Department of Mathematics
University of Fribourg (Switzerland)

Applications of Linear Barycentric Rational Interpolation

THESIS

presented to the Faculty of Science of the University of Fribourg (Switzerland)
in consideration for the award of the academic grade of
Doctor scientiarum mathematicarum

by

Georges Klein

from the

Grand Duchy of Luxembourg

Thesis No: 1762
epubli.de
2012

Accepted by the Faculty of Science of the University of Fribourg (Switzerland) upon the recommendation of:

PROF. DR. JEAN-PAUL BERRUT, Thesis Supervisor
University of Fribourg, Switzerland

PROF. DR. MICHAEL S. FLOATER, Examinator
University of Oslo, Norway

PROF. DR. KAI HORMANN, Examinator
Università della Svizzera italiana, Lugano, Switzerland

PROF. DR. LLOYD N. TREFETHEN FRS, Examinator
University of Oxford, UK

PROF. DR. RUTH KELLERHALS, President of the Jury
University of Fribourg, Switzerland

Fribourg, October 5, 2012

Thesis Supervisor

PROF. DR. JEAN-PAUL BERRUT

Dean

PROF. DR. ROLF INGOLD

To D. S.

“Ceux qui comprennent
ne comprennent pas
que l’on ne comprenne pas.
Et ceux-ci doutent
que ceux-là comprennent.”

Paul Valéry,
Mauvaises pensées et autres (1942)

Abstract

This thesis is a collection of properties and applications of linear barycentric rational interpolation, mainly with the weights presented by Floater and Hormann in 2007.

We are motivated by the counterintuitive and provable impossibility of constructing from equispaced data an approximation scheme that converges very rapidly to the approximated function and is simultaneously computationally stable. Linear barycentric rational interpolation with the weights presented by Floater and Hormann turns out to be very efficient in practice, especially when the function is sampled at equispaced nodes. We pursue the investigation of this interpolation scheme to understand the reason for its efficiency with equispaced nodes, and to what extent it is suited for applications other than the approximation of functions.

In a first part, we analyse several properties of Floater–Hormann interpolation, such as its convergence for differentiable and analytic functions, the stability of its evaluation in barycentric form and the condition of this interpolation problem. The comparison with other well established schemes which do not necessarily allow the nodes to be equispaced reveals that Floater–Hormann interpolation with equispaced nodes is very competitive, but of course not optimal as a general method of approximation. Nevertheless it is extremely easy to implement, can be evaluated quickly and gives analytic approximations. The linearity of such barycentric rational interpolation in the data makes it additionally well suited for applications, which are the subject of the second part of this thesis. We investigate the approximation of derivatives, integrals and antiderivatives with methods directly derived from the interpolation scheme; a special focus lies on methods for equispaced samples. In the last part, we present an extension to the original Floater–Hormann interpolation which is supposed to alleviate some remaining drawbacks. Most of the properties and applications that have been the subject of investigations for the original Floater–Hormann interpolants are analysed again for this extended scheme.

Résumé

Ce travail est une collection de propriétés et d'applications de l'interpolation rationnelle linéaire barycentrique, principalement de celle faisant usage des poids proposés par Floater et Hormann en 2007.

Nous avons été motivés par le fait contre-intuitif et démontrable qu'il est impossible de construire à partir de données équidistantes une méthode d'approximation convergeant très rapidement et qui soit simultanément stable pour l'évaluation numérique. L'interpolation rationnelle linéaire barycentrique avec poids de Floater et Hormann s'avère très efficiente en pratique, surtout si la fonction est échantillonnée en des points équidistants. Cette méthode d'approximation est étudiée de manière plus détaillée, pour comprendre la raison de son efficience avec les nœuds équidistants et dans quelle mesure elle se prête à des applications autres que l'approximation de fonctions.

Dans une première partie, nous analysons un certain nombre de propriétés de l'interpolation de Floater et Hormann, comme par exemple la convergence pour des fonctions dérivables ou analytiques, la stabilité de son évaluation sous forme barycentrique et la condition de ce problème d'interpolation. La comparaison avec d'autres méthodes bien connues, ne permettant pas nécessairement le choix des nœuds équidistants montre que l'interpolation de Floater et Hormann avec nœuds équidistants est très compétitive, mais certainement pas optimale en tant que méthode générale d'approximation. Elle est néanmoins extrêmement simple à implémenter, peut être évaluée rapidement numériquement et fournit des approximations analytiques. De par sa linéarité en les données, ce genre d'interpolants rationnels barycentriques se prête en plus pour les applications, qui constituent la deuxième partie de ce travail. Nous étudions l'approximation de dérivées, d'intégrales et de primitives avec des méthodes issues directement de l'interpolation; nous nous intéressons tout particulièrement aux méthodes pour points équidistants. Dans une dernière partie, nous présentons une extension de l'interpolation de Floater et Hormann, construite pour atténuer certains défauts résiduels. La plupart des propriétés et applications étudiées jusqu'ici pour l'interpolation de Floater et Hormann originelle sont analysées à nouveau pour cette extension.

Zusammenfassung

Diese Arbeit ist eine Sammlung von Eigenschaften und Anwendungen der linearen baryzentrischen rationalen Interpolation, hauptsächlich mit den von Floater und Hormann in 2007 eingeführten Gewichten.

Unsere Motivation entsprang aus der kontraintuitiven und beweisbaren Unmöglichkeit aus äquidistanten Daten eine Näherungsmethode zu entwickeln, welche sehr schnell konvergiert und gleichzeitig stabil numerisch ausgewertet werden kann. Lineare baryzentrische rationale Interpolation mit den Gewichten von Floater und Hormann ist sehr effizient in der Praxis, insbesondere für Funktionen welche lediglich an äquidistanten Stützstellen gegeben werden können. Diese Näherungsmethode wird genauer untersucht, um zu verstehen warum sie insbesondere mit äquidistanten Stützstellen so effizient ist und in wie fern sie auch für andere Anwendungen als die Annäherung von Funktionen geeignet ist.

In einem ersten Teil untersuchen wir Eigenschaften der Floater–Hormann Interpolation, wie z.B. die Konvergenz bei differenzierbaren oder analytischen Funktionen, die Stabilität der numerischen Auswertung in baryzentrischer Form und die Kondition dieses Interpolationsproblems. Der Vergleich mit anderen wohlbekannten Methoden, welche die Wahl der äquidistanten Stützstellen nicht notgedrungen erlauben, zeigt, dass Floater–Hormann Interpolation mit äquidistanten Stützstellen sehr leistungsfähig ist, aber selbstverständlich nicht optimal als allgemeine Näherungsmethode ist. Nichtsdestoweniger ist sie einfach zu programmieren, kann schnell numerisch ausgewertet werden und liefert analytische Näherungen. Die Linearität in den gegebenen Daten dieser baryzentrischen rationalen Interpolierenden erlaubt es, diese auch in Anwendungen einzusetzen, wie wir in einem zweiten Teil dieser Arbeit sehen werden. Wir untersuchen die Annäherung von Ableitungen, Integralen und Stammfunktionen mit Methoden, welche direkt aus den Interpolationsmethoden hergeleitet werden; besonderes Augenmerk liegt auf Methoden für äquidistante Stützstellen. In einem letzten Teil führen wir eine Erweiterung der originalen Floater–Hormann Interpolation ein, welche vor allem ein paar übrig bleibende Nachteile beseitigen sollte. Die meisten Eigenschaften und Anwendungen welche bereits für die originale Floater–Hormann Interpolation untersucht wurden, werden noch einmal für die Erweiterung analysiert.

Acknowledgements

First of all I would like to thank Jean-Paul Berrut. It was through an interesting seminar work under his guidance during my undergraduate studies that I became fascinated by numerical analysis, as it allows one to immediately see and feel the theory through numerical computations. It was also Jean-Paul who showed me the 2007 paper by Floater and Hormann on their approach to linear barycentric rational interpolation, which we applied for solving two-point boundary value problems in my master thesis. Encouraged by the promising results observed, we decided that I could start a PhD work under his guidance. Throughout the generation of this thesis, Jean-Paul showed much enthusiasm for my investigations and was constantly available for constructive discussions and advice. His kindness and modesty as a person and his wisdom on numerical methods and the numerical approach in general were very valuable to me throughout my doctoral studies. I am very grateful to him for having accepted to be my advisor for this thesis.

I would like to thank Richard Baltensperger and Aleš Janka for constructive, stimulating and encouraging discussions as well as numerical advice occasionally during the past few years.

It was a very instructive experience to write my first paper together with Michael Floater and Jean-Paul Berrut, which we prepared during a stay at the University of Oslo. I am very thankful to Mike for sharing with me his knowledge and his precision in writing scientific papers and also for acting as an examiner for this thesis.

I also thank Len Bos, Stefano De Marchi and Kai Hormann for letting me be part of the project of the study of the Lebesgue constants and for the very nice collaborations which followed. I especially thank Kai for the many emails exchanged, the individual discussions and his great availability and helpfulness, also beyond the project on Lebesgue constants, as well as for having agreed to act as a jury member. It was my great pleasure to be invited by Stefano to the Dolomites Research Week in 2011, where we continued our investigations of the Lebesgue constants, and also to be entrusted with the co-organisation of a session of the DWCAA12 conference.

I thank Nick Trefethen for a week in Oxford, for the unforgettable formal dinner at Balliol College, for his valuable advice in many respects, for the comments on the draft of the joint paper with Stefan Güttel, and for having agreed to act as a jury member for this thesis as well as for his many suggestions to enhance this document. I would also like to thank the whole Chebfun team for the friendly reception at the Oxford Mathematical Institute and Pedro Gonnet

for his additional advice. I especially thank Stefan very much for the fruitful collaboration, past and to come, his visits to Fribourg and for sharing his many skills with me.

Special thanks go to Joris Van Deun, whom I met at my first conference in the Dolomites. His knowledge and his great interest in numerical methods in general and in my first small achievements were the first steps into a few but very valuable conversations we had at various places, including Alba, Fribourg, Leuven and Oxford. His thoughts about life as a researcher helped and stimulated me very much in my beginnings to better understand the numerical analysis community and to feel more comfortable within it. In that same sense I am thankful to Irem Yaman and Oliver Salazar Celis for their interest in my research and for a short but intense visit to Antwerp. It was in the train back to Leuven that I had the essential idea which led to the contents of Chapter 7.

In order to include everybody, I thank the whole Department of Mathematics at the University of Fribourg for the good ambiance, the very favourable working conditions, the complete trust in my duties at the Department and for the great travel support, which was also partly provided by the Swiss National Science Foundation.

Last but not least, I thank Floriane for many encouraging words and for her encouragement, and I thank my family.

Contents

List of Notations

β	local mesh ratio, 18
$\ell_i(x)$	fundamental Lagrange functions, 10
λ_i	weights in barycentric polynomial interpolation, 11
$\lambda_i(x)$	Floater–Hormann blending functions, 16
Λ_n	Lebesgue constant, 32
$\Lambda_n(x)$	Lebesgue function, 32
$\mathcal{P}_n$	vector space of polynomials of degree less than or equal to n, 32
$\mu_i(x)$	modified Floater–Hormann blending functions, 18
μ_n	normalised node counting measure, 49
ω_k	quadrature weights, 112
$\phi(x)$	node density, 49
ε	unit roundoff or machine precision, 29
ε_p	perturbation due to rounding, noise, or measurement imprecision, 33
C	factor defining the variable blending parameter, 52
$C^k[a,b]$	set of functions which are k times differentiable on $[a,b]$, 18
D	parameter for extended rational interpolation, 141
d	Floater–Hormann blending parameter, 16
$d(n)$	variable blending parameter, 52
$D_{ij}^{(k)}$	element of the kth differentiation matrix and finite difference weight, 98
$e(x)$	interpolation error, 19
f_i	interpolated function values, 10
h	maximal distance between neighbouring nodes, mesh size, 15
h_*	minimal distance between neighbouring nodes, 35
I	definite integral, 112
I_n	index set, 17
J_i	index set, 17
K	generic constant that does not depend on n, 2

$L(x)$	nodal polynomial, 11
M	global mesh ratio, 34
n	number of nodes minus one, 10
$p_n(x)$	polynomial interpolant of degree less than or equal to n, 11
p_n^*	best approximation in $\mathcal{P}_n$, 32
$r_n(x)$	barycentric rational interpolant, 14
s	singularity of the interpolated function, 50
$s(x)$	Floater–Hormann denominator polynomial, 18
$U^\mu(z)$	logarithmic potential, 49
$u_n(z)$	discrete potential, 49
w_i	weights in rational barycentric interpolation, 14
x_i	interpolation points or nodes, 10

Chapter 1

Introduction

1.1 Motivation

We are interested in approximating smooth functions, i.e., functions with a fair number of continuous derivatives, whose values are given at a few points only, either from measurements or by evaluation. For complicated functions, it might be difficult to gather many values and it is therefore easier to work with an approximation instead of the function itself, provided the approximation is in some sense close enough to the function. This also holds true for applications other than the approximation of the function, such as differentiation, integration, rootfinding and many more. It is most natural to obtain the values of the function at equispaced measurement or evaluation points, called nodes, especially if one does not have much a priori knowledge about the function.

Generally, the interpolating polynomial, that is, the polynomial which passes through the given data values and whose degree is at most one less than the number of data points, gives approximations that look more or less satisfactory to the eye as long as the degree is small. This is the way it was used before the era of computers and it is also the way it was designed: Some values of functions evaluated at equispaced points are kept in a table and the values of these functions between the nodes are approximated through interpolation. Since every computation had to be done by hand, it seems advisable to handle only interpolating polynomials with smallest possible degree attaining the desired accuracy, which was rarely more than a few digits. Now that calculation with higher accuracy is easily feasible with the help of computers, there

is an obvious temptation of using interpolants of high degree. While this still leads to very accurate results with certain classes of nodes, the opposite is true for other classes, whose most prominent example is unfortunately equispaced nodes. Many solutions to this difficulty have been suggested in the past, and one of these is the rational interpolation scheme introduced in 2007 by Floater and Hormann [46], which we study in much detail in this work. This scheme is a blend of polynomial interpolants, each of which has the same maximal degree, usually much less than the number of interpolation nodes. Many applications of polynomial interpolation can simply be transformed into applications of these rational interpolants. Nearly everything seems to work fine in practice; the only drawback is that the theory becomes more intricate than that of polynomial interpolation as everything becomes rational. For this reason, we have tried to prove, for a selection of applications, most of the results observed in practice. However, some observations still remain conjectures and would need to be investigated in much more depth. Fortunately, we believe that those which we did not prove are not the most essential.

Let us now describe some of the above with a bit more precision; the exact details will be given in the forthcoming chapters. Polynomial interpolation with Chebyshev points is much more reliable than with equispaced nodes. The Chebyshev points most used in the theory are the *Chebyshev points of the first kind*,

$$x_i = -\cos\Big(\frac{2i+1}{n+1}\frac{\pi}{2}\Big), \quad i = 0, \dots, n,$$

which are the roots of the $(n+1)$st *Chebyshev polynomial*, defined as $T_{n+1}(x) = \cos\big((n+1)\arccos(x)\big)$, and the *Chebyshev points of the second kind*,

$$x_i = -\cos\Big(\frac{i}{n}\pi\Big), \quad i = 0, \dots, n,$$

which are the extrema of T_n augmented by the boundary points. For a $k-1$ times differentiable function f with a kth derivative of bounded variation, the polynomial interpolant p_n with Chebyshev points in $[-1,1]$ converges at the *algebraic* rate $O(n^{-k})$ as n increases, that is, the error satisfies

$$\max_{x\in[-1,1]} |f(x) - p_n(x)| \leq Kn^{-k},$$

where K is a constant. For a function analytic (complex differentiable) and bounded in some well defined ellipse, the convergence is *exponential* or *geometric* at the rate $O(\rho^{-n})$, where ρ is a parameter greater than 1 depending on the ellipse.

All this is not necessarily true if the nodes are equispaced. Polynomial interpolation with equispaced nodes may in some cases converge in theory but will in general diverge in practice because of the amplification of rounding errors. Even worse, for some classes of functions, such interpolation does not even converge in theory. With equispaced nodes, not only does polynomial interpolation not perform reliably: It is simply not possible to construct an approximation method with equispaced nodes which *simultaneously* converges exponentially fast and is stable, i.e., does not suffer from an excessive amplification of small errors in the data. This statement has been proven by Platte, Trefethen and Kuijlaars in [85], where the authors also present an extended list of currently available methods for approximation with equispaced nodes; see also [67].

As stable approximation and exponential convergence is not possible with equispaced nodes, a natural and fascinating question arises: *How close to exponential can the convergence of a stable method for equispaced nodes come?* This method should, in addition, be suitable for applications.

Theoretical exponential convergence is not always needed in practice. Once the relative approximation error has reached a value close to machine precision, typically about 10^{-16}, one can not tell the difference between a method whose theoretical speed of convergence is the same for all n and whose error does not decrease any more in practice because it has attained machine precision, and a method that has variable theoretical speed of convergence, fast for small n and, once the desired accuracy is attained, slower afterwards, so that the error is merely maintained at the same size. Floater–Hormann interpolation is such a method, provided the blending parameter is chosen adequately, as we shall see in Chapter 4.

Many methods for interpolation with equispaced nodes have been presented; see, e.g., [85] for an overview. We shall concentrate on linear barycentric rational interpolants, which are analytic in the nodes' hull and well suited for applications because of their linearity in the data.

The results presented in this work are a collection of properties of Floater–Hormann interpolation as well as applications of this approximation scheme. The applications are explained and proven essentially for Floater–Hormann interpolation, but they could be established analogously with any linear barycentric interpolation scheme, since each such scheme is completely defined by its barycentric weights, as we shall see in Chapter 2; in most cases only the convergence proofs need to be adjusted for any new scheme. This is a favourable situation in view of the plausible event that some time in the future novel choices for barycentric weights will be presented.

1.2 Overview

We now come to a short overview of the chapters of the present work, for which the draft version of Trefethen's textbook *Approximation Theory and Approximation Practice* [114] was a valuable general reference.

We begin with an introduction to barycentric interpolation and review some of its properties in Chapter 2. First, we explain the general numerical problem we are concerned with throughout this work and review polynomial interpolation as well as the steps from its Lagrangian representation to the two barycentric forms. After comparing these forms, and after some historical notes, we show how to pass from polynomial interpolation in barycentric form to linear barycentric rational interpolation, in particular Berrut's interpolants, and give some known facts. This will very naturally lead to rational Floater–Hormann interpolation, which is blended polynomial interpolation. Its construction is presented as well as some useful properties for subsequent chapters. The location of the poles of rational interpolation is very important; we therefore show how they can be approximated and how they are distributed in the complex plane. The chapter ends with two numerical examples of Floater–Hormann interpolation.

After this introduction to the interpolation scheme we are mainly concerned with, we analyse two important properties in Chapter 3, namely the numerical stability of Floater–Hormann interpolation in barycentric form and the condition of this interpolation problem. We review the essential definitions of stability as well as results about the stability of polynomial interpolation in both barycentric forms, before presenting the analogues for linear barycentric rational interpolation. Parallels between the stability of polynomial and rational barycentric interpolation are drawn. For a rigorous study of the stability, the Lebesgue functions and constants, which at the same time are the condition numbers, need to be treated. After a short introduction and overview of well established results for Lebesgue constants associated with polynomial interpolation, we present lower and upper bounds on the Lebesgue constants associated with Floater–Hormann interpolation with equispaced and quasi-equispaced nodes. This study is the result of two collaborations with Len Bos, Stefano De Marchi and Kai Hormann, who had already performed such an investigation for Berrut's interpolant. A new approach to the problem enables us to derive the desired bounds in a succinct way. These results reveal why Floater–Hormann interpolation with equispaced and quasi-equispaced nodes is so much more stable and well-conditioned than polynomial interpolation with these nodes, which is catastrophically unstable even with a moderate number of such nodes, a problem that lowered unjustifiably the view of many researchers of polynomial

interpolation in general and of equispaced nodes.

Depending on the distribution of the nodes, and in particular with equispaced nodes, the polynomial interpolant may diverge in exact arithmetic for analytic functions if they have singularities too close to the interval of interpolation; this behaviour is known as Runge's phenomenon. As Floater–Hormann interpolation is blended polynomial interpolation, it is natural to investigate under which circumstances these interpolants also suffer from Runge's phenomenon. This is the subject of Chapter 4. The convergence and divergence of polynomial interpolation of analytic functions is best described by potential theory. The essential elements of this theory are reviewed, leading to the theorem on convergence and divergence, which gives the speed of convergence or divergence depending on the location of the singularities of the interpolated function closest to the interval with respect to well defined level curves. The latter depend on the distribution of the nodes and allow one to immediately determine whether Runge's phenomenon occurs or not. After the exposition of polynomial interpolation of analytic functions, we generalise this theory to blended polynomial interpolation, i.e., to Floater–Hormann interpolation. This generalisation is a result of a collaboration with Stefan Güttel. We also obtain level curves describing the convergence and divergence of linear barycentric rational interpolation of analytic functions depending on the location of the singularity of the function closest to the interval with respect to these new level lines. We analyse under what conditions Runge's phenomenon can occur in Floater–Hormann interpolation and explain why this phenomenon is less likely to show up than in polynomial interpolation. For symmetric and equispaced nodes we give more details. From the knowledge of the convergence and divergence behaviour, and that of the Lebesgue constants associated with Floater–Hormann interpolation, we are able to give a recommendation for how to choose the blending parameter d in a nearly optimal way, namely by balancing the fast convergence and the growing condition number. This strategy is demonstrated in extensive numerical tests. The chapter closes with a discussion of a good choice of interpolation nodes for Floater–Hormann interpolation, and we give some heuristic clues as to why we think that equispaced nodes are close to optimal for these interpolants.

With the knowledge of the properties of Floater–Hormann interpolation acquired in Chapters 2-4, we apply linear barycentric rational interpolation to the approximation of derivatives in Chapter 5. The kth order derivative of a function is approximated by the same derivative of its rational interpolant; the approximation order is found to decrease roughly by one for each differentiation. We first study the approximation error at the nodes for the first and second derivatives with arbitrary nodes, and higher order derivatives with equispaced

and quasi-equispaced nodes. Thereafter we analyse the convergence behaviour of the approximation of the first and second derivatives at intermediate points in between the nodes with arbitrary nodes. For higher order derivatives with equispaced and quasi-equispaced nodes, we present a cheaper alternative for computing approximations of such derivatives with almost the same approximation order as at the nodes. The theory for the approximation of the first and second derivatives with arbitrary nodes was established in a collaboration with Michael Floater; the combination of individual ideas and approaches enabled us to construct the convergence theory together. With the knowledge of this theory, we built the rational analogues of finite difference formulas, which are originally based on polynomial interpolation and allow one to construct approximations of derivatives from the function values. We compare our rational finite difference formulas with the classical polynomial ones and conclude, among other things, that the rational formulas are especially effective for the approximation of derivatives near the ends of the interval. These observations are illustrated with numerical examples.

Linear barycentric rational interpolants can also be used for the approximation of integrals and antiderivatives, as we discuss in Chapter 6. From the linearity of these interpolants, the construction of quadrature rules is obvious. We begin with an overview of polynomial interpolatory quadrature rules and state some well known relevant results about quadrature rules obtained from polynomial interpolants. Then we explain how to construct quadrature rules and approximations of antiderivatives from arbitrary linear barycentric rational interpolants. For Floater–Hormann interpolants, we give three methods; the first is based upon Chebfun. The second is a family of quadrature rules; and the third is an alternative, based upon the solution of differential equations. For the rational quadrature rules with equispaced nodes, we show convergence, namely that the approximation is one order higher than that of the interpolant itself. We demonstrate and compare the three methods with numerical examples. Thereafter, and as an additional application of the theory from Chapter 4, we derive rational quadrature rules and approximations of antiderivatives for analytic functions, which are also illustrated with examples.

In Chapter 7 we present a construction that extends Floater–Hormann interpolation in the case of equispaced nodes. The exponential growth of the Lebesgue constants associated with Floater–Hormann interpolants as the blending parameter d increases prohibits the use of large values of d, which, in exact arithmetic, would lead to faster convergence. This growth of the Lebesgue constants comes from the Lebesgue functions, which, however, yield only a few large oscillations close to the ends of the interval. Considering further nodes and con-

structing adequate function values from the given ones alleviates this drawback and leads to what we call extended Floater–Hormann interpolation. After presenting the details of this construction, we prove the convergence thereof and give bounds on the Lebesgue constants, which behave very favourably, increasing only slowly with d and the number of nodes. Extended Floater–Hormann interpolation turns out to be very well-conditioned, as we also demonstrate with examples. As this construction is also linear in the data, it is suited for the applications presented in Chapters 5 and 6 for the original family of interpolants. We prove similar results for the extended family of interpolants and illustrate them with examples, showing that the approximation errors are often smaller than with the original family. We give additional heuristic arguments as why these interpolants work well with equispaced nodes and explain why a good choice of d becomes less important.

We close this work with a final chapter presenting conclusions and an outlook.

Chapter 2

Barycentric Interpolation

This chapter gives an overview of the interpolation techniques we shall be concerned with throughout this thesis. We begin with polynomial interpolation, its barycentric forms and some historical notes in Section 2.1. In the following section we explain the link between polynomial barycentric interpolation and its rational counterpart, and present some early constructions by Berrut in linear barycentric rational interpolation. These were further generalised by Floater and Hormann, who proposed a whole family of linear rational interpolants which we describe in Section 2.3. In this work we study the applications of this family of interpolants and related ones. Some elementary properties are already established in this chapter, and others follow in subsequent parts, complemented with several numerical results.

2.1 Polynomial Interpolation

The basic numerical approximation problem we are addressing is the following: We suppose we are given $n+1$ distinct ordered points, called *nodes*, in a closed interval $[a, b]$ of the real line, $a = x_0 < x_1 < \ldots < x_n = b$, and corresponding values $f_0, \ldots, f_n$, which may or may not stem from a real or complex function evaluated at the nodes. The aim is to find a function g from a finite-dimensional linear subspace of $(C[a, b], \|\cdot\|)$, the Banach space of all continuous functions on the interval $[a, b]$ normed with the maximum norm on that interval, i.e., $\|\cdot\| = \max_{a\le x\le b} |\cdot|$, such that g interpolates the data at the nodes, i.e., g satisfies the *interpolation property*

$$g(x_i) = f_i, \qquad i = 0, \ldots, n.$$

There are two major classes of interpolants: The one we are interested in contains interpolants that are linear in the data; this means that g may be represented as a linear combination

$$g(x) = \sum_{i=0}^{n} b_i(x) f_i, \tag{2.1}$$

of some basis functions b_i which do not depend on f_i. The methods from the second class may not be written in this form; examples include B-splines, least-squares fits, Padé approximants, best approximants and classical rational interpolants. We are mainly concerned with the former class since its properties turn out to be valuable in applications such as the approximation of derivatives, integrals and solutions of differential equations. We will give further details in the forthcoming chapters.

The most classical example of linear interpolation between the given set of $n+1$ values is the unique polynomial interpolant p_n of degree less than or equal to n, which interpolates the data and can be written in Lagrange form,

$$p_n(x) = \sum_{i=0}^{n} \ell_i(x) f_i,$$

where

$$\ell_i(x) = \prod_{\substack{j=0 \\ j\neq i}}^{n} \frac{x - x_j}{x_i - x_j} \tag{2.2}$$

are the *fundamental Lagrange functions*, with the *Lagrange property*

$$\ell_i(x_j) = \begin{cases} 1, & j = i, \\ 0, & j \neq i. \end{cases} \tag{2.3}$$

The evaluation of p_n written in this form requires $O(n^2)$ operations for every point x. The above formula for the polynomial interpolant dates to the end of the 18th century, and was documented by Lagrange [79] in 1795 but also earlier by Waring [125] in 1779. Euler [43] had derived in 1783 a formula close to Newton's formula and leading to Lagrange's formula. Many other closely related representations of polynomial interpolation exist, such as those of Newton, Gauss, Stirling, Bessel and Everett; see [104] for details. We refer to [80] for an extensive review of the history of interpolation starting from the ancient times and for numerous references.

Throughout this document, we stick to the *barycentric interpolation form*, which presents a number of advantages for numerical computation, as we shall see below; see also [17]. The barycentric form of p_n may be obtained from its Lagrange form. First we define the *nodal polynomial*

$$L(x) = \prod_{i=0}^{n} (x - x_i) \tag{2.4}$$

and the *barycentric weights*, or simply *weights*,

$$\lambda_i = 1 \Big/ \prod_{\substack{j=0 \\ j \neq i}}^{n} (x_i - x_j). \tag{2.5}$$

A simple computation shows that $\lambda_i = 1/L'(x_i)$ [60]. The fundamental Lagrange functions from (2.2) may now be written as

$$\ell_i(x) = L(x) \frac{\lambda_i}{x - x_i},$$

so that p_n is also represented by

$$p_n(x) = L(x) \sum_{i=0}^{n} \frac{\lambda_i}{x - x_i} f_i, \tag{2.6}$$

which is often called the "modified Lagrange formula" (Higham [63]) or the "first form of the barycentric interpolation formula" (Rutishauser [93]). Equation (2.6) was already presented in 1825 by Jacobi in [70], where he computes

the partial fraction decomposition of p_n/L and expresses it as the sum in the right-hand side of (2.6). Jacobi stresses the immediate relation to Lagrange interpolation.

Once the barycentric weights are available, which requires $O(n^2)$ operations if they are computed from (2.5), the polynomial may be evaluated in only $O(n)$ operations for every x. This is the same complexity as with Newton's form once the divided differences of the values $f_0, \ldots, f_n$ for that formula are computed, which can also be done in $O(n^2)$ operations. Winrich [129] compares the number of operations needed for the evaluation of p_n with the Lagrange form, the barycentric form and Aitken's and Neville's algorithms; see [60]. He concludes that the barycentric formula should be preferred if the number of nodes is greater than or equal to 5. The barycentric weights only need to be computed once for every set of nodes, since they are independent of the point x where p_n is evaluated. Once the weights are computed, as they do not depend on f, the polynomial interpolant of any function can be evaluated in $O(n)$ operations. Closed analytic expressions for the weights are available for equispaced nodes and Chebyshev points [17], as well as Gauss–Legendre, Gauss–Jacobi and many more; see [123, 124]. The polynomial interpolant between such nodes can thus be completely evaluated in only $O(n)$ operations. For equispaced nodes, $x_i = a + i(b-a)/n$, the weights are, see e.g. [99],

$$\lambda_i = (-1)^{n-i} n^n \binom{n}{i} \Big/ \big((b-a)^n n!\big).$$

Higham showed in [63] that (2.6) is unconditionally stable. There is also a "second (true, 'eigentliche') form of the barycentric formula" (Rutishauser [93]), which is often simply called *the barycentric formula*, and which can be evaluated in $[a, b]$ as stably as the first, provided the nodes are distributed in a suitable way. We give further details on this and on the numerical stability in Chapter 3. The second barycentric formula follows from the simple observation that p_n is the unique polynomial of degree less than or equal to n interpolating the values. If all of these are equal to 1, which corresponds to sampling the degree zero polynomial 1, then, because of uniqueness, its polynomial interpolant simplifies to that constant polynomial, so that

$$1 = L(x) \sum_{i=0}^{n} \frac{\lambda_i}{x - x_i}.$$

This equation is also presented by Jacobi in [70] as the partial fraction decomposition of $1/L$. We may divide p_n written in the first barycentric form (2.6)

by the above complicated representation of 1 to obtain

$$p_n(x) = \sum_{i=0}^{n} \frac{\lambda_i}{x - x_i} f_i \Big/ \sum_{i=0}^{n} \frac{\lambda_i}{x - x_i}, \tag{2.7}$$

where the nodal polynomial has been cancelled and thus does not need to be computed any more. From the above derivation, it becomes clear that (2.7), which looks like a rational function, is in fact a polynomial and interpolates between the given values. An additional advantage of the second barycentric form is that the weights appear in the numerator and denominator, so that common factors cancel, leading to simpler expressions for the λ_i. For equispaced nodes, the simplified weights are simply

$$\lambda_i = (-1)^i \binom{n}{i};$$

for Chebyshev points of the second kind, they oscillate in sign and have the magnitude 1, except for the first and last, which are equal to $\pm 1/2$; see [96]. For explicit expressions for the weights for some other distributions of nodes, we refer to [17, 123, 124] and the references therein.

This property that the weights for the second barycentric form can be simplified is not a purely aesthetic one. It shows that the weights are scale-invariant, whereas those for the first barycentric form are not and may lead to overflow or underflow depending on n and also on the length of the interval; see (2.5).

Accommodating additional data in the polynomial interpolant can be done in $O(n)$ operations per supplementary node, if p_n is written in a barycentric form, since the weights (2.5) can be updated easily.

The barycentric formula was already known at least in the 1940s; it is described by Taylor [107] in 1945 for equispaced nodes and by Dupuy [38] in 1948 mainly for equispaced nodes, but he also mentions nonequispaced nodes and tensor product interpolation on an equispaced grid. These two papers were apparently the first to publish the idea of dividing the polynomial interpolant by a complicated representation of 1. The barycentric formula for interpolation with arbitrary and equispaced nodes is mentioned by Bulirsch and Rutishauser [24] in 1968. The formula was later documented by Salzer [96] in 1972, mainly for Chebyshev points of the second kind, but also for Chebyshev points of the first kind; differentiation with such nodes is also explained. Henrici derives the formula in his textbook [61] in 1982. Thereafter, the barycentric formula is also used for rational interpolation and studied essentially by Berrut, Schneider and Werner. However, it was not until 2004 with the Berrut–Trefethen SIAM

Review paper [17] that the barycentric formula became a widely appreciated technique for polynomial interpolation. We refer to Section 10 of that same reference for a more detailed historical review of the barycentric form of the polynomial interpolant and for reasons why it took so long for its benefits to become known to the numerical analysis community; see also [113].

2.2 Linear Barycentric Rational Interpolation

The second barycentric formula allows a direct transition from polynomial interpolation to barycentric rational interpolation. If the weights λ_i are changed to other nonzero weights w_i, then $L(x)\sum_{i=0}^{n} w_i/(x-x_i)$ is not necessarily equal to 1 any longer and the correspondingly modified expression (2.7) becomes a true rational function. The following lemma extends this observation. The present form comes from [13]; the first statement was already presented and proven by Werner in [128] and brought to his attention in a private communication from Schneider. The second statement is proven in [15].

Lemma 2.1. *(i) Let $\{(x_i, w_i, f_i)\}_{i=0,\ldots,n}$ be a set of $n+1$ real or complex triplets with $x_j \neq x_k$ for $j \neq k$. Then if all $w_i \neq 0$, the rational function*

$$r_n(x) = \sum_{i=0}^{n} \frac{w_i}{x-x_i} f_i \Big/ \sum_{i=0}^{n} \frac{w_i}{x-x_i}, \tag{2.8}$$

interpolates f_i at x_i: $\lim_{x\to x_i} r_n(x) = f_i$.

(ii) Conversely, every rational interpolant whose numerator and denominator are polynomials of degree not exceeding n may be written in the barycentric form (2.8) for some weights w_i.

The first statement of this lemma is a simple calculation and reveals an additional advantage of barycentric interpolation: The interpolation property $r_n(x_i) = f_i$ is always guaranteed, even if the weights w_i are computed with errors, as long as they remain nonzero. For the second statement of the lemma, it is enough to write the polynomials p and q in the numerator and denominator of the rational interpolant r in Lagrange form and to observe that $p = rq$ and therefore define the weights by $w_i = \lambda_i q(x_i)$, with λ_i from (2.5). This relation between the barycentric weights and the values of the denominator polynomial can be exploited to derive further characteristics of barycentric rational interpolation. Indeed, as pointed out in [16], the relation between the weights and

the denominator shows that a rational interpolant represented as a quotient of polynomials is completely determined by its denominator.

The second statement of Lemma 2.1 is very important for detecting properties of a given interpolant in classical rational interpolation. Indeed, once the numerator and denominator are found and put into the reduced form, i.e., common factors are simplified, the barycentric form gives information on possibly unattainable points and subintervals where poles might occur; see Corollary 7 and Proposition 8 of [97] for details.

If the locations of some of the poles of the function are known, it is possible to attach them to the barycentric rational interpolant. For techniques of how to achieve this see, e.g., [3, 10].

It becomes obvious that barycentric rational interpolation allows one to choose the nodes *and* the weights. The rational interpolant r_n is *linear* if the barycentric weights do not depend on the values $f_0, \ldots, f_n$. The additional freedom coming from the choice of the weights, as compared to polynomial interpolation where the weights are fixed, is advantageous if interpolation is needed in sets of nodes that may not be chosen at will, especially if the nodes need to be equispaced. As mentioned earlier, polynomial interpolation with equispaced nodes is not reliable. In 1988, Berrut proved [9] that the barycentric interpolant (2.8) with weights $w_i = (-1)^i$ has no real poles for every distribution of the nodes. Moreover he conjectured linear convergence as $h \to 0$ at the rate $O(h)$, which means that $\|f - r_n\| \leq Kh$, with

$$h = \max_{0 \leq i \leq n-1} (x_{i+1} - x_i) \tag{2.9}$$

and K a constant independent of n. For equispaced nodes, he introduced and studied numerically the same choice of weights with the first and last weight divided by 2, as in polynomial interpolation with Chebyshev points. Faster convergence is observed than with the former choice of weights, and $O(h^2)$ convergence was conjectured in [5].

2.3 Floater–Hormann Interpolation

2.3.1 The Construction

In 2007, Floater and Hormann [46] presented a whole family of linear barycentric rational interpolants and provided explicit formulas for the barycentric weights. Their result was added to the Numerical Recipes book [88] in the very same year. The construction includes the weights of Berrut presented at the end of

the preceding section as special cases; the second for equispaced nodes only. To establish the formulas, one chooses an integer parameter $d \in \{0, 1, \ldots, n\}$ and denotes by p_i, $i = 0, 1, \ldots, n-d$, the unique polynomial of degree less than or equal to d interpolating the subset of $d+1$ values $f_i, f_{i+1}, \ldots, f_{i+d}$. The rational function

$$r_n(x) = \frac{\sum_{i=0}^{n-d} \lambda_i(x) p_i(x)}{\sum_{i=0}^{n-d} \lambda_i(x)}, \tag{2.10}$$

with

$$\lambda_i(x) = \frac{(-1)^i}{(x - x_i) \cdots (x - x_{i+d})}, \tag{2.11}$$

interpolates the data and can be interpreted as a blend of the polynomial interpolants $p_0, \ldots, p_{n-d}$ with the blending functions $\lambda_0(x), \ldots, \lambda_{n-d}(x)$. The choice $d = n$ recovers polynomial interpolation. We shall call the rational interpolants from (2.10) *Floater–Hormann interpolants* as a subclass of linear barycentric interpolants. The parameter d is sometimes called the *blending parameter*. As is clear from the notation, we suppose from now on that n and d, $d \leq n$, are always nonnegative integers. As the authors state in [46], the family of interpolants (2.10) was independently found by Floater and Hormann while they were working on interpolation on nested planar curves. The underlying idea, as presented by Floater in his talk at the DWCAA09 conference, is to observe that Berrut's interpolant, which was conjectured to converge linearly, is the solution of the following problem involving *first* order divided differences,

$$\sum_{i=0}^{n} (-1)^i g[x_i, x] = 0.$$

The solution g to such a problem involving the $n - d + 1$ divided differences $g[x_i, \ldots, x_{i+d}, x]$ then is precisely r_n from (2.10), and it turns out that this interpolant converges at the rate $O(h^{d+1})$ to f, as we will see in Theorem 2.3.

2.3.2 Properties

The barycentric form of r_n is obtained by representing the blended polynomials in Lagrange form and rearranging the sums [46, Sect. 4], leading to the weights

$$w_i = (-1)^{i-d} \sum_{k \in J_i} \prod_{\substack{j=k \\ j \neq i}}^{k+d} \frac{1}{|x_i - x_j|}, \tag{2.12}$$

with

$$J_i = \{k \in I_n : i - d \le k \le i\}, \quad \text{where} \quad I_n = \{0, \ldots, n - d\}. \tag{2.13}$$

We call the basis functions

$$\frac{w_i}{x - x_i} \Big/ \sum_{j=0}^{n} \frac{w_j}{x - x_i}, \quad i = 0, \ldots, n, \tag{2.14}$$

the *Lagrange fundamental rational functions*. The derivation in [46] of the above explicit formulas for the barycentric weights reveals an interesting relationship between the denominator in the original representation (2.10) of r_n and the denominator in the barycentric representation, namely,

$$\sum_{i=0}^{n-d} \lambda_i(x) = \sum_{i=0}^{n} \frac{w_i}{x - x_i}. \tag{2.15}$$

This will be very useful in investigations of the Lebesgue functions in Chapter 3.

As the weights are all nonzero, the interpolation property of r_n is verified by Lemma 2.1. For *equispaced* nodes, the *simplified* weights oscillate in sign, and their absolute values for $0 \le d \le 3$ are

$$\begin{aligned} &1, 1, \ldots, 1, 1, && d = 0, \\ &1, 2, 2, \ldots, 2, 2, 1, && d = 1, \\ &1, 3, 4, 4, \ldots, 4, 4, 3, 1, && d = 2, \\ &1, 4, 7, 8, 8, \ldots, 8, 8, 7, 4, 1, && d = 3. \end{aligned}$$

The factor by which the original weights have been multiplied to find the simplified integer weights is $d!h^d$. The above values for the weights can be used directly for barycentric interpolation, so that in this case, as for polynomial interpolation with Chebyshev points, the weights do not necessarily need to be computed. Floater–Hormann interpolation with equispaced nodes and small d is thus also a method for interpolation in only $O(n)$ operations per point of evaluation.

With $d = 0$, the weights are all the same in absolute value for any distribution of the nodes, see (2.12), so that this case is exactly the one already presented by Berrut [9] in 1988. Dividing by 2 the simplified weights for $d = 1$ and equispaced nodes yields the values suggested by Berrut in [9]. For $n \ge 2d$ the

absolute values of the simplified weights are given as [19]

$$|w_i| = \begin{cases} \sum_{k=0}^{i} \binom{d}{k}, & i \leq d, \\ 2^d, & d \leq i \leq n-d, \\ |w_{n-i}|, & i \geq n-d. \end{cases} \tag{2.16}$$

If $n < 2d$ then the above formulas remain correct provided the weights are computed symmetrically, beginning with w_0 and w_n.

We now state two more results from [46] characterising the interpolants r_n.

Theorem 2.2. *The rational interpolant r_n in* (2.10) *has no real poles.*

This theorem is proven in [46] by multiplying the numerator and denominator of r_n by $(-1)^{n-d}L(x)$, where L is the nodal polynomial from (2.4), and showing that the resulting denominator polynomial

$$s(x) = \sum_{i=0}^{n-d} \mu_i(x), \quad \text{with} \quad \mu_i(x) = (-1)^{n-d} L(x) \lambda_i(x), \tag{2.17}$$

is positive for every real x. The rational function r_n is thus infinitely smooth and even analytic in a certain neighbourhood of $[a, b]$; see also Section 2.3.3.

The next result we state from [46] gives the speed of convergence via a bound on the interpolation error. We begin with the definition of a local mesh ratio, which is only needed in the case $d = 0$,

$$\beta = \max_{1 \leq i \leq n-2} \min\Big\{ \frac{x_{i+1} - x_i}{x_i - x_{i-1}}, \frac{x_{i+1} - x_i}{x_{i+2} - x_{i+1}} \Big\}. \tag{2.18}$$

Theorem 2.3. *Suppose that $f \in C^{d+2}[a, b]$. Then,*

$$\|f - r_n\| \leq \begin{cases} h(1+\beta)(b-a)\frac{\|f''\|}{2}, & d = 0,\ n \text{ odd}, \\ h(1+\beta)\big((b-a)\frac{\|f''\|}{2} + \|f'\|\big), & d = 0,\ n \text{ even}, \\ h^{d+1}(b-a)\frac{\|f^{(d+2)}\|}{d+2}, & d \geq 1,\ n-d \text{ odd}, \\ h^{d+1}\big((b-a)\frac{\|f^{(d+2)}\|}{d+2} + \frac{\|f^{(d+1)}\|}{d+1}\big), & d \geq 1,\ n-d \text{ even}. \end{cases}$$

In exact arithmetic and for fixed d, the rational interpolant thus converges algebraically to f for every $d \geq 1$ and under a bounded mesh ratio condition for $d = 0$. The only hypothesis made on f is that it should be sufficiently many

times differentiable. Numerical tests have revealed that the regularity hypothesis $f \in C^{d+2}[a,b]$ for observing the convergence order $d+1$ may sometimes be weakened; Floater and Hormann demonstrated that the C^0 function $f(x) = |x|$ can be approximated with order 1. Other tests revealed that $f \in C^d[a,b]$ is sometimes enough to observe the rate $d+1$; examples include $f(x) = |x|^{d+1}$ with d even. This, however, does not seem to be true for arbitrary C^d functions.

Theorem 2.3 shows that different error bounds apply depending on the parity of $n-d$. The constant in these bounds is larger if $n-d$ is even than if this difference is odd. The behaviour of the error observed in practice often corresponds to this theoretical result and thus leads to oscillatory curves when the error is represented with varying n and fixed d; such an error behaviour can also be caused by functions that are easier to resolve either if n is even or odd, e.g., functions with a peak and symmetric with respect to the midpoint of the interval.

More can be said on the convergence of Floater–Hormann interpolants if f is analytic in a neighbourhood of $[a,b]$ and if the parameter d is allowed to change with n; see Section 4.2. One immediate advantage of these interpolants is the fact that the interpolation error depends on the maximum norm of just the $(d+2)$nd derivative of f, as opposed to the dependence on the $(n+1)$st derivative in the polynomial case.

For the proof of Theorem 2.3, the authors of [46] start from the original representation (2.10) of r_n and use the Newton error formula to investigate the numerator of the interpolation error e at a point $x \in [a,b]$ that is not a node,

$$e(x) = f(x) - r_n(x) = \frac{\sum_{i=0}^{n-d} (-1)^i f[x_i, \ldots, x_{i+d}, x]}{\sum_{i=0}^{n-d} \lambda_i(x)}, \tag{2.19}$$

where $f[x_i, \ldots, x_{i+d}, x]$ is the $(d+1)$st order divided difference of f corresponding to $x_i, \ldots, x_{i+d}$ and x. The factors involving the $(d+1)$st and $(d+2)$nd derivatives of f in the error bounds in Theorem 2.3 come from bounds on the numerator in the right-hand side of (2.19). For $d \geq 1$, the upper bound from [46] on the denominator of (2.19), which is also the denominator of (2.10), yields the factor h^{d+1} giving the convergence speed:

$$\left| \sum_{i=0}^{n-d} \lambda_i(x) \right| \geq \frac{1}{d! h^{d+1}}. \tag{2.20}$$

The steps leading to the above bound include an interesting additional result if

$d \geq 1$ and for $x_k < x < x_{k+1}$, $k = 0, \ldots, n-1$, namely that

$$\left| \sum_{i=0}^{n-d} \lambda_i(x) \right| \geq |\lambda_j(x)|, \tag{2.21}$$

for every $j \in J_k \setminus \{k-d\}$.

From the upper bounds on the interpolation error in Theorem 2.3 involving the norm of the $(d+1)$st and $(d+2)$nd derivatives of the interpolated function, the following corollary on the reproduction of polynomials is immediate.

Corollary 2.4. *The rational interpolant r_n in* (2.10) *reproduces polynomials of degree $\leq d$ for any n and of degree $\leq d+1$ if $n-d$ is odd.*

One major question arises when Floater–Hormann interpolation is to be used in practice: What is a good choice for the parameter d? If the nodes are Chebyshev points, the choice $d = n$ is of course close to optimal since the rational interpolant simplifies to the polynomial in that case and polynomial interpolation with Chebyshev points is near-best among polynomial interpolants with degree less than or equal to n; see also Chapter 4. The authors of *Numerical Recipes* [88] suggest that one should "start with small values of d before trying larger values", but also "we might actually encourage experimentation with high order (say, > 6)". The recommendation from ALGLIB (`http://www.alglib.net`) is a bit more precise; however, it is only valid for the case of equispaced nodes and the like. They suggest that in most cases, d should not be taken larger than 20 and the optimal choice is often between 3 and 8. Furthermore if no additional information about f other than its values at the nodes is available, then $d = 3$ is a good default choice. In later discussions and especially in Chapter 4 we will give a recommendation for how to choose d when a bit more is known about the function.

2.3.3 Approximate Location of the Poles[1]

To the best of our knowledge, the precise location of the complex poles of r_n has not yet been determined analytically. In the case $d = 0$, which corresponds to Berrut's interpolant, pole-free regions in the complex plane have been derived by Jones and Welfert in [72] by investigating the roots of the function on the right-hand side of (2.15), which is the denominator of r_n when written in barycentric

[1] This is partly joint work with Piers Lawrence and Stefan Güttel, and originated from an observation by Nick Hale.

form. However, the authors did not mention the relation to barycentric rational interpolation.

The denominator polynomial $\sum_{i=0}^{n-d} \mu_i(x)$ is a real valued function and has no real roots, see Theorem 2.2, so the number of poles is always even. From its definition, it is easy to see that the denominator polynomial has degree less than or equal to $n-d$. The sum in the denominator has $n-d+1$ terms, and every term is a polynomial of degree $n-d$ with leading coefficient 1 and oscillating sign, so that the degree is $n-d-1$ if $n-d$ is odd and otherwise it is $n-d$.

To determine the location of the poles experimentally, we need to find the roots of the denominator $\sum_{i=0}^{n} \frac{w_i}{x-x_i}$. There exists a very elegant way to compute these roots via generalised eigenvalue problems. The method we use stems from [29], where the roots of polynomials in Lagrange and Hermite interpolation bases are studied. Before treating the poles, we show the theoretical approach for determining the real roots of the interpolant r_n. Since the denominator of r_n is nonzero, the real roots of the interpolant are the same as those of the numerator polynomial $N_n(x) := L(x)\sum_{i=0}^{n} \frac{w_i}{x-x_i} f_i$; the weights may be in any form, original or simplified, since common factors can be simplified also here. Let us define the $(n+2)\times(n+2)$ matrices

$$A := \begin{pmatrix} x_0 & & & f_0 \\ & \ddots & & \vdots \\ & & x_n & f_n \\ w_0 & \dots & w_n & 0 \end{pmatrix}, \qquad B := \begin{pmatrix} E & 0 \\ 0 & 0 \end{pmatrix}, \tag{2.22}$$

where elements not displayed are zero and E is the $(n+1)\times(n+1)$ identity matrix. These matrices form the so-called companion matrix pencil. Then any root r of N_n is also an eigenvalue of the generalised eigenvalue problem $Av = rBv$, with the eigenvector $v = (\frac{f_0}{r-x_0}, \dots, \frac{f_n}{r-x_n}, 1)^T$. This procedure will give $n+2$ eigenvalues which are potentially infinite or spurious.

Companion matrix pencils are not new. They are well known for polynomials given in the canonical basis [65] and there are explicit forms, called colleague matrices [55], for polynomials in the Chebyshev basis and for other bases of orthogonal polynomials, where they are called comrade matrices [6, 7, 101, 102].

This method for rootfinding can immediately be used to locate the poles of barycentric rational interpolants: It is sufficient to replace all the f_i in (2.22) by 1 to form the companion matrix pencil for the denominator. As the degree of the denominator polynomial is known in advance, the exact number of poles is known. For this reason, we can deflate the matrix pencil sufficiently often, i.e., $d+2$ or $d+3$ times, to eliminate infinite roots; see [29] for details of this

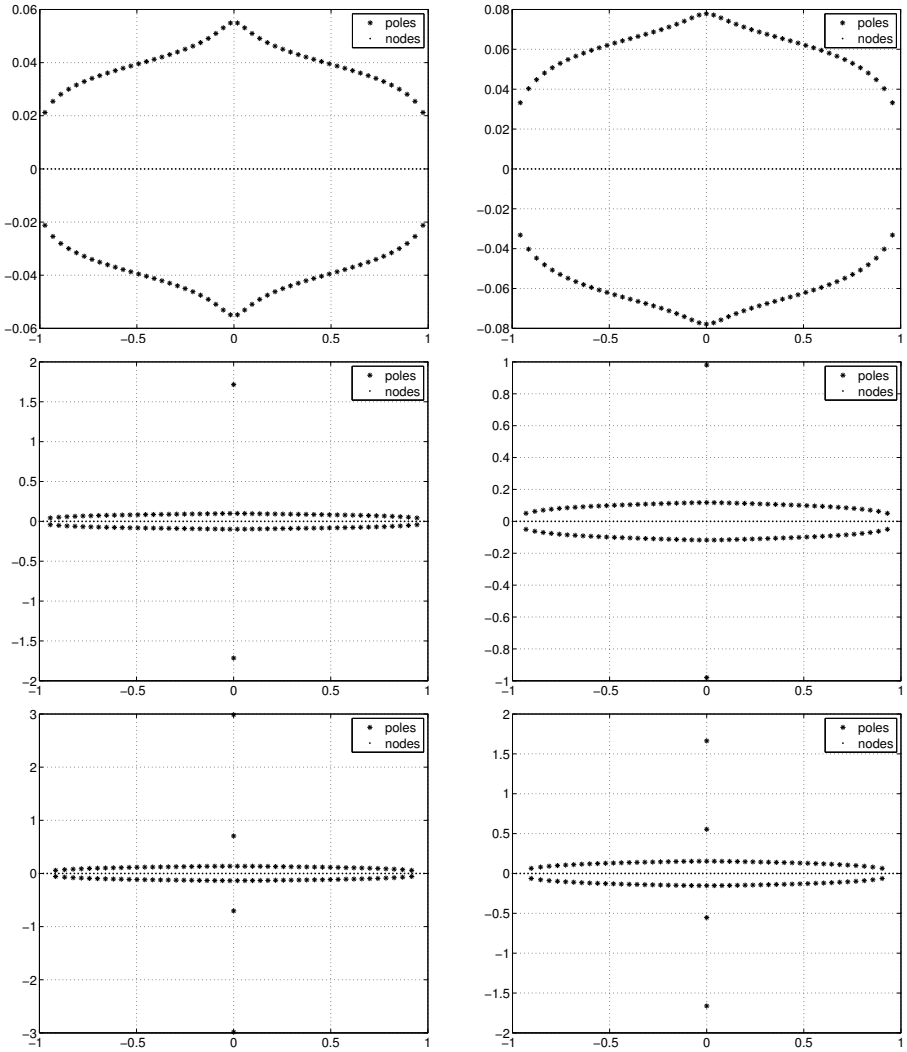

Figure 2.1: Approximate location of the poles of Floater–Hormann interpolation for equispaced nodes with $n = 100$ and $d = 0, 1, 2, 3, 4, 5$ (from left to right and top to bottom).

Table 2.1: Absolute errors and experimental convergence order (e.o.) in Floater–Hormann interpolation of $f(x) = \exp(x)$ in $[-1, 1]$.

n	$d = 0$	e.o.	$d = 1$	e.o.	$d = 3$	e.o.	$d = 5$	e.o.
10	1.2e−01		3.7e−03		5.0e−05		8.7e−07	
20	6.2e−02	0.9	9.4e−04	2.0	3.6e−06	3.8	1.9e−08	5.5
40	3.2e−02	0.9	2.4e−04	2.0	2.4e−07	3.9	3.4e−10	5.8
80	1.6e−03	1.0	5.9e−05	2.0	1.5e−08	4.0	5.6e−12	5.9
160	8.3e−03	1.0	1.5e−05	2.0	9.8e−10	4.0	9.1e−14	6.0
320	4.2e−03	1.0	3.7e−06	2.0	6.2e−11	4.0	4.0e−15	4.5
640	2.1e−03	1.0	9.2e−07	2.0	3.9e−12	4.0	4.9e−15	−0.3

procedure.

In Figure 2.1 we show the location of the poles of Floater–Hormann interpolants with equispaced nodes. These pictures and further experiments, not displayed, lead to the following observations for which we have not found any theoretical explanation yet. There are always d or $d+1$ purely imaginary poles; the latter is the case if d is odd. The real parts of the poles range from the $(d+1)$st to the $(n+1-d)$th node, which might follow from Theorem 12 in [72] applied to $d \geq 1$, since the $(d+1)$st until the $(n+1-d)$th barycentric weights have all the same absolute value, see (2.16), so that the idea of the proof of that theorem would work in this case and in the limit $n \to \infty$. Experiments with various values of n reveal that most of the poles come closer to the interval as n increases; only the purely imaginary ones move away.

2.3.4 Numerical Results

We only show two numerical examples of Floater–Hormann interpolation with equispaced nodes and the barycentric formula; a few more can be found in the original paper [46] and in subsequent sections.

The first example is the interpolation of the entire function $\exp(x)$ in $[-1, 1]$ with $d = 0, 1, 3, 5$ and various values of n. The maximal absolute approximation error throughout the interval as well as the experimental rates of convergence are documented in Table 2.1. For the notion of experimental order of convergence, see [33]. For this example, Theorem 2.3 is nicely illustrated, and for $d = 5$ the error reaches machine precision and remains at that magnitude also for larger

Table 2.2: Absolute errors in Floater–Hormann interpolation of Runge's function $1/(1+25x^2)$ in $[-1,1]$.

n	$d=0$	$d=1$	$d=3$	$d=5$	$d=20$	$d=50$
50	1.2e−03	4.7e−05	5.9e−07	8.1e−07	1.8e−02	4.8e+06
500	1.2e−04	4.7e−07	3.0e−11	1.1e−14	6.7e−13	4.8e−04
5e+3	1.2e−05	4.6e−09	1.2e−14	1.1e−14	1.0e−14	1.3e−07
5e+4	1.2e−06	4.7e−11	2.9e−14	2.9e−14	3.6e−14	2.8e−14
5e+5	1.2e−07	5.1e−13	1.2e−13	9.9e−14	1.1e−13	1.2e−13

n. The amount by which the error might increase again with increasing n is explained in Section 3.3. For large d and n, Theorem 2.3 might not always be satisfied in double precision arithmetic since rounding errors might be amplified during the interpolation process; the more precise reason is the conditioning of this interpolation procedure, as we will see in Section 3.3.

Our second example is Runge's [92] function $1/(1+25x^2)$ in $[-1,1]$, which is *the* archetypical example when interpolation with equispaced nodes is tested, since polynomial interpolation diverges in that case, even in exact arithmetic. The reason is the location of the poles of this function, which are in some sense too close to the interval. We shall come back to the theoretical explanation in more detail in Section 4.1. Table 2.2 displays the absolute interpolation errors with $d = 0, 1, 3, 5, 20, 50$ and n between 50 and 500 000. With $d = 0$ and 1, the experimental orders 1 and 2 can easily be read from the tabulated errors. With all other values of d, convergence is also guaranteed, as opposed to the polynomial case, which is implicitly shown with $d = n = 50$. The maximal error is about 5 million: This is essentially Runge's phenomenon and may also include the effects of the very bad conditioning of polynomial interpolation between equispaced nodes. With $d = 50$ and larger values of n, d is "far enough" away from n, so that the behaviour of the rational interpolant does not resemble that of the polynomial any longer and the approximation becomes good again. The intuitive reason why there is no Runge phenomenon any longer, despite the fact that the rational interpolant is nothing other than a blend of polynomial interpolants of degree as high as 50 here, is that each polynomial from the blend only covers a small part of the interval and does not "see" the poles of Runge's example. We give more details of this observation in Section 4.2. The bad conditioning of polynomial interpolation with equispaced nodes is not observed

in the rational analogue, either. In this example, Floater–Hormann interpolation evaluated with the barycentric formula is obviously well-conditioned and stable. The next chapter will give theoretical evidence for the stability of the evaluation of the rational interpolant as well as its good condition, especially with equispaced nodes.

Chapter 3

Condition and Numerical Stability

An idea that makes sense in exact arithmetic need not necessarily work in numerical floating point arithmetic with finite precision. Issues related to rounding errors and their amplification, to the condition of problems and the stability of algorithms need to be treated.

Polynomial interpolation between equispaced nodes for analytic functions with bounded derivatives such as $\exp(x)$ converges exponentially in infinite precision; see, e.g., the error bound (4.19). However, such interpolation is highly ill-conditioned, and also, evaluating the polynomial with the second barycentric formula is unstable as the number of nodes becomes large. The opposite is true of polynomial interpolation between Chebyshev points.

After a brief introduction in Section 3.1 to the concepts of numerical stability and conditioning, we study in Section 3.2 the stability of polynomial barycentric interpolation and linear barycentric rational interpolation, including Floater–Hormann interpolation. Section 3.3 deals with Lebesgue functions and Lebesgue constants, i.e., the condition numbers, which will in addition allow us to draw conclusions about the numerical stability of the second barycentric formula for polynomial and linear rational interpolation. In particular, it will turn out that Floater–Hormann interpolation with small d and equispaced nodes is nearly as stable and well-conditioned as polynomial interpolation in Chebyshev points.

The main references for this chapter are Higham's book [62] and his paper [63], Salazar Celis's thesis [95] and the papers [18, 19, 64] on the Lebesgue constants for Floater–Hormann interpolation.

3.1 Basic Concepts

We begin with a review of some basic notions from numerical analysis. First we treat the concept of stability of a method. This gives us information on how much a result computed with a certain algorithm may differ from the value computed in exact arithmetic. The only sources of error considered in this investigation are the imprecision due to storing numbers on a computer and that coming from previous computations. A numerical algorithm or method for computing $f(x) = y$, which produces the result $\widehat{y}$ in finite precision, is *numerically stable* [62] if, for any x for which the method is designed,

$$\widehat{y} + \Delta y = f(x + \Delta x), \qquad |\Delta y| \leq \epsilon |y|, \quad |\Delta x| \leq \eta |x|, \tag{3.1}$$

holds with sufficiently small ϵ and η. The absolute and relative errors of $\widehat{y}$ are called *forward errors*, and a method is called *forward stable* if the forward errors are small for every admissible input data. For given data, the *backward error* is the smallest $|\Delta x|$ or $|\Delta x|/|x|$ such that in exact arithmetic $\widehat{y} = f(x + \Delta x)$; i.e., $x + \Delta x$ is the input data for which $\widehat{y}$ is the exact result. If the backward error of every result computed by a method is small, then this method is called *backward stable*. The expression in (3.1) is called *mixed forward-backward error*.

A concept which is independent of the method or algorithm and depends only on the problem itself is the *condition*. The *condition number* of a problem is a measure for the effect on the output caused by errors in the input, which may be due to perturbations in the data, such as measurement errors in a physical experiment. If the condition number of a problem is small, then the problem is *well-conditioned*, otherwise it is *ill-conditioned*. The notions of forward error, backward error and condition number are related as follows,

$$\text{forward error} \lesssim \text{condition number} \times \text{backward error}.$$

An ill-conditioned problem can thus cause any method, including a backward stable, to produce large forward errors.

In Section 3.2 we investigate the stability of barycentric polynomial interpolation and linear barycentric rational interpolation, and in Section 3.3 we address the condition of interpolating data with these methods.

3.2 Stability of Linear Barycentric Interpolation

We are now prepared to review the stability of polynomial interpolation with the first and second barycentric form as presented in [63] and that of linear barycentric rational interpolation as given in [95]. The error analysis requires the definition of the standard model of floating point arithmetic [62]:

$$fl(x \text{ op } y) = (x \text{ op } y)(1+\delta)^{\pm 1}, \qquad |\delta| \leq \varepsilon, \quad \text{op} = +, -, *, /,$$

with ε the unit roundoff. The error bounds below are displayed compactly with the use of the *error counter*

$$\langle k \rangle = \prod_{i=1}^{k} (1+\delta_i)^{\rho_i}, \qquad \rho_i = \pm 1, \quad |\delta_i| \leq \varepsilon. \tag{3.2}$$

In the following, the data $f_0, \ldots, f_n$ and the nodes $x_0, \ldots, x_n$ are assumed to be floating point numbers.

Theorem 3.1. *The polynomial interpolant $\widehat{p}_n$ as computed from the first barycentric form* (2.6) *satisfies*

$$\widehat{p}_n(x) = L(x) \sum_{i=0}^{n} \frac{\lambda_i}{x - x_i} f_i \langle 5n+5 \rangle_i. \tag{3.3}$$

This result from [63] follows from carefully counting the numbers of operations involved in computing the barycentric weights and in the evaluation of p_n. The first barycentric form is thus backward stable since equation (3.3) shows that the computed polynomial interpolant can be interpreted as the exact interpolant of only slightly perturbed data. This formula can also be shown to be forward stable. To state the result from [63], we need the definition of the *condition number* at a point x of a function $g_n[f]$ interpolating f in $n+1$ nodes, namely

$$\text{cond}(x, g_n, f) = \lim_{\eta \to 0} \sup \left\{ \left| \frac{g_n[f](x) - g_n[f + \Delta f](x)}{\eta g_n[f](x)} \right|, |\Delta f| \leq \eta |f| \right\}.$$

Corollary 3.2. *Under the same assumptions as in Theorem 3.1, it follows that*

$$\frac{|p_n(x) - \widehat{p}_n(x)|}{|p_n(x)|} \leq \frac{(5n+5)\varepsilon}{1 - (5n+5)\varepsilon} \text{cond}(x, p_n, f).$$

The second barycentric formula is only forward stable [63] under certain hypotheses on the nodes. Before we state this result, we define the *Lebesgue function* associated with polynomial interpolation,

$$\Lambda_n(x) = \sum_{i=0}^{n} \frac{|\lambda_i|}{|x - x_i|} \Big/ \left| \sum_{i=0}^{n} \frac{\lambda_i}{x - x_i} \right|.$$

Theorem 3.3. *The polynomial interpolant $\widehat{p}_n$ as computed from the second barycentric form* (2.7) *satisfies*

$$\frac{|p_n(x) - \widehat{p}_n(x)|}{|p_n(x)|} \leq (3n+4)\mathrm{cond}(x, p_n, f)\varepsilon + (3n+2)\Lambda_n(x)\varepsilon + O(\varepsilon^2). \quad (3.4)$$

Equation (3.4) shows that the second barycentric formula is roughly as forward stable as the first barycentric formula under the condition that the Lebesgue function associated with polynomial interpolation is not larger than $\mathrm{cond}(x, p_n, f)$. The Lebesgue function is discussed in much more detail in Section 3.3, where favourable and not so favourable choices for the nodes are also discussed.

A similar result to Theorem 3.3 holds for linear barycentric rational interpolation [95], under the assumption that the barycentric weights can be computed by a backward stable algorithm, i.e., such that the computed weights $\widehat{w}_i$ satisfy

$$\widehat{w}_i = w_i\big(1 \pm O(\kappa_W \varepsilon)\big),$$

where κ_W is the relative condition number of the barycentric weights w_i. Such a condition holds for instance for the weights in Floater–Hormann interpolation. It follows from (2.12) and (3.2) that

$$\begin{aligned}
\widehat{w}_i &= \sum_{k \in J_i} (-1)^k \langle 1 \rangle_k \langle |J_i| - 1 \rangle_k \langle 2d \rangle_k \prod_{\substack{j=k \\ j \neq i}}^{k+d} \frac{1}{x_i - x_j} \\
&= \sum_{k \in J_i} (-1)^k \prod_{\substack{j=k \\ j \neq i}}^{k+d} \frac{1}{x_i - x_j} \langle |J_i| + 2d \rangle_k \\
&= w_i\big(1 \pm O((|J_i| + 2d)\varepsilon)\big)
\end{aligned}$$

where $|J_i|$ denotes the number of elements in J_i.

Theorem 3.4. *The linear barycentric rational interpolant* $\widehat{r}_n$ *computed from* (2.8) *satisfies*

$$\begin{aligned}\frac{|r_n(x)-\widehat{r}_n(x)|}{|r_n(x)|} \leq &\big(n+4+O(\kappa_W)\big)\mathrm{cond}(x,r_n,f)\varepsilon \\ &+\big(n+2+O(\kappa_W)\big)\Lambda_n(x)\varepsilon+O(\varepsilon^2),\end{aligned} \tag{3.5}$$

where $\Lambda_n(x)$ *is the Lebesgue function* (3.7) *associated with linear barycentric rational interpolation.*

Floater–Hormann interpolation is thus forward stable for nodes for which the associated Lebesgue function is small enough. We will now see that this is the case with equispaced nodes and reasonably small d.

3.3 Lebesgue Functions and Lebesgue Constants

3.3.1 Definition and Properties

The original definition of the Lebesgue function and constant comes from the general theory of linear interpolation operators; see [26, 87]. Suppose we have a set X, which contains the $n+1$ interpolation nodes and which is the domain of the functions being considered. We are looking for a linear interpolant g from an $(n+1)$-dimensional vector space A, such that g can be represented as in (2.1) by a linear combination of basis functions b_i satisfying the Lagrange property like the ℓ_i in (2.3). If f is any function defined on X, an interpolant g can be found by writing

$$Lf=\sum_{i=0}^{n} b_i f(x_i),$$

with L a linear and idempotent operator. Such an operator is called a *projection*. For the *Lebesgue function* associated with L,

$$\Lambda_n(x)=\sum_{i=0}^{n}|b_i(x)|,$$

it follows [26, 87] that the *Lebesgue constant*

$$\Lambda_n=\sup_{x\in X}\Lambda_n(x)$$

is the norm of the associated interpolation operator L, namely

$$\|L\| = \sup_{\|f\|\leq 1} \|Lf\| = \Lambda_n.$$

An immediate application of the Lebesgue constant is the characterisation of the approximation error via *best approximation* in A. For a function f defined on X, a best approximation g^* to f from A satisfies

$$\|f - g^*\| = \inf_{u\in A} \|f - u\|.$$

From the triangle inequality and the hypotheses on L, it follows that

$$\|f - Lf\| \leq (1 + \Lambda_n)\|f - g^*\|. \tag{3.6}$$

Let us return to the particular case where $X = [a, b]$ and g is the polynomial interpolant p_n or a linear rational interpolant r_n to f. The Lebesgue function associated with g in barycentric form is given by

$$\Lambda_n(x) = \sum_{i=0}^{n} \frac{|w_i|}{|x - x_i|} \Big/ \left| \sum_{i=0}^{n} \frac{w_i}{x - x_i} \right|, \tag{3.7}$$

and the Lebesgue constant is

$$\Lambda_n = \max_{a\leq x\leq b} \Lambda_n(x). \tag{3.8}$$

We denote by $\mathcal{P}_n$ the $(n+1)$-dimensional vector space of polynomials of degree less than or equal to n and by p_n^* the best approximation in $\mathcal{P}_n$ of a fixed function f. The existence and uniqueness of p_n^* are shown, e.g., in [87]. The approximation error in polynomial interpolation is bounded via the Lebesgue constant associated with p_n as in (3.6),

$$\|f - p_n\| \leq (1 + \Lambda_n)\|f - p_n^*\|. \tag{3.9}$$

Since the Floater–Hormann interpolants r_n from (2.8) with weights (2.12) reproduce polynomials of degree at least d, it easily follows from the triangle inequality that

$$\|f - r_n\| \leq (1 + \Lambda_n)\|f - p_d^*\|,$$

where p_d^* is the best approximation to f in $\mathcal{P}_d$ and Λ_n is the Lebesgue constant associated with r_n.

This is an additional motivation to look at the Lebesgue constants, in addition to Theorems 3.3 and 3.4. Here is yet another one; we write g_n for p_n or r_n. Let every data value f_i be given with an absolute error or perturbation of at most ε_p, for example due to rounding, noise, or measurement imprecision. Then the maximum distance in $[a, b]$ between the interpolant $\widetilde{g}_n$ of the perturbed data and the interpolant g_n of the exact data is bounded as

$$\max_{a\le x\le b} |g_n(x) - \widetilde{g}_n(x)| \le \varepsilon_p \max_{a\le x\le b} \Lambda_n(x) = \varepsilon_p \Lambda_n,$$

as can be easily seen from the linearity of g_n. Thus, Λ_n is the worst possible error amplification and, since g is linear in the data, coincides with the condition number of the interpolation process [85].

3.3.2 The Condition of Polynomial Interpolation

Numerous authors have derived results about the Lebesgue function and constant associated with polynomial interpolation in various kinds of nodes; see [22, 23, 100] and the references therein. It is well known [22] that the Lebesgue constant associated with polynomial interpolation between $n+1$ nodes distributed in any way always increases at least logarithmically with the number of nodes,

$$\Lambda_n \ge \frac{2}{\pi}\log(n+1) + \frac{2}{\pi}\Big(\gamma + \log\big(\frac{4}{\pi}\big)\Big),$$

where $\gamma \approx 0.577$ is Euler's constant. Such a rate is achieved, for instance, for Chebyshev points of the first and second kind [23, 39, 90, 100],

$$\Lambda_n \le 1 + \frac{2}{\pi}\log(n+1). \tag{3.10}$$

In contrast to this favourable behaviour, the Lebesgue constant for polynomial interpolation at equispaced nodes grows exponentially, with the asymptotic behaviour

$$\Lambda_n \sim \frac{2^{n+1}}{en\log(n)}$$

as $n \to \infty$. More detailed results and other approaches to describing the error amplification may be found in [40, 61, 98, 116] and the references therein. The bad condition, together with Runge's phenomenon [42, 92], see Chapter 4, makes polynomial interpolation in equispaced nodes usually useless for $n \ge 50$ and not much better for smaller n. In fact, interpolation in these nodes is a

challenging problem; it was shown in [85] that it is not possible to develop an approximation method which is simultaneously well-conditioned and converges at an exponential rate as the number of nodes increases. It is, however, possible to have well-conditioned interpolation between equispaced nodes which converges at an algebraic rate. We shall see in the next section that this is achieved by linear rational interpolation under some restrictions.

3.3.3 The Condition of Linear Barycentric Rational Interpolation[1]

Berrut and Mittelmann [15] determine linear rational interpolants with small Lebesgue constants for given nodes by numerically optimising the denominator of the interpolant. Here we shall concentrate on the family of barycentric rational interpolants introduced by Floater and Hormann.

Bos, De Marchi and Hormann [18] analysed the Lebesgue constant associated with Berrut's rational interpolant, i.e., the Floater–Hormann interpolant with $d = 0$, and showed the following result.

Theorem 3.5. *The Lebesgue constant associated with Berrut's interpolant in equispaced nodes satisfies*

$$c_n \log(n+1) \leq \Lambda_n \leq 2 + \log(n), \tag{3.11}$$

where $c_n = 2n/(4+n\pi)$*, with* $\lim_{n\to\infty} c_n = 2/\pi$*.*

Logarithmic growth of the Lebesgue constant associated with this interpolant for increasing n has also been derived for more general, so called well-spaced, nodes; see [20].

The general case $d \geq 1$ requires a different approach, since the study of the Lebesgue function in the form (3.7) results in rather complicated expressions, whereas the original form of the rational interpolants as blends of polynomials allows for much shorter proofs.

We will now show that the Lebesgue constant associated with the family of Floater–Hormann interpolants with $d \geq 1$ grows logarithmically in the number of interpolation nodes if these are equispaced or quasi-equispaced. This is achieved by establishing logarithmic upper and lower bounds.

To define *quasi-equispaced* nodes [41], we assume that there exists some *global mesh ratio* $M \geq 1$, which does not depend on n, such that

[1] The figures in this section stem from [19, 64] and were prepared by Kai Hormann.

$$\frac{h}{h_*} \leq M, \tag{3.12}$$

with h from (2.9) and

$$h_* = \min_{0\leq j\leq n-1}(x_{j+1} - x_j),$$

i.e., the minimal distance between neighbouring nodes. Such node distributions may arise from experiments, where the aim is to sample a signal at a constant rate, which in turn is perturbed by imprecision in the measurement locations. We study the behaviour of the derivatives of the rational interpolants for such nodes in Chapter 5. Notice that equispaced nodes are quasi-equispaced nodes with global mesh ratio $M = 1$. For simplicity, we assume that the interpolation interval is $[0, 1]$.

The Lebesgue function associated with linear barycentric interpolation is invariant under rescaling and translation of the interval, under the assumption that the weights are also invariant under such a transformation. The latter condition is satisfied in Floater–Hormann interpolation since the simplified weights depend only on relative distances between the nodes, as can be deduced from equation (2.12). To see the invariance of the Lebesgue function, we map the nodes and the evaluation point $x \in [a, b]$ by the map $x \mapsto (x - a)/(b - a)$ onto $[0, 1]$. A short computation with (3.7) shows that

$$\Lambda_n\big((x - a)/(b - a)\big) = \Lambda_n(x). \tag{3.13}$$

We will now derive lower and upper bounds on the Lebesgue constants associated with Floater–Hormann interpolation. This was done in collaboration with Bos, De Marchi and Hormann; see [19, 64]. We only present the results for quasi-equispaced nodes since those for equispaced nodes then follow immediately by letting $M = 1$. Let us begin with a lemma.

Lemma 3.6. *If the interpolation nodes are quasi-equispaced, then the weights in* (2.12) *satisfy*

$$W_i \leq |w_i| \leq M^d W_i,$$

where

$$W_i = \frac{1}{h^d d!} \sum_{k\in J_i} \binom{d}{i-k}$$

for $i = 0, 1, \ldots, n$. *Moreover,*

$$W_i \leq \frac{2^d}{h^d d!} =: W,$$

with equality if and only if $d \leq i \leq n-d$.

Proof. By the definition of the minimal and maximal node distances h_* and h, the distance between two arbitrary nodes x_i and x_j satisfies

$$h_*|i-j| \leq |x_i - x_j| \leq h|i-j|. \tag{3.14}$$

This leads to the lower bound

$$\begin{aligned}
|w_i| &= \sum_{k\in J_i} \prod_{j=k, j\neq i}^{k+d} \frac{1}{|x_i - x_j|} \\
&\geq \frac{1}{h^d} \sum_{k\in J_i} \prod_{j=k, j\neq i}^{k+d} \frac{1}{|i-j|} \\
&= \frac{1}{h^d} \sum_{k\in J_i} \frac{1}{(i-k)!(k+d-i)!} = \frac{1}{h^d d!} \sum_{k\in J_i} \binom{d}{i-k}
\end{aligned}$$

and similarly, using (3.12), to the upper bound

$$|w_i| \leq \frac{1}{h_*^d d!} \sum_{k\in J_i} \binom{d}{i-k} \leq \frac{M^d}{h^d d!} \sum_{k\in J_i} \binom{d}{i-k}.$$

The statement about the upper bound W of W_i follows directly from the definition of the index set J_i in (2.13). □

As a consequence of Lemma 3.6, the absolute values of the nonsimplified weights for rational interpolation in quasi-equispaced nodes may have larger ranges of magnitude than the weights for equispaced nodes with the same value of d. The quotient of the largest barycentric weight by the smallest in absolute values plays an important role, at least in polynomial interpolation, since the associated Lebesgue constant can be bounded [15, 17] as

$$\Lambda_n \geq \frac{1}{2n^2} \frac{\max_{0\leq i\leq n} |w_i|}{\min_{0\leq i\leq n} |w_i|},$$

which follows from Markov's inequality [106]. This quotient thus gives a first intuitive estimation on the quality of an interpolant; see also [52].

Theorem 3.7. *If the interpolation nodes are quasi-equispaced or equispaced ($M = 1$), then the Lebesgue constant associated with Floater–Hormann interpolation* (3.8) *satisfies*

$$\Lambda_n \le \big(2 + M\log(n)\big) \cdot \begin{cases} \frac{3}{4}M, & d = 0, \\ 2^{d-1}M^d, & d \ge 1. \end{cases} \tag{3.15}$$

Proof. From (3.7) it follows that $\Lambda_n(x_k) = 1$ for $k = 0, 1, \ldots, n$. Therefore, let $x_k < x < x_{k+1}$ for some k with $0 \le k \le n-1$, and rewrite the Lebesgue function in (3.7) as

$$\Lambda_n(x) = \frac{(x - x_k)(x_{k+1} - x)\sum_{i=0}^{n} \frac{|w_i|}{|x-x_i|}}{(x - x_k)(x_{k+1} - x)\big|\sum_{i=0}^{n} \frac{w_i}{x-x_i}\big|}.$$

We denote the numerator of $\Lambda_n(x)$ in the above expression by $N_k(x)$ and the denominator by $D_k(x)$ and then derive an upper bound on the numerator $N_k(x)$, following the proof of Theorem 2 in [18]. By Lemma 3.6,

$$N_k(x) = (x-x_k)(x_{k+1}-x)\sum_{j=0}^{n} \frac{|w_j|}{|x-x_j|} \le M^d W (x-x_k)(x_{k+1}-x)\sum_{j=0}^{n} \frac{1}{|x-x_j|},$$

and further, using the inequalities (3.12) and (3.14), and the fact that $x_k < x < x_{k+1}$,

$$\begin{aligned} \frac{N_k(x)}{M^d W} &\le (x_{k+1} - x_k) + (x - x_k)(x_{k+1} - x)\bigg(\sum_{j=0}^{k-1} \frac{1}{x - x_j} + \sum_{j=k+2}^{n} \frac{1}{x_j - x}\bigg) \\ &\le h + \Big(\frac{h}{2}\Big)^2 \bigg(\sum_{j=0}^{k-1} \frac{1}{x_k - x_j} + \sum_{j=k+2}^{n} \frac{1}{x_j - x_{k+1}}\bigg) \\ &\le h + \Big(\frac{h}{2}\Big)^2 \frac{M}{h}\bigg(\sum_{j=1}^{k} \frac{1}{j} + \sum_{j=1}^{n-k-1} \frac{1}{j}\bigg) \\ &\le h + \frac{Mh}{2}\log(n). \end{aligned}$$

To establish a lower bound on the denominator $D_k(x)$, we distinguish the cases $d = 0$ and $d \ge 1$. If $d = 0$[2], then the weights in (2.12) simplify to $w_i = (-1)^i$, and we have

[2]This proof for the case $d = 0$ is entirely due to Kai Hormann.

$$D_k(x) = (x-x_k)(x_{k+1}-x)\Big|\sum_{i=0}^{n}\frac{(-1)^i}{x-x_i}\Big|$$
$$\geq (x-x_k)(x_{k+1}-x)\Big(-\frac{1}{x-x_{k-1}}+\frac{1}{x-x_k}+\frac{1}{x_{k+1}-x}-\frac{1}{x_{k+2}-x}\Big) \tag{3.16}$$
$$= (x_{k+1}-x_k)-(x-x_k)(x_{k+1}-x)\Big(\frac{1}{x-x_{k-1}}+\frac{1}{x_{k+2}-x}\Big) \geq \frac{2}{3}h_*, \tag{3.17}$$

where the inequality in (3.17) can be obtained by first multiplying both sides with $3(x-x_{k-1})(x_{k+2}-x)$ and then verifying that the quadratic polynomial

$$\begin{aligned} Q_k(x) =& \big(3(x_{k+1}-x_k)-2h_*\big)(x-x_{k-1})(x_{k+2}-x) \\ &- 3(x-x_k)(x_{k+1}-x)(x_{k+2}-x_{k-1}) \end{aligned}$$

is nonnegative. Substituting $a = x_k - x_{k-1}$, $b = x_{k+1}-x_k$, $c = x_{k+2}-x_{k+1}$, and $y = x - x_k$, we find

$$Q_k(y+x_k) = (3a+2h_*+3c)y^2+(2ah_*-2bh_*-2ch_*-6ab)y+a(b+c)(3b-2h_*);$$

it is then easy to check that the minimal value of this quadratic expression is

$$\frac{a+b+c}{3a+2h_*+3c}\big(6ac(b-h_*)+a(bc-h_*^2)+b(ac-h_*^2)+c(ab-h_*^2)\big),$$

which is nonnegative, because $0 < h_* \leq \min(a,b,c)$. Notice that the inequality in (3.16) also holds if $k=0$ or $k=n-1$, for example by letting $x_{-1} = x_0 - h$ and $x_{n+1} = x_n + h$.

To handle the case $d \geq 1$, we use the two results (2.15) and (2.21) from [46] to get

$$D_k(x) = (x-x_k)(x_{k+1}-x)\Big|\sum_{i=0}^{n}\frac{w_i}{x-x_i}\Big| = (x-x_k)(x_{k+1}-x)\Big|\sum_{i=0}^{n-d}\lambda_i(x)\Big|$$
$$\geq (x-x_k)(x_{k+1}-x)|\lambda_j(x)| = \frac{(x-x_k)(x_{k+1}-x)}{\prod_{i=j}^{j+d}|x_i-x|},$$

for any $j \in J_k \setminus \{k-d\}$. It then follows from (3.14) that

$$
\begin{aligned}
D_k(x) &\geq \frac{1}{\prod_{j=i}^{k-1}(x-x_j)\prod_{j=k+2}^{i+d}(x_j-x)} \\
&\geq \frac{1}{\prod_{j=i}^{k-1}(x_{k+1}-x_j)\prod_{j=k+2}^{i+d}(x_j-x_k)} \\
&\geq \frac{1}{h^{d-1}(k+1-i)!(i+d-k)!} \geq \frac{1}{h^{d-1}d!}. \qquad \square
\end{aligned}
$$

For equispaced nodes, i.e., $M = 1$, the upper bound in Theorem 3.7 improves the one given in [18] for $d = 0$ by a factor of $3/4$, and for $d \geq 1$ it simplifies to the one given in [19].

Theorem 3.8. *If the interpolation nodes are quasi-equispaced or equispaced ($M = 1$), then the Lebesgue constant associated with Floater–Hormann interpolation* (3.8) *satisfies*

$$
\Lambda_n \geq \frac{1}{2^{d+2}M^{d+1}}\binom{2d+1}{d} \cdot \begin{cases} \big(2+\log(2n+1)\big), & d=0, \\ \log\big(\frac{n}{d}-1\big), & d \geq 1. \end{cases} \tag{3.18}
$$

Proof. From numerical experiments, see Figure 3.1, it appears that for $d \geq 2$ the Lebesgue function (3.7) with equispaced nodes obtains its maximum approximately halfway between x_0 and x_1 (and halfway between x_{n-1} and x_n because of the symmetry with respect to the mid-point of the interval). In order to establish the claimed lower bound, we derive a lower bound for the numerator $N(x) = \sum_{i=0}^{n} \frac{|w_i|}{|x-x_i|}$ and an upper bound for the denominator $D(x) = \big|\sum_{i=0}^{n} \frac{w_i}{x-x_i}\big|$ of the Lebesgue function in (3.7) at the midpoint $x^* = (x_0 + x_1)/2$ between the first two interpolation nodes also for quasi-equispaced nodes.

According to (3.14), the distance between x^* and x_i satisfies

$$
\frac{h_*}{2}|2i-1| \leq |x^*-x_i| \leq \frac{h}{2}|2i-1|. \tag{3.19}
$$

For the numerator, we omit some terms of the sum and use the lower bound W from Lemma 3.6 as well as inequality (3.19) to get

$$
N(x^*) = \sum_{i=0}^{n} \frac{|w_i|}{|x^*-x_i|} \geq \sum_{i=d}^{n-d} \frac{|w_i|}{|x^*-x_i|} \geq W \sum_{i=d}^{n-d} \frac{1}{|x^*-x_i|} \geq W\frac{2}{h}\sum_{i=d}^{n-d} \frac{1}{|2i-1|}.
$$

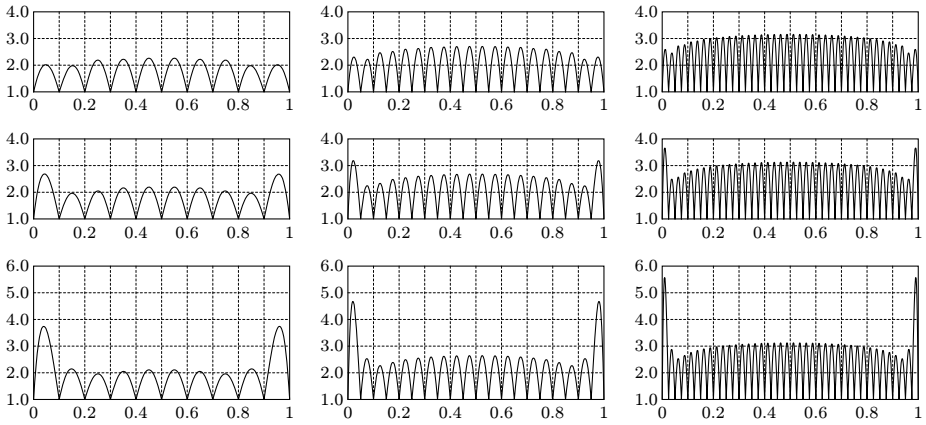

Figure 3.1: Lebesgue functions associated with Floater–Hormann interpolants for $d = 1$ (top), $d = 2$ (middle), and $d = 3$ (bottom) with $n + 1$ equispaced nodes for $n = 10, 20, 40$.

If $d = 0$, we then have

$$\frac{N(x^*)}{W} \geq \frac{2}{h}\Big(1 + \sum_{i=1}^{n} \frac{1}{2i-1}\Big) \geq \frac{2}{h}\Big(1 + \int_1^{n+1} \frac{\mathrm{d}x}{2x-1}\Big) = \frac{1}{h}\big(2 + \log(2n+1)\big),$$

and for $d \geq 1$ we find

$$\begin{aligned}\frac{N(x^*)}{W} \geq& \frac{2}{h} \sum_{i=d}^{n-d} \frac{1}{2i-1} \geq \frac{2}{h} \int_d^{n-d+1} \frac{\mathrm{d}x}{2x-1} \\ =& \frac{1}{h} \log\Big(\frac{2n-2d+1}{2d-1}\Big) \geq \frac{1}{h} \log\Big(\frac{n}{d} - 1\Big).\end{aligned}$$

To bound the denominator, we first rewrite it in terms of the functions λ_j and notice that $\lambda_0(x^*)$ and $\lambda_1(x^*)$ both have the same sign, and that the subsequent $\lambda_j(x^*)$ oscillate in sign and decrease in absolute value. Therefore,

$$\begin{aligned}D(x^*) =& \Big|\sum_{i=0}^{n} \frac{w_i}{x - x_i}\Big| = \Big|\sum_{i=0}^{n-d} \lambda_i(x^*)\Big| \\ \leq& |\lambda_0(x^*)| + |\lambda_1(x^*)| = \frac{1}{\prod_{i=0}^{d} |x^* - x_i|} + \frac{1}{\prod_{i=1}^{d+1} |x^* - x_i|}.\end{aligned}$$

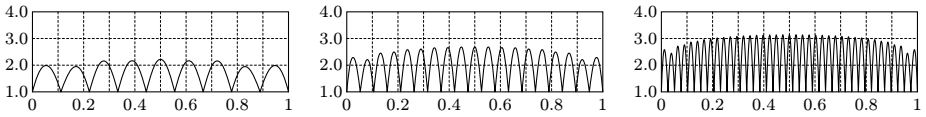

Figure 3.2: Lebesgue functions associated with Floater–Hormann interpolants for $d = 1$ with $n + 1$ equispaced nodes for $n = 9, 19, 39$.

We then conclude, using first (3.19) and then (3.12), that

$$\begin{aligned} D(x^*) &\leq \frac{2^{d+1}}{h_*^{d+1}\prod_{i=0}^{d}|2i-1|} + \frac{2^{d+1}}{h_*^{d+1}\prod_{i=1}^{d+1}|2i-1|} \\ &= \frac{2^{d+1}}{h_*^{d+1}}\left(\frac{2d+1}{\prod_{i=1}^{d+1}(2i-1)} + \frac{1}{\prod_{i=1}^{d+1}(2i-1)}\right) \leq \frac{2^{2d+2}M^{d+1}}{h^{d+1}d!\binom{2d+1}{d}}. \end{aligned} \qquad \square$$

Notice that the lower bound in Theorem 3.8 simplifies to the one given in [19] for $d \geq 1$ and equispaced nodes with $M = 1$. Moreover, the sum on the left-hand side of the last line in the above proof may be bounded from above by $(d+1)/(d!2^{d-1})$. This would give a slightly weaker but more eloquent leading factor $2^{d-2}/((d+1)M^{d+1})$ in the lower bound (3.18) of Λ_n, clearly showing that the Lebesgue constant grows exponentially with increasing d.

The following improved bound for the case $d = 1$ and equispaced nodes, which turns out to be again very similar to the one for $d = 0$ in (3.11), is essentially due to Bos and De Marchi.

Proposition 3.9. *If $d = 1$ and the interpolation nodes are equispaced, then the Lebesgue constant associated with Floater–Hormann interpolation* (3.8) *satisfies*

$$\Lambda_n \geq a_n \log(n) + b_n,$$

where $\lim_{n\to\infty} a_n = 2/\pi$ and $\lim_{n\to\infty} b_n = 0$.

The proof of Proposition 3.9 is similar to that of Theorem 3.8, except that one may use $x^* = 1/2$. According to numerical experiments, this is where the maximum of the Lebesgue function appears to occur with $d = 1$; see Figure 3.2.

We performed experiments to verify numerically that the behaviour of the Lebesgue constants associated with the family of barycentric rational interpolants is as predicted by the theoretical results. We began with investigating the Lebesgue constants with equispaced nodes. Figure 3.3 confirms that the

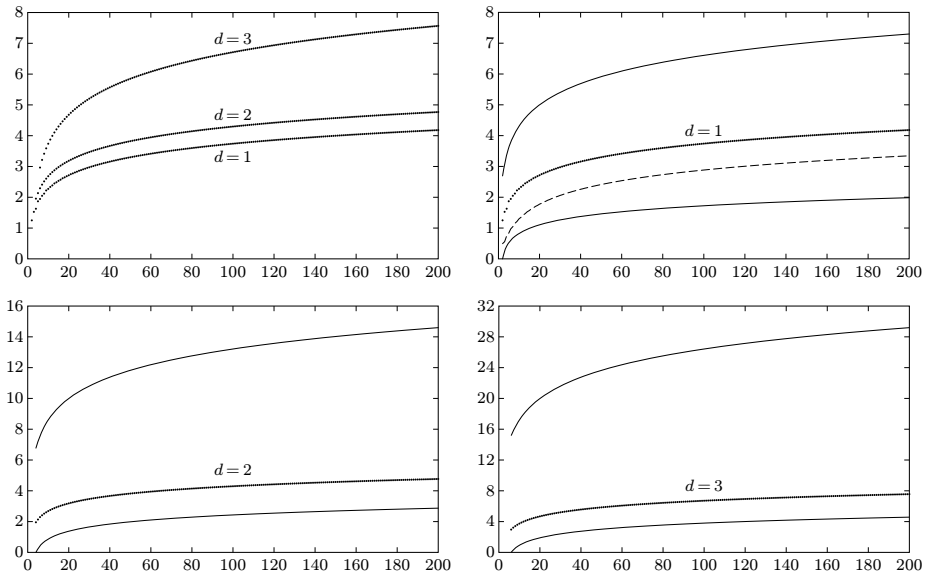

Figure 3.3: Comparison of the Lebesgue constants associated with Floater–Hormann interpolants with $n+1$ equispaced nodes for $2d \leq n \leq 200$ and $d = 1, 2, 3$ (top left) with the upper and lower bounds of Theorems 3.7 and 3.8. For $d = 1$, the improved lower bound in Proposition 3.9 is shown by the dashed curve (top right).

growth of Λ_n is logarithmic in the number of interpolation nodes. We omit plotting the case $d = 0$ here since numerical observations reveal that this case has similar Lebesgue constants to the case $d = 1$; compare Figure 3.3 (top right) and Figure 2 in [18]. These results in Figure 3.3 further suggest that for fixed d, the coefficient $\binom{2d+1}{d}/2^{d+2}$ of the logarithmic term in our lower bound in Theorem 3.8 is a better estimate of the true value than the factor 2^{d-1} in our upper bound in Theorem 3.7.

The lower and upper bounds indicate that, for fixed n, the growth of the Lebesgue constants with respect to d is exponential, which is confirmed by the examples in Figure 3.4. Finally, Figure 3.1 demonstrates that this exponential growth seems to always happen near the boundary of the interpolation interval, whereas the behaviour of the Lebesgue function away from the boundary is almost independent of d. This observation will be crucial in the construction of

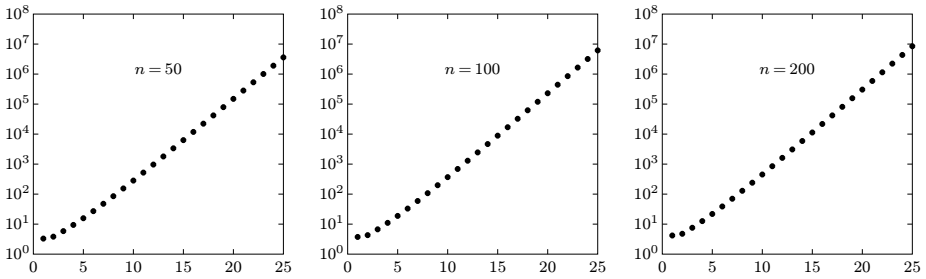

Figure 3.4: Lebesgue constants associated with Floater–Hormann interpolants with a fixed number of $n + 1$ equispaced nodes for $1 \leq d \leq 25$.

our extension of the Floater–Hormann rational interpolants in Chapter 7.

We finally present some results of extensive numerical experiments with quasi-equispaced nodes[3]. First, the situation where a set of equispaced nodes is perturbed randomly with a fixed maximum relative perturbation $\delta \in (0, 1/2)$ is considered. Since Lebesgue constants are invariant under translation and uniform rescaling of the interpolation nodes, see (3.13), we simply used

$$x_i = i + \delta_i, \qquad i = 0, \ldots, n,$$

with randomly chosen $\delta_i \in [-\delta, \delta]$. Hence, the global mesh ratio is at most $M = (1 + 2\delta)/(1 - 2\delta)$. For fixed d and n, we generated $N = 1,000,000$ different sets of such interpolation nodes and computed the corresponding Lebesgue constants. The vertical bars in Figure 3.5 show the ranges of these N Lebesgue constants and the crosses mark their averages. We observe that the variance of the Lebesgue constants grows rapidly with both d and δ and only slowly with n. Moreover, the average values are not too far from the Lebesgue constants in the equispaced setting with the same number of interpolation nodes, which are indicated by the circles in Figure 3.5, and the relative distance decreases with d. The case $d = 3$ also reveals that the random perturbations can actually improve the Lebesgue constant, and that this becomes less likely as n increases. In principle, one could expect the circles to always lie within the range of Lebesgue constants, as the perfectly uniform distribution is just a special case of the randomly perturbed nodes. The fact that this is not always the case in our experiments is explained by the bullets in Figure 3.5, which mark

[3]The major part of this investigation was carried out by Kai Hormann.

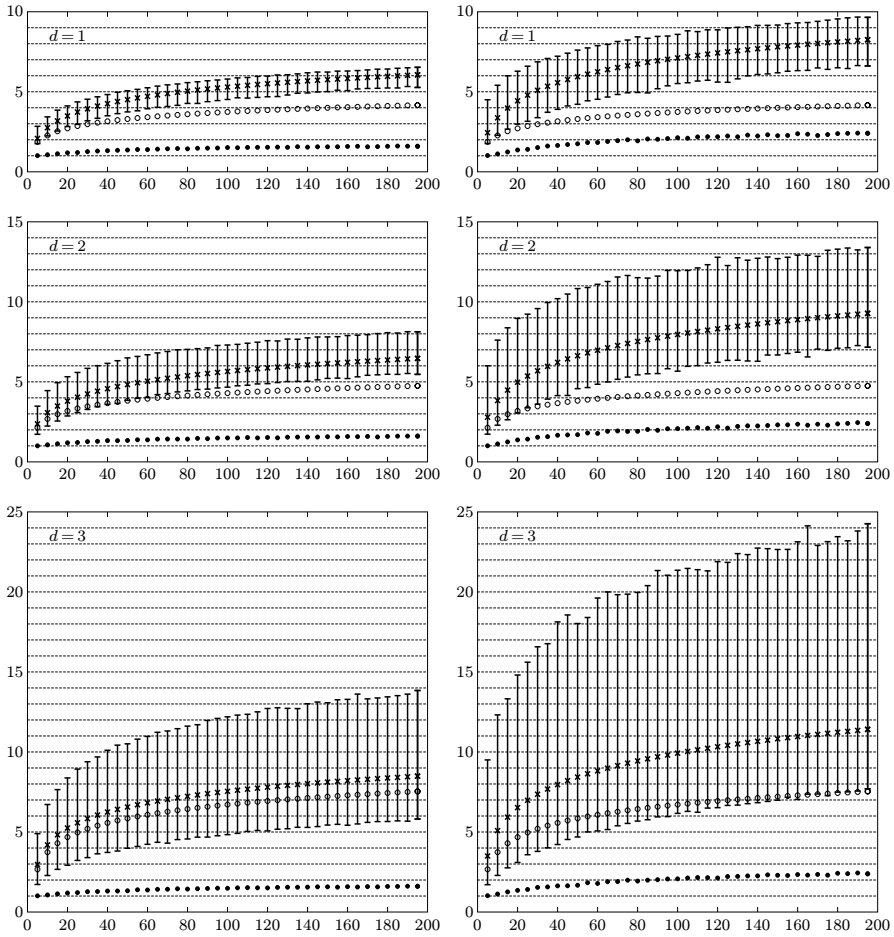

Figure 3.5: Minimum, maximum, and average Lebesgue constants (vertical bars with crosses) for quasi-equispaced nodes with a random perturbation of at most $\delta = 1/6$ (left) and $\delta = 3/10$ (right), that is, with global mesh ratios at most $M = 2$ (left) and $M = 4$ (right). Circles indicate the Lebesgue constants for equispaced nodes, bullets mark the minimal global mesh ratios. Plots for $d = 0$ are not shown because they are very similar to those for $d = 1$.

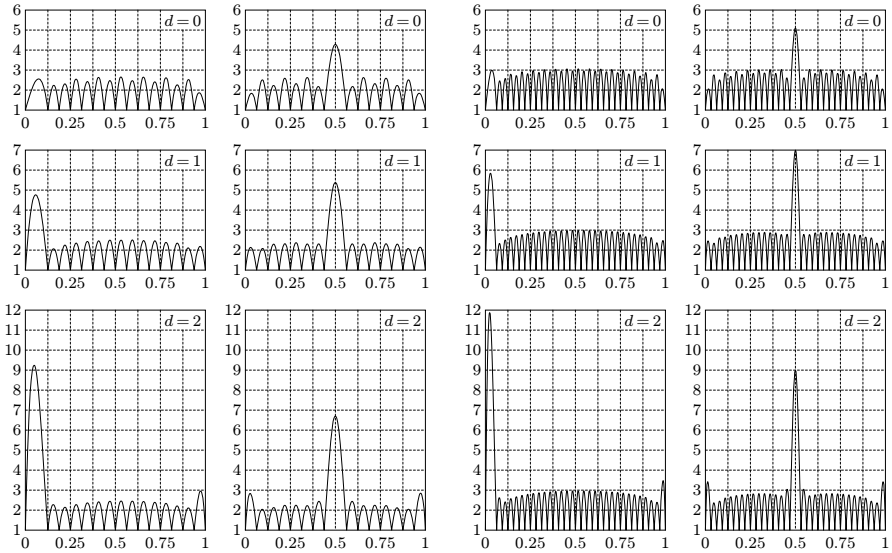

Figure 3.6: Lebesgue functions for two specific sets of $n = 15$ (left) and $n = 31$ (right) quasi-equispaced nodes with fixed global mesh ratio $M = 2$.

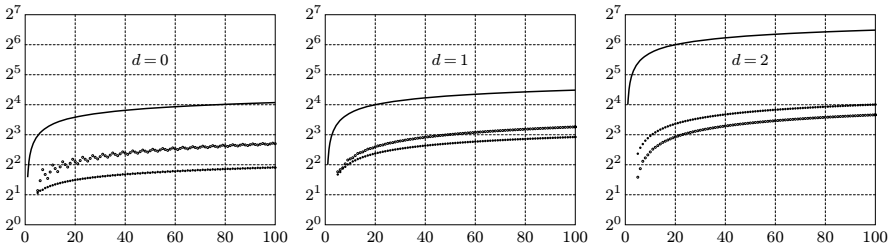

Figure 3.7: Lebesgue constants (bullets and circles) for the two specific sets of quasi-equispaced nodes with global mesh ratio $M = 2$ that are shown in Figure 3.6, compared with the upper bound in Theorem 3.7.

the smallest global mesh ratio M that we found among the N randomly created sets of nodes; the largest M is not shown because it was close to $(1+2\delta)/(1-2\delta)$ in all examples.

Figure 3.6 shows the Lebesgue functions for some sets of quasi-equispaced

nodes with global mesh ratio $M = 2$. We tested many such sets of nodes and found two cases to give particularly large Lebesgue constants: In the first case, we let the first subinterval $[x_0, x_1]$ be of length M and all others of length 1, in the second case we let the central subinterval $[x_{\lfloor n/2 \rfloor}, x_{\lfloor n/2 \rfloor+1}]$ be of length M and again all others of length 1. For a better comparison, we scaled the nodes so that $[x_0, x_n] = [0, 1]$. Except for $d = 0$ and the first set of nodes, the maximum of the Lebesgue function is always found in the single interval of length M, and for $d \geq 2$, the first set gives larger Lebesgue constants than the second set. This can also be seen in Figure 3.7, where the bullets mark the Lebesgue constants for the first set of nodes and the circles correspond to the second set. We observe that these are still quite far from the theoretical upper bound that we derived in Theorem 3.7, and it remains to be figured out, as discussed in [23] in the case of polynomial interpolation, if there are node configurations with a fixed global mesh ratio that lead to yet larger Lebesgue constants than those displayed here.

Concluding Remarks

From this chapter it becomes clear that the Lebesgue constants associated with Floater–Hormann interpolation are reasonably small when the nodes are equispaced or quasi-equispaced and when d is small to moderate. In this setting Floater–Hormann interpolation in barycentric form is thus forward stable—with no significant difference to polynomial interpolation evaluated with the second barycentric formula between Chebyshev points, see Theorems 3.3 and 3.4—has a small condition number and yields approximation errors that are not too far from those obtained with best polynomial interpolation of low degree. The bound on the forward error (3.5) is even very pessimistic in some cases: In Table 2.2 we have seen that interpolation with very large n and large d (up to $d = 50$) can still produce very small errors even if the Lebesgue constant is very large with these parameters. The only drawback for the moment is the rather slow algebraic convergence, which we shall remedy in Chapter 4, at least in the case when the interpolated function is analytic. In Chapter 7 we present an extension of the original Floater–Hormann family of rational interpolants that yields even smaller Lebesgue constants and thus allows for stable interpolation even with larger values of d.

Chapter 4

Interpolation of Analytic Functions

Polynomial interpolants of analytic functions may suffer from Runge's phenomenon [92], that is, the sequence of polynomials for an increasing number of nodes does not converge to the interpolated function between the nodes. This phenomenon of exact arithmetic is famous and often associated with polynomial interpolation and equispaced nodes in general. However, in polynomial interpolation it is not limited to equispaced nodes; on the other hand, it does not show up if the nodes are distributed in a special way. Even polynomial interpolation with equispaced nodes does not always suffer from this phenomenon, e.g., when the interpolated function is analytic in a sufficiently large region around the interval or when the interval of interpolation shrinks with the number of nodes.

Floater and Hormann demonstrated in [46] that their interpolant does not exhibit Runge-type behaviour with equispaced nodes. This might be a bit surprising at first sight since this family of rational interpolants is basically blended polynomial interpolation. Can Runge's phenomenon completely disappear with this kind of interpolants?

Much can be deduced from potential theory about convergence, divergence and Runge's phenomenon for polynomial interpolation and other schemes. We review parts of these well known results in Section 4.1. In the section following it, we generalise them to blended polynomial interpolation and establish a convergence result of Floater–Hormann interpolation for analytic functions. Moreover the question of a near optimal choice for the blending parameter d

is addressed, especially for equispaced nodes. The latter investigation builds upon recent convergence results as well as the analysis of the condition from Section 3.3.3. We end this chapter with a discussion on a good choice of interpolation nodes.

The main references for this chapter are [89, 94] regarding the well established potential theory and [57] for the recent analogue for Floater–Hormann interpolation.

4.1 Potential Theory for Polynomial Interpolation

Polynomial interpolants may converge or diverge geometrically, depending on the distribution of the nodes and the domain of analyticity of the interpolated function. To make this statement more precise, we take advantage of some notions and a result on polynomial interpolation stated in [48]; see also [94]. The essential ingredient in the analysis of the convergence behaviour of polynomial interpolation to analytic functions is the *Hermite error formula*.

Theorem 4.1. *If f is analytic inside a positively oriented contour Γ in the complex plane enclosing the nodes, then, for every z inside Γ, the interpolation error is given by the Hermite error formula,*

$$f(z) - p_n(z) = \frac{L(z)}{2\pi \mathrm{i}} \int_\Gamma \frac{f(t)}{L(t)(t-z)} \, \mathrm{d}t, \tag{4.1}$$

where L is the nodal polynomial from (2.4).

The proof of this theorem [31, 48] derives from the residue theorem, see, e.g., [77, 91]: It is sufficient to notice from the remark following (2.5) that the polynomial interpolant may be written as

$$p_n(z) = L(z) \sum_{i=0}^{n} \frac{f(x_i)}{L'(x_i)(z-x_i)}$$

(see also [25]), and that the integrand on the right-hand side of (4.1) has the simple pole $t = z$ with residue $f(z)/L(z)$ and the simple poles $t = x_i$, $i = 0, \ldots, n$, with residues $f(x_i)/(L'(x_i)(x_i - z))$.

Formula (4.1) motivates us to further investigate the quotient $L(z)/L(t)$ in order to obtain more information about the interpolation error, especially in

the interval $[a, b]$. It can already be observed that, if Γ largely surrounds the interval, then the absolute value of the denominator in that quotient becomes large as well, so that the absolute interpolation error is small. We now make this observation more accurate with the help of potential theory.

Let us begin by defining the discrete potential

$$u_n(z) = \frac{-1}{n+1} \sum_{i=0}^{n} \log |z - x_i|,$$

which is simply $-\log(|L(z)|^{1/(n+1)})$. The asymptotic convergence behaviour of polynomial interpolation can be derived after passing from discrete potentials to their continuous analogues, logarithmic potentials. Let the nodes x_i be distributed in $[a, b]$ according to a probability measure μ with support $[a, b]$ and positive piecewise continuous *node density*

$$\phi(x) = \frac{\mathrm{d}\mu}{\mathrm{d}x}(x) > 0 \quad \text{for } x \in [a, b]. \tag{4.2}$$

This statement can be made precise by defining the normalised *node counting measures*

$$\mu_n = \frac{1}{n+1} \sum_{i=0}^{n} \delta_{x_i}, \tag{4.3}$$

where δ_x denotes the Dirac unit measure at x. We require that $\mu_n \to \mu$, in the sense of weak-star convergence of measures, which means that $\int g \, \mathrm{d}\mu_n \to \int g \, \mathrm{d}\mu$ for every continuous function g defined on $[a, b]$. Associated with the limiting measure μ, which we will also refer to as the *node measure*, is a *logarithmic potential*

$$U^{\mu}(z) := -\int_a^b \log |z - x| \, \mathrm{d}\mu(x) = -\int_a^b \phi(x) \log |z - x| \, \mathrm{d}x. \tag{4.4}$$

By the hypothesis on the density of μ, the real-valued function U^{μ} is harmonic in the complex plane $\mathbb{C}$ and decays like $-\log |z|$ as $|z| \to \infty$. The asymptotic convergence or divergence as $n \to \infty$ of a sequence of polynomial interpolants p_n for an analytic function f is then described by the following theorem; see, e.g., [48].

Theorem 4.2. *For a given node measure μ and the associated potential U^μ, let f be analytic inside $\mathcal{C}_s$, the level line of U^μ which passes through a singularity s of f. The polynomial interpolant p_n of f then converges to f inside $\mathcal{C}_s$ and diverges outside, and*

$$\lim_{n\to\infty} |f(z) - p_n(z)|^{1/n} = \exp\bigl(U^\mu(s) - U^\mu(z)\bigr). \tag{4.5}$$

In the special case of equispaced nodes, i.e., if $\phi(x) = 1/(b-a)$, and with the change of variables $x \mapsto 2(x-a)/(b-a) - 1$ and the reasoning from [48, §3.4], the logarithmic potential in (4.4) becomes

$$U^\mu_{\text{eq}}(z) = -\log\Bigl(\frac{b-a}{2e}\Bigr) - \frac{1}{2}\mathrm{Re}\bigl((1-z')\log(1-z') - (-1-z')\log(-1-z')\bigr), \tag{4.6}$$

where $z' = (2z-a-b)/(b-a)$.

Chebyshev points of the second kind are distributed according to the density function $\phi(x) = 1/(\pi\sqrt{1-x^2})$, and the associated logarithmic potential is

$$U^\mu_{\text{Ch}}(z) = -\int_{-1}^{1} \frac{\log|z-x|}{\pi\sqrt{1-x^2}}\,\mathrm{d}x = -\log\Bigl|z + \sqrt{z}\cdot\sqrt{z-z^{-1}}\Bigr| + \log(2), \tag{4.7}$$

which turns out to be constant on the interval $[-1,1]$. The associated node measure μ therefore has the property that there is no potential difference on $[-1,1]$; it is the so-called *equilibrium measure*. Interpolation nodes which are distributed according to this measure are, in an asymptotic sense, optimal for polynomial interpolation.

In Figure 4.1 we have plotted the asymptotic rates of convergence or divergence of polynomial interpolation on $[-1,1]$ with equispaced nodes (on the left) and Chebyshev points of the second kind (on the right) as a function of s. To be more precise, for each s in $(-2,2)\times(-2\mathrm{i},2\mathrm{i})$, we took the maximum over $[-1,1]$ of the expression on the right-hand side of (4.5). The picture on the left shows that polynomial interpolation with equispaced nodes does not converge throughout $[-1,1]$ if f has a singularity s too close to the interval. On the other hand, the picture on the right shows that polynomial interpolation with Chebyshev points converges exponentially throughout the interval.

The Runge Phenomenon

Theorem 4.2 gives an explanation of Runge's phenomenon [42, 92]. Polynomial interpolation with equispaced nodes on $[-1,1]$ for the Runge example $1/(1+$

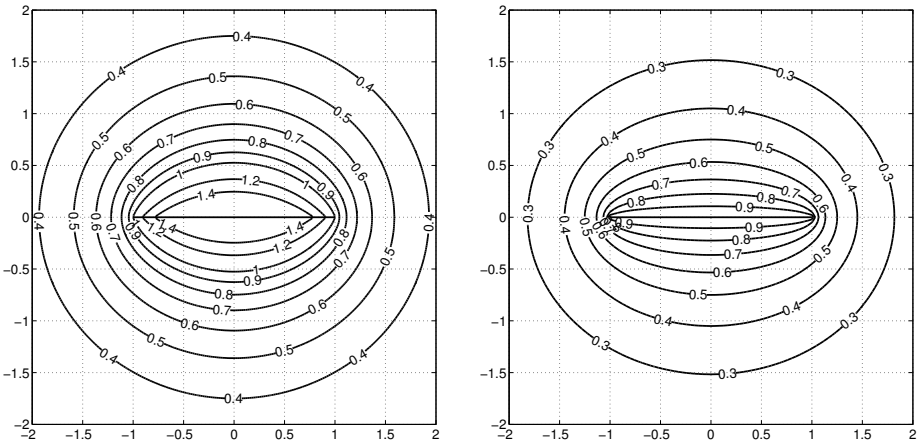

Figure 4.1: Level lines of $\exp\bigl(U^\mu(s) - \min_{-1\le x\le 1} U^\mu(x)\bigr)$ in the complex plane for polynomial interpolation with equispaced nodes (left) and Chebyshev points (right). The central segment represents the interval $[-1, 1]$.

$25x^2$) does not converge on the whole interval. This is explained by the fact that the level line of $U^\mu_{\rm eq}$ passing through the singularities $s = \pm{\rm i}/5$ does not enclose $[-1, 1]$ but only the middle part $[-0.7266, 0.7266]$, as can be seen on the left picture in Figure 4.1.

The so-called Runge phenomenon appears in the form of oscillations toward the ends of the interval, whereas the interpolant converges on the middle part of the interval. This phenomenon is not limited to equispaced nodes and it occurs with every distribution of nodes which has a potential U^μ such that $\exp(U^\mu(s) - \min_{a\le x\le b} U^\mu(x))$ is larger than 1 on level lines which circumscribe a region in the complex plane that includes $[a, b]$ or parts of it. The polynomial interpolant of a function which has singularities in that region then presents a behaviour similar to Runge's phenomenon. For Chebyshev points, the level line of $U^\mu_{\rm Ch}$ that passes through $s = \pm{\rm i}/5$ includes the interval $[-1, 1]$; see the right part of Figure 4.1. The phenomenon therefore does not appear when interpolating a function, such as that given above, between Chebyshev points and other nodes with quadratic clustering toward the ends of the interval. These polynomial interpolants converge geometrically as the number of nodes increases.

4.2 Potential Theory for Floater–Hormann Interpolation

Convergence results for analytic functions, based on potential theory as in Theorem 4.2, have also been derived for other interpolation schemes, such as linear barycentric rational interpolation in mapped Chebyshev points [5], radial basis functions [83, 84], and nonlinear rational interpolants [103, 121, 122]. We present such a theory for the family of barycentric rational interpolants (2.10) in this section. The error bound for differentiable functions from Theorem 2.3 shows that Runge's phenomenon does not occur for d fixed. In exact arithmetic, the rational interpolants converge algebraically to f as n increases, provided the function is sufficiently smooth, even if it has poles very close to $[a, b]$.

The rational interpolants (2.10) are blends of the local polynomial interpolants p_i, and we use this fact in the derivation of an asymptotic upper bound on the interpolation error for analytic functions. We begin with some statements about the node densities of the subsets $\{x_i, \ldots, x_{i+d}\}$, which are the interpolation nodes of p_i. We then develop the main convergence result. Thereafter we are concerned with the important special cases of symmetric nodes and equispaced nodes, for which we further expand the expression of the bound on the interpolation error and give additional insight into the convergence of the interpolants.

4.2.1 General Node Densities

We assume from now on that the parameter d, defining a particular rational interpolant from the family (2.10), is a variable nonnegative integer $d(n)$ such that

$$d(n)/n \to C, \quad n \to \infty, \tag{4.8}$$

for a fixed $C \in (0, 1]$. In practice, one could choose $d(n) = \text{round}(Cn)$. By the positivity of ϕ from (4.2), the *cumulative node distribution*

$$\Phi(x) := \mu([a, x]) = \int_a^x \phi(y)\,\mathrm{d}y$$

is a continuous and strictly monotonically increasing function on the interval $[a, b]$ and therefore has a continuous inverse

$$\Phi^{-1} : [0, 1] \to [a, b].$$

With this definition and that of the node counting measure μ_n from (4.3), we have the following lemma.

Lemma 4.3. *Assume that $\mu_n \to \mu$ as $n \to \infty$, where the limit measure μ has piecewise continuous positive density, and let $x_{j(n)}$ be an arbitrary sequence of nodes. Then $x_{j(n)} \to x$ for some x if and only if there exists some $q \in [0,1]$ such that $(j(n)+1)/(n+1) \to q$. In this case, $x = \Phi^{-1}(q)$.*

Proof. Under the assumptions on μ, the set $\{x\}$ is a continuity set of μ for every point $x \in [a,b]$. The condition that $\mu_n \to \mu$ thus implies that $\mu_n([a,x]) \to \mu([a,x])$. By the definition of μ_n and Φ, this is equivalent to

$$\lim_{n\to\infty} \frac{\#\{x_i : x_i \le x\}}{n+1} = \Phi(x).$$

With $j(n) := \max\{0, \#\{x_i : x_i \le x\} - 1\}$ and $q := \Phi(x)$ we obtain the asserted relation

$$\lim_{n\to\infty} \frac{j(n)+1}{n+1} = q. \qquad \square$$

We now suppose that $j(n)$ is a sequence of indices such that $j(n) \le n - d(n)$ and $x_{j(n)} \to \alpha$ for some $\alpha \in [a,b]$. Under the condition (4.8) on $d(n)$ it follows from Lemma 4.3 that the sequence of nodes $x_{j(n)+d(n)}$ also converges to a point in $[a,b]$, which we call $\beta(\alpha)$, and satisfies

$$\lim_{n\to\infty} x_{j(n)+d(n)} = \Phi^{-1}(C + \Phi(\alpha)) = \beta(\alpha).$$

The nodes $x_{j(n)}, \ldots, x_{j(n)+d(n)}$ are therefore asymptotically contained in the interval $[\alpha, \beta(\alpha)]$, and they are distributed with the density ϕ restricted to that interval. More precisely, the normalised counting measures

$$\nu_{j(n)} = \frac{1}{d(n)+1} \sum_{i=0}^{d(n)} \delta_{x_{j(n)+i}} \tag{4.9}$$

converge (in the weak sense) to a probability measure ν_α with support $[\alpha, \beta(\alpha)]$ and density

$$\frac{\mathrm{d}\nu_\alpha}{\mathrm{d}x}(x) = \frac{\phi(x)}{\Phi(\beta(\alpha)) - \Phi(\alpha)} = \frac{\phi(x)}{C}.$$

Note that ν_α simply is the normalised restriction of μ to the interval $[\alpha, \beta(\alpha)]$. The study of the convergence of measures and their associated potentials is an important tool in logarithmic potential theory [89, 94]. A main ingredient is

the so-called *principle of descent* (see [94, Theorem I.6.8]): If S is compact and $(\sigma_n)_n$ is a sequence of finite positive Borel measures with $\operatorname{supp}(\sigma_n) \subset S$, then

$$\text{for } z_n \to z \text{ and } \sigma_n \to \sigma, \qquad \liminf_{n\to\infty} U^{\sigma_n}(z_n) \geq U^{\sigma}(z).$$

This relation will allow us to take advantage of polynomial approximation theory for quantifying the convergence of the local interpolating polynomials $p_{j(n)}$. The following two lemmas are a first step in this direction as they give asymptotic upper and lower bounds on the rational function

$$\sum_{i=0}^{n-d(n)} \lambda_i(z),$$

which appears in $r_n(z)$ and plays an important role in the subsequent analysis.

Lemma 4.4. *For any $C \in (0,1]$ and $z \in \mathbb{C} \setminus [a,b]$, we have*

$$\limsup_{n\to\infty} \left| \sum_{i=0}^{n-d(n)} \lambda_i(z) \right|^{1/(n+1)} \leq \max_{\alpha\in[a,\Phi^{-1}(1-C)]} \exp\big(CU^{\nu_\alpha}(z)\big).$$

Proof. For every n and every $z \in \mathbb{C} \setminus [a,b]$ we can find a dominating term $\lambda_j(z)$ with index $j = j(n)$ such that

$$\left| \sum_{i=0}^{n-d(n)} \lambda_i(z) \right| \leq (n - d(n) + 1)|\lambda_{j(n)}(z)|. \tag{4.10}$$

With a simple computation involving (4.9), we rewrite the second factor of the above right-hand side:

$$\begin{aligned} |\lambda_{j(n)}(z)| &= \prod_{i=0}^{d(n)} |z - x_{j(n)+i}|^{-1} \\ &= \exp\Big(-\sum_{i=0}^{d(n)} \log|z - x_{j(n)+i}|\Big) = \exp\big((d(n)+1)U^{\nu_{j(n)}}(z)\big). \end{aligned}$$

From the sequence of nodes $x_{j(n)}$ we can select a subsequence, which we also denote by $x_{j(n)}$, and which has the property that $x_{j(n)} \to \alpha$ for $\alpha \in [a,b]$ fixed.

Upon taking the $(n+1)$st root and using the fact that $(d(n)+1)/(n+1) \to C$ as $n \to \infty$, we obtain that

$$\limsup_{n\to\infty} \left| \sum_{i=0}^{n-d(n)} \lambda_i(z) \right|^{1/(n+1)} \leq \limsup_{n\to\infty} \exp\bigl(CU^{\nu_{j(n)}}(z)\bigr).$$

Since $z \notin [a,b]$, it follows from the weak-star convergence $\nu_{j(n)} \to \nu_\alpha$ that the lim sup on the right-hand side is equal to $\exp(CU^{\nu_\alpha}(z))$. □

In the proof of the next lemma we reuse similar tools to derive a lower bound on the same function, this time evaluated at $x \in [a,b]$.

Lemma 4.5. *For any $C \in (0,1]$ and $x \in [a,b]$, we have*

$$\liminf_{n\to\infty} \left| \sum_{i=0}^{n-d(n)} \lambda_i(x) \right|^{1/(n+1)} \geq \max_{\substack{\alpha\in[a,\Phi^{-1}(1-C)] \\ \text{s.t. } x\in\operatorname{supp}(\nu_\alpha)}} \exp\bigl(CU^{\nu_\alpha}(x)\bigr).$$

Proof. We first suppose that $x \in (x_k, x_{k+1})$. For every n and every such x it is shown in the proof of [46, Thm 2] that

$$\left| \sum_{i=0}^{n-d(n)} \lambda_i(x) \right| \geq |\lambda_j(x)|$$

for all $j \in J_k \setminus \{k-d(n)\}$. On the other hand, if x coincides with a node x_i, then this inequality is trivially valid (the left-hand side is $+\infty$). For every n, we choose $j = j(n)$ such that $\lambda_{j(n)}(x)$ is the largest in absolute value, and we form the corresponding sequence $j(n)$. From the sequence of nodes $x_{j(n)}$ we select, as in the proof of Lemma 4.4, a subsequence such that $x_{j(n)} \to \alpha$. Upon taking the $(n+1)$st root on both sides of the above inequality, the principle of descent yields the asserted relation analogously as in the proof of Lemma 4.4. □

We are now prepared to investigate the asymptotic convergence of the family of rational interpolants (2.10). As explained, for instance, in [83, Thm. 3.2 and Corol. 3.4], it will be sufficient to investigate interpolants of "prototype functions" $g(x,s) = 1/(s-x)$ with a simple pole $s \in \mathbb{C} \setminus [a,b]$. An explicit expression for the local polynomial interpolants p_i of such a function is

$$p_i(x) = \frac{1 - \frac{\lambda_i(s)}{\lambda_i(x)}}{s-x}.$$

It is easy to verify that this indeed is a polynomial, numerator and denominator being polynomials that vanish at $x = s$, and that it satisfies $p_i(x_j) = 1/(s - x_j)$ for all nodes x_j involved in λ_i. Hence, the rational interpolant (2.10) of g, which we denote by $r_n[g]$, is

$$r_n[g](x) = \frac{1}{s-x} \cdot \frac{\sum_{i=0}^{n-d} \lambda_i(x)\big(1 - \frac{\lambda_i(s)}{\lambda_i(x)}\big)}{\sum_{i=0}^{n-d} \lambda_i(x)} = \frac{1}{s-x} \cdot \left(1 - \frac{\sum_{i=0}^{n-d} \lambda_i(s)}{\sum_{i=0}^{n-d} \lambda_i(x)}\right), \quad (4.11)$$

and therefore

$$g(x,s) - r_n[g](x) = \frac{1}{s-x} \cdot \frac{\sum_{i=0}^{n-d} \lambda_i(s)}{\sum_{i=0}^{n-d} \lambda_i(x)}. \quad (4.12)$$

To make the subsequent notation more compact, we define the "potential function"

$$V^{C,\mu}(z) := \begin{cases} \max_{\alpha \in [a, \Phi^{-1}(1-C)]} CU^{\nu_\alpha}(z), & z \in \mathbb{C} \setminus [a,b], \\ \max\limits_{\substack{\alpha \in [a, \Phi^{-1}(1-C)] \\ \text{s.t. } z \in \operatorname{supp}(\nu_\alpha)}} CU^{\nu_\alpha}(z), & z \in [a,b]. \end{cases} \quad (4.13)$$

Combining Lemmas 4.4 and 4.5 and using the monotonicity of the exponential function (note that the potentials U^{ν_α} are real-valued functions), we finally arrive at

$$\limsup_{n \to \infty} |g(x,s) - r_n[g](x)|^{1/n} \leq \exp\big(V^{C,\mu}(s) - V^{C,\mu}(x)\big).$$

This statement closely resembles that from Theorem 4.2 and contains the latter as a special case: The rational interpolants reduce (asymptotically) to polynomial interpolants if $C = 1$, and in this case the function $V^{C,\mu}$ in (4.13) reduces to U^μ. Moreover, the potential $V^{C,\mu}(z)$ is a continuous function when the condition "$z \in \operatorname{supp}(\nu_\alpha)$" in the second case of (4.13) is redundant, which is the case for any reasonable node measure μ.

The uniform convergence over the *whole* interval $[a, b]$ is often of major interest in approximation theory. To establish such a result for the rational interpolants (2.10), we define the contours

$$\mathcal{C}_R := \left\{ z \in \mathbb{C} : \frac{\exp\big(V^{C,\mu}(z)\big)}{\min_{x \in [a,b]} \exp\big(V^{C,\mu}(x)\big)} = R \right\}, \quad (4.14)$$

which can be seen as levels of "worst-case" convergence with rate at least R for every point $x \in [a, b]$.

If an arbitrary function f is analytic inside a simple, closed, and rectifiable curve $\mathcal{C}$ contained in a closed simply connected region around the nodes, it can be represented by the Cauchy integral formula

$$f(x) = \frac{1}{2\pi \mathrm{i}} \int_{\mathcal{C}} \frac{f(s)}{s-x}\,\mathrm{d}s = \frac{1}{2\pi \mathrm{i}} \int_{\mathcal{C}} f(s) g(x,s)\,\mathrm{d}s.$$

The representation (4.11) and the linearity of the rational interpolants imply that

$$r_n(x) = \frac{1}{2\pi \mathrm{i}} \int_{\mathcal{C}} \frac{f(s)}{s-x} \cdot \left(1 - \frac{\sum_{i=0}^{n-d} \lambda_i(s)}{\sum_{i=0}^{n-d} \lambda_i(x)}\right) \mathrm{d}s$$

is the rational interpolant to f. The interpolation error therefore is

$$f(x) - r_n(x) = \frac{1}{2\pi \mathrm{i}} \int_{\mathcal{C}} \frac{f(s)}{s-x} \cdot \frac{\sum_{i=0}^{n-d} \lambda_i(s)}{\sum_{i=0}^{n-d} \lambda_i(x)}\,\mathrm{d}s,$$

which is a *Hermite-type error formula* [31] for the family of rational interpolants (2.10). Finally,

$$\|f - r_n\| \leq D \frac{\sup_{s\in\mathcal{C}} \left|\sum_{i=0}^{n-d} \lambda_i(s)\right|}{\min_{x\in[a,b]} \left|\sum_{i=0}^{n-d} \lambda_i(x)\right|},$$

where $D = \frac{\mathrm{length}(\mathcal{C}) \max_{s\in\mathcal{C}} |f(s)|}{2\pi\, \mathrm{dist}([a,b],\mathcal{C})}$ is a constant independent of n. We summarise the above expansion in the following theorem.

Theorem 4.6. *Let f be a function analytic in an open neighbourhood of $[a,b]$, and let $R > 0$ be the smallest number such that f is analytic in the interior of $\mathcal{C}_R$ defined in* (4.14)*. Then the rational interpolants r_n defined by* (2.10)*, with limiting node measure μ and $d(n)/n \to C$, satisfy*

$$\limsup_{n\to\infty} \|f - r_n\|^{1/n} \leq R.$$

In Figure 4.2 we illustrate the level lines $\mathcal{C}_R$ for the parameter value $C = 0.2$ with equispaced nodes on the left and with nodes distributed according to the density $\mathrm{d}\mu/\,\mathrm{d}x = \phi(x) = (4+\arctan(4x))/8$ on the right. The interval is $[-1,1]$. In Section 4.2.3, we shall give more details on how we computed these level lines in the case of equispaced nodes; for arbitrary node densities we integrated the potential function $V^{C,\mu}$ numerically. For equispaced nodes, a line corresponding to $R = 1$ appears close to the interval, which means that r_n may not converge

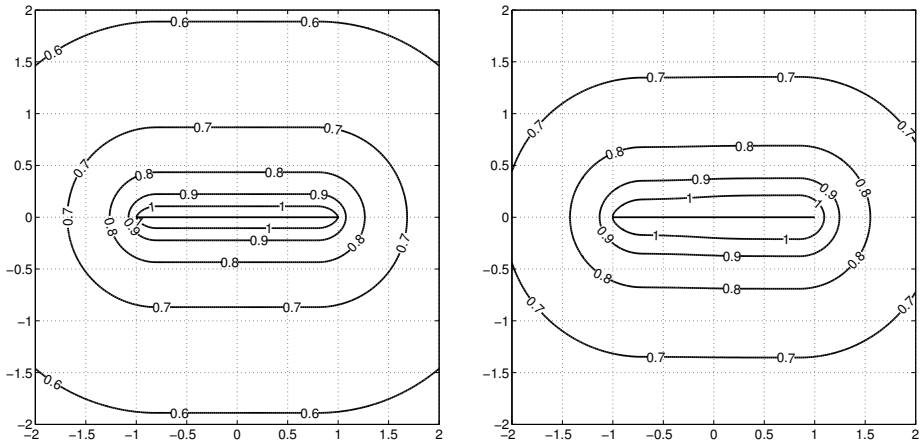

Figure 4.2: Level lines of convergence for barycentric rational interpolation for $C = 0.2$ with equispaced nodes (left) and nodes distributed according to the density $\phi(x) = (4 + \arctan(4x))/8$ (right) on $[-1, 1]$.

throughout $[a, b]$ to a function f with a singularity s in the interior of that curve. Runge's phenomenon is therefore likely to appear for such a function if d increases with n. An *intuitive* explanation of why there is no Runge phenomenon with d fixed goes as follows: The interpolation error may be written as

$$f(x) - r_n(x) = \frac{\sum_{i=0}^{n-d} \lambda_i(x)\big(f(x) - p_i(x)\big)}{\sum_{i=0}^{n-d} \lambda_i(x)}$$

(see [46]) and is a blend of an increasing number of polynomial interpolation errors as n increases. The polynomials p_i have fixed maximal degree d and interpolate f in subintervals $[x_i, x_{i+d}]$ of decreasing length, so that the region where f needs to be analytic shrinks.

Theorem 4.6 gives only an asymptotic *upper bound* on the rate of convergence of $r_n \to f$ as $n \to \infty$, as opposed to the polynomial case, where equality holds; see Theorem 4.2. This means that, unlike in the case with polynomial interpolation, we generally cannot infer the level line $\mathcal{C}_R$ with the closest singularity of f from a known or observed approximation rate R. This is not a problem of the above derivation, but an intrinsic property of barycentric rational interpolation with alternating signs in the blending functions λ_i, as we explain below.

4.2.2 Symmetric Nodes

One reason why convergence may be faster than predicted by the asymptotic upper bound from Theorem 4.6 is cancellation in the terms of the error representation (4.12). As an illustration, we assume that the nodes x_i are pairwise symmetric with respect to the midpoint $(a+b)/2$ of the interval, i.e.,

$$\frac{x_i + x_{n-i}}{2} = \frac{a+b}{2}, \quad i = 0, 1, \ldots, n.$$

For these nodes, the upper bound in Lemma 4.4 may be crude for points z relatively close to the interval and with $\mathrm{Re}(z) \in [a, b]$. This situation is illustrated in Figure 4.3, which shows the level lines of $|\sum_{i=0}^{n-d} \lambda_i(z)|^{1/(n+1)}$ for equispaced nodes on $[-1, 1]$ with $n = 100$ and $d = 9$ in the top left picture, and the levels of the asymptotic upper bound $\exp(V^\mu(z))$. The level lines agree well if $\mathrm{Re}(z)$ is outside $[-1, 1]$. The reason for the observed discrepancy above and below the midpoint of the interval is the fact that some terms in the sum

$$\sum_{i=0}^{n-d(n)} \lambda_i(z) = \sum_{i=0}^{n-d(n)} \frac{(-1)^i}{(z - x_i) \cdots (z - x_{i+d(n)})}$$

cancel mutually for certain values of z, or may at least be reduced. For example, the absolute value of the sum of the ℓth and $(n - d(n) - \ell)$th terms, evaluated at a point z with $\mathrm{Re}(z) = (a+b)/2$, yields after a short computation

$$\begin{aligned} |\lambda_\ell(z) + \lambda_{n-d(n)-\ell}(z)| &= |\lambda_\ell(z) + (-1)^{n+1}\overline{\lambda_\ell(z)}| \\ &= \begin{cases} 2|\mathrm{Re}(\lambda_\ell(z))| & \text{if } n \text{ is odd,} \\ 2|\mathrm{Im}(\lambda_\ell(z))| & \text{if } n \text{ is even.} \end{cases} \end{aligned} \tag{4.15}$$

This simplification occurs for all $0 \le \ell \le \lfloor (n-d)/2 \rfloor$ and obviously reduces the interpolation error at these particular points z, causing the cusp in the level curves.

However, if we slightly perturb the equispaced grid, leading to *almost equispaced nodes*, then the levels of $|\sum_{i=0}^{n-d} \lambda_i(z)|^{1/(n+1)}$ look much more similar to the predicted level curves $\exp(V^\mu(z))$; see the bottom left picture in Figure 4.3. The presented convergence theory is asymptotic in the sense that we only require convergence $\mu_n \to \mu$ of the node measures in a weak sense. Not necessarily all the μ_n need to have perfectly symmetric mass points in order to satisfy this condition. It is therefore not reasonable to expect that our theory will capture

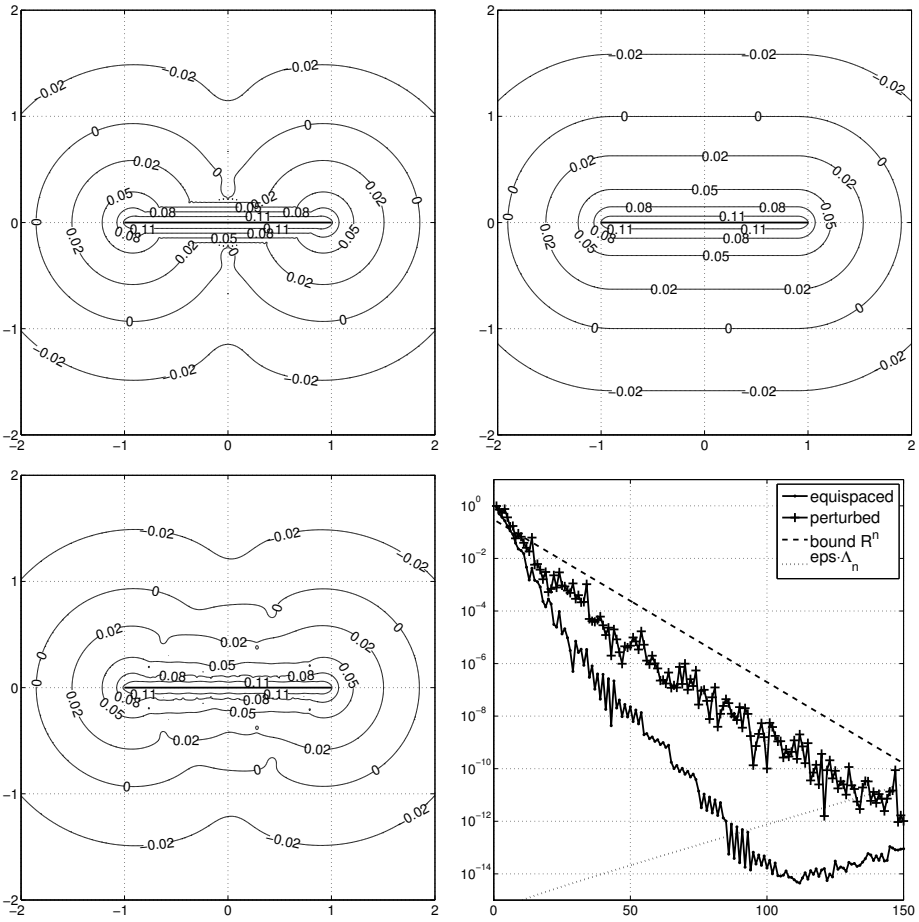

Figure 4.3: Top: Level lines of $|\sum_{i=0}^{n-d} \lambda_i(z)|^{1/(n+1)}$ (left) with $d = 9$ for $n = 100$ equispaced nodes in $[-1, 1]$ (solid central line) and level lines of $\exp(V^\mu(z))$ (right) on a $\log_{10}$ scale. Bottom: Level lines for almost equispaced nodes (left). Relative error curve for the interpolation of $f(x) = 1/(x - 0.3\mathrm{i})$ with $C = 0.1$ and both node sequences, asymptotic relative error bound, and upper bound on $\texttt{eps}\cdot\Lambda_n$ (right).

the error reduction effects appearing with symmetric nodes. The difference between the levels of $|\sum_{i=0}^{n-d}\lambda_i(z)|$ for equispaced nodes and almost equispaced nodes also becomes apparent from the error curves for the interpolation of a function with a singularity inside the cusp region, say, $f(x) = 1/(x-0.3\mathrm{i})$. This is illustrated in the bottom right picture in Figure 4.3. The error curves for the interpolation of f for $1 \le n \le 150$ with $C = 0.1$ and both node sequences indicate that the symmetry in equispaced nodes results in a convergence acceleration as compared to the predicted rate. The oscillating behaviour of the curve corresponding to interpolation with equispaced nodes is caused by different reductions of $|\sum_{i=0}^{n-d}\lambda_i(z)|$ taking place for even and odd values of n; see (4.15).

4.2.3 Equispaced Nodes

In the case of equispaced nodes x_i, distributed according to the density $\phi(x) = 1/(b-a)$ on $[a,b]$, we can give a more explicit statement of the rates of convergence. First, the subintervals of the local polynomial interpolants stay of constant length $C(b-a)$ as α varies in $[a, b-C(b-a)]$:

$$[\alpha, \beta(\alpha)] = [\alpha, \alpha + C(b-a)].$$

A formula for the potential of ν_α is given explicitly from (4.6):

$$\begin{aligned} U^{\nu_\alpha}(z) &= -\int_\alpha^{\beta(\alpha)} \frac{1}{\beta(\alpha)-\alpha}\log|z-x|\,\mathrm{d}x \\ &= -\log\Big(\frac{C(b-a)}{2e}\Big) \\ &\quad -\frac{1}{2}\mathrm{Re}\big((1-z')\log(1-z') - (-1-z')\log(-1-z')\big) \\ &=: U_0 + U_1^{\nu_\alpha}(z), \end{aligned} \tag{4.16}$$

where $z' = (2z-2\alpha)/(Cb-Ca)-1$ and U_0 and $U_1^{\nu_\alpha}(z)$ are defined in the obvious way, i.e., such that $U_0 = -\log(C(b-a)/(2e))$. Furthermore, the right-hand side of the inequality in Lemma 4.5 can be easily bounded from below for $x \in [a,b]$: Since on the real line all the U^{ν_α} are concave and symmetric with respect to the midpoint of the subinterval $[\alpha, \alpha + C(b-a)]$ and are simply translates of each other when α is varied, the minimum

$$\min_{x\in[a,b]} \exp\big(V^{C,\mu}(x)\big) = \min_{x\in[a,b]} \max_{\alpha\in[a,\Phi^{-1}(1-C)]} \exp\big(CU^{\nu_\alpha}(x)\big)$$

is obtained either for $x = a$ and $\alpha = a$ or by symmetry for $x = b$ and $\alpha = \Phi^{-1}(1 - C)$. Choosing the former pair of parameters and with $x' = -1$, it follows that

$$U^{\nu_a}(a) = -\log\Big(\frac{C(b-a)}{2e}\Big) - \log(2) = \log\Big(\frac{e}{C(b-a)}\Big),$$

so that

$$\min_{x\in[a,b]} \exp\big(V^{C,\mu}(x)\big) = \Big(\frac{e}{C(b-a)}\Big)^C = 2^{-C}\exp(CU_0). \tag{4.17}$$

With

$$V_1^{C,\mu}(z) = \max_{\alpha\in[a,\Phi^{-1}(1-C)]} CU_1^{\nu_\alpha}(z) \tag{4.18}$$

and equation (4.16), the contours can thus be given more explicitly as

$$\mathcal{C}_R = \big\{z \in \mathbb{C} : 2^C \exp\big(V_1^{C,\mu}(z)\big) = R\big\}.$$

Finally it can be seen from (4.10) in the proof of Lemma 4.4 for the special case of equispaced nodes, that the maximum in (4.18) is attained for $\alpha = \max\{a, \mathrm{Re}(s) - C(b-a)/2\}$, with s, the singularity of f closest to $[a, b]$ on $\mathcal{C}_R$.

Remark. The lower bound

$$\min_{x\in[a,b]} \left|\sum_{i=0}^{n-d} \lambda_i(x)\right| \geq \left(\Big(\frac{b-a}{n}\Big)^{d+1} d!\right)^{-1}$$

was derived in the proof of [46, Thm. 2], here for the case of equispaced nodes, and is valid for any set of nodes after the factor $(b-a)/n$ is replaced by h. This lower bound asymptotically coincides with (4.17): Using Stirling's approximation

$$d! \sim \sqrt{2\pi d}\Big(\frac{d}{e}\Big)^d,$$

we obtain, with $d = Cn$ and upon taking the nth root, that

$$\left(\Big(\frac{b-a}{n}\Big)^{d+1} d!\right)^{-1/n} \sim \left(\Big(\frac{b-a}{n}\Big)^{Cn+1} \sqrt{2\pi Cn}\Big(\frac{Cn}{e}\Big)^{Cn}\right)^{-1/n} \sim \Big(\frac{e}{C(b-a)}\Big)^C$$

as $n \to \infty$.

4.2.4 Stabilisation

In the literature there are only few recommendations for how to choose the parameter d in r_n. As suggested in [88], in practice, d is typically chosen as a fixed small integer and n is successively increased until the desired accuracy is achieved. Based on the asymptotic convergence theory from Section 4.2.1, we may give a different recommendation for how to choose d if some information on f is available—for example, its region of analyticity. For simplicity of exposition, we will focus on equispaced nodes, but the same reasoning applies to arbitrary nonconstant node density functions ϕ.

For an arbitrary distribution of interpolation nodes and $f \in C^{d+2}[a,b]$, the interpolation error decreases like $O(h^{d+1})$, as explained in Chapter 2. Large values of d should therefore lead to fast convergence as long as the function f is sufficiently smooth. However, rounding errors and their amplification during the interpolation process also come into play as we explained in Section 3.3.3. We thus need to address the growth of the condition number, i.e., the Lebesgue constant Λ_n, which increases exponentially with d at least for equispaced nodes; see (3.15). When d grows too rapidly with n, the rounding errors might be so strongly amplified that the approximation error increases again after a certain accuracy has been reached for relatively small n.

To see how the interpolation error might behave in the presence of rounding of the data, we suppose that a relative perturbation $f_i\varepsilon_i$ is added to every function value f_i, where every $|\varepsilon_i|$ is less than or equal to some positive ε. The rational interpolants of these perturbed values will be denoted by $\widetilde{r}_n$. As the interpolants are linear in the data, the error can be estimated as

$$\begin{aligned}
\|f-\widetilde{r}_n\| &= \max_{a\le x\le b}\left| f(x) - \frac{\sum_{i=0}^n \frac{w_i}{x-x_i}(f_i+f_i\varepsilon_i)}{\sum_{i=0}^n \frac{w_i}{x-x_i}}\right| \\
&\le \|f-r_n\| + \varepsilon \max_{a\le x\le b} \frac{\sum_{i=0}^n \frac{|w_i|}{|x-x_i|}|f_i|}{\left|\sum_{i=0}^n \frac{w_i}{x-x_i}\right|} \\
&\le \|f-r_n\| + \varepsilon\|f\|\Lambda_n.
\end{aligned}$$

The numerical error is thus governed by two terms—the theoretical error from exact arithmetic and the amplification of the rounding error. If f satisfies the hypotheses of Theorem 4.6, then for n *large enough*,

$$\|f-\widetilde{r}_n\| \lessapprox DR^n + \varepsilon\|f\|\Lambda_n.$$

In IEEE double precision, $\varepsilon = \texttt{eps} = 2^{-52} \approx 2.22\cdot 10^{-16}$.

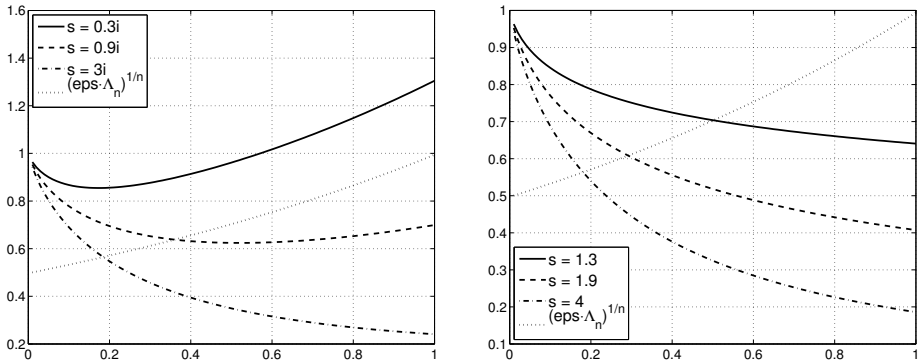

Figure 4.4: Convergence rates R for C varying in $(0, 1]$ and various s on the upper imaginary axis (left), and on the positive real line outside the interval $[-1, 1]$ (right); in both pictures, the dotted line shows the upper bound on the nth root of the amplification of rounding errors for $n = 50$.

With the choice $d = \text{round}(Cn)$ for a fixed $C \in (0, 1]$, the upper bound (3.15) on Λ_n grows at least like 2^{Cn}. This indicates that simultaneously large C and n are prohibitive, which equivalently corresponds to choosing a large parameter d for r_n. The convergence rate R in Theorem 4.6 also depends on C—however, not monotonically for every admissible singularity s. The solid, dashed, and dash-dot lines in Figure 4.4 illustrate the behaviour of R for $C \in (0, 1]$ and various values of s on the upper imaginary axis in the left picture, and positive values outside the interval $[-1, 1]$ on the right. The dotted lines show the nth root of the upper bound (3.15) on the Lebesgue constants multiplied by ε. The maximum of the dotted line and that corresponding to a convergence rate gives a good approximation for $(R^n + \varepsilon\Lambda_n)^{1/n}$, which can be interpreted as the observed convergence or divergence rate. We set both factors D and $\|f\|$ to 1 since we focus on relative errors. One can see that R monotonically decreases with increasing C when s is a real number, so that in this case it would be attractive to choose C as large as possible in exact arithmetic. On the other hand, if s is on the imaginary axis, then choosing too large a C might result in a large R as well; see also Figures 4.5 to 4.10.

Let us now investigate what a good choice of C would be if we want to find a compromise between fast convergence and a reasonably small condition number. From [85] we know that no interpolation method for equispaced nodes exists that converges geometrically and whose condition number does not in-

crease geometrically. Our aim thus is the determination of an optimal value of C for a given singularity s and number of nodes n. This C should lead to an interpolation error that is nearly as small as possible and that is not dominated by the amplification of errors in the interpolated data. The suggestion we presented in [57] is to minimise an upper bound on the numerical error, which is a superposition of geometric convergence or divergence of the interpolant in exact arithmetic and the amplification of rounding errors:

$$\begin{aligned}\text{observed error}(C,n) \approx\ & \text{interpolation error in exact arithmetic}\\ & + \text{imprecision} \times \text{condition number}\\ \lessapprox\ & D\Big(\exp\big(V^{C,\mu}(s) - C\big)\big(C(b-a)\big)^C\Big)^n\\ & + \texttt{eps}\cdot 2^{Cn-1}\big(2+\log(n)\big)\|f\|\\ =:\ & \text{predicted error}(C,n).\end{aligned}$$

Similar reasoning had been followed by the authors of [116]. We propose determining $C \in (0,1]$ such that the predicted error is nearly minimal. We performed this minimisation for the functions $f_1(x) = 1/(x-1.5)$ and $f_2(x) = 1/(x-0.3\mathrm{i})$ and display the results in Figure 4.5 together with the values of C chosen by the minimisation process for each n. Observe that we are minimising only an approximate upper bound on the observed error; the convergence of the interpolation process can be faster than predicted, for instance, because of symmetry effects, as described in Section 4.2.2, or other favourable simplifications in the error term. This is also the reason for the nonmonotone error curve obtained with the interpolation of f_2. One can expect only that the observed error curve will stay (up to a constant factor) below the predicted slope. We observed that this nonmonotone behaviour disappears with almost equispaced nodes, or if one heuristically chooses $D < 1$ in order to give more importance to the term with the Lebesgue constant. The rightmost picture illustrates the choice of d from the minimisation process. For small n, d may be increased quickly, but then it needs to be decreased again in order to maintain the attained accuracy and avoid the growth of the condition number.

Remark[1]. Our asymptotic upper bound on the interpolation error may be crude due to symmetry effects. However, we observe that the convergence is typically *geometric* with a rate $\widetilde{R} \le R$ if $d = \text{round}(Cn)$, that is, $\text{error}(C,n) \approx K\widetilde{R}^n$ with some constant K. In many applications, the closest singularity s of f, in terms of level lines, is not known or difficult to determine. In such

[1] This remark is entirely due to Stefan Güttel.

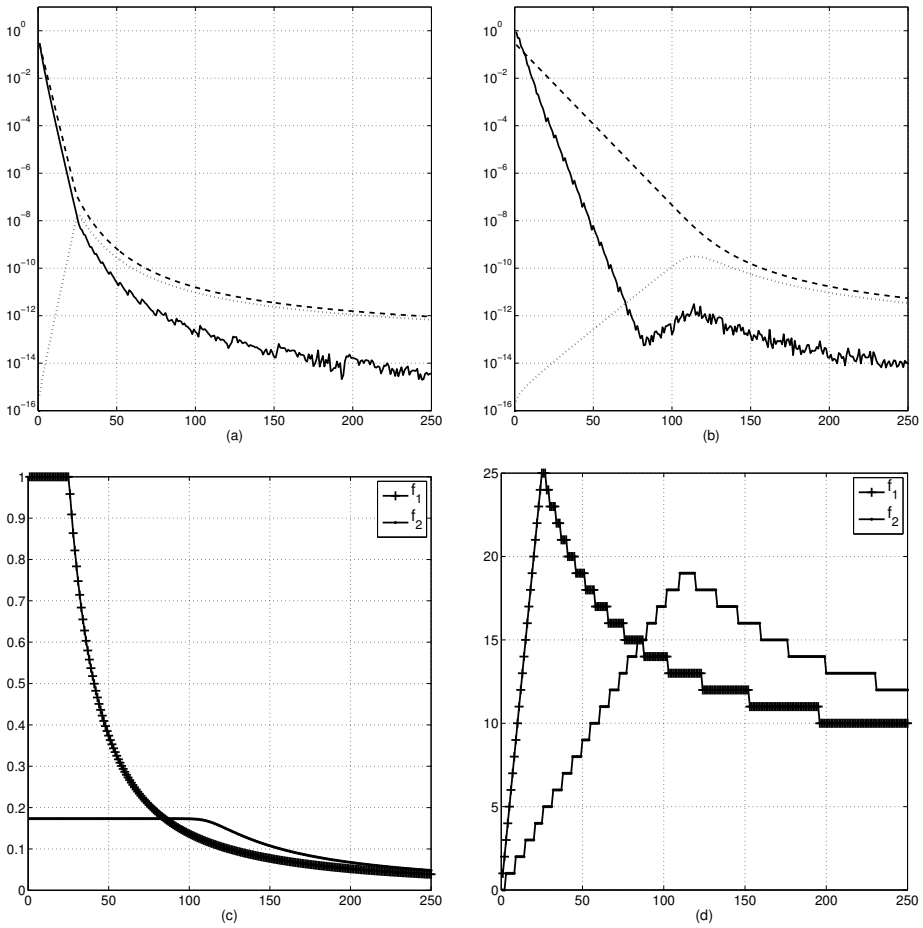

Figure 4.5: Relative errors (solid line) together with predicted relative error slope (dashed line) and upper bound on $\texttt{eps}\cdot\Lambda_n$ (dotted line) for the functions $f_1(x) = 1/(x-1.5)$ in the picture labelled (a), and $f_2(x) = 1/(x-0.3\mathrm{i})$ (b) after choosing, for each $n \in \{1, \ldots, 250\}$, the value of C such that the predicted error slope is nearly minimal (c) and interpolating with $d = \mathrm{round}(Cn)$ (d).

situations, interpolation is usually done for an increasing number n of nodes until the approximation is accurate enough. The indication that convergence is geometric allows us to establish a heuristic method for the estimation of a value of C for which the approximation error, $\mathrm{error}(C,n)$, is below some prescribed relative tolerance *reltol* and n is as small as possible. To this end, we assume that we have an estimator $\mathrm{esterr}(C,n)$ for the error of the rational interpolant with n nodes and parameter $d = \mathrm{round}(Cn)$. Such an estimator can be obtained, e.g., from the evaluation of $|f(x) - r_n(x)|$ at sufficiently many points in the interval. After choosing moderate numbers n_1 and n_2 of nodes, say, $n_1 = 10$ and $n_2 = 40$, and computing $\mathrm{esterr}(C,n_1)$ and $\mathrm{esterr}(C,n_2)$, these estimators can be used to calculate the observed convergence or divergence rate as

$$\widetilde{R} \approx \left(\frac{\mathrm{esterr}(C,n_2)}{\mathrm{esterr}(C,n_1)}\right)^{1/(n_2-n_1)}.$$

Slightly more sophisticated and robust ways of calculating $\widetilde{R}$ could certainly be derived, e.g., by taking into account more than just two values of n. Under the assumption that convergence is indeed geometric, in order to find a smallest possible n, we are interested in minimising the rate $\widetilde{R}$ among all $C \in (0,1]$ under the constraint that the error contribution of the growing Lebesgue constant stays below *reltol*. In the following we sketch a simple golden-section search [88, Sec. 10.2] for locally optimal C and n:

1. Set $C_1 = 0$ and $C_4 = 1$.
2. Set $C_2 = \phi C_1 + (1-\phi)C_4$ and $C_3 = (1-\phi)C_1 + \phi C_4$, where $\phi = \frac{\sqrt{5}-1}{2}$.
3. Compute estimates $\mathrm{esterr}(C_j,n_1)$ and $\mathrm{esterr}(C_j,n_2)$ of the interpolation error for $j = 1,\ldots,4$ (or reuse previously computed values).
4. Compute an estimate for the slope of convergence as
$$\widetilde{R}_j = \left(\frac{\mathrm{esterr}(C_j,n_2)}{\mathrm{esterr}(C_j,n_1)}\right)^{1/(n_2-n_1)}, \quad j = 1,\ldots,4.$$
5. Compute critical values $\overline{n}_j$ such that
$$\widetilde{R}_j^{\overline{n}_j} = 2^{C_j\overline{n}_j - 1}(2 + \log \overline{n}_j)\mathtt{eps}, \quad j = 1,\ldots,4.$$
6. If $\log(\mathit{reltol})/\log(\widetilde{R}_j) > \overline{n}_j$ for some j, set $R_j = 1 + C_j$ (the desired accuracy is not attainable with this value of C_j, and hence R_j is interpreted as divergence).

7. If $\widetilde{R}_2 \geq \widetilde{R}_3$, then set $C_1 = C_2$; else set $C_4 = C_3$.

8. If $C_4 - C_1$ is larger than some tolerance (e.g., 0.01), go to Step 2.

9. Use C_4 as an approximation to the optimal C, and $n = \log(\mathit{reltol})/\log(\widetilde{R}_4)$.

4.2.5 Numerical Experiments

In this section we demonstrate the convergence result from Theorem 4.6 and the approach for the stabilisation of the rational interpolants presented in Section 4.2.4. We sampled in $1 \leq n \leq 250$ points from $[-1, 1]$ various functions whose regions of analyticity are known, and computed the relative errors in their rational interpolants. Moreover, we computed the relative error for all admissible values of d with every n and kept the smallest error together with the corresponding d. For every example, we display, in addition to the relative interpolation error, the chosen values of C and d in the smaller pictures.

The first example, the interpolation of $f(x) = \log(1.2-x)/(x^2+2)$, illustrates the behaviour of the relative interpolation error for C fixed, namely 0.01, 0.03, and 0.1; see Figure 4.6 (left). The steps in the error curves are caused by the restriction that d must be an integer and every step corresponds to a change of d, taken as $d = \mathrm{round}(Cn)$. The curve labelled *var C* is obtained by choosing C and d according to the minimisation process, whose values are shown for each n in the pictures on the right. For $n \approx 150$, the value $C = 0.1$ is optimal, and the corresponding curves intersect. The curve corresponding to constant C increases again because of the bad conditioning with d too large.

Figure 4.7 shows that for the interpolation of $f(x) = \arctan(\pi x)$ the error behaviour is similar to that of $f_2(x) = 1/(x - 0.3\mathrm{i})$ in Figure 4.5. Since the poles $\pm\mathrm{i}/\pi$ are too close to the interval, polynomial interpolation of this function suffers from Runge's phenomenon. We observed that rational interpolation with a fixed $d = 6$ converges rather slowly and a relative error of 10^{-12} is attained with as many as $n = 100$ nodes, whereas with the help of our minimisation process, the latter error is already reached for $n = 50$. For small n, our method chose near-optimal values of d leading to minimal errors. However, with n larger than 75, this is not true any longer. This is due to the fact that the level lines of $|\sum_{i=0}^{n-d} \lambda_i(z)|^{1/(n+1)}$ with equispaced nodes yield cusps in the region where f has poles; see again the top left picture in Figure 4.3. The speed of convergence is thus faster than predicted, and the slope of the predicted error is less steep than that of the experimental error, so that the minimisation procedure does not take the growth of the Lebesgue constant into account, which leads to a

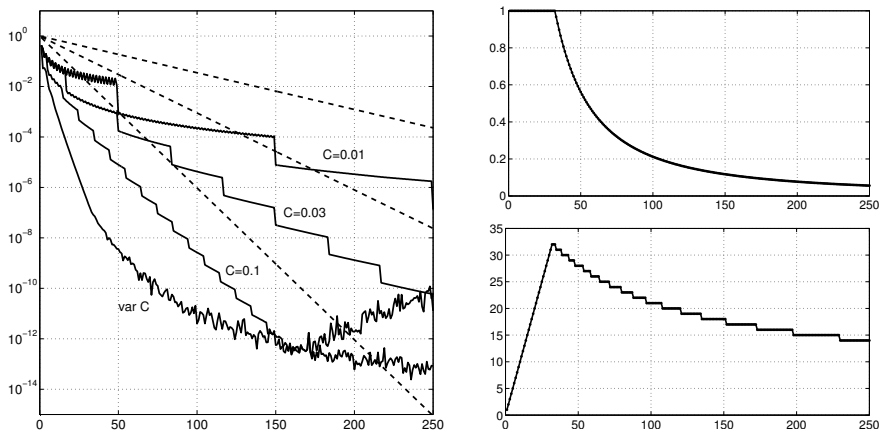

Figure 4.6: Relative errors for the interpolation of $f(x) = \log(1.2 - x)/(x^2 + 2)$.

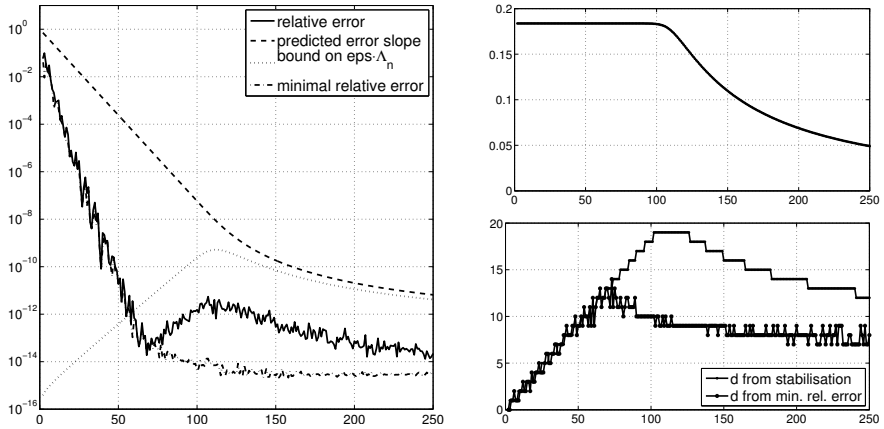

Figure 4.7: Relative errors for the interpolation of $f(x) = \arctan(\pi x)$.

recommendation of slightly too large values of d compared to optimal values. This is, however, not a contradiction to our theory: The error decreased faster than predicted and therefore suffers earlier from the amplification for rounding errors measured by the Lebesgue constant. If we considerably increase the contribution of the Lebesgue constant in our minimisation algorithm, then such a hump in the error curve does not show up, and the values chosen for d are almost equal to those giving minimal errors. This strategy might be adopted when it is known that a function has a singularity exactly in the cusp region of the level lines describing the speed of convergence.

We investigated the same example with quasi-equispaced nodes and global mesh ratio $M = 3$ and we estimated the error using the bound on the Lebesgue constant associated with this kind of nodes. The resulting error follows the predicted error curve more closely. We omit the corresponding plots. We repeated a similar computation with numerically estimated Lebesgue constants, which again gave very similar results. Such a procedure might be used for nodes with unknown Lebesgue constants or bounds thereof.

We also applied the minimisation process to the Runge example $1/(1+25x^2)$ and observed error curves such as those in Figure 4.7—however, with a slightly slower convergence; a relative error of about 10^{-14} was observed for $n \approx 125$. This was to be expected since the poles are closer to the interval. The chosen values for C did not exceed 0.12, so that d increased very slowly and decreased for n larger than 200.

A function that is similar to Runge's example is $f(x) = (1+x)/(1+\sin(x)^2)$, which we approximated on $[-5, 5]$. The closest poles to the interval are at $s = \pm\operatorname{arcsinh}(1)\mathrm{i} \approx \pm 0.8814\mathrm{i}$. Due to symmetry effects, the error decreases faster than predicted, increases again following the growth of the condition number and finally decreases till 10^{-15} with larger n and that relative error remains small even with much larger values of n; see Figure 4.8.

The relative error in the interpolation of $f(x) = \Gamma(x + 1.1)$ is plotted in Figure 4.9. This function has a singularity at $s = -1.1$, which is very close to the left endpoint of the interval. The convergence therefore is rather slow. In the minimisation process, d can thus be chosen larger, as effects of the growing Lebesgue constants become apparent only with larger n. Therefore, the values for d chosen by our method are very close to those giving the smallest relative errors. The estimate of the condition number is slightly too large since we manipulate only upper bounds on the Lebesgue constants, which are not always very tight, and the Lebesgue constant itself is already an upper bound on the condition number.

Although the function $f(x) = \sin(x)$ is entire, we may arbitrarily take $s = 10$

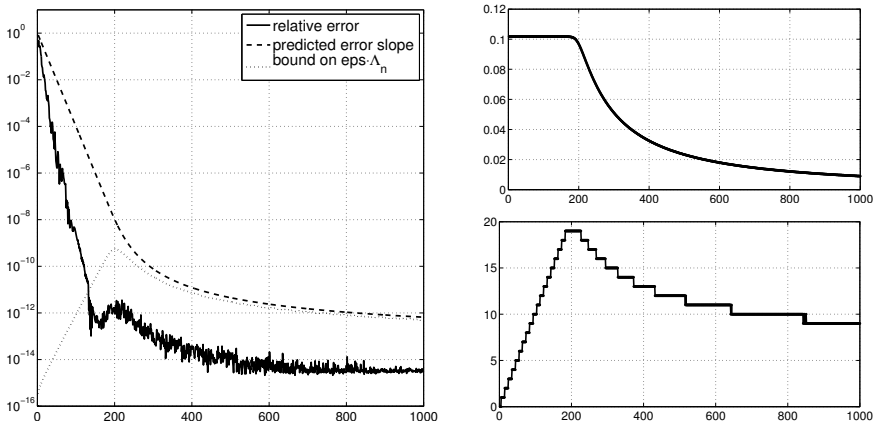

Figure 4.8: Relative errors for the interpolation of $f(x) = (1+x)/(1+\sin(x)^2)$ on $[-5,5]$ with $2 \le n \le 1000$.

in our algorithm, which is sufficiently far away from the interval to guarantee fast convergence. We changed the interval to $[-5,5]$ for this example in order to compare the present results with those obtained in [46, Tab. 1] for $d = 4$ constant. The authors of the latter paper tabulated absolute errors, equivalent to relative errors in this example, on the order of 10^{-12} for $n > 600$. Figure 4.10 shows that an error of the same magnitude is attained for $n \approx 30$ already with the choice of d of our minimisation process. The chosen values for d are smaller than in most of the previous examples. For this example we plotted the error for $2 \le n \le 1000$ to show that it remains close to machine precision even for large values of n. With n in the tens of thousands the error did not increase significantly either. We also observed that for each n, the value of d that gives the smallest error is very close to the value we determined with the stabilisation process, so that the errors displayed are almost smallest possible.

The next demonstration concerns the heuristic choice of "optimal" parameters n and d without having any information on the function f, as proposed in the remark of Section 4.2.4. For all functions in this section we have run this algorithm with a targeted relative error tolerance of 10^{-6} and 10^{-9}, respectively. The convergence slope was estimated using interpolants with $n_1 = 10$ and $n_2 = 40$ points. The results are shown in Table 4.1. They indicate that this approach can give quite good results, although a careful user would always revalidate the accuracy of the obtained interpolant r_n by comparing f and r_n

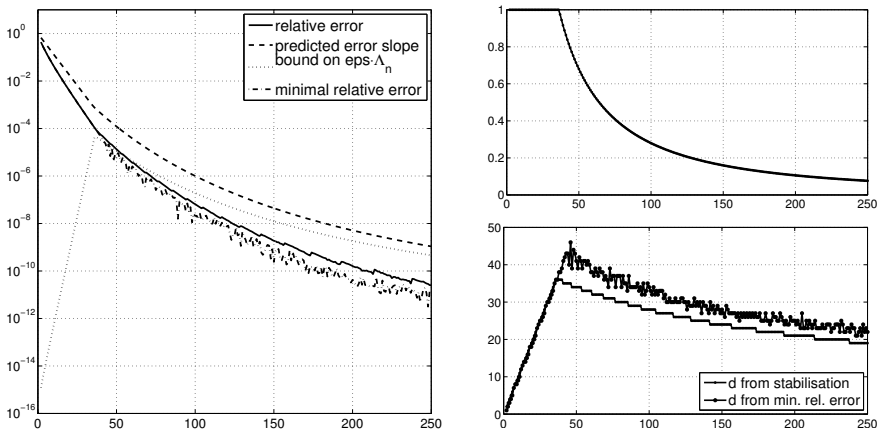

Figure 4.9: Relative errors for the interpolation of $f(x) = \Gamma(x + 1.1)$.

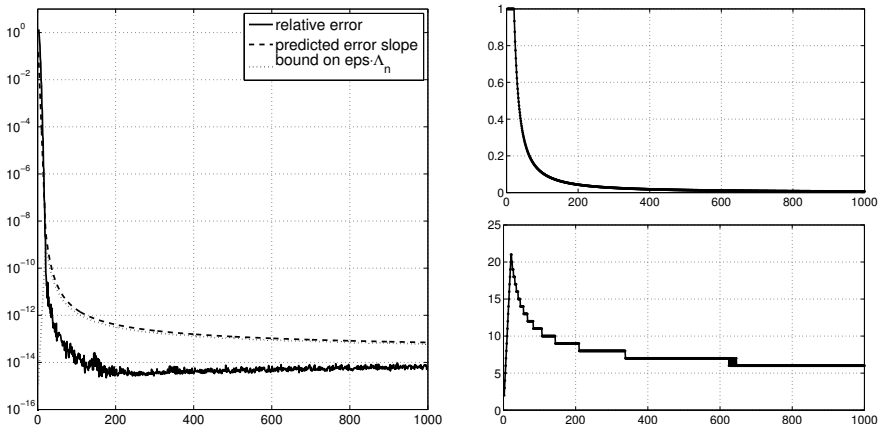

Figure 4.10: Relative errors for the interpolation of $f(x) = \sin(x)$ on $[-5, 5]$ with $2 \leq n \leq 1000$.

Table 4.1: Results of the algorithm described in the remark of Section 4.2.4.

Target relative error	Function f	C_{opt}	n	d	Observed error
10^{-6}	$\log(1.2-x)/(x^2+2)$	0.61	40	24	$5.56 \cdot 10^{-8}$
	$\Gamma(x+1.1)$	0.42	75	32	$5.90 \cdot 10^{-7}$
	$\Gamma(x+2)$	0.29	26	7	$3.66 \cdot 10^{-8}$
	$\arctan(\pi x)$	0.19	31	6	$2.27 \cdot 10^{-7}$
	$\sin(5x)$	0.33	22	7	$4.80 \cdot 10^{-6}$
10^{-9}	$\log(1.2-x)/(x^2+2)$	0.30	73	22	$2.36 \cdot 10^{-10}$
	$\Gamma(x+1.1)$	0.15	151	22	$3.06 \cdot 10^{-9}$
	$\Gamma(x+2)$	0.29	39	11	$2.51 \cdot 10^{-11}$
	$\arctan(\pi x)$	0.19	47	9	$2.47 \cdot 10^{-10}$
	$\sin(5x)$	0.39	34	11	$5.50 \cdot 10^{-10}$

at sufficiently many points.

The last example demonstrates that the presented convergence theory is also valid for nodes that are not equispaced. To this end we consider the node density $\phi(x) = (4 + \arctan(4x))/8$ on the interval $[-1, 1]$. We have computed the nodes distributed according to this density by evaluating the inverse cumulative node distribution $\Phi^{-1}(i/n)$ for $i = 0, 1, \ldots, n$; see Figure 4.11 (left). The level lines with the convergence rates for $C = 0.2$ are shown in Figure 4.2 (right). They predict the somewhat counterintuitive effect which we observe numerically in Figure 4.11 (right): When interpolating $f_+(x) = (x - 1.2)^{-2}$, the convergence is slower than for $f_-(x) = (x + 1.2)^{-2}$, although for f_+ the singularity $s = 1.2$ is to the right of $[-1, 1]$, where the nodes are about twice as dense as on the left end of the interval. This suggests that local refinement close to a singularity may not yield the desired faster convergence; the overall node distribution ϕ must be taken into account.

4.3 A Good Choice of Nodes

We keep the parameter d variable as in the preceding section, i.e., we suppose that $d = d(n) = \text{round}(Cn)$ with C between 0 and 1. If $C = 1$, the rational interpolants r_n simplify to the polynomial p_n and if $C = 0$, then r_n is Berrut's interpolant. As we want to discuss a good choice of interpolation nodes, we begin with polynomial interpolation, i.e., with the case $C = 1$. A standard result [31, Chap. 3.1] on the error in polynomial interpolation says that for

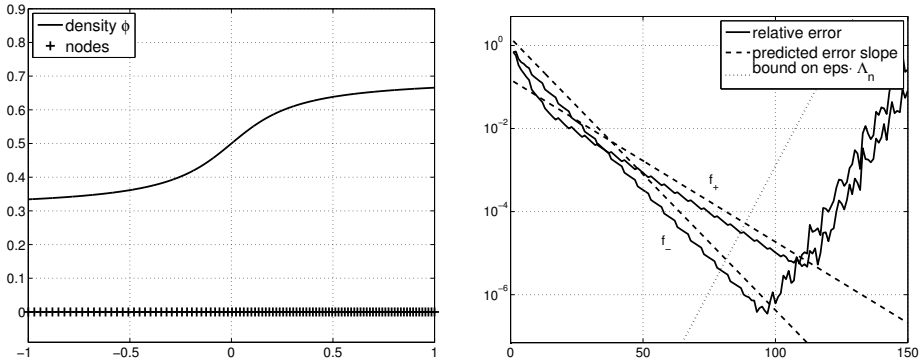

Figure 4.11: Nonequispaced nodes, $n = 100$, with their density $\phi(x) = (4 + \arctan(4x))/8$ on $[-1, 1]$ (left) and convergence of the interpolation in these nodes for f_- and f_+ with singularity $s = -1.2$ and $s = 1.2$, respectively, together with the predicted error slope 0.860^n and 0.914^n, respectively, for $1 \le n \le 150$ (right).

$f \in C[a, b]$ and $x \in [a, b]$,

$$|f(x) - p_n(x)| \le \|f^{(n+1)}\| \frac{|x - x_0| \cdots |x - x_n|}{(n + 1)!}. \tag{4.19}$$

This error bound may be split into two parts as explained, e.g., in [31, Chap. 3.3]. The first part involves the $(n + 1)$st derivative of f and cannot be influenced by a good choice of nodes. However, the other factor, the absolute value of the nodal polynomial L, and its maximum over $[a, b]$ depend only on the nodes. We suppose without restriction of generality that the interval of interpolation is $[-1, 1]$. Any other interval $[a, b]$ can be obtained via the transformation $x \mapsto (a+b)/2+x(b-a)/2$. It can be proven [31, 87] that $\|L\|$ is smallest if the nodes are the Chebyshev points of the first kind. With this choice of nodes, $L(x) = T_{n+1}(x)/2^n$, where T_{n+1} is the $(n + 1)$st Chebyshev polynomial, and the latter equi-oscillates between its extrema; see Figure 4.12. This is a characterisation of best approximation, and $T_{n+1}/2^n$ can be seen as the best monic polynomial approximation of degree $n + 1$ to the zero function, see, e.g., [118], so that $\|L\|$ is minimal if the nodes are Chebyshev points of the first kind. $\|L\|$ is almost as small with Chebyshev points of the second kind, whereas with equispaced points it shows large oscillations at the ends of the interval; see again Figure 4.12.

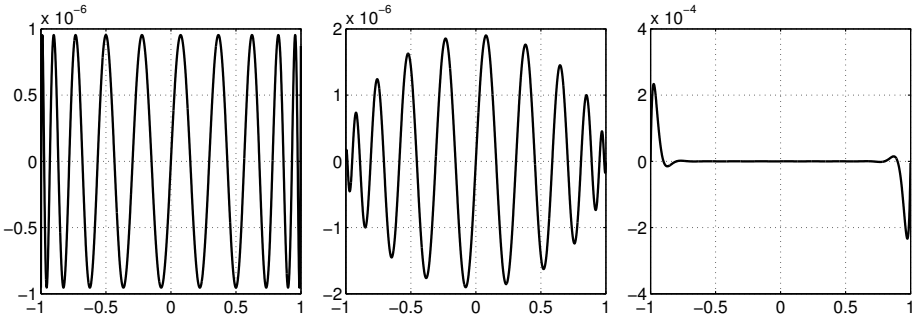

Figure 4.12: The nodal polynomial L with $n = 20$ and Chebyshev points of the first kind (left), second kind (middle), and equispaced points (right); L equi-oscillates with Chebyshev points of the first kind (left).

In polynomial interpolation, the optimality of Chebyshev points of the first kind as well as the near-optimality of Chebyshev points of the second kind, is also seen by the fact that the corresponding logarithmic potential (4.7) is constant in the interval; see [110]. There is no difference between Chebyshev points of either kind since their density is the same, so that the logarithmic potential for both kinds of Chebyshev points is identical. The picture on the right-hand side in Figure 4.14 illustrates that this potential is constant over the interval; see the curve labelled $C = 1$. This is not the case for the potential (4.6) associated with equispaced points in polynomial interpolation as can be seen in the picture on the left in Figure 4.14. The relation between equi-oscillation of L and the constant logarithmic potential comes from the fact that $-\log(|L|^{1/(n+1)})$ is the discrete analogue of the potential.

Two additional arguments in favour of Chebyshev points can be found, e.g., in [8]. The first addresses near-optimality of polynomial interpolation in Chebyshev points. Equations (3.9) and (3.10) reveal that the interpolation error is bounded by $2 + \frac{2}{\pi}\log(n+1)$ multiplied by the error of best polynomial approximation of degree n. The approximation error, in maximum norm, of a merely continuous function is therefore within a factor of 10 of the best polynomial approximation if $n < 10^5$. For this reason, polynomial interpolation between Chebyshev points is often called *near-best*. If the interpolated function is analytic in a closed ellipse with foci ± 1 and whose semimajor and semiminor axis length sum up to κ, then the error is on the order of $O(\kappa^{-n})$, as $n \to \infty$. This result is closely related to Theorem 4.2.

Chebyshev points are doubtlessly a very good choice of nodes for polynomial interpolation. Chebyshev points of the second kind are often preferred, since they include the endpoints of the interval and yield results which are nearly as good as those obtained with Chebyshev points of the first kind.

One could stop the theory at this point and admit that, whenever very good approximation is needed, it is enough to sample f at Chebyshev points and to interpolate the data by a polynomial interpolant. However, one cannot always choose the interpolation nodes and sometimes the data is only available at equispaced nodes, e.g., if the nodes are measurement points in an experiment from applied natural science. Moreover equispaced nodes are a much more natural choice if we want to sample a function on which we do not have further information regarding differentiability and other behaviour. We have seen in Section 3.3.2 that polynomial interpolation with equispaced nodes is ill-conditioned, which means that small perturbations in the data will be highly amplified during the interpolation process. Even worse, the interpolants do not always converge for analytic functions which have singularities near the interval as we explained in Section 4.1.

As mentioned in Chapter 1, a similar behaviour is to be expected from any method with equispaced nodes which would converge exponentially in theory: The condition number of such a method necessarily grows at the same exponential rate [85].

We have seen so far that Floater–Hormann interpolation with equispaced nodes is much more successful than polynomial interpolation. We showed stability of the interpolants when evaluated in barycentric form, the good condition as long as d is reasonably small, fast convergence for analytic functions and absence of Runge's phenomenon if d is kept constant. This indicates that the combination of Floater–Hormann interpolation with equispaced nodes is powerful. We want to add two more (heuristic) arguments in favour of this statement; these two are, however, only of speculative nature and might give additional hints why equispaced nodes are nearly optimal for Floater–Hormann interpolation. At the end of Section 3.3.3 we already observed that the Lebesgue constant increases most of the time as soon as the nodes are slightly changed from being equispaced; see Figure 3.5.

Chebyshev points of the first kind are optimal in polynomial interpolation because they minimise L, the part of the error that does not depend on f. The error in Floater–Hormann interpolation can be written as in (2.19), namely as a quotient whose numerator depends on the function and whose denominator depends only on the nodes. The denominator may be rewritten as

$$(-1)^{n-d}\frac{L(x)}{\sum_{i=0}^{n-d}\mu_i(x)}, \tag{4.20}$$

where $\mu_i(x) = (-1)^{n-d}L(x)\lambda_i(x)$, as in the proof of Theorem 2.2, shown in [46], on the absence of real poles. It is proven there that $\sum_{i=0}^{n-d}\mu_i(x)$ is positive for all real x, so that the rational function (4.20) cannot have more than $n+1$ real roots. It is thus natural to think that if the function (4.20) equi-oscillates in the interval, then the corresponding nodes should be at least close to optimal. We plotted this expression with equispaced nodes, $n = 100$ and various values of d in Figure 4.13. It can be observed that for $d = 0$, corresponding to Berrut's interpolant, the function almost equi-oscillates, and with increasing d, equi-oscillation occurs only in the middle part of the interval whereas near the ends, higher oscillations show up. This observation and the similar behaviour of the Lebesgue functions motivate to investigate nodes that are distributed equally in the middle part of the interval and cluster more and more toward the ends as d increases so as to reach Chebyshev points for $d = n$.

In addition to the almost equi-oscillation of (4.20), the potential function (4.13) associated with equispaced nodes is almost constant in the interval for small C, which corresponds to small d as well. As C decreases, the potential is constant over a larger part of the interval; see the left picture in Figure 4.14. The exact opposite is true with Chebyshev points. This gives a heuristic explanation of observations from extensive numerical experiments that Floater–Hormann interpolation with equispaced nodes improves when d is decreased from n toward 0. Again, the opposite is observed with Chebyshev points: As d increases from 0 to n, the interpolation quality becomes better until it reaches the good approximation of polynomial interpolation in Chebyshev points.

From the proven theoretical arguments and the above heuristics, one can state that the combination of Floater–Hormann interpolation and equispaced nodes is almost as successful a combination as polynomial interpolation between Chebyshev points of the second kind. The speed of convergence is of course not exactly the same and the condition and stability must be slightly worse [85], but Floater–Hormann interpolation is still a stable and well-conditioned interpolation scheme, that can be evaluated easily and cheaply with the barycentric form and gives good to near optimal approximation with the natural equispaced nodes.

The closeness to optimal of equispaced nodes for Floater–Hormann interpolation can be observed in Figure 4.15, where we computed again the denominator of r_n, this time with equispaced nodes and optimised nodes. The latter

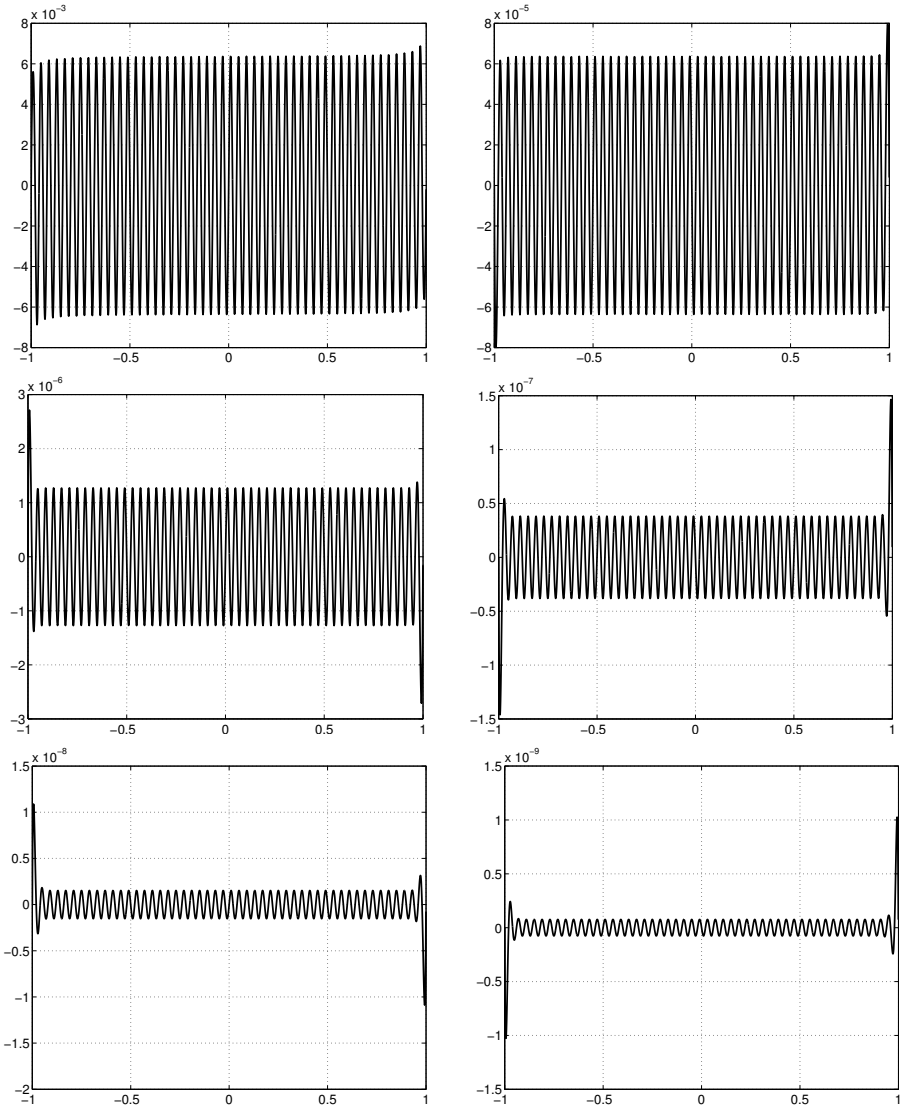

Figure 4.13: The reciprocal of the denominator $\sum_{i=0}^{n-d} \lambda_i(x)$ of r_n plotted as $L(x)/\sum_{i=0}^{n-d} \mu_i(x)$ with $n = 100$ and $d = 0, 1, 2, 3, 4, 5$ (from left to right and top to bottom).

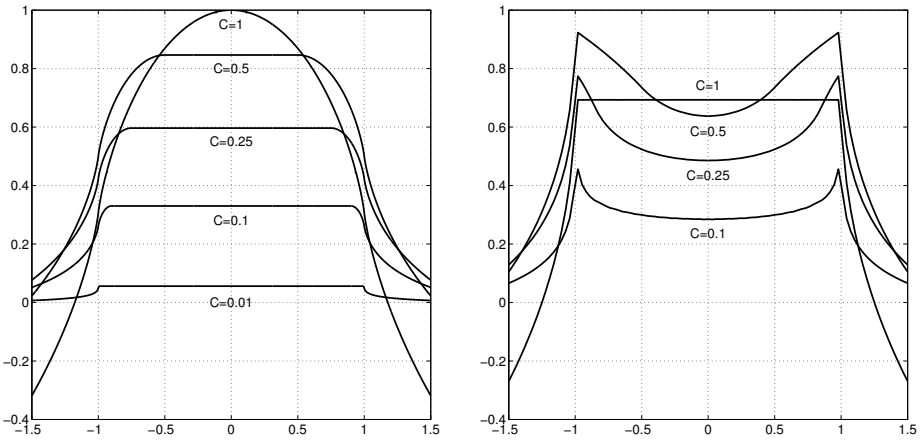

Figure 4.14: The potential function $V^{C,\mu}$ for equispaced nodes (left) and Chebyshev points (right) and various values of C.

are obtained with MATLAB's `fmincon`, configured to give the minimum under the constraints $x_1 < x_2 < \ldots < x_{n-2} < x_{n-1}$ using sequential quadratic programming (sqp); see [81]. The argument which we passed to `fmincon` is the maximum of $L(x)/\sum_{i=0}^{n-d} \mu_i(x)$ in 3000 equispaced nodes and the starting vector[2] was a vector rescaled to $[-1, 1]$, whose first and last few nodes are taken from a set of $d + 1$ Chebyshev points of the second kind and the nodes in the middle are equispaced. The picture on the right clearly shows that the nodes from the optimisation are nearly equispaced, they cluster only slightly toward the ends of the interval. This behaviour seems natural since Floater–Hormann interpolation is blended polynomial interpolation and the first and last few of these polynomials only share parts of their nodes with some of the remaining polynomials. Therefore, a distribution like that of Chebyshev points throughout the interval is not particularly good for the blended polynomials which interpolate only between subsets of the nodes in the interval, so that Chebyshev points are not much better than random points in Floater–Hormann interpolation.

[2]This idea of this distribution of nodes was originally formulated by Kai Hormann during a discussion on a slightly different topic.

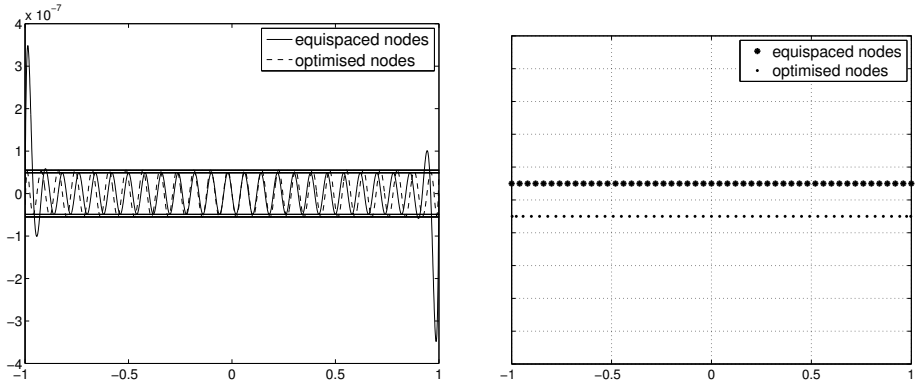

Figure 4.15: The reciprocal of the denominator $\sum_{i=0}^{n-d} \lambda_i(x)$ of r_n plotted as $L(x)/\sum_{i=0}^{n-d} \mu_i(x)$ with $n = 50$ and $d = 4$, with optimised nodes and equispaced nodes (left). The horizontal lines indicate the maximum of the denominator function; the outer lines give the overall maximum with optimal nodes and the inner lines show the maximum in the middle of the interval with equispaced nodes. The picture on the right shows the location of the optimised, nearly equispaced nodes compared with that of equispaced nodes.

Chapter 5

Approximation of Derivatives

Derivatives of polynomial interpolants lead in a natural way to approximations of derivatives of the interpolated function, e.g., through finite differences [47, 48, 49, 50]. In polynomial and spline interpolation the kth derivative of the interpolant, as a function of the mesh size h, typically converges at the rate of $O(h^{d+1-k})$ as $h \to 0$, where d is the degree of the polynomial or spline [33]. Taking into account that the rational Floater–Hormann interpolant r_n is a blend of polynomial interpolants of degree at most d, it is not unreasonable to expect that the kth derivative of the interpolation error e from (2.19) satisfies

$$\|e^{(k)}\| \leq Kh^{d+1-k}.$$

Here and in all what follows, K is again a constant depending only on d, on derivatives of f, on the length $b-a$ of the interval and possibly on other constants. Notice that the linearity of the derivative implies that the derivative of the interpolation error matches the error of the approximation of the derivative of a function by the same derivative of the interpolant.

In this chapter we establish this convergence rate for r_n with arbitrary nodes in the cases $k = 1, 2$ and with equispaced and quasi-equispaced nodes for derivatives of higher order, $k = 3, \ldots, d$, but then only at the nodes. We suggest an even cheaper alternative with almost the same rate for the approximation of derivatives of higher order at intermediate points. Thereafter, as an application, we present improved finite difference formulas deduced from these rational

interpolants. The formulas contain the classical polynomial finite differences as a special case, but are more stable for calculating one-sided approximations of derivatives as well as derivatives close to boundaries.

The main references for this chapter are [14] and [76], in addition to those cited in the text.

5.1 Error at the Nodes

We start from the representation (2.19) of the interpolation error e and denote its numerator by

$$A(x) := \sum_{i=0}^{n-d} (-1)^i f[x_i, \ldots, x_{i+d}, x] \tag{5.1}$$

and the denominator by

$$B(x) := \sum_{i=0}^{n-d} \lambda_i(x), \tag{5.2}$$

so that

$$e(x) = \frac{\sum_{i=0}^{n-d} (-1)^i f[x_i, \ldots, x_{i+d}, x]}{\sum_{i=0}^{n-d} \lambda_i(x)} = \frac{A(x)}{B(x)}. \tag{5.3}$$

Consider the first derivative of e at a node x_j, $0 \le j \le n$. We look at

$$q_j(x) := \frac{e(x)}{x - x_j}, \tag{5.4}$$

which, by the definition of the derivative and the fact that $e(x_j) = 0$, satisfies

$$e'(x_j) = \lim_{x \to x_j} q_j(x). \tag{5.5}$$

This motivates us to look at the product $(x - x_j)B(x)$. Defining the functions

$$B_j(x) := \sum_{i \in J_j} (-1)^i \prod_{\substack{k=i \\ k \neq j}}^{i+d} \frac{1}{x - x_k} \tag{5.6}$$

and

$$C_j(x) := \sum_{i \in I_n \setminus J_j} (-1)^i \prod_{k=i}^{i+d} \frac{1}{x - x_k}, \tag{5.7}$$

where the index sets I_n and J_j are defined in (2.13), we may rewrite q_j as

$$q_j(x) = \frac{A(x)}{g_j(x)} \tag{5.8}$$

with

$$g_j(x) := B_j(x) + (x - x_j)C_j(x). \tag{5.9}$$

Lemma 5.1.[1]

$$e'(x_j) = \frac{A(x_j)}{B_j(x_j)}.$$

Proof. By (5.4), (5.8) and (5.9) we have

$$q_j(x) = \frac{A(x)}{B_j(x) + (x - x_j)C_j(x)}, \tag{5.10}$$

and taking the limit of both sides as $x \to x_j$ gives the result; see (5.5). □

We can use this formula to obtain an error bound at the nodes that requires f to be in $C^{d+2}[a, b]$, the same as for the bound on the interpolation error itself from Theorem 2.3. We introduce the following notation:

$$d_i(x) := |x - x_i|, \quad d_{ik} := |x_i - x_k|,$$

for nodes x_i and x_k and for $x \in [a, b]$, and when it is clear, we also write $d_i = d_i(x)$.

Theorem 5.2. *If $f \in C^{d+2}[a, b]$, then*

$$|e'(x_j)| \le Kh^d, \quad 0 \le j \le n.$$

Proof. With $x = x_j$ in (5.6), the products alternate in sign as $(-1)^i$ does, so that all the terms in the sum have the same sign and

$$|B_j(x_j)| = \sum_{i \in J_j} \prod_{\substack{k=i \\ k \neq j}}^{i+d} d_{jk}^{-1}.$$

[1] This lemma and the idea of working with the definition of the derivative are due to an earlier collaboration of Jean-Paul Berrut with Michael Floater.

Therefore, by choosing any $i \in J_j$, we deduce that

$$\frac{1}{|B_j(x_j)|} \leq \prod_{\substack{k=i \\ k\neq j}}^{i+d} d_{jk} \leq Kh^d, \quad \forall\, i \in J_j. \tag{5.11}$$

On the other hand it has been shown in [46] that

$$|A(x)| \leq K, \quad x \in [a,b], \tag{5.12}$$

whence the bound follows. □

To deal with higher derivatives, we consider the Taylor expansions of e and q_j at $x = x_j$, namely

$$e(x) = (x-x_j)e'(x_j) + \frac{1}{2!}(x-x_j)^2 e''(x_j) + \frac{1}{3!}(x-x_j)^3 e'''(x_j) + \cdots \tag{5.13}$$

and

$$q_j(x) = q_j(x_j) + (x-x_j)q_j'(x_j) + \frac{1}{2!}(x-x_j)^2 q_j''(x_j) + \cdots .$$

Dividing (5.13) by $x - x_j$ and equating terms in the two expansions imply that

$$e^{(k)}(x_j) = kq_j^{(k-1)}(x_j), \tag{5.14}$$

in particular,

$$e''(x_j) = 2q_j'(x_j). \tag{5.15}$$

Differentiating (5.10) and substituting $x = x_j$ give

$$q_j'(x_j) = \frac{A'(x_j)}{B_j(x_j)} - \frac{B_j'(x_j)A(x_j)}{B_j^2(x_j)} - \frac{C_j(x_j)A(x_j)}{B_j^2(x_j)}, \tag{5.16}$$

which we will use to derive a bound for $e''(x_j)$. We begin with a lemma.

Lemma 5.3. *If $f \in C^{d+2+k}[a,b]$ for $k \in \mathbb{N}$, then*

$$|A^{(k)}(x)| \leq K, \quad x \in [a,b].$$

Proof. The case $k = 0$ has been treated in [46]. For $k \neq 0$ and using the derivative formula for divided differences (see [2] and [69]), we have

$$A^{(k)}(x) = k! \sum_{i=0}^{n-d} (-1)^i f[x_i, \ldots, x_{i+d}, (x)^{k+1}],$$

where $(x)^k$ stands for a k-fold argument. Then, with a similar approach to [46, p. 322], $A^{(k)}(x)/k!$ is seen to be

$$-\sum_{i=0,\ i\ \text{even}}^{n-d-1}(x_{i+d+1}-x_i)f[x_i,\dots,x_{i+d+1},(x)^{k+1}],$$

if $n-d$ is odd and

$$-\sum_{i=0,\ i\ \text{even}}^{n-d-2}(x_{i+d+1}-x_i)f[x_i,\dots,x_{i+d+1},(x)^{k+1}]+f[x_{n-d},\dots,x_n,(x)^{k+1}],$$

if $n-d$ is even. Using the same argument as in [46], we are done. □

Theorem 5.4. *If $d\geq 1$ and if $f\in C^{d+3}[a,b]$, then*

$$|e''(x_j)|\leq Kh^{d-1}.$$

Proof. We write equation (5.15) with (5.16) in the form

$$e''(x_j)=2(L_1-L_2-L_3),$$

where

$$L_1:=\frac{A'(x_j)}{B_j(x_j)},\quad L_2:=\frac{B_j'(x_j)A(x_j)}{B_j^2(x_j)},\quad L_3:=\frac{C_j(x_j)A(x_j)}{B_j^2(x_j)},$$

and we show that

$$|L_1|\leq Kh^d, \tag{5.17}$$

and

$$|L_2|,\ |L_3|\leq Kh^{d-1}. \tag{5.18}$$

Equation (5.17) immediately follows from (5.11) and Lemma 5.3. To deal with L_2, we notice that

$$B_j'(x)=\sum_{i\in J_j}(-1)^{i+1}\sum_{\substack{m=i\\ m\neq j}}^{i+d}\frac{1}{x-x_m}\prod_{\substack{k=i\\ k\neq j}}^{i+d}\frac{1}{x-x_k},$$

so that its absolute value in $x=x_j$ is bounded by

$$|B_j'(x_j)|\leq\sum_{i\in J_j}\sum_{\substack{m=i\\ m\neq j}}^{i+d}d_{jm}^{-1}\prod_{\substack{k=i\\ k\neq j}}^{i+d}d_{jk}^{-1}. \tag{5.19}$$

To derive a bound of the quotient of (5.19) with $|B_j(x_j)|^2$, we use (5.11) with $i \in J_j$ equal to the index in the outer sum in (5.19):

$$\frac{|B_j'(x_j)|}{|B_j(x_j)|^2} \le \sum_{i\in J_j} \sum_{\substack{m=i\\ m\neq j}}^{i+d} d_{jm}^{-1} \prod_{\substack{k=i\\ k\neq j}}^{i+d} d_{jk}^2 \prod_{\substack{k=i\\ k\neq j}}^{i+d} d_{jk}^{-1} \le \sum_{i\in J_j} \sum_{\substack{m=i\\ m\neq j}}^{i+d} \prod_{\substack{k=i\\ k\neq j\\ k\neq m}}^{i+d} d_{jk} \le Kh^{d-1}$$

and this, together with (5.12), gives the bound on L_2 in (5.18).

Finally, we treat L_3. We split $C_j(x_j)$ into two parts,

$$C_j(x_j) = \sum_{i=0}^{j-d-1} (-1)^i \prod_{k=i}^{i+d} \frac{1}{x_j - x_k} + \sum_{i=j+1}^{n-d} (-1)^i \prod_{k=i}^{i+d} \frac{1}{x_j - x_k},$$

where empty sums are meant to equal 0. The terms in both sums alternate in sign, and increase and decrease, respectively, in absolute value, so that

$$|C_j(x_j)| \le \prod_{k=j-d-1}^{j-1} d_{jk}^{-1} + \prod_{k=j+1}^{j+1+d} d_{jk}^{-1}. \tag{5.20}$$

We now divide every term in equation (5.20) by $|B_j(x_j)|^2$. Using (5.11) with $i = j - d$ for the first term and $i = j$ for the second, we obtain

$$\frac{|C_j(x_j)|}{|B_j(x_j)|^2} \le \frac{\prod_{k=j-d}^{j-1} d_{jk}^2}{\prod_{k=j-d-1}^{j-1} d_{jk}} + \frac{\prod_{k=j+1}^{j+d} d_{jk}^2}{\prod_{k=j+1}^{j+d+1} d_{jk}} = \frac{\prod_{k=j-d}^{j-1} d_{jk}}{d_{j,j-d-1}} + \frac{\prod_{k=j+1}^{j+d} d_{jk}}{d_{j,j+d+1}}.$$

Since

$$\frac{d_{j,j-d}}{d_{j,j-d-1}} \le 1, \quad \frac{d_{j,j+d}}{d_{j,j+d+1}} \le 1,$$

it follows that

$$\frac{|C_j(x_j)|}{|B_j(x_j)|^2} \le \prod_{k=j-d+1}^{j-1} d_{jk} + \prod_{k=j+1}^{j+d-1} d_{jk} \le Kh^{d-1},$$

which, together with (5.12), gives the bound on L_3 in (5.18). □

Let us now investigate the approximation of derivatives of higher order at equispaced and quasi-equispaced nodes.

Theorem 5.5. *Suppose n, d, $d \le n$, and k, $k \le d$, are positive integers and $f \in C^{d+1+k}[a,b]$. If the nodes are equispaced or quasi-equispaced, then*

$$|e^{(k)}(x_j)| \le Kh^{d+1-k}, \quad 0 \le j \le n. \tag{5.21}$$

Proof. For $k = 1$ and $k = 2$, the statement is covered by the general case of differentiation at arbitrarily distributed nodes from Theorems 5.2 and 5.4.

We expand the derivatives of q_j in (5.14) by applying the Leibniz rule to the right-hand side of (5.8):

$$q_j^{(k-1)}(x) = \sum_{\ell=0}^{k-1} \binom{k-1}{\ell} A^{(k-1-\ell)}(x) \big(g_j^{-1}(x)\big)^{(\ell)}. \tag{5.22}$$

Lemma 5.3 guarantees that the absolute value of every factor $A^{(k-1-\ell)}(x)$ is bounded in $[a,b]$ by a constant; we thus merely look at the last factor of the terms in the sum. We will bound the ℓth derivative of the reciprocal of $g_j(x)$ for $\ell = 0, 1, \ldots, k-1$ at $x = x_j$. To this aim, we apply the "set partition version" of the Faà di Bruno formula for higher order derivatives of composite functions as given in [71], see also [28, 45],

$$(g_j^{-1}(x))^{(\ell)} = \sum (-1)^p \frac{p!}{g_j^{p+1}(x)} \prod_{i=1}^{\ell} \big(g_j^{(i)}(x)\big)^{b_i}, \tag{5.23}$$

where the sum runs over all partitions of the set $\{1, 2, \ldots, \ell\}$ and, for each partition, p is its number of blocks and b_i is the number of these blocks with precisely i elements.

We shall use (5.23) to show (5.21) by means of a lower bound on $g_j(x_j)$ and an upper bound on the ℓth derivative of $g_j(x)$ at $x = x_j$ for $\ell = 1, \ldots, k-1$. The former has been explicitly established in the proof of Theorem 5.2 and is valid for any distribution of the nodes:

$$|g_j(x_j)| = |B_j(x_j)| \ge \prod_{\substack{k=i \\ k \ne j}}^{i+d} d_{jk}^{-1} \ge Kh^{-d}, \quad \forall i \in J_j. \tag{5.24}$$

For the latter we consider

$$g_j^{(\ell)}(x_j) = B_j^{(\ell)}(x_j) + \ell C_j^{(\ell-1)}(x_j) \tag{5.25}$$

and use the Leibniz rule to obtain

$$B_j^{(\ell)}(x) = \sum_{i\in J_j} (-1)^{i+\ell}\ell! \sum_{|\mathbf{L}_{i,j}|=\ell} \prod_{\substack{k=i\\k\neq j}}^{i+d} \frac{1}{(x-x_k)^{1+\ell_{k-i}}},$$

where $\mathbf{L}_{i,j} := (\ell_0, \ldots, \ell_{j-i-1}, \ell_{j-i+1}, \ldots, \ell_d)$, $j = 0, \ldots, n$ and $i \in J_j$, are vectors whose components are nonnegative integers which sum up to

$$|\mathbf{L}_{i,j}| := \sum_{\substack{k=0\\k\neq j-i}}^{d} \ell_k.$$

Taking the absolute value of $B_j^{(\ell)}$ at $x = x_j$ leads to

$$|B_j^{(\ell)}(x_j)| \leq \sum_{i\in J_j} \ell! \sum_{|\mathbf{L}_{i,j}|=\ell} \prod_{\substack{k=i\\k\neq j}}^{i+d} d_{jk}^{-(1+\ell_{k-i})} \leq Kh^{-(d+\ell)}, \tag{5.26}$$

since every product in the inner sum involves the reciprocal of $d+\ell$ factors

$$d_{jk} \geq |j-k|h_* \geq Kh.$$

Analogously for C_j, with $\mathbf{L} := (\ell_0, \ldots, \ell_d)$, it follows that

$$C_j^{(\ell-1)}(x) = (\ell-1)! \sum_{|\mathbf{L}|=\ell-1} \sum_{i\in I_n\setminus J_j} (-1)^{i+l-1} \prod_{k=i}^{i+d} \frac{1}{(x-x_k)^{1+\ell_{k-i}}}.$$

We split the inner sum into its two parts with consecutive indices

$$\sum_{i=0}^{j-d-1} (-1)^{i+l-1} \prod_{k=i}^{i+d} \frac{1}{(x-x_k)^{1+\ell_{k-i}}} + \sum_{i=j+1}^{n-d} (-1)^{i+l-1} \prod_{k=i}^{i+d} \frac{1}{(x-x_k)^{1+\ell_{k-i}}},$$

where empty sums are meant to equal 0. The terms in the left and right sums alternate in sign, and increase and decrease, respectively, in absolute value. Therefore, we obtain

$$|C_j^{(\ell-1)}(x_j)| \leq (\ell-1)! \sum_{|\mathbf{L}|=\ell-1} \left(\prod_{k=j-d-1}^{j-1} d_{jk}^{-(1+\ell_{k-j+d+1})} + \prod_{k=j+1}^{j+1+d} d_{jk}^{-(1+\ell_{k-j-1})} \right),$$

which is bounded by a constant times $h^{-d-\ell}$, in the same fashion as (5.26). This result and (5.26) again, inserted into (5.25), yield

$$|g_j^{(\ell)}(x_j)| \le Kh^{-(d+\ell)}, \quad \ell = 1, \ldots, k-1. \tag{5.27}$$

Finally we show by induction that for $\ell = 1, \ldots, k-1$,

$$|(g_j^{-1}(x_j))^{(\ell)}| \le Kh^{d-\ell}. \tag{5.28}$$

For $\ell = 0$ and $\ell = 1$, this has been established in Theorems 5.2 and 5.4. Now suppose that (5.28) holds for a certain ℓ. Following Johnson's proof of the set partition version of the Faà di Bruno formula in [71], we use the form (5.23) to facilitate the step from the ℓth to the $(\ell+1)$st derivative of the reciprocal of g_j. Differentiating (5.23) adds terms to the sum which equal the former ones with one of the following two changes:

$$\frac{1}{g_j^{p+1}(x)} \longrightarrow -(p+1)\frac{g_j'(x)}{g_j^{p+2}(x)} \quad \text{or} \quad (g_j^{(i)}(x))^{b_i} \longrightarrow b_i (g_j^{(i)}(x))^{b_i-1} g_j^{(i+1)}(x).$$

Equations (5.24) and (5.27) show that the bound on the $(\ell+1)$st derivative of the reciprocal of g_j at $x = x_j$ includes an additional factor $1/h$ as compared with the bound on the ℓth derivative. □

5.2 Error at Intermediate Points

We now turn our attention to intermediate points, i.e., the $x \in [a, b]$ that are not interpolation points. For the first derivative we obtain the same rate of convergence as at the nodes, namely $O(h^d)$, but only under the stricter condition that $f \in C^{d+3}[a, b]$.

Theorem 5.6. *If $d \ge 2$ and if $f \in C^{d+3}[a, b]$, then*

$$\|e'\| \le Kh^d.$$

Proof. Due to the continuity of e', it is sufficient to let $x \in (x_j, x_{j+1})$ and to show that

$$|e'(x)| \le Kh^d, \tag{5.29}$$

independently of j. To establish (5.29), we differentiate (5.3), to obtain

$$e'(x) = \frac{A'(x)}{B(x)} - A(x)\frac{B'(x)}{B^2(x)}. \tag{5.30}$$

In the proof of Theorem 2 of [46] it was shown that, see also (2.20),

$$|B(x)| \geq \frac{1}{d!h^{d+1}}, \quad \forall x \in [a,b],$$

and so, from Lemma 5.3, it follows that

$$\frac{|A'(x)|}{|B(x)|} \leq Kh^{d+1}.$$

Since $|A(x)| \leq K$, it remains to show that

$$\frac{|B'(x)|}{|B^2(x)|} \leq Kh^d. \tag{5.31}$$

We use the following index sets introduced in [46] and which subdivide I_n:

$$\begin{aligned} I_1 &= \{i \in I_n : i \leq j-d\}, \\ I_2 &= \{i \in I_n : j-d+1 \leq i \leq j\}, \\ I_3 &= \{i \in I_n : j+1 \leq n-d\}. \end{aligned}$$

Now

$$\begin{aligned} |B'(x)| &= \left| \sum_{i \in I_n} \sum_{m=0}^{d} \frac{(-1)^{i+1}}{(x-x_i)\cdots(x-x_{i+d})(x-x_{i+m})} \right| \\ &\leq \sum_{m=0}^{d} (M_{m,1} + M_{m,2} + M_{m,3}), \end{aligned} \tag{5.32}$$

where we have interchanged the summation order and set

$$M_{m,p} := \left| \sum_{i \in I_p} \frac{(-1)^{i+1}}{(x-x_i)\cdots(x-x_{i+d})(x-x_{i+m})} \right|, \quad p = 1,2,3.$$

If $I_p = \varnothing$, then $M_{m,p} = 0$. I_2 is not empty since $d \geq 2$.

For every fixed m, the terms in the sums in $M_{m,1}$ and $M_{m,3}$ alternate in sign, and increase and decrease, respectively, in absolute value and so

$$M_{m,1} \leq \frac{1}{d_{j-d}\cdots d_j d_{j-d+m}} \quad \text{and} \quad M_{m,3} \leq \frac{1}{d_{j+1}\cdots d_{j+1+d} d_{j+1+m}}.$$

In the same proof in [46, p. 322], it has been shown that, see also (2.21),

$$|B(x)| \geq |\lambda_i(x)|, \quad \forall\, i \in I_2. \tag{5.33}$$

Next, we divide $M_{m,1}$ by $|B(x)|^2$ and use (5.33) with $i = j - d + 1$, so that

$$\frac{M_{m,1}}{|B(x)|^2} \leq \frac{d_{j-d+1}^2 \cdots d_{j+1}^2}{d_{j-d} \cdots d_j d_{j-d+m}} = \frac{d_{j-d+1} \cdots d_j d_{j+1}^2}{d_{j-d} d_{j-d+m}},$$

and since $d_j/d_{j-d+m} \leq 1$ for $m = 0, \ldots, d$ and $d_{j-d+1}/d_{j-d} \leq 1$:

$$\frac{M_{m,1}}{|B(x)|^2} \leq d_{j-d+2} \cdots d_{j-1} d_{j+1}^2 \leq K h^d.$$

Similarly

$$\frac{M_{m,3}}{|B(x)|^2} \leq K h^d.$$

Finally we bound $M_{m,2}/|B(x)|^2$. Choosing the same $i \in I_2$ in (5.33) as in each term of the sum in $M_{m,2}$, it follows

$$\frac{M_{m,2}}{|B(x)|^2} \leq \sum_{i \in I_2} \frac{d_i^2 \cdots d_{i+d}^2}{d_i \cdots d_{i+d} d_{i+m}} = \sum_{i \in I_2} d_i \cdots d_{i+m-1} d_{i+m+1} \cdots d_{i+d} \leq K h^d.$$

Thus (5.31) follows from (5.32). □

In the case $d = 1$, we obtain the same rate of convergence, $O(h)$, as for the larger d in Theorem 5.6 but only under a bounded local mesh ratio.

Theorem 5.7. *If $d = 1$ and if $f \in C^4[a, b]$, then*

$$\|e'\| \leq K(\widetilde{\beta} + 1)h,$$

where

$$\widetilde{\beta} := \max\Big\{ \max_{1 \leq i \leq n-1} \frac{d_{i,i+1}}{d_{i,i-1}}, \max_{0 \leq i \leq n-2} \frac{d_{i+1,i}}{d_{i+1,i+2}} \Big\}.$$

Proof. Again we determine the open subinterval (x_j, x_{j+1}) containing x and consider (5.30). Since the bounds for $|A(x)|$, $|A'(x)|$ and $|B(x)|$ from the previous theorem also hold for $d = 1$, we bound $|B'(x)|/|B(x)|^2$ for $d = 1$ and

$I_2 = \{j\}$. Using similar arguments as in that theorem, we obtain

$$\begin{aligned}
\frac{|B'(x)|}{|B(x)|^2} &\le \sum_{m=0}^{1}\left(\frac{1}{|B(x)|^2}\left|\sum_{i\in I_n}\frac{(-1)^{i+1}}{(x-x_i)(x-x_{i+1})(x-x_{i+m})}\right|\right)\\
&\le \sum_{m=0}^{1}\left(\frac{d_j^2 d_{j+1}^2}{d_{j-1}d_j d_{j-1+m}} + \frac{d_j^2 d_{j+1}^2}{d_j d_{j+1} d_{j+m}} + \frac{d_j^2 d_{j+1}^2}{d_{j+1}d_{j+2}d_{j+1+m}}\right)\\
&\le 2\frac{d_{j+1}^2}{d_{j-1}} + d_{j+1} + d_j + 2\frac{d_j^2}{d_{j+2}}\\
&\le 2(2\widetilde{\beta}+1)h \le 4(\widetilde{\beta}+1)h. \qquad \square
\end{aligned}$$

For the second derivative the mesh ratio enters every bound.

Theorem 5.8. *If $d \ge 3$ and if $f \in C^{d+4}[a,b]$, then*

$$\|e''\| \le K(\widetilde{\beta}+1)h^{d-1}.$$

Proof. We continue to work with $x \in (x_j, x_{j+1})$, and we express the error e in (5.3) as

$$e(x) = \psi(x)\widetilde{e}(x),$$

where

$$\psi(x) := (x-x_j)(x-x_{j+1}), \quad \widetilde{e}(x) := \frac{A(x)}{\widetilde{B}(x)} \quad \text{and} \quad \widetilde{B}(x) := \psi(x)B(x).$$

Now, by the Leibniz rule,

$$\begin{aligned}
e''(x) &= \sum_{i=0}^{2}\binom{2}{i}\psi^{(2-i)}(x)\widetilde{e}^{(i)}(x)\\
&= 2\frac{A(x)}{\widetilde{B}(x)} + 2\psi'(x)\left(\frac{A'(x)}{\widetilde{B}(x)} - A(x)\frac{\widetilde{B}'(x)}{\widetilde{B}^2(x)}\right) \qquad (5.34)\\
&\quad + \psi(x)\left(\frac{A''(x)}{\widetilde{B}(x)} - 2A'(x)\frac{\widetilde{B}'(x)}{\widetilde{B}^2(x)} + 2A(x)\frac{\widetilde{B}'^2(x)}{\widetilde{B}^3(x)} - A(x)\frac{\widetilde{B}''(x)}{\widetilde{B}^2(x)}\right).
\end{aligned}$$

Every factor $A^{(k)}(x)$ can be bounded using Lemma 5.3. In the coming arguments we use the following result.

Lemma 5.9. *If $d \geq 1$ and if $x \in [a,b]$, then*

$$\frac{1}{\left|\widetilde{B}(x)\right|} \leq Kh^{d-1}.$$

Proof. For $x \in (x_j, x_{j+1})$, the definition of $\widetilde{B}$ reads

$$\widetilde{B}(x) = \psi(x)B(x) = \psi(x)\sum_{i=0}^{n-d}\lambda_i(x).$$

Since

$$|B(x)| \geq |\lambda_i(x)|,$$

for any $i \in I_2$, we deduce from the definition of λ_i that

$$\frac{1}{\left|\widetilde{B}(x)\right|} \leq \frac{\prod_{k=i}^{i+d} d_k}{d_j d_{j+1}} = \prod_{\substack{k=i \\ k\neq j,j+1}}^{i+d} d_k \leq Kh^{d-1}, \quad \forall i \in I_2, \tag{5.35}$$

which evidently holds also at the nodes. □

The factors which remain to be bounded are the following:

$$N_1(x) := \frac{\widetilde{B}'(x)}{\widetilde{B}^2(x)}, \quad N_2(x) := \psi(x)\frac{\widetilde{B}'(x)}{\widetilde{B}(x)}, \quad N_3(x) := \psi(x)\frac{\widetilde{B}''(x)}{\widetilde{B}^2(x)}. \tag{5.36}$$

We split $\widetilde{B}$ into five parts:

$$\widetilde{B}(x) = \psi(x)\Bigg(\sum_{i=0}^{j-d-1}\lambda_i(x) + \lambda_{j-d}(x) + \sum_{i=j-d+1}^{j}\lambda_i(x) + \lambda_{j+1}(x) + \sum_{i=j+2}^{n-d}\lambda_i(x)\Bigg) \tag{5.37}$$

$$=: K_1(x) + K_2(x) + K_3(x) + K_4(x) + K_5(x).$$

For symmetry reasons, it is sufficient to study the first three terms, K_1, K_2 and K_3, since K_4 and K_5 are analogous to K_2 and K_1. We begin with the first derivative of K_1:

$$K_1'(x) = \psi'(x)\sum_{i=0}^{j-d-1}\lambda_i(x) + \psi(x)\sum_{i=0}^{j-d-1}\lambda_i'(x). \tag{5.38}$$

The terms in both sums alternate in sign and increase in absolute value; we deduce that

$$|K_1'(x)| \le 2h \prod_{k=j-d-1}^{j-1} d_k^{-1} + d_j d_{j+1} \sum_{m=j-d-1}^{j-1} d_m^{-1} \prod_{k=j-d-1}^{j-1} d_k^{-1}. \tag{5.39}$$

We next turn to K_2, which after simplification reads

$$K_2(x) = (x - x_{j+1})(-1)^{j-d} \prod_{k=j-d}^{j-1} (x - x_k)^{-1}. \tag{5.40}$$

It follows that

$$|K_2'(x)| \le \prod_{k=j-d}^{j-1} d_k^{-1} + d_{j+1} \sum_{m=j-d}^{j-1} d_m^{-1} \prod_{k=j-d}^{j-1} d_k^{-1}. \tag{5.41}$$

We may rewrite K_3 as

$$K_3(x) = \sum_{i=j-d+1}^{j} (-1)^i \prod_{\substack{k=i \\ k\neq j,j+1}}^{i+d} (x - x_k)^{-1}, \tag{5.42}$$

which yields the following bound for its derivative:

$$|K_3'(x)| \le \sum_{i=j-d+1}^{j} \sum_{\substack{m=i \\ m\neq j,j+1}}^{i+d} d_m^{-1} \prod_{\substack{k=i \\ k\neq j,j+1}}^{i+d} d_k^{-1}. \tag{5.43}$$

In view of deriving a bound on N_1, we first take the quotient of (5.39) with $\left|\widetilde{B}(x)\right|^2$. Choosing $i = j - d + 1$ in (5.35), we obtain

$$\begin{aligned} \frac{|K_1'(x)|}{\left|\widetilde{B}(x)\right|^2} &\le 2h \frac{\prod_{k=j-d+1}^{j-1} d_k^2}{\prod_{k=j-d-1}^{j-1} d_k} + d_{j+1} \sum_{m=j-d-1}^{j-1} \frac{d_j \prod_{k=j-d+1}^{j-1} d_k^2}{d_m \prod_{k=j-d-1}^{j-1} d_k} \\ &\le 2h \frac{\prod_{k=j-d+1}^{j-1} d_k}{d_{j-d-1} d_{j-d}} + h \sum_{m=j-d-1}^{j-1} \frac{d_j \prod_{k=j-d+1}^{j-1} d_k}{d_m d_{j-d-1} d_{j-d}}. \end{aligned}$$

Since $d_{j-d+1}/d_{j-d-1} \le 1$ and $d_{j-d+2}/d_{j-d} \le 1$, and $d_j/d_m \le 1$ for $m \le j-1$, we see that

$$\frac{|K_1'(x)|}{\left|\widetilde{B}(x)\right|^2} \le 2h \prod_{k=j-d+3}^{j-1} d_k + h \sum_{m=j-d-1}^{j-1} \prod_{k=j-d+3}^{j-1} d_k \le K h^{d-2}.$$

With similar arguments, a bound of the same order may be derived for $|K_2'|/\left|\widetilde{B}\right|^2$ and for $|K_3'|/|\widetilde{B}|^2$. The result is

$$|N_1(x)| \leq Kh^{d-2}.$$

To deal with N_2, we use the mesh ratio $\widetilde{\beta}$. Again we begin with the term involving K_1', choose $i = j - d + 1$ in (5.35) and, instead of cancelling factors in the numerator and denominator, we use the fact that $d_k/d_{k-1} \leq 1$ for $k = j - d + 1, \ldots, j - 1$:

$$\begin{aligned}|\psi(x)|\frac{|K_1'(x)|}{\left|\widetilde{B}(x)\right|} &\leq 2h\frac{d_j d_{j+1}\prod_{k=j-d+1}^{j-1} d_k}{\prod_{k=j-d-1}^{j-1} d_k} + d_{j+1}^2 \sum_{m=j-d-1}^{j-1} \frac{d_j^2 \prod_{k=j-d+1}^{j-1} d_k}{d_m \prod_{k=j-d-1}^{j-1} d_k} \\ &\leq 2h\frac{d_j d_{j+1}}{d_{j-d-1}d_{j-1}} + h\sum_{m=j-d-1}^{j-1} \frac{d_j^2 d_{j+1}}{d_m d_{j-d-1} d_{j-1}}.\end{aligned}$$

Since $d_j/d_{j-d-1} \leq 1$, and $d_j/d_m \leq 1$ for $m \leq j - 1$, we obtain

$$|\psi(x)|\frac{|K_1'(x)|}{\left|\widetilde{B}(x)\right|} \leq 2h\frac{d_{j+1}}{d_{j-1}} + h\sum_{m=j-d-1}^{j-1} \frac{d_{j+1}}{d_{j-1}} \leq K\widetilde{\beta}h.$$

Similar arguments lead to a bound of the same order for $|\psi(x)||K_2'|/|\widetilde{B}|$. For $|\psi(x)||K_3'|/|\widetilde{B}|$ we may cancel the whole product in every term of the inner sum without making use of the mesh ratio:

$$|\psi(x)|\frac{|K_3'(x)|}{\left|\widetilde{B}(x)\right|} \leq d_j d_{j+1} \sum_{i=j-d+1}^{j} \sum_{\substack{m=i \\ m\neq j,j+1}}^{i+d} d_m^{-1} \leq Kh.$$

Thus we have

$$|N_2(x)| \leq K(\widetilde{\beta} + 1)h.$$

A bound for N_3 may be derived using similar arguments as for N_1 and the following observation: The differentiation of $\widetilde{B}'$ leads to an extra factor $(x - x_i)^{-1}$ in some of the terms of $\widetilde{B}''$. Since $i \neq j, j + 1$, the absolute value of this factor can be eliminated through multiplication with $|\psi|$:

$$\frac{|\psi(x)|}{|x - x_i|} = \frac{d_j d_{j+1}}{d_i} \leq Kh.$$

Consequently

$$|N_3(x)| \leq Kh^{d-1}.$$

This last step concludes the proof, since bringing together all the bounds on the terms of the expansion (5.34) of e'' yields the claimed result. □

Theorem 5.10. *If $d = 2$ and if $f \in C^6[a, b]$, then*

$$\|e''\| \leq K(\widetilde{\beta}^2 + \widetilde{\beta} + 1)h.$$

Proof. If we again expand the factors N_1, N_2 and N_3 in (5.36) in the special case $d = 2$, we see that everyone of them may be bounded by a linear function of $\widetilde{\beta}$. □

The kth ($k \geq 3$) order derivative of f at intermediate points may also be approximated by the kth derivative of r_n, evaluated at such a point. However, if x is not a node, the expressions given in [97] for $r_n^{(k)}(x)$ as barycentric rational interpolants of divided differences get more and more expensive to evaluate and the formulas for the corresponding error $e^{(k)}(x)$ become very intricate. For this reason, and inspired by the polynomial case, we suggest that higher order derivatives of a function f be approximated at intermediate points by the rational interpolant $R_n^{(k)}$ of the approximations $r_n^{(k)}(x_i) =: f_i^{(k)}$ of corresponding higher order derivatives at the nodes:

$$R_n^{(k)}(x) := \sum_{i=0}^{n} \frac{w_i}{x - x_i} f_i^{(k)} \Big/ \sum_{i=0}^{n} \frac{w_i}{x - x_i}. \tag{5.44}$$

In Section 5.3, we shall review and use elegant formulas for $f_i^{(k)}$ involving differentiation matrices. The following proposition shows that the maximum norm of the error,

$$E^{(k)}(x) := R_n^{(k)}(x) - f^{(k)}(x),$$

decreases almost as $O(h^{d+1-k})$ with an increasing number of equispaced or quasi-equispaced nodes.

Proposition 5.11. *Suppose n, d, $d \leq n$, and k, $k \leq d$, are positive integers and $f \in C^{d+2+k}[a, b]$. If the nodes are equispaced or quasi-equispaced, then*

$$\|E^{(k)}\| \leq Kh^{d+1-k}\big(1 + \log(n)\big). \tag{5.45}$$

Proof. As the error $|E^{(k)}|$ is equal to $|e^{(k)}|$ at the nodes, we need only consider intermediate points $x \neq x_j$. First, we see that $f^{(k)}$ belongs to $C^{d+2}[a,b]$ and may thus be interpolated by the rational function $r_n[f^{(k)}]$ with parameter d and approximation rate $O(h^{d+1})$, leading to

$$|E^{(k)}(x)| \leq \left|R_n^{(k)}(x) - r_n[f^{(k)}](x)\right| + \left|r_n[f^{(k)}](x) - f^{(k)}(x)\right|. \tag{5.46}$$

The first term may be bounded by

$$\frac{\sum_{i=0}^{n} \left|\frac{w_i}{x-x_i}\right| |f_i^{(k)} - f^{(k)}(x_i)|}{\left|\sum_{i=0}^{n} \frac{w_i}{x-x_i}\right|} \tag{5.47}$$

and the second by Kh^{d+1-k}, using $h \leq (b-a)$. We have shown in Theorem 5.5 that

$$|f_i^{(k)} - f^{(k)}(x_i)| = |e^{(k)}(x_i)| \leq Kh^{d+1-k},$$

so that (5.47) may be further bounded by

$$Kh^{d+1-k}\Lambda_n(x),$$

where $\Lambda_n(x)$ is the Lebesgue function from Section 3.3.3 associated with the rational interpolant r_n. We have seen in Theorem 3.7 that the maximum Λ_n of $\Lambda_n(x)$ is bounded by $2^{d-1}M^d(2+M\log(n))$ for $d \geq 1$ and by $\frac{3}{4}M(2+M\log(n))$ for $d = 0$, independently of the length of the interval. The sum of the bounds on the first and second terms in (5.46) yields the claimed result. □

Observe that the Lebesgue constant shows up in this proof. It coincides with the condition number of the interpolation process; see also Section 3.3.1. This measure very naturally comes into play since $R_n^{(k)}$ can be interpreted as the rational interpolant to the perturbed values $f_i^{(k)}$ of the derivatives $f^{(k)}(x_i)$ at the nodes.

Notice that a result similar to that of Proposition 5.11 holds for any distribution of nodes, provided the Lebesgue constant associated with those nodes may be bounded appropriately.

In the next section we apply the results of Theorem 5.5 for constructing rational finite difference formulas.

5.3 Rational Finite Difference (RFD) Formulas

Finite difference (FD) methods based on polynomial interpolants have a long tradition and are still the subject of much investigation; see, for instance, [47,

48, 49, 50]. The idea behind the corresponding formulas is the approximation of the kth order derivative of f at some point $\xi \in [a, b]$ by the kth derivative of the polynomial interpolant p_n of f at ξ. In view of the linearity of p_n in the data $f_0, \ldots, f_n$, an FD formula is obtained as follows:

$$f^{(k)}(\xi) \approx p_n^{(k)}(\xi) = \sum_{j=0}^{n} c_j f_j.$$

For the determination of the weights c_j, Fornberg has presented a very efficient algorithm, which requires an average of four operations per weight. (Weights of finite differences must be distinguished from the weights of the barycentric form of rational interpolants.)

As an application of Theorem 5.5 and Proposition 5.11, we now introduce rational finite difference[2] (RFD) formulas for the approximation of the kth derivative of a sufficiently smooth function. For the approximation at a node, we compute

$$f^{(k)}(x_i) \approx r_n^{(k)}(x_i) = \sum_{j=0}^{n} D_{ij}^{(k)} f_j, \tag{5.48}$$

where $D_{ij}^{(k)}$ is the kth derivative at the node x_i of the jth Lagrange fundamental rational function (2.14). At an intermediate point $\xi \in [a, b]$, we consider

$$f^{(k)}(\xi) \approx R_n^{(k)}(\xi) = \frac{\sum_{i=0}^{n} \frac{w_i}{\xi - x_i} f_i^{(k)}}{\sum_{i=0}^{n} \frac{w_i}{\xi - x_i}} = \sum_{j=0}^{n} \frac{\sum_{i=0}^{n} \frac{w_i}{\xi - x_i} D_{ij}^{(k)}}{\sum_{i=0}^{n} \frac{w_i}{\xi - x_i}} f_j,$$

which is also a finite difference formula, since the coefficients in the linear combination of the f_j are constant for fixed ξ.

The methods presented here may be based on any linear barycentric rational interpolant. We nevertheless focus on the Floater–Hormann family of barycentric rational interpolants.

In order to establish formulas for the weights $D_{ij}^{(k)}$ in (5.48), we use the centro-skew symmetric differentiation matrix $D^{(1)}$ with elements

$$D_{ij}^{(1)} := \begin{cases} \frac{w_j}{w_i} \frac{1}{x_i - x_j}, & i \neq j, \\ -\sum_{\ell=0, \ell \neq i}^{n} D_{i\ell}^{(1)}, & i = j, \end{cases} \tag{5.49}$$

[2]The idea of these formulas is originally due to Jean-Paul Berrut.

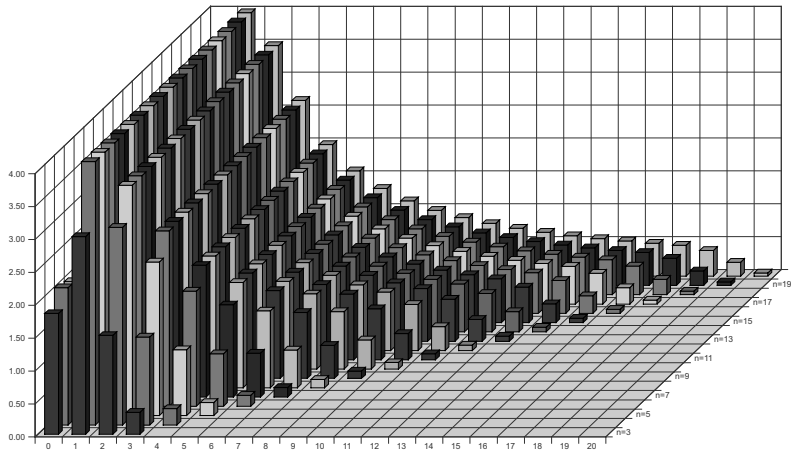

Figure 5.1: Absolute values of the weights for *one-sided* RFD formulas with $d = 3$ and $n = 3, \ldots, 20$ for the approximation of the first derivative at $x = 0$ on an integer grid, $x_0 = 0, \ldots, x_n = n$.

introduced in [5] for the first order derivative and similarly [58, 108, 109],

$$D_{ij}^{(k)} := \begin{cases} \frac{k}{x_i - x_j}\left(\frac{w_j}{w_i} D_{ii}^{(k-1)} - D_{ij}^{(k-1)}\right) \\ \qquad = k\Big(D_{ij}^{(1)} D_{ii}^{(k-1)} - \frac{D_{ij}^{(k-1)}}{x_i - x_j}\Big), & i \neq j, \\ -\sum_{\ell=0, \ell \neq i}^{n} D_{i\ell}^{(k)}, & i = j, \end{cases} \tag{5.50}$$

for higher order derivatives. The formulas for these differentiation matrices were originally derived in [4, 5] from Proposition 11 given by Schneider and Werner in [97]. The negative sum, which guarantees exact derivatives of constants, should always be used for the computation of the diagonal elements of such matrices as this is crucial for better accuracy and stability [4].

From (5.48) it is obvious that the approximations of $f^{(k)}(x_i)$ at all the nodes can be computed at once in the form of a vector by multiplying the vector $\mathbf{f}$ of all the values f_j from the left by $D^{(k)}$.

The weights for the approximation of the kth derivative at the node x_i are given by the elements of the $(i+1)$st row of $D^{(k)}$. As each element of the ith row of $D^{(k)}$ depends only on three elements of the ith row in $D^{(1)}$ and $D^{(k-1)}$,

Table 5.1: Weights for *one-sided* RFD formulas with $d = 4$ and $n = 4, 5, 6, 7$ for the approximation of the first four derivatives at $x = 0$ on an integer grid, $x_0 = 0, \ldots, x_n = n$.

0	1	2	3	4	5	6	7
1*st derivative (order 4)*: $D_{0j}^{(1)}$, $j = 0, \ldots, n$,							
$-\frac{25}{12}$	4	-3	$\frac{4}{3}$	$-\frac{1}{4}$			
$-\frac{137}{60}$	5	-5	$\frac{10}{3}$	$-\frac{5}{4}$	$\frac{1}{5}$		
$-\frac{9}{4}$	5	$-\frac{11}{2}$	$\frac{14}{3}$	$-\frac{11}{4}$	1	$-\frac{1}{6}$	
$-\frac{949}{420}$	5	$-\frac{11}{2}$	5	$-\frac{15}{4}$	$\frac{11}{5}$	$-\frac{5}{6}$	$\frac{1}{7}$
2*nd derivative (order 3)*: $D_{0j}^{(2)}$, $j = 0, \ldots, n$,							
$\frac{35}{12}$	$-\frac{26}{3}$	$\frac{19}{2}$	$-\frac{14}{3}$	$\frac{11}{12}$			
$\frac{15}{4}$	$-\frac{77}{6}$	$\frac{107}{6}$	-13	$\frac{61}{12}$	$-\frac{5}{6}$		
$\frac{319}{90}$	$-\frac{25}{2}$	$\frac{77}{4}$	$-\frac{161}{9}$	11	$-\frac{41}{10}$	$\frac{25}{36}$	
$\frac{379}{105}$	$-\frac{529}{42}$	$\frac{8129}{420}$	$-\frac{809}{42}$	$\frac{211}{14}$	$-\frac{1903}{210}$	$\frac{293}{84}$	$-\frac{127}{210}$
3*rd derivative (order 2)*: $D_{0j}^{(3)}$, $j = 0, \ldots, n$,							
$-\frac{5}{2}$	9	-12	7	$-\frac{3}{2}$			
$-\frac{17}{4}$	$\frac{71}{4}$	$-\frac{59}{2}$	$\frac{49}{2}$	$-\frac{41}{4}$	$\frac{7}{4}$		
$-\frac{2129}{600}$	$\frac{47}{3}$	$-\frac{3553}{120}$	$\frac{476}{15}$	$-\frac{2519}{120}$	$\frac{613}{75}$	$-\frac{57}{40}$	
$-\frac{22363}{5880}$	$\frac{229}{14}$	$-\frac{1221}{40}$	$\frac{1465}{42}$	$-\frac{1641}{56}$	$\frac{1287}{70}$	$-\frac{1223}{168}$	$\frac{631}{490}$
4*th derivative (order 1)*: $D_{0j}^{(4)}$, $j = 0, \ldots, n$,							
1	-4	6	-4	1			
3	-14	26	-24	11	-2		
$\frac{1774}{1125}$	$-\frac{83}{10}$	$\frac{2827}{150}$	$-\frac{5383}{225}$	$\frac{451}{25}$	$-\frac{5741}{750}$	$\frac{637}{450}$	
$\frac{9701}{4410}$	$-\frac{3127}{294}$	$\frac{33253}{1470}$	$-\frac{26069}{882}$	$\frac{2719}{98}$	$-\frac{27577}{1470}$	$\frac{6901}{882}$	$-\frac{2113}{1470}$

Table 5.2: Weights for *centered* RFD formulas with $d = 4$ and $n = 4, 6, 8$ for the approximation of the first four derivatives at $x = 0$ on an integer grid, $x_0 = -n/2, \ldots, x_n = n/2$.

-4	-3	-2	-1	0	1	2	3	4
1st derivative (order 4): $D^{(1)}_{n/2,j}$, $j = 0, \ldots, n$,								
		$\frac{1}{12}$	$-\frac{2}{3}$	0	$\frac{2}{3}$	$-\frac{1}{12}$		
	$-\frac{1}{42}$	$\frac{5}{28}$	$-\frac{11}{14}$	0	$\frac{11}{14}$	$-\frac{5}{28}$	$\frac{1}{42}$	
$\frac{1}{64}$	$-\frac{5}{48}$	$\frac{11}{32}$	$-\frac{15}{16}$	0	$\frac{15}{16}$	$-\frac{11}{32}$	$\frac{5}{48}$	$-\frac{1}{64}$
2nd derivative (order 3): $D^{(2)}_{n/2,j}$, $j = 0, \ldots, n$,								
		$-\frac{1}{12}$	$\frac{4}{3}$	$-\frac{5}{2}$	$\frac{4}{3}$	$-\frac{1}{12}$		
	$\frac{1}{63}$	$-\frac{5}{28}$	$\frac{11}{7}$	$-\frac{355}{126}$	$\frac{11}{7}$	$-\frac{5}{28}$	$\frac{1}{63}$	
$-\frac{1}{128}$	$\frac{5}{72}$	$-\frac{11}{32}$	$\frac{15}{8}$	$-\frac{1835}{576}$	$\frac{15}{8}$	$-\frac{11}{32}$	$\frac{5}{72}$	$-\frac{1}{128}$
3rd derivative (order 2): $D^{(3)}_{n/2,j}$, $j = 0, \ldots, n$,								
		$-\frac{1}{2}$	1	0	-1	$\frac{1}{2}$		
	$\frac{109}{588}$	$-\frac{365}{294}$	$\frac{1133}{588}$	0	$-\frac{1133}{588}$	$\frac{365}{294}$	$-\frac{109}{588}$	
$-\frac{1763}{12288}$	$\frac{2845}{3072}$	$-\frac{17017}{6144}$	$\frac{3415}{1024}$	0	$-\frac{3415}{1024}$	$\frac{17017}{6144}$	$-\frac{2845}{3072}$	$\frac{1763}{12288}$
4th derivative (order 1): $D^{(4)}_{n/2,j}$, $j = 0, \ldots, n$,								
		1	-4	6	-4	1		
	$-\frac{109}{441}$	$\frac{365}{147}$	$-\frac{1133}{147}$	$\frac{4826}{441}$	$-\frac{1133}{147}$	$\frac{365}{147}$	$-\frac{109}{441}$	
$\frac{1763}{12288}$	$-\frac{2845}{2304}$	$\frac{17017}{3072}$	$-\frac{3415}{256}$	$\frac{327787}{18432}$	$-\frac{3415}{256}$	$\frac{17017}{3072}$	$-\frac{2845}{2304}$	$\frac{1763}{12288}$

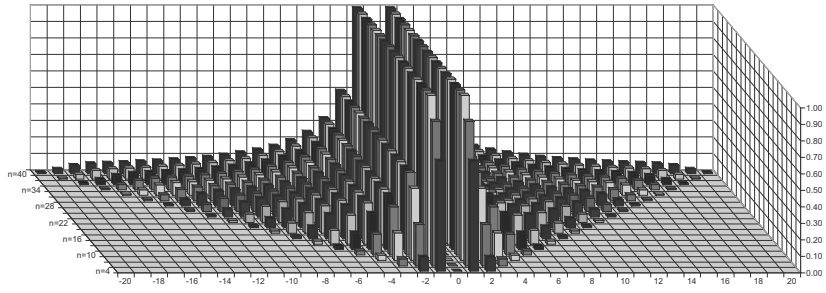

Figure 5.2: Absolute values of the weights for *centered* RFD formulas with $d = 3$ and $n = 4, 6, \ldots, 40$ for the approximation of the first derivative at $x = 0$ on an integer grid, $x_0 = -n/2, \ldots, x_n = n/2$.

the computation of every additional RFD weight will require an average of four operations, as in the polynomial case.

In Tables 5.1 and 5.2, we list the weights for the first left one-sided ($i = 0$) and centered ($i = n/2$) RFD formulas on an equispaced grid for $d = 4$, with a mesh size $h = 1$. With a different spacing, the weights must be divided by h^k. Since the differentiation matrices are centro-skew-symmetric for odd k and centro-symmetric for even k, the weights of the right one-sided RFD formulas are the same as those of the left one-sided RFD ($i = n$), though taken from right to left and multiplied by $(-1)^k$. Observe that for $d = n$ the RFD weights are the same as the FD weights. This is due to the fact that, in this special case, the rational interpolant r_n is the polynomial interpolant. Polynomial FD formulas indeed are a special case of their rational analogues.

Figures 5.1 and 5.2, show the absolute values of the weights of the first one-sided and centered RFD formulas with $d = 3$ for the approximation of the first derivative at $x = 0$ on an equispaced grid. Let us discuss these values and compare them with those obtained from polynomials in [48]; see Figures 3.2-1 and 3.2-2 in that reference.

The difference between polynomial FD and RFD is most striking with *one-sided* FD weights. In the polynomial case, the weights grow exponentially at the center of the interval with increasing order or number of points [48]. The weights presented here behave much more favourably. To see this, we recall that the quotient of the largest to the smallest Floater–Hormann interpolation weight [46]

is less than or equal to 2^d if the nodes are equispaced. Equation (5.49) shows that the one-sided RFD weights for the approximation of the first derivative oscillate in sign and decrease in absolute value (relatively) at least as $1/(i-j)$, where i is the index of the maximal weight. Thus the weight $D_{00}^{(1)}$ is neither the largest nor the smallest. This yields the following property:

$$\frac{1}{b-a} \leq |D_{0j}^{(1)}| \leq \frac{2^d}{h}, \quad j = 0, \ldots, n,$$

for every n and d, $d \leq n$. For the approximation of higher order derivatives, (5.50) shows that the maximal weight is roughly bounded by $O(k!2^{kd}/h^k)$ for every n and the neighbouring weights decrease again as $1/(i-j)$ in absolute value. The rather small values of the weights influence positively the numerical stability of the computation of one-sided RFD approximations as compared to their polynomial analogues.

An additional advantage of one-sided RFD is the fact that the maximal weight has index less than $n/2$. This is very favourable since, as the derivative of a function is a local property, it is not natural to give much importance to function values located too far away from the point of interest.

In the *centered* case, the RFD weights behave similarly to the polynomial ones. For the approximation of the first derivative, they are bounded by 1 and decrease in absolute value as $1/(i-j)$ from the maximal weight with index i. For higher order derivatives, the RFD weights also decrease in absolute value, like $1/(i-j)$ for odd and $1/(i-j)^2$ for even order derivatives; see the first formula in (5.50) and observe that $D_{\frac{n}{2}\frac{n}{2}}^{(k)} = 0$ when both k and n are even.

5.4 Numerical Results

To illustrate our theoretical results, we begin this section with the approximation of the first and second order derivatives of functions at the nodes and at intermediate points.

We start with an example of [46], namely the interpolation of the function $f_1(x) = \sin(x)$ for $x \in [-5, 5]$. We use the rational interpolant with $d = 4$ and f_1 sampled at equispaced nodes. Our aim is to survey the estimated approximation orders of the derivatives of the interpolant and compare them with the results obtained for the interpolant itself. We compute the error at the same eleven nodes for different values of n. Table 5.3 shows the errors and orders for the first and second derivatives. Comparing these results with the approximated

Table 5.3: Error in the derivatives of r_n with $d = 4$ interpolating f_1 at equispaced nodes.

	first derivative		second derivative	
n	error	order	error	order
10	1.2e−01		5.0e−01	
20	5.2e−03	4.5	4.5e−02	3.5
40	1.9e−04	4.7	3.3e−03	3.8
80	7.2e−06	4.7	2.5e−04	3.7
160	2.9e−07	4.6	2.1e−05	3.6
320	1.3e−08	4.5	1.9e−06	3.4
640	6.8e−10	4.3	1.9e−07	3.3

Table 5.4: Error in the derivatives of r_n with $d = 3$ interpolating f_2 at equispaced nodes.

	first derivative		second derivative	
n	error	order	error	order
10	4.1e−01		1.5e+00	
20	3.3e−02	3.6	2.7e−01	2.5
40	9.4e−05	8.5	1.6e−03	7.4
80	1.9e−06	5.7	7.2e−05	4.5
160	1.4e−07	3.7	1.4e−05	2.3
320	1.2e−08	3.5	2.3e−06	2.7
640	1.5e−09	3.0	3.1e−07	2.9

orders in [46], we see that the order decreases almost exactly by one unit at every differentiation.

With the next example we study the convergence rates at intermediate points. For that purpose, we sampled Runge's function $f_2(x) = 1/(1 + x^2)$ at equispaced nodes in the interval $[-5, 5]$. We chose $d = 3$ and computed the maximum error at 1000 equispaced points inside the interval which are not nodes. Table 5.4 displays our results, which illustrate Theorems 5.6 and 5.8 in this particular case.

Table 5.5: Error in the derivatives of r_n with $d = 3$ interpolating f_3 between Chebyshev points of the second kind.

	first derivative		second derivative	
n	error	order	error	order
10	2.8e−01		2.0e+01	
20	7.7e−02	1.9	2.0e+00	3.3
40	1.2e−02	2.7	5.9e−01	1.7
80	1.5e−03	3.0	1.6e−01	1.9
160	2.0e−04	2.9	3.9e−02	2.0
320	2.4e−05	3.0	9.9e−03	2.0
640	3.0e−06	3.0	2.5e−03	2.0

The results of this chapter might be applied to the numerical solution of differential equations. For this reason we experiment with the exact solution of a model problem from Stoer and Bulirsch [105], adapted to the interval $[-1, 1]$, namely

$$f_3(x) = \frac{e^{-20}}{1+e^{-20}} e^{10(x+1)} + \frac{1}{1+e^{-20}} e^{-10(x+1)} - \cos^2\Big(\frac{\pi}{2}(x+1)\Big).$$

This time, we sample the function at Chebyshev points of the second kind and interpolate the computed values using the rational interpolant with $d = 3$. Table 5.5 shows the maximum error at 1000 equispaced points and the experimental convergence rates. Again, the kth derivative of the rational interpolant converges at the rate of $O(h^{d+1-k})$ as $h \to 0$ in the cases $k = 1, 2$. We also supply a graphical survey of this same experiment at even values of n in Figure 5.3. In a log-log scale, the error curves in the approximation of the first two derivatives of f_3 are added to those of its rational interpolant. For $n \geq 20$ the curves are nearly straight lines of slopes -4, -3 and -2.

We sampled all three functions at equispaced nodes and at Chebyshev points of the second kind. The experimental convergence rates, in the cases not displayed in Tables 5.3, 5.4 and 5.5, are very similar and thus omitted.

We repeated the computation with f_2, this time using the cubic spline with not-a-knot end conditions. Since Runge's function is analytic, the spline interpolant and its derivatives have the same convergence orders as the rational interpolant with $d = 3$ and its derivatives; see [33]. Table 5.6 reveals that the

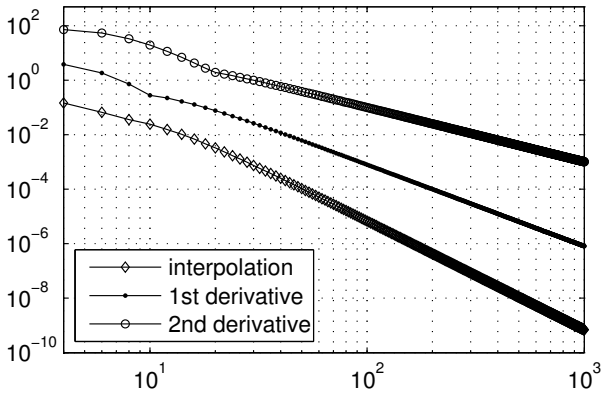

Figure 5.3: Errors in the rational interpolation with $d = 3$ of f_3 sampled at $4 \leq n \leq 1000$ Chebyshev points of the second kind in $[-1, 1]$ and approximation of its first and second derivatives.

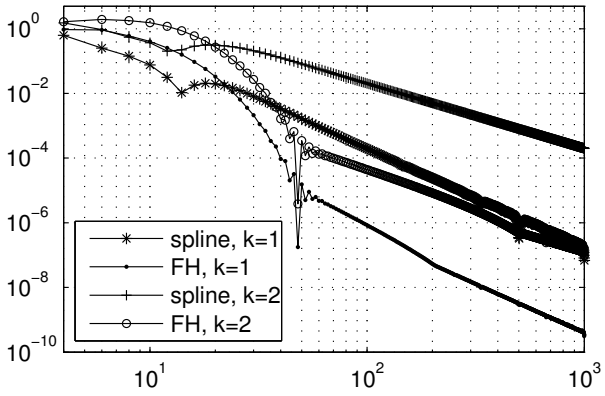

Figure 5.4: Errors in the spline and rational (FH) approximations with $d = 3$ of the first ($k = 1$) and second ($k = 2$) derivatives of f_2 sampled at $4 \leq n \leq 1000$ equispaced points in $[-5, 5]$.

Table 5.6: Error in the derivatives of the cubic spline interpolating f_2.

	first derivative		second derivative	
n	error	order	error	order
10	7.6e−02		3.7e−01	
20	2.0e−02	1.9	2.9e−01	0.3
40	3.4e−03	2.5	1.1e−01	1.4
80	3.9e−04	3.1	2.5e−02	2.2
160	4.7e−05	3.0	6.1e−03	2.0
320	5.9e−06	3.0	1.5e−03	2.0
640	7.1e−07	3.1	3.8e−04	2.0

experimental orders coincide for large enough n, but the error in the rational function is considerably smaller than that with the spline. The difference is due to the faster error decay of the derivatives of the rational interpolant for small values of n. Figure 5.4 confirms this observation: For $n \geq 50$ the curves corresponding to the errors in the spline and rational approximations of the first respectively second derivative of f_2 are almost parallel.

Let us now focus on the approximation of derivatives via finite differences and rational finite differences.

We illustrate Theorem 5.5 and the observations from Section 5.3 concerning the weights involved in polynomial and rational FD approximation. To this end, we investigate the approximation of the second and fourth order derivatives of Runge's original example f_2 at the nodes $x = -5$ and $x = 0$, and the modified example $f_4(x) = 1/(1+25x^2)$ at $x = 0$. We sampled them both at odd numbers of equispaced points, f_2 in the interval $[-5, 5]$, respectively $[0, 5]$ for one-sided FD approximation, and f_4 in $[-5, 5]$ for centered FD approximation. In the rational interpolant r_n, we chose $d = 4$, the minimal value according to Theorem 5.5, to guarantee decreasing errors in the RFD approximation of the fourth order derivative.

Our aim was to observe estimated approximation orders of RFD approximations and to compare the error behaviours of polynomial and rational FD. For every example we computed the absolute error in the various approximations. We perform this survey graphically on a log-log scale to avoid less informative tables and to display more details and data.

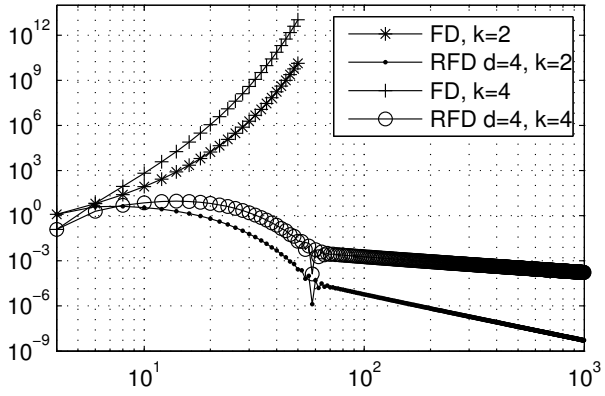

Figure 5.5: Errors in the *one-sided* FD, respectively RFD, approximations at $x = -5$ (with $d = 4$ and $n = 4, \ldots, 1000$) of the second and fourth order derivatives of f_2 sampled in $[-5, 5]$.

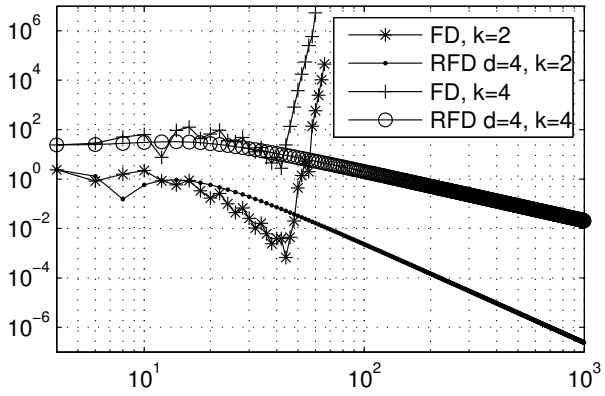

Figure 5.6: Errors in the *one-sided* FD, respectively RFD, approximations at $x = 0$ (with $d = 4$ and $n = 4, \ldots, 1000$) of the second and fourth order derivatives of f_2 sampled in $[0, 5]$.

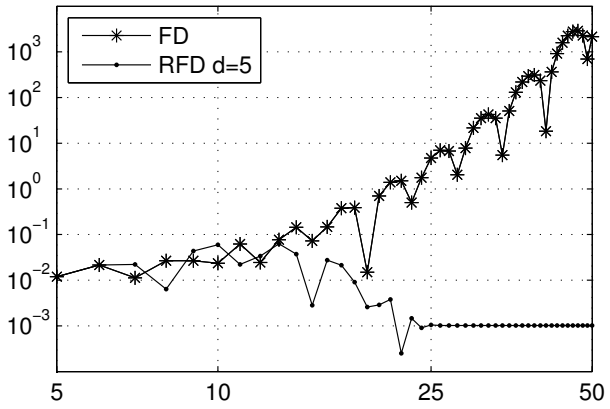

Figure 5.7: Errors in the *one-sided* FD, respectively RFD, approximations at $x = 0$ (with $d = 5$ and $n = 5, \ldots, 50$) of the first order derivative of $\exp(-50(x - 0.4)^2) + \sinh(x)$ sampled in $[0, 0.04n]$.

Figure 5.5 displays the absolute errors in *one-sided* polynomial FD and RFD approximations at $x = -5$ of the second ($k = 2$) and fourth ($k = 4$) order derivatives of f_2. The error in polynomial FD grows very rapidly with increasing values of n. This failure arises from Runge's phenomenon and the large growth rates of the polynomial FD weights [48]; for $n = 140$, the largest absolute value of these weights for the approximation of the fourth order derivative is 7.6e+41. In RFD with $d = 4$, in contrast, the largest value is only 28.6. From the slopes of the curves, we see that the errors in RFD approximation of the second and fourth order derivatives of f_2 decrease at an experimental rate of 3, respectively 1, which is the lower bound $d + 1 - k$ on the order predicted in Theorem 5.5. The good quality of approximation in the present example is also to be expected in RFD approximation at nodes near the ends of the interval.

In the next example, namely *one-sided* FD and RFD approximations at $x = 0$ of the same derivatives of f_2 as above, Runge's phenomenon does not appear, only the bad conditioning of one-sided polynomial FD makes the error grow as n becomes larger than 45; see Figure 5.6. The errors in RFD approximation are larger until this value of n, but, as n increases, they keep decreasing at experimental rates of 4, respectively 2, which is one more unit than predicted

by the upper bound (5.21) on the error.

We performed a similar study with the analytic function $f(x) = \exp(-50(x-0.5)^2)$; namely we sampled it in $[0,1]$ and compared *one-sided* FD and RFD approximations at $x = 0$ of the first and third derivatives for various values ≤ 6 of d and for n between d and 1000. The convergence rates of the RFD approximation from Theorem 5.5 are observable as soon as $n \geq 15$, whereas the relative error in polynomial FD is larger than one with small values of n and starts diverging when n exceeds 50.

The absolute errors in *centered* polynomial FD and RFD approximations of the second and fourth order derivatives of f_4 at $x = 0$ nearly coincide. We thus omit plotting them. While the error in polynomial FD decays exponentially, as expected since there is no Runge phenomenon in the middle of the interval, the error in the RFD approximation decreases much faster than predicted in this particular example for n between 50 and 450. For larger values of n, the absolute errors start oscillating. Other examples with *centered* RFD yield the expected algebraic decay of the error as the number of nodes increases, while the errors in polynomial *centered* FD always decayed exponentially for smooth functions.

Another approach to finite difference approximation is to keep n constant and small, and decrease h as much as needed to reach the desired accuracy. In this setting, the classical FD methods are best since their weights have roughly the same magnitude, the Runge phenomenon disappears due to the shrinkage of the interval—the regions, where the function needs to be analytic for the polynomial interpolant to converge, shrink—and every reasonable continuous function can be represented by a polynomial of small degree in a short interval.

Finally we performed the converse experiment: We choose h fixed and increase n, a situation which may arise when the function cannot be sampled with arbitrarily small resolution. The first derivative of $f(x) = \exp(-50(x-0.4)^2) + \sinh(x)$ was approximated at $x = 0$ using $n+1$ function values in the interval $[0, 0.04n]$. Increasing the number of nodes and thus the information about the function does not help to gain any accuracy when using polynomial FD methods; see Figure 5.7. With RFD and the proper choice of d, it is possible to further decrease the error in the approximation of the derivative in this example. The methods presented in Section 4.2.4 may be used for the determination of a good d.

Chapter 6

Quadrature Rules

Linear interpolation schemes very naturally lead to quadrature rules. In this chapter we present the construction of two linear rational schemes for the approximation of antiderivatives, which simplify to linear rational quadrature rules when evaluated at the right end of the interval. The first kind is obtained through the direct numerical integration of the Lagrange fundamental rational functions; the other is based on the solution of a simple boundary value problem.

We begin this chapter with a short introduction to quadrature rules and state a selection of relevant and known results. In Section 6.2 we explain in more detail how to obtain quadrature rules from barycentric rational interpolants in general and thereafter, in Section 6.3, we present our concepts for the approximation of antiderivatives and integrals of differentiable functions. The convergence order of the quadrature rule obtained from integrating Floater–Hormann interpolants is studied and shown to be one unit larger than that of the interpolation, if the nodes are equispaced. The efficiency of the various approaches is demonstrated with numerical tests. Finally we apply our convergence theory for Floater–Hormann interpolation of analytic functions from Section 4.2 to numerical quadrature and establish in Section 6.4 even more efficient rules for functions whose nearest singularity to the interval is known.

This chapter is essentially built upon the results presented in [75], whereas the last part on the efficient rules for analytic functions will be published in [56]. Most of the general theory on quadrature rules presented in the introduction stems from [21].

6.1 Introduction

We suppose again that we are given the discrete data $f_0, \ldots, f_n$, corresponding to a real or complex function f which is defined and integrable on a bounded interval $[a, b]$ and sampled at strictly ordered nodes in $[a, b]$. Our aim is to either approximate the (definite) integral

$$I := \int_a^b f(x)\,\mathrm{d}x \tag{6.1}$$

by a quadrature rule $Q_n = \sum_{k=0}^{n} \omega_k f_k$, where the ω_k are called the quadrature weights, or to approximate an antiderivative (primitive) of f. A more general setting would be the approximation of the integral in (6.1) with f multiplied by some positive weight function; see, e.g., [32, 78]. We shall concentrate here only on the special case when that weight function is equal to 1. No fundamental reason would hinder the extension of the study from this chapter to integrals with nontrivial weight functions.

Let us begin by reviewing some basic but important results on numerical quadrature. These can be found, e.g., in [21, 32, 78].

The most straightforward idea leading to quadrature rules is to integrate the unique polynomial interpolant of degree at most n of the given data. The quadrature weights are then simply given by the exact integral of the fundamental Lagrange functions (2.2). Such quadrature rules are called interpolatory quadrature rules and Newton–Cotes rules in the special case of equispaced nodes. The most common Newton–Cotes rules are the trapezoid rule (2 nodes, convergence order 3), Simpson's rule (3 nodes, convergence order 5), Simpson's $\frac{3}{8}$ rule (4 nodes, convergence order 5) and Boole's rule (5 nodes, convergence order 7). These methods are not suited for long intervals. Subdividing the interval into subintervals and defining a Newton–Cotes rule on each of these leads to composite Newton–Cotes rules, which are very simple and often perform sufficiently well. The error formula for the composite trapezoid rule, the Euler–McLaurin formula [11, 61], shows that this very simple method is very accurate if the integrand and its derivatives are periodic.

An important characterisation of quadrature rules is their degree of precision [69], also called degree of accuracy [78]. A quadrature rule has degree of precision m if it is exact for all polynomials in $\mathcal{P}_m$ and if there exists at least one polynomial in $\mathcal{P}_{m+1}$ for which the rule is not exact.

The positivity of the quadrature weights also plays an important role. It can be shown that quadrature rules with positive weights have many good proper-

ties; here are just a few. The first, taken from [86] (see also [21]), deals with the convergence of such a rule.

Theorem 6.1. *Let $\{Q_n\}$ be a sequence of interpolatory quadrature rules or a sequence of quadrature rules which converge for all polynomials. If all the weights are positive, then $\{Q_n\}$ converges for all Riemann-integrable functions.*

The weights of the Newton–Cotes rules are not all positive. Their asymptotic behaviour has been studied in [82, 86]. It can be shown that for these rules $\lim_{n\to\infty} |\omega_k| = \infty$ except for $k = 1, 2$, so that the rules do not converge to I as n increases. The author of [21] concludes that this reveals "die praktische Unbrauchbarkeit dieser Verfahren" (meaning the uselessness of these rules in practice). In addition to the unfavourable properties of the weights, it can be shown [86] that Newton–Cotes rules also diverge for some analytic functions, as is also the case for polynomial interpolation with equispaced nodes; see also Section 4.1 and Section 6.4 for quadrature rules obtained from integrating linear barycentric rational interpolants of analytic functions.

The positivity of the weights and the degree of precision are related by the following result [21].

Theorem 6.2. *A quadrature rule with degree of precision m has at least $\lfloor \frac{m}{2} \rfloor + 1$ positive weights.*

Moreover, the approximation error of a given quadrature rule can be compared with the best approximation via the degree of precision and the positivity of the weights [32, 78].

Theorem 6.3. *A quadrature rule Q_n with degree of precision m satisfies*

$$\Big| \int_a^b f(x)\,\mathrm{d}x - Q_n \Big| \leq \Big((b-a) + \sum_{k=0}^{n} |\omega_k| \Big) e_m^*,$$

where $e_m^* = \inf\{\|p_m - f\| : p_m \in \mathcal{P}_m\}$. If all the quadrature weights are nonnegative, then

$$\Big| \int_a^b f(x)\,\mathrm{d}x - Q_n \Big| \leq 2(b-a) e_m^*.$$

The condition of a quadrature rule also directly depends on the positivity of the weights; see, e.g., [66]. If the data f_k are perturbed to $\widetilde{f}_k$, such that all $|\widetilde{f}_k - f_k| \leq \varepsilon_p$ for some positive ε_p, then

$$\Big| \sum_{k=0}^{n} \omega_k f_k - \sum_{k=0}^{n} \omega_k \widetilde{f}_k \Big| \leq \varepsilon_p \sum_{k=0}^{n} |\omega_k|. \tag{6.2}$$

If the weighs are all positive and if the quadrature rule integrates constant functions exactly, then this upper bound is simply $\varepsilon_p(b-a)$, whereas it is necessarily larger if the weights are not all positive.

From the discussion in Section 4.3 of the appropriate choice of nodes for polynomial interpolation, we know that Chebyshev points of the first and second kinds are well suited for such interpolation, whereas equispaced nodes are not; see also [30] for a study of analytic integrands and the included historic part at the end. Therefore it is not surprising that the same is true for quadrature rules, called Clenshaw–Curtis rules [27] when the nodes are Chebyshev points. For Clenshaw–Curtis methods, Chebyshev points of the second kind are called practical abscissas [32, 78] while Chebyshev points of the first kind are the classical ones. Methods based on the latter are sometimes called Fejér methods [78] or Pólya methods [21]. Their degree of precision is n and it was shown, among other things in [44, 68], that the weights are all positive so that by Theorem 6.1 these methods converge for all Riemann-integrable functions. The speed of convergence and other interesting properties of Fejér and Clenshaw–Curtis quadrature are investigated in [127]; a fast construction of these methods is described in [120].

Interpolatory quadrature rules have a total of $2n + 2$ degrees of freedom, namely the nodes and the weights. It is therefore obviously possible to construct methods that have degree of precision $2n+1$. On the other hand, one can show that no interpolatory rule exists that integrates exactly all polynomials from $\mathcal{P}_{2n+2}$; see, e.g., [21]. Quadrature rules that attain the maximum degree of precision are called Gauss rules [53]. In our case, when the weight function is equal to one, if $[a, b] = [-1, 1]$ the rule is called Gauss–Legendre and the nodes are the roots of the $(n + 1)$st Legendre polynomial P_{n+1}; for various representations, see [32]. It follows from Theorem 6.2 that the weights in Gauss rules are all positive. The convergence speed for integrands that are several times differentiable or analytic is described and proven in [112], where the author mainly compares the Gauss–Legendre rule with Clenshaw–Curtis quadrature and observes that in most cases both methods perform almost equally well; see also [127].

Gauss quadrature integrates exactly polynomials of the highest possible degree. However, these methods do not include the endpoints of the interval $[a, b]$. Quadrature rules with maximum degree of precision, $2n-1$ in this case, including the ends of the interval, are the Lobatto rules whose nodes are the roots of $P_{n+1} - P_{n-1}$, where P_n is again the nth Legendre polynomial. These methods are implemented as adaptive rules in the MATLAB built-in command `quadl`; more details about adaptive quadrature may be found in [54]. A successful

alternative is of course Clenshaw–Curtis quadrature.

Gauss-type quadrature rules that integrate rational functions exactly have been investigated; see, e.g., [36, 37, 119] and the references therein. Such quadrature rules, require the knowledge of the poles of the rational function. Related approaches are explained, e.g., in [35, 51, 117, 126], where the rational interpolant is supposed to simulate the poles of the integrand, so that again the poles are fixed in advance and do not depend on n.

If, for the approximation of I from (6.1), we are free to choose the set of nodes at which the function f is to be sampled, we can opt for any efficient distribution of points. The situation is different when the nodes cannot be chosen. If the data set stems from measurements, for instance, it is most likely that these are taken on a regular grid. As we have seen, Newton–Cotes quadrature rules diverge or are unstable with a growing number of points. One way to avoid problems is using Gregory rules [21] or composite Newton–Cotes rules of low order such as the composite trapezoid or Simpson rules. Their frequent use in practical calculations documents the importance of these slowly converging formulas for nonperiodic functions; see [32] and the included (not very serious) reference to M. Abramowitz, that "95% of all practical work in numerical analysis boiled down to applications of Simpson's rule and linear interpolation".

Any attempt to construct geometrically converging interpolants from equispaced data necessarily fails, as it leads to the Gibbs and Runge phenomena [85]. It is plausible that the same is true for interpolatory quadrature rules from nonperiodic equispaced samples. The interpolation schemes for equispaced nodes presented in [85] as well as those from [67] may also be used to derive quadrature rules. Other ideas have been pursued; see, e.g., [66].

In what follows, we present methods for the approximation of antiderivatives and integrals. We analyse some of their properties for equispaced points. The methods obtained can be called rational interpolatory quadrature rules; see [36, 117] and the references therein for other such methods.

6.2 Integration of Barycentric Rational Interpolants

Every linear interpolant with no poles in $[a, b]$,

$$\sum_{k=0}^{n} \gamma_k(x) f_k \approx f(x),$$

trivially leads to a linear quadrature rule through the integration of the factors γ_k. The behaviour of the so-obtained rule regarding convergence and stability simply follows from the respective properties of the interpolant. In the case of an $(n+1)$-point linear rational interpolant (2.8) with arbitrary nonzero weights w_k, we have

$$I = \int_a^b f(x)\,\mathrm{d}x \approx \int_a^b r_n(x)\,\mathrm{d}x = \int_a^b \frac{\sum_{k=0}^{n} \frac{w_k}{x-x_k} f_k}{\sum_{k=0}^{n} \frac{w_k}{x-x_k}}\,\mathrm{d}x = \sum_{k=0}^{n} \omega_k f_k =: Q_n, \tag{6.3}$$

where

$$\omega_k := \int_a^b \frac{\frac{w_k}{x-x_k}}{\sum_{k=0}^{n} \frac{w_k}{x-x_k}}\,\mathrm{d}x \tag{6.4}$$

is the integral of the kth Lagrange fundamental rational function. For the pointwise approximation of antiderivatives, it is enough to replace $\int_a^b$ by $\int_a^x$ in (6.3) and (6.4), and to proceed analogously as above. If r_n is a true rational interpolant with nonconstant denominator, then the so-called quadrature weights ω_k can be easily determined in exact arithmetic only if the poles are known.

The choice $w_k = \lambda_k$ of (2.5) in (6.3) reproduces the Newton–Cotes rules. The same is true if $d = n$ in the Floater–Hormann interpolant, since it then coincides with the interpolating polynomial.

For the computation of the weights (6.4), we decided to neglect algebraic methods as they mostly require the polynomials in the numerator and denominator of r_n to be in canonical form. The step from the representation (2.8) of these polynomials to the canonical one is impeded by stability problems [61].

For a rational interpolant whose denominator degree exceeds 4 there is no formula for the poles, and the question of the location of the poles of Floater–Hormann interpolation is not yet settled; see also Section 2.3.3. As we wish to avoid approximating complex poles and determining expensive partial fraction decompositions, we pursue two ideas for generating linear quadrature rules

based on linear rational interpolants. If the location of the poles were known, techniques like those described in [37] could be applied to r_n.

Equation (6.2) gives a bound on the condition number of a quadrature rule involving the quadrature weights. If the quadrature rule is obtained from a barycentric interpolant, it is natural that the condition number of the quadrature rule may also be characterised via the Lebesgue constant associated with the interpolant, which we investigate in Section 3.3. Indeed, with the expression for the weights from (6.4), it follows that

$$\begin{aligned}\sum_{k=0}^{n} |\omega_k| &= \sum_{k=0}^{n} \left| \int_a^b \frac{\frac{w_k}{x-x_k}}{\sum_{k=0}^{n} \frac{w_k}{x-x_k}} \,\mathrm{d}x \right| \\ &\leq \int_a^b \sum_{k=0}^{n} \left| \frac{\frac{w_k}{x-x_k}}{\sum_{k=0}^{n} \frac{w_k}{x-x_k}} \right| \mathrm{d}x = \int_a^b \Lambda_n(x) \,\mathrm{d}x \leq (b-a)\Lambda_n,\end{aligned} \tag{6.5}$$

which is a rather crude bound on the expression on the left-hand side of (6.2). We can therefore expect the quadrature rules to be better conditioned than the interpolation scheme itself. The smallest condition numbers are obtained when all the quadrature weights are positive, in which case the condition is overestimated by the Lebesgue constant. We investigate the positivity of the weights with equispaced nodes in Section 6.3.5.

Under direct rational integration we shall here mean the approximation of an antiderivative via that of an accurate approximation of r_n, e.g., a polynomial interpolant in Chebyshev points. Direct rational quadrature will be the result of applying existing quadrature rules such as Gauss–Legendre or Clenshaw–Curtis to approximate the integrals in (6.4).

Indirect rational quadrature uses the fact that the integral (6.1) may be obtained through the solution of an ordinary differential equation; see, e.g., [110, Chap. 12]. This will lead to approximations of antiderivatives and integrals.

6.3 Quadrature Rules for Differentiable Functions

6.3.1 Direct Rational Integration (DRI)

In Chebfun [8, 111, 115], functions are approximated to almost machine precision by polynomial interpolants between sufficiently many Chebyshev points of the second kind. More than 200 MATLAB commands are overloaded in this toolbox. One of these is `cumsum`, which in MATLAB computes the cumulative sum

of a vector, and in Chebfun approximates the antiderivative of functions. This command can also be used to integrate a rational interpolant r_n interpolating at arbitrary nodes. We suppose that n is not excessively large, to avoid the number of Chebyshev points required to resolve r_n becoming too large. In this way, r_n is implicitly approximated close to machine precision by a polynomial interpolant in Chebyshev points, which is then integrated, and the integration constant is chosen such that the antiderivative is equal to 0 at the left end of the interval. We shall call this method *direct rational integration* (DRI). This procedure can be established from any other method that allows one to compute antiderivatives of functions. DRI can as well be used for the approximation of definite integrals and it is not as prone to loose accuracy as IRQ, which we describe in Section 6.3.4 and mainly uses differentiation. We will see in Section 6.3.5 that the maximum errors in DRI approximation are almost identical to those in DRQ approximation of definite integrals, which we establish in Section 6.3.2. Moreover, the approximation obtained from DRI is the Chebfun analogue of an entire function, in this case a polynomial, so that it is well suited for further computations and approximations such as the solution of, e.g., integral equations.

6.3.2 Direct Linear Rational Quadrature (DRQ)

The linearity of the rational interpolant (2.8) leads to the quadrature rule (6.3) with the weights ω_k given by (6.4). Since the integrand in (6.4) is analytic and may be evaluated at every point in the interval, we can approximate the integral by any efficient quadrature rule with rapid convergence, such as Gauss–Legendre or Clenshaw–Curtis. Let $\omega_k^{\mathcal{D}}$, $k = 0, \dots, n$, be corresponding approximations of the weights in (6.4); the *direct rational quadrature rule* then replaces Q_n by

$$I = \int_a^b f(x)\,\mathrm{d}x \approx \sum_{k=0}^{n} \omega_k^{\mathcal{D}} f_k. \tag{6.6}$$

If we do not need the weights, we may apply a rule directly on the whole interpolant, since r_n can be evaluated stably everywhere in the interval. Not evaluating the quadrature weights explicitly can thus make for much faster quadrature. Notice that this could be done as well with the classical nonlinear rational interpolant, whose barycentric representation is computed in [16].

The convergence of such a quadrature rule is guaranteed, provided the interpolant itself converges. Moreover, if the interpolation error converges as h^p for some p as $h \to 0$, then the integration error will converge to 0 at least with the

same order, if we choose a quadrature rule for the integral of r_n that converges at a rate $O(h^q)$ with $q \geq p$; indeed,

$$\left|\int_a^b f(x)\,\mathrm{d}x - \sum_{k=0}^{n} \omega_k^{\mathcal{D}} f_k\right| \leq \int_a^b |f(x) - r_n(x)|\,\mathrm{d}x + \left|\int_a^b r_n(x)\,\mathrm{d}x - \sum_{k=0}^{n} \omega_k^{\mathcal{D}} f_k\right| \leq K_1 h^p + K_2 h^q \leq K h^p, \tag{6.7}$$

where K, K_1, and K_2 are constants.

By a similar argument, we see that the degree of precision of the direct rational quadrature rule attains at least the highest integer s such that every polynomial of degree at most s is exactly reproduced by the interpolant.

We have thus established that the integral of every function f with a converging rational interpolant can be approximated, by a direct rational quadrature rule, with at least the same accuracy as the interpolant. For the Floater–Hormann interpolant r_n, this yields the following result, which is valid for any distribution of the nodes and which we shall tighten in particular cases; see Section 6.3.3.

Theorem 6.4. *Suppose n and d, $d \leq n$, are nonnegative integers, $f \in C^{d+2}[a,b]$ and r_n is the rational interpolant with parameter d given in* (2.10)*. Let the quadrature weights ω_k in* (6.4) *be approximated by a quadrature rule which converges at least at the rate $O(h^{d+1})$ and has degree of precision at least $d+1$. Then*

$$\left|\int_a^b f(x)\,\mathrm{d}x - \sum_{k=0}^{n} \omega_k^{\mathcal{D}} f_k\right| \leq K h^{d+1},$$

where K is to be multiplied by the mesh ratio β from (2.18) *in the case $d = 0$. The quadrature rule* (6.6) *has degree of precision d for any n and $d+1$ if $n-d$ is odd.*

The ratio β shows up as well in the corresponding result of Theorem 2.3 on the convergence of the rational interpolants. The last statement stems from the fact that r_n reproduces polynomials of the said degrees; see Corollary 2.4.

6.3.3 Properties of DRQ with Equispaced Nodes

In this section we study the theoretical behaviour of DRQ when the rational interpolant r_n in (6.3) is a member of the Floater–Hormann family of linear rational interpolants. We shall first investigate the convergence rates of the DRQ rules for equispaced nodes. We show that, in this special case, the rate

of approximation of the quadrature rule is $O(h^{d+2})$ if the rational interpolant converges at the rate $O(h^{d+1})$. At the end of this section we establish the degree of precision and the symmetry of these rules. Some of the tools we use in the proofs stem from [69].

Let us begin with a symmetry property of the denominator of the rational interpolant (2.10). In what follows, we denote this denominator as in Chapter 5 by

$$B(x) = \sum_{i=0}^{n-d} \lambda_i(x) \tag{6.8}$$

and call $\overline{x} := \frac{a+b}{2}$ the midpoint of the interval $[a, b]$.

Lemma 6.5. *Suppose the nodes are distributed symmetrically about $\overline{x}$, i.e., $(\overline{x} - x_i) = (x_{n-i} - \overline{x})$ for all i. Then the denominator B in (6.8) of (2.10) is either symmetric or antisymmetric about $\overline{x}$, in the sense that for every real x,*

$$B(\overline{x} + x) = (-1)^{n+1} B(\overline{x} - x). \tag{6.9}$$

Proof. We show that for every $i \in \{0, \ldots, n-d\}$ the following identity holds:

$$\lambda_i(\overline{x} + x) = (-1)^{n+1} \lambda_{n-d-i}(\overline{x} - x). \tag{6.10}$$

By the definition (2.11), we have

$$\lambda_i^{-1}(\overline{x} + x) = (-1)^i \prod_{k=0}^{d} (\overline{x} + x - x_{i+k}).$$

Since the nodes are distributed symmetrically about $\overline{x}$, it follows that the above right-hand side is equal to

$$(-1)^{i+d+1} \prod_{k=0}^{d} (\overline{x} - x - x_{n-i-k}).$$

Reversing the order of the factors in the last product, and with (2.11) again, we obtain (6.10). □

For the next steps, we use the real functions

$$\Omega_n(y) := \int_{x_{d+1}}^{y} \frac{1}{B(x)} \, \mathrm{d}x. \tag{6.11}$$

This definition trivially leads to the following corollary of Lemma 6.5.

Corollary 6.6. *For any positive integers n and d, $d \le n$,*

$$\begin{aligned} &\Omega_n(x_{d+1}) = 0, \\ &\Omega_n(x_{n-d-1}) = 0, && \text{if } n \text{ is even}, \\ &\Omega_n(x_{n-d-1}) = 2\Omega_n(\overline{x}), && \text{if } n \text{ is odd}. \end{aligned}$$

Before we state the next lemma, we recall from the sketched proof of Theorem 2.2 that the reciprocal of the denominator B may be rewritten as

$$\frac{1}{B(x)} = (-1)^{n-d}\frac{L(x)}{s(x)}, \tag{6.12}$$

where L is the nodal polynomial (2.4) and s, defined in (2.17), is positive for all real x. This means that the reciprocal of the denominator changes sign only at the $n+1$ nodes x_i.

The following lemma will be essential for our proof of the convergence rates.

Lemma 6.7. *Suppose the nodes are equispaced. Then Ω_n does not change sign in (x_{d+1}, x_{n-d-1}). In particular, if $d \le n/2 - 1$, then*

$$\Omega_n(y) < 0. \tag{6.13}$$

Proof. We will show (6.13) only for $d \le n/2 - 1$ and $y \in (x_{d+1}, \overline{x})$. The other cases then become obvious from Lemma 6.5. The claim (6.13) remains to be checked at $y = x_{d+3}, x_{d+5}, \dots$ in $(x_{d+1}, \overline{x})$, since by (6.12) the function $1/B(x)$ changes sign exclusively at the nodes x_i and is negative in (x_{d+1}, x_{d+2}), independently of n and d. In order to prove (6.13), we show that

$$\int_{x_k}^{x_{k+2}} \frac{1}{B(x)}\,\mathrm{d}x < 0, \tag{6.14}$$

for $k = d+1, d+3, \dots$ such that $[k, k+2] \subseteq [d+1, n/2]$. This means that every negative contribution to $\Omega_n(y)$ dominates the positive contribution that immediately follows it. It is then easy to see that the remaining contribution to $\Omega_n(n/2)$ is also negative, if it occurs.

We first transform (6.14) into an integral over one subinterval

$$\int_{x_k}^{x_{k+2}} \frac{1}{B(x)}\,\mathrm{d}x = \int_{x_k}^{x_{k+1}} \Big(\frac{1}{B(x)} + \frac{1}{B(x+h)}\Big)\,\mathrm{d}x.$$

Since the nodes are equispaced, we can express $B(x+h)$ in terms of $B(x)$:

$$B(x+h) = \lambda_0(x+h) - B(x) + \lambda_{n-d}(x). \tag{6.15}$$

This lets us further modify the right-hand side of (6.14) to

$$\int_{x_k}^{x_{k+2}} \frac{1}{B(x)}\,\mathrm{d}x = \int_{x_k}^{x_{k+1}} \frac{\lambda_0(x+h)+\lambda_{n-d}(x)}{B(x)B(x+h)}\,\mathrm{d}x.$$

Finally, we discuss the sign of the last integrand. The denominator is negative since $x \in (x_k, x_{k+1})$ and $B(x)$ changes sign at the nodes. As $x \geq x_{d+1}$, we see from its definition (2.11), that $\lambda_0(x+h)$ is positive. Moreover, $\lambda_{n-d}(x)$ is smaller in magnitude than $\lambda_0(x+h)$ for $x \leq \overline{x}$. Thus the numerator is positive and the left-hand side of (6.13) may be interpreted as a sum of negative terms. □

An essential ingredient of our proof of the convergence rates will be the following change of variable:

$$x = a + th, \quad t \in [0, n]. \tag{6.16}$$

This will enable us to separate the powers of h from the constant factor in the error term. As a preparation, we introduce the functions

$$\overline{\lambda}_i(t) := \frac{(-1)^i}{(t-i)\cdots(t-(i+d))}, \quad i = 0, \ldots, n-d, \quad \text{and} \quad \overline{B}(t) := \sum_{i=0}^{n-d} \overline{\lambda}_i(t),$$

which are the $\lambda_i(x)$ defined in (2.11), and $B(x)$ from (6.8), after changing variables and neglecting the powers of h.

The next lemma shows that the integral of $\overline{\lambda}_0$ is bounded.

Lemma 6.8. *For any positive integers n and d, $d \leq n$, the integral*

$$\int_{d+1}^{n/2} \overline{\lambda}_0(t+1)\,\mathrm{d}t$$

is bounded as a function of n.

Proof. We first observe that, after a partial fraction decomposition, $\overline{\lambda}_0(t+1)$ may be expressed as

$$\overline{\lambda}_0(t+1) = \sum_{i=0}^{d} \frac{K_i}{t+1-i}, \quad \text{where} \quad K_i := \frac{(-1)^{i+d}}{i!(d-i)!}.$$

This expression is now easy to integrate,

$$\int_{d+1}^{n/2} \overline{\lambda}_0(t+1)\,\mathrm{d}t = \sum_{i=0}^{d} K_i \log\left(\frac{n}{2}+1-i\right) - \sum_{i=0}^{d} K_i \log(d+2-i).$$

As the last term does not depend on n, it is constant for fixed d. We will show that the first term converges to 0 as $n \to \infty$ for fixed d. To this end, we use the property of the log function to transform products into sums,

$$\begin{aligned}\sum_{i=0}^{d} K_i \log\Big(\frac{n}{2}+1-i\Big) =& \frac{(-1)^d}{d!}\sum_{i=0}^{d}(-1)^i\binom{d}{i}\log\Big(\frac{n}{2}+1-i\Big)\\ =& \frac{(-1)^d}{d!}\log\Big(\frac{P(n/2)}{Q(n/2)}\Big),\end{aligned}$$

where P and Q are monic polynomials of the same degree in $n/2$, since $\sum_{i=0}^{d}(-1)^i\binom{d}{i} = 0$. Consequently, this term vanishes as $n \to \infty$. □

As a last preparation for the main results, we prove yet another lemma.

Lemma 6.9. *For any positive integers n and d, $d \le n/2 - 1$, the expressions*

$$\int_{d+1}^{n/2} \frac{1}{\overline{B}(t)}\,\mathrm{d}t \quad \text{and} \quad \int_{d+1}^{n/2} \frac{(t-n/2)/n}{\overline{B}(t)}\,\mathrm{d}t$$

are bounded as functions of n.

Proof. As in the proof of Lemma 6.7, we may split the integrals into two parts. To this end, we define the set

$$\mathcal{K} := \Big\{k = d+1, d+3, \ldots \,\Big|\, [k,k+2] \subset \Big[d+1, \frac{n}{2}\Big]\Big\}.$$

Moreover let

$$\mathcal{R} := [d+1, n/2] \setminus \bigcup_{k\in\mathcal{K}} [k,k+2]$$

be the remaining part of the interval $[d+1, n/2]$. The integrals over $\mathcal{R}$ are clearly bounded, since $\overline{B}_n(t)$ is bounded from below as shown in Theorems 2 and 3 from [46], see also (2.20), and $(t-n/2)/n$ is smaller than $1/2$ in absolute value for $0 \le t \le n/2$. We proceed to show the boundedness of the first part of the first integral,

$$\sum_{k\in\mathcal{K}} \int_k^{k+2} \frac{1}{\overline{B}(t)}\,\mathrm{d}t = \sum_{k\in\mathcal{K}} \int_k^{k+1} \frac{\overline{\lambda}_0(t+1) + \overline{\lambda}_{n-d}(t)}{\overline{B}(t)\overline{B}(t+1)}\,\mathrm{d}t.$$

We have shown in the proof of Lemma 6.7 that the integrand does not change sign. Thus we may study the denominator separately. Its reciprocal is bounded by K^2, where $K = d!$ if $d \neq 0$ and $K = 2$ if $d = 0$. We may therefore write

$$\left| \sum_{k \in \mathcal{K}} \int_k^{k+2} \frac{1}{\overline{B}(t)} \, \mathrm{d}t \right| \leq K^2 \left| \int_{d+1}^{n/2} \overline{\lambda}_0(t+1) \, \mathrm{d}t + \int_{d+1}^{n/2} \overline{\lambda}_{n-d}(t) \, \mathrm{d}t \right|.$$

The first term is covered by Lemma 6.8 and the second is obviously bounded by $(n/2)\overline{\lambda}_{n-d}(n/2)$, which converges to a constant for $d = 0$ and vanishes as $n \to \infty$ if $d > 0$.

To deal with the second integral of the claim, we proceed analogously. First, we observe that

$$\begin{aligned} &\sum_{k \in \mathcal{K}} \int_k^{k+2} \frac{(t - n/2)/n}{\overline{B}(t)} \, \mathrm{d}t \\ &\quad = \sum_{k \in \mathcal{K}} \frac{1}{n} \int_k^{k+1} \frac{(t - n/2)(\overline{\lambda}_0(t+1) + \overline{\lambda}_{n-d}(t)) + \overline{B}(t)}{\overline{B}(t)\overline{B}(t+1)} \, \mathrm{d}t. \end{aligned}$$

Similar arguments as above lead to

$$\left| \sum_{k \in \mathcal{K}} \int_k^{k+2} \frac{(t - n/2)/n}{\overline{B}(t)} \, \mathrm{d}t \right| \leq \frac{K^2}{2} \left| \int_{d+1}^{n/2} \overline{\lambda}_0(t+1) \, \mathrm{d}t + \frac{n}{2} \overline{\lambda}_{n-d}(n/2) \right| + \frac{K}{2},$$

which is clearly bounded. □

The preceding lemmas help us to prove the main results.

Theorem 6.10. *Suppose n and d, $d \leq n/2 - 1$, are nonnegative integers, $f \in C^{d+3}[a, b]$, and r_n is the rational interpolant with parameter d given in* (2.10) *and interpolating f at equispaced nodes. Let the quadrature weights ω_k in* (6.4) *be approximated by a quadrature rule converging at least at the rate $O(h^{d+2})$. Then*

$$\left| \int_a^b f(x) \, \mathrm{d}x - \sum_{k=0}^{n} \omega_k^{\mathcal{D}} f_k \right| \leq K h^{d+2}.$$

The hypothesis $d \leq n/2 - 1$ is no real limitation, for two reasons. Firstly, d is meant to be fixed in advance and not to depend on n here. In consequence the hypothesis on d will become satisfied as n increases. Secondly, if $d \geq n/2$, we can use Theorem 6.4 and (6.7) to change the factor $b - a$ into $2dh$ and derive an error

bound depending on h^{d+2}. Integrating a function usually makes it smoother so that the hypothesis on the differentiability of f in Theorem 6.10 are not very natural when compared to the hypothesis for the interpolant to converge. As observed in Section 2.3.2, this condition might sometimes be weakened. In Section 6.10, we shall briefly come back to this observation with an example.

Proof. As exemplified in (6.7) it is sufficient to study the integral of the interpolation error,

$$\int_a^b \big(f(x) - r_n(x)\big)\,\mathrm{d}x. \tag{6.17}$$

We rewrite the interpolation error at $x \in [a,b]$ as at the beginning of Section 5.1 as

$$f(x) - r_n(x) = \frac{\sum_{i=0}^{n-d}(-1)^i f[x_i,\dots,x_{i+d},x]}{\sum_{i=0}^{n-d}\lambda_i(x)} = \frac{A(x)}{B(x)}.$$

We have shown in Lemma 5.3 that the numerator A is bounded by a constant depending only on d, on low order derivatives of f and on the length of the interval. As usual, such bounds will be denoted generically by K.

Our study of (6.17) begins with a splitting of the integral into three parts:

$$\int_a^b \big(f(x) - r_n(x)\big)\,\mathrm{d}x = \int_a^{x_{d+1}} \frac{A(x)}{B(x)}\,\mathrm{d}x + \int_{x_{d+1}}^{x_{n-d-1}} \frac{A(x)}{B(x)}\,\mathrm{d}x + \int_{x_{n-d-1}}^{b} \frac{A(x)}{B(x)}\,\mathrm{d}x.$$

The first and last terms are bounded by Kh^{d+2}: Simply apply the change of variable (6.16) and take the maximum norm. The difficult part is the second one. We will show that the oscillations of the reciprocal of B almost cancel throughout that central part of the interval $[a,b]$. To see this, we recall the definition (6.11) of Ω_n and integrate by parts:

$$\int_{x_{d+1}}^{x_{n-d-1}} \frac{A(x)}{B(x)}\,\mathrm{d}x = A(x_{n-d-1})\Omega_n(x_{n-d-1}) - \int_{x_{d+1}}^{x_{n-d-1}} A'(x)\Omega_n(x)\,\mathrm{d}x.$$

On account of Corollary 6.6 we know that $\Omega_n(x_{n-d-1})$ vanishes if n is even. If n is odd, it is equal to $2\Omega_n(\overline{x})$, which with the change of variable (6.16) and by Lemma 6.9 may be bounded by Kh^{d+2}. Lemma 6.7 enables us to deal with the second term by applying the mean value theorem for integrals:

$$\int_{x_{d+1}}^{x_{n-d-1}} A'(x)\Omega_n(x)\,\mathrm{d}x = A'(\xi)\int_{x_{d+1}}^{x_{n-d-1}} \Omega_n(x)\,\mathrm{d}x$$

for some $\xi \in [x_{d+1}, x_{n-d-1}]$. Since we assume that $f \in C^{d+3}[a,b]$, $A'(\xi)$ is bounded by a constant, as shown in Lemma 5.3. Since $x - \overline{x}$ is an antiderivative of 1, one more integration by parts yields

$$\int_{x_{d+1}}^{x_{n-d-1}} \Omega_n(x)\,\mathrm{d}x = (x_{n-d-1} - \overline{x})\Omega_n(x_{n-d-1}) - \int_{x_{d+1}}^{x_{n-d-1}} \frac{x-\overline{x}}{B(x)}\,\mathrm{d}x.$$

If n is odd, the last integral vanishes, as its integrand is antisymmetric about $\overline{x}$ by a trivial modification of Lemma 6.5. If n is even, we repeat the change of variable (6.16) and we use the symmetry of the integrand about $\overline{x}$. To conclude by means of Lemma 6.9, we recall that $h = (b-a)/n$. □

Lemma 6.5 and Corollary 6.6 enable us to establish a more general result about the degree of precision of the DRQ rule with a rational Floater–Hormann interpolant. The nodes only need to be distributed symmetrically about $\overline{x}$.

Theorem 6.11. *Suppose n and d, $d \le n$, are nonnegative integers, r_n in the DRQ rule is the rational interpolant with parameter d given in* (2.10) *and the nodes are distributed symmetrically about $\overline{x}$. Let the linear quadrature rule Q approximating the integral of r_n be symmetric and have degree of precision at least $d+2$. Then the resulting DRQ rule has degree of precision*

$$\begin{array}{ll} d+2, & \text{if } n \text{ is even and } d \text{ is odd,} \\ d+1, & \text{if } n \text{ and } d \text{ are even,} \\ d+1, & \text{if } n \text{ is odd and } d \text{ is even,} \\ d, & \text{if } n \text{ and } d \text{ are odd.} \end{array}$$

Proof. The last two results follow immediately from Corollary 2.4.

The proof for the remaining claims will be divided into two parts. Firstly, we show that the interpolation error for x^{d+2}, and x^{d+1}, respectively, is antisymmetric about $\overline{x}$. Secondly, we use this result to prove that x^{d+2} and x^{d+1}, respectively, are integrated exactly by DRQ in these cases.

We begin with the case where n is even and d is odd. Following the lines of the proof of Theorem 2 in [46] for $f(x) = x^{d+2}$, we write the interpolation error for $x \in [a,b]$ as

$$r_n(x) - x^{d+2} = \sum_{i=0,\ i \text{ even}}^{n-d-1} (x_{i+d+1} - x_i)x^{d+2}[x_i, \ldots, x_{i+d+1}, x] \Big/ B(x),$$

where $x^{d+2}[x_i, \ldots, x_{i+d+1}, x]$ stands for the corresponding divided difference of order $d+3$ of x^{d+2}, which is equal to 1; see for example [69]. Thus the numerator

is constant and the whole function is anti-symmetric by Lemma 6.5. Similar arguments may be used for the case where both n and d are even.

Now we treat the second part of the proof. To this aim let $P(x)$ be the polynomial under consideration, that is, either x^{d+2} or x^{d+1}. As the linear quadrature rule Q has degree of precision at least $d+2$, the total quadrature error of the DRQ rule is

$$\int_a^b P(x)\,\mathrm{d}x - Q[r_n] = \Big(\int_a^b P(x)\,\mathrm{d}x - Q[P]\Big) + \big(Q[P] - Q[r_n]\big) = Q[P - r_n].$$

Since Q is assumed to be symmetric and the interpolation error $P(x) - r_n(x)$ is anti-symmetric, this error vanishes. □

Finally we use Lemma 6.5 to show that the DRQ rule with a Floater–Hormann interpolant is symmetric if the nodes are distributed symmetrically about $\overline{x}$.

Theorem 6.12. *The DRQ rule* (6.6)*, as determined by the hypotheses of the previous theorem, is symmetric.*

Proof. We show that the Lagrange fundamental rational functions

$$R_k(x) := \frac{w_k}{x - x_k}\Big/ B(x)$$

are pairwise symmetric about $\overline{x}$, that is,

$$R_k(\overline{x} + x) = R_{n-k}(\overline{x} - x)$$

for every real x. The symmetry of Q then guarantees that $\omega_k^{\mathcal{D}} = \omega_{n-k}^{\mathcal{D}}$. Notice that the denominator in the barycentric representation (2.8) of r_n is equal to the denominator B in (2.10); see (2.15). The denominator B does not depend on k and (6.9) holds. As the nodes are supposed to lie symmetrically about $\overline{x}$, we see that $\overline{x} + x - x_k = -(\overline{x} - x - x_{n-k})$. We finally show that

$$w_k = (-1)^n w_{n-k}. \tag{6.18}$$

The barycentric weights are given in (2.12) as

$$w_k = (-1)^{k-d} \sum_{i \in J_k} \prod_{\substack{j=i \\ j \neq k}}^{i+d} |x_k - x_j|^{-1}, \tag{6.19}$$

with J_k from (2.13). The fact that, by the symmetry of the nodes,

$$|x_k - x_j| = |x_{n-k} - x_{n-j}|$$

and a rearrangement of the factors in the product and of the terms in the sum in (6.19) yields (6.18). □

6.3.4 Indirect Linear Rational Quadrature (IRQ)

As an alternative to integrating a rational interpolant of f as described in Section 6.3.2, we shall now follow a different approach, in which the integral is seen as the solution of an initial value problem, and which we shall call *indirect rational quadrature*. This method is again applicable to any distribution of the nodes. Approximating the integral I from (6.3) then requires the solution of a full system of linear equations of order n—or an equivalent method—but the procedure yields, like DRI, an approximation of an antiderivative of f; I is then automatically approximated by the endpoint value of the latter.

For this purpose, we approximate an antiderivative in the interval $[a, b]$ by a linear rational interpolant

$$r_n(x) \approx \int_a^x f(y)\,\mathrm{d}y, \tag{6.20}$$

which we determine as the solution of the induced first order initial value problem

$$r_n'(x) \approx f(x), \quad u_0 = r_n(a) = 0, \quad x \in [a, b]; \tag{6.21}$$

we solve (6.21) by the collocation solver for boundary value problems introduced in [12] for the second order case; see also [110].

Here, this merely requires the first derivative at the nodes of a rational interpolant written in barycentric form with nonzero weights, which can be computed as in Section 5.3: Denote by $\mathbf{u}$ the vector $(u_0, \ldots, u_n)^T$ of the unknown values of r_n at the nodes and let $\mathbf{u}'$ be the vector containing the first derivative of r_n at the nodes; then

$$\mathbf{u}' = D\mathbf{u},$$

where D is the differentiation matrix $D^{(1)}$, whose elements are given in (5.49). Applying collocation to (6.21) (with the initial condition $u_0 = 0$)—i.e., requiring equality in (6.21) at the nodes $x_1, \ldots, x_n$—leads to a system of n equations for the n unknowns $u_1, \ldots, u_n$:

$$\sum_{j=1}^{n} D_{ij} u_j = f_i, \quad i = 1, \ldots, n. \tag{6.22}$$

Inserting into (2.8) the values u_k obtained from solving this system yields an approximation, valid in the whole interval, of an antiderivative of f:

$$\int_a^x f(y)\,\mathrm{d}y \approx r_n(x) = \sum_{k=0}^{n} \frac{w_k}{x - x_k} u_k \Big/ \sum_{k=0}^{n} \frac{w_k}{x - x_k}, \quad x \in [a, b]. \tag{6.23}$$

At $x = b$, the last expression is equal to u_n, an approximation of the integral of f over the interval $[a, b]$:

$$\int_a^b f(y)\,\mathrm{d}y \approx r_n(b) = u_n.$$

In contrast to DRQ, and similarly to DRI, IRQ yields not only the value u_n approximating the integral (6.1), but also approximate values of the antiderivative $\int_a^x f(y)\,\mathrm{d}y$ at $x_1, \ldots, x_{n-1}$ as $u_1, \ldots, u_{n-1}$ and at all other $x \in [a, b]$ as the interpolant (6.23). For sets of weights w_k leading to interpolants with no poles in $[a, b]$, this approximate antiderivative is analytic.

Again, we can derive explicit formulas for the weights of the corresponding quadrature rule[1]. To this end, we use Cramer's rule with the notation of [65], which denotes by $D \underset{n}{\leftarrow} \mathbf{y}$ the matrix D with its nth column replaced by $\mathbf{y}$. Let $\widetilde{D}$ be the differentiation matrix D deprived of its first row and column (recall that $u_0 = 0$), let $\widetilde{\mathbf{f}} := (f_1, \ldots, f_n)^T$ and let $\mathbf{e}_k$ be the kth canonical vector in $\mathbb{R}^n$. Then

$$u_n = \frac{\det(\widetilde{D} \underset{n}{\leftarrow} \widetilde{\mathbf{f}})}{\det(\widetilde{D})} = \sum_{k=1}^{n} \omega_k^{\mathcal{I}} f_k,$$

where the quadrature weights are given by

$$\omega_k^{\mathcal{I}} := \frac{\det(\widetilde{D} \underset{n}{\leftarrow} \mathbf{e}_k)}{\det(\widetilde{D})}, \quad k = 1, \ldots, n.$$

6.3.5 Numerical Results

To illustrate the theoretical results from Section 6.3.3 and the efficiency of the methods introduced so far in this chapter, we approximate the integral and the antiderivative of two functions, Runge's function $f_1(x) = 1/(1 + x^2)$ and $f_2(x) = \sin(x)$. We sampled them both at equispaced nodes, f_1 in the interval $[-5, 5]$. We investigate f_2 in the nonsymmetric interval $[-4, 5]$ to avoid an

[1] The formulas for these weights were derived by Jean-Paul Berrut.

Table 6.1: Error in the approximation of the integrals of f_1 and f_2 with DRQ and IRQ.

n	Runge ($d = 3$)				Sine ($d = 4$)			
	DRQ	Order	IRQ	Order	DRQ	Order	IRQ	Order
10	7.5e−02		4.0e−01		2.5e−03		1.1e−01	
20	1.3e−03	5.8	1.0e−02	5.3	5.0e−05	5.6	5.0e−03	4.4
40	1.0e−06	10.3	4.6e−05	7.8	7.8e−07	6.0	1.9e−04	4.7
80	6.0e−09	7.4	4.9e−06	3.2	1.2e−08	6.0	7.6e−06	4.7
160	1.8e−10	5.1	4.2e−07	3.6	1.8e−10	6.0	3.1e−07	4.6
320	5.4e−12	5.0	3.6e−08	3.5	2.8e−12	6.0	1.3e−08	4.6
640	1.6e−13	5.0	3.2e−09	3.5	8.6e−14	5.0	5.8e−10	4.5

approximation of 0, since the DRQ rule is symmetric and f_2 is an odd function. We used the Floater–Hormann interpolants with the same d as in [46], i.e., $d = 3$ for f_1 and $d = 4$ for f_2, so that the error in the approximation of definite integrals can be compared with that of the interpolation of the same functions.

We want to observe estimated approximation orders with the DRQ, IRQ and DRI rules and compute the DRQ rule with the MATLAB built-in command `quadl` applied to r_n and with a very small tolerance. As mentioned earlier, this can be done in many different ways. We did the same experiments with the help of Chebfun: r_n was re-interpolated between sufficiently many Chebyshev points and then integrated with the overloaded command `sum` from Chebfun. However, we do not display the results as they are almost identical to those obtained with the former method. In [75] we computed the DRQ rules with the help of the Gauss–Legendre rule with 1000 points. This may be too expensive in many cases. However, the Chebfun command `legpts`, an implementation of a method presented in [59], provides a very fast algorithm for the parameters of Gauss–Legendre rules. Moreover, the Legendre points and weights need to be computed only once for all the investigated examples.

Table 6.1 illustrates Theorem 6.10 on the convergence rates of the DRQ rule. We find experimental orders of about 5 for the approximation of the integral of f_1 and 6 for that of f_2 for large enough n, in accordance with the predicted $d + 2$. With the IRQ rule, the order in the approximation of the integrals is smaller than with DRQ. Several examples, including those displayed here, show

Table 6.2: Error in the approximation of an antiderivative of f_1 and f_2 with DRI and IRQ.

n	Runge ($d = 3$)				Sine ($d = 4$)			
	DRI	Order	IRQ	Order	DRI	Order	IRQ	Order
10	7.5e−02		4.1e−01		6.7e−03		1.2e−01	
20	1.3e−03	5.8	1.1e−02	5.3	1.1e−04	5.9	5.4e−03	4.5
40	1.0e−06	10.3	4.9e−05	7.8	1.5e−06	6.2	2.1e−04	4.7
80	6.0e−09	7.4	5.3e−06	3.2	2.1e−08	6.2	8.1e−06	4.7
160	1.8e−10	5.1	4.5e−07	3.6	3.1e−10	6.1	3.3e−07	4.6
320	5.4e−12	5.0	3.9e−08	3.5	4.6e−12	6.1	1.4e−08	4.6
640	1.6e−13	5.0	3.4e−09	3.5	7.2e−14	6.0	6.1e−10	4.5

the experimental order $d + 1/2$.

In Table 6.2 we display the results on the approximation of antiderivatives of f_1 and f_2 obtained with DRI and IRQ, this time after interpolating the elements of the vector $\mathbf{u}$, as described in Section 6.3.4. The errors are computed as the maximum over 3000 equispaced points in the interval. Table 6.2 yields very similar values as Table 6.1; for IRQ this was to be expected from its construction. For DRI, this is not as obvious. Nevertheless, it can be observed that the approximation quality with DRI is as good as DRQ, with the same experimental rates of convergence, namely also $d+2$ in the presented examples, but for the approximation of an antiderivative on the whole interval.

The slower convergence of the IRQ rule as compared with the DRQ rule is one reason why we did not further study the theoretical convergence behaviour of the former. Additionally, we observed that some of the quadrature weights in the IRQ rule are negative for almost every admissible choice of n and d. On the other hand, numerical tests revealed that the weights in the DRQ rule, computed with the Chebfun command `sum` applied on the integral of the fundamental rational functions, are positive at least for n between d and 2500 for $0 \leq d \leq 5$. In consequence, these rules are stable; see (6.2) from Section 6.1 and [66]. Moreover, since the DRQ rules with fixed d converge for sufficiently smooth functions, they converge in particular for all polynomials and thus converge for every Riemann-integrable function; see Theorem 6.1.

In a second experiment, we compare graphically the DRQ and IRQ rules for various rather low values of d, namely $d = 5, 6, 7$, with Newton–Cotes rules; see Figure 6.1. We sampled the function $f_3(x) = \sin(100x) + 2$ at equispaced nodes in $[0, 1]$ and repeated the same computations as in the previous examples, i.e., we computed the DRQ rule with the `quadl` command. The example function is chosen so as to avoid the approximation of 0 while integrating $\sin(100x)$ over one period. The standard Newton–Cotes rules (for $d = n$) are known to be unstable and to diverge with a growing number of points. We omit plotting these results and concentrate on the composite Simpson rule and on the composite Boole rule. The slopes of the curves reflect the experimental order 4 of the composite Simpson rule for sufficiently large n [32] and the order 6 of the composite Boole rule. We see here, in the top picture for DRQ and in the bottom one for IRQ, rapidly decreasing errors for our quadrature rules based on linear barycentric rational interpolants interpolating between a large number of equispaced points. With an adequate choice of the parameter d, these quadrature rules outperform composite Newton–Cotes rules, including those with higher theoretical convergence rates; we do not show the corresponding results. For small to moderate values of n, the error of composite Simpson is smallest in this example: For such n, the piecewise parabolic interpolant turns out to be more accurate than the linear rational one. Notice that in our rules n may be any positive number, whereas it must be of the form $2k + 1$ in composite Simpson and $4k + 1$ in composite Boole.

We also approximated an antiderivative of f_3 with the DRI and IRQ rules. The errors are again similar in size to those displayed in Figure 6.1, which confirms that the DRI rule usually outperforms IRQ.

Finally we have repeated the experiments of the present section using B-spline interpolants of order $d + 1$ computed with the `spapi` command from MATLAB's curve-fitting toolbox. We omit presenting the results since these spline-based methods yield almost identical errors as the DRQ and DRI rules in our examples; only the errors with Runge's function f_1 were a bit smaller with DRI than with the antiderivative obtained from the splines. The approximation quality, as measured from the errors only, is thus nearly identical with the methods constructed from linear rational interpolation and those from splines. The former approximations, however, are analytic whereas the latter are merely a few times differentiable.

As an illustration of the remark under Theorem 6.10 about the differentiability hypothesis on the integrand, Figure 6.2 shows the absolute errors in the approximation of a very particular example, namely the approximation of

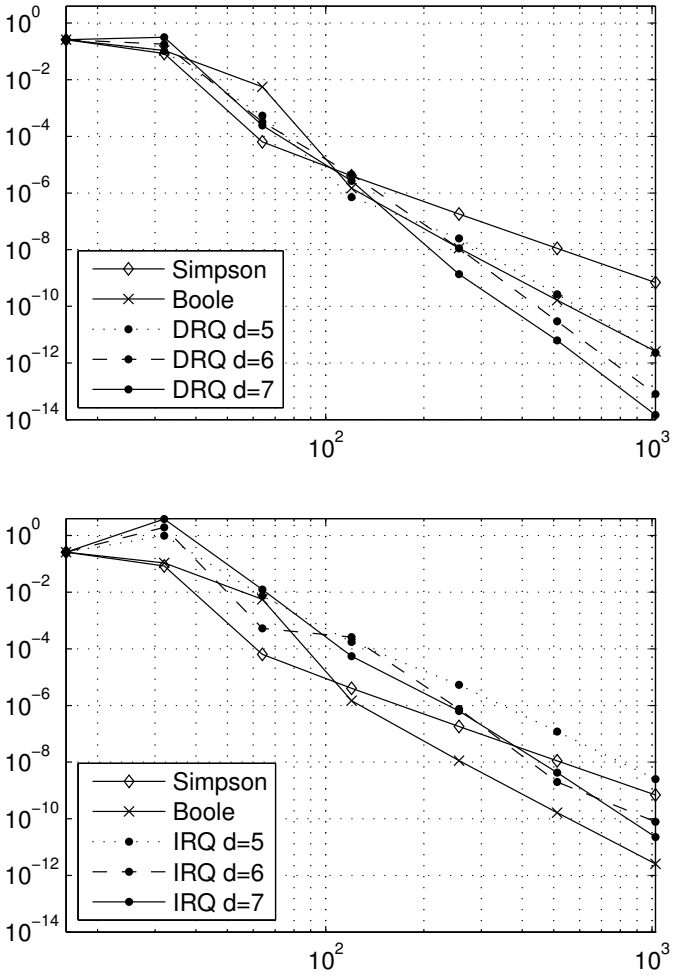

Figure 6.1: Comparison of the errors in the composite Simpson and Boole rules with DRQ (top) and with IRQ (bottom) for f_3 and with $16 \leq n \leq 1024$.

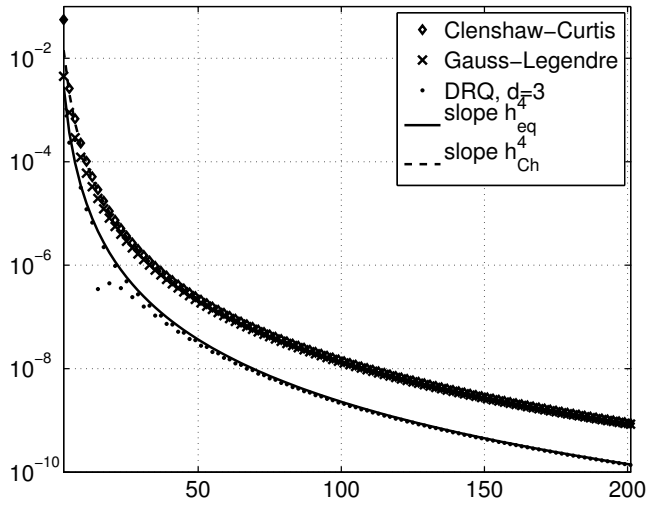

Figure 6.2: Comparison of the errors in Clenshaw–Curtis, Gauss–Legendre and DRQ with $d = 3$ for $f(x) = |x|^3$ and with $3 \leq n \leq 201$ odd.

$\int_{-1}^{1} |x|^3 \, \mathrm{d}x$ with Clenshaw–Curtis, Gauss–Legendre and DRQ with $d = 3$. It can be seen that all three methods yield convergence rate $O(h^4)$ for this example, which is more than what is to be expected from the standard theory for all three methods; see also [32, 112]. Such a behaviour of the first two rules for functions with low continuity is studied in [130]. The DRQ rule with $d = 2$ also shows the same error behaviour, but the values are almost the same as those obtained from Clenshaw–Curtis and Gauss–Legendre. With $d = 2$, the condition $f \in C^d[-1, 1]$ was thus sufficient for DRQ to converge at the rate $d + 2$; increasing d does not improve the convergence speed any more. Such an exceptional behaviour as the one displayed in Figure 6.2 is of course not observed in general. In the present example, the singularity of the integrand is located where the Chebyshev and Legendre grids are coarser by a factor $\pi/2$ than the equispaced grid and one can see that Clenshaw–Curtis and Gauss–Legendre need $\pi/2$ times more nodes to reach the same accuracy as this DRQ rule, which can clearly be seen from the picture since $h_{\mathrm{Ch}} \approx (\pi/2) h_{\mathrm{eq}}$, where for every n, h_{Ch} and h_{eq} are the maximum spacing between Chebyshev points and equispaced nodes, respectively.

6.4 Rational Quadrature Rules for Analytic Functions

Polynomial interpolation does not converge for arbitrary analytic functions and arbitrary nodes, due to Runge's phenomenon. For this reason Newton–Cotes rules diverge, but the proof is not as trivial as it may seem; divergence of the interpolant does not necessarily imply divergence of the quadrature rule obtained from that interpolant. Ouspensky [82] showed the unfavourable growth of the weights in Newton–Cotes rules and Pólya [86] proved divergence for certain analytic functions through an example. Davis [30] describes ellipses in which a function must be analytic for Newton–Cotes rules to converge in exact arithmetic.

The purpose of this section is to describe the convergence of DRQ and DRI rules for analytic functions. For now we cannot reproduce a similar theory as in [30], relying on the sum of the absolute values of the quadrature weights, since we have only the integral form (6.4) of the weights at hand and the bound on this sum involving the Lebesgue constant is too crude for such an analysis.

Let us suppose as in Section 4.2 that the blending parameter $d = d(n)$ is variable and increases with n, e.g., as $d(n) = \text{round}(Cn)$, for some positive constant $C \leq 1$. The convergence theory from Section 4.2 for rational interpolation of analytic functions can be trivially extended to the approximation of antiderivatives and integrals. As the integral operator is linear, a crude but sufficient bound on the error in the approximate antiderivative is the integral of the absolute interpolation error. This is the same reasoning as in equation (6.7) from Section 6.3.2 for quadrature rules obtained from linear interpolation schemes. With the notions from Section 4.2, the following result is an immediate consequence of Theorem 4.6 and the standard estimation of integrals.

Corollary 6.13. *Let f be a function analytic in an open neighbourhood of $[a, b]$, and let $R > 0$ be the smallest number such that f is analytic in the interior of $\mathcal{C}_R$ defined in* (4.14)*. Then the antiderivative of the rational interpolant r_n defined by* (2.10)*, with limiting node measure μ and $d(n)/n \to C$, satisfies*

$$\limsup_{n\to\infty} \left| \int_a^x f(y)\,\mathrm{d}y - \int_a^x r_n(y)\,\mathrm{d}y \right|^{1/n} \leq R,$$

for any $x \in [a, b]$.

More details on this method are not required since its behaviour will be very similar to that of the interpolation of analytic functions.

We will concentrate for the remainder of this section on the quadrature rules with equispaced nodes, even if all of what follows could immediately be done for any distribution of nodes and also for any other method obtained from an interpolation scheme with similar convergence and conditioning properties. The computation of the approximations is done with the techniques explained for differentiable functions and fixed d in Sections 6.3.1 and 6.3.2; see also Section 6.3.5.

In Sections 4.2.3 and 4.2.4 we have derived specialised results for equispaced nodes, more explicit contours in the former and, in the latter, a minimisation procedure that balances the exponential convergence of the rational interpolants with the exponential growth of the Lebesgue constants for increasing d in order to deduce a near optimal choice for d. We directly apply these methods to the approximation of antiderivatives and compare the obtained results with another method designed for the approximation with equispaced nodes as well as two methods which are almost optimal and require a special distribution of the nodes. In some cases the present methods with equispaced nodes perform almost as efficiently as those claimed to be close to optimal; this is evidently not the case for every example. The described strategy for finding good values for d does not take advantage of the slightly faster convergence of the quadrature rules as compared to interpolation, which was shown in Section 6.3.3 for fixed d, nor of the smaller condition numbers; see (6.5). The value chosen for d is simply the one that is near optimal for interpolation.

For $f(x) = 1/(1+8x^2)$ and $f(x) = 1/(1+8(x-0.5)^2)$, we approximated the antiderivative which is equal to 0 at the left end of the interval and is obviously equal to the definite integral over $[a, b]$ at the right end. The antiderivatives are obtained after choosing an appropriate value for d via the minimisation algorithm for interpolation and applying DRI on f. Besides, we computed the minimal error over all admissible choices of d, and show the values of d, as well as $C = d/n$, that lead to the minimum. In addition to the errors in the antiderivatives, we investigated the errors in the approximation of the definite integral with Clenshaw–Curtis and Gauss–Legendre quadrature. The results of this experiment are displayed in Figures 6.3 and 6.4. We do not show the errors in the approximation of the integral with the DRQ rule since these do not significantly differ from those in the antiderivatives. Gauss–Legendre performs best and Clenshaw–Curtis is very close, at least until the usual kink appears; see [127]. In the first example, our method with equispaced nodes performs nearly as well as Clenshaw–Curtis quadrature. These small errors are partly due to the symmetry effects already discussed in Section 4.2.2. The small hump in Figure 6.3 can be avoided by increasing the importance of the Lebesgue

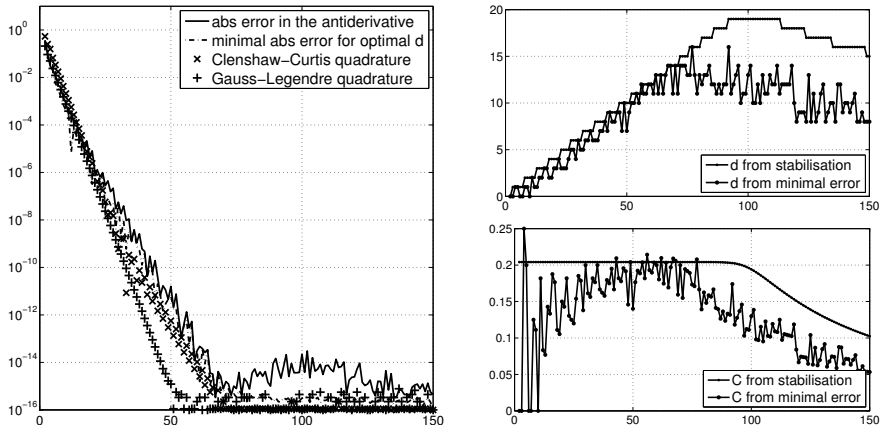

Figure 6.3: Maximal absolute errors in the approximation of an antiderivative of $f(x) = 1/(1 + 8x^2)$ on $[-1, 1]$ with $2 \leq n \leq 150$.

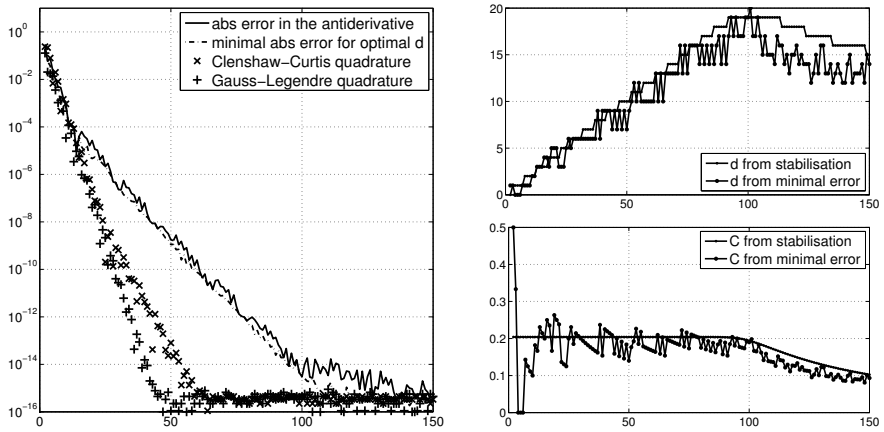

Figure 6.4: Maximal absolute errors in the approximation of an antiderivative of $f(x) = 1/(1 + 8(x - 0.5)^2)$ on $[-1, 1]$ with $2 \leq n \leq 150$.

constant as explained in Section 4.2.5. The errors in the second example behave exactly as expected and a better value for d can hardly be found. This comes from the fact that the singularities of the integrated function lie outside the cusp regions of the level lines which describe the speed of convergence as illustrated in Figure 4.3. The results obtained with our first example are quite similar to those obtained with the nonlinear least squares procedures presented in [66], also designed for equispaced nodes.

The methods presented which produce analytic approximations yield small errors; the only drawback in the schemes for analytic integrands is that the region of analyticity of the latter needs to be known in advance, unless one applies a procedure to approximately determine it or one uses the idea from the Remark in Section 4.2.4.

Chapter 7

Extended Floater–Hormann Interpolation and Applications

The barycentric rational Floater–Hormann interpolants are good approximations and are also suited for some applications of interpolation, such as those explained so far in Chapters 5 and 6. In particular, if the nodes may not be chosen at will and, e.g., need to be equispaced, then these rational interpolants are a much better scheme for smooth approximation than their polynomial analogue, which, e.g., are ill-conditioned and lead to Runge's phenomenon. Numerical experiments show that the approximation quality with equispaced nodes and measured with the error of approximation only, is similar to that of splines. For such nodes, however, the condition of Floater–Hormann interpolation deteriorates exponentially with increasing d; see Chapter 3. As mentioned already, it is not possible to construct a well-conditioned method with geometric convergence from data at equispaced nodes [85].

In Section 7.1, we present an extension of the Floater–Hormann family of barycentric rational interpolants for the equispaced case, before showing in Section 7.2 that this extension has a very small Lebesgue constant for any choice of n and d. In Section 7.3, we investigate some of its properties in the applications already studied for the original family. We conclude with numerical examples in Section 7.4.

The main reference for this chapter is [74].

7.1 Extension of the Floater–Hormann Family of Barycentric Rational Interpolants

It is well known that polynomial interpolation with equispaced nodes is ill-conditioned; see Section 3.3.2. The Lebesgue constant Λ_n grows exponentially with increasing degree n and amounts already to about $3 \cdot 10^{12}$ for $n = 50$. In floating-point arithmetic, the interpolation process will generally fail to give any accuracy at all whenever Λ_n exceeds the reciprocal of the precision of the data, which is typically at most 10^{16}. As the rational function (2.10) is a blend of polynomial interpolants of degree at most d, increasing d deteriorates the condition of the approximation method; see Section 3.3.3. For given data, $f_0, \ldots, f_n$, r_n may display large oscillations between the nodes toward the ends of the interval for a large d. The study of the Lebesgue function $\Lambda_n(x)$ from (3.7) with equispaced nodes explains this behaviour independently of the interpolation data. As we already observed in the experiments at the end of Section 3.3.3, it turns out that, for given d, $\Lambda_n(x)$ has at most d large oscillations at the ends and is much smaller in the remaining part of the interval; see Figure 3.1. This behaviour in the middle part is very similar to that of the Lebesgue function associated with polynomial interpolation at Chebyshev points of the second kind; see the middle picture in Figure 7.1.

In order to improve the condition of r_n, one may want to move the high oscillations out of the interval $[a, b]$. One possibility is to evaluate and use the rational function only in the middle part of the interval, neglecting d subintervals at each end of the interval. An alternative approach consists in adding $2d$ new data values $\widetilde{f}_{-d}, \ldots, \widetilde{f}_{-1}$ and $\widetilde{f}_{n+1}, \ldots, \widetilde{f}_{n+d}$, corresponding to additional nodes $x_{-d}, \ldots, x_{-1}$ and $x_{n+1}, \ldots, x_{n+d}$, constructed from a smooth extension of f beyond x_0 and x_n using only the given data $f_0, \ldots, f_n$. The global data set is then interpolated by a rational function r_{n+2d} and evaluated only in the interval $[a, b]$. The procedure resembles that of adding "fictitious points" in finite difference approximation [48, Section 5.1] but it is not quite the same. Yet another approach, which also constructs data outside the interval and aims at regularising interpolants of nonperiodic functions, is that of Fourier extensions; see [67] and the references therein.

Here we will look more closely at the second approach, which adds $2d$ new data values, since the first approach is trivial and implies a waste of data. The new data may be generated through a numerical Taylor expansion at each end of the interval, where the derivatives involved are approximated by one-sided finite difference formulas; see Section 5.3. To be precise, we choose positive integers

$\widetilde{n} \ll n$ and $\widetilde{d} \leq \widetilde{n}$, and compute $r_{\widetilde{n}}^{(k)}(x_0)$ and $r_{\widetilde{n}}^{(k)}(x_n)$, the kth derivatives at x_0 and x_n, respectively, of the rational interpolant of the values $f_0, \ldots, f_{\widetilde{n}}$ and $f_{n-\widetilde{n}}, \ldots, f_n$, respectively, both with parameter $\widetilde{d}$, for $k = 1, \ldots, \widetilde{d}$, provided f is $2\widetilde{d}+1$ times continuously differentiable; see Section 5.1. Then we set

$$\widetilde{f}_i := \begin{cases} f_0 + \displaystyle\sum_{k=1}^{\widetilde{d}} r_{\widetilde{n}}^{(k)}(x_0)\frac{(x_i - x_0)^k}{k!} =: T_{x_0}[r_{\widetilde{n}}](x_i), & -d \leq i \leq -1, \\ f_i, & 0 \leq i \leq n, \\ f_n + \displaystyle\sum_{k=1}^{\widetilde{d}} r_{\widetilde{n}}^{(k)}(x_n)\frac{(x_i - x_n)^k}{k!} =: T_{x_n}[r_{\widetilde{n}}](x_i), & n+1 \leq i \leq n+d. \end{cases} \tag{7.1}$$

Our extension of the Floater–Hormann family of barycentric rational interpolants is then

$$\widetilde{r}_n(x) := \sum_{i=-d}^{n+d} \frac{w_i}{x - x_i}\widetilde{f}_i \Big/ \sum_{i=-d}^{n+d} \frac{w_i}{x - x_i}, \tag{7.2}$$

where the barycentric weights w_i are computed by means of the formula (2.12) as for the original family, but this time for $n+2d+1$ nodes. If $d = 0$ the rational interpolant r_n remains unchanged; we may thus ignore this choice here.

From the above construction it is clear that no additional input data is required for the interpolation. Moreover, the derivatives $r_{\widetilde{n}}^{(k)}(x_0)$ and $r_{\widetilde{n}}^{(k)}(x_n)$ can be computed very efficiently because of the following observations. The differentiation matrices from (5.50) need to be computed recursively, but every coefficient in $D^{(k)}$ in a given row only depends on coefficients from the same row of $D^{(k-1)}$ and $D^{(1)}$. Since $r_{\widetilde{n}}^{(k)}(x_0)$ is the result of the scalar product of the first row of $D^{(k)}$ by the vector $(f_0, \ldots, f_{\widetilde{n}})^T$ and $r_{\widetilde{n}}^{(k)}(x_n)$ can be computed as the scalar product of that same row by $(f_n, \ldots, f_{n-\widetilde{n}})^T$ in the given inverted sequence, only one row of each differentiation matrix is required.

The construction (7.1) only makes sense if the nodes are equispaced, as it fights the special behaviour of the Lebesgue constants associated with original Floater–Hormann interpolation exclusively between these nodes; we therefore suppose from now on that we are dealing with equispaced nodes. Before giving the first result on $\widetilde{r}_n$, we introduce the following notation:

$$D := \min\{d, \widetilde{d}\}. \tag{7.3}$$

Theorem 7.1. *Suppose n, d, $\widetilde{n}$, $\widetilde{n} < n$, and $\widetilde{d}$, $\widetilde{d} \le \widetilde{n}$, are positive integers and assume that $f \in C^{d+2}[a-dh, b+dh] \cap C^{2\widetilde{d}+1}([a, a+\widetilde{n}h] \cup [b-\widetilde{n}h, b])$ is sampled at $n+1$ equispaced nodes in $[a, b]$. Then*

$$\|f - \widetilde{r}_n\| \le Kh^{D+1}. \tag{7.4}$$

Moreover, $\widetilde{r}_n$ has no real poles and reproduces polynomials of degree at most D for any n and $\min\{\widetilde{d}, d+1\}$ if $n+d$ is odd.

Proof. Suppose we are given the exact data $f_{-d}, \dots, f_{n+d}$. This allows us to form the rational interpolant r_{n+2d} with parameter d, whose rate of convergence is $O(h^{d+1})$. We let $x \in [a, b]$ and expand the absolute value of the error as

$$|f(x) - \widetilde{r}_n(x)| \le |f(x) - r_{n+2d}(x)| + |r_{n+2d}(x) - \widetilde{r}_n(x)|. \tag{7.5}$$

The first term is bounded by Kh^{d+1}, where K is a constant depending only on d and derivatives of f; see Theorem 2.3. As usual, we generically denote such constants by K. The second term of the above right-hand side may be bounded from above by

$$d!h^{d+1} \sum_{\substack{-d \le i \le -1 \\ n+1 \le i \le n+d}} \frac{|w_i|}{|x - x_i|} |f_i - \widetilde{f}_i|, \tag{7.6}$$

where we took the original definition (2.12) of the weights, and treated the denominator of (7.2) as in (2.20) and (2.21):

$$\left| \sum_{i=-d}^{n+d} \frac{w_i}{x - x_i} \right| = \left| \sum_{i=-d}^{n} \lambda_i(x) \right| \ge \frac{1}{d!h^{d+1}}. \tag{7.7}$$

Let us look at $|f_i - \widetilde{f}_i|$ for $i = -d, \dots, -1$; for $i = n+1, \dots, n+d$ the argument goes analogously. We denote by $T_{x_0}[f]$ the Taylor expansion of degree $\widetilde{d}$ of f about x_0. Then

$$\begin{aligned} |f_i - \widetilde{f}_i| &\le \big|f(x_i) - T_{x_0}[f](x_i)\big| + \big|T_{x_0}[f](x_i) - \widetilde{f}_i\big| \\ &\le \big|f^{(\widetilde{d}+1)}(\xi_i)\big| \frac{|x_i - x_0|^{\widetilde{d}+1}}{(\widetilde{d}+1)!} + \sum_{k=1}^{\widetilde{d}} \big|f^{(k)}(x_0) - r_{\widetilde{n}}^{(k)}(x_0)\big| \frac{|x_i - x_0|^k}{k!}, \end{aligned}$$

for some $\xi_i \in [x_i, x_0]$. It is shown in Section 5.1 that

$$\big|f^{(k)}(x_0) - r_{\widetilde{n}}^{(k)}(x_0)\big| \le Kh^{\widetilde{d}+1-k}, \quad 1 \le k \le \widetilde{d},$$

and since $|x_i - x_0|^k \leq d^k h^k$, we have

$$|f_i - \widetilde{f}_i| \leq K h^{\widetilde{d}+1}, \tag{7.8}$$

for $i = -d, \ldots, -1$ and $i = n+1, \ldots, n+d$. Finally, we use the result from Lemma 3.6 that the weights w_i in (7.6) are bounded by

$$|w_i| \leq \frac{2^d}{d! h^d}. \tag{7.9}$$

Since $x \in [a, b]$ and thus $|x - x_i| \geq h$, for $i = -d, \ldots, -1$ and $i = n+1, \ldots, n+d$, the claimed result (7.4) follows. The fact that $\widetilde{r}_n$ in (7.2) has no real poles is trivial since the rational function r_n has no real poles for any number of nodes. Equation (7.5) with a polynomial p instead of f reveals that the first term vanishes if $\deg(p) \leq \widetilde{d}$, see also (7.6) and (7.1), and that the second term is equal to 0 if $\deg(p) \leq d+1$ for $n+d$ odd and $\deg(p) \leq d$ for $n+d$ even, due to Corollary 2.4 for $n + 2d + 1$ nodes. □

The additional smoothness hypothesis in Theorem 7.1 and in the following results, as compared to those for the original r_n, might be weakened if f can be extended sufficiently smoothly from $[a, b]$ to $[a - dh, b + dh]$. On the other hand, additional smoothness properties of f encourage the use of the interpolants we have presented, which are numerically better conditioned than r_n, as we will see in the next section.

7.2 Lebesgue Constants

In Sections 3.3.2 and 3.3.3 we investigated the Lebesgue functions and constants associated with polynomial and Floater–Hormann interpolation. We shall now do the same for extended Floater–Hormann interpolation; for this we denote the associated Lebesgue function by $\widetilde{\Lambda}_n(x)$ and the Lebesgue constant by $\widetilde{\Lambda}_n$.

If the exact values of f are taken as $\widetilde{f}_i$ at x_i for $i = -d, \ldots, -1$ and $i = n+1, \ldots, n+d$, then the Lebesgue function associated with extended Floater–Hormann interpolation is the analogue of (3.7). Since these $\widetilde{f}_i$ may be obtained in a different way than presented in (7.1), and as we do not use the interpolants outside the interval $[a, b]$, we may ignore uncertainties in these values and consider the analogue of (3.7) in $[a, b]$ for the study of the condition. With this interpretation, the extended rational interpolants (7.2) have Lebesgue constants that grow logarithmically in n and d, as the following theorem shows.

Theorem 7.2. *For positive integers n and d, the Lebesgue constant $\widetilde{\Lambda}_n$ for the basis of the extended barycentric rational interpolants at $n+1$ equispaced nodes in the interval $[a,b]$ is bounded from above as*

$$\widetilde{\Lambda}_n \leq \frac{2^{d-1}}{d!\sum_{i=0}^{d-1}\frac{1}{\prod_{j=i}^{d-2}(d-\frac{1}{2}-j)\prod_{\ell=1}^{i}(\ell+\frac{1}{2})}}\big(2+\log(n+2d)\big). \tag{7.10}$$

Remark: The leading quotient in (7.10) is less than or equal to 1 for all positive d and for $d \geq 5$ it almost becomes constant, so that

$$\widetilde{\Lambda}_n \leq 0.65\big(2+\log(n+2d)\big), \quad \text{for } d \geq 5.$$

This means that $\widetilde{\Lambda}_n$ grows merely logarithmically with n and d, and that its upper bound is very close to the bound from (3.10) for polynomial interpolation with Chebyshev points of the first and second kinds.

Proof. This proof uses some tools from Section 3.3.3. If $x = x_k$ for $k = 0,\ldots,n$, then $\widetilde{\Lambda}_n(x) = 1$, because of the interpolation property. Suppose that $x_k < x < x_{k+1}$ for $k \in \{0,\ldots,n-1\}$. We multiply the numerator and the denominator of the Lebesgue function associated with (7.2) by $(x-x_k)(x_{k+1}-x)$ and take (7.9) into account to obtain

$$\widetilde{\Lambda}_n(x) \leq \frac{2^d}{d!h^d}\frac{\sum_{i=-d}^{n+d}\big|\frac{1}{x-x_i}\big|(x-x_k)(x_{k+1}-x)}{\big|\sum_{i=-d}^{n+d}\frac{w_i}{x-x_i}\big|(x-x_k)(x_{k+1}-x)} = \frac{2^d}{d!h^d}\frac{N(x)}{D(x)}, \tag{7.11}$$

where N and D are defined such as to match the numerator and the denominator, respectively, of the middle expression without the factor $2^d/(d!h^d)$. Let us first look at the numerator:

$$\begin{aligned} N(x) &= x_{k+1}-x_k+(x-x_k)(x_{k+1}-x)\Big(\sum_{i=-d}^{k-1}\frac{1}{x-x_i}+\sum_{i=k+2}^{n+d}\frac{1}{x_i-x}\Big) \\ &\leq h+\Big(\frac{h}{2}\Big)^2\Big(\sum_{i=-d}^{k-1}\frac{1}{x_k-x_i}+\sum_{i=k+2}^{n+d}\frac{1}{x_i-x_{k+1}}\Big). \end{aligned}$$

Since the nodes x_i are equispaced, the first sum simplifies to $\sum_{i=1}^{k+d}\frac{1}{ih}$, which is less than $\log(2k+2d+1)/h$, and analogously for the second sum. An upper bound for the numerator now follows:

$$\begin{aligned} N(x) &\leq h+\frac{h}{4}\log\big((2k+2d+1)(2n+4d-(2k+2d+1))\big) \\ &\leq h+\frac{h}{4}\log\Big(\frac{2n+4d}{2}\Big)^2 = h+\frac{h}{2}\log(n+2d). \end{aligned} \tag{7.12}$$

For our study of the denominator D we recall some statements from Section 2.3 and adapt them to the present setting. To begin with, we rewrite D using the original definition of the denominator of the rational interpolant,

$$D(x) = (x - x_k)(x_{k+1} - x)\Big|\sum_{i=-d}^{n} \lambda_i(x)\Big|.$$

With the notations

$$\mu_i(x) = (-1)^{n+d}\lambda_i(x)\prod_{j=-d}^{n+d}(x - x_j)$$

and

$$s(x) = \sum_{i=-d}^{n+d} \mu_i(x)$$

it can be shown, see Theorem 2.2 and [46], that

$$s(x) > 0 \quad \text{and} \quad s(x) \geq \sum_{i=k-d+1}^{k} \mu_i(x) =: s_2(x) > 0.$$

We do not need to modify the notation here even though we already defined s and the μ_i since they are defined in the same manner, the only difference being that the value of n in (2.17) has been implicitly increased to $n+2d$. From these results, we may proceed with

$$D(x) \geq (x - x_k)(x_{k+1} - x)\frac{s_2(x)}{\prod_{j=-d}^{n+d}|x - x_j|},$$

whose right-hand side yields after cancellations

$$\sum_{i=k-d+1}^{k} \widetilde{\lambda}_i(x) \quad \text{with} \quad \widetilde{\lambda}_i(x) := \frac{1}{\prod_{j=i}^{k-1}(x - x_j)\prod_{\ell=k+2}^{i+d}(x_\ell - x)}.$$

We may now deduce that the last sum has exactly one minimum in (x_k, x_{k+1}), at $x = x_k + \frac{h}{2}$. It is not difficult to see that the sum, as a function of x, is symmetric about $x_k + \frac{h}{2}$. Moreover, its derivative

$$\frac{\mathrm{d}}{\mathrm{d}x}\sum_{i=k-d+1}^{k} \widetilde{\lambda}_i(x) = \sum_{i=k-d+1}^{k} \widetilde{\lambda}_i(x)\Big(-\sum_{j=i}^{k-1}\frac{1}{x - x_j} + \sum_{\ell=k+2}^{i+d}\frac{1}{x_\ell - x}\Big) \tag{7.13}$$

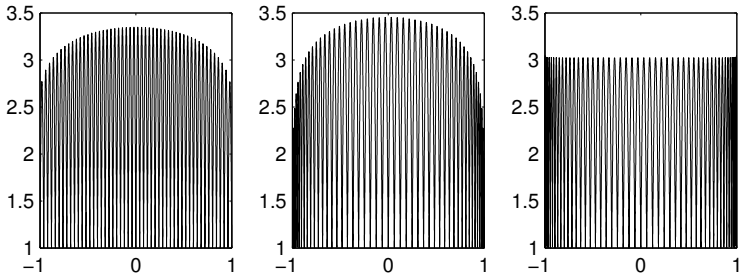

Figure 7.1: Lebesgue functions with $n = 50$ associated with extended Floater–Hormann interpolation with equispaced nodes and $d = 3$ (left), polynomial interpolation with Chebyshev points of the second kind (center) and first kind (right).

is negative in $(x_k, x_k + \frac{h}{2})$, since the mth term ($m = 1, \ldots, \lfloor \frac{d}{2} \rfloor$) in the sum in the right-hand side is negative and strictly larger in absolute value than the $(d-m)$th term, which is positive. The middle term, if it exists, is also negative in the interval in question. A similar argument shows that (7.13) is strictly positive in $(x_k + \frac{h}{2}, x_{k+1})$ and it is easy to see that it vanishes at $x = x_k + \frac{h}{2}$. We have thus established that

$$D(x) \geq h^{-d+1} \sum_{i=k-d+1}^{k} \frac{1}{\prod_{j=i}^{k-1}(k+\frac{1}{2}-j)\prod_{\ell=k+2}^{i+d}(\ell-k-\frac{1}{2})}.$$

After a rearranging the indices, this gives a bound on the denominator:

$$D(x) \geq h^{-d+1} \sum_{i=0}^{d-1} \frac{1}{\prod_{j=i}^{d-2}(d-\frac{1}{2}-j)\prod_{\ell=1}^{i}(\ell+\frac{1}{2})}. \tag{7.14}$$

Together with (7.11), the bounds on the numerator (7.12) and denominator (7.14) yield the upper bound (7.10) for the Lebesgue constants. □

To conclude this section, we take a look at the behaviour of the Lebesgue functions and constants associated with various methods; see also [34, 73]. In Figure 7.1 we compare graphically the Lebesgue functions associated with extended Floater–Hormann interpolation with equispaced nodes and $d = 3$, and polynomial interpolation with Chebyshev points of the second and first kinds,

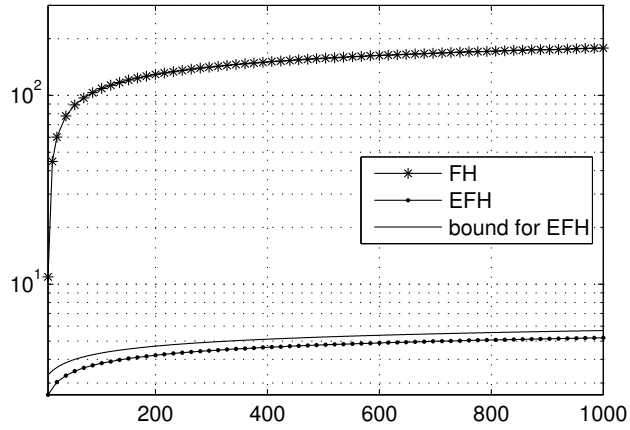

Figure 7.2: Lebesgue constants associated with Floater–Hormann (FH) and extended Floater–Hormann (EFH) interpolation with $d = 8$ and $8 \leq n \leq 1000$, together with the upper bound on the EFH Lebesgue constant.

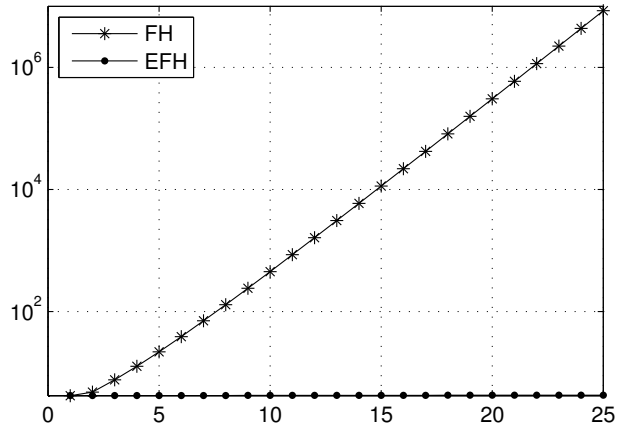

Figure 7.3: Lebesgue constants associated with Floater–Hormann (FH) and extended Floater–Hormann (EFH) interpolation with equispaced nodes, $n = 200$, and $1 \leq d \leq 25$.

all three with $n = 50$. The maxima of the respective functions, i.e., the Lebesgue constants, are very close to one another; the shapes of the first two are similar. This picture does not fundamentally change with a different choice of n or d. From Figure 7.2 we see that, already with $d = 8$, the difference in magnitude between the Lebesgue constants associated with the original Floater–Hormann interpolation and its extended counterpart is striking. This is further stressed with Figure 7.3 which shows that the Lebesgue constant associated with Floater–Hormann interpolation grows exponentially with d whereas that associated with the extended family increases very slowly from 4.19 with $d = 1$ to 4.26 with $d = 25$.

7.3 Applications

The family of rational interpolants (2.10) may be used in applications such as the approximation of derivatives of a function, its integral or an antiderivative, as presented in Chapters 5 and 6. In this section, we will investigate the behaviour of $\widetilde{r}_n$ from (7.2) regarding these applications for functions sampled at equispaced nodes. The rates of convergence stay roughly the same as with the original family of interpolants, but the constants involved in the error bounds are smaller in many cases, as documented in Section 7.4.

7.3.1 Differentiation

Let us begin with the approximation of derivatives at the nodes.

Theorem 7.3. *Suppose n, d, $\widetilde{n}$, $\widetilde{n} < n$, $\widetilde{d}$, $\widetilde{d} \le \widetilde{n}$, and k, $k \le D$, are positive integers and assume that $f \in C^{d+1+k}[a-dh, b+dh] \cap C^{2\widetilde{d}+1}([a, a+\widetilde{n}h] \cup [b-\widetilde{n}h, b])$ is sampled at $n+1$ equispaced nodes in $[a, b]$. Then*

$$|f^{(k)}(x_j) - \widetilde{r}_n^{(k)}(x_j)| \le Kh^{D+1-k}, \quad -d \le j \le n+d.$$

Proof. Suppose again we are given the exact data $f_{-d}, \ldots, f_{n+d}$. We split the absolute value of the kth derivative of the interpolation error at $x = x_j$, $-d \le j \le n+d$, into two parts,

$$|f^{(k)}(x_j) - \widetilde{r}_n^{(k)}(x_j)| \le |f^{(k)}(x_j) - r_{n+2d}^{(k)}(x_j)| + |r_{n+2d}^{(k)}(x_j) - \widetilde{r}_n^{(k)}(x_j)|. \quad (7.15)$$

The first term is bounded by Kh^{d+1-k} for $j = -d, \ldots, n+d$ as an immediate consequence of Theorem 5.5. It therefore remains to bound the second term.

For $j = 0, \dots, n$ we may consider the expression inside the absolute values and write it as

$$\frac{\mathrm{d}^{k-1}}{\mathrm{d}x^{k-1}}\bigg|_{x=x_j} \frac{\sum_{i=-d}^{-1} \frac{w_i}{x-x_i}(f_i - \widetilde{f}_i) + \sum_{i=n+1}^{n+d} \frac{w_i}{x-x_i}(f_i - \widetilde{f}_i)}{g_j(x)} k, \tag{7.16}$$

where

$$g_j(x) := (x - x_j) \sum_{i=-d}^{n} \lambda_i(x).$$

Expression (7.16) is obtained by comparing coefficients in the Taylor expansions at $x = x_j$ of the interpolation error and this error divided by $(x - x_j)$; see the beginning of Section 5.1. Let us call A the numerator of the quotient in (7.16). Applying the Leibniz rule and dividing by k then yields

$$\sum_{\ell=0}^{k-1} \binom{k-1}{\ell} A^{(k-1-\ell)}(x)\big(g_j(x)\big)^{-1}\big)^{(\ell)}\bigg|_{x=x_j}.$$

The ℓth derivative of the reciprocal of g_j is bounded by $Kh^{d-\ell}$ at $x = x_j$, as shown in the proof of Theorem 5.5. It remains to deal with the absolute values of the derivatives of A at $x = x_j$. We observe that for $0 \le \ell \le k-1$,

$$|A^{(k-1-\ell)}(x_j)| \le (k-1-\ell)! \sum_{\substack{-d \le i \le -1 \\ n+1 \le i \le n+d}} |w_i||x_j - x_i|^{\ell-k}|f_i - \widetilde{f}_i|,$$

which we bound from above by $Kh^{\widetilde{d}+1-d-k+\ell}$, using (7.8) and (7.9). This, together with the bound on the ℓth derivative of the reciprocal of g_j in the absolute value of (7.16), gives the bound $Kh^{\widetilde{d}+1-k}$ for the first term in (7.15) in the present case.

For $j = -d, \dots, -1$ and $j = n+1, \dots, n+d$ we write the expression inside the absolute values in the first term in (7.15) as

$$\frac{\mathrm{d}^{k}}{\mathrm{d}x^{k}}\bigg|_{x=x_j} \frac{\sum_{i=-d}^{-1} w_i \frac{x-x_j}{x-x_i}(f_i - \widetilde{f}_i) + \sum_{i=n+1}^{n+d} w_i \frac{x-x_j}{x-x_i}(f_i - \widetilde{f}_i)}{g_j(x)}. \tag{7.17}$$

With the Leibniz rule and the observation that the ℓth derivative, $1 \le \ell \le k$, at $x = x_j$ of the numerator $\widetilde{A}$ of the quotient in (7.17) is bounded as

$$|\widetilde{A}^{(\ell)}(x_j)| \le \ell! \sum_{\substack{-d \le i \le -1 \\ n+1 \le i \le n+d \\ i \ne j}} |w_i||x_j - x_i|^{-\ell}|f_i - \widetilde{f}_i|$$

and that $|\widetilde{A}(x_j)| \leq Kh^{\widetilde{d}+1-d}$, it follows that the first term in (7.15) is also bounded by $Kh^{\widetilde{d}+1-k}$ in this second case. □

This result leads us to define *extended rational finite difference* (ERFD) methods, which are FD methods, see Section 5.3, derived from the extended family of rational interpolants (7.2) for the approximation at the nodes in $[a, b]$, of the kth derivative of a sufficiently smooth function,

$$f^{(k)}(x_i) \approx \widetilde{r}_n^{(k)}(x_i) = \sum_{j=-d}^{n+d} D_{ij}^{(k)} \widetilde{f}_j =: \widetilde{f}_i^{(k)}. \tag{7.18}$$

The weights $D_{ij}^{(k)}$ are the elements from the $(d+1)$st to the $(n+d+1)$st row of the $(n+2d+1) \times (n+2d+1)$ differentiation matrix $D^{(k)}$ from (5.49) and (5.50); the indices are shifted according to the indices of the extended set of nodes. The weights for the first order left one-sided ERFD approximation, i.e., that at $x = x_0$, satisfy

$$\frac{1}{2^d(b-a+dh)} \leq |D_{0j}^{(1)}| \leq \frac{1}{h}, \quad j = -d, \ldots, n+d,$$

for all admissible n and d.

For the approximation of the kth derivative of a function f at intermediate points $x \in [a, b]$, we suggest, similarly to the end of Section 5.2, see, e.g., Proposition 5.11, to interpolate the approximations at the nodes $\widetilde{f}_i^{(k)}$ from (7.18) by a rational function (7.2):

$$\widetilde{R}_n^{(k)}(x) := \sum_{i=-d}^{n+d} \frac{w_i}{x-x_i} \widetilde{f}_i^{(k)} \Big/ \sum_{i=-d}^{n+d} \frac{w_i}{x-x_i}. \tag{7.19}$$

This formula is less expensive to evaluate outside the nodes than the exact derivative of $\widetilde{r}_n$ and, as we shall now see, it follows from Theorems 7.2 and 7.3 that its rate of convergence to the exact derivative of f throughout the interval $[a, b]$ is almost the same as the $O(h^{D+1-k})$ rate at the nodes from Theorem 7.3.

Proposition 7.4. *Suppose n, d, $\widetilde{n}$, $\widetilde{n} < n$, $\widetilde{d}$, $\widetilde{d} \leq \widetilde{n}$, and k, $k \leq D$, are positive integers and assume that $f \in C^{d+2+k}[a-dh, b+dh] \cap C^{2\widetilde{d}+1}([a, a+\widetilde{n}h] \cup [b-\widetilde{n}h, b])$ is sampled at $n+1$ equispaced nodes in $[a, b]$. Then*

$$\|f^{(k)} - \widetilde{R}_n^{(k)}\| \leq Kh^{D+1-k}\big(1 + \log(n+2d)\big).$$

Proof. The function f is supposed to belong to $C^{d+2+k}[a-dh, b+dh]$. Its kth derivative may be interpolated at the nodes $x_{-d}, \ldots, x_{n+d}$ with approximation rate $O(h^{d+1})$ by the rational function $r_{n+2d}[f^{(k)}]$ with parameter d from (2.10). For $x \in [a, b]$, we expand the absolute value of the error as

$$|f^{(k)}(x) - \widetilde{R}_n^{(k)}(x)| \leq \left|f^{(k)}(x) - r_{n+2d}[f^{(k)}](x)\right| + \frac{\sum_{i=-d}^{n+d} \frac{|w_i|}{|x-x_i|} |f^{(k)}(x_i) - \widetilde{f}_i^{(k)}|}{\left|\sum_{i=-d}^{n+d} \frac{w_i}{x-x_i}\right|}. \tag{7.20}$$

From Theorem 7.2, we see that the second term is bounded by

$$\widetilde{\Lambda}_n \max_{-d \leq i \leq n+d} |f^{(k)}(x_i) - \widetilde{f}_i^{(k)}|,$$

which is less than $h^{D+1-k}(1+\log(n+2d))$; see also Theorem 7.3. This, combined with the $O(h^{d+1})$ bound on the first term in (7.20), gives the result. □

7.3.2 Quadrature and Approximation of Antiderivatives

Suppose we want to approximate the integral of an integrable function f over the interval $[a, b]$, where it is sampled at $n+1$ equispaced nodes. For data available at equispaced nodes and at a few additional points, quadrature rules obtained from applying endpoint corrections to the trapezoid rule are derived in [1] and in the references therein. In Section 6.3.2 we studied direct rational quadrature (DRQ), which is based on the original Floater–Hormann family (2.10). Let us follow the same approach for the extended family (7.2), i.e.,

$$\int_a^b f(x)\,\mathrm{d}x \approx \int_a^b \widetilde{r}_n(x)\,\mathrm{d}x = \int_a^b \frac{\sum_{i=-d}^{n+d} \frac{w_i}{x-x_i} \widetilde{f}_i}{\sum_{\ell=-d}^{n+d} \frac{w_\ell}{x-x_\ell}}\,\mathrm{d}x = \sum_{i=-d}^{n+d} \omega_i \widetilde{f}_i,$$

where this time

$$\omega_i := \int_a^b \frac{\frac{w_i}{x-x_i}}{\sum_{\ell=-d}^{n+d} \frac{w_\ell}{x-x_\ell}}\,\mathrm{d}x. \tag{7.21}$$

Analogously as for the weights in the DRQ rule, the integrand in the definition of the quadrature weights ω_i may be evaluated at every point in the interval $[a, b]$. For this reason we approximate the integral giving ω_i by an efficient quadrature rule, e.g., Gauss–Legendre or Clenshaw–Curtis, and call the approximated

weights $\omega_i^{\mathcal{D}}$. The corresponding method, the *extended direct rational quadrature* (EDRQ), then reads

$$\int_a^b f(x)\,\mathrm{d}x \approx \sum_{i=-d}^{n+d} \omega_i^{\mathcal{D}} \widetilde{f}_i. \tag{7.22}$$

An explicit knowledge of the weights $\omega_i^{\mathcal{D}}$ is not always necessary in practice: As $\widetilde{r}_n$ is analytic, it is sufficient to apply a quadrature rule on the interpolant to directly compute (7.22). The following theorem gives the main properties of EDRQ.

Theorem 7.5. *Suppose n, d, $\widetilde{n}$, $\widetilde{n} < n$, and $\widetilde{d}$, $\widetilde{d} \leq \widetilde{n}$, are positive integers and assume that $f \in C^{d+3}[a-dh, b+dh] \cap C^{2\widetilde{d}+1}([a, a+\widetilde{n}h] \cup [b-\widetilde{n}h, b])$ is sampled at $n+1$ equispaced nodes in $[a,b]$. Let the quadrature weights ω_i in (7.21) be approximated by a linear quadrature rule Q converging at least at the rate $O(h^{d+2})$. Then*

$$\left| \int_a^b f(x)\,\mathrm{d}x - \sum_{i=-d}^{n+d} \omega_i^{\mathcal{D}} \widetilde{f}_i \right| \leq K h^{D+2} \log(n).$$

Moreover, if the quadrature rule Q is symmetric and has degree of precision at least $\min\{d+1, \widetilde{d}\}$, then the resulting EDRQ rule is symmetric and its degree of precision is D for any n and $\min\{d+1, \widetilde{d}\}$ if $n+d$ is odd.

Proof. We begin with splitting the absolute value of the quadrature error into two parts,

$$\left| \int_a^b f(x)\,\mathrm{d}x - \sum_{i=-d}^{n+d} \omega_i^{\mathcal{D}} \widetilde{f}_i \right| \leq \left| \int_a^b \big(f(x) - \widetilde{r}_n(x)\big)\,\mathrm{d}x \right| + \left| \int_a^b \widetilde{r}_n(x)\,\mathrm{d}x - \sum_{i=-d}^{n+d} \omega_i^{\mathcal{D}} \widetilde{f}_i \right|.$$

The second part is bounded by Kh^{D+2} because of the rate of convergence of Q. We subdivide the first part into

$$\int_a^{x_1} \big|f(x) - \widetilde{r}_n(x)\big|\,\mathrm{d}x + \left| \int_{x_1}^{x_{n-1}} \big(f(x) - \widetilde{r}_n(x)\big)\,\mathrm{d}x \right| + \int_{x_{n-1}}^b \big|f(x) - \widetilde{r}_n(x)\big|\,\mathrm{d}x.$$

The sum of the first and last terms is bounded by $2h\|f - \widetilde{r}_n\|$, which is less than or equal to Kh^{D+2} by Theorem 7.1. To treat the middle term, we assume that

the values of the function f are given at all the nodes $x_{-d}, \ldots, x_{n+d}$ and we interpolate it by the rational function (2.10) with parameter d at these nodes. After adding and subtracting $r_{n+2d}(x)$ in the argument, simplifying and writing the interpolation error as in (2.19), the latter becomes

$$\sum_{\substack{-d\le i\le -1\\ n+1\le i\le n+d}} w_i(f_i-\widetilde{f}_i)\int_{x_1}^{x_{n-1}} \frac{1}{(x-x_i)\sum_{\ell=-d}^{n}\lambda_\ell(x)}\,\mathrm{d}x + \int_{x_1}^{x_{n-1}} \frac{\sum_{i=-d}^{n}(-1)^i f[x_i,\ldots,x_{i+d},x]}{\sum_{\ell=-d}^{n}\lambda_\ell(x)}\,\mathrm{d}x.$$

It is shown in the proof of Theorem 6.10 that the absolute value of the second term is bounded by Kh^{d+2}, as it corresponds to the integral of the interpolation error over the middle part of the interval of interpolation, i.e., the part without the first and last $d+1$ sub-intervals. The factors $(x-x_i)$ in the integrand of the first part of the above expression do not change sign in the interval $[x_1, x_{n-1}]$ since $i \notin \{1,\ldots,n-1\}$; the application of the mean value theorem for integrals yields

$$\sum_{\substack{-d\le i\le -1\\ n+1\le i\le n+d}} w_i(f_i-\widetilde{f}_i)\frac{1}{\sum_{\ell=-d}^{n}\lambda_\ell(\xi_i)}\int_{x_1}^{x_{n-1}} \frac{1}{x-x_i}\,\mathrm{d}x,$$

for some $\xi_i \in [x_1, x_{n-1}]$. The claimed error bound now follows with (7.7), (7.8) and (7.9).

The symmetry of EDRQ follows directly from Theorem 6.12, which shows that the integrand in the mth quadrature weight is symmetric to that in the $(n+1-m)$th weight with respect to the midpoint of the interval. The degree of precision follows from the fact that the extended rational interpolants reproduce polynomials of the claimed degree; see Theorem 7.1. □

We also studied indirect rational quadrature (IRQ), a method based on linear barycentric rational interpolation for the approximation of an antiderivative of such a function; see Section 6.3.4. With the extended family of rational interpolants (7.2), the method (EIRQ) becomes the following: We approximate $\int_a^x f(y)\,\mathrm{d}y$ by the rational function

$$\widetilde{r}_n(x), \tag{7.23}$$

interpolating the result u of a collocation [12] at the nodes $x_{-d}, \ldots, x_{n+d}$ applied to the initial value problem

$$\widetilde{r}_n'(x) \approx \widetilde{f}(x), \quad u_0 = 0, \quad x \in [a-dh, b+dh],$$

where $\widetilde{f}(x)$ is a function with values $\widetilde{f}_{-d}, \ldots, \widetilde{f}_{n+d}$ at the nodes as defined in Section 7.1. In other words, we solve the system of $n + 2d$ equations

$$\sum_{\substack{j=-d \\ j\neq 0}}^{n+d} \widetilde{D}^{(1)}_{ij} u_j = \widetilde{f}_i, \quad i = -d, \ldots, -1, 1, \ldots, n+d,$$

with $\widetilde{D}^{(1)}$ the first order differentiation matrix from Section 5.3, without its $(d+1)$st row and column, and insert the so-obtained $u_{-d}, \ldots, u_{n+d}$ into (7.23). Note that, analogously to IRQ, u_n gives an approximation of the integral of f over $[a, b]$ and that the approximation (7.23) is analytic.

The experimentally more accurate approach presented in Section 6.3.1, namely to compute an antiderivative of the rational interpolant with the `cumsum` command from Chebfun, also works with extended Floater–Hormann interpolation and we shall call that method extended direct rational integration (EDRI).

7.4 Numerical Results

Let us now look at a few numerical examples illustrating the results and remarks from Sections 7.1 to 7.3. The examples document the error behaviour of extended Floater–Hormann interpolation from equispaced samples and of its applications, i.e., the approximation of derivatives, integrals and antiderivatives. We compare it with original Floater–Hormann interpolation with the same value of the parameter d and with B-splines of order $d + 1$ obtained with the `spapi` command from the MATLAB curve-fitting toolbox. In all the tests, the values of n are even and the parameters for the extended Floater–Hormann interpolants remain fixed at $\widetilde{d} = 7$ and $\widetilde{n} = 11$. The experimental convergence rates may be read from the slopes in the logarithmic plots. The errors are computed as the maximum absolute values of the differences between the interpolant and the exact function at 2000 equispaced points in the interval $[a, b]$.

Figure 7.4 shows the interpolation of Runge's function $f_1(x) = 1/(1 + x^2)$ in the interval $[-5, 5]$ for the three interpolants with theoretical convergence rate $O(h^5)$. The slopes in the error curves are almost identical for n large enough, but the errors in the rational interpolants are much smaller than those in the spline. With the interpolation of $\sin(x)$ the picture is similar, only the values of the errors are closer together; we omit the corresponding plot.

The next example deals with the conditioning of the interpolation process. It is well known that, due to the Runge phenomenon, the polynomial interpolant of

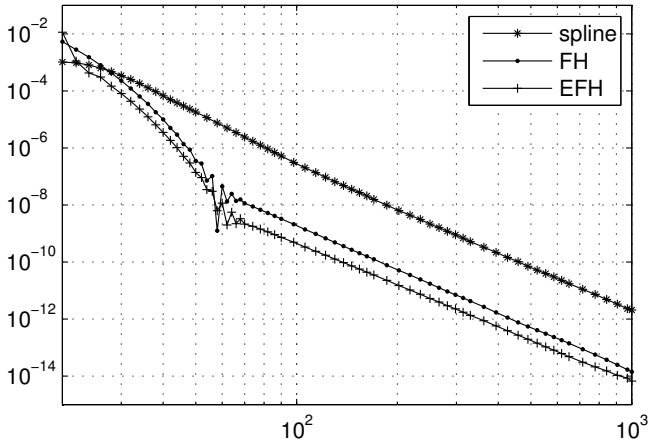

Figure 7.4: Error behaviour of spline, Floater–Hormann (FH) and extended Floater–Hormann (EFH) interpolation of f_1 in $[-5, 5]$ with $d = 4$ and $20 \leq n \leq 1000$.

f_1, sampled at equispaced nodes in the chosen interval, diverges as n increases. Suppose the perturbation 10^{-12} is alternatively added and subtracted to the given data, i.e., to the sample of f_1 at equispaced points. In Figure 7.5 the error behaviour of the investigated interpolants of the perturbed f_1 in $[-5, 5]$ with $n = 1000$ is shown as a function of d. As the value of the theoretical convergence order of the spline and the blending parameter of Floater–Hormann interpolants varies from 2 to 51, a minimum is attained in the error but the latter increases exponentially thereafter; see Section 4.2.4. Once it reaches its minimum, the error in the extended Floater–Hormann interpolant, in contrast, remains smallest possible, namely 10^{-12}, which is the magnitude of the perturbation of the data. One may therefore conclude that choosing an inadequate value for d is much less likely with the extended Floater–Hormann interpolants than with the original family. Even more extreme examples confirm this observation: Floater–Hormann interpolation with d too large and for severely perturbed data yields large deviations toward the ends of the interval, whereas the extended interpolants merely oscillate in the direct vicinity of the perturbation and with small amplitude. *Every possible choice of d becomes admissible with the extended family of rational interpolants*: The interpolation of $\sin(x)$ in $[-5, 5]$ with $n = 50000$

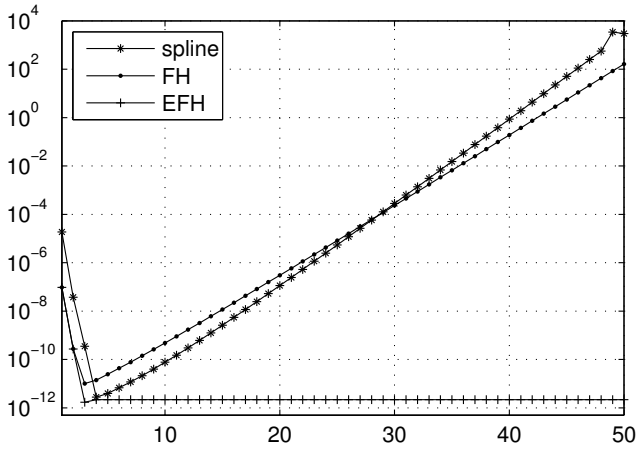

Figure 7.5: Error behaviour of spline, Floater–Hormann (FH) and extended Floater–Hormann (EFH) interpolation of f_1 with sign alternating 10^{-12}-perturbation of the data, $n = 1000$ and $1 \leq d \leq 50$.

and $d = 200$ gives an error of $3 \cdot 10^{-12}$, this is in clear contrast to the error of 0.68 with the original Floater–Hormann interpolant.

This observation is further stressed by the plots in Figure 7.6. Similarly to the experiments from Section 4.3 for the original Floater–Hormann family, we plotted the reciprocal of the denominator $\sum_{i=-d}^{n} \lambda_i(x)$ of the extended Floater–Hormann interpolants with $d = 1$ and $d = 6$. This time the denominator function yields a behaviour that is very close to equi-oscillation, does this for the other values of d as well, and thus might show that equispaced nodes are very close to optimal for these rational interpolants. What is more, this behaviour avoids a larger increase of the approximation error toward the ends of the interval than in the middle.

As mentioned earlier, the question about a good choice of d for extended Floater–Hormann interpolation is not as important as for the classical one; the choice of $\widetilde{n}$ and $\widetilde{d}$ must, however, be adequate. With the former it is most often sufficient to take d rather large: Due to the good conditioning, the errors are then mostly automatically smallest. It is because of this observation and the fact that no noticeable improvements were possible that we refrain from developing a similar theory as in Section 4.2 for the extended family.

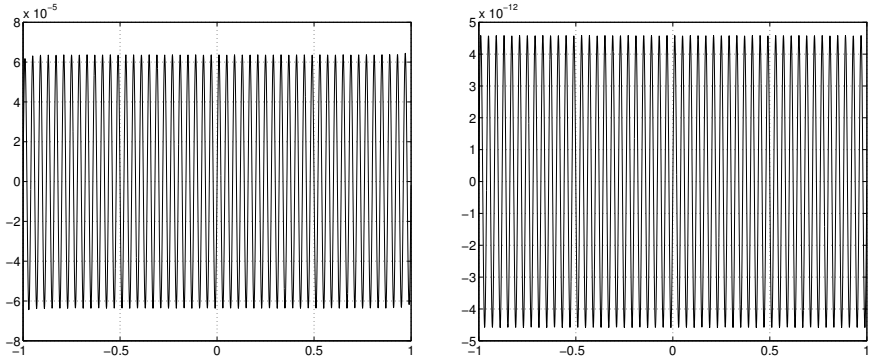

Figure 7.6: The reciprocal of the denominator $\sum_{i=-d}^{n} \lambda_i(x)$ of $\widetilde{r}_n$ plotted as $L(x)/\sum_{i=-d}^{n} \mu_i(x)$ with $n = 100$ and $d = 1$ and $d = 6$.

We now turn our attention to the applications of the presented interpolants described in Section 7.3. The first and second derivatives of $f_2(x) = \sin(x)$ are approximated as suggested at the end of Section 7.3.1, namely by the rational interpolant with $d = 4$ of the derivatives at the nodes of the interpolant of f_2; see (7.19) for the extended family and the analogous expression (5.44) for the original family of rational interpolants. In Figure 7.7, where k denotes the order of differentiation, we see that the experimental convergence rates are similar with the three methods, as to be expected. The approximation based on the extended family yields smaller errors in both cases, with a remarkable difference in the approximation of the second derivative, where the errors with the methods based on the spline interpolant of order 5 and the original family almost coincide.

One-sided rational finite difference approximation at the ends and RFD approximation near the ends of the interval are very successful for large numbers of nodes, as already noticed in Section 5.4. One-sided ERFD approximation at $x = -5$ with $d = 4$ of the second and fourth derivatives of f_1, sampled in $[-5, 5]$, still improves upon RFD; see Figure 7.8. In this example the experimental rates of convergence are even larger with the ERFD method; classical FD approximation fails, mainly because of Runge's phenomenon. The fact that the extended family of rational interpolants displays reduced oscillations toward the ends of the interval as compared to the original family definitely helps in this application, in addition to the better conditioning.

We consider the approximation of an antiderivative and the integral of

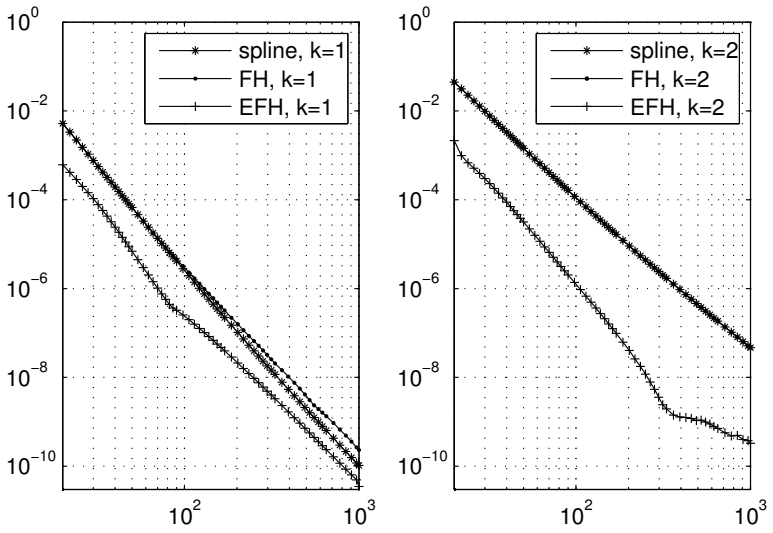

Figure 7.7: Error behaviour of spline, Floater–Hormann (FH) and extended Floater–Hormann (EFH) approximations of the first (left) and second derivative (right) of f_2 in $[-5, 5]$ with $d = 4$ and $20 \leq n \leq 1000$.

$f_3(x) = \sin(100x) + 2$ in the interval $[0, 1]$ using rational interpolants with $d = 5$. The errors in the indirect rational methods for the approximation of an antiderivative are larger than those obtained with the antiderivative of the spline interpolant of order 6; see Figure 7.9. This result was to be expected since the former methods use the differentiation matrix for the computation of the derivative at the nodes of the rational interpolant approximating the antiderivative, which does not improve the convergence rates; numerical experiments with the IRQ in Section 6.3.5 already revealed experimental orders $O(h^{d+1/2})$. It must, nevertheless, be kept in mind that the indirect rational methods give analytic approximations of an antiderivative. The Chebfun based EDRI method preforms slightly better than spline integration. As can be seen in Figure 7.9, the experimental convergence rate is the same as with the spline, namely $d + 2$ for sufficiently large n.

Figure 7.10 displays the errors in the approximation of the integral of f_3 over the interval $[0, 1]$ by the integral of the spline interpolant of order 6 of f_3,

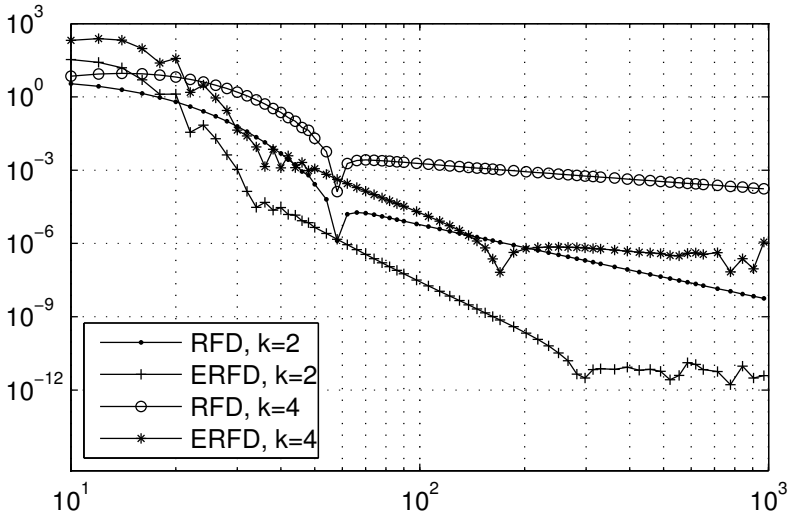

Figure 7.8: Error behaviour of one-sided RFD and ERFD (with $d = 4$ and $10 \leq n \leq 1000$) approximations at $x = -5$ of the second and fourth derivatives of f_1 sampled in $[-5, 5]$.

by DRQ and EDRQ with $d = 5$ as well as the composite Boole rule, i.e., the composite Newton–Cotes rule of order 6. To be specific about the direct quadrature rules, the integrals of the rational interpolants are approximated here as in Section 6.3.5 with the MATLAB built-in command `quadl` with high precision and also with the Chebfun command `sum`; both alternatives yield nearly identical plots with the present example. For small values of n, the four methods yield similar results. With larger values of n (≥ 150), the error curves show smallest errors in the EDRQ. In this example EDRQ beats Boole's rule, whereas DRQ does not. Numerical experiments reveal that there are even fewer negative quadrature weights in the EDRQ rules than in the DRQ ones, which already contain only few of them for small values of d; see the end of Section 6.3.5.

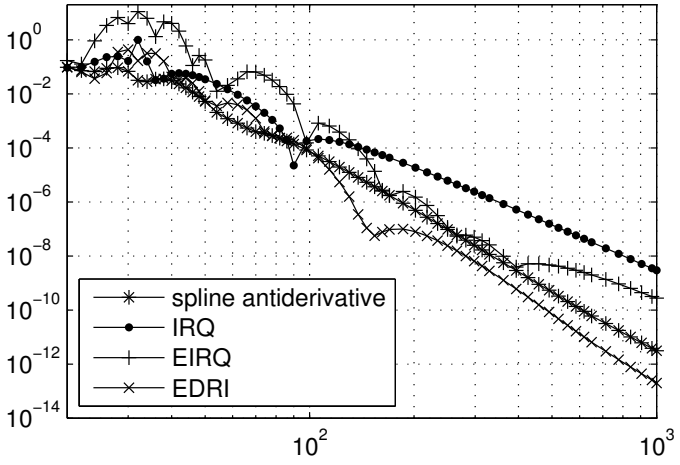

Figure 7.9: Error behaviour of spline antiderivative, IRQ and EIRQ of f_3 in $[0, 1]$ with $d = 5$ and $20 \leq n \leq 1000$.

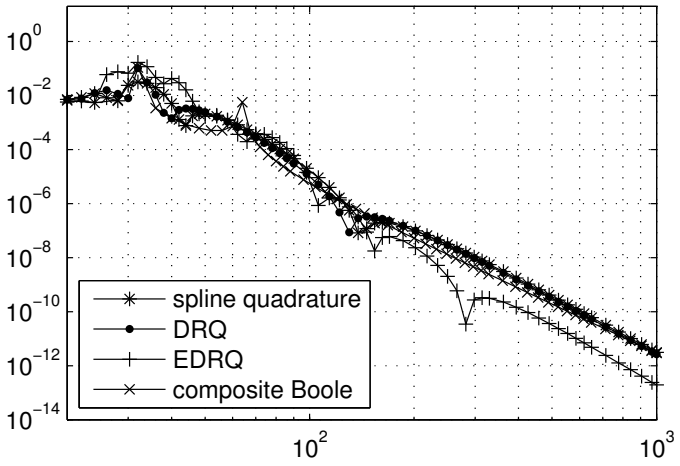

Figure 7.10: Error behaviour of spline quadrature, DRQ, EDRQ and Boole's rule for the integral of f_3 in $[0, 1]$ with $d = 5$ and $20 \leq n \leq 1000$.

Chapter 8

Conclusions and Outlook

The promising fact that the blended polynomial interpolation scheme presented by Floater and Hormann in [46] allows fast, stable and well-conditioned interpolation between equispaced nodes, even for the notorious Runge function, was a major motivation for this thesis. The goal was to study why the combination of this interpolation scheme with equispaced nodes is so successful, which arguments explain the stability and the good condition, how to choose the blending parameter d, and whether the scheme is suited for an initial selection of applications. The publication of the paper [85] by Platte, Trefethen and Kuijlaars, which shows that it is not possible to construct an approximation scheme between equispaced nodes that is simultaneously well-conditioned and exponentially convergent, further led me to investigate how close the convergence of Floater–Hormann interpolation to analytic functions sampled at equispaced nodes can be to exponential, while still being well-conditioned.

After the introduction in Chapter 1, our investigations began in Chapter 2, where we reviewed the Floater–Hormann interpolation scheme with some of its fundamental properties, such as the convergence rates for differentiable functions, the absence of real poles and the barycentric representation. Moreover, we studied experimentally the location of the complex poles with equispaced nodes. A theoretical knowledge of the exact location of the poles would lead to interesting additional results: Among others, a rigorous explanation of the near-optimality of equispaced nodes, quadrature rules which do not rely on the second "stage" of a more accurate rule, a different approach to bounding derivatives using Markov-type inequalities, and some knowledge of the region around the nodes in the complex plane where at least some accuracy can be expected.

We studied the stability of Floater–Hormann interpolation evaluated with the barycentric formula in Chapter 3 and observed that this scheme with small values of d and equispaced nodes is almost as forward stable as polynomial interpolation between Chebyshev points evaluated with the second barycentric formula. The investigation of the numerical condition was presented in that same chapter. The Lebesgue constants have been proven to grow logarithmically with increasing n and exponentially with d, when the nodes are equispaced or quasi-equispaced. This shows that Floater–Hormann interpolation with such nodes and small to moderate values of d is well-conditioned.

Thereafter, in Chapter 4, a further property of the Floater–Hormann scheme was studied, namely that of its convergence or divergence rate when d increases with n while interpolating analytic functions; this rate was found to be exponential in exact arithmetic; the speed depends on the location of the singularity of the interpolated function nearest to the interval. Since simultaneous exponential convergence and well-conditioned interpolation between equispaced nodes is provably impossible, we had to derive a stabilisation algorithm that allows one to balance the fast convergence and the growing condition number so as to obtain a very satisfactory approximation between equispaced nodes; this was demonstrated with several numerical examples. As a by-product, we obtained a good recommendation for the choice of d. The elements from potential theory used to prove the aforementioned convergence results, together with other ideas, lead to a discussion about a good choice of nodes for Floater–Hormann interpolation; moreover, rather speculative arguments combined with confirmed ones make us believe that equispaced nodes must be near optimal for this interpolation scheme, when d is small compared to n.

Following the properties of Floater–Hormann interpolation in Chapters 2-4, we presented in Chapters 5 and 6 two applications, namely the approximation of derivatives, and that of integrals and antiderivatives.

We began with the analysis of the convergence of derivatives of linear rational interpolants to the respective derivatives of functions and proved that the rate roughly decreases by one unit per differentiation. For the approximation of higher order derivatives between the nodes, we suggested a fast and cheap alternative to the computation of the exact derivatives of the interpolant. The results on convergence and the formulas for derivatives of barycentric rational interpolants allowed us to construct rational finite difference formulas, the rational analogues of polynomial finite differences. These new formulas are more stable for calculating one-sided approximations of derivatives with equispaced nodes. The results and observations from Chapter 5 may be used in the construction of methods for the solution of differential equations.

As a second class of applications we presented the DRI and IRQ methods for the approximation of antiderivatives and DRQ for integrals. These methods can be used with any barycentric rational interpolation scheme, but for IRQ the scheme must be linear. The rate of convergence of DRQ with Floater–Hormann interpolation and equispaced nodes is shown to be one unit larger than for the interpolation itself; the same was observed numerically for DRI, while IRQ gave experimental rates one half smaller than the interpolation. In our theoretical results we had to impose slightly stricter hypotheses on the differentiability than for the interpolation. This was also the case with our convergence results for the derivatives. These hypotheses can sometimes be weakened, also for the interpolation itself; this was observed numerically. The convergence of the DRQ rule with variable d and equispaced nodes for analytic functions immediately follows from the theory in Chapter 4. A similar stabilisation algorithm was tested on several examples.

Inspired from the investigations on the Lebesgue functions and constants in Chapter 3, and especially from the fact that the Lebesgue functions grow much faster at the ends of the interval that in the middle, we constructed an extension of the Floater–Hormann interpolation scheme for equispaced nodes with very small condition numbers. Some of its additional properties were studied and it was shown that the convergence rates for the approximation of derivatives and integrals are the same as with the original family. Numerical tests showed that the error is often smaller with the extended scheme. Perturbations in the data do not significantly deteriorate the interpolation quality and the choice of a good value of d becomes less important, since it is sufficient to take it moderately large. The approach of the choice of d followed in from Chapter 4 did not really improve the accuracy of the approximation in some numerical tests and was therefore left aside.

We hope that with the properties presented and the initial selection of applications, which we expect to extend in the future, we could bring convincing arguments that the Floater–Hormann family of linear barycentric rational interpolants with equispaced nodes is a competitive scheme combining ease of use, fast and stable evaluation by means of the barycentric formula, in addition to well-conditioned interpolation and rather fast convergence. We finally mention that a catalogue of methods for the approximation of functions sampled at equispaced nodes exists, and a comparison between them is being prepared by Rodrigo Platte.

Bibliography

[1] Alpert, B. K. High-order quadratures for integral operators with singular kernels. *J. Comput. Math. Appl. 60* (1995), 367–378.

[2] Atkinson, K. E. *An Introduction to Numerical Analysis,* 2nd ed. Wiley, New York, 1989.

[3] Baltensperger, R. Barycentric rational interpolation with asymptotically monitored poles. *Numer. Algorithms 57* (2011), 67–81.

[4] Baltensperger, R., and Berrut, J.-P. The errors in calculating the pseudospectral differentiation matrices for Čebyšev–Gauss–Lobatto points. *Comput. Math. Appl. 37* (1999), 41–48. Corrigenda in Comput. Math. Appl. 38 (1999), 119.

[5] Baltensperger, R., Berrut, J.-P., and Noël, B. Exponential convergence of a linear rational interpolant between transformed Chebyshev points. *Math. Comp. 68* (1999), 1109–1120.

[6] Barnett, S. A companion matrix analogue for orthogonal polynomials. *Linear Algebra Appl. 12* (1975), 197–208.

[7] Barnett, S. Some applications of the comrade matrix. *Int. J. Control 21* (1975), 849–855.

[8] Battles, Z., and Trefethen, L. N. An extension of MATLAB to continuous functions and operators. *SIAM J. Sci. Comput. 25* (2004), 1743–1770.

[9] Berrut, J.-P. Rational functions for guaranteed and experimentally well-conditioned global interpolation. *Comput. Math. Appl. 15* (1988), 1–16.

[10] Berrut, J.-P. The barycentric weights of rational interpolation with prescribed poles. *J. Comput. Appl. Math. 86* (1997), 45–52.

[11] Berrut, J.-P. A circular interpretation of the Euler–Maclaurin formula. *J. Comput. Appl. Math. 189* (2006), 375–386.

[12] Berrut, J.-P., and Baltensperger, R. The linear rational pseudospectral method for boundary value problems. *BIT Numer. Math. 41* (2001), 868–879.

[13] Berrut, J.-P., Baltensperger, R., and Mittelmann, H. D. Recent developments in barycentric rational interpolation. In: M. G. de Bruin, D. H. Mache and J. Szabados, Trends and Applications in Constructive Approximation. International Series of Numerical Mathematics, vol. 151, 27–51. Birkhäuser, Basel, 2005.

[14] Berrut, J.-P., Floater, M. S., and Klein, G. Convergence rates of derivatives of a family of barycentric rational interpolants. *Appl. Numer. Math. 61* (2011), 989–1000.

[15] Berrut, J.-P., and Mittelmann, H. D. Lebesgue constant minimizing linear rational interpolation of continuous functions over the interval. *Comput. Math. Appl. 33* (1997), 77–86.

[16] Berrut, J.-P., and Mittelmann, H. D. Matrices for the direct determination of the barycentric weights of rational interpolation. *J. Comput. Appl. Math. 78* (1997), 355–370.

[17] Berrut, J.-P., and Trefethen, L. N. Barycentric Lagrange interpolation. *SIAM Rev. 46* (2004), 501–517.

[18] Bos, L., De Marchi, S., and Hormann, K. On the Lebesgue constant of Berrut's rational interpolant at equidistant nodes. *J. Comput. Appl. Math. 236* (2011), 504–510.

[19] Bos, L., De Marchi, S., Hormann, K., and Klein, G. On the Lebesgue constant of barycentric rational interpolation at equidistant nodes. *Numer. Math. 121* (2012), 461–471.

[20] Bos, L., De Marchi, S., Hormann, K., and Sidon, S. Bounding the Lebesgue constant for Berrut's rational interpolant at general nodes. *J. Approx. Theory*. To appear.

[21] Brass, H. *Quadraturverfahren.* Studia Mathematica, vol. 3, Vandenhoeck & Ruprecht, Göttingen, 1977.

[22] Brutman, L. On the Lebesgue function for polynomial interpolation. *SIAM J. Numer. Anal. 15* (1978), 694–704.

[23] Brutman, L. Lebesgue functions for polynomial interpolation—a survey. *Ann. Numer. Math. 4* (1997), 111–127.

[24] Bulirsch, R., and Rutishauser, H. Interpolation und genäherte Quadratur. In *Sauer, R., Szabó, I. (eds) Mathematische Hilfsmittel des Ingenieurs, Teil III* (Springer, Berlin, 1877).

[25] Cauchy, A. L. Sur un nouveau genre de calcul analogue au calcul infinitésimal. In: Exercices de Mathématiques, vol. I, 11–24. Beaucé–Rusand, Saint–Suplice, 1826. And in: Œuvres Complètes d'Augustin Cauchy, IIe série, tome 6, 23–37. Gauthier–Villars, Paris, 1887.

[26] Cheney, W., and Light, W. *A Course in Approximation Theory.* Brooks/ Cole, Pacific Grove, CA, 2000.

[27] Clenshaw, C. W., and Curtis, A. R. A method for numerical integration on an automatic computer. *Numer. Math. 2* (1960), 197–205.

[28] Comtet, L. *Advanced Combinatorics.* D. Reidel Publishing Company, Dordrecht, The Netherlands, 1974.

[29] Corless, R. M., and Lawrence, P. W. Infinite eigenstructure of an arrowhead pencil. Tech. rep., in preparation, 2012.

[30] Davis, P. J. On a problem in the theory of mechanical quadratures. *Pacific J. Math. 5* (1955), 669–674.

[31] Davis, P. J. *Interpolation and Approximation.* Dover, New York, 1975.

[32] Davis, P. J., and Rabinowitz, P. *Methods of Numerical Integration, 2nd ed.* Computer Science and Applied Mathematics, Academic Press, Orlando, 1984.

[33] de Boor, C. *A Practical Guide to Splines.* Applied Mathematical Sciences, vol. 27, Springer, New York, 1978.

[34] De Boor, C., and Pinkus, A. Proof of the conjectures of Bernstein and Erdös concerning the optimal nodes for polynomial interpolation. *J. Approx. Theory 24* (1978), 289–303.

[35] de la Calle Ysern, B. Error bounds for rational quadrature formulae of analytic functions. *Numer. Math. 101* (2005), 251–271.

[36] Deckers, K., Bultheel, A., Cruz-Barroso, R., and Perdomo-Pío, F. Positive rational interpolatory quadrature formulas on the unit circle and the interval. *Appl. Numer. Math. 60* (2010), 1286–1299.

[37] Deckers, K., Van Deun, J., and Bultheel, A. Rational Gauss–Chebyshev quadrature formulas for complex poles outside $[-1, 1]$. *Math. Comp. 77* (2008), 967–983.

[38] Dupuy, M. Le calcul numérique des fonctions par l'interpolation barycentrique. *C. R. Acad. Sci., Paris 226* (1948), 158–159.

[39] Ehlich, H., and Zeller, K. Auswertung der Normen von Interpolationsoperatoren. *Math. Ann. 164* (1966), 105–112.

[40] Eisinberg, A., Fedele, G., and Franzè, G. Lebesgue constant for Lagrange interpolation on equidistant nodes. *Anal. Theory Appl. 20* (2004), 323–331.

[41] Elling, V. A Lax–Wendroff type theorem for unstructured quasi-uniform grids. *Math. Comp. 76* (2007), 251–272.

[42] Epperson, J. F. On the Runge example. *Amer. Math. Monthly 94* (1987), 329–341.

[43] Euler, L. *Opera Omnia. Series Prima: Opera Mathematica*, 5th ed., vol. 15. Teubner, Lipsiæ, 1927.

[44] Fejér, L. Mechanische Quadraturen mit positiven Cotesschen Zahlen. *Math. Z. 37* (1933), 287–309.

[45] Flanders, H. From Ford to Faà. *Amer. Math. Monthly 108* (2001), 559–561.

[46] Floater, M. S., and Hormann, K. Barycentric rational interpolation with no poles and high rates of approximation. *Numer. Math. 107* (2007), 315–331.

[47] Fornberg, B. Generation of finite difference formulas on arbitrary spaced grids. *Math. Comp. 51* (1988), 699–706.

[48] Fornberg, B. *A Practical Guide to Pseudospectral Methods*. Cambridge University Press, Cambridge, UK, 1996.

[49] Fornberg, B. Calculation of weights in finite difference formulas. *SIAM Rev. 40* (1998), 685–691.

[50] Fornberg, B., and Sloan, D. M. A review of pseudospectral methods for solving partial differential equations. In: Acta Numer., Cambridge University Press, Cambridge, UK, 1994, 203–267.

[51] Gautschi, W. Gauss-type quadrature rules for rational functions. In: H. Brass and G. Hämmerlin, Numerical integration IV, International Series of Numerical Mathematics, vol. 112, 111–130. Birkhäuser, Basel, 1993.

[52] Gautschi, W. Moments in quadrature problems. *Comput. Math. Appl. 33* (1997), 105–118.

[53] Gautschi, W. *Orthogonal Polynomials. Computation and Approximation*. Numerical Mathematics and Scientific Computation. Oxford University Press, Oxford, UK, 2004.

[54] Gonnet, P. *Adaptive Quadrature Re-Revisited*. PhD thesis, Swiss Federal Institute of Technology, Zürich, 2009.

[55] Good, I. J. The colleague matrix, a Chebyshev analogue of the companion matrix. *Q. J. Math. 12* (1961), 61–68.

[56] Güttel, S., and Klein, G. Efficient integration with equispaced nodes. In preparation.

[57] Güttel, S., and Klein, G. Convergence of linear barycentric rational interpolation for analytic functions. *SIAM J. Numer. Anal. 50* (2012), 2560–2580.

[58] Hale, N., and Tee, W. Conformal maps to multiply-slit domains and applications. *SIAM J. Sci. Comput. 31* (2009), 3195–3215.

[59] Hale, N., and Townsend, A. Fast and accurate computation of Gauss–Legendre and Gauss–Jacobi quadrature nodes and weights. Submitted.

[60] Henrici, P. *Elements of Numerical Analysis.* Wiley, New York, 1964.

[61] Henrici, P. *Essentials of Numerical Analysis with Pocket Calculator Demonstrations.* Wiley, New York, 1982.

[62] Higham, N. J. *Accuracy and Stability of Numerical Algorithms,* 2nd ed. SIAM, Philadelphia, 2002.

[63] Higham, N. J. The numerical stability of barycentric Lagrange interpolation. *IMA J. Numer. Anal. 24* (2004), 547–556.

[64] Hormann, K., Klein, G., and De Marchi, S. Barycentric rational interpolation at quasi-equidistant nodes. *Dolomites Res. Notes Approx. 5* (2012), 1–6.

[65] Horn, R. A., and Johnson, C. R. *Matrix Analysis.* Cambridge University Press, Cambridge, UK, 1985.

[66] Huybrechs, D. Stable high-order quadrature rules with equidistant points. *J. Comput. Appl. Math. 231* (2009), 933–947.

[67] Huybrechs, D. On the Fourier extension of nonperiodic functions. *SIAM J. Numer. Anal. 47* (2010), 4326–4355.

[68] Imhof, J. P. On the method for numerical integration of Clenshaw and Curtis. *Numer. Math. 5* (1963), 138–141.

[69] Isaacson, E., and Keller, H. B. *Analysis of Numerical Methods.* Wiley, New York, 1966.

[70] Jacobi, C. G. J. *Disquisitiones Analyticae de Fractionibus Simplicibus.* PhD thesis, Berlin, 1825.

[71] Johnson, W. P. The curious history of Faà di Bruno's formula. *Amer. Math. Monthly 109* (2002), 217–234.

[72] Jones, J. W., and Welfert, B. D. Zero-free regions for a rational function with applications. *Adv. Comput. Math. 3* (1995), 265–289.

[73] Kilgore, T. A. Optimization of the norm of the Lagrange interpolation operator. *Bull. Amer. Math. Soc. 83* (1977), 1069–1071.

[74] Klein, G. An extension of the Floater–Hormann family of barycentric rational interpolants. *Math. Comp.*. To appear.

[75] Klein, G., and Berrut, J.-P. Linear barycentric rational quadrature. *BIT Numer. Math. 52* (2012), 407–424.

[76] Klein, G., and Berrut, J.-P. Linear rational finite differences from derivatives of barycentric rational interpolants. *SIAM J. Numer. Anal. 50* (2012), 643–656.

[77] Königsberger, K. *Analysis 2.* 5., korrigierte Aufl. Springer, Berlin, 2004.

[78] Krommer, A. R., and Ueberhuber, C. W. *Computational Integration.* SIAM, Philadelphia, 1998.

[79] Lagrange, J. L. Leçons Élémentaires sur les Mathématiques, données à l'École Normale en 1795. In: Œuvres de Lagrange, vol. 7, 183–287. Gauthier–Villars, Paris, 1877.

[80] Meijering, E. A chronology of interpolation: from ancient astronomy to modern signal and image processing. *Proceedings of the IEEE 90* (2002), 319–342.

[81] Nocedal, J., and Wright, S. J. *Numerical Optimization.* Springer, New York, 1999.

[82] Ouspensky, J. Sur les valeurs asymptotiques des coefficients de Cotes. *Bull. Amer. Math. Soc. 31* (1925), 145–156.

[83] Platte, R. B. How fast do radial basis function interpolants of analytic functions converge? *IMA J. Numer. Anal. 31* (2011), 1578–1597.

[84] Platte, R. B., and Driscoll, T. A. Polynomials and potential theory for Gaussian radial basis function interpolation. *SIAM J. Numer. Anal. 43* (2005), 750–766.

[85] Platte, R. B., Trefethen, L. N., and Kuijlaars, A. B. J. Impossibility of fast stable approximation of analytic functions from equispaced samples. *SIAM Rev. 53* (2011), 308–318.

[86] Pólya, G. Über die Konvergenz von Quadraturverfahren. *Math. Z. 37* (1933), 264–286.

[87] Powell, M. J. D. *Approximation Theory and Methods.* Cambridge University Press, Cambridge, UK, 1981.

[88] Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. *Numerical Recipes: The Art of Scientific Computing,* 3rd ed. Cambridge University Press, Cambridge, UK, 2007.

[89] Ransford, T. *Potential Theory in the Complex Plane.* Cambridge University Press, Cambridge, UK, 1995.

[90] Rivlin, T. J. The Lebesgue constants for polynomial interpolation. In: H. G. Garnir, K. R. Unni and J. H. Williamson, Functional Analysis and its Applications. Lecture Notes in Mathematics, vol. 399, 422–437, Springer, Berlin, 1974.

[91] Rudin, W. *Real and Complex Analysis.* McGraw-Hill, New York, 1966.

[92] Runge, C. Über empirische Funktionen und die Interpolation zwischen äquidistanten Ordinaten. *Zeit. Math. Phys. 46* (1901), 224–243.

[93] Rutishauser, H. *Vorlesungen über Numerische Mathematik. Band 1: Gleichungssysteme, Interpolation und Approximation.* Birkhäuser, Basel, 1976.

[94] Saff, E. B., and Totik, V. *Logarithmic Potentials with External Fields.* Springer-Verlag, Berlin, 1997.

[95] Salazar Celis, O. *Practical Rational Interpolation of Exact and Inexact Data.* PhD thesis, Antwerpen, 2008.

[96] Salzer, H. E. Lagrangian interpolation at the Chebyshev points $x_{n,\nu} = \cos(\nu\pi/n), \nu = 0(1)n$; some unnoted advantages. *Comput. J. 15* (1972), 156–159.

[97] Schneider, C., and Werner, W. Some new aspects of rational interpolation. *Math. Comp. 47* (1986), 285–299.

[98] Schönhage, A. Fehlerfortpflanzung bei Interpolation. *Numer. Math. 3* (1961), 62–71.

[99] Schwarz, H. R. *Numerische Mathematik,* 4th ed. Teubner, Stuttgart, 1997.

[100] Smith, S. J. Lebesgue constants in polynomial interpolation. *Ann. Math. Inform. 33* (2006), 109–123.

[101] Specht, W. Die Lage der Nullstellen eines Polynoms. III. *Math. Nachr. 16* (1957), 369–389.

[102] Specht, W. Die Lage der Nullstellen eines Polynoms. IV. *Math. Nachr. 21* (1960), 201–222.

[103] Stahl, H. Convergence of rational interpolants. *Bull. Belg. Math. Soc. Simon Stevin 3* (1996), 11–32.

[104] Steffensen, J. F. *Interpolation.* 2nd ed. Chelsea, New York, 1950.

[105] Stoer, J., and Bulirsch, R. *Numerische Mathematik II.* 3., verb. Aufl. Springer, Berlin, 1990.

[106] Szabados, J., and Vértesi, P. *Interpolation of Functions.* World Scientific, Singapore, 1990.

[107] Taylor, W. J. Method of Lagrangian curvilinear interpolation. *J. res. Nat. Bur. Stand. 35* (1945), 151–155.

[108] Tee, W. *An Adaptive Rational Spectral Method for Differential Equations with Rapidly Varying Solutions.* PhD thesis, University of Oxford, UK, 2006.

[109] Tee, W., and Trefethen, L. N. A rational spectral collocation method with adaptively transformed Chebyshev grid points. *SIAM J. Sci. Comput. 28* (2006), 1798–1811.

[110] Trefethen, L. N. *Spectral Methods in* Matlab. Software, Environments, and Tools, vol. 10, SIAM, Philadelphia, 2000.

[111] Trefethen, L. N. Computing numerically with functions instead of numbers. *Math. Comput. Sci. 1* (2007), 9–19.

[112] Trefethen, L. N. Is Gauss quadrature better than Clenshaw–Curtis? *SIAM Rev. 50* (2008), 67–87.

[113] Trefethen, L. N. Six myths of polynomial interpolation and quadrature. *Maths. Today 47* (2011), 184–188.

[114] Trefethen, L. N. *Approximation Theory and Approximation Practice.* SIAM, Philadelphia, 2013.

[115] Trefethen, L. N., et al. *Chebfun Version 4.0.* The Chebfun Development Team, 2011. http://www.maths.ox.ac.uk/chebfun/.

[116] Trefethen, L. N., and Weideman, J. A. C. Two results on polynomial interpolation in equally spaced points. *J. Approx. Theory 65* (1991), 247–260.

[117] Van Assche, W., and Vanherwegen, I. Quadrature formulas based on rational interpolation. *Math. Comp. 61* (1993), 765–783.

[118] Van Deun, J. Electrostatics and ghost poles in near best fixed pole rational interpolation. *Electron. Trans. Numer. Anal. 26* (2007), 439–452.

[119] Van Deun, J., Bultheel, A., and González Vera, P. On computing rational Gauss–Chebyshev quadrature formulas. *Math. Comp. 75* (2006), 307–326.

[120] Waldvogel, J. Fast construction of the Fejér and Clenshaw–Curtis quadrature rules. *BIT Numer. Math. 46* (2006), 195–202.

[121] Wallin, H. Potential theory and approximation of analytic functions by rational interpolation. In: Proceedings of the Colloquium on Complex Analysis at Joensuu, Lecture Notes in Math., Springer, Berlin, 1979, 434–450.

[122] Walsh, J. L. *Interpolation and Approximation by Rational Functions in the Complex Domain*, 5th ed. AMS, Providence, RI, 1969.

[123] Wang, H., Huybrechs, D., and Vandewalle, S. Explicit barycentric weights for polynomial interpolation in the roots or extrema of classical orthogonal polynomials. arXiv:1202.0154.

[124] Wang, H., and Xiang, S. On the convergence rates of Legendre approximation. *Math. Comp. 81* (2012), 861–877.

[125] Waring, E. Problems concerning interpolations. *Phil. Trans. R. Soc. 69* (1779), 59–67.

[126] Weideman, J. A. C., and Laurie, D. P. Quadrature rules based on partial fraction expansions. *Numer. Algorithms 24* (2000), 159–178.

[127] Weideman, J. A. C., and Trefethen, L. N. The kink phenomenon in Fejér and Clenshaw–Curtis quadrature. *Numer. Math. 107* (2007), 707–727.

[128] Werner, W. Polynomial interpolation: Lagrange versus Newton. *Math. Comp. 43* (1984), 205–217.

[129] Winrich, L. B. Note on a comparison of evaluation schemes for the interpolating polynomial. *Comput. J. 12* (1969), 154–155.

[130] Xiang, S., and Bornemann, F. On the convergence rates of Gauss and Clenshaw–Curtis quadrature for functions of limited regularity. *SIAM J. Numer. Anal.*. To appear.

Index

Georges Klein

Personal Information

Name	Georges Klein
Date of birth	February 13, 1985
Nationality	Luxembourg
Marital status	single
Languages	Luxembourgish, French, German, English, Spanish (basics)
Website	http://homeweb.unifr.ch/kleing/pub/

Education

Feb. 2009 – Oct. 2012	PhD in Mathematics, University of Fribourg, Switzerland
Oct. 2007 – Jan. 2009	Master of Science in Mathematics, University of Fribourg, Switzerland
Oct. 2004 – Sept. 2007	Bachelor of Science in Mathematics, University of Fribourg, Switzerland, Minor: Physics
Sept. 1997 – July 2004	Diplôme de Fin d'Études Secondaires, Type B, Lycée classique d'Echternach, Luxembourg

Academic Positions

Feb. 2009 – Jan. 2013	Research assistant in the Group of Prof. Jean-Paul Berrut, Department of Mathematics, University of Fribourg
Apr. 2008 – Jan. 2009	Undergraduate assistant, Department of Mathematics, University of Fribourg

Professional Experience

Mar. 2012 – Aug. 2012	Lecturer for Analysis 2, German class, College of Engineering and Architecture of Fribourg
Sept. 2010 – Dec. 2012	Mid-faculty representative in the Council of the Department of Mathematics, University of Fribourg

Jan. 2010 – Dec. 2011	Webmaster for the Swiss Doctoral Program in Mathematics
Sept. 2009 – Dec. 2012	Teaching assistant for the exercises of the lecture Introduction to Numerical Analysis
Feb. 2009 – June 2009	Teaching assistant for the exercises of the lectures Analysis III and IV
Aug. 2008 – Dec. 2012	IT assistant for the Department of Mathematics, University of Fribourg

Conference Talks

July 2012	*Convergence of linear barycentric rational interpolation for analytic functions*, International Congress on Computational and Applied Mathematics (ICCAM2012), Ghent, Belgium
July 2012	*Convergence of linear barycentric rational interpolation for analytic functions*, International Conference on Curves and Surfaces, Oslo, Norway
July 2011	*The condition of linear barycentric rational interpolation from equispaced samples* (and chairman of the session Numerical Methods II), International Conference on Industrial and Applied Mathematics (ICIAM2011), Vancouver, Canada
June 2011	*Linear rational finite differences and applications*, 24th Biennial Conference on Numerical Analysis, University of Strathclyde, Glasgow, United Kingdom
Feb. 2011	*Linear barycentric rational interpolation from equispaced samples and applications*, 7th Graduate Colloquium of the Swiss Doctoral Program, Lausanne, Switzerland
July 2010	*Linear barycentric rational quadrature*, International Congress on Computational and Applied Mathematics (ICCAM2010), Leuven, Belgium
June 2010	*Convergence rates of derivatives of a family of barycentric rational interpolants*, International Conference on Curves and Surfaces, Avignon, France
Feb. 2010	*Selected topics in linear rational interpolation* (awarded the prize for the best talk), 6th Graduate Colloquium of the Swiss Doctoral Program, Geneva, Switzerland

Research visits

Jan. 2012	University of Oxford, visiting the group of Prof. L. N. Trefethen (1 week)
Sept. 2011	Dolomites Research Week on Approximation 2011 (DRWA11), Alba di Canazei, Italy
Nov. 2009	University of Oslo, visiting Prof. M. S. Floater (3 days)

Organization

Sept. 2012	Session "Polynomial and rational approximation" at the 3rd Dolomites Workshop on Constructive Approximation and Applications 2012 (DWCAA12), Alba di Canazei, Italy
Oct. 2012	Prof. Berrut's 60th birthday colloquium

Publications

S. Güttel, and G. Klein, *Convergence of linear barycentric rational interpolation for analytic functions*, SIAM J. Numer. Anal., **50** (2012), pp. 2560–2580

G. Klein, *An extension of the Floater-Hormann family of barycentric rational interpolants*, Math. Comp., accepted for publication

K. Hormann, G. Klein, and S. De Marchi, *Barycentric rational interpolation at quasi-equidistant nodes*, Dolomites Res. Notes Approx. **5** (2012), pp. 1–6

G. Klein, and J.-P. Berrut, *Linear rational finite differences from derivatives of barycentric rational interpolants*, SIAM J. Numer. Anal. **50** (2012), pp. 643–656

L. Bos, S. De Marchi, K. Hormann, and G. Klein, *On the Lebesgue constants of barycentric rational interpolation at equidistant nodes*, Numer. Math. **121** (2012), pp. 461–471

G. Klein, and J.-P. Berrut, *Linear barycentric rational quadrature*, BIT Numer. Math. **52** (2012), pp. 407–424

J.-P. Berrut, M. S. Floater, and G. Klein, *Convergence rates of derivatives of a family of barycentric rational interpolants*, Appl. Numer. Math. **61** (2011), pp. 989–1000