v'(z)\dfrac{\hat{u}'(z)}{\hat{\theta}(z)\hat{v}'(z)} + p^{*'}(z)v(z)}{\theta(z)v(z)(v'(z)\dfrac{\hat{u}'(z)}{\hat{\theta}(z)\hat{v}'(z)} + q^{*'}(z))}$$

$$= \frac{q^{*'}(z)u'(z) + u'(z)v'(z)\dfrac{\hat{u}'(z)}{\hat{\theta}(z)\hat{v}'(z)} + p^{*'}(z)v'(z) - q^{*'}(z)u'(z)}{\theta(z)v'(z)(v'(z)\dfrac{\hat{u}'(z)}{\hat{\theta}(z)\hat{v}'(z)} + q^{*'}(z))}$$

$$= \frac{u'(z)(q^{*'}(z) + v'(z)\dfrac{\hat{u}'(z)}{\hat{\theta}(z)\hat{v}'(z)}) - \gamma t'_{0,n}(z)}{\theta(z)v'(z)(v'(z)\dfrac{\hat{u}'(z)}{\hat{\theta}(z)\hat{v}'(z)} + q^{*'}(z))}, \tag{7.3}$$

since $p^{*'}(z)v'(z) - q^{*'}(z)u'(z) = -\gamma t_{0,n}(z)/\theta(z) = -\gamma t'_{0,n}(z)$. The result now follows. □

Note that the degree of (7.2) may exceed the given type $[L, M]$; in other words, expanding the continued-fraction by cross multiplying the denominators and numerators, one would find that (7.2) becomes

$$\frac{U(z)}{V(z)} = \frac{v'(z)(u'(z)\hat{u}'(z) + p^{*'}(z)\hat{\theta}(z)\hat{v}'(z))}{v'(z)(\theta(z)v'(z)\hat{u}'(z) + \theta(z)q^{*'}(z)\hat{\theta}(z)\hat{v}'(z))} = \frac{v'(z)U(z)}{v'(z)V(z)}. \tag{7.4}$$

However, upon cancellation of the common factor $v'(z)$, the degree type is indeed $[L, M]$.

In the continued-fraction form (7.2), we can see the importance of the $\theta(z)$ function, as illustrated in the example below.

Example 7.1 *The linear rational interpolant of type* $[2, 1]$ *for the data* $\{(1, f_0, 0),\ (2, 1, 1),\ (3, 2, 1),\ (4, -3, 1)\}$ *is*

$$\begin{pmatrix} U(z) \\ V(z) \end{pmatrix} = \begin{pmatrix} -3z+5 & (z-3)(z-2) \\ (z-1) & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 8 \end{pmatrix}. \tag{7.5}$$

Hence, the continued-fraction form is

$$\frac{U(z)}{V(z)} = \frac{1}{(z-1)}\left(-3z+5+\frac{(z-3)(z-2)}{0+\dfrac{1}{8}}\right), \tag{7.6}$$

where $\theta(z) = (z-1)$. *However, without the use of* $\theta(z)$, *we would have*

$$\frac{U(z)}{V(z)} = \frac{1}{(z-1)}\left(-3z+5+\frac{(z-3)(z-2)(z-1)}{0+(z-1)\dfrac{1}{8}}\right), \tag{7.7}$$

which at $z = 1$ *gives an undetermined* $0/0$ *result.*

The order of evaluation in (7.2) is from bottom right to top left (i.e., in (7.2), we start with the evaluation of $\hat{u}(z_j)/\hat{\theta}(z_j)\hat{v}(z_j))$. Problems arise in evaluating 0/0. In the following, we show this happens only if we encounter unattainable points.

Unattainability is defined through the linear solution (i.e., a point z_σ is an unattainable point if and only if $|U(z_\sigma)| + |V(z_\sigma)| = 0$). But it is the rational form that we are interested in. So we will develop the equivalence between the linear condition and the rational condition. For $n+1 \le \sigma \le N$, $|U(z_\sigma)| + |V(z_\sigma)| = 0$ if and only if $|\hat{u}(z_\sigma)| + |\hat{v}(z_\sigma)| = 0$ as given by Theorem 3.3. The following theorem relates unattainable points in the range of $0 \le \sigma \le n$.

Theorem 7.1 *Let*

$$A(z) = q^{*'}(z) + v'(z)\frac{\hat{u}'(z)}{\hat{\theta}(z)\hat{v}'(z)}. \tag{7.8}$$

For $\sigma = 0, \ldots, n$, z_σ *is an unattainable point with respect to* $[L, M]$ *if and only if*

$$\begin{cases} |u'(z_\sigma)| + |\theta(z_\sigma)v'(z_\sigma)| = 0, & \textit{when } \hat{v}'(z_\sigma) = 0, \\ (|u'(z_\sigma)| + |\theta(z_\sigma)v'(z_\sigma)|)\,|A(z_\sigma)| = 0, & \textit{when } \hat{v}'(z_\sigma) \neq 0 \ \&\ z_\sigma \notin C, \\ |u'(z_\sigma)A(z_\sigma) - \gamma t'_{0,n}(z_\sigma)| = 0, & \textit{when } \hat{v}'(z_\sigma) \neq 0 \ \&\ z_\sigma \in C. \end{cases} \tag{7.9}$$

Proof: If $\hat{v}'(z_\sigma) = 0$, $0 \le \sigma \le n$, then from (7.1),

$$\begin{aligned} \begin{pmatrix} U(z_\sigma) \\ V(z_\sigma) \end{pmatrix} &= \hat{u}'(z_\sigma)\begin{pmatrix} u'(z_\sigma) \\ \theta(z_\sigma)v'(z_\sigma) \end{pmatrix} + \hat{\theta}(z_\sigma)\hat{v}'(z_\sigma)\begin{pmatrix} p^{*'}(z_\sigma) \\ \theta(z)q^{*'}(z) \end{pmatrix} \\ &= \hat{u}'(z_\sigma)\begin{pmatrix} u'(z) \\ \theta(z_\sigma)v'(z_\sigma) \end{pmatrix}. \end{aligned} \tag{7.10}$$

Thus, $|U(z_\sigma)| + |V(z_\sigma)| = 0$ if and only if $|u'(z_\sigma)| + |\theta(z_\sigma)v'(z_\sigma)| = 0$ since $\hat{u}'(z_\sigma) \neq 0$ when $\hat{v}'(z_\sigma) = 0$.

Now if $\hat{v}'(z_\sigma) \neq 0$, from (7.1),

$$|U(z)| + |V(z)| = |u'(z)\hat{u}'(z) + p^{*'}(z)\hat{\theta}(z)\hat{v}'(z)| + |\theta(z)(v'(z)\hat{u}'(z) + q^{*'}(z)\hat{\theta}(z)\hat{v}'(z))|. \tag{7.11}$$

Similar to the proof of Lemma 7.1, we first multiply the two terms by $v'(z)/\hat{\theta}(z)\hat{v}'(z)$ and then add and subtract $q^{*'}(z)u'(z)$ in the first term only to get

$$\begin{aligned} \left|\frac{v'(z)}{\hat{\theta}(z)\hat{v}'(z)}\right| (|U(z)| + |V(z)|) &= |u'(z)(q^{*'}(z) + v'(z)\frac{\hat{u}'(z)}{\hat{\theta}(z)\hat{v}'(z)}) - \gamma t'_{0,n}(z)| + \\ & \quad |\theta(z)v'(z)(v'(z)\frac{\hat{u}'(z)}{\hat{\theta}(z)\hat{v}'(z)} + q^{*'}(z))|, \end{aligned} \tag{7.12}$$

or

$$|U(z)| + |V(z)| = \left[|u'(z)A(z) - \gamma t'_{0,n}(z)| + |\theta(z)v'(z)A(z)|\right]\left|\frac{\hat{\theta}(z)\hat{v}'(z)}{v'(z)}\right|. \tag{7.13}$$

Note that $\hat{v}'(z) \neq 0$; in particular, for the minimal step size case, $\hat{v}'(z) = 1$ (see Theorems 4.1 and 4.2). Now, for $z_\sigma \notin \mathcal{C}$, $t'_{0,n}(z_\sigma) = 0$, so z_σ is an unattainable point if and only if

$$\begin{aligned} |U(z_\sigma)| + |V(z_\sigma)| &= |u'(z_\sigma)A(z_\sigma)| + |\theta(z_\sigma)v'(z_\sigma)A(z_\sigma)| = 0, \\ &= (|u'(z_\sigma)| + |\theta(z_\sigma)v'(z_\sigma)|)|A(z_\sigma)| = 0. \end{aligned} \tag{7.14}$$

This follows because $\hat{\theta}(z_\sigma)\hat{v}'(z_\sigma) \neq 0$. (Note that $v'(z_\sigma) \neq 0$ for $0 \leq \sigma \leq n$ because of $\theta(z)$, see Theorem 4.1.) For $z_\sigma \in \mathcal{C}$, $\theta(z_\sigma) = 0$, so z_σ is an unattainable point if and only if

$$0 = |U(z_\sigma)| + |V(z_\sigma)| = |u'(z_\sigma)A(z_\sigma) - \gamma t'_{0,n}(z_\sigma)|. \tag{7.15}$$

□

By applying Lemma 7.1 recursively, the full continued-fraction form of (3.39) is given by

$$\frac{U(z)}{V(z)} = \frac{1}{\theta^{(0)}(z)v'^{(0)}(z)}\Big[u'^{(0)}(z) + \frac{-\gamma^{(0)}t'_{n_0+1,n_1}(z)}{q^{*'(0)}(z) + \frac{v'^{(0)}(z)}{\theta^{(1)}(z)v'^{(1)}(z)}\Big[u'^{(1)}(z) + \ddots \left\{\frac{v'^{(k-1)}(z)u'^{(k)}(z)}{\theta^{(k)}(z)v'^{(k)}(z)}\right\}}. \tag{7.16}$$

The generalization to this full continued-fraction form of the unattainability test (Theorem 7.1) is given in Corollary 7.1 below. The results of this corollary which relate to the continued fraction form are equivalent to that of Corollary 3.1 which relate to the linear form.

Corollary 7.1 *Let*

$$A^{(i)}(z) = q^{*'(i)}(z) + \frac{v'^{(i)}(z)}{\theta^{(i+1)}(z)v'^{(i+1)}(z)}\Big[u'^{(i+1)}(z) + \frac{-\gamma^{(i+1)}t'_{n_{i+1}+1,n_{i+2}}(z)}{q^{*'(i+1)}(z) + \ddots \left\{\frac{v'^{(k-1)}(z)u'^{(k)}(z)}{\theta^{(k)}(z)v'^{(k)}(z)}\right\}}. \tag{7.17}$$

For $\sigma = n_i + 1, \ldots, n_i + t_i$, for $i < k$, z_σ is an unattainable point with respect to $[L, M]$ if and only if

$$\begin{cases} |u'^{(i)}(z_\sigma)| + |\theta^{(i)}(z_\sigma)v'^{(i)}(z_\sigma)| = 0, & \textit{when } |A^{(i)}(z_\sigma)| = \infty, \\ (|u'^{(i)}(z_\sigma)| + |\theta^{(i)}(z_\sigma)v'^{(i)}(z_\sigma)|)\,|A^{(i)}(z_\sigma)| = 0, & \textit{when } |A^{(i)}(z_\sigma)| < \infty, \ \&\ z_\sigma \notin \mathcal{C}^{(i)}, \\ |u'^{(i)}(z_\sigma)A^{(i)}(z_\sigma) - \gamma t'_{n_i+1,n_i+t_i}(z_\sigma)| = 0, & \textit{when } |A^{(i)}(z_\sigma)| < \infty \ \&\ z_\sigma \in \mathcal{C}^{(i)}. \end{cases} \tag{7.18}$$

Proof: First, we let $s(z) = s^{(i)}(z)$ interpolating z_j, $j = n_i + 1, \ldots, n_i + t_i$, and

$$\begin{pmatrix} \hat{u}(z) \\ \hat{\theta}(z)\hat{v}(z) \end{pmatrix} = s^{(i+1)}(z) \cdots s^{(k-1)}(z) \begin{pmatrix} u'^{(k)}(z) \\ \theta^{(k)}(z)v'^{(k)}(z) \end{pmatrix} \tag{7.19}$$

interpolating z_j $j = n_i + 1, \ldots, N$. Then the result is a consequence of Theorem 7.1. □

The above Corollary gives us an efficient way to test whether a point z_σ is an unattainable point in its continued-fraction form. In the following chapters, the expression $A^{(i)}(z)$ in Corollary 7.1 is called the tail of the i^{th} iteration.

Chapter 8

Error Analysis of the Evaluation

In this chapter, we introduce a point-wise measure of error for rational interpolation. This measure accommodates a wide range of values including infinity. In the second section, we perform an error analysis on the evaluation with respect to this point-wise error bound. We then prove that Algorithm 4.2 is weakly stable.

8.1 Point-wise Error Measure

Once the linear rational interpolant pair $(U(z), V(z))$ is computed for the interpolation points $\{(z_j, f_j, g_j)\}_{j=0,\ldots,N}$, one might want to prove that the rational residual error

$$\frac{\left\| \begin{pmatrix} U(z_0)/V(z_0) + f_0/g_0 \\ U(z_1)/V(z_1) + f_1/g_1 \\ \vdots \\ U(z_N)/V(z_N) + f_N/g_N \end{pmatrix} \right\|}{\left\| \begin{pmatrix} f_0/g_0 \\ f_1/g_1 \\ \vdots \\ f_N/g_N \end{pmatrix} \right\|} \tag{8.1}$$

is small. This is a good measure of the size of residual errors if all the function values f_j/g_j are of the same order of magnitude, since any norm $\|\cdot\|$ used would tend to place more importance on the large values than the smaller values. However, if there is a wide range of function values in a given data set, this measure of the size of the residual error is not insightful in the sense that it places little significance on the small values.

This is not to say that the above measure using the vector norm is not useful. In fact, if all the function values are of a certain order and only a few are extremely small, in practice, one would require a higher accuracy with the large numbers and lower accuracy with the small ones. This is because relative importance is usually placed on the larger numbers. On the other hand, if all the function values are of a certain order and only a few are extremely large, then this would not be a desirable situation, since the majority of the function values would be ignored. But this could happen in a data set. In fact, one of the reasons we prefer rational interpolation over polynomial interpolation is that rational interpolation can interpolate poles (or, at least very large values). Hence, there is a need to develop a more meaningful way to measure how well we interpolate a given set of data.

In rational interpolation, if $U(z_j)/V(z_j)$ is the computed rational function that approximates $-f_j/g_j$ at the point z_j, then

$$\left|\frac{U(z_j)}{V(z_j)}+\frac{f_j}{g_j}\right| \tag{8.2}$$

is the absolute point-wise error and

$$\frac{\left|\frac{U(z_j)}{V(z_j)}+\frac{f_j}{g_j}\right|}{\left|\frac{f_j}{g_j}\right|} \tag{8.3}$$

is the point-wise relative error. This relative error is undefined at $f_j/g_j = 0$ and $f_j/g_j = \infty$ (where $g_j = 0$). One of our objectives is to define a new point-wise error measure which overcomes problems with $f_j/g_j = 0$ and $f_j/g_j = \infty$.

Definition 8.1 *Given the computed rational interpolant $U(z_j)/V(z_j)$ that approximates $-f_j/g_j$, the point-wise pseudo-error is defined to be*

$$E(f_j,g_j,U(z_j),V(z_j)) = \frac{\left|\frac{U(z_j)}{V(z_j)}+\frac{f_j}{g_j}\right|}{1+\left|\frac{U(z_j)}{V(z_j)}\right|}|g_j|. \tag{8.4}$$

Let us examine this measure E closely. First, we note that if $|f_j| \le |g_j|$, then, because of the normalization $|g_j| = 1$, (8.4) becomes

$$E(f_j,g_j,U(z_j),V(z_j)) = \frac{\left|\frac{U(z_j)}{V(z_j)}+\frac{f_j}{g_j}\right|}{1+\left|\frac{U(z_j)}{V(z_j)}\right|}. \tag{8.5}$$

For relatively large $|U(z_j)/V(z_j)|$, this measure of error is close to the relative error. On the other hand, for small $|U(z_j)/V(z_j)| \ll 1$, this measure of error is close to the absolute error. For most applications, this is desirable; if we think of $U(z_j)/V(z_j)$, $j = 0,\ldots,N$, as a vector then any norm for this vector, like E, downplays the significance of small values of $U(z_j)/V(z_j)$. The idea of adding one to the relative base in the denominator is not new; in fact, many numerical subroutines use this idea.

Second, if $1 = |f_j| > |g_j|$, observe that (8.4) can be rewritten as

$$E(f_j, g_j, U(z_j), V(z_j)) = \frac{\left|\frac{V(z_j)}{U(z_j)} + \frac{g_j}{f_j}\right|}{1 + \left|\frac{U(z_j)}{V(z_j)}\right|} |f_j|, \tag{8.6}$$

or, simply as

$$E(f_j, g_j, U(z_j), V(z_j)) = \frac{\left|\frac{V(z_j)}{U(z_j)} + \frac{g_j}{f_j}\right|}{1 + \left|\frac{U(z_j)}{V(z_j)}\right|}. \tag{8.7}$$

In other words, for large values of $U(z_j)/V(z_j) \geq 1$, E is a measure of the reciprocal error. Again this is desirable; for large $U(z_j)/V(z_j)$, we are willing to accept a large absolute error $U(z_j)/V(z_j) + f_j/g_j$ to the same proportion that we are willing to accept a small absolute error of the reciprocals $V(z_j)/U(z_j) + g_j/f_j$.

Note that (8.4) can also be written as

$$\begin{aligned} E(f_j, g_j, U(z_j), V(z_j)) &= \frac{\left|\frac{U(z_j)}{V(z_j)} + \frac{f_j}{g_j}\right|}{1 + \left|\frac{U(z_j)}{V(z_j)}\right|} |g_j| \qquad (8.8) \\ &= \frac{|g_j U(z_j) + f_j V(z_j)|}{|U(z_j)| + |V(z_j)|}. \qquad (8.9) \end{aligned}$$

So, $E(f_j, g_j, U(z_j), V(z_j))$ corresponds to the residual error $|g_j U(z_j) + f_j V(z_j)|$ normalized by $|U(z_j)| + |V(z_j)|$ and $\|\,(\,f_j \quad g_j\,)\,\|$ (by our normalization $\|\,(\,f_j \quad g_j\,)\,\| = 1$). Thus, we see that a small residual error at z_j does not imply a small pseudo-error. To achieve a small pseudo-error, $|U(z_j)| + |V(z_j)|$ must be large, i.e., z_j must not be nearly-unattainable.

8.2 Error Analysis of the Evaluation

In this section, we translate the residual error bounds from the linear solution to bounds of the pseudo-error of the rational continued-fraction form. In other words, with the solution of Algorithm 4.2 $U^{(k+1)}(z)/V^{(k+1)}(z)$, we want to obtain a bound for the pseudo-error

$$E(f_j, g_j, U^{(k+1)}(z_j), V^{(k+1)}(z_j)) = \frac{\left|\frac{U^{(k+1)}(z_j)}{V^{(k+1)}(z_j)} + \frac{f_j}{g_j}\right|}{1 + \left|\frac{U^{(k+1)}(z_j)}{V^{(k+1)}(z_j)}\right|} |g_j|, \quad j = 0, \ldots, N, \tag{8.10}$$

where $g_j \neq 0$. For $g_j = 0$ the reciprocal form is needed for the analysis, but since the reciprocal forms of (8.6) and (8.10) are equivalent, we only use the form of (8.10) for the analysis (i.e., we assume without loss of generality that $g_j \neq 0$).

In the following analysis, we take advantage of a certain observation simplifies our presentation.

Given the continued-fraction form of the solution from Algorithm 4.2,

$$\frac{U^{(k+1)}(z)}{V^{(k+1)}(z)} = \tag{8.11}$$

$$\frac{1}{V^{(i)}(z)} \left(U^{(i)}(z) + \frac{-\det(S^{(i)}(z))}{Q^{*(i)}(z) + V^{(i)}(z) \frac{1}{v^{(i)}(z)} \left(u'^{(i)}(z) + \frac{-\gamma^{(i)} t'_{n_i+1,n_{i+1}}(z)}{A^{(i)}(z)} \right)} \right),$$

where as in Corollary 7.1 $A^{(i)}(z)$ represents the tail of the i^{th} iteration. (Note that due to lack of space, we use $v^{(i)}(z) = \theta^{(i)}(z) v'^{(i)}(z)$ here and in the following continued-fraction forms).

For $z_j \notin C^{(i)}$, $j = n_i + 1, \ldots, n_{i+1}$,

$$\frac{U^{(k+1)}(z_j)}{V^{(k+1)}(z_j)} = \frac{1}{V^{(i)}(z_j)} \left(U^{(i)}(z_j) + \frac{-\det(S^{(i)}(z_j))}{Q^{*(i)}(z_j) + V^{(i)}(z_j) \frac{u'^{(i)}(z_j)}{v^{(i)}(z_j)}} \right), \tag{8.12}$$

because $t'_{n_i+1,n_{i+1}}(z_j) = 0$, $j = n_i + 1, \ldots, n_{i+1}$. In other words,

$$\frac{U^{(k+1)}(z_j)}{V^{(k+1)}(z_j)} = \frac{U^{(i+1)}(z_j)}{V^{(i+1)}(z_j)}, \quad j = n_i + 1, \ldots, n_{i+1}, z_j \notin C^{(i)}. \tag{8.13}$$

For $z_j \in \mathcal{C}^{(i)}$, $j = n_i + 1, \ldots, n_{i+1}$, we proceed as follow. We first write (8.11) as

$$\frac{U^{(k+1)}(z)}{V^{(k+1)}(z)} = \frac{1}{V^{(i-1)}(z)}\Big(U^{(i-1)}(z) + \tag{8.14}$$

$$\frac{-\det(S^{(i-1)}(z))}{Q^{*(i-1)}(z) + \frac{V^{(i-1)}(z)}{v^{(i-1)}(z)}(u'^{(i-1)}(z) + \frac{-\gamma^{(i-1)}t'_{n_{i-1}+1,n_i}(z)}{q^{*'(i-1)}(z) + \frac{v'^{(i-1)}(z)}{v^{(i)}(z)}(u'^{(i)}(z) + \frac{-\gamma^{(i)}t'_{n_i+1,n_{i+1}}(z)}{A^{(i)}(z)})})}\Big) \tag{8.15}$$

where we have expanded also the $(i-1)^{th}$ iteration. Equivalently,

$$\frac{U^{(k+1)}(z)}{V^{(k+1)}(z)} = \frac{1}{V^{(i-1)}(z)}\Big(U^{(i-1)}(z) + \tag{8.16}$$

$$\frac{-\det(S^{(i-1)}(z))}{Q^{*(i-1)}(z) + \frac{V^{(i-1)}(z)}{v^{(i-1)}(z)}(u'^{(i-1)}(z) + \frac{-\theta^{(i)}(z)\gamma^{(i-1)}t'_{n_{i-1}+1,n_i}(z)A^{(i)}(z)}{\theta^{(i)}(z)q^{*'(i-1)}(z)A^{(i)}(z) + \frac{v'^{(i-1)}(z)}{v'^{(i)}(z)}(u'^{(i)}(z)A^{(i)}(z) + -\gamma^{(i)}t'_{n_i+1,n_{i+1}}(z))})}\Big).$$

Therefore,

$$\frac{U^{(k+1)}(z_j)}{V^{(k+1)}(z_j)} = \frac{1}{V^{(i-1)}(z_j)}\left(U^{(i-1)}(z_j) + \frac{-\det(S^{(i-1)}(z_j))}{Q^{*(i-1)}(z_j) + V^{(i-1)}(z_j)\frac{u'^{(i-1)}(z_j)}{v^{(i-1)}(z_j)}}\right), \tag{8.17}$$

because $\theta^{(i)}(z_j) = 0$ for $z_j \in \mathcal{C}^{(i)}$. In other words,

$$\frac{U^{(k+1)}(z_j)}{V^{(k+1)}(z_j)} = \frac{U^{(i)}(z_j)}{V^{(i)}(z_j)}, \quad j = n_i + 1, \ldots, n_{i+1}, z_j \in \mathcal{C}^{(i)}. \tag{8.18}$$

From (8.13) and (8.18), we can conclude that the pseudo-error $E(f_j, g_j, U^{(k+1)}(z_j), V^{(k+1)}(z_j))$, $j = 0, \ldots, N$, of (8.10) is equivalent to the pseudo-error $E(f_j, g_j, U^{(i+1)}(z_j), V^{(i+1)}(z_j))$ for $z_j \notin \mathcal{C}^{(i)}$ and the pseudo-error $E(f_j, g_j, U^{(i)}(z_j), V^{(i)}(z_j))$ for $z_j \in \mathcal{C}^{(i)}$, $j = n_i + 1, \ldots, n_i + t_i$, $0 \leq i \leq k$. In the following, we first present an error analysis for the case where $z_j \notin \mathcal{C}^{(i)}$ and followed by a discussion on unattainability. Then, we give the results for the case where $z_j \in \mathcal{C}^{(i)}$.

Let

$$\epsilon_j^{(i)} = \frac{r_j^{(i)}}{w_j^{(i)}} + \frac{u'^{(i)}(z_j)}{\theta^{(i)}(z_j)v'^{(i)}(z_j)} \tag{8.19}$$

denote the absolute residual error at iteration i. Then the local pseudo-error is

$$\begin{aligned} E(r_j^{(i)}, w_j^{(i)}, u'^{(i)}(z_j), \theta^{(i)}(z_j)v'^{(i)}(z_j)) &= \frac{|\epsilon_j^{(i)}|}{1 + \left| \dfrac{u'^{(i)}(z_j)}{\theta^{(i)}(z_j)v'^{(i)}(z_j)} \right|} |w_j^{(i)}| \\ &= \frac{|\epsilon_j^{(i)}| \cdot |\theta^{(i)}(z_j)v'^{(i)}(z_j)w_j^{(i)}|}{|\theta^{(i)}(z_j)v'^{(i)}(z_j)| + |u'^{(i)}(z_j)|}. \end{aligned} \tag{8.20}$$

Lemma 8.1 *For z_j, $j = n_i + 1, \ldots, n_{i+1}, z_j \notin \mathcal{C}^{(i)}$, the pseudo-error locally is bounded by*

$$\bar{\alpha}_j^{(i)} E(r_j^{(i)}, w_j^{(i)}, u'^{(i)}(z_j), \theta^{(i)}(z_j)v'^{(i)}(z_j)) \leq \frac{2\,i\,\tau\bar{\psi}_j \max_{0\leq l<i} \lambda_{t_l+5} + 2(t_i^4 + 3t_i^3)\mu}{|u'^{(i)}(z_j)| + |\theta^{(i)}(z_j)v'^{(i)}(z_j)|} + O(\mu^2). \tag{8.21}$$

Proof: With the normalization $\|u'^{(i)}(z)\| + \|\theta^{(i)}(z)v'^{(i)}(z)\| = 1$, it follows from (6.33) together with the first column of (6.73) in Theorem 6.4 that

$$\bar{\alpha}_j^{(i)}(w_j^{(i)}u'^{(i)}(z_j) + r_j^{(i)}\theta^{(i)}(z_j)v'^{(i)}(z_j)) = E_{w_j} - \bar{\alpha}_j^{(i)}(\delta w_j^{(i)}u'^{(i)}(z_j) + \delta r_j^{(i)}\theta^{(i)}(z_j)v'^{(i)}(z_j)), \tag{8.22}$$

for $j = n_i + 1, \ldots, n_i + t_i$, where $|E_{w_j}| \leq 2(t_i^4 + 3t_i^3)\mu$, and

$$\begin{aligned} |\bar{\alpha}_j^{(i)}(\delta w_j^{(i)}u'^{(i)}(z_j) + \delta r_j^{(i)}\theta^{(i)}(z_j)v'^{(i)}(z_j))| &\leq \frac{|\delta w_j^{(i)}u'^{(i)}(z_j)| + |\delta r_j^{(i)}\theta^{(i)}(z_j)v'^{(i)}(z_j)|}{\|\,(\,\bar{w}_j \quad \bar{r}_j\,)\,\|} \\ &\leq \frac{|\delta w_j^{(i)}| + |\delta r_j^{(i)}|}{\|\,(\,\bar{w}_j^{(i)} \quad \bar{r}_j^{(i)}\,)\,\|} \\ &\leq \frac{2\,\|\,(\,\delta r_j^{(i)} \quad \delta w_j^{(i)}\,)\,\|}{\|\,(\,\bar{w}_j^{(i)} \quad \bar{r}_j^{(i)}\,)\,\|} \\ &< 2\,i\,\tau\bar{\psi} \max_{0\leq l<i} \lambda_{t_l+5} + O(\mu^2), \end{aligned} \tag{8.23}$$

where we use Theorem 6.3 for the bound of the residual. Therefore, the bound for the local absolute error is

$$\begin{aligned} |\epsilon_j^{(i)}| &= \left| \frac{r_j^{(i)}}{w_j^{(i)}} + \frac{u'^{(i)}(z_j)}{\theta^{(i)}(z_j)v'^{(i)}(z_j)} \right| \\ &\leq \frac{2i\tau\bar{\psi}_j \max_{0\leq l<i} \lambda_{t_l+5} + 2(t_i^4 + 3t_i^3)\mu}{\bar{\alpha}_j^{(i)} \cdot |w_j^{(i)}\theta^{(i)}(z_j)v'^{(i)}(z_j)|} + O(\mu^2). \end{aligned} \tag{8.24}$$

Substituting (8.24) into (8.20) get

$$E(r_j^{(i)}, w_j^{(i)}, u'^{(i)}(z_j), \theta^{(i)}(z_j)v'^{(i)}(z_j)) \leq \frac{2\,i\,\tau\bar{\psi}_j \max_{0\leq l<i} \lambda_{t_l+5} + 2(t_i^4 + 3t_i^3)\mu}{\bar{\alpha}_j^{(i)}(|u'^{(i)}(z_j)| + |\theta^{(i)}(z_j)v'^{(i)}(z_j)|)} + O(\mu^2), \tag{8.25}$$

and the result follows. □

Remark 8.1 *For the first step (i.e. $i = 0$), the error is*

$$E(f_j, g_j, u'^{(0)}(z_j), \theta^{(0)}(z_j)v'^{(0)}(z_j)) = \frac{2(t_0^4 + 3t_0^3)\mu}{|u'^{(0)}(z_j)| + |\theta^{(0)}(z_j)v'^{(0)}(z_j)|},$$

which is error due to Gaussian elimination alone. If the initial step size happens to be $t_0 = N + 1$, then the method of solution corresponds to solving the entire system using Gaussian elimination. As can be seen from this, at nearly unattainable points z_σ (where $|u'^{(0)}(z_\sigma)| + |\theta^{(0)}(z_j)v'^{(0)}(z_\sigma)|$ is small), the bound for the pseudo-error using the Gaussian Elimination method can get arbitrarily large, as expected.

Lemma 8.2 *Consider the points z_j, $j = n_i + 1, \ldots, n_i + t_i$, $z_j \notin C^{(i)}$. If $|\epsilon_j^{(i)}| \leq 1$ and $|\epsilon_j^{(i)} w_j^{(i)}|$ is so small that*

$$\frac{1}{2}|\det(S^{(i)}(z_j))|\,(|f_j| + |g_j|) \geq |\epsilon_j^{(i)} w_j^{(i)}|\,(|U^{(i)}(z_j)| + |V^{(i)}(z_j)|),$$

then the global pseudo-error satisfies

$$E(f_j, g_j, U^{(i+1)}(z_j), V^{(i+1)}(z_j)) \leq 16\tau\bar{\psi}_j\bar{\alpha}_j^{(i)} E(r_j^{(i)}, w_j^{(i)}, u'^{(i)}(z_j), \theta^{(i)}(z_j)v'^{(i)}(z_j)) + O(\mu^2). \tag{8.26}$$

Proof: At the i^{th} iteration we have available $S^{(i)}(z) = s^{(0)}(z) \cdots s^{(i-1)}(z)$. So the exact rational interpolant satisfies

$$-\frac{f_j}{g_j} = \frac{1}{V^{(i)}(z_j)}\left(U^{(i)}(z_j) + \frac{-\det(S^{(i)}(z_j))}{Q^{*^{(i)}}(z_j) + V^{(i)}(z_j)\dfrac{-r_j^{(i)}}{w_j^{(i)}}}\right) \tag{8.27}$$

which implies

$$-\frac{r_j^{(i)}}{w_j^{(i)}} = \frac{1}{V^{(i)}(z_j)}\left(\frac{\det(S^{(i)}(z_j))}{V^{(i)}(z_j)\dfrac{f_j}{g_j} + U^{(i)}(z_j)} - Q^{*^{(i)}}(z_j)\right). \tag{8.28}$$

For computed counterpart, we have

$$\begin{aligned}
\frac{U^{(i+1)}(z_j)}{V^{(i+1)}(z_j)} &= \frac{1}{V^{(i)}(z_j)}\left(U^{(i)}(z_j) + \frac{-\det(S^{(i)}(z_j))}{Q^{*^{(i)}}(z_j) + V^{(i)}(z_j)\dfrac{u'^{(i)}(z_j)}{\theta^{(i)}(z_j)v'^{(i)}(z_j)}}\right) \\
&= \frac{1}{V^{(i)}(z_j)}\left(U^{(i)}(z_j) + \frac{-\det(S^{(i)}(z_j))}{Q^{*^{(i)}}(z_j) + V^{(i)}(z_j)(\dfrac{-r_j^{(i)}}{w_j^{(i)}} + \epsilon_j^{(i)})}\right).
\end{aligned} \tag{8.29}$$

Now, using (8.28) and the equation $U^{(i)}(z_j) + V^{(i)}(z_j)f_j/g_j = w_j^{(i)}/g_j$ in (8.29), we obtain

$$\frac{U^{(i+1)}(z_j)}{V^{(i+1)}(z_j)} = \frac{-\det(S^{(i)}(z_j))f_j/g_j + \epsilon_j^{(i)}U^{(i)}(z_j)\,w_j^{(i)}/g_j}{\det(S^{(i)}(z_j)) + \epsilon_j^{(i)}V^{(i)}(z_j)\,w_j^{(i)}/g_j}. \tag{8.30}$$

Thus,

$$\left|\frac{U^{(i+1)}(z_j)}{V^{(i+1)}(z_j)} + \frac{f_j}{g_j}\right| = \left|\frac{\epsilon_j^{(i)}(w_j^{(i)}/g_j)^2}{\det(S^{(i)}(z_j)) + \epsilon_j^{(i)}V^{(i)}(z_j)\,w_j^{(i)}/g_j}\right|, \tag{8.31}$$

and

$$\begin{aligned}
&E(f_j, g_j, U^{(i+1)}(z_j), V^{(i+1)}(z_j)) \\
&\quad\le \frac{|\epsilon_j^{(i)}|{w_j^{(i)}}^2}{|-\det(S^{(i)}(z_j))f_j + \epsilon_j^{(i)}U^{(i)}(z_j)\,w_j^{(i)}| + |g_j\det(S^{(i)}(z_j)) + \epsilon_j^{(i)}V^{(i)}(z_j)\,w_j^{(i)}|} \\
&\quad\le \frac{|\epsilon_j^{(i)}|{w_j^{(i)}}^2}{|\det(S^{(i)}(z_j))f_j| - |\epsilon_j^{(i)}U^{(i)}(z_j)\,w_j^{(i)}| + |g_j\det(S^{(i)}(z_j))| - |\epsilon_j^{(i)}V^{(i)}(z_j)\,w_j^{(i)}|} \\
&\quad\le \frac{|\epsilon_j^{(i)}|{w_j^{(i)}}^2}{|\det(S^{(i)}(z_j))|(|f_j| + |g_j|) - |\epsilon_j^{(i)}w_j^{(i)}|(|U^{(i)}(z_j) + |V^{(i)}(z_j)|}.
\end{aligned} \tag{8.32}$$

If $\frac{1}{2}|\det(S^{(i)}(z_j))|\,(|f_j| + |g_j|) \ge |\epsilon_j^{(i)}w_j^{(i)}|\,(|U^{(i)}(z_j)| + |V^{(i)}(z_j)|)$, then

$$\begin{aligned}
E(f_j, g_j, U^{(i+1)}(z_j), V^{(i+1)}(z_j)) &\le \frac{2|\epsilon_j^{(i)}|{w_j^{(i)}}^2}{|\det(S^{(i)}(z_j))|(|f_j| + |g_j|)} \\
&\le \frac{2|\epsilon_j^{(i)}|{w_j^{(i)}}^2}{|\det(S^{(i)}(z_j))|},
\end{aligned} \tag{8.33}$$

because $\max\{|f_j|, |g_j|\} = 1$. Substituting

$$\begin{aligned}
|\epsilon_j^{(i)}| &= E(r_j^{(i)}, w_j^{(i)}, u'^{(i)}(z_j), \theta^{(i)}(z_j)v'^{(i)}(z_j))\frac{1}{|w_j^{(i)}|}(1 + \left|\frac{u'^{(i)}(z_j)}{\theta^{(i)}(z_j)v'^{(i)}(z_j)}\right|) \\
&= E(r_j^{(i)}, w_j^{(i)}, u'^{(i)}(z_j), \theta^{(i)}(z_j)v'^{(i)}(z_j))\frac{1}{|w_j^{(i)}|}\frac{|\theta^{(i)}(z_j)v'^{(i)}(z_j)| + |u'^{(i)}(z_j)|}{|\theta^{(i)}(z_j)v'^{(i)}(z_j)|}
\end{aligned} \tag{8.34}$$

into (8.33) to obtain

$$\begin{aligned}
&E(f_j, g_j, U^{(i+1)}(z_j), V^{(i+1)}(z_j)) \\
&\quad\le \frac{2\bar{\alpha}_j^{(i)}E(r_j^{(i)}, w_j^{(i)}, u'^{(i)}(z_j), \theta^{(i)}(z_j)v'^{(i)}(z_j))|w_j^{(i)}|(|\theta^{(i)}(z_j)v'^{(i)}(z_j)| + |u'^{(i)}(z_j)|)}{\bar{\alpha}_j^{(i)}|\theta^{(i)}(z_j)v'^{(i)}(z_j)|\,|\det(S^{(i)}(z_j))|}.
\end{aligned} \tag{8.35}$$

Thus, with $|\epsilon_j^{(i)}| \le 1$ it follows that

$$\frac{|u'^{(i)}(z_j)| + |\theta^{(i)}(z_j)v'^{(i)}(z_j)|}{|\theta^{(i)}(z_j)v'^{(i)}(z_j)|} \le 2\frac{|r_j^{(i)}| + |w_j^{(i)}|}{|w_j^{(i)}|} \tag{8.36}$$

(see Remark 8.2 at the end of this chapter). With (8.36) and $\bar{\alpha}_j^{(i)} = 1/\max\{|\bar{r}_j^{(i)}|, |\bar{w}_j^{(i)}|\}$, (8.35) becomes

$$\begin{aligned} &E(f_j, g_j, U^{(i+1)}(z_j), V^{(i+1)}(z_j)) \\ &\quad\le \frac{4\bar{\alpha}_j^{(i)} E(r_j^{(i)}, w_j^{(i)}, u'^{(i)}(z_j), \theta^{(i)}(z_j)v'^{(i)}(z_j)) \max\{|\bar{w}_j^{(i)}|, |\bar{r}_j^{(i)}|\}(|w_j^{(i)}| + |r_j^{(i)}|)}{|\det(S^{(i)}(z_j))|}. \end{aligned} \tag{8.37}$$

But,

$$w_j^{(i)} = (g_j U^{(i)}(z_j) + f_j V^{(i)}(z_j)) \tag{8.38}$$

$$r_j^{(i)} = (g_j P^{*^{(i)}}(z_j) + f_j Q^{*^{(i)}}(z_j)) \tag{8.39}$$

which implies

$$\begin{aligned} |w_j^{(i)}| &\le \max\{|g_j|, |f_j|\}(|U^{(i)}(z_j)| + |V^{(i)}(z_j)|) \\ &= |U^{(i)}(z_j)| + |V^{(i)}(z_j)| \end{aligned} \tag{8.40}$$

$$\begin{aligned} |r_j^{(i)}| &\le \max\{|g_j|, |f_j|\}(|P^{*^{(i)}}(z_j)| + |Q^{*^{(i)}}(z_j)|) \\ &= |P^{*^{(i)}}(z_j)| + |Q^{*^{(i)}}(z_j)|. \end{aligned} \tag{8.41}$$

Thus,

$$\max\{|\bar{w}_j^{(i)}|, |\bar{r}_j^{(i)}|\} \le 2\,\|S^{(i)}(z_j)\|, \tag{8.42}$$

because $\max\{|\bar{w}_j^{(i)}|, |\bar{r}_j^{(i)}|\} \le 2\max\{|w_j^{(i)}|, |r_j^{(i)}|\}$. Also, from (8.40) and (8.41)

$$\begin{aligned} |w_j^{(i)}| + |r_j^{(i)}| &\le |U^{(i)}(z_j)| + |V^{(i)}(z_j)| + |P^{*^{(i)}}(z_j)| + |Q^{*^{(i)}}(z_j)| \\ &\le 2\|S^{(i)^{adj}}(z_j)\|. \end{aligned} \tag{8.43}$$

Substitution of (8.42) and (8.43) into (8.37) gives

$$\begin{aligned} &E(f_j, g_j, U^{(i+1)}(z_j), V^{(i+1)}(z_j)) \\ &\quad\le \frac{16\bar{\alpha}_j^{(i)} E(r_j^{(i)}, w_j^{(i)}, u'^{(i)}(z_j), \theta^{(i)}(z_j)v'^{(i)}(z_j))\|S^{(i)}(z_j)\|\,\|S^{(i)^{adj}}(z_j)\|}{|\det(S^{(i)}(z_j)|} \\ &\quad= 16\bar{\alpha}_j^{(i)} E(r_j^{(i)}, w_j^{(i)}, u'^{(i)}(z_j), \theta^{(i)}(z_j)v'^{(i)}(z_j))\kappa(S^{(i)}(z_j)) \\ &\quad= 16\bar{\alpha}_j^{(i)} E(r_j^{(i)}, w_j^{(i)}, u'^{(i)}(z_j), \theta^{(i)}(z_j)v'^{(i)}(z_j))(\kappa(\bar{S}^{(i)}(z_j)) + O(\mu)). \end{aligned} \tag{8.44}$$

With Lemma 8.1 and $\kappa(\bar{S}^{(i)}(z_j)) \le \tau\bar{\psi}_j$, the result follows. □

We now discuss the effect of unattainability. The above results rely on the fact that $t'_{n_i+1,n_{i+1}}(z_j) = 0$ for $j = n_i + 1, \ldots, n_i + t_i$, $z_j \notin C^{(i)}$ in (8.11). If we were to include the term $-\gamma^{(i)}t'_{n_i+1,n_{i+1}}(z_j)/A^{(i)}(z_j)$, which is first added to $u'^{(i)}(z_j)$ (see (8.11)), in our analysis, then instead of $\epsilon_j^{(i)}$, we would have $\epsilon_j^{(i)} + \epsilon_j^{*(i)}$, where

$$\epsilon_j^{*(i)} = \frac{-\gamma^{(i)}t'_{n_i+1,n_{i+1}}(z_j)}{\theta^{(i)}(z_j)v'^{(i)}(z_j)\,A^{(i)}(z_j)}, \tag{8.45}$$

as the local absolute error. Note that we can have difficulty only when the denominator of this expression precisely equals to zero, i.e., $A^{(i)}(z_j) = 0$, in which case, we have the calculation of 0/0. This $A^{(i)}(z_j)$ corresponds to the expression in Corollary 3.1 for checking unattainable points. We would not be able to compute this expression exactly. However, its magnitude reflects whether or not a point is nearly unattainable. Let us first convert this local absolute error into the local point-wise error:

$$\begin{aligned} \bar{\alpha}_j^{(i)} E^{*(i)}(r_j^{(i)}, w_j^{(i)}, u'^{(i)}(z_j), v^{(i)}(z_j)) &\leq \frac{|\epsilon_j^{*(i)}||\bar{\alpha}_j^{(i)} w_j^{(i)}\theta^{(i)}(z_j)v'^{(i)}(z_j)|}{|u'^{(i)}(z_j)| + |\theta^{(i)}(z_j)v'^{(i)}(z_j)|} \\ &= \frac{|\gamma^{(i)}t'_{n_i+1,n_{i+1}}(z_j)|\,|\bar{\alpha}_j^{(i)}w_j^{(i)}|}{A^{(i)}(z_j)(|u'^{(i)}(z_j)| + |\theta^{(i)}(z_j)v'^{(i)}(z_j)|)} \\ &= |\gamma^{(i)}t'_{n_i+1,n_{i+1}}(z_j)|\,\Omega_j \end{aligned} \tag{8.46}$$

where,

$$\Omega_j = \frac{|\bar{\alpha}_j^{(i)}w_j^{(i)}|}{A^{(i)}(z_j)(|u'^{(i)}(z_j)| + |\theta^{(i)}(z_j)v'^{(i)}(z_j)|)}. \tag{8.47}$$

If Ω_j is large, then we treat z_j as a numerical unattainable point. Thus, near z_j, we should expect a large error. However, z_j is still accurately interpolated linearly. Notice that this expression of Ω_j is given at the local level. One can apply Lemma 8.2 to convert these expressions to the global level. A discussion on the relative size of Ω_j is given in §9.3.

We now obtain a bound for $E(f_j, g_j, U^{(i)}(z_j), V^{(i)}(z_j))$ for the case $z_j \in \mathcal{C}^{(i)}$.

Lemma 8.3 *For $j = n_i + 1, \ldots, n_{i+1}, z_j \in \mathcal{C}^{(i)}$, the pseudo-error locally is bounded by*

$$\bar{\alpha}_j^{(i)} E(r_j^{(i-1)}, w_j^{(i-1)}, u'^{(i-1)}(z_j), v^{(i-1)}(z_j)) \leq \left(\frac{2\,i\,\tau\bar{\psi}_j \max_{0<l<i}\lambda_{t_l+5} + \tau\mu}{|u'^{(i-1)}(z_j)| + |\theta^{(i-1)}(z_j)v'^{(i-1)}(z_j)|}\right) + O(\mu^2). \tag{8.48}$$

Proof: From (6.33), we have

$$\bar{\alpha}_j^{(i)}w_j^{(i)} = \bar{\alpha}_j^{(i)}\bar{w}_j^{(i)} + \bar{\alpha}_j^{(i)}\delta w_j^{(i)}, \tag{8.49}$$

so that

$$|\bar{\alpha}_j^{(i)}w_j^{(i)}| \leq i\tau\bar{\psi}_j \max_{0\leq l<i} \lambda_{t_l+5} + \tau\mu + O(\mu^2), \tag{8.50}$$

since the first $\bar{\alpha}_j^{(i)}\bar{w}_j^{(i)}$ is bounded by $\tau\mu$ and, with Theorem 6.3, the second term is bounded by

$$\begin{aligned}
|\bar{\alpha}_j^{(i)}\delta w_j^{(i)}| &= \frac{|\delta w_j^{(i)}|}{\|\,(\,\bar{r}_j^{(i)} \quad \bar{w}_j^{(i)}\,)\,\|} \\
&\leq \frac{\|\,(\,\delta w_j^{(i)} \quad \delta r_j^{(i)}\,)\,\|}{\|\,(\,\bar{w}_j^{(i)} \quad \bar{r}_j^{(i)}\,)\,\|} \\
&\leq i\,\tau\,\bar{\psi}_j \max_{0\leq l<i} \lambda_{t_l+5} + O(\mu^2). \qquad (8.51)
\end{aligned}$$

Now, $w_j^{(i)} = w_j^{(i-1)}u^{(i-1)}(z_j) + r_j^{(i-1)}\theta^{(i-1)}(z_j)v^{(i-1)}(z_j)$ so that

$$\begin{aligned}
|\epsilon_j^{(i-1)}| &= \left| \frac{r_j^{(i-1)}}{w_j^{(i-1)}} + \frac{u^{(i-1)}(z_j)}{\theta^{(i-1)}(z_j)v^{(i-1)}(z_j)} \right| \\
&= \left| \frac{\bar{\alpha}_j^{(i)}w_j^{(i)}}{\bar{\alpha}_j^{(i)}w_j^{(i-1)}\theta^{(i-1)}(z_j)v^{(i-1)}(z_j)} \right| \\
&\leq \frac{i\,\tau\bar{\psi}_j \max_{0<l<i} \lambda_{t_l+5} + \tau\mu}{|\bar{\alpha}_j^{(i)}w_j^{(i-1)}\theta^{(i-1)}(z_j)v^{(i-1)}(z_j)|} + O(\mu^2). \qquad (8.52)
\end{aligned}$$

Substituting (8.52) into (8.20) get

$$\begin{aligned}
E^{(i-1)}(r_j^{(i-1)}, w_j^{(i-1)}, u'^{(i-1)}(z_j), \theta^{(i-1)}(z_j)v'^{(i-1)}(z_j)) \quad &\leq \\
\frac{1}{\bar{\alpha}_j^{(i)}} \left(\frac{i\,\tau\bar{\psi}_j \max_{0\leq l<i} \lambda_{t_l+5} + \tau\mu}{|u'^{(i-1)}(z_j)| + |\theta^{(i-1)}(z_j)v'^{(i-1)}(z_j)|} \right) &+ O(\mu^2), \qquad (8.53)
\end{aligned}$$

and the result follows. □

With Lemma 8.3, we have the following corollary of Lemma 8.2.

Corollary 8.1 *Consider the points z_j, $j = n_i + 1, \ldots, n_i + t_i$, $z_j \in \mathcal{C}^{(i)}$. If $|\epsilon_j^{(i-1)}| \leq 1$ and $|\epsilon_j^{(i-1)}w_j^{(i-1)}|$ is so small that*

$$\frac{1}{2}|\det(S^{(i-1)}(z_j))|\,(|f_j| + |g_j|) \geq |\epsilon_j^{(i-1)}w_j^{(i-1)}|\,(|U^{(i-1)}(z_j)| + |V^{(i-1)}(z_j)|),$$

then the global pseudo-error satisfies

$$E(f_j, g_j, U^{(i)}(z_j), V^{(i)}(z_j)) \leq 16\tau\bar{\psi}_j\bar{\alpha}_j^{(i)}E(r_j^{(i-1)}, w_j^{(i-1)}, u'^{(i-1)}(z_j), v^{(i-1)}(z_j)) + O(\mu^2). \qquad (8.54)$$

Proof: The proof is similar to the proof of Lemma 8.2. □

Thus, a similar point-wise error bound is attained for the case $z_j \in \mathcal{C}^{(i)}$.

Regarding unattainability, similar to the treatment of the case where $z_j \not\in C^{(i)}$, we obtain a local absolute error $\epsilon_j^{(i-1)} + \epsilon_j^{*^{(i-1)}}$, where

$$\epsilon_j^{*^{(i-1)}} = \frac{-\theta^{(i)}(z_j)\gamma^{(i-1)}t'_{n_{i-1}+1,n_i}(z_j)A^{(i)}(z_j)}{v^{(i-1)}(z_j)(\theta^{(i)}(z_j)q^{*'^{(i-1)}}(z_j) + \frac{v'^{(i-1)}(z_j)}{v'^{(i)}(z_j)}(u'^{(i)}(z_j)A^{(i)}(z_j) - \gamma^{(i)}t'_{n_i+1,n_{i+1}}(z_j)))}. \tag{8.55}$$

Notice that we might encounter difficulty when the denominator of this expression precisely equals to zero, i.e., $(u'^{(i)}(z_j)A^{(i)}(z_j) - \gamma^{(i)}t'_{n_i+1,n_{i+1}}(z_j)) = 0$, in which case, we have the calculation of 0/0. This $(u'^{(i)}(z_j)A^{(i)}(z_j) - \gamma^{(i)}t'_{n_i+1,n_{i+1}}(z_j))$ corresponds to the expression in Corollary 3.1 for checking unattainable points. We would not be able to compute this expression exactly. However, its magnitude reflects whether or not a point is near unattainable. Let us first convert this local absolute error into the local point-wise error:

$$\begin{aligned}\bar{\alpha}_j^{(i)}E^*(r_j^{(i-1)}, w_j^{(i-1)}, u'^{(i-1)}(z_j), v^{(i-1)}(z_j)) &\le \frac{\bar{\alpha}_j^{(i)}\,|\epsilon_j^{*^{(i-1)}}|\,|w_j^{(i-1)}\theta^{(i-1)}(z_j)v^{(i-1)}(z_j)|}{|u'^{(i-1)}(z_j)| + |\theta^{(i-1)}(z_j)v'^{(i-1)}(z_j)|} \\ &= \frac{|\theta^{(i)}(z_j)\gamma^{(i-1)}t'_{n_{i-1}+1,n_i}(z_j)|\,|\bar{\alpha}_j^{(i)}w_j^{(i-1)}|}{(u'^{(i)}(z_j)A^{(i)}(z_j) - \gamma^{(i)}t'_{n_i+1,n_{i+1}}(z_j))(|u'^{(i-1)}(z_j)| + |\theta^{(i-1)}((z_j)v'^{(i-1)}(z_j)|)} \\ &= |\theta^{(i)}(z_j)\gamma^{(i-1)}t'_{n_{i-1}+1,n_i}(z_j)|\,\Omega_j,\end{aligned} \tag{8.56}$$

where

$$\Omega_j = \frac{|\bar{\alpha}_j^{(i)}w_j^{(i-1)}|}{(u'^{(i)}(z_j)A^{(i)}(z_j) - \gamma^{(i)}t'_{n_i+1,n_{i+1}}(z_j))(|u'^{(i-1)}(z_j)| + |\theta^{(i-1)}(z_j)v'^{(i-1)}(z_j)|)}. \tag{8.57}$$

If Ω_j is large, then we treat z_j as a numerical unattainable point. Notice that this expression of Ω_j is given at the local level. One can apply Corollary 8.1 to convert these expressions to the global level. A discussion on the relative size of Ω_j is given in §9.3.

We now summarize the principal result of this thesis in Theorems 8.1 and 8.2 below.

Theorem 8.1 *If the conditions of Lemma 8.2 and Corollary 8.1 are satisfied, then the pseudo-error of z_j, for each $i \le k$, $j = n_i + 1, \ldots, n_i + t_i$, is bounded by*

$$E(f_j, g_j, U^{(k+1)}(z_j), V^{(k+1)}(z_j)) \le \frac{32i\tau^2\bar{\psi}_j^2 \max_{0\le l<i}\lambda_{t_l+5} + 32\tau\bar{\psi}_j(t_i^4 + 3t_i^3)\mu}{|u'^{(i)}(z_j)| + |\theta^{(i)}(z_j)v'^{(i)}(z_j)|} + O(\mu^2), \tag{8.58}$$

if $z_j \not\in C^{(i)}$, and

$$E(f_j, g_j, U^{(k+1)}(z_j), V^{(k+1)}(z_j)) \le \frac{32i\tau^2\bar{\psi}_j^2 \max_{0\le l<i}\lambda_{t_l+5} + 16\tau^2\bar{\psi}_j\mu}{|u'^{(i-1)}(z_j)| + |\theta^{(i-1)}(z_j)v'^{(i-1)}(z_j)|} + O(\mu^2), \tag{8.59}$$

if $z_j \in C^{(i)}$.

Proof: With (8.13) and (8.18), the result follows directly from Lemmas 8.1, 8.2, 8.3 and Corollary 8.1. □

Note that in Theorem 8.1, the pseudo-error $E(f_j, g_j, U^{(k+1)}(z_j), V^{(k+1)}(z_j))$ can be large if ψ_j is large. However, in these situation, the point z_j is either a near-duplicate point or $\kappa^{(i)}$ is large (i.e., the problem is ill-conditioned). Numerical experiments that illustrate these situations are given in Chapter 9.

Theorem 8.2 *If the conditions of Lemma 8.2 and Corollary 8.1 are satisfied, then Algorithm 4.2 is weakly stable for Problem 1.1.*

Proof:

From (8.9) and (8.58), it follows that there exists δ_j such that

$$\frac{g_j U^{(k+1)}(z_j) + f_j V^{(k+1)}(z_j)}{|U^{(k+1)}(z_j)| + |V^{(k+1)}(z_j)|} = \delta_j, \quad j = 0, \ldots, N, \tag{8.60}$$

where

$$|\delta_j| \le \begin{cases} \dfrac{32 i \tau^2 \bar{\psi}_j^2 \max\limits_{0 \le l < i} \lambda_{t_l+5} + 32\tau\bar{\psi}_j (t_i^4 + 3t_i^3)\mu}{|u'^{(i)}(z_j)| + |\theta^{(i)}(z_j) v'^{(i)}(z_j)|} + O(\mu^2) & \text{if } \ z_j \notin \mathcal{C}^{(i)}, \\ \dfrac{32 i \tau^2 \bar{\psi}_j^2 \max\limits_{0 \le l < i} \lambda_{t_l+5} + 16\tau^2\bar{\psi}_j \mu}{|u'^{(i-1)}(z_j)| + |\theta^{(i-1)}(z_j) v'^{(i-1)}(z_j)|} + O(\mu^2) & \text{if } \ z_j \in \mathcal{C}^{(i)}. \end{cases} \tag{8.61}$$

Thus,

$$g_j U^{(k+1)}(z_j) + f_j V^{(k+1)}(z_j) = \delta_j (|U^{(k+1)}(z_j)| + |V^{(k+1)}(z_j)|), \quad j = 0, \ldots, N. \tag{8.62}$$

Let

$$\bar{f}_j = f_j - \delta_j \operatorname{sign}(V^{(k+1)}(z_j)) \tag{8.63}$$

and

$$\bar{g}_j = g_j - \delta_j \operatorname{sign}(U^{(k+1)}(z_j)). \tag{8.64}$$

Then

$$\bar{g}_j U^{(k+1)}(z_j) + \bar{f}_j V^{(k+1)}(z_j) = 0, \quad j = 0, \ldots, N, \tag{8.65}$$

and

$$\begin{aligned} \frac{\|\,(\, g_j - \bar{g}_j \quad f_j - \bar{f}_j \,)\,\|}{\|\,(\, g_j \quad f_j \,)\,\|} &= \frac{\|\,(\, \delta_j \operatorname{sign}(U^{(k+1)}(z_j)) \quad \delta_j \operatorname{sign}(V^{(k+1)}(z_j)) \,)\,\|}{\|\,(\, g_j \quad f_j \,)\,\|} \\ &= |\delta_j|, \end{aligned} \tag{8.66}$$

where $|\delta_j|$ is bound according to (8.61). That is, $(U^{(k+1)}(z), V^{(k+1)}(z))$ is the exact solution of the problem (8.65) with perturbed input $\{(\bar{g}_j, \bar{f}_j)\}_{j=0,\ldots,N}$ where the perturbation is by $|\delta_j|$ in (8.66) and is bounded by (8.61). Similarly, from Theorem 6.2, we know that, for $i = 0, \ldots, k$, $j = n_i + 1, \ldots, n_i + t_i$,

$$\frac{\|S^{(i)}(z_j) - \bar{S}^{(i)}(z_j)\|}{\|\bar{S}^{(i)}(z_j)\|} \le i \cdot \tau \cdot \bar{\psi}_j \cdot \max_{0 \le l \le i-1} \lambda_{t_l+3} + O(\mu^2). \tag{8.67}$$

From, (5.64), it follows that Algorithm 4.2 is weakly stable. □

In the proof of Thm 8.2, we need to show that both $\|(\,g_j - \bar{g}_j \quad f_j - \bar{f}_j\,)\|/\|(\,g_j \quad f_j\,)\|$ and $\|S^{(i)}(z_j) - \bar{S}(i)(z_j)\|/\|\bar{S}^{(i)}(z_j)\|$ are small for all well-conditioned problems. Since the first term $\|(\,g_j - \bar{g}_j \quad f_j - \bar{f}_j\,)\|/\|(\,g_j \quad f_j\,)\|$ is bounded using the result of the bound for the pseudo-error, which in turn requires that the bound of $\|S^{(i)}(z_j) - \bar{S}^{(i)}(z_j)\|/\|\bar{S}^{(i)}(z_j)\|$ be small in proving it, it follows then that if the pseudo-error is small, Algorithm 4.2 is weakly stable. Furthermore, since a small bound for $\|(\,g_j - \bar{g}_j \quad f_j - \bar{f}_j\,)\|/\|(\,g_j \quad f_j\,)\|$ implies the computed solution is the exact solution of a nearby problem, another way of stating our result is that Algorithm 4.2 computes a solution which is the exact solution of a nearby problem whenever the problem is well-conditioned.

We conclude this chapter by addressing the conditions of Lemma 8.2 and Corollary 8.1 which are required for Theorem 8.1 and Theorem 8.2 to be valid.

Remark 8.2 *Note that the conditions of Lemma 8.2 (similarly for the conditions of Corollary 8.1), are always satisfied in that $\epsilon_j^{(i)}$ is always small since the local interpolation is done by Gaussian elimination and thus the assumption*

$$\frac{1}{2}|\det(S^{(i)}(z_j))|\,(|f_j| + |g_j|) \ge |\epsilon_j^{(i)} w_j^{(i)}|\,(|U^{(i)}(z_j)| + |V^{(i)}(z_j)|)$$

is reasonable. Furthermore, if $|\epsilon_j^{(i)}| \le 1$, it follows that

$$\begin{aligned} |\epsilon_j^{(i)}|\,|w_j^{(i)}| &\le |w_j^{(i)}| \\ &\le |r_j^{(i)}| + |w_j^{(i)}| \\ \left|\frac{u'^{(i)}(z_j)}{|\theta^{(i)}(z_j)v'^{(i)}(z_j)} + \frac{r_j^{(i)}}{w_j^{(i)}}\right| &\le \frac{|r_j^{(i)}| + |w_j^{(i)}|}{|w_j^{(i)}|} \\ \left|\frac{u'^{(i)}(z_j)}{|\theta^{(i)}(z_j)v'^{(i)}(z_j)} + 1 - 1 + \frac{r_j^{(i)}}{w_j^{(i)}}\right| &\le \frac{|r_j^{(i)}| + |w_j^{(i)}|}{|w_j^{(i)}|} \end{aligned} \tag{8.68}$$

From (8.68), we get

$$\left|\frac{u'^{(i)}(z_j)}{|\theta^{(i)}(z_j)v'^{(i)}(z_j)} + 1\right| - \left|-1 + \frac{r_j^{(i)}}{w_j^{(i)}}\right| \le \frac{|r_j^{(i)}| + |w_j^{(i)}|}{|w_j^{(i)}|}$$

$$\frac{|u'^{(i)}(z_j) + \theta^{(i)}(z_j)v'^{(i)}(z_j)|}{|\theta^{(i)}(z_j)v'^{(i)}(z_j)} \leq 2\frac{|r_j^{(i)}| + |w_j^{(i)}|}{|w_j^{(i)}|}, \tag{8.69}$$

and

$$\begin{aligned} \left|\frac{u'^{(i)}(z_j)}{|\theta^{(i)}(z_j)v'^{(i)}(z_j)} - 1\right| - \left|1 + \frac{r_j^{(i)}}{w_j^{(i)}}\right| &\leq \frac{|r_j^{(i)}| + |w_j^{(i)}|}{|w_j^{(i)}|} \\ \frac{|u'^{(i)}(z_j) - \theta^{(i)}(z_j)v'^{(i)}(z_j)|}{|\theta^{(i)}(z_j)v'^{(i)}(z_j)|} &\leq 2\frac{|r_j^{(i)}| + |w_j^{(i)}|}{|w_j^{(i)}|}. \end{aligned} \tag{8.70}$$

It follows from (8.69) and (8.70) that

$$\frac{|u'^{(i)}(z_j)| + |\theta^{(i)}(z_j)v'^{(i)}(z_j)|}{|\theta^{(i)}(z_j)v'^{(i)}(z_j)|} \leq 2\frac{|r_j^{(i)}| + |w_j^{(i)}|}{|w_j^{(i)}|}. \tag{8.71}$$

Chapter 9

Numerical Results

In this chapter, we present and discuss experimental results to augment the pseudo-error analysis of Algorithm 4.2 in Chapter 8. The reported experiments are all typical cases for the particular classes of experiments. Unless stated otherwise, all the rational interpolants reported in this chapter are of type $[\frac{N}{2}, \frac{N}{2} - 1]$, where N is an even number of interpolation points. In other words, all solution paths are the staircase paths on and immediately below the diagonal in the rational interpolant table.

Algorithm 4.2 was implemented in the MATLAB programming language with double precision arithmetic. MATLAB is an interactive, matrix-based system for scientific calculations; it is an outgrowth of the LINPACK and EISPACK projects. Due to the formulation of Algorithm 4.2, MATLAB, with its built-in functions of matrix computations, is particularly suitable for the implementation.

9.1 Scaling and Pseudo-Error

In this section, we discuss the scaling of the data and we illustrate the strength of the pseudo-error measure (8.4) introduced in Chapter 8.

Given the original data $\{(z'_j, f'_j, g'_j)\}_{j=0,\ldots,N}$ in some domain $[a, b]$ and range $[c, d]$, we first linearly map this data to the interpolation domain $[-1, 1]$. For the range, we linearly map the majority of interpolation values, which are in the range $[c', d']$, to the range $[-1, 1]$. In other words, we obtain the interpolation data $\{(z_j, f_j, g_j)\}_{j=0,\ldots,N}$ such that

$$z_j = -1 + \frac{2(z'_j - a)}{b - a}, \quad -1 \leq z_j \leq 1 \tag{9.1}$$

and

$$\frac{f_j}{g_j} = -1 + \frac{2(f'_j/g'_j - c')}{d' - c'}. \tag{9.2}$$

Since the range $[c', d']$ of the majority of the interpolation values is used, the range of f_j/g_j is yet to be determined. The meaning of majority is left to the user's discretion since this scaling procedure depends heavily on the data. A general guideline is that the majority of the data points is the subset of the data which requires a higher degree of accuracy compared to the rest of the data.

In the following, we present an interpolation problem of the function $TAN(z)$.

The table below depicts the differences between the relative error (8.3) (denoted as R.E.) and the pseudo-error (8.4) (denoted as P.E.) for the interpolation of the $TAN(z)$ function. The eight interpolation points z'_j generated were in the range $-\pi/2 \leq z'_j \leq \pi/2$. To illustrate the problem with the conventional relative error measure where the range of data is large, we have chosen the fifth point to be close to $\pi/2$ so that $TAN(z'_4)$ is relatively large compared to the other points in the set (see Table 9.1). (Note that the heading $U(z_j\pi/2)/V(z_j\pi/2)$ denotes the continued-fraction form; we use this heading for spacing reasons only). In this experiment, we need not scale the range because the magnitude of the majority of the function values is around one. It can be seen that the relative error at the fifth point is of size $O(10^{-3})$, which is significantly larger than the relative error at the other points. Furthermore, if the vector norm (8.1) were to be used to measure its error size, the fifth point would dominate and thus a relative error of size $O(10^{-3})$ would result. However, such an error measure, which only places importance on large interpolation values, completely overshadows how well the other points in the set have been interpolated. In contrast, the pseudo-error measure automatically adjusts the relative importance of the data according to its magnitude.

z_j	$TAN(z_j\pi/2)$	$U(z_j\pi/2)/V(z_j\pi/2)$	R.E.	P.E.
0.6772794093	1.800721810e+00	-1.800721810e+00	0.0e+00	0.0e+00
-0.0967719415	-1.531907447e-01	1.531907447e-01	3.1e-15	4.1e-16
0.9132027635	7.289059341e+00	-7.289059341e+00	3.0e-15	3.7e-16
-0.7056935271	-2.006778747e+00	2.006778747e+00	6.6e-16	2.2e-16
1.0000000000	4.491214388e+13	-4.529196873e+13	8.5e-03	1.9e-16
0.5388728003	1.130236830e+00	-1.130236830e+00	0.0e+00	0.0e+00
-0.1116767686	-1.772432828e-01	1.772432828e-01	3.6e-15	5.4e-16
0.2412402420	3.981833116e-01	-3.981833116e-01	2.8e-16	7.9e-17

Table 9.1: Experiment 9.1: Eight Points of $TAN(z)$

9.2 Ill-posed Points

Ill-posed points, as defined in this section, are the points associated with the singular blocks. Here we demonstrate the ability of Algorithm 4.2 to handle ill-posed points. We begin by describing a procedure for generating test data appropriate for this demonstration.

The theorem below describes the intrinsic relationship between unattainable points and singular blocks (hence singular LRIS's).

Theorem 9.1 *Let $r(z) \in \mathcal{R}(L,M)$ be the rational interpolant obtained from its linear rational interpolant $(U(z),V(z))$ of type $[L,M]$ interpolating the points $\{(z_j,f_j,g_j)\}_{j=0,N}$, where $N = L+M$. Suppose that $(U(z),V(z))$ accidentally interpolates the next k points $\{(z_j,f_j,g_j)\}_{j=N+1,\ldots,N+k}$, but not the subsequent k^* points $\{(z_j,f_j,g_j)\}_{j=N+k+1,\ldots,N+k+k^*}$, where $1 \le k^* \le k$. Then these k^*, not interpolated by $(U(z),V(z))$, are unattainable in the set $\{(z_i,f_j,g_j)\}_{i=0,\ldots,N+k+k^*}$ for all rational interpolants $r^* \in \mathcal{R}(L+l,M+m)$, where $k+k^* = l+m$.*

Proof: Suppose there is an $r^*(z) \in \mathcal{R}(L+l,M+m)$, obtained from its linear interpolant $(U^*(z),V^*(z))$ interpolating the points $\{(z_j,f_j,g_j)\}_{j=0,\ldots,N+k+k^*}$. Then $U^*(z)V(z)-V^*(z)U(z)$ has $N+k+1$ zeros. This follows because both $(U(z),V(z))$ and $(U^*(z),V^*(z))$ interpolate the first $N+k+1$ points. But $U^*(z)V(z)-V^*(z)U(z) \in \mathcal{P}_{N+k}$ because $deg(U^*(z)V(z)-V^*(z)U(z)) \le \max\{(L+l)+M,(M+m)+L\} \le L+M+k^* \le N+k$. Hence, $U^*(z)V(z) = V^*(z)U(z)$ which implies $r^*(z) = r(z)$. But $r(z)$ does not interpolate $\{(z_j,f_j,g_j)\}_{j=N+k+1,\ldots,N+k+k^*}$ as given. □

Theorem 9.1 describes the entries containing unattainable points in the lower triangular region B of a square singular block in Fig. 9.1. In this region, an entry contains at least one unattainable point with respect to the $[L,M]$ entry, and up to k unattainable points in the entry $[L+k,M+k]$.

The entries $[L+l,M+m]$ where $0 < l+m \le k$, are inscribed in the upper triangular region A. Note that Region A contains no unattainable points with respect to the entry $[L,M]$ since it is constructed by the extra k points that is accidentally interpolated by the entry $[L,M]$. We name these k points the *singular points*; they are so called because these points create the singular block. Numerically, we refer to them as the *ill-posed points*. Thus, by using Theorem 9.1, we can generate data sets with singular blocks and unattainable points. By perturbing the data, ill-posed points are generated.

Note that, as pointed out in Remark 2.2 of Chapter 2, unattainable points need not

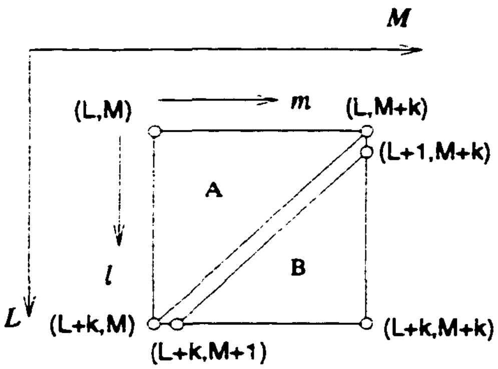

Figure 9.1: A square singular block.

be associated with singular blocks. Thus, the entry described in Theorem 9.1 may have unattainable points z_σ, $0 \leq \sigma \leq N$, of its own.

In the following, we present three experiments to illustrate how Algorithm 4.2 handles ill-posed points on a solution path. The ill-posed points are located at z_3 and z_8 on the solution path in all three experiments. The three experiments consist of 16 points with two numerically simulated singular blocks generated by applying Theorem 9.1 of 2×2 located at $(1,1)$ and $(4,3)$ in the rational interpolation table along the staircase path.

The data sets are generated as follows. We first obtain a $[1,1]$ type interpolant, call it $(u_1(z), v_1(z))$, from the randomly generated points, z_0, z_1, z_2. Then, the next point is generated as $f_3/g_3 = u_1(z_3)/v_1(z_3) + \zeta_1$, where $\zeta_.$ is a small random perturbation. The next 4 points, z_4, z_5, z_6, z_7, are randomly generated. With these eight points, a rational interpolant of type $[4,3]$, call it $(u_2(z), v_2(z))$, is obtained. Lastly, for the second ill-posed point, we use $(u_2(z), v_2(z))$ to generate the function value of the next point, z_8, i.e., $f_8/g_8 = u_2(z_8)/v_2(z_8) + \zeta_2$, and the remaining seven points are generated randomly. With these 16 points, the task is to construct a rational interpolation of type $[8,7]$.

In Experiment 9.2, $O(\zeta_1) = 10^{-9}$ and $O(\zeta_2) = 10^{-9}$, this is to examine the effect of the same magnitude ill-posed points. The results are tabulated in Table 9.2. The pseudo-errors P.E. are calculated using the final solution type $[8,7]$.

Notice in Table 9.2 that without skipping ($\tau = \infty$) over ill-posed points, after encountering and accepting the first ill-posed point z_3, the residual error increased dramatically and remained large for the remaining points. The stability parameter $\tau^{(i)}(z_j)$ behaved similarly; this is so because accepting an ill-posed point causes the resulting solution $S^{(i)}(z_j)$ to be ill-conditioned at the points that follow. The second ill-posed point z_8 makes little or

			$\tau = 10^5$			$\tau = \infty$		
j	z_j	f_j/g_j	i	$\tau^{(i)}(z_{j+1})$	P.E.	i	$\tau^{(i)}(z_{j+1})$	P.E.
0	0.90025857	2.58554998	0	7.0e+00	0.0e+00	0	7.0e+00	0.0e+00
1	-0.53772297	0.41353851	1	5.2e+01	2.7e-16	1	5.2e+01	2.7e-16
2	0.21368517	1.36179193	2	2.5e+02	1.3e-16	2	2.5e+02	1.3e-16
3	-0.02803506	1.02153372	-	4.7e+11	5.6e-17	3	4.7e+11	5.6e-17
4	0.78259793	-0.29426374	-	5.9e+11	6.7e-14	4	1.3e+12	1.3e-05
5	0.52419367	0.62633299	3	5.9e+03	1.6e-15	5	4.1e+12	4.3e-07
6	-0.08706467	-0.98027740	4	8.6e+01	1.1e-14	6	1.8e+19	3.5e-05
7	-0.96299271	-0.72221824	5	2.5e+04	9.0e-16	7	4.0e+18	9.7e-08
8	0.64281433	-0.06224507	-	6.6e+10	3.4e-15	8	8.4e+21	1.6e-06
9	-0.11059327	-2.90269834	-	4.1e+12	2.1e-14	9	1.7e+21	2.6e-05
10	0.23086470	0.75638402	6	3.1e+04	9.4e-15	10	3.1e+18	4.4e-06
11	0.58387407	-1.96415805	7	7.8e+04	1.8e-14	11	1.0e+19	1.7e-06
12	0.84362594	-0.12714617	8	2.3e+04	4.4e-14	12	5.4e+18	6.5e-05
13	0.47641449	0.41801396	9	5.1e+03	1.2e-15	13	1.7e+20	7.5e-07
14	-0.64746771	13.76489290	10	4.6e+03	1.7e-14	14	1.2e+21	1.2e-05
15	-0.18858757	-1.72741382	11	—	7.5e-15	15	—	3.0e-06

Table 9.2: Experiment 9.2: Two Ill-posed Points: $O(\zeta_1) = 10^{-9}$ and $O(\zeta_2) = 10^{-9}$.

no difference to the points that follow.

On the other hand, by setting the stability tolerance to $\tau = 10^5$, the ill-posed points were skipped over. Notice that for a "singular" block of 2 × 2, a step size $t_i = 3$ is needed to skip over it on a staircase path so as to avoid accepting an ill-conditioned LRIS. In Table 9.2, we can see that the two ill-posed points are skipped over by a step size of three.

In Experiment 9.3, $O(\zeta_1) = 10^{-3}$ and $O(\zeta_2) = 10^{-8}$, we examine the effect of a mild ill-posed point following a more ill-posed point. In Table 9.3, the first ill-posed point at z_3 is a mild one compared to the severity of instability at the second ill-posed point at z_8. We see that there is a distinct drop in accuracy corresponding to the intensity of the instability at these points.

When the position of the two ill-posed points is switched, in Experiment 9.4 we see that the loss of accuracy caused by the first ill-posed point remains for the rest of the points. The now second milder ill-posed point has no effect on the accuracy at all. Thus, it appears that the effect of ill-posed points on residual errors are additive (not compounded) if there should be more than one ill-posed point along the solution path.

From these experiments, it is seen that once an ill-posed point is accepted, its detrimental effect lingers for the remaining points in the set. On the other hand, when the stability parameter tolerance τ is set appropriately, 10^5 in these experiments, the two ill-posed points are skipped over. Since no ill-posed point is accepted, no detrimental effect is present.

			$\tau = 10^5$			$\tau = \infty$		
j	z_j	f_j/g_j	i	$\tau^{(i)}(z_{j+1})$	P.E.	i	$\tau^{(i)}(z_{j+1})$	P.E.
0	-0.81560759	-3.88530037	0	6.9e+01	0.0e+00	0	6.9e+01	0.0e+00
1	-0.70569353	-5.76386112	1	2.4e+03	2.4e-17	1	2.4e+03	2.4e-17
2	-0.66769184	-7.54717730	2	6.5e+01	4.9e-17	2	6.5e+01	4.9e-17
3	-0.24884668	-0.32684179	-	1.3e+06	5.4e-16	3	1.3e+06	5.4e-16
4	-0.15022172	-0.01871128	-	1.7e+06	6.1e-16	4	2.6e+06	3.2e-13
5	-0.11167677	-0.18145134	3	6.2e+04	7.5e-16	5	1.7e+06	2.0e-13
6	-0.09677194	-0.07294883	4	7.9e+02	9.4e-16	6	2.2e+06	7.4e-14
7	0.22022715	0.22188710	5	7.3e+02	4.1e-16	7	2.2e+06	1.4e-12
8	0.24124024	0.24033784	-	2.3e+09	2.1e-15	8	4.3e+11	1.6e-12
9	0.40298520	-1.71353984	-	3.1e+09	2.8e-16	9	4.6e+11	9.8e-09
10	0.53887280	-2.68517483	6	3.7e+02	2.0e-16	10	4.8e+11	3.0e-09
11	0.62423564	-2.90802720	7	1.4e+03	8.3e-17	11	2.4e+16	3.2e-09
12	0.66630292	-0.17187021	8	4.7e+04	6.6e-15	12	2.1e+16	9.7e-09
13	0.67727941	0.09363880	9	2.6e+03	2.1e-14	13	1.3e+16	5.9e-09
14	0.73986586	1.61148582	10	2.5e+02	2.1e-16	14	8.8e+15	2.3e-09
15	0.91320276	2.17468763	11	—	1.2e-15	15	—	3.1e-10

Table 9.3: Experiment 9.3: Two Ill-posed Points: $O(\zeta_1) = 10^{-3}$ and $O(\zeta_2) = 10^{-8}$.

			$\tau = 10^5$			$\tau = \infty$		
j	z_j	f_j/g_j	i	$\tau^{(i)}(z_{j+1})$	P.E.	i	$\tau^{(i)}(z_{j+1})$	P.E.
0	-0.96848036	0.45627641	0	1.7e+03	0.0e+00	0	1.7e+03	0.0e+00
1	-0.96729013	0.45956152	1	5.6e+02	3.8e-17	1	5.6e+02	3.8e-17
2	-0.88483782	0.69269728	2	6.1e+03	6.6e-17	2	6.1e+03	6.6e-17
3	-0.83184188	0.84856901	-	2.4e+09	6.0e-17	3	2.4e+09	6.0e-17
4	-0.69278727	-0.09849212	-	1.6e+09	7.3e-15	4	1.9e+10	3.3e-08
5	-0.61985082	0.43176590	3	1.5e+03	4.7e-15	5	4.1e+08	8.0e-09
6	-0.29349909	0.78568322	4	1.4e+03	3.7e-16	6	2.5e+14	1.6e-09
7	-0.26486392	-0.45379506	5	5.0e+02	1.7e-15	7	4.8e+13	2.2e-09
8	-0.11634341	3.14075781	-	2.6e+06	2.5e-16	8	4.3e+13	1.9e-11
9	-0.09128970	-1.36597790	-	4.7e+05	8.3e-16	9	2.3e+13	1.4e-09
10	0.17383694	0.74650729	6	1.5e+03	1.7e-15	10	2.2e+13	1.0e-09
11	0.21708072	1.02845648	7	6.4e+02	1.7e-15	11	2.0e+13	2.0e-10
12	0.26290233	1.06790443	8	5.4e+02	2.1e-15	12	1.8e+13	1.3e-10
13	0.35128930	-0.45554608	9	5.4e+02	1.1e-15	13	1.8e+13	4.4e-09
14	0.38533879	-1.05449175	10	2.3e+02	1.0e-15	14	1.7e+13	1.4e-09
15	0.43526884	-0.66601293	11	—	1.2e-15	15	—	6.7e-10

Table 9.4: Experiment 9.4: Two Ill-posed Points: $O(\zeta_1) = 10^{-8}$ and $O(\zeta_2) = 10^{-3}$.

j	z_j	f_j/g_j	$U(z_j)/V(z_j)$	P.E.	Ω_j
0	-3	-3	3	0.0e+00	0.0e+00
2	-2	-2	NaN	NaN	∞
1	-1	-3	3	0.0e+00	1

Table 9.5: Experiment 9.5: Example 2.1 of Chapter 2.

j	z_j	f_j/g_j	$U(z_j)/V(z_j)$	P.E.	Ω_j
0	-3	-3	3	0.0e+00	1.3e+00
2	-2	-2	2	2.4e-17	1.5e-16
1	-1	-3	3	0.0e+00	3.1e+15
1	0	0	0	0.0e+00	1.0e+00

Table 9.6: Experiment 9.6: Example 2.2 of Chapter 2.

9.3 Unattainable Points

In this section, we examine the relationship between unattainable points and Ω_j introduced in (8.47) and (8.57). We begin by using the first two examples in Chapter 2. Note that the interpolation domain of these examples is outside $[-1,1]$. For ease of illustration, we present these examples in their original domain.

As discussed in detail, the unattainable points in Examples 2.1 and 2.2 of Chapter 2 are $z_1 = -2$ and $z_2 = -1$, respectively. The results of these two examples are tabulated in Tables 9.5 and 9.6. Notice that in Table 9.5, $\Omega_1 = \infty$; this indicates that z_1 is an unattainable point. Indeed, since Example 2.1 is a small problem (only three points), the numerical solution happens to be the exact solution; thus, $\Omega_1 = \infty$. Because the factor $(z - z_1)$ is present in both the numerator and denominator of the solution, $U(z_1)/V(z_1) = NaN$ (Not a Number). Typically, however, the numerical solution is not the exact solution. So, at an unattainable point, such as z_2 in Example 2.2, Ω_j takes on some large value (see Table 9.6).

An algebraic unattainable point is clearly defined in Definition 2.1. While we can use this definition numerically if the numerical solution happens to be the exact solution (e.g. Example 2.1 in Table 9.5), this is not true in general. The reason being, any small perturbations introduced in the original system make the algebraic unattainable points attainable in the algebraic sense (e.g. Example 2.2 in Table 9.6). Upon examining the *P.E.* column in Table 9.6 alone, it can be concluded that all points in this example are attainable to within machine epsilon. Thus, the algebraic notion of unattainable points does not carry over to the numerical setting.

On the other hand, the original data set could display no unattainable points with respect to Definition 2.1. But with perturbations introduced by rounding errors, the numerical solution may now display unattainable points with respect to Definition 2.1. Thus, to deem such points as numerical unattainable is meaningful only if these points reveal something important about the rational interpolant, exact and numerical, at or near these points. We study this aspect using Experiment 9.6 (Example 2.2).

To show an unattainable point pictorially, we plotted the rational interpolant of Experiment 9.6 in Figure 9.2. To plot the rational interpolant, we sampled 100 evenly spaced points in $[-3, 0]$. Notice that the point $(-1, -3)$ is seemingly not interpolated by the rational interpolant. This would indeed be the case for the exact solution, but according to Table 9.6, we know the numerical solution does interpolate at $z_2 = -1$. It is only when we plotted the 100 near points (50 before z_2 and 50 after), we see the rational interpolant interpolates at z_2 (see Figure 9.3). These 100 near points are $\{z_2 \pm i \cdot 10^{-10},\ i = 1, \ldots, 50\}$. The implication of Figure 9.3 is that there is a zero and a pole very close to z_2, but not at z_2.

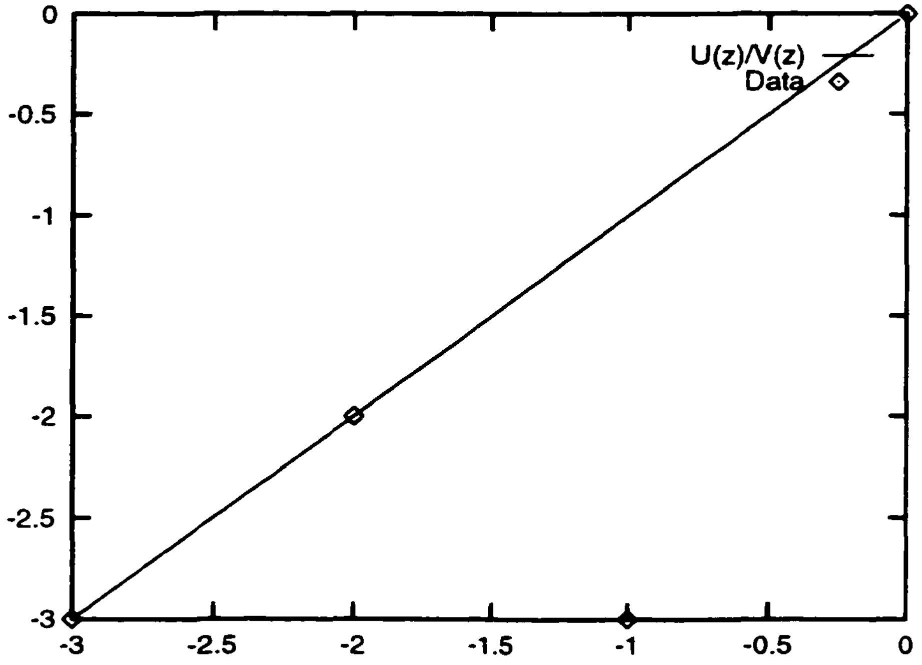

Figure 9.2: 100 points of U(z)/V(z)

Conjecture 9.1 *The magnitude of Ω_j indicates the closeness of a zero and a pole together at z_j. The larger the magnitude of Ω_j, the closer they are, with the two collapsing into the same point at z_j when $\Omega_j = \infty$.*

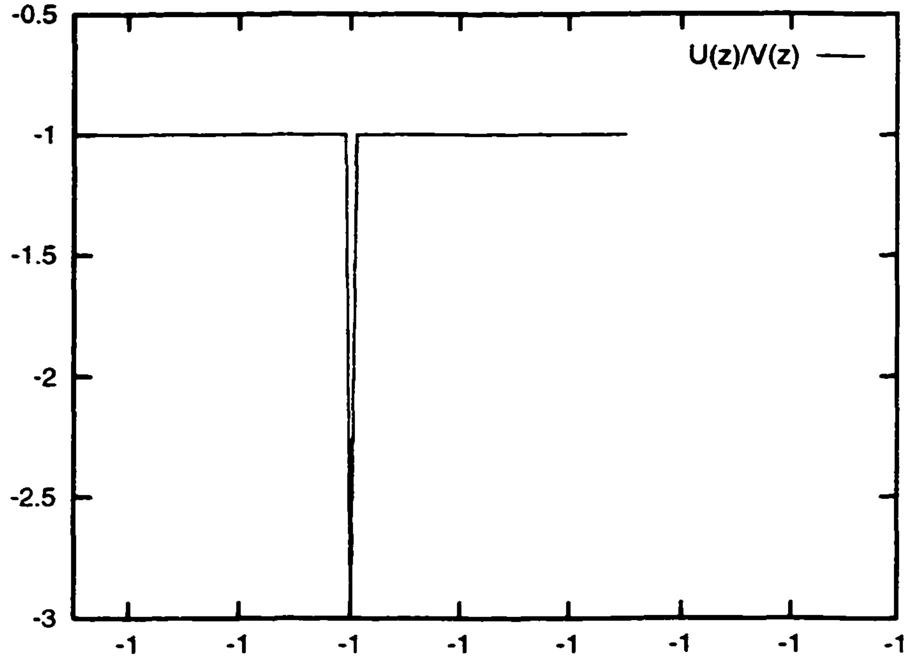

Figure 9.3: 100 points near -1.

With this idea, if the original data set contains a zero and a pole near an interpolation z_σ, then we report z_σ as a numerical unattainable point.

We now show an example that has a smaller Ω_j magnitude than that in Table 9.6. Once again, we used Theorem 9.1 to create unattainable points. Table 9.7 shows the results of this experiment. From Table 9.7 and Figure 9.4, we see that Ω_j of points z_8 and z_9 are at least $O(10^7)$ larger than the other points. However, $\Omega_8 = 5.1 \times 10^7$ and $\Omega_9 = 1.7 \times 10^7$ are not as large as the previous example. Now, plotting the near points of $z_9 = -0.2$ as done in the last example, shows a relative gradual change as opposed to that of Figure 9.3 (See

j	z_j	f_j/g_j	$U(z_j)/V(z_j)$	P.E.	Ω_j
0	-0.901906344	-2.470780208e-01	2.470780208e-01	0.0e+00	1.9e+00
1	-0.655600609	4.743535599e-02	-4.743535599e-02	1.3e-17	6.0e-01
2	-0.543922194	2.234554592e-01	-2.234554592e-01	4.5e-17	2.3e-01
3	-0.288567791	7.619392475e-01	-7.619392475e-01	1.3e-16	6.8e-01
4	-0.100715791	1.274809722e+00	-1.274809722e+00	0.0e+00	6.4e-02
5	0.044121831	1.661143578e+00	-1.661143578e+00	6.9e-17	2.1e-07
6	0.426708887	1.589973018e+00	-1.589973018e+00	0.0e+00	1.4e+01
7	0.510677144	1.323842058e+00	-1.323842058e+00	3.8e-16	2.4e+01
8	0.600000000	2.092716407e-01	-2.092716420e-01	1.1e-09	5.1e+07
9	-0.200000000	4.550916908e-01	-4.550918273e-01	9.4e-08	1.7e+07

Table 9.7: Experiment 9.7: Two numerical unattainable points

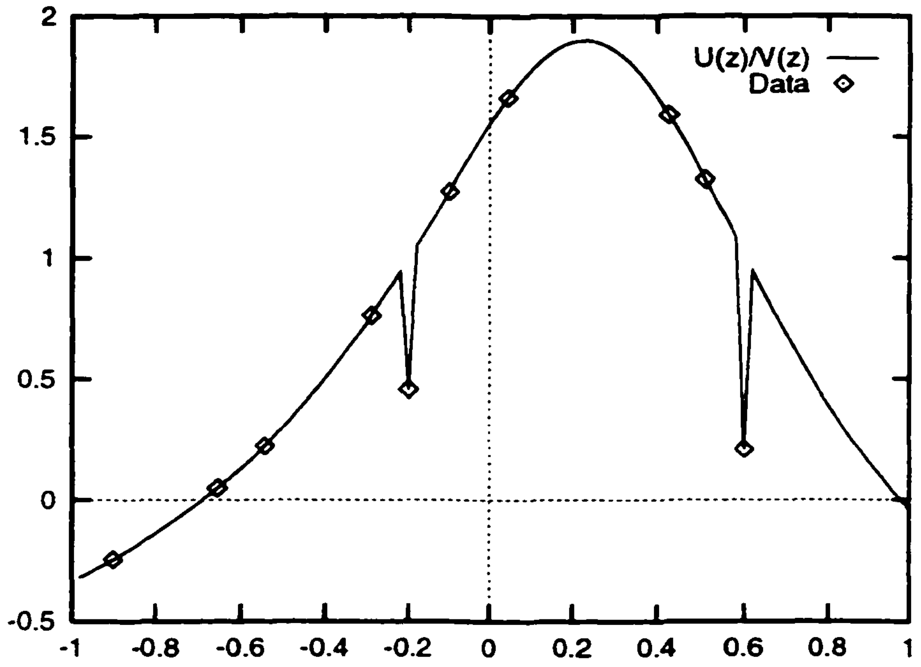

Figure 9.4: $U(z)/V(z)$ of Experiment 9.7

Figure 9.5). Indeed, we can also see that on the left of z_9, the interpolant curves upward to infinity, indicating a pole. Note that the 100 near points are of the same closeness as that of Figure 9.3 (i.e., we use the points $\{z_9 \pm i \cdot 10^{-10},\ i = 1, \ldots, 50\}$).

9.4 Close Points

In this section, we investigate the aspect of rational interpolation where interpolation points are close together. We begin by discussing the concepts of close points analytically with two examples to illustrate the relationship between the condition number $\kappa^{(i)}$ of a submatrix and the parameters ψ_j and Ω_j. Numerical examples are then presented to augment our discussion.

For simplicity, in the following discussion, we use only two points z_α and z_β for the discussion. We demonstrate an intrinsic relationship between close interpolation points and unattainable points.

Algebraically, a point z_α is distinct from z_β if $z_\beta - z_\alpha \neq 0$. Numerically, however, the distinctiveness between two points is not as sharp and clear. Two floating point numbers z_α and z_β are distinct if for $z_\alpha \neq 0$ [37]

$$|z_\alpha|\frac{\mu}{2} \leq |z_\beta - z_\alpha|. \tag{9.3}$$

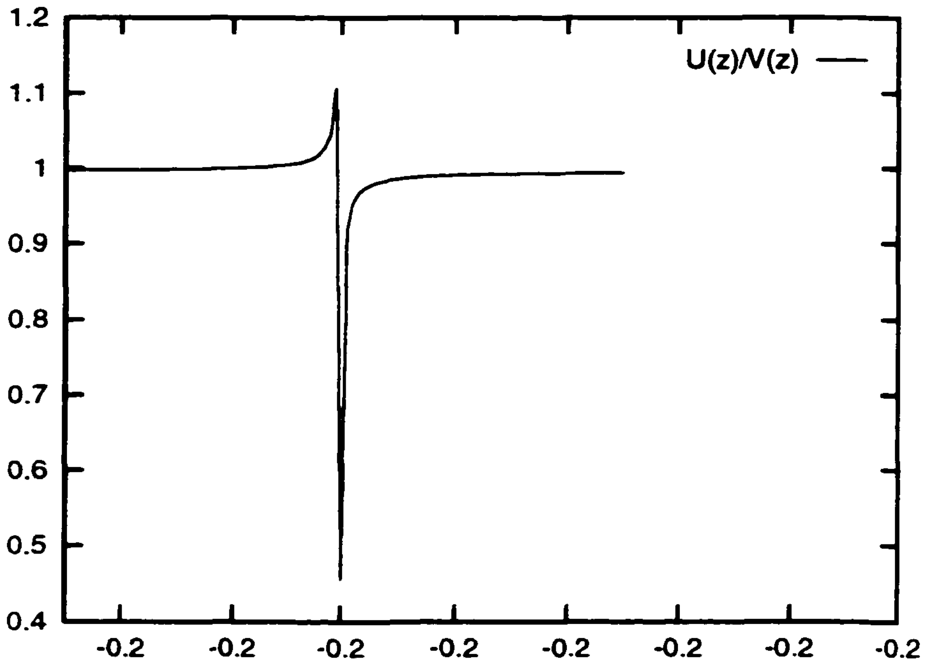

Figure 9.5: 100 points near −0.2.

Should the size of $|z_\beta - z_\alpha|$ be smaller than $|z_\alpha|\frac{\mu}{2}$, then z_β is stored as z_α.

Since we need to deal with close interpolation points, hence identical points, it is best to first examine what this means algebraically. The following theorem describes the effect on the linear rational interpolation when two points ϵ_z between z_α and z_β are separated by an arbitrarily small distance ϵ_z (i.e., $|z_\beta - z_\alpha| = \epsilon_z$).

Theorem 9.2 *Let $\mathcal{F} = \{(z_j, f_j, g_j)\}_{j=0,N}$, $N > 1$ and let $(U(z), V(z))$ be the linear rational interpolant, interpolating $\mathcal{F}$ be of type $[L, M]$ where $L = M$ if N is even, $L = M + 1$ if N is odd. Assume that $M_{L,M}$ is nonsingular. If at two points z_α and $z_\beta = z_\alpha + \epsilon_z$ in $\mathcal{F}$, $f_\alpha g_\beta - f_\beta g_\alpha = \epsilon_f$, then the linear solution $(U(z), V(z))$ satisfies*

$$|U(z_\alpha)| + |V(z_\alpha)| \le 4\frac{|\epsilon_z|}{|\epsilon_f|}\max\{L, M\}. \tag{9.4}$$

Proof: With $N > 1$ and the degree type specified, we have $\deg(U(z)) > 0$ and $\deg(V(z)) > 0$. With this condition, we eliminate the case where the solution is a polynomial or an inverse of a polynomial. Now, $(U(z), V(z))$ must satisfy the conditions:

$$\begin{aligned} f_\alpha V(z_\alpha) + g_\alpha U(z_\alpha) &= 0, && (9.5)\\ f_\beta V(z_\alpha + \epsilon_z) + g_\beta U(z_\alpha + \epsilon_z) &= 0. && (9.6) \end{aligned}$$

Using Taylor's expansion on (9.6), it becomes

$$f_\beta(V(z_\alpha) + \epsilon_z \frac{d}{dz} V(z_v^*)) + g_\beta(U(z_\alpha) + \epsilon_z \frac{d}{dz} U(z_u^*)) = 0, \tag{9.7}$$

where $z_\alpha \leq z_v^*, z_u^* \leq z_\beta$. From (9.5) and (9.7), we get

$$(f_\alpha g_\beta - f_\beta g_\alpha) U(z_\alpha) + f_\alpha \epsilon_z (f_\beta \frac{d}{dz} V(z_v^*) + g_\beta \frac{d}{dz} U(z_u^*)) = 0. \tag{9.8}$$

So

$$\begin{aligned} |\epsilon_f|\,|U(z_\alpha)| &= |(f_\alpha g_\beta - f_\beta g_\alpha) U(z_\alpha)| \\ &= |f_\alpha \epsilon_z (f_\beta \frac{d}{dz} V(z_v^*) + g_\beta \frac{d}{dz} U(z_u^*))|, \\ &\leq |f_\alpha \epsilon_z|\,(|f_\beta| + |g_\beta|) \max\{L, M\}. \end{aligned} \tag{9.9}$$

Similarly,

$$(f_\beta g_\alpha - f_\alpha g_\beta) V(z_\alpha) + g_\alpha \epsilon_z (f_\beta \frac{d}{dz} V(z_v^*) + g_\beta \frac{d}{dz} U(z_u^*)) = 0, \tag{9.10}$$

and

$$\begin{aligned} |\epsilon_f|\,|V(z_\alpha)| &= |(f_\beta g_\alpha - f_\alpha g_\beta) V(z_\alpha)| \\ &= |g_\alpha \epsilon_z (f_\beta \frac{d}{dz} V(z_v^*) + g_\beta \frac{d}{dz} U(z_u^*))|, \\ &\leq |g_\alpha \epsilon_z|\,(|f_\beta| + |g_\beta|) \max\{L, M\},. \end{aligned} \tag{9.11}$$

With $|f_\beta| + |g_\beta| \leq 2$, the addition of (9.9) and (9.11) yields (9.4). □

Remark 9.1 *Note that for $N = 1$ in Theorem 9.2, there are only two points in the data set. So the rational interpolant of type $[1, 0]$ that interpolates the two close points is*

$$(U(z), V(z)) = (z - \frac{f_\alpha g_\beta \epsilon_z}{\epsilon_f} - z_\alpha, \frac{g_\alpha g_\beta \epsilon_z}{\epsilon_f}). \tag{9.12}$$

Thus

$$\begin{aligned} |U(z_\alpha)| + |V(z_\alpha)| &= |\frac{f_\alpha g_\beta \epsilon_z}{\epsilon_f}| + |\frac{g_\alpha g_\beta \epsilon_z}{\epsilon_f}| \\ &= |\frac{\epsilon_z}{\epsilon_f} g_\beta| \cdot (|f_\alpha| + |g_\alpha|) \\ &\leq 2 |\frac{\epsilon_z}{\epsilon_f} g_\beta|. \end{aligned} \tag{9.13}$$

The two variables ϵ_z and ϵ_f in Theorem 9.2 can be of any value in the ranges of $|\epsilon_z| \leq 2$ and $|\epsilon_f| \leq \infty$. However, it is the small values near zero that we are interested in. For the case where $\epsilon_z = 0$ and $\epsilon_f \neq 0$, we have $|U(z_\alpha)| + |V(z_\alpha)| = 0$, making z_α an unattainable point (Theorem 2.3). In other words, the solution $(U(z), V(z))$ can be written in the form $((z - z_\alpha)U^*(z), (z - z_\alpha)V^*(z))$. Physically, when two interpolation points having different function values collapse into one single point, say at z_α, then the point z_α is unattainable. In other words, a rational interpolant cannot model a discontinuity.

Thus, Theorem 9.2 together with Theorem 2.3 indicate that discontinuity at a point implies unattainability at that point. We capture this phenomenon in the parameter Ω. Numerical examples will be given later in this section.

But what if $\epsilon_f = \epsilon_z = 0$? Then the point z_α is attainable; since in such a case, we simply have a duplicate data point in the system, not a discontinuity. A duplicate data point results in a duplicate equation in the system (1.5). A minimal solution from such a system can still be obtained simply by solving the system with the duplicate point removed.

However, numerically, unless the two points are indeed identical, we cannot remove any point since two points can be close together but not be exact duplicates. Thus, unless we assign a threshold to detect "numerically" equal points, we should treat all interpolation points, including identical points, the same way. Gaussian elimination with complete pivoting in the algorithm gives a solution $(U(z), V(z))$ which guarantees small residual errors. This solution, however, is undesirable because a small change in the function value at a nearly equal point will cause a large change in $U(z)/V(z)$. Thus, it is important and appropriate to warn the user that an ill-conditioned system is encountered. We do so by reporting the condition number $\kappa^{(i)}$ of each matrix of the subsystem for solving $(u^{(i)}(z), v^{(i)}(z))$.

We now illustrate the problem of close points by two examples below. These examples show an algebraic case of an interpolation problem with duplicate points. Although in practice we may not have exact duplicate points, these examples serve to illustrate the limiting case of an interpolation problem containing close points. We use these two examples to demonstrate how the algorithm handles close points and its consequences. Note that we use integers in these examples so that they can be followed easily.

Example 9.1 *Applying Algorithm 4.2, the first LRIS of type* $[1,0]$ *of the data in Table 9.8 on the staircase path is*

$$s^{(0)}(z) = \begin{pmatrix} z-2 & -(z-1) \\ 1 & (z-1) \end{pmatrix} \tag{9.14}$$

interpolating the first two points. This LRIS is acceptable since $\kappa(s^{(0)}(z_2)) = 4$.

Remark 9.2 *If we perturb* $f_1 = 1$ *to* $f_1 = 1 - \epsilon$, *where* ϵ *is* $O(\mu)$, *the rational interpolant* $(u^{(0)}(z), v^{(0)}(z))$ *of type* $[1,0]$ *from Algorithm 4.2 would be* $(z-1, 0)$. *Indeed,* $(z-1, 0)$ *is the only solution for any nonzero perturbation* ϵ *which makes the point* $z_0 = z_1 = 1$ *unattainable*

j	0	1	2	3	4
z_j	1	1	2	3	4
f_j	1	1	2	0	−1
g_j	1	1	1	1	1

Table 9.8: Data used to illustrate rational interpolation concepts.

(see Remark 9.1). Thus, a small change in the input data can cause a large change in the output. More importantly, a small change in the input data can cause an attainable point to become unattainable.

In fact, since the second point is a duplicate of the first, all possible solutions of the first LRIS of type [1, 0] in Example 9.1 are

$$s^{(0)}(z) = \begin{pmatrix} a(z-1)-1 & -(z-1) \\ 1 & (z-1) \end{pmatrix} \tag{9.15}$$

or

$$s^{(0)}(z) = \begin{pmatrix} b(z-1) & -(z-1) \\ 0 & (z-1) \end{pmatrix} \tag{9.16}$$

interpolating the first two points, where a and b are arbitrary constants. Because of Remark 9.2, it is important to warn the user if the data set contains duplicate (or close) points. We do so by providing the condition number $\kappa^{(i)}$ of the matrix of solving subsystem $(u^{(i)}(z), v^{(i)}(z))$. In the Example 9.1, $\kappa^{(0)} = \infty$, which indicates a duplicate point is encountered.

Note that a large $\kappa^{(i)}$ may result from accepting a solution that is inside a singular block. But in such an instance, $\tau^{(i)}$ would not be within τ, and Algorithm 4.2 would reject such a solution unless i is the last iteration. Thus, a large $\kappa^{(i)}$ (except for $i = k$) can only mean duplicate (or close) points in the system.

We now examine a situation where it might cause Algorithm 4.2 to be $O(N^4)$: If Algorithm 4.2 selects the solution where $a = 0$ in (9.15) then

$$s^{(0)}(z) = \begin{pmatrix} -1 & -(z-1) \\ 1 & (z-1) \end{pmatrix}. \tag{9.17}$$

Such a LRIS would not be accepted since $\det(s^{(0)}(z)) = 0$ and hence $\kappa(s^{(0)}(z_2)) = \infty$. So Algorithm 4.2 would increase the step size by one. Algebraically, the next possible LRIS is

$$s^{(0)}(z) = \begin{pmatrix} u^{(0)}(z) & (z-2)p^{(0)}(z) \\ v^{(0)}(z) & (z-2) \end{pmatrix} \tag{9.18}$$

interpolating the first three points, where $(u^{(0)}(z), v^{(0)}(z)) = (cz+2, z-c-3)$ or $(z+2d, dz-3d-1)$ and $p^{(0)}(z) = e(z-1)-1$. The variables c, d and e are arbitrary constants.

One can see that it is possible to construct an $s^{(0)}(z)$ such that $\det(s^{(0)}(z)) = 0$. In this case, choosing $(u^{(0)}(z), v^{(0)}(z)) = (z + 2d, dz - 3d - 1)$ and setting $d = 0$ and $e = -1$ in (9.18), we get

$$s^{(0)}(z) = \begin{pmatrix} z & -z(z-2) \\ -1 & (z-2) \end{pmatrix}, \tag{9.19}$$

and hence $\det(s^{(0)}(z)) = 0$. It is now easy to see that it is possible for our algorithm to be $O(N^4)$ if every new $s^{(0)}(z)$ is selected so that $\det(s^{(0)}(z)) = 0$. However, such a scenario is unlikely when we choose to set the arbitrary parameters in Gaussian elimination with complete pivoting to one to solve for $(u^{(0)}(z), v^{(0)}(z))$ and $(p^{(0)}(z), q^{(0)}(z))$.

When the close points (or duplicate points) appear together in a cluster, as in Example 9.1, the condition number of the matrix of the subsystem identifies the problem. However, if the close points do not appear together, then we need the parameter ψ to identify this problem. We now illustrate this in Example 9.2 below.

Example 9.2 *For the data in Table (9.9), the LRIS of type* $[0,0]$ *is*

j	0	1	2	3	4
z_j	1	2	1	3	4
f_j	1	2	1	0	−1
g_j	1	1	1	1	1

Table 9.9: Data used to illustrate rational interpolation concepts.

$$s^{(0)}(z) = \begin{pmatrix} -1 & (z-1) \\ 1 & 0 \end{pmatrix} \tag{9.20}$$

and the second LRIS of type $[1,1]$ *is*

$$s^{(1)}(z) = \begin{pmatrix} 5z-11 & (z-2)(z-3) \\ (z-1) & 0 \end{pmatrix}. \tag{9.21}$$

Note that $s^{(0)}(z)$ *is acceptable because* $\kappa(s^{(0)}(z_1)) = 2$, *and that* $s^{(1)}(z)$ *is acceptable because* $\kappa(s^{(0)}(z_4)s^{(1)}(z_4)) = 5.5$.

Example 9.2 contains the same data as Example 9.1 except for a change in position of the points z_1 and z_2. With this change, the solution no longer contains arbitrary constants. Furthermore, the condition numbers of the matrices for solving $(u^{(0)}(z), v^{(0)}(z))$ and $(u^{(1)}(z), v^{(1)}(z))$ are 1 and 3, respectively. These condition numbers do not indicate any problems with the data. However,

$$\psi_2 = \frac{\kappa(s^{(0)}(z_2))}{\kappa(s^{(0)}(z_1))} = \infty \tag{9.22}$$

results, which indicates that z_2 is one of the previous points interpolated. Thus, from Examples 9.1 and 9.2, we can see that the condition number of the matrix of the subsystem and the parameter ψ serve to identify close points in the data set.

We now present a numerical experiment with several variations to illustrate the concepts of close points as we had just described. This experiment is constructed from a known rational interpolant of type [5, 4]. We first generate 10 interpolation points with points z_3, z_4 only 10^{-9} apart from each other. Table 9.10 clearly shows the large $\kappa^{(i)}$ for these two close points.

j	z_j	f_j/g_j	i	$\tau^{(i)}(z_{j+1})$	P.E.	Ω_j	ψ_j	$\kappa^{(i)}$
0	-0.80000000	-0.8661676	0	9.7e+00	0.0e+00	3.6e-02	1.0e+00	1.0e+00
1	-0.60000000	-0.9011914	1	3.1e+02	0.0e+00	5.4e+00	1.0e+00	1.0e+00
2	-0.40000000	-1.0393922	2	5.8e+02	2.3e-16	1.3e-01	1.0e+00	1.0e+00
3	-0.20000000	-1.3435272	3	2.4e+08	1.3e-16	3.8e+00	1.0e+00	3.1e+07
4	-0.19999990	-1.3435274	3	8.1e+03	3.2e-16	3.8e+00	1.0e+00	3.1e+07
5	0.20000000	-4.0972028	4	3.8e+03	4.5e-17	4.4e-01	1.0e+00	1.0e+00
6	0.40000000	40.8827586	5	2.2e+03	8.6e-16	4.1e-01	1.0e+00	1.0e+00
7	0.60000000	3.4850267	6	1.4e+03	2.4e-15	1.7e-01	1.0e+00	1.0e+00
8	0.80000000	1.9732991	7	1.0e+03	1.6e-15	4.4e-01	1.0e+00	1.0e+00
9	1.00000000	1.5000000	8	0.0e+00	6.7e-17	1.0e+00	1.0e+00	1.0e+00

Table 9.10: Experiment 9.10: Cluster of two close points.

We illustrate Theorem 9.2 by using the same data as Table 9.10, except for altering f_4/g_4 to take on the value 0.5 and thereby introducing a numerical discontinuity (a rapid change in function values). Table 9.11, shows that Ω_3 and Ω_4 increased by a factor of 10^{14}. Note that after the interpolation at the discontinuity, the accuracy of the remaining interpolation deteriorates dramatically.

j	z_j	f_j/g_j	i	$\tau^{(i)}(z_{j+1})$	P.E.	Ω_j	ψ_j	$\kappa^{(i)}$
0	-0.80000000	-0.8661676	0	9.7e+00	0.0e+00	7.1e-02	1.0e+00	1.0e+00
1	-0.60000000	-0.9011914	1	3.1e+02	0.0e+00	4.5e+00	1.0e+00	1.0e+00
2	-0.40000000	-1.0393922	2	5.8e+02	2.3e-16	1.1e-01	1.0e+00	1.0e+00
3	-0.20000000	-1.3435272	3	2.4e+08	7.5e-11	4.7e+14	1.0e+00	5.9e+00
4	-0.19999990	0.5000000	3	2.0e+02	4.0e-11	1.5e+15	1.0e+00	5.9e+00
5	0.20000000	-4.0972028	4	1.2e+02	4.0e-09	1.5e-09	1.0e+00	1.0e+00
6	0.40000000	40.8827586	5	2.9e+02	8.7e-09	2.7e-01	1.0e+00	1.0e+00
7	0.60000000	3.4850267	6	1.2e+03	2.6e-08	2.0e+00	1.0e+00	1.0e+00
8	0.80000000	1.9732991	7	2.0e+03	2.2e-08	1.2e-01	1.0e+00	1.0e+00
9	1.00000000	1.5000000	8	0.0e+00	3.3e-08	1.0e+00	1.0e+00	1.0e+00

Table 9.11: Experiment 9.11: An illustration of Theorem 9.2.

We now illustrate the effect of the parameter ψ_j when the close points are located in different places. Here, in Table 9.12, we use the identical data as Table 9.10 except we now distribute the two close points in points z_3, z_8. Notice that none of the $\kappa^{(i)}$ indicates any abnormality. However, ψ_8 increased dramatically.

j	z_j	f_j/g_j	i	$\tau^{(i)}(z_{j+1})$	P.E.	Ω_j	ψ_j	$\kappa^{(i)}$
0	-0.80000000	-0.8661676	0	9.7e+00	0.0e+00	3.6e-02	1.0e+00	1.0e+00
1	-0.60000000	-0.9011914	1	3.1e+02	0.0e+00	5.4e+00	1.0e+00	1.0e+00
2	-0.40000000	-1.0393922	2	5.8e+02	2.3e-16	1.3e-01	1.0e+00	1.0e+00
3	-0.20000000	-1.3435272	3	1.1e+02	1.3e-16	5.0e-02	1.0e+00	1.0e+00
4	0.20000000	-4.0972028	4	5.3e+02	4.5e-17	6.0e-01	1.0e+00	1.0e+00
5	0.40000000	40.8827586	5	3.6e+02	2.3e-16	1.8e-01	1.0e+00	1.0e+00
6	0.60000000	3.4850267	6	3.0e+02	2.4e-15	5.5e-01	1.0e+00	1.0e+00
7	0.80000000	1.9732991	7	1.5e+10	1.6e-15	1.4e+00	1.0e+00	4.3e+00
8	-0.19999990	-1.3435274	7	1.6e+03	3.2e-16	9.1e-01	4.8e+07	4.3e+00
9	1.00000000	1.5000000	8	0.0e+00	6.7e-17	1.0e+00	1.0e+00	1.0e+00

Table 9.12: Experiment 9.12: An illustration of factor ψ_j.

Table 9.13 shows that with discontinuity appearing in separate locations z_3 and z_8, Algorithm 4.2 recognizes it as close points. In Table 9.13, Ω_8 shows no abnormality. However, ψ_8 has the same large value as in Table 9.12.

j	z_j	f_j/g_j	i	$\tau^{(i)}(z_{j+1})$	P.E.	Ω_j	ψ_j	$\kappa^{(i)}$
0	-0.80000000	-0.8661676	0	9.7e+00	0.0e+00	7.1e-02	1.0e+00	1.0e+00
1	-0.60000000	-0.9011914	1	3.1e+02	0.0e+00	4.5e+00	1.0e+00	1.0e+00
2	-0.40000000	-1.0393922	2	5.8e+02	2.3e-16	1.1e-01	1.0e+00	1.0e+00
3	-0.20000000	-1.3435272	3	1.1e+02	1.3e-16	7.1e+05	1.0e+00	1.0e+00
4	0.20000000	-4.0972028	4	5.3e+02	4.5e-17	4.5e-01	1.0e+00	1.0e+00
5	0.40000000	40.8827586	5	3.6e+02	2.3e-16	1.7e-01	1.0e+00	1.0e+00
6	0.60000000	3.4850267	6	3.0e+02	2.4e-15	6.0e-01	1.0e+00	1.0e+00
7	0.80000000	1.9732991	7	1.5e+10	1.6e-15	1.3e+01	1.0e+00	5.5e+00
8	-0.19999990	0.5000000	7	2.8e+02	1.1e-07	3.8e+00	4.8e+07	5.5e+00
9	1.00000000	1.5000000	8	0.0e+00	6.7e-17	1.0e+00	1.0e+00	1.0e+00

Table 9.13: Experiment 9.13: A numerical discontinuity at points z_3 and z_8.

To summarize, we have illustrated in the above experiments that

1. the parameter $\kappa^{(i)}$ captures close points when they appear together,
2. the parameter ψ_j captures close points when they appear in separate locations, and
3. the parameter Ω_j (as also shown in §9.3) captures unattainable points in the data.

9.5 Comparison of Werner's Algorithm

In this section, we compare the performance of Algorithm 4.2 with Werner's algorithm [60]. We begin by highlighting Werner's work expressed in our notation.

Werner's algorithm gives the linear rational interpolant

$$\begin{pmatrix} U^{(k+1)}(z) \\ V^{(k+1)}(z) \end{pmatrix} = \begin{pmatrix} u^{(0)}(z) & t_{n_0+1,n_1}(z) \\ 1 & 0 \end{pmatrix} \cdots \begin{pmatrix} u^{(k-1)}(z) & t_{n_{k-1}+1,n_k}(z) \\ 1 & 0 \end{pmatrix} \begin{pmatrix} u^{(k)}(z) \\ 1 \end{pmatrix}, \tag{9.23}$$

where $u^{(i)}(z) \in \mathcal{P}_l$ and l is specified by the user. With (9.23), the continued-fraction is

$$\frac{U(z)}{V(z)} = u^{(0)}(z) + \cfrac{t_{n_0+1,n_1}(z)}{u^{(1)}(z) + \cfrac{t_{n_1+1,n_2}(z)}{u^{(2)}(z) + \cfrac{t_{n_2+1,n_3}(z)}{u^{(3)}(z) + \ddots \cfrac{t_{n_{k-1}+1,n_k}(z)}{u^{(k)}(z)}}}}. \tag{9.24}$$

The relationship between Werner's representation and ours is

$$\begin{aligned} v^{(i)}(z) &= 1, & (9.25)\\ p^{*(i)}(z) &= t_{n_i+1,n_{i+1}}(z), & (9.26)\\ q^{*(i)}(z) &= 0. & (9.27) \end{aligned}$$

The consequences of Werner's choice of representation is that a polynomial $u^{(i)}(z)$ is used to interpolate the residual of step i, compared to $u^{(i)}(z)/v^{(i)}(z)$ in our case.

If $l = 0$, then Werner's algorithm produces a rational interpolant that is the same as Algorithm 4.2 for the case where each step size is one (i.e., $t_i = 1$), except for a normalization factor. The step size of Werner's algorithm is not limited to one; it can vary with different l. However, the different selections of l for changing its step size are for representation purposes only. They serve no significant purposes relating to stability. As discussed in Chapter 4, for efficiency, l is best to be set at 0. So, in the following, only the case $l = 0$ is considered.

The original algorithm of Werner [60] is similar to Algorithm 4.2 without skipping over ill-posed points on the solution path, except occasionally it moves interpolation points forward: if the partial solution $s^{(0)}(z) \cdots s^{(i)}(z)$ of the i^{th} step accidentally interpolates at a point z_β, where $n_i + 1 \le \beta \le N$ (i.e., the partial solution $|s^{(0)}(z_\beta) \cdots s^{(i)}(z_\beta)|$ has accidentally interpolated the point at z_β), then z_β is brought forth in the i^{th} step and $t_{n_i+1,n_{i+1}}(z)$ is multiplied by the factor $(z - z_\beta)$ to form $(z - z_\beta)t_{n_i+1,n_{i+1}}(z)$, which is similar to the $\theta^{(i)}(z)$ function we introduced. Henceforth, this algorithm is referred to as Werner's algorithm without reordering.

Later, Werner adopted a reordering scheme suggested by Graves-Morris [30] to his original algorithm [61]. The reordering scheme is not an *a posteriori* reordering of the interpolation points; rather, it is a reordering of the interpolation points based on Graves-Morris' error analysis of Werner's Algorithm. There is no proof that this particular choice of reordering is the best choice out of all the possible combinations [30]. Nonetheless, experimental results do show a marked improvement over the original algorithm in many cases. Henceforth, Werner's algorithm with Graves-Morris' reordering scheme is referred to as Werner's algorithm with reordering.

We use the "relative accuracy" $EPS = 10^{-10}$ in Werner's algorithm [61] in double precision. In the implementation of the algorithm, EPS is used in two instances [61]: to test if the residuals have been accidentally interpolated, and to test if the point is unattainable where a modification of the algebraic definition of unattainable points is used.

For comparison, the data from Experiment 9.2 is used. The results are tabulated in Table 9.14. Without reordering, the results resemble those in Experiment 9.2 with $\tau = \infty$, where there was no skipping over the ill-posed points. The differences in the pseudo-error is due to different normalizations of $s^{(i)}(z)$. In particular, it is due to a different representation of the interpolation value: Werner uses f_j/g_j as opposed to $(g_j \quad f_j)$.

			Werner's Algorithm, P.E.	
j	z_j	f_j/g_j	W/O Reordering	Reordering
0	0.90025857	2.58554998	0.0e+00	0.0e+00
1	-0.53772297	0.41353851	5.6e-17	0.0e+00
2	0.21368517	1.36179193	0.0e+00	0.0e+00
3	-0.02803506	1.02153372	2.2e-16	4.4e-16
4	0.78259793	-0.29426374	6.5e-06	0.0e+00
5	0.52419367	0.62633299	1.0e-06	1.1e-16
6	-0.08706467	-0.98027740	1.1e-05	2.2e-16
7	-0.96299271	-0.72221824	3.0e-07	3.3e-16
8	0.64281433	-0.06224507	2.9e-06	0.0e+00
9	-0.11059327	-2.90269834	1.2e-04	0.0e+00
10	0.23086470	0.75638402	4.3e-06	1.1e-16
11	0.58387407	-1.96415805	4.2e-06	0.0e+00
12	0.84362594	-0.12714617	2.1e-06	0.0e+00
13	0.47641449	0.41801396	4.0e-07	0.0e+00
14	-0.64746771	13.76489290	6.1e-04	1.8e-15
15	-0.18858757	-1.72741382	4.8e-05	2.2e-16

Table 9.14: Experiment 9.14: Two Ill-posed Points: $O(\zeta_1) = 10^{-9}$ and $O(\zeta_2) = 10^{-9}$.

With reordering, the pseudo-error is reduced significantly. If this reordering was used in Algorithm 4.2, the low pseudo-error is also apparent for both cases: with and without skipping over ill-posed points. The results are tabulated in Table 9.15. Indeed, since both with and without skipping over ill-posed points give identical results, one can conclude that no ill-posed points were present.

			$\tau = 10^5$			$\tau = \infty$		
j	z_j	f_j/g_j	i	$\tau^{(i)}(z_{j+1})$	P.E.	i	$\tau^{(i)}(z_{j+1})$	P.E.
0	0.64281433	-0.06224507	0	2.8e+01	0.0e+00	0	2.8e+01	0.0e+00
1	0.58387407	-1.96415805	1	5.0e+01	7.4e-17	1	5.0e+01	7.4e-17
2	0.84362594	-0.12714617	2	4.6e+01	2.5e-17	2	4.6e+01	2.5e-17
3	0.90025857	2.58554998	3	1.5e+02	8.0e-17	3	1.5e+02	8.0e-17
4	0.47641449	0.41801396	4	2.5e+00	2.3e-16	4	2.5e+00	2.3e-16
5	-0.64746771	13.76489290	5	2.4e+01	1.3e-17	5	2.4e+01	1.3e-17
6	-0.53772297	0.41353851	6	1.6e+01	1.2e-16	6	1.6e+01	1.2e-16
7	0.21368517	1.36179193	7	4.1e+02	2.6e-16	7	4.1e+02	2.6e-16
8	0.23086470	0.75638402	8	7.0e+02	1.9e-16	8	7.0e+02	1.9e-16
9	0.78259793	-0.29426374	9	3.7e+02	4.3e-17	9	3.7e+02	4.3e-17
10	0.52419367	0.62633299	10	2.0e+01	0.0e+00	10	2.0e+01	0.0e+00
11	-0.02803506	1.02153372	11	5.0e+01	1.7e-16	11	5.0e+01	1.7e-16
12	-0.08706467	-0.98027740	12	5.6e+02	4.5e-16	12	5.6e+02	4.5e-16
13	-0.11059327	-2.90269834	13	7.6e+01	4.5e-16	13	7.6e+01	4.5e-16
14	-0.18858757	-1.72741382	14	2.9e+01	7.0e-17	14	2.9e+01	7.0e-17
15	-0.96299271	-0.72221824	15	—	3.9e-16	15	—	3.9e-16

Table 9.15: Experiment 9.15: Data from Experiment 9.14 after reordering.

Werner's reordering algorithm performed much better than the one without reordering. However, the reordering approach is considered to be not an inductive approach by others [33], as one cannot add more data and proceed to higher degrees since the interpolation points need to be reordered. As such, if more points are added, one must start over. Nonetheless, using the reordering scheme, Werner's algorithm produces accuracy similar to the case where $\tau = 10^5$. While it helped in the above experiment, there is no proof that such a reordering scheme would remove ill-posed points completely.

To illustrate this, consider the experiment given in Table 9.16, where the reordering did not result in smaller pseudo-errors compared to the original order. Thus, the reordered sequence of data may not give smaller residual errors in general.

With reordered sequence, Algorithm 4.2 skipped over the ill-posed point. (See Table 9.17).

Werner's algorithm has major problems when dealing with accidentally interpolated points and in identifying unattainable points: The numerical algorithm uses the same

			Werner's Alg., P.E.		Algorithm 4.2 ($\tau = 10^5$)		
j	z_j	f_j/g_i	W/ Reordg.	W/O	i	$\tau^{(i)}(z_{j+1})$	P.E.
3	0.9	9.0000000000e-01	1.1e-16	0.0e+00	0	3.5e+00	5.8e-17
1	-0.2	1.0000000000e+06	1.2e-10	0.0e+00	1	4.9e+00	0.0e+00
0	-0.9	3.0000000000e-01	0.0e+00	5.6e-17	2	1.1e+01	0.0e+00
2	0.5	3.0000000010e-01	1.0e-10	5.6e-17	3	-	2.1e-16

Table 9.16: Experiment 9.16: Werner's Reordering Scheme.

			($\tau = 10^5$)			($\tau = \infty$)		
j	z_j	f_j/g_i	i	$\tau^{(i)}(z_{j+1})$	P.E.	i	$\tau^{(i)}(z_{j+1})$	P.E.
0	-0.9	3.0000000000e-01	0	1.7e+00	0.0e+00	0	1.7e+00	0.0e+00
2	0.5	3.0000000010e-01	1	8.2e+10	0.0e+00	1	8.2e+10	4.3e-17
1	-0.2	1.0000000000e+06	1	9.1e+10	3.1e-17	2	9.5e+10	1.2e-06
3	0.9	9.0000000000e-01	1	-	5.8e-17	3	-	1.5e-06

Table 9.17: Experiment 9.17: Werner's Reordering Scheme.

concept as the algebraic algorithm by simple relationships with "equals to 0" in the algebraic case replaced by "less than EPS". The algorithm breaks down completely in the bordering cases where points are around the threshold of accidentally interpolated by the partial solution. For example, the data points $\{(-0.9, 0.3), (-0.3, 0.3), (0.5, 0.30000000017), (0.9, 0.30000000017)\}$ has a solution $U(z)/V(z) = (0.3+0.30000000017)/2$ that interpolates the four points within the specified tolerance. But Werner's algorithm reports 'problem not solvable' even with the reordering scheme. It should also be noted that Werner's algorithm produced a solution without any warning in the four experiments of the previous section on close points. Thus, Werner's algorithm is not reliable.

Furthermore, Werner's numerical algorithm [61] does not detect unattainable points accurately. In his original paper [60], Werner does not offer any suggestions as to how to treat unattainable points numerically. Instead, he leaves the interpretation of the algebraic definition of unattainable points to the users. In his implementation of the algorithm [61], Werner uses the algebraic definition of an unattainable point with a minor modification using EPS. As described in §9.3, this strategy does not work well numerically.

9.6 Comparison with Gaussian Elimination

In this section, we compare Algorithm 4.2 with the well-known Gaussian elimination method. Aside from the fact that Gaussian elimination is an $O(N^3)$ algorithm, we illustrate that it may not be a good choice for rational interpolation for a large data set.

As mentioned in Chapter 1, the straightforward way to solve the linear rational interpolation problem to obtain $(U(z), V(z))$ is to directly solve the system of equations in (1.5) by using the Gaussian elimination method. Since Gaussian elimination is known to be stable, it always gives small residuals (i.e., $|U(z_j)g_j + V(z_j)f_j|, j = 0, \ldots, N$ is small) for the linear rational interpolation problem. However, the goal is to obtain a small residual in the rational form (i.e., we want small $|U(z_j)/V(z_j) + f_j/g_j|, j = 0, \ldots, N$). Hence, using Gaussian elimination to solve the linear rational (1.5), the representation of the rational form is inevitably $U(z)/V(z)$, where $U(z)$ and $V(z)$ are polynomials in the forms of (1.2).

For small N, say $N < 20$, Gaussian elimination does indeed provide $(U(z), V(z))$ such that $U(z)/V(z)$ gives good results for the rational interpolation. It does not give good results only when the problem is ill-conditioned or when the problem contains unattainable points.

For large data sets, the representation of one polynomial over another does not seem to be a good candidate for the rational interpolant. This is because the condition number of the problem $\kappa(A)$ in (5.1) is large. The following two experiments are designed to illustrate that the representation of rational function plays an important role in rational interpolation when the number of input data N is large.

The first experiment is constructed using 30 randomly generated points over $z \in [-1, 1]$. We ask the algorithms to generate a rational function of the degree type $[15, 14]$. We see in Table 9.18 that the P.E. using Gaussian elimination is considerably larger compared to that obtained using Algorithm 4.2. For this experiment, the condition number was 6.5×10^8. Note that the large unattainability measures for Gaussian elimination $(1/(|U(z_j)|+|V(z_j)|))$ correspond to the large pseudo-errors (see Remark 8.1).

In contrast, we note that the P.E. using Algorithm 4.2 was smaller. Here, the representation of the rational function is a continued-fraction. In this form, it separates points into sections, where each section is interpolated by a low degree interpolant. But as pointed out earlier, low degree $U(z)/V(z)$ gives good rational interpolation. In this experiment the condition of this problem was at most $\tau = 10^5$ because $\kappa(S^{(i)}(z_j)) \leq \tau$ for all $i = 0, \cdots, k$, $j = n_i + 1, \cdots, n_i + t_i$, and $\kappa^{(i)} = 1$, $i = 0, \cdots, k$.

To further support that the representation plays an important role in rational interpolation for large N, in the next experiment, we use a larger N to amplify this effect. In this experiment we use $N = 233$ so that the rational interpolant is of type $[116, 116]$. These 233 data points are the closing indices of the Dow Jones Industrial Average Index (DJII) taken during the 233 business days in 1998. We first map these data into the range $[-1, 1]$ using (9.2) and each data point is evenly distributed in $[-1, 1]$.

The results presented in Table 9.19 illustrate the differences between the two formulation of the problem. With Gaussian elimination, the lowest P.E. was $O(10^{-10})$ and the highest P.E. was $O(1)$, with the majority (over 80%) being $O(10^{-2})$. For this experiment the condition number was 3.1×10^{20}. In comparison, Algorithm 4.2 with $\tau = 10^7$, the highest P.E. was $O(10^{-14})$ and lowest was $O(10^{-17})$. The condition number for this problem was at most $\tau = 10^7$ because, again, $\kappa(S^{(i)}(z_j)) \leq \tau$ for all $i = 0, \cdots, k$, $j = n_i + 1, \cdots, n_i + t_i$, and $\kappa^{(i)} = 1$, $i = 0, \cdots, k$. Clearly, this experiment showed that Algorithm 4.2 with a continued-fraction representation performed better than Gaussian elimination method with the rational interpolant represented in a quotient of two polynomials.

Note that the unattainability measures $(1/(|U(z_j)| + |V(z_j)|))$ for Gaussian elimination are $O(10^{16})$ for the majority of the points. Although we cannot generalize this observation for all problems with large N, when we examined the coefficients of $(U(z), V(z))$ in Experiment 9.19, it was found that the coefficients of low degrees (0–20) are very small (e.g., the coefficients of the degrees 0–3 are $O(10^{-16})$). On the other hand, for the coefficients of higher degrees (21–116), the coefficients are $O(10^{-3})$. Hence, with $z_j \in [-1, 1]$, $(1/(|U(z_j)| + |V(z_j)|))$ is large for most z_j. Since the majority of the coefficients are relatively large, we conclude that there must be cancellation error when computing $U(z_j)$ and $V(z_j)$.

9.7 Stability Parameter Tolerance τ

In this section, we discuss the significance of the size of the stability parameter τ.

We first note that the pseudo-error bound in Lemma 8.2 (and Corollary 8.1) is bounded proportional to $(\tau\bar{\psi}_j)^2$. It is observed that the quadratic component in $(\tau\bar{\psi}_j)^2$ gives a gross overestimate of the error: The reason is that the final expression in Lemma 8.2 (and Corollary 8.1) is the result of an application of the inequality $\kappa(\bar{S}^{(i)})(z_j) \leq \tau\bar{\psi}_j$ (one in Lemma 8.2 (or Corollary 8.1)). However, the size of $\kappa(\bar{S}^{(i)})(z_j)$ can be substantially smaller than $\tau\bar{\psi}_j$. Our numerical results indicate that operational bound $(\tau\bar{\psi}_j)$ is more appropriate.

This does not mean that the bounds lack merit. As pointed out by Wilkinson [62], *a priori* bounds are not, in general, quantities that should be used in practice. The reason being, these bounds are much weaker than what they might have been because of the necessity of restricting the mass of detail to a reasonable level and because of the limitations imposed by expressing the errors in terms of norms. Thus, practical error bounds should usually be determined by some form of *a posteriori* error analysis, since this takes full advantage of the statistical distribution of rounding errors and of any special features.

			G. E.		Algorithm 4.2			
j	z_j	f_j/g_j	P.E.	**	i	$\tau^{(i)}(z_{j+1})$	P.E.	Ω_j
0	0.83418	-1.237374	9.7e-13	3.6e+05	0	3.7e+00	0.0e+00	3.0e+02
1	-0.75344	-0.930254	1.2e-12	1.9e+04	1	2.0e+02	5.8e-17	4.4e+01
2	-0.97310	0.862671	3.5e-14	1.0e+03	2	1.3e+02	7.7e-16	3.7e-01
3	-0.26061	-0.276569	1.1e-12	8.3e+04	3	2.5e+02	8.7e-17	2.5e-01
4	0.39728	0.730844	7.2e-10	2.3e+08	4	1.8e+04	1.2e-15	1.3e+00
5	0.77869	1.121567	8.1e-13	1.7e+05	5	1.2e+03	1.7e-13	7.3e-02
6	0.18754	-0.284016	1.3e-10	8.2e+06	6	1.1e+03	3.5e-16	4.2e+00
7	-0.68661	-0.776042	1.0e-12	3.4e+04	7	4.4e+02	5.6e-16	7.5e+01
8	-0.36662	0.652790	2.5e-12	6.1e+05	8	6.8e+02	1.0e-15	3.7e-01
9	-0.53320	0.502309	1.3e-11	6.8e+05	9	8.0e+03	2.0e-15	2.9e+00
10	-0.98315	0.645265	3.1e-14	6.3e+02	10	9.0e+02	2.7e-16	1.7e+01
11	-0.20619	-0.403711	2.9e-12	7.0e+04	11	3.5e+03	2.8e-16	1.5e+01
12	0.29973	0.346760	7.5e-10	1.3e+09	12	1.7e+03	1.0e-14	1.4e+02
13	-0.82999	0.476665	1.9e-13	8.0e+03	13	6.3e+03	1.9e-14	4.0e-01
14	0.53761	4.102545	1.5e-10	1.1e+07	14	8.6e+03	1.2e-14	1.2e+01
15	0.93940	8.110617	1.6e-12	1.5e+05	15	1.6e+04	1.0e-14	6.8e+01
16	0.42959	-3.406006	6.8e-09	4.6e+08	16	6.1e+03	7.0e-15	2.9e+01
17	0.56392	1.697704	1.2e-10	7.2e+06	17	4.1e+03	6.6e-15	1.3e+00
18	-0.52487	1.610918	2.1e-11	1.4e+06	18	1.4e+03	8.2e-15	2.6e+01
19	-0.60853	1.321497	1.3e-12	1.6e+05	19	9.0e+02	1.2e-14	2.3e+00
20	-0.47357	-0.231178	1.1e-11	2.6e+05	20	1.7e+04	1.4e-15	1.1e+00
21	0.42757	-10.472610	1.1e-08	5.7e+08	21	7.4e+03	2.8e-15	1.0e+01
22	0.95519	-0.121938	7.7e-13	3.8e+04	22	2.6e+03	1.2e-14	1.6e-01
23	0.27424	-12.802512	4.3e-09	1.5e+08	23	6.3e+02	1.4e-14	6.7e+00
24	0.09184	1.839470	2.5e-11	1.5e+06	24	8.3e+03	1.1e-14	1.6e-01
25	0.69611	1.355777	5.0e-12	4.3e+05	25	6.2e+03	2.3e-14	7.3e+01
26	0.60419	0.011102	6.0e-11	1.4e+07	26	6.3e+03	1.8e-14	9.8e-02
27	0.33661	1.583598	2.1e-08	2.4e+09	27	3.7e+04	8.8e-16	1.4e+02
28	0.34196	-2.144574	2.5e-07	7.9e+09	28	1.4e+04	3.4e-16	4.5e-01
29	0.64128	3.705727	2.7e-11	2.6e+06	29	—	2.2e-14	1.0e+00

Table 9.18: Experiment 9.18: G.E. vs. Algorithm 4.2 ($** = \frac{1}{|U(z_j)|+|V(z_j)|}$).

			G. E.		Algorithm 4.2			
j	z_j	f_j/g_j	P.E.	**	i	$\tau^{(i)}(z_{j+1})$	P.E.	Ω_j
0	-0.99141	0.78984	1.7e-10	5.8e+07	0	2.3e+02	6.2e-17	1.1e+02
1	-0.98283	0.89142	3.2e-09	3.1e+08	1	2.7e+04	8.2e-16	1.3e+01
2	-0.97424	0.94192	6.0e-08	3.7e+09	2	1.5e+04	4.0e-16	1.8e+02
3	-0.96566	0.83923	1.7e-07	1.8e+10	3	8.2e+04	2.2e-15	8.0e+00
4	-0.95708	0.82968	5.6e-07	4.0e+10	4	2.3e+04	6.7e-16	4.4e+00
5	-0.94849	0.81229	3.9e-06	3.6e+11	5	2.8e+04	1.8e-16	3.8e+02
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
100	-0.13304	0.09820	6.8e-01	9.8e+16	100	1.6e+04	1.0e-15	1.2e+02
101	-0.12446	0.03339	1.5e+00	1.2e+17	101	9.0e+03	2.5e-15	9.6e+01
102	-0.11588	0.35971	1.7e+02	1.5e+17	102	5.7e+03	2.5e-15	8.4e+00
103	-0.10729	0.46493	1.2e+00	8.3e+16	103	6.6e+03	2.5e-15	1.1e-01
104	-0.09871	0.49944	6.2e-01	5.5e+16	104	4.2e+04	1.2e-15	2.5e-01
105	-0.09012	0.52104	4.4e-01	4.1e+16	105	1.4e+06	7.3e-17	9.8e+01
106	-0.08154	0.53788	3.7e-01	3.3e+16	106	4.9e+06	2.3e-15	3.8e+00
107	-0.07296	0.52385	3.6e-01	2.8e+16	107	1.1e+06	3.0e-15	3.4e+00
108	-0.06437	0.73260	2.0e-01	2.6e+16	108	3.9e+05	2.6e-15	4.4e+01
109	-0.05579	0.73260	2.1e-01	2.5e+16	109	1.8e+05	3.8e-15	7.3e-01
110	-0.04721	0.79939	1.9e-01	2.5e+16	110	9.6e+04	4.3e-16	1.2e+04
111	-0.03862	0.91442	1.4e-01	2.6e+16	111	5.7e+04	2.4e-15	4.9e+01
112	-0.03004	0.96044	1.6e-01	3.0e+16	112	3.6e+04	4.5e-16	2.7e+00
113	-0.02145	0.94978	2.3e-01	3.8e+16	113	2.4e+04	1.0e-14	3.4e+01
114	-0.01287	0.84764	4.7e-01	5.4e+16	114	2.3e+04	3.9e-15	1.6e+01
115	-0.00429	0.85971	1.5e+00	1.1e+17	115	1.2e+04	8.4e-16	1.8e+01
116	0.00429	0.69697	8.4e-01	1.9e+17	116	1.7e+04	5.2e-16	8.4e+00
117	0.01287	0.70735	2.3e-01	9.3e+16	117	7.0e+03	1.2e-15	4.6e+00
118	0.02145	0.68996	8.9e-02	5.0e+16	118	2.1e+04	5.9e-16	8.6e+00
119	0.03004	0.78280	9.0e-02	3.5e+16	119	4.0e+04	6.9e-16	1.8e+01
120	0.03862	0.68479	2.0e-02	2.9e+16	120	1.4e+04	1.8e-15	6.2e-03
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
228	0.96566	-0.71280	4.0e-07	3.1e+10	228	7.3e+04	4.3e-15	7.0e+00
229	0.97424	-0.60415	3.8e-08	5.4e+09	229	2.1e+04	9.2e-15	2.0e+00
230	0.98283	-0.59979	2.6e-08	1.9e+09	230	9.6e+03	9.9e-15	1.7e+02
231	0.99141	-0.52056	5.0e-09	3.4e+08	231	1.5e+05	3.0e-15	7.1e+00
232	1.00000	-0.53582	1.0e-10	2.0e+07	232	—	8.5e-15	1.0e+00

Table 9.19: Experiment 9.19: G.E. vs. Algorithm 4.2 ($** = \frac{1}{|U(z_j)|+|V(z_j)|}$).

Notice that if τ is set to be too small, then $\bar{\tau}^{(i)}(z_{n_i+1}) \leq \tau$ is never achieved, which can lead to an $O(N^4)$ algorithm. On the other hand, if τ is set to be too large, the degree of accuracy of the interpolation is compromised. Thus, a balance between the too extremes is required.

The optimal size for τ varies from experiments to experiments. The following are some guidelines for choosing a suitable size for τ. In a double precision setting, first use a moderately small τ (e.g. $\tau = 10^5$) for several runs of the particular type of problem of interest. If the resulting output contains large step sizes, increase τ gradually and repeat until the step sizes are small or the maximum allowable size of τ is reached.

9.8 Summary

The above experiments show that Algorithm 4.2 handles ill-posed points without any difficulties. And, for stable problems, Algorithm 4.2 gives small pseudo-errors for the Cauchy problem. However for ill-conditioned problems, it gives relatively large pseudo-errors. But for these cases, Algorithm 4.2 indicates the points that cause instability by using the parameters $\kappa^{(i)}$ and ψ_j for close (and/or duplicate) points. Also, the parameter Ω_j alerts the user for data containing unattainable points.

Chapter 10

Summary and Conclusions

In this thesis, we have developed and analyzed an algorithm—Algorithm 4.2—for numerically computing rational interpolants for the Cauchy interpolation problem. For interpolating $\{(z_j, f_j, g_j)\}$, for $j = 0, \ldots, N$, Algorithm 4.2 gives two polynomials

$$\begin{pmatrix} U(z) \\ V(z) \end{pmatrix} = s^{(0)}(z) \cdots s^{(k-1)}(z) \begin{pmatrix} u^{(k)}(z) \\ v^{(k)}(z) \end{pmatrix}. \tag{10.1}$$

Our goal is to solve not the linear problem but rather the rational (nonlinear) problem. In rational form, we express $U(z)/V(z)$ as a continued-fraction. This continued-fraction form is obtained directly from (10.1); we do not need to expand $(\, U(z) \quad V(z)\,)^t$.

There are advantages in expressing $U(z)/V(z)$ as a continued-fraction. Fewer operations are required for evaluation. More importantly, when the solution is expressed as a continued-fraction, the nonlinear problem has certain desirable stability properties.

To evaluate the accuracy of interpolation by a continued-fraction, we introduced a nonlinear point-wise measure. This measure places greatest emphasis on function values of size $O(1)$; it accommodates large and small values by assigning diminishing weights to them. For small values, it measures the absolute error of $U(z_j)/V(z_j) - f_j/g_j$ and for large values it measure its inverse (i.e., $V(z_j)/U(z_j) - g_j/f_j$).

In terms of this point-wise measure, we showed that the continued-fraction representation gives a small error (we call this the pseudo-error) in all cases, except when the problem is ill-conditioned. By definition, we say that a problem is well-conditioned if the condition number defined in (5.61) is not too large. With this definition, the error bounds are used to show that Algorithm 4.2 is weakly stable for solving the nonlinear problem. That is, Algorithm 4.2 gives a good solution whenever the problem is well-conditioned.

In the stability proofs, we showed that at z_j the pseudo-error is bounded by $O((\tau\psi_j)^2\mu/(|u^{(i)}(z_j)| + |v^{(i)}(z_j)|))$. Experimentally, however, we illustrated in Chapter 9 that the pseudo-error is bounded instead by $O((\tau\psi_j)\mu/(|u^{(i)}(z_j)| + |v^{(i)}(z_j)|))$. Problems which contain nearly duplicate points are identified *a posteriori* by large value of ψ_j at such points[1]. Problems which contain nearly unattainable points are identified by large values of Ω_j at such points, where Ω_j includes the term $(|u^{(i)}(z_j)| + |v^{(i)}(z_j)|)$ in its expression. That is, small values of $(|u^{(i)}(z_j)| + |v^{(i)}(z_j)|)$, or correspondingly large values of Ω_j, imply that z_j is nearly unattainable. So, if the problem does not contain unattainable points, for all well-conditioned problems where there are no nearly duplicate points and all the sub-problems are well-conditioned, the pseudo-error is bounded by $O(\tau\mu)$ in practice.

We compared Algorithm 4.2 experimentally with two well known algorithms, namely, Werner's algorithm and the Gaussian elimination method.

We showed that Werner's algorithm without reordering of interpolation points does not interpolate accurately in the presence of ill-posed points. With a certain reordering of data, the accuracy of interpolation improves substantially even in the presence of ill-posed points in the data (Experiment 9.14). Nevertheless, Werner's algorithm (with reordering) in comparison with Algorithm 4.2 has a number of disadvantages, namely:

- The requirement of reordering of interpolation points is a drawback because the algorithm then becomes non-inductive [33] in the sense that one cannot add further data and proceed to higher degrees. It is not a restriction, of course, if all the data is given *a priori*.
- Even with reordering and the resulting improvement of accuracy, Werner's algorithm in all our experiments always gave larger pseudo errors than did Algorithm 4.2 (e.g., Experiment 9.16).
- A proof of the stability of Werner's algorithm is not yet available. In this direction, it has been shown that by reordering of the interpolation points [64, 29], algebraically one can remove all singular blocks (except possibly the last one) on a solution path. A numerical equivalent has not yet been found.
- As demonstrated in §9.5, Werner's algorithm is not reliable because it does not always alert the user when a problem is ill-conditioned, nor does it always give a solution when one is available within the specified tolerance.

Algorithm 4.2 matches the accuracy of Gaussian elimination for interpolation problems where N is small. We showed in §9.6 for interpolation problems where N is large, Algorithm 4.2 interpolates more accurately than Gaussian elimination. Since it is well known that Gaussian elimination gives small linear residual errors, the lower interpolation accuracy is due to the representation of the rational function, i.e., a large condition number of the problem, in this case where one polynomial over another was used. On the other hand,

[1] For cases where the nearly duplicate points appear in a sequence, $\kappa^{(i)}$ is used instead as illustrated in §9.4.

using the formulation of (5.6), the condition number of the problem remains small even for interpolation problems with large N.

We now address other general open questions in rational interpolation.

The definition of an unattainable point in a numerical setting still needs further thought. As given in Definition 2.1, an unattainable point is described with respect to its exact interpolant of a certain type. Numerically, lacking the exact solution, we defined unattainability with respect to the computed solution. Although this may be a natural extension from the original definition, questions about the relationship between algebraic and numerical unattainability need to be addressed.

In this thesis, we focussed on Cauchy interpolation, where the interpolation points are distinct. More general formulations are discussed in the literature. Interpolation points at which the function values as well as one or more derivatives are specified are called the confluent points. Problems that require finding a rational function which interpolates data containing confluent points are called the rational Hermite interpolation problems [8, 21, 23, 32]. Other names such as the osculatory rational problem [20, 50], the multipoint-Padé problem [5] or the Newton-Padé interpolation problem [9, 36] are also used. There are a number of algebraic algorithms [9, 16, 33, 55] which compute rational interpolants allowing confluent points in the data. However, the performance of these algorithms in a numerical setting has not yet been studied. Indeed, without appropriate modifications, it is clear that these algorithms are not numerically stable. As such, one of the challenges is to develop a numerically stable algorithm for the rational Hermite interpolation problem.

Another related problem is the Padé interpolation (approximation) problem where all the interpolation points are the same, with a function value and derivatives specified at that point. Note that this interpolation is a special case of the Hermite interpolation and it is different from the Cauchy problem. Numerically *fast* stable algorithms [18, 17, 7, 58, 35] have been developed for the Padé interpolation problem. It turns out that by first interpolating the data by a polynomial of sufficient degree, the problem of rational interpolation becomes one of Padé approximation (see [32, 12], for example).

So, as long as there are fast algorithms for stable polynomial interpolation, it appears that they can be used together with fast Padé algorithms to develop fast stable algorithms for rational interpolation. Unfortunately, polynomial interpolation breaks down when there are poles or large function values in the data. Nevertheless, there may be situations where this approach may be fruitful (e.g., when the interpolation data is already represented as a polynomial). We leave this as a topic for future research.

There are other *fast* algorithms designed for (linear) discrete least squares rational approximation [14, 55, 56, 57]. It is an interesting topic to study the behavior of unattainability when the interpolation problem is solved as a least squares problem (i.e., $L + M \ll N$). As a special case, when the degree of the rational form is sufficiently high (i.e., $L + M = N$), one can use these algorithms to solve the linear rational interpolation problem. Further studies are required to assess the performance of these algorithms in the nonlinear case.

There are also *superfast* algorithms [2, 32] requiring $O(N \log^2 N)$ operations for computing rational interpolants for the Cauchy problem. However, numerically stable *superfast* algorithms have not been studied. Thus, the development of a stable *superfast* algorithm for the rational interpolation is another challenging research problem.

Bibliography

[1] B. D. Q. Anderson and A. C. Antoulas. Rational interpolation and state-variable realizations. *Linear Algebra and Its Applications*, 137/138:479–509, 1990.

[2] A. C. Antoulas and B. D. Q. Anderson. On the scalar rational interpolation problem. *IMA Journal of Mathematical Control & Information*, 3:61–88, 1986.

[3] A. C. Antoulas and B. D. Q. Anderson. On the problem of stable rational interpolation. *Linear Algebra and Its Applications*, 124:301–329, 1989.

[4] K. E. Atkinson. *An Introduction to Numerical Analysis*. John Wiley & Sons, New York, second edition, 1989.

[5] JR. G. A. Baker and P. Graves-Morris. *Padé Approximants*. Cambridge University Press, New York, second edition, 1996.

[6] B. Beckermann. A reliable method for computing M-Padé approximants on arbitrary staircases. *Journal of Computational and Applied Mathematics*, 40:19–42, 1992.

[7] B. Beckermann. The stable computational of formal orthogonal polynomials. *Numerical Algorithms*, 11:1–23, 1996.

[8] B. Beckermann and C. Carstensen. A reliable modification of the cross rule for rational Hermite interpolation. *Numerical Algorithms*, 3:29–44, 1992.

[9] B. Beckermann and C. Carstensen. QD-type algorithms for the nonnormal Newton-Padé approximation table. *Constructive Approximation*, 12:307–329, 1996.

[10] B. Beckermann and G. Labahn. Fraction-free computation of matrix rational interpolants and matrix gcd's. *Publication ANO 375, Université de Lille, Submitted*, 1997.

[11] V. Belevitch. Interpolation matrices. *Philips Research Reports*, 25:387–369, 1970.

[12] R. Brent, F. G. Gustavson, and D. Y. Y. Yun. Fast solution of Toeplitz systems of equations and computation of Padé approximants. *Journal of Algorithms*, 1:259–295, 1980.

[13] A. Bultheel and B. De Moor. Rational approximation in linear systems and control. *Journal of Computational and Applied Mathematics, To appear*, November 1999.

[14] A. Bultheel and M. van Barel. Vector orthogonal polynomials and least squares approximation. *SIAM Journal on Matrix Analysis and Applications*, 16(3):863–885, 1995.

[15] J. R. Bunch. The weak and strong stability of algorithms in numerical linear algebra. *Linear Algebra and Its Applications*, 88/89:49–66, 1987.

[16] S. Cabay, M. H. Gutknecht, and R. Meleshko. Stable rational interpolation? In *MTNS '93*, 1993.

[17] S. Cabay, A. Jones, and G. Labahn. Computation of numerical Padé-Hermite and simultaneous Padé system II: A weakly stable algorithm. *SIAM Journal on Matrix Analysis and Applications*, 17(2):268–297, 1996.

[18] S. Cabay and R. Meleshko. A weakly stable algorithm for Padé approximants and the inversion of Hankel matrices. *SIAM Journal on Matrix Analysis and Applications*, 14(3):735–765, 1993.

[19] A. L. Cauchy. Sur la formule de Lagrange relative à l'interpolation. *Cours D'analyse de L'école Royale Polytechnique*, Note V of Analyse Algébrique:525–529, 1821.

[20] G. Claessens. A new algorithm for osculatory rational interpolation. *Numerische Mathematik*, 27:77–83, 1976.

[21] G. Claessens. The rational Hermite interpolation problem and some related recurrence formulas. *Computers & Mathematics with Applications*, 2:117–123, 1976.

[22] G. Claessens. On the Newton-Padé approximation problem. *Journal of Approximation Theory*, 22:150–160, 1978.

[23] G. Claessens. A useful identity for the rational Hermite interpolation table. *Numerische Mathematik*, 29:227–231, 1978.

[24] Ö. Eğecioğlu and Ç. K. Keç. A fast algorithm for rational interpolation via orthogonal polynomials. *Mathematics of Computation*, 53(187):249–264, 1989.

[25] George E. Forsythe and Cleve B. Moler. *Computer Solution of Linear Algebraic Systems*. Prentice-Hall, 1967.

[26] L. Gemignani. Rational interpolation via orthogonal polynomials. *Computers & Mathematics with Applications*, 26(5):27–34, 1993.

[27] G. H. Golub and C. F. van Loan. *Matrix Computations*. The Johns Hopkins University Press, London, second edition, 1989.

[28] W. B. Gragg. The Padé table and its relation to certain algorithms of numerical analysis. *SIAM Review*, 14(1):1–59, January 1972.

[29] P. R. Graves-Morris. Efficient reliable rational interpolation. In M. G. deBruin and H. van Rossum, editors, *Padé Approximation and Its Applications*, pages 28–63, 1980.

[30] P. R. Graves-Morris. Practical, reliable, rational interpolation. *J. Institute of Mathematics and Its Applications*, 25:267–286, 1980.

[31] P. R. Graves-Morris and T. R. Hopkins. Reliable rational interpolation. *Numerische Mathematik*, 36:121–128, 1981.

[32] F. G. Gustavson and D. Y. Y. Yun. Fast algorithms for rational Hermite approximation and solution of Toeplitz systems. *IEEE Trans. on Circuits and System*, 26(9):750–753, 1979.

[33] M. H. Gutknecht. Continued fractions associated with the Newton-Padé table. *Numerische Mathematik*, 56:547–589, 1989.

[34] M. H. Gutknecht. Block structure and recursiveness in rational interpolation. In E. W. Cheney, C. K. Chui, and L. L. Schumaker, editors, *Approximation Theory VII*, pages 93–130. Academic Press, 1992.

[35] M. H. Gutknecht. Stable row recurrences for the Padé table and generically superfast look-ahead solvers for non-Hermitian Toeplitz systems. *Linear Algebra and Its Applications*, 188/89:351–421, 1993.

[36] M. H. Gutknecht. The multipoint Padé table and general recurrences for rational interpolation. In A. Cuyt, editor, *Nonlinear Numerical Methods and Rational Approximation II*, pages 109–135. Khuwer Academic Publishers, 1994.

[37] N. J. Higham. *Accuracy and Stability of Numerical Algorithms.* SIAM, Philadelphia, 1996.

[38] F. B. Hilderband. *Introduction to Numerical Analysis.* McGraw-Hill, New York, 1956.

[39] C. G. J. Jacobi. Über die darstellung einer reihe gegebener werte durch eine gebrochene rationale funktion. *Crelle J. für die Reine und Angew. Math.*, 30:127–156, 1846.

[40] G. K. Kristiansen. A rootfinder using a nonmonotone rational approximation. *SIAM Journal on Scientific and Statistical Computing*, 6:118–127, 1985.

[41] L. Kronecker. Zur theorie der elimination einer variabelen aus zwei algebraisthen gleichungen. In *Monatsberichte der königlich*, pages 535–600. preussischen Akademie der Wissenschaften zu, Berlin, 1881.

[42] F. M. Larkin. A class of methods for tabular interpolation. *Proceedings of the Cambridge Philological Society*, 63:1101–1114, 1967.

[43] F. M. Larkin. Some techniques for rational interpolation. *Computer Journal*, 10:178–187, 1967.

[44] F. M. Larkin. Root-finding by fitting rational functions. *Mathematics of Computation*, 35(151):803–816, July 1980.

[45] H. Maehly and C. Witzgall. Tschebyscheff-approximationed in kleinen intervallen II, stetigkeitssätze für gebrochen rationale approximationed. *Numerische Mathematik*, 2:293–307, 1960.

[46] J. Meinguet. On the solubility of the Cauchy interpolation problem. In A. Talbot, editor, *Approximation Theory*, pages 137–163. Academic Press, New York, 1970.

[47] E. H. Neville. Iterative interpolation. *Journal of Indian Mathematical Society*, 20:87–120, 1934.

[48] V. Norton. Algorithm 631, finding a bracketed zero by Larkin's method of rational interpolation. *ACM Trans. on Mathematical Software*, 11:120–134, 1985.

[49] K. Rost and Z. Vavrin. Recursive solution of Löwner-Vandermonde systems of equations. *Linear Algebra and Its Applications*, 233:51–65, 1996.

[50] H. E. Salzer. Note on osculatory rational interpolation. *Mathematics of Computation*, 16:486–491, 1962.

[51] G. W. Stewart. *Introduction to Matrix Computations.* Academic Press, New York, 1973.

[52] J. Stoer. Algorithmen zur interpolation mit rationalen funktionen. *Numerische Mathematik*, 3:285–304, 1961.

[53] J. Stoer. A direct method for Chebyshev approximation by rational functions. *Journal of the Association for Computing Machinery*, 11(1):59–69, 1964.

[54] M. van Barel and A Bultheel. A new approach to the rational interpolation problem. *Journal of Computational and Applied Mathematics*, 32:281–289, 1990.

[55] M. van Barel and A Bultheel. A new formal approach to the rational interpolation problem. *Numerische Mathematik*, 62:87–122, 1992.

[56] M. van Barel and A Bultheel. Discrete linearized least squares approximation on the unit circle. *Journal of Computational and Applied Mathematics*, 50:545–563, 1994.

[57] M. van Barel and A Bultheel. Orthogonal polynomial vectors and least squares approximation for a discrete inner product. *Electronic Transactions on Numerical Analysis*, 3:1–23, 1995.

[58] M. van Barel and A Bultheel. Look-ahead methods for block Hankel systems. *Journal of Computational and Applied Mathematics*, 86:311–333, 1997.

[59] M. van Barel and Z. Vavrin. Inversion of a block löwner matrix. *Journal of Computational and Applied Mathematics*, 69:261–284, 1996.

[60] H. Werner. A reliable method for rational interpolation. In L. Wuytack, editor, *Padé Approximation and Its Applications*, pages 257–277. Berlin-Heidelberg-New York, Springer, 1979.

[61] H. Werner. Algorithm 51, a reliable and numerically stable program for rational interpolation of Lagrange data. *Computing*, 31:269–286, 1983.

[62] J. H. Wilkinson. Modern error analysis. *SIAM Review*, 13:548–568, 1971.

[63] L. Wuytack. An algorithm for rational interpolation similar to the qd algorithm. *Numerische Mathematik*, 20:418–424, 1973.

[64] L. Wuytack. On some aspects of the rational interpolation problem. *SIAM Journal on Numerical Analysis*, 11(1):52–60, 1974.