Electronic Transactions on Numerical Analysis.
Volume 41, pp. 443-464, 2014.

ISSN 1068-9613.

ETNA
Kent State University
http://etna.math.kent.edu

EFFICIENT HIGH-ORDER RATIONAL INTEGRATION AND DEFERRED CORRECTION WITH EQUISPACED DATA*

STEFAN GÜTTEL† AND GEORGES KLEIN‡

Abstract. Stable high-order linear interpolation schemes are well suited for the accurate approximation of antiderivatives and the construction of efficient quadrature rules. In this paper we utilize for this purpose the family of linear barycentric rational interpolants by Floater and Hormann, which are particularly useful for interpolation with equispaced nodes. We analyze the convergence of integrals of these interpolants to those of analytic functions as well as functions with a finite number of continuous derivatives. As a by-product, our convergence analysis leads to an extrapolation scheme for rational quadrature at equispaced nodes. Furthermore, as a main application of our analysis and target of the present paper, we present and investigate a new iterated deferred correction method for the solution of initial value problems, which allows to work efficiently even with large numbers of equispaced data. This so-called rational deferred correction (RDC) method turns out to be highly competitive with other methods relying on more involved implementations or non-equispaced node distributions. Extensive numerical experiments are carried out, comparing the RDC method to the well established spectral deferred correction (SDC) method by Dutt, Greengard and Rokhlin.

Key words. Quadrature, barycentric rational interpolation, extrapolation, initial value problems, deferred correction.

AMS subject classifications. 65D05, 41A20, 65D30, 65B05.

1. Introduction.

We are concerned with the problem of approximating the antiderivative $\int_a^y f(x)\,\mathrm{d}x$, for possibly many values $y \in [a,b]$, of a function f which is sufficiently smooth, defined on a real interval $[a,b]$ and not periodic. More precisely, we are interested in computing this integral to high precision when the function is given via its values f_i at $n+1$ equispaced nodes $x_i = a + ih$, $h = (b-a)/n$, $i = 0, 1, \ldots, n$. If y is one of these nodes, then one can simply apply a quadrature rule, or a combination of several rules, restricted to the interval $[a,y]$. Otherwise, one can compute the integrals for each node using a quadrature rule as described above and interpolate between these results to obtain an approximation of the antiderivative which is then defined throughout the interval $[a,b]$. There are of course many other possibilities and we will describe our idea (DRI) after the following outline on quadrature rules and methods derived from linear rational interpolation.

For the approximation of integrals, common and basic quadrature rules include the composite trapezoid and Simpson's rules, which converge at the rate $O(h^2)$ and $O(h^4)$, respectively. Higher order methods, e.g., the Newton–Cotes rules, can be obtained from integrating the unique polynomial interpolant of degree $\leq n$ of the given data. It is however well-known, that in finite precision arithmetic the approximations from these rules diverge with increasing n because of ill-conditioning; see [34]. Even in exact arithmetic, when f is not analytic in a sufficiently large region around the nodes, divergence will occur due to Runge's phenomenon [10, 38]. If the nodes need not be equispaced, then very efficient and stable classical rules, such as Gauss–Legendre and Clenshaw–Curtis rules, are readily available; see, e.g., [9, 18, 42]. We will assume here that the nodes are required to be equispaced.

Various higher order quadrature methods for equispaced nodes are available, e.g., composite Newton–Cotes rules, Newton–Gregory rules [23], schemes based on splines [12], least

*Received August 15, 2014. Accepted August 22, 2014. Published online on December 18, 2014. Recommended by Lothar Reichel. The second author's work was supported by the Swiss National Science Foundation under Grant No. PBFRP2-142826.

†School of Mathematics, The University of Manchester, Oxford Road, Manchester M13 9PL, United Kingdom, (stefan.guettel@manchester.ac.uk).

‡Department of Mathematics, University of Fribourg, Pérolles, 1700 Fribourg, Switzerland, (georges.klein@unifr.ch).

ETNA
Kent State University
http://etna.math.kent.edu

squares approximations [24], Fourier extensions [25], and rational interpolation, to name just a few. For a general overview of the topic we refer to [11, 30]. Direct rational quadrature (DRQ) [29] is based on the barycentric rational interpolation scheme proposed by Floater and Hormann [14] and consists of interpolating the given data with a linear rational interpolant and afterwards integrating that interpolant with an efficient method to obtain an approximation of the integral. We will further investigate this scheme and proceed to recall its essential ingredients.

Given a nonnegative integer $d \leq n$, the so-called *blending parameter*, consider polynomial interpolants p_i of degree $\leq d$ for the nodes $x_i, \ldots, x_{i+d}$ and data $f_i, \ldots, f_{i+d}$ ($i = 0, \ldots, n-d$). The rational interpolant is then a "blend" of these polynomial interpolants,

$$r_n(x) = \frac{\sum_{i=0}^{n-d} \lambda_i(x) p_i(x)}{\sum_{i=0}^{n-d} \lambda_i(x)}, \qquad \lambda_i(x) = \frac{(-1)^i}{(x-x_i)\cdots(x-x_{i+d})}. \tag{1.1}$$

This rational interpolant, which is linear in the data $f_0, \ldots, f_n$, can be written in barycentric form

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

for its evaluation. For details and formulas of the barycentric weights w_i, which do not depend on the f_i, see [14]. From now on, we will always refer to this family of interpolants each time we use the expression "linear (barycentric) rational interpolation". For functions in $C^{d+2}[a,b]$ and equispaced nodes, the convergence rate is $O(h^{d+1})$. For analytic functions and variable d, asymptotic rates of convergence depending on the region of analyticity of f have been established in [19]. We will recall and expand the corresponding results for equispaced nodes in Section 2. The key idea in our analysis in [19] was to allow d to vary with n so as to balance the convergence speed and the condition number in order to obtain fast convergence with small values of n until the relative error comes close to machine precision and to maintain, or even improve, that accuracy with larger values of n. This study allowed to investigate the behaviour of linear rational interpolation with respect to the barrier from [37] on the convergence speed of a well-conditioned approximation method from equispaced nodes. We remind the reader that it is proven in [37] that every geometrically converging approximation scheme from equispaced samples is ill-conditioned. The condition number associated with linear interpolation is the maximum of the Lebesgue function [39], the Lebesgue constant. For linear barycentric rational interpolation it is given by

$$\Lambda_{n,d} = \max_{a\leq x\leq b} \Lambda_{n,d}(x) = \max_{a\leq x\leq b} \sum_{i=0}^{n} \left|\frac{w_i}{x-x_i}\right| \Big/ \left|\sum_{i=0}^{n} \frac{w_i}{x-x_i}\right|.$$

The Lebesgue constant associated with linear rational interpolation for equispaced nodes is bounded as (see [4], and [26] for tighter bounds)

$$\frac{1}{2^{d+2}} \binom{2d+1}{d} \log\left(\frac{n}{d} - 1\right) \leq \Lambda_{n,d} \leq 2^d\big(1 + \log(n)/2\big), \tag{1.2}$$

where the leading factor in the lower bound is larger than $2^{d-2}/(d+1)$; see, e.g., section 3.3.3 in [28]. This clearly shows that the Lebesgue constant grows exponentially with increasing d.

ETNA
Kent State University
http://etna.math.kent.edu

A quadrature rule can be obtained with the above linear rational interpolant as follows (see [29]):

$$I[f] \approx \int_a^b r_n(x)\,\mathrm{d}x = \sum_{i=0}^{n} f_i \int_a^b \frac{w_i}{x-x_i} \Big/ \sum_{k=0}^{n} \frac{w_k}{x-x_k}\,\mathrm{d}x = \sum_{i=0}^{n} \omega_i f_i, \tag{1.3}$$

where the quadrature weights are defined by

$$\omega_i = \int_a^b \frac{w_i}{x-x_i} \Big/ \sum_{k=0}^{n} \frac{w_k}{x-x_k}\,\mathrm{d}x. \tag{1.4}$$

The sum of the absolute values of the quadrature weights, $\sum_{i=0}^{n} |\omega_i|$, is a useful measure of the stability of a quadrature rule, as it is a bound on the amplification of inaccuracies in the given data. An interpolatory quadrature rule usually inherits the properties of the interpolation scheme from which it is derived. This is also the case for the stability of the rational quadrature rule in the right-hand side of (1.3). The sum of the absolute values of the quadrature weights can be related to the Lebesgue constant as

$$\sum_{i=0}^{n} |\omega_i| \le \int_a^b \sum_{i=0}^{n} \left| \frac{\frac{w_i}{x-x_i}}{\sum_{i=0}^{n} \frac{w_i}{x-x_i}} \right| \mathrm{d}x = \int_a^b \Lambda_{n,d}(x)\,\mathrm{d}x \le (b-a)\Lambda_{n,d}. \tag{1.5}$$

This is of course a crude upper bound, since the left-hand side is equal to $(b-a)$ if all the ω_i are positive. It will, however, be sufficient for our analysis, since the quadrature weights in DRQ rules with $d \ge 6$ are not all positive, as was observed experimentally in [29].

Since there is no closed formula at the present time for the computation of the quadrature weights (1.4), these may be approximated efficiently by a sufficiently accurate quadrature rule, e.g., Gauss–Legendre, Clenshaw–Curtis quadrature as implemented in Matlab, for example, in the Chebfun system[1]. Notice that the integrand in (1.4) can be evaluated stably and efficiently at every point in $[a,b]$; see section 3.2 in [28] or [22] for details within the polynomial framework which also apply to our setting. If the quadrature weights ω_i are not needed explicitly, then one of the already mentioned classical quadrature rules may be applied directly on r_n for faster approximation of $I[f]$; this resulting type of method is what is called direct rational quadrature (DRQ) in [29]. For a sufficiently smooth function f and with equispaced nodes, the DRQ rules converge at the rate $O(h^{d+2})$, as was shown in [29]. Very similar strategies can be applied to construct approximations of integrals, as we will present in Section 2, namely by approximating the antiderivative of a linear rational interpolant r_n with another already existing efficient method; this type of method will be called direct rational integration (DRI).

In Section 2 of this work we establish a new asymptotic convergence result for linear rational integration and the quadrature rule (1.3) applied to an analytic function f, with equispaced nodes and a variable blending parameter d chosen proportional to n. We then complement this result with another theorem on the rate of convergence for the approximation of antiderivatives of functions with finitely many continuous derivatives, under the condition that d is fixed. Based on the analogous result from [29] for the rational quadrature rule (1.3), we construct a Richardson-type extrapolation scheme. We illustrate all our theoretical findings with some numerical examples in Section 3.

In Section 4 we apply the rational integration scheme DRI to the solution of initial value problems via the iterated deferred correction method; this application is the target of our

[1]The open source software Chebfun is available at http://chebfun.org.

ETNA
Kent State University
http://etna.math.kent.edu

paper. This method can be viewed as an extrapolation scheme that iteratively eliminates terms in the error expansion of an initial low-order approximation to the solution, without using finer grids as would be the case with Richardson extrapolation. This idea is not new, it at least dates back to work by Zadundaisky [44] and Pereyra [35, 36] who in the 1960s developed methods for equispaced data, and was popularized again around the year 2000 by Dutt, Greengard & Rokhlin [13], who presented a more stable method by combining the Picard integral formulation of the initial value problem with spectral interpolation at Gauss–Legendre nodes. This combination is commonly called *spectral deferred correction* (SDC). Building on the integral formulation idea we present a new deferred correction method, called *rational deferred correction* (RDC), which uses linear barycentric rational interpolation in *equispaced* time points. Our theoretical results and numerical observations in Sections 2–3 allow us to give insight into the convergence behavior of RDC. This will be demonstrated with several numerical examples, like the Brusselator, Van der Pol, and Burgers' equation. We find that with a careful choice of parameters, the attained accuracy of RDC is competitive with SDC, and sometimes even slightly better. Computationally, RDC may be advantageous in particular for high-dimensional stiff problems, because it allows for constant time steps and may therefore require fewer recomputations or refactorizations of Jacobians. We conclude this paper with a summary and some ideas for future work.

2. Rational integration from equispaced samples. The integration scheme we are describing is based on linear barycentric rational interpolation and the DRQ rules from [29]. We will couple the obtained scheme with a recommendation for how to choose the blending parameter d, depending on the region of analyticity of the integrand. We focus on equispaced nodes in the following, partly to keep the exposition simple, but mainly because it is a situation often encountered in practice and most relevant for our application in Section 4. Moreover, the error analysis from [19], which we build upon, can be simplified if the interpolation nodes are equispaced. We only mention here that all what follows could also be done with more general node distributions.

We begin by expanding an asymptotic bound on the interpolation error with equispaced nodes, showing how it can be generalized to quadrature, and giving a strategy for a good choice of d. Building upon our work in [19], for every n, we seek for an optimal $C \in (0,1]$ which will determine the blending parameter as

$$d = d(n) = \operatorname{round}(Cn).$$

With this choice, the relative error decreases in practical computations as fast as possible without being dominated by the amplification of rounding errors.

We start from the Hermite-type error formula

$$f(x) - r_n(x) = \frac{1}{2\pi \mathrm{i}} \int_{\mathcal{C}} \frac{f(s)}{s-x} \cdot \frac{\sum_{i=0}^{n-d} \lambda_i(s)}{\sum_{i=0}^{n-d} \lambda_i(x)} \, \mathrm{d}s, \tag{2.1}$$

where $\mathcal{C}$ is a contour in a region around the nodes which does not contain any singularities of f. This error formula (2.1) was obtained in [19] from the Cauchy integral formula for f and the explicit representation of the rational interpolant of $1/(s-x)$. To estimate $f - r_n$, we need to bound $|\sum_{i=0}^{n-d} \lambda_i(s)|$ from above for $s \in \mathcal{C}$, and $|\sum_{i=0}^{n-d} \lambda_i(x)|$ from below for $x \in [a,b]$. A bound on the latter was obtained in [14], the asymptotic analogue of which was given in [19] as

$$\liminf_{n\to\infty} \left| \sum_{i=0}^{n-d(n)} \lambda_i(x) \right|^{1/n} \geq \left(\frac{e}{C(b-a)} \right)^C =: V_0. \tag{2.2}$$

ETNA
Kent State University
http://etna.math.kent.edu

To estimate $|\sum_{i=0}^{n-d}\lambda_i(s)|$, we observe that

$$\left|\sum_{i=0}^{n-d}\lambda_i(s)\right| \le (n-d+1)|\lambda_j(s)|, \tag{2.3}$$

where j is the index of a term $\lambda_i(s)$ with maximal absolute value, which also depends on the location of s. The node x_j, whose index j corresponds to the one in (2.3), satisfies

$$x_j \in \left[\max\left\{a, \mathrm{Re}(s) - \left\lfloor\frac{d}{2}\right\rfloor h\right\}, \max\left\{a, \mathrm{Re}(s) - \left\lfloor\frac{d}{2}+1\right\rfloor h\right\}\right].$$

This follows from the definition of the λ_i in (1.1) and the fact that the nodes are equispaced. Any λ_j can be rewritten in the form

$$|\lambda_j(s)| = \exp\left(-\sum_{i=0}^{d}\log|s-x_{j+i}|\right).$$

Let us define $\alpha = \alpha(s) = \max\{a, \mathrm{Re}(s) - C(b-a)/2\}$. As $n\to\infty$, we have

$$\sum_{i=0}^{d}\log|s-x_{j+i}| \to \frac{d+1}{C(b-a)}\int_{\alpha}^{\alpha+C(b-a)}\log|s-x|\,\mathrm{d}x,$$

so that, for $d/n \to C$ and upon taking the n-th root, we arrive at

$$\limsup_{n\to\infty}\left|\sum_{i=0}^{n-d}\lambda_i(s)\right|^{1/n} \le \exp\Big(-\int_{\alpha}^{\alpha+C(b-a)}\frac{\log|s-x|}{b-a}\,\mathrm{d}x\Big).$$

The above integral can be further evaluated and amounts to (see [15, Section 3.4] and [19, Section 2.3])

$$\begin{aligned}\int_{\alpha}^{\alpha+C(b-a)}\frac{\log|s-x|}{b-a}\,\mathrm{d}x &= C\log\Big(\frac{C(b-a)}{2e}\Big)\\ &\quad + \frac{C}{2}\mathrm{Re}\big((1-s')\log(1-s') - (-1-s')\log(-1-s')\big)\\ &=: -\log(V_0) - \log(2^C) - \log\big(V(s)\big),\end{aligned}$$

with $s' = (2s-2\alpha)/(Cb-Ca) - 1$ and

$$V(s) = \left|\frac{(-1-s')^{-1-s'}}{(1-s')^{1-s'}}\right|^{C/2}.$$

This now gives

$$\limsup_{n\to\infty}\left|\sum_{i=0}^{n-d}\lambda_i(s)\right|^{1/n} \le 2^C V_0 V(s). \tag{2.4}$$

The bound on the error in the quadrature rule for sufficiently large n now follows from integrating both sides of (2.1) with respect to x over $[a,b]$, the standard estimation of integrals and both (2.2) and (2.4),

$$\left|\int_a^b f(x)\,\mathrm{d}x - \int_a^b r_n(x)\,\mathrm{d}x\right| \le \frac{(b-a)\,\mathrm{length}(\mathcal{C})\max_{s\in\mathcal{C}}|f(s)|}{2\pi\,\mathrm{dist}([a,b],\mathcal{C})}\,2^{Cn}V(s)^n.$$

ETNA
Kent State University
http://etna.math.kent.edu

Let us define the following contours for later reference, their typical shape is displayed in Figure 2.2 in [19]:

$$\mathcal{C}_R := \{z \in \mathbb{C} : 2^C V(z) = R\}. \tag{2.5}$$

The above reasoning can be trivially extended to the approximation of antiderivatives of f, especially

$$F(y) = \int_a^y r_n(x)\,\mathrm{d}x, \qquad y \in [a,b], \tag{2.6}$$

the one with value 0 at the left end of the interval. The study of the conditioning of the quadrature described in the Introduction (see (1.5)) also carries over to the approximation of antiderivatives with little effort, so that the Lebesgue constant multiplied by $(b-a)$ may again be taken as a crude but valid upper bound on the condition number. Similarly to practical computations with the quadrature rule, we suggest to first approximate r_n to machine precision by a polynomial, which is then integrated to give an approximation of F in (2.6). A convenient way of implementing this is by using the Chebfun `cumsum` command. Note that the result of `cumsum` is a polynomial approximation to the antiderivative F, which can be cheaply evaluated at any point y. We call *direct rational integration* (DRI) the combination of these methods. An alternative would be to multiply the vector of the function values by an integration matrix whose entries are the integrals from a to each node of all the rational basis functions. These integrals can be computed with any efficient quadrature rule based on any distribution of nodes. We always assume that the integral of r_n can be approximated close to machine precision, and therefore consider this additional error as practically negligible. The computational complexity is relatively low, since it involves the evaluation of r_n, which takes $O(n)$ operations, combined with the approximation procedure in Chebfun, which is targeted at being as efficient as possible.

We now summarize the above derivation as our first main theorem.

THEOREM 2.1. *Let f be a function analytic in an open neighborhood 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 (2.5). Then the antiderivative with value 0 at $x = a$ of the rational interpolant r_n defined by (1.1), with equispaced nodes and $d(n)/n \to C$, satisfies*

$$\limsup_{n\to\infty} \left| \int_a^y f(x)\,\mathrm{d}x - \int_a^y r_n(x)\,\mathrm{d}x \right|^{1/n} \le R,$$

for any $y \in [a,b]$.

Notice that this result includes the convergence of the quadrature rules as the special case when $y = b$, and a similar bound may be obtained for any other distribution of nodes with a slightly modified asymptotic bound, obtained from that for the interpolant in [19] combined with the standard estimation of integrals.

As already mentioned, one cannot expect exponential convergence of an approximation scheme with equispaced nodes in practice, and the same is true for the quadrature scheme we are concerned with, since it is derived from linear rational interpolation with equispaced nodes. Theorem 2.1 does not take into account the conditioning of the numerical approximation problem, which is however essential in numerical computations with equispaced nodes. We suggest to apply the stabilization from [19] also for the approximation of antiderivatives as follows: we minimize the sum of the theoretical error from Theorem 2.1 and the amplification of relative rounding errors, which is the product of the machine precision ε_{M} (typically

ETNA
Kent State University
http://etna.math.kent.edu

$\varepsilon_{\mathrm{M}} = 2^{-52} \approx 2.22 \cdot 10^{-16}$) and the Lebesgue constant bounded by (1.2), to find an appropriate value of C and thus d. We proceed by collecting all the constants into K and obtain that, for sufficiently large n,

$$\max_{a\le y\le b}\left|\int_a^y f(x)\,\mathrm{d}x - \int_a^y r_n(x)\,\mathrm{d}x\right| \le K(R^n + \varepsilon_{\mathrm{M}}\,\Lambda_{n,d}) = K2^{Cn}(V(z)^n + \varepsilon_{\mathrm{M}}(1+\log(n)/2)). \tag{2.7}$$

With the knowledge of the singularity s of f closest to the interval (in terms of the level lines $\mathcal{C}_R$ defined in (2.5)), we can then search for a value of C which minimizes the above right-hand side, and thus the relative numerical error to be expected. We will illustrate this procedure with an example in Section 3.

Let us now investigate the convergence rate of DRI with equispaced nodes and fixed d for functions with a finite number of continuous derivatives; this number can obviously be as small as 1. The theorem below shows that the approximation order is $d+2$, and for the special case of DRQ (i.e., $x = b$) this was proven in [29]. Moreover, the constant factor in the upper bound on the error merely depends on the norm of a low order derivative of f, which depends on d and not on n. The proof below is quite technical; we use several tools with the same notation as in [14, 29], in the hope that this will help the reader.

THEOREM 2.2. *Assume n and d, $0 < d \le n/2-1$, are positive integers, $f \in C^{d+3}[a,b]$, and the nodes are equispaced. Then for any $x \in [a,b]$,*

$$\left|\int_a^x f(y) - r_n(y)\,\mathrm{d}y\right| \le Kh^{d+2},$$

where K is a constant depending only on d, on derivatives of f, and on the length $(b-a)$ of the interval.

Proof. Throughout this proof, we generically denote by K any constant factor that does not depend on n. Since $x \in [a,b]$ is fixed, we can write it as $x = a + Th$, with $T \in [0,n]$. We begin by showing why we only need to study the claimed bound in details for $T \in \mathcal{K} := \{k = d+1, d+3, \ldots : k \le n-d-1\}$. All other cases are easily dealt with for the following reasons. Note that we always regroup two subintervals together, which leads to the fact that $\mathcal{K}$ only involves odd indexes.

If $T < d+3$, then by the standard estimation of integrals and the bound on the interpolation error, it follows that

$$\left|\int_a^x f(y) - r_n(y)\,\mathrm{d}y\right| \le \int_a^x |f(y) - r_n(y)|\,\mathrm{d}y \le |x-a|Kh^{d+1} \le (d+3)Kh^{d+2}. \tag{2.8}$$

For $T \in [d+3, n-d-1] \setminus \mathcal{K}$, we call k the largest element of $\mathcal{K}$ smaller than T. Then,

$$\begin{aligned}\left|\int_a^x f(y) - r_n(y)\,\mathrm{d}y\right| &\le \int_a^{x_{d+1}} |f(y) - r_n(y)|\,\mathrm{d}y + \left|\int_{x_{d+1}}^{x_k} f(y) - r_n(y)\,\mathrm{d}y\right| \\ &\quad + \int_{x_k}^{x} |f(y) - r_n(y)|\,\mathrm{d}y \\ &\le (d+1)Kh^{d+2} + \left|\int_{x_{d+1}}^{x_k} f(y) - r_n(y)\,\mathrm{d}y\right|.\end{aligned}$$

ETNA
Kent State University
http://etna.math.kent.edu

With $T > n-d-1$, we proceed similarly,

$$\begin{aligned}\left|\int_a^x f(y)-r_n(y)\,\mathrm{d}y\right| &\le \int_a^{x_{d+1}} |f(y)-r_n(y)|\,\mathrm{d}y + \left|\int_{x_{d+1}}^{x_{n-d-1}} f(y)-r_n(y)\,\mathrm{d}y\right| \\ &\quad + \int_{x_{n-d-1}}^{x} |f(y)-r_n(y)|\,\mathrm{d}y \\ &\le Kh^{d+2},\end{aligned}$$

where the first and third terms are bounded using the standard estimation of integrals as in (2.8), and the bound on the middle term comes from the proof of Theorem 6.1 in [29].

For the investigation of the claimed error bound with $T \in \mathcal{K}$ we denote the interpolation error as in [14, 29] for any $y \in [a,b]$ by

$$f(y)-r_n(y) = \frac{\sum_{i=0}^{n-d}(-1)^i f[x_i,\dots,x_{i+d},y]}{\sum_{i=0}^{n-d}\lambda_i(y)} = \frac{A(y)}{B(y)}, \tag{2.9}$$

where A and B are defined in the obvious way, and we call

$$\Omega_n(x) = \int_{x_{d+1}}^{x} \frac{1}{B(y)}\,\mathrm{d}y.$$

We may now apply integration by parts to split the approximation error into two parts,

$$\int_{x_{d+1}}^{x} \frac{A(y)}{B(y)}\,\mathrm{d}y = A(x)\Omega_n(x) - \int_{x_{d+1}}^{x} A'(y)\Omega_n(y)\,\mathrm{d}y, \tag{2.10}$$

and study the first term in a first step. The numerator A can be bounded by a constant; see the proofs of Theorems 2 and 3 in [14]. With the change of variable $y = a+th$ we rewrite Ω_n as

$$\Omega_n(x) = h^{d+2}\int_{d+1}^{T} \frac{1}{\overline{B}(t)}\,\mathrm{d}t,$$

where $\overline{B}$ is the B from (2.9) after changing variables and neglecting the powers of h. Analogously, we define $\overline{\lambda_i}$ from λ_i and show that the integral of the reciprocal of $\overline{B}$ is bounded by a constant. From the definition of B, we have that

$$\int_{d+1}^{T} \frac{1}{\overline{B}(t)}\,\mathrm{d}t = \sum_{k=d+1}^{T-2}\int_k^{k+2} \frac{1}{\overline{B}(t)}\,\mathrm{d}t = \sum_{k=d+1}^{T-2}\int_k^{k+1} \frac{\overline{\lambda}_0(t+1)+\overline{\lambda}_{n-d}(t)}{\overline{B}(t)\overline{B}(t+1)}\,\mathrm{d}t.$$

Both functions $\overline{\lambda}_0(t+1)$ and $\overline{\lambda}_{n-d}(t)$ do not change sign in the subintervals $[d+1, n-d-1]$, so that we can apply to each of them the mean value theorem for integrals. Moreover, it can be deduced easily from the proofs of Theorems 2 and 3 in [14] that the reciprocal of $\overline{B}$ may be bounded by a constant, and therefore

$$\begin{aligned}\left|\int_{d+1}^{T} \frac{1}{\overline{B}(t)}\,\mathrm{d}t\right| &\le \sum_{k=d+1}^{T-2}\int_k^{k+1} \left|\frac{\overline{\lambda}_0(t+1)}{\overline{B}(t)\overline{B}(t+1)}\right|\mathrm{d}t + \sum_{k=d+1}^{T-2}\int_k^{k+1} \left|\frac{\overline{\lambda}_{n-d}(t)}{\overline{B}(t)\overline{B}(t+1)}\right|\mathrm{d}t \\ &\le K\left(\left|\int_{d+1}^{T}\overline{\lambda}_0(t+1)\,\mathrm{d}t\right| + \left|\int_{d+1}^{T}\overline{\lambda}_{n-d}(t)\,\mathrm{d}t\right|\right).\end{aligned}$$

ETNA
Kent State University
http://etna.math.kent.edu

Since the nodes are equispaced and because of the structure of λ_i from (1.1), we have the following partial fraction decomposition,

$$\overline{\lambda}_0(t+1) = \frac{(-1)^d}{d!}\sum_{i=0}^{d}(-1)^i\binom{d}{i}\frac{1}{t+1-i},$$

$$\overline{\lambda}_{n-d}(t) = \frac{(-1)^{n-d}}{d!}\sum_{i=0}^{d}(-1)^i\binom{d}{i}\frac{1}{t-(n-i)}.$$

On one hand,

$$\frac{d!}{(-1)^d}\int_{d+1}^{T}\overline{\lambda}_0(t+1)\,\mathrm{d}t = \log\left(\frac{P(T)}{Q(T)}\right) - \sum_{i=0}^{d}(-1)^i\binom{d}{i}\log(d+2-i),$$

which goes to a constant as $n \to \infty$, since the second term does not depend on n and in the first one P and Q are monic polynomials in the same degree in T, which increases as n increases. On the other hand,

$$\frac{d!}{(-1)^{n-d}}\int_{d+1}^{T}\overline{\lambda}_{n-d}(t)\,\mathrm{d}t = \log\left(\frac{P(n-T)}{Q(n-T)}\right) - \log\left(\frac{P(n)}{Q(n)}\right),$$

which tends to 0 as n increases for similar reasons as those stated above. We have just established that the first term in (2.10) is bounded by Kh^{d+2}. In a second step we treat the second term. We recall Lemma 6.2 from [29], which states that Ω_n does not change sign in $[x_{d+1}, x_{n-d-1}]$, and we apply again the mean value theorem, i.e.,

$$\int_{x_{d+1}}^{x} A'(y)\Omega_n(y)\,\mathrm{d}y = A'(\xi)\int_{x_{d+1}}^{x}\Omega_n(y)\,\mathrm{d}y$$

for some $\xi \in [x_{d+1}, x_{n-d-1}]$. The absolute value of A' can be bounded by a constant; see Lemma 2 in [3]. We proceed by integration by parts,

$$\int_{x_{d+1}}^{x}\Omega_n(y)\,\mathrm{d}y = (x-a)\Omega_n(x) - \int_{x_{d+1}}^{x}\frac{y-a}{B(y)}\,\mathrm{d}y.$$

The first term has been treated above and for the second we use the preceding change of variables

$$\begin{aligned}\int_{x_{d+1}}^{x}\frac{y-a}{B(y)}\,\mathrm{d}y &= h^{d+3}\int_{d+1}^{T}\frac{t}{\overline{B}(t)}\,\mathrm{d}t\\ &= h^{d+3}\sum_{k=d+1}^{T-2}\int_{k}^{k+1}\frac{t\big(\overline{\lambda}_0(t+1)+\overline{\lambda}_{n-d}(t)\big)+\overline{B}(t)}{\overline{B}(t)\overline{B}(t+1)}\,\mathrm{d}t,\end{aligned}$$

and further process the latter expression analogously to the first step by making use of the additional power of h. ∎

The above Theorem 2.2 states that for fixed d the error in the quadrature is of the order of h^{d+2}. Such a bound allows to apply Richardson extrapolation; see, e.g., [11]. Since for a given n the quadrature error is bounded by K/n^{d+2} for some constant K, the error with only

ETNA
Kent State University
http://etna.math.kent.edu

half as many nodes and the same d is bounded by $2^{d+2}K/n^{d+2}$. Therefore, if we denote by I_n the rational quadrature rule with $n+1$ nodes, the extrapolated rule

$$\widetilde{I}_n = \frac{2^{d+2}I_n - I_{n/2}}{2^{d+2}-1} \tag{2.11}$$

will be of the order of h^{d+3} at least. In the generic case, one cannot expect that the order further increases when Richardson extrapolation with the obvious modifications is iterated on the present type of rational quadrature. The error in the quadrature scheme or in the underlying interpolation process must indeed not necessarily be representable as an expansion in regularly increasing powers of h, which exists for instance for the error of the trapezoid rule under some assumptions. An example of the three approaches to rational integration studied in the present section is provided at the end of the next section on numerical examples.

In Section 4 we will apply rational quadrature for the solution of initial value problems via iterated deferred correction. That method has the same aim as Richardson extrapolation, namely iteratively eliminate terms in the error expansion corresponding to increasing powers of h, but it does not require to use a finer grid. The iterative scheme will be applied on an initial solution obtained using a first order method and whose error can be written as an expansion in powers of h.

3. Numerical examples on integration. To illustrate the theoretical results from the previous section, we present a numerical example for each of the three topics, namely the fast convergence of the approximation of antiderivatives with variable d, the convergence behaviour with constant blending parameter, and Richardson extrapolation. We stress that all the examples displayed are computed exclusively from equispaced data.

To better understand the results and to situate the rational integration methods within classical ones, we also display the error in the approximation of antiderivatives using B-splines of degree $d+1$ and in the quadrature with Newton–Gregory rules for the examples with fixed convergence rates.

The behavior of the error in the approximation of antiderivatives with variable d is closely related to that of the interpolation of functions in the same setting, which we analyzed in greater detail in [19]. We show here the stabilization procedure from (2.7) with numerical optimization in practice, with two functions, namely $f_1(x) = e^{1/(x+2)}$ and $f_2(x) = e^{1/(x+1.5)}$ on $[-1,1]$. Both functions have a singularity to the left of the interval at different distances. The maximal relative error of the approximation of the antiderivative on the interval therefore decays at different speeds, faster when the singularity lies further away; see Figure 3.1. In addition, we show the smallest error attainable with an optimal choice of d found by searching for the smallest error with all admissible values of d. In the pictures on the right, we show the values of the blending parameter involved in the examples, the one on top refers to f_1 and the other one to f_2. We observe that we can increase d linearly with n as long as the latter is small enough, but then need to decrease d again so as to guarantee that the condition number does not exceed the error from exact arithmetic. One observes geometric convergence with small n and thereafter algebraic error decay with varying rates. Observe that we conservatively overestimate the condition number as explained earlier, so that the error curves lie a bit above the smallest attainable errors.

Our second example is the approximation of an antiderivative of $1/(1+5x^2)$ on $[-1,1]$ with constant values of the blending parameter, namely $d = 1, 3, 5, 9$; see Figure 3.2. We compare these results with those obtained from spline integration computed using Matlab's `fnint` and the `spapi` command from the "curve fitting" toolbox. The degree of the splines is $d+1$ for each curve labelled d; it is chosen such that the rates of convergence match those of the rational approximations. For $n \geq 30$, the maximal absolute errors over the

ETNA
Kent State University
http://etna.math.kent.edu

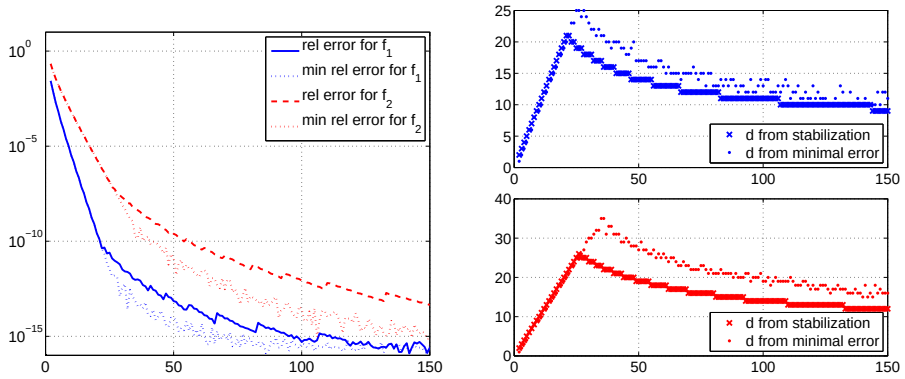

FIGURE 3.1. *Relative errors (left) in the approximation of an antiderivative of* $f_1(x) = e^{1/(x+2)}$ *and* $f_2(x) = e^{1/(x+1.5)}$ *on* $[-1,1]$ *with* $2 \le n \le 150$*, compared to the smallest error with an optimal* d*. The pictures on the right (top:* f_1*, bottom:* f_2*) compare the values of* d *chosen by the stabilization and the optimal values.*

interval decrease at the algebraic rates $d+2$ as expected from Theorem 2.2. The rates of the spline based approximations are similar, the constant in the error term is larger with small degree. The curves with larger degree almost coincide. The rational approximations deliver however an analytic approximation whereas the splines are merely a few times continuously differentiable. Observe that with DRI and smaller n the error decreases at the much faster exponential rate as was the case when d increased with n in the example above, but here d is kept constant. This behavior is not restricted to the integrand under consideration. We observed that it appears whenever f has a singularity close to the middle of the interval, more precisely inside the curve that joins some of the complex poles of the rational interpolant. We propose to investigate this important effect more in detail in the future. In the theory part of the present paper we have not taken this effect into account.

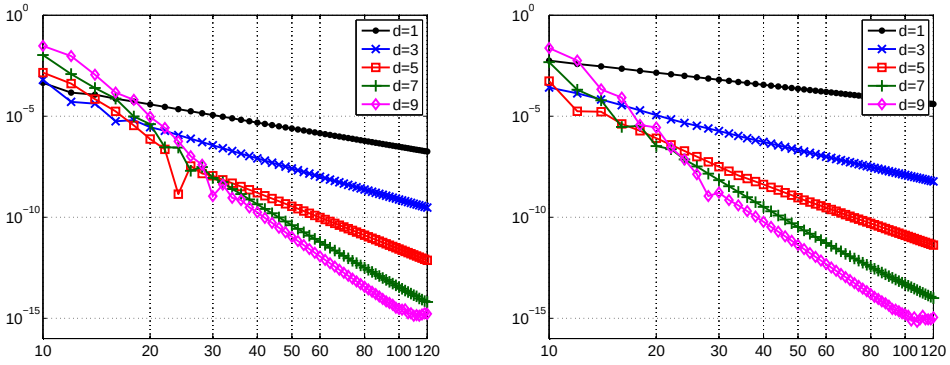

FIGURE 3.2. *Absolute errors in the approximation of an antiderivative of* $1/(1+5x^2)$ *on* $[-1,1]$ *with* $10 \le n \le 120$ *and various values of the theoretical convergence rates* $d+2$*; left: DRI and right: antiderivatives of B-splines of degree* $d+1$*.*

Before ending this section, we present an example of Richardson extrapolation outlined at the end of the previous section; see (2.11). In Table 3.1, we give the absolute errors and the corresponding experimental convergence rates for the rational quadrature scheme with $d=2$ and one iteration of Richardson extrapolation for the approximation of $\int_{-1}^{1} e^{1/(1+x^2)}\,\mathrm{d}x$. The experimental order has increased almost exactly by 1 from $d+2$ to $d+3$ after one

ETNA
Kent State University
http://etna.math.kent.edu

Richardson iteration. The same convergence rates with equispaced nodes can be seen with Newton–Gregory rules, which are obtained from successively "correcting" the trapezoid rule; see, e.g., [23] for the explicit construction. More precisely, the terms involving higher order derivatives of the integrand in the Euler–Maclaurin formula for the error of the trapezoid rule are approximated via expansions of forward and backward differences. Collecting the values of the integrand at the nodes yields weights for higher order quadrature rules. In this construction, only the weights corresponding to a few values near the ends of the interval are affected. These rules are efficient for the approximation of integrals, but for the approximation of antiderivatives one would need further constructions which are less readily available than antiderivatives of linear rational interpolants.

TABLE 3.1
Absolute errors in the direct rational quadrature (DRQ) of $e^{1/(1+x^2)}$ on $[-1,1]$ with $d = 2$, Richardson extrapolation as well as 4th and 5th order Newton–Gregory rules (NG) together with experimental convergence orders.

	DRQ, $d=2$		extrapolation		4th order NG		5th order NG	
n	error	order	error	order	error	order	error	order
10	2.04e−04				2.80e−05		2.27e−04	
20	1.22e−05	4.07	6.20e−07		5.83e−06	2.26	3.18e−06	6.17
40	7.41e−07	4.04	2.26e−08	4.78	4.34e−07	3.75	5.72e−08	5.80
80	4.57e−08	4.02	7.08e−10	4.99	2.86e−08	3.92	1.28e−09	5.48
160	2.83e−09	4.01	2.22e−11	5.00	1.83e−09	3.97	3.33e−11	5.27
320	1.76e−10	4.01	6.91e−13	5.00	1.15e−10	3.99	9.57e−13	5.12
640	1.10e−11	4.00	2.49e−14	4.80	7.17e−12	4.01	3.55e−14	4.75

4. Deferred correction. We will now investigate a main application of the theoretical results from section 2, namely the solution of initial value problems via iterated deferred correction. It is well-known that the error of an Euler scheme can be expanded as a series in powers of h [21]. Moreover, we know from [14], that the linear barycentric rational interpolants reproduce polynomials of degree up to at least d, and Theorem 2.1 shows that antiderivatives can be approximated stably to relatively high order with equispaced nodes. These results let us hope, and later confirm, that deferred correction schemes can be established using rational interpolants with equispaced nodes, polynomial reproduction guarantees that the error expansion of the initial approximation is conserved, and integration to high order accuracy allows to carry out several correction sweeps so as to obtain solutions of initial value problems with high precision.

Iterated correction schemes originate with papers by Zadunaisky [44] and later [45] building upon earlier ideas (see the historical notes in [17, 35]), and were originally aimed at estimating errors in the solution of boundary value problems using so-called "pseudo-systems" or neighboring problems. These techniques were then used by Pereyra [35, 36] to iteratively increase the order of approximation by one per correction sweep, usually until reaching the precision of the collocation solution [16, 17, 40, 43]. In these initial publications, the nodes used were essentially equispaced, and it is illustrated in [20] that the order does not necessarily increase by one per correction sweep with arbitrary points. Around the year 2000 appeared the paper [13], which reiterated investigations on deferred correction schemes by introducing a very successful combination of Gauss–Legendre quadrature and the Picard integral formulation of the initial value problem, so as to avoid numerical differentiation. Possible choices of other quadrature schemes are described in [31]. Recently, modified iterated correction schemes were presented [7, 8], which use Runge–Kutta methods and guarantee an increase in the order per sweep by more than one, under precisely investigated conditions on the data. With additional modifications to the scheme, such increase

ETNA
Kent State University
http://etna.math.kent.edu

can also be obtained with nonequispaced nodes [41], namely with a fixed point iteration before each correction sweep. Moreover, deferred correction methods have been successfully implemented on parallel computers; see, e.g., [6, 33].

In order to comply with the usual notation in time dependent problems, we change the variable x from the previous sections to the time variable t on the standard interval $[0, T]$, and denote by x exclusively a space variable.

Consider an initial value problem for a function $u : [0, T] \to \mathbb{C}^N$,

$$u'(t) = f(t, u(t)), \qquad u(0) = u_0 \in \mathbb{C}^N \text{ given}, \tag{4.1}$$

and a numerically computed approximate solution $\widetilde{u} \approx u$. In the *deferred correction method* for iteratively improving the accuracy of this numerical solution, an equation for the error $e = u - \widetilde{u}$ is solved repeatedly by a low-order integrator. The approximate solution to this error equation is then added to $\widetilde{u}$ to give an improved approximation to u. Dutt, Greengard & Rokhlin [13] greatly enhanced the stability of deferred correction for initial value problems by two modifications, which we will briefly outline.

First, numerical differentiation can be avoided by reformulating the problem (4.1) as a Picard integral

$$u(t) = u(0) + \int_0^t f(\tau, u(\tau))\, \mathrm{d}\tau, \tag{4.2}$$

or equivalently,

$$\widetilde{u}(t) + e(t) = u(0) + \int_0^t f(\tau, \widetilde{u}(\tau) + e(\tau))\, \mathrm{d}\tau. \tag{4.3}$$

Using (4.2) for defining the residual

$$r(t) = u(0) + \int_0^t f(\tau, \widetilde{u}(\tau))\, \mathrm{d}\tau - \widetilde{u}(t), \tag{4.4}$$

we immediately find from (4.3)

$$e(t) = r(t) + \int_0^t f(\tau, \widetilde{u}(\tau) + e(\tau)) - f(\tau, \widetilde{u}(\tau))\, \mathrm{d}\tau = r(t) + \int_0^t g(\tau, e(\tau))\, \mathrm{d}\tau, \tag{4.5}$$

with $g(\tau, e(\tau)) := f(\tau, \widetilde{u}(\tau) + e(\tau)) - f(\tau, \widetilde{u}(\tau))$, which is a Picard-type formulation for the error $e(t)$.

Second, after this reformulation the deferred correction procedure proposed in [13] involves integration of polynomial interpolants of degree n, where n can be quite large to achieve sufficient accuracy. To prevent this polynomial interpolation from being unstable, it was advocated in [13] to interpolate the integrand in (4.2) at Gauss–Legendre points or Chebyshev points $t_0 < t_1 < \cdots < t_n$ in the interval $[0, T]$. Assume that $\widetilde{u}$ is given as the interpolation polynomial of the values $\widetilde{u}_j$ at the points t_j, then the residual r in (4.4) can be approximated stably and very accurately via numerical integration; see also the discussion in [31]. Because such an interpolation process will achieve spectral accuracy in time, the resulting method is called *spectral deferred correction* (SDC) [13]. Formula (4.5) now allows for the construction of time stepping procedures for the error e, starting from $e_0 = e(0) = 0$, e.g., by the explicit Euler method

$$e_j = e_{j-1} + (r_j - r_{j-1}) + (t_j - t_{j-1}) \cdot g(t_{j-1}, e_{j-1}), \quad j = 1, \ldots, n,$$

ETNA
Kent State University
http://etna.math.kent.edu

or the implicit Euler method

$$e_j = e_{j-1} + (r_j - r_{j-1}) + (t_j - t_{j-1}) \cdot g(t_j, e_j), \quad j = 1, \ldots, n.$$

Finally, the approximation $\widetilde{u}$ is updated via $\widetilde{u}_{\text{new}} = \widetilde{u} + e$, which concludes one deferred correction sweep.

In the following we propose a modification of the spectral deferred correction approach which we will refer to as *rational deferred correction* (RDC). In this new scheme (4.1) is still reformulated as the Picard integral (4.2), but then the integrand is interpolated at *equispaced* nodes $t_j = jT/n$ ($j = 0, 1, \ldots, n$). This is possible because, as we have seen in the previous sections, the barycentric rational interpolants proposed in [14] can attain stable high-order accuracy even for equispaced nodes, provided that the blending parameter d is chosen adequately. Through numerical examples later in this section, we will demonstrate that almost the same accuracy can be achieved with RDC from equispaced data than with SDC from clustered data.

There are many situations where it is more natural or even unavoidable to work with approximations to the solution $\widetilde{u}$ at equispaced time points. For example, if $\widetilde{u}$ is computed via a multistep method then equispaced time points t_i are typically preferable, or the right-hand side f of (4.1) may incorporate measurements that are only known at certain time points. Moreover, in implicit time stepping methods where the action of the inverse of a shifted Jacobian is required, constant time steps avoid recomputations or refactorizations of large matrices; see also our example in Section 4.5. A need for equal time steps also arises in some parallelized deferred correction schemes, such as the revisionist integral deferred correction method in [6].

We merely claim here and proof later in much more detail that, as is the case with similar deferred correction schemes, the iteratively corrected solution converges to the collocation solution of (4.2), and that the approximation order is increased by one per iteration if the time stepping method is explicit. The highest attainable order is $d + 2$ after $d + 1$ sweeps, which is that of the rational collocation solution of the same problem. Experiments revealed that further improvement in the RDC and the SDC solutions can be achieved by conformally mapping the nodes to a grid that is more clustered toward the ends of the time interval for RDC and more evenly spaced for SDC. We refrain from giving further details on this observation since this matter goes beyond the scope of the present paper. Notice that the accuracy achieved by RDC and by SDC is very similar.

4.1. Stability and accuracy regions. The stability and the accuracy of a numerical method for the solution of an initial value problem are usually studied via its performance when applied on the so-called Dahlquist test equation on $[0, 1]$ (see, e.g., [21]),

$$u'(t) = \lambda u(t), \qquad u(0) = 1.$$

If $\widetilde{u}$ is the solution obtained with the method under consideration, then the region $\{\lambda \in \mathbb{C} : |\widetilde{u}(1)| \leq 1\}$ is called the *stability region*. A method is *A-stable* if its stability region includes the whole negative half plane, and $A(\alpha)$*-stable* if it includes only a sector with opening angle 2α in the same half plane. The set $\{\lambda \in \mathbb{C} : |u(1) - \widetilde{u}(1)| < \varepsilon\}$ is called the *accuracy region* for a prescribed target accuracy ε.

We plotted the stability and accuracy regions for several values of the parameters defining RDC with equal time steps, and focus on those which we also use for the numerical examples later in this section. Figure 4.1 shows the stability regions for RDC with explicit Euler, for $n = 14$ and $n = 21$, respectively, for various values of d. The number of correction sweeps is equal to $d + 1$. Each plot contains the accuracy region for $\varepsilon = 10^{-8}$ as well. The

ETNA
Kent State University
http://etna.math.kent.edu

sizes of the RDC stability and accuracy regions with moderate values of d, say 14 or 15, are similar to those of the regions of SDC presented in [13]. Moreover, we notice that the accuracy region becomes larger as d increases, which is natural, since the underlying rational interpolants reach higher theoretical order of convergence with larger values of d. On the other hand, the stability regions decrease with increasing d, and this was also to be expected since, as discussed in Section 1, the conditioning of the rational interpolants deteriorates with increasing d.

In Figure 4.2 we present stability and accuracy regions with RDC based on implicit Euler. This time, the stability regions are located outside the enclosed domains. We executed 8 correction sweeps in each setting. This value was determined experimentally since there is for the moment no clear indication of an optimal number of sweeps with the implicit method. We observed that as soon as the number of sweeps exceeds 10, the stability regions become smaller and do not extend to $-\infty$ on the real line, and in our numerical experiments no further improvement in the solution was achieved with such a high number of corrections. The pictures indicate that RDC with implicit Euler has very favorable stability properties, even with large n. The methods are obviously not A-stable, but they are $A(\alpha)$-stable with a large angle α. A comparison of the upper and lower rows of pictures in this figure shows that the accuracy regions increase in size as n increases and d remains fixed; notice also that the target accuracy was increased from $\varepsilon = 10^{-7}$ to $\varepsilon = 10^{-9}$. We can thus expect RDC to be stable also for stiff initial value problems, and to attain relatively high accuracy with mildly stiff problems.

Let us now turn to the numerical experiments with RDC applied to some selected test problems. Clearly, such experiments cannot be exhaustive, but with each example we try to illustrate another aspect of RDC. All reported errors of a numerical solution $\widetilde{u}$ available at time points t_j are measured in uniform norm both in space and time compared to a highly accurate (or exact) reference solution $u(t_j)$, i.e.,

$$\text{error}(\widetilde{u}) = \frac{\max_j \|u(t_j) - \widetilde{u}(t_j)\|_\infty}{\max_j \|u(t_j)\|_\infty}.$$

4.2. Analytic example: blow-up equation. This first example is rather artificial, but it admits an analytic solution and gives insight into how the asymptotic convergence theory presented in Section 2, Theorem 2.1, relates to the convergence of the RDC method. Consider the following blow-up equation for $t \in [0, 1]$,

$$u'(t) = \frac{u(t)^2}{s}, \quad u(0) = 1, \quad s > 1.$$

The exact solution $u(t) = s/(s-t)$ is analytic in an open neighborhood of the interval $[0, 1]$, and has a simple pole at $t = s$. We have chosen $s = 1.25$. Starting from an initial explicit Euler run with points $t_j = j/n$ ($j = 0, 1, \ldots, n$), we execute RDC sweeps until the norm of the correction has fallen below 10^{-14} and, as a result, the approximations stagnate. We repeat this experiment for $n = 5, 10, \ldots, 100$, and take $d = \text{round}(Cn)$ with $C = 0.2$. In Figure 4.3 (left) we show the stagnation level of RDC depending on the number of points n. We clearly observe an initial geometric convergence, and indeed the rate of convergence $R = 0.717$ corresponds to the level line going through the singularity s; see Figure 4.3 (right). For comparison, we also show in Figure 4.3 (left) the interpolation error of the exact solution with the same barycentric rational interpolants. For larger values of n we observe that the accuracy of RDC (and the underlying interpolation scheme) suffers as soon as the growing Lebesgue constant (here indicated by the increasing line) is of the same order as the interpolation error. This is in line with our findings in the previous sections: one has to be cautious with choosing large values for d.

ETNA
Kent State University
http://etna.math.kent.edu

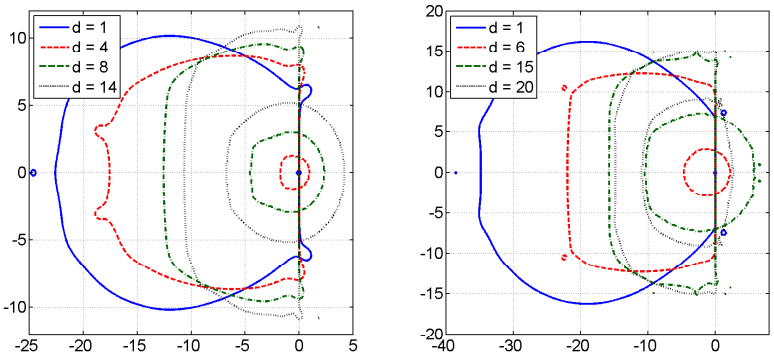

FIGURE 4.1. *Stability regions (outer) and accuracy regions (inner) with target accuracy* $\varepsilon = 10^{-8}$ *for RDC with explicit Euler with* $d+1$ *sweeps and* $n = 14$ *(left) and* $n = 21$ *(right).*

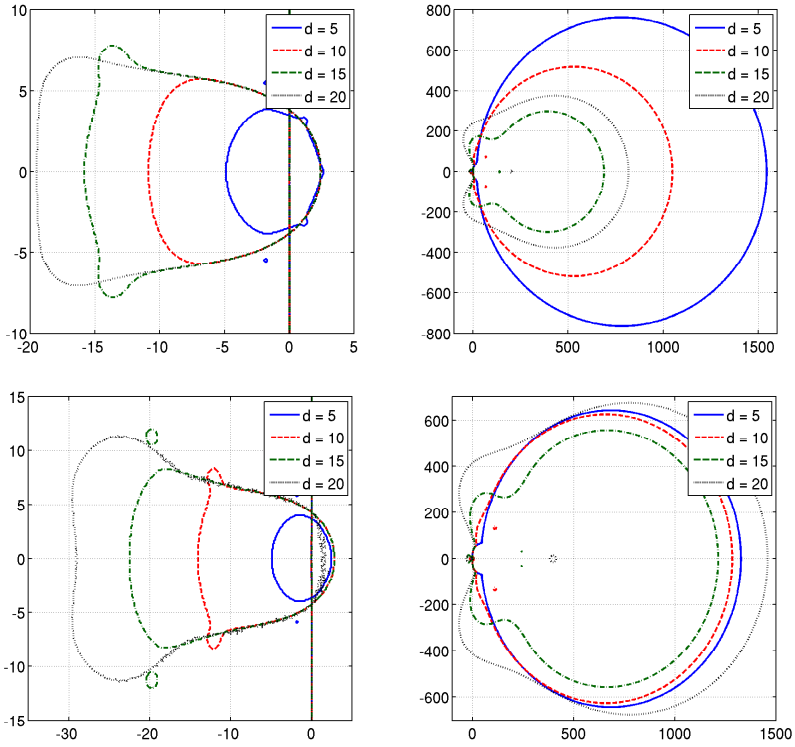

FIGURE 4.2. *Accuracy and stability regions for RDC with implicit Euler, 8 sweeps, and target accuracy* $\varepsilon = 10^{-7}$ *and* $n = 20$ *(top), and* $\varepsilon = 10^{-9}$ *and* $n = 40$ *(bottom). The pictures on the left are obtained from zooming into the pictures on the right to better visualize the accuracy regions.*

4.3. Nonstiff example: Brusselator. We consider the Brusselator [21] problem for $t \in [0, 12]$,

$$\begin{aligned} u_1'(t) &= 1 + u_1(t)^2 u_2(t) - 4u_1(t), & u_1(0) &= 0, \\ u_2'(t) &= 3u_1(t) - u_1(t)^2 u_2(t), & u_2(0) &= 1. \end{aligned}$$

ETNA
Kent State University
http://etna.math.kent.edu

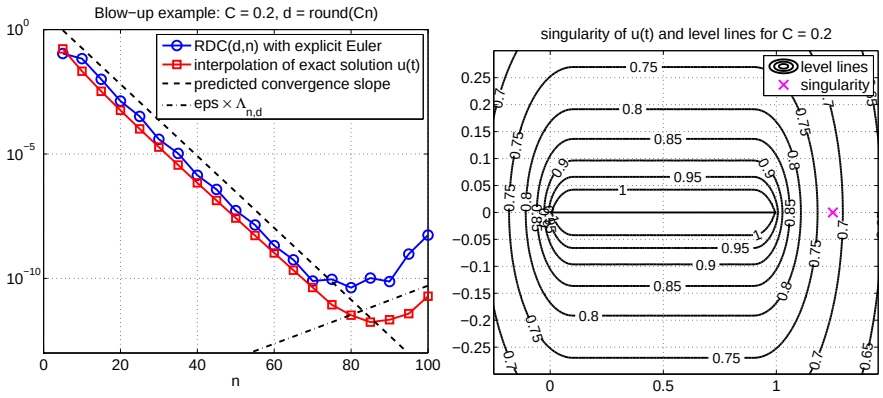

FIGURE 4.3. *Convergence of RDC with* $n = 5, 10, \ldots, 100$ *and* $d = \mathrm{round}(Cn)$, $C = 0.2$ *for the blow-up example (left). The convergence slope* $R = 0.717$ *can be read off from the level lines shown on the right, and this prediction is in good agreement with the observed convergence as long as the Lebesgue constant* $\Lambda_{n,d}$, *also indicated, has not grown above the error level.*

With this example our aim is to compute a solution of high-order accuracy on a time grid with equispaced points $t_j = 12j/n_{\mathrm{total}}$ $(j = 0, 1, \ldots, n_{\mathrm{total}})$, where $n_{\mathrm{total}} = 720$. To this end we partition the time grid in slices with $n + 1$ points, where $n = 15, 20, 40, 60, 80$, and run s sweeps of RDC driven by explicit Euler on each of the n_{total}/n time slices, while keeping the blending parameter $d = 15$ fixed. In Figure 4.4 (left) we show the attained accuracy as a function of the number of sweeps. Note that a number of 0 sweeps corresponds to a plain run of the explicit Euler method, without doing any correction sweeps. We observe that the convergence slopes for all choices of n are essentially the same, but a considerably smaller stagnation level can be achieved when n is increased. This indicates that the interpolation process extracts "more information" from a larger number of equispaced data points. The attainable accuracy will be limited by the condition number of the interpolation scheme, the Lebesgue constant $\Lambda_{n,d}$. We have computed the Lebesgue constant $\Lambda_{80,15} \approx 8.1 \times 10^3$, and indicate the level $\varepsilon_{\mathrm{M}} \times \Lambda_{80,15}$ with a horizontal dashed line in Figure 4.4 (left). As expected, the stagnation level of RDC cannot be below that line. When d is further increased, one will observe an even higher stagnation level and instabilities due to the exponential growth of $\Lambda_{n,d}$ in d.

In Figure 4.4 (right) we show the convergence of SDC for the same problem. Note that SDC requires explicit Euler time stepping at Chebyshev nodes, and in order to make sure that the cost per sweep is exactly the same as with RDC, we have again partitioned the time interval $[0, 12]$ in n_{total}/n time slices of equal length, then choosing $n + 1$ Chebyshev points on each of the time slices. We evaluate the resulting Chebyshev interpolation polynomial at exactly the same $n + 1$ equispaced time points we had chosen for checking the error of RDC.

4.4. Stiff example: Van der Pol equation. We now consider the Van der Pol equation [21] for $t \in [0, 10]$,

$$\begin{aligned} u_1'(t) &= u_2(t), & u_1(0) &= 2, \\ u_2'(t) &= 10(1 - u_1(t)^2)u_2(t) - u_1(t), & u_2(0) &= 0. \end{aligned}$$

The purpose of this example is to demonstrate the behavior of RDC at a mildly stiff equation, and to demonstrate the dependence of the stagnation level of the scheme on the blending

ETNA
Kent State University
http://etna.math.kent.edu

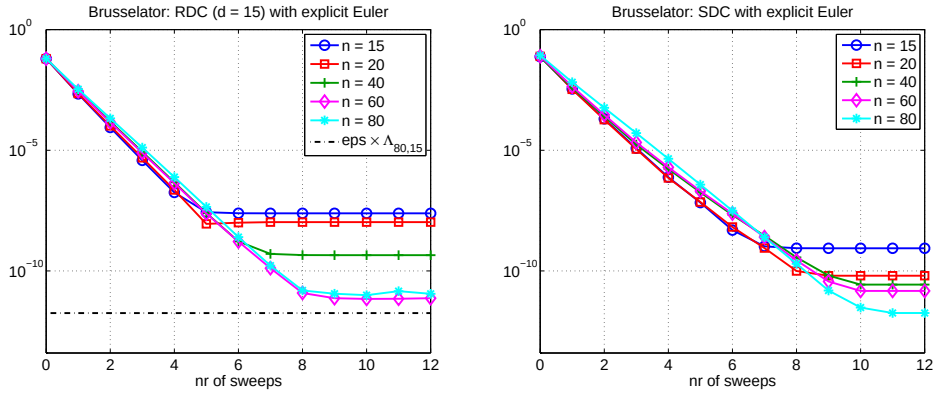

FIGURE 4.4. *Convergence (relative errors) of RDC and SDC for the Brusselator example with* $n_{\text{total}} = 720$. *In this experiment we vary the number* n, *the number of points per time slice, and show the error reduction with each sweep. For RDC the blending parameter* $d = 15$ *is fixed. In the left picture we also show the level of the Lebesgue constant* $\Lambda_{n,d}$ *for* $n = 80$ *and* $d = 15$.

parameter d. Our aim is to compute a solution of high-order accuracy on a time grid with equispaced points $t_j = 10j/n_{\text{total}}$ ($j = 0, 1, \ldots, n_{\text{total}}$), where $n_{\text{total}} = 1800$. To this end we partition this time mesh into 45 intervals, and compute a local rational interpolant on each of them with $n = 40$ and varying $d = 5, 10, 15, 20$. The RDC iteration is driven by implicit Euler, and the involved nonlinear equations have been solved by Newton's method with an error tolerance of 10^{-14} implemented in the Matlab routine `nsoli`; see [27]. The error reduction of RDC per sweep is shown in Figure 4.5 (left). For comparison, we also show the error reduction of SDC, and we note that the stagnation level of SDC is lower. This, however, is not surprising as SDC is evaluating the right-hand side of the differential equation at Chebyshev points, whereas RDC uses evaluations at equispaced time points exclusively.

In Figure 4.5 (right) we show for each of the 45 time slices and sweeps the decay of the norms of updates $\|e\|_\infty$ defined in (4.5), and the decay of the residuals $\|r\|_\infty$ defined in (4.4) (the plot is for $d = 15$). We observe that the error reduction per sweep is quite fast on all time slices except a few of them which are nearby a complex singularity of the exact solution $u(t)$. In this case, a local refinement of the time mesh would reduce the stagnation level of the RDC iteration, but we have not implemented this as we are focusing on uniformly equispaced data in this paper.

4.5. Implicit-explicit time stepping: Burgers' equation. A feature of deferred correction methods is the ease with which high-order semi-implicit methods can be constructed from simple low-order methods; see, e.g., [5, 32]. In our last example we consider a semilinear initial value problem of the form

$$Mu'(t) = Ku(t) + g(t, u(t)), \quad u(0) = u_0, \tag{4.6}$$

where $M, K \in \mathbb{R}^{N\times N}$ are possibly large matrices, and g is a nonlinear term that can be treated by an explicit integrator. A simple method for integrating such problems on a time grid $t_0, t_1, \ldots, t_n$ is the implicit-explicit Euler combination

$$Mu_{j+1} = Mu_j + h_j K u_{j+1} + h_j g(t_j, u_j), \quad h_j = t_{j+1} - t_j.$$

This recursion can be reformulated as

$$u_{j+1} = (M - h_j K)^{-1}(Mu_j + h_j g(t_j, u_j)),$$

ETNA
Kent State University
http://etna.math.kent.edu

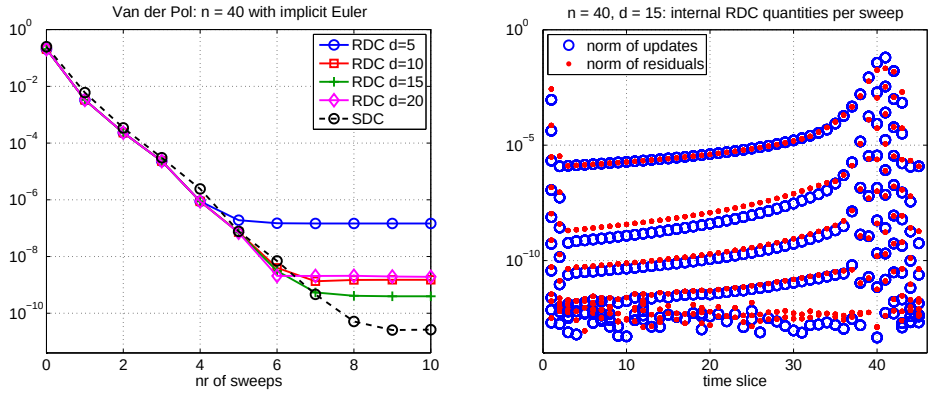

FIGURE 4.5. *Left: convergence (relative errors) of RDC and SDC for the Van der Pol equation. In this experiment we vary the blending parameter $d = 5, 10, 15, 20$ for RDC. The number $n = 40$ is fixed, so that there is a constant number of Newton solves per sweep and time slice. Right: the norms of updates and residuals in RDC on each of the 45 time slices. In this example all norms are monotonically decreasing, so that the top-most markers correspond to the first sweep, the ones below to the second sweep, and so on.*

which involves the solution of a linear system per time step. Now a computational advantage of running a deferred correction iteration with constant time steps $h_j = h$ becomes apparent: if the involved linear systems are solved with a direct solver, then only a single LU factorization of the constant matrix $(M - hK)$ needs to be computed and stored. Similar savings are possible with higher-order time stepping methods within deferred correction iterations. For an overview of such high-order methods we refer to [5], and the references therein.

A popular test problem for implicit-explicit time stepping is Burgers' equation for a function $u(t, x)$ defined on $[0, 0.5] \times [-1, 1]$ (see, e.g., [1, 2, 5, 32])

$$\partial_t u = \nu\, \partial_{xx} u - u\, \partial_x u, \qquad u(t, -1) = u(t, 1) = 0, \quad u(0, x) = -\sin(\pi x).$$

Following [2] closely, we have chosen a rather small viscosity $\nu = 1/(100\pi)$ and discretized the problem in space by collocation at $N = 512$ Chebyshev points $x_j \in [-1, 1]$. The space-discretized problem is of the form (4.6), with M being the identity matrix and K a collocation matrix for the operator $\nu\partial_{xx}$. The time integration is performed for $t \in [0, 0.5]$ via deferred correction.

In Figure 4.6 we show the error of SDC and RDC when the total number of time steps $n_{\text{total}} = 40, 80, \ldots, 2560$ is varied. We have grouped $n = 40$ time steps into one time slice and then applied a varying number of SDC or RDC sweeps on each slice. The blending parameter for RDC was always chosen as $d = 15$. We observe that, as long as stagnation does not occur, RDC achieves a slightly higher accuracy than SDC. A partial explanation is given by the fact that the maximal distance between n Chebyshev points is by a factor $\approx \pi/2$ larger than the distance between equispaced points on the same time interval (for $n \to \infty$ this factor is exact). It is therefore possible that a low-order time stepping method within RDC delivers higher accuracy than the same method used within SDC. In order to support this explanation we have run SDC with an increased number of $\text{round}(n\pi/2)$ nodes per time slice, and obtained errors closer to those of RDC. The improved accuracy of RDC comes in addition to the computational savings that are possible by reusing the same LU factorization of $(M - hK)$ for each constant time step h, which should be particularly attractive for large-scale problems.

ETNA
Kent State University
http://etna.math.kent.edu

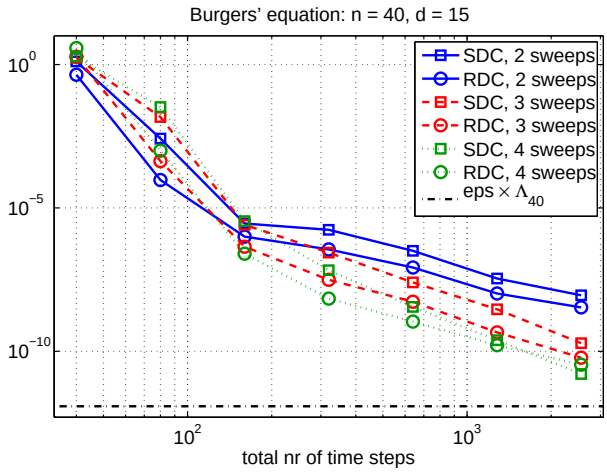

FIGURE 4.6. *Convergence comparison (relative errors) of an implicit-explicit SDC and RDC method for the Burgers' equation with a varying number of time steps on the interval* $[0, 0.5]$. *Each time slice has* $n = 40$ *nodes, and for RDC the blending parameter was chosen as* $d = 15$.

5. Summary. In this paper we have analyzed a numerical quadrature scheme based on barycentric rational interpolation, with an emphasis on situations where data is available at equispaced nodes. In the first part, devoted to the analysis of this quadrature scheme, we have presented two theorems that complement each other. Theorem 2.1 characterizes the asymptotic convergence speed of the scheme in the case where the integrand is analytic in the neighborhood of the integration interval. The expected convergence can be read off from certain level lines in the complex plane, and this theorem also allows for an improved choice of the blending parameter d, as we have demonstrated in Section 3. Theorem 2.2, on the other hand, covers the case where the integrand is not necessarily analytic and possesses a finite number of continuous derivatives. This theorem allowed us to apply extrapolation to the quadrature scheme. We compared the performance of the rational integration schemes with splines and Newton–Gregory rules. The results are similar, the linear rational interpolants are however easier to handle in numerous applications. An important application of quadrature is the solution of initial value problems. In Section 4 we have shown how the barycentric rational interpolation scheme can be used to construct a robust and efficient deferred correction algorithm using equispaced points in time, even when the number of points is quite large (say, 40–80). We have called this method rational deferred correction (RDC). On several numerical examples RDC performed comparably to spectral deferred correction (SDC), although the stagnation level of RDC may be higher than with SDC. This stagnation effect is caused by a faster growing condition number, on which bounds are available so that this effect is in principle controllable. RDC depends on the choice of the blending parameter d which comes from the rational interpolation scheme on which RDC is based. This may lead to some difficulty if no additional information about the functions involved in the initial value problem is available, but we found from numerous experiments that $d = 15$ is a good default choice.

Several questions remain open, in particular with RDC, and we plan to address some of them in the future. First of all, it should be investigated why RDC sometimes converges faster than SDC, before stagnation sets in. A partial explanation was given for the Burgers' equation, but this is far from a rigorous analysis. We would also like to understand better how the error of the approximate solution reduces with each sweep of RDC until stagnation occurs at the level of the rational collocation solution. Extensions to the presented scheme with eq-

ETNA
Kent State University
http://etna.math.kent.edu

uispaced nodes could possibly include the use of Runge–Kutta integrators in the expectation that the order increases by the order of that integrator per sweep, and the implementation of RDC on parallel computers.

REFERENCES

[1] U. M. ASCHER, S. J. RUUTH, AND B. T. R. WETTON, *Implicit-explicit methods for time-dependent partial differential equations*, SIAM J. Numer. Anal., 32 (1995), pp. 797–823.
[2] C. BASDEVANT, M. DEVILLE, P. HALDENWANG, J. M. LACROIX, J. OUAZZANI, R. PEYRET, P. ORLANDI, AND A. T. PATERA, *Spectral and finite difference solutions of the Burgers equation*, Comput. & Fluids, 14 (1986), pp. 23–41.
[3] 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.
[4] L. BOS, S. DE MARCHI, K. HORMANN, AND G. KLEIN, *On the Lebesgue constant of barycentric rational interpolation at equidistant nodes*, Numer. Math., 121 (2012), pp. 461–471.
[5] A. BOURLIOUX, A. T. LAYTON, AND M. L. MINION, *High-order multi-implicit spectral deferred correction methods for problems of reactive flow*, J. Comput. Phys., 189 (2003), pp. 651–675.
[6] A. CHRISTLIEB, C. B. MACDONALD, AND B. W. ONG, *Parallel high-order integrators*, SIAM J. Sci. Comput., 32 (2010), pp. 818–835.
[7] A. CHRISTLIEB, B. ONG, AND J.-M. QIU, *Comments on high-order integrators embedded within integral deferred correction methods*, Commun. Appl. Math. Comput. Sci., 4 (2009), pp. 27–56.
[8] ———, *Integral deferred correction methods constructed with high order Runge-Kutta integrators*, Math. Comp., 79 (2010), pp. 761–783.
[9] C. W. CLENSHAW AND A. R. CURTIS, *A method for numerical integration on an automatic computer*, Numer. Math., 2 (1960), pp. 197–205.
[10] P. DAVIS, *On a problem in the theory of mechanical quadratures*, Pacific. J. Math., 5 (1955), pp. 669–674.
[11] P. J. DAVIS AND P. RABINOWITZ, *Methods of Numerical Integration*, 2nd ed., Academic Press, Orlando, 1984.
[12] C. DE BOOR, *A Practical Guide to Splines*, Springer, New York, 1978.
[13] A. DUTT, L. GREENGARD, AND V. ROKHLIN, *Spectral deferred correction methods for ordinary differential equations*, BIT, 40 (2000), pp. 241–266.
[14] M. S. FLOATER AND K. HORMANN, *Barycentric rational interpolation with no poles and high rates of approximation*, Numer. Math., 107 (2007), pp. 315–331.
[15] B. FORNBERG, *A Practical Guide to Pseudospectral Methods*, Cambridge University Press, Cambridge, 1996.
[16] R. FRANK AND C. W. UEBERHUBER, *Iterated defect correction for the efficient solution of stiff systems of ordinary differential equations*, Nordisk Tidskr. Informationsbehandling (BIT), 17 (1977), pp. 146–159.
[17] ———, *Iterated defect correction for differential equations. I. Theoretical results*, Computing, 20 (1978), pp. 207–228.
[18] W. GAUTSCHI, *Orthogonal Polynomials: Computation and Approximation*, Oxford University Press, New York, 2004.
[19] S. GÜTTEL AND G. KLEIN, *Convergence of linear barycentric rational interpolation for analytic functions*, SIAM J. Numer. Anal., 50 (2012), pp. 2560–2580.
[20] E. HAIRER, *On the order of iterated defect correction*, Numer. Math., 29 (1977/78), pp. 409–424.
[21] E. HAIRER, S. P. NØRSETT, AND G. WANNER, *Solving Ordinary Differential Equations. I. Nonstiff Problems*, Springer, Berlin, 1987.
[22] N. J. HIGHAM, *The numerical stability of barycentric Lagrange interpolation*, IMA J. Numer. Anal., 24 (2004), pp. 547–556.
[23] F. B. HILDEBRAND, *Introduction to Numerical Analysis*, McGraw-Hill, New York, 1956.
[24] D. HUYBRECHS, *Stable high-order quadrature rules with equidistant points*, J. Comput. Appl. Math., 231 (2009), pp. 933–947.
[25] ———, *On the Fourier extension of nonperiodic functions*, SIAM J. Numer. Anal., 47 (2010), pp. 4326–4355.
[26] B. A. IBRAHIMOGLU AND A. CUYT, *Sharp bounds for Lebesgue constants of barycentric rational interpolation*, Tech. Report, Computational Mathematics, University of Antwerp, 2013.
[27] C. T. KELLEY, *Solving Nonlinear Equations with Newton's Method*, SIAM, Philadelphia, 2003.
[28] G. KLEIN, *Applications of Linear Barycentric Rational Interpolation*, Ph.D. Thesis No. 1762, Faculty of Science, University of Fribourg, 2012.
[29] G. KLEIN AND J.-P. BERRUT, *Linear barycentric rational quadrature*, BIT, 52 (2012), pp. 407–424.
[30] A. R. KROMMER AND C. W. UEBERHUBER, *Computational Integration*, SIAM, Philadelphia, 1998.
[31] A. T. LAYTON AND M. L. MINION, *Implications of the choice of quadrature nodes for Picard integral deferred corrections methods for ordinary differential equations*, BIT, 45 (2005), pp. 341–373.

ETNA
Kent State University
http://etna.math.kent.edu

[32] M. L. MINION, *Semi-implicit spectral deferred correction methods for ordinary differential equations*, Commun. Math. Sci., 1 (2003), pp. 471–500.
[33] ———, *A hybrid parareal spectral deferred corrections method*, Commun. Appl. Math. Comput. Sci., 5 (2010), pp. 265–301.
[34] J. OUSPENSKY, *Sur les valeurs asymptotiques des coefficients de Cotes*, Bull. Amer. Math. Soc., 31 (1925), pp. 145–156.
[35] V. PEREYRA, *On improving an approximate solution of a functional equation by deferred corrections*, Numer. Math., 8 (1966), pp. 376–391.
[36] ———, *Iterated deferred corrections for nonlinear boundary value problems*, Numer. Math., 11 (1968), pp. 111–125.
[37] R. B. PLATTE, L. N. TREFETHEN, AND A. B. J. KUIJLAARS, *Impossibility of fast stable approximation of analytic functions from equispaced samples*, SIAM Rev., 53 (2011), pp. 308–318.
[38] G. PÓLYA, *Über die Konvergenz von Quadraturverfahren*, Math. Z., 37 (1933), pp. 264–286.
[39] M. J. D. POWELL, *Approximation Theory and Methods*, Cambridge University Press, Cambridge, 1981.
[40] H. J. STETTER, *The defect correction principle and discretization methods*, Numer. Math., 29 (1977/78), pp. 425–443.
[41] T. TANG, H. XIE, AND X. YIN, *High-order convergence of spectral deferred correction methods on general quadrature nodes*, J. Sci. Comput., 56 (2013), pp. 1–13.
[42] L. N. TREFETHEN, *Is Gauss quadrature better than Clenshaw-Curtis?*, SIAM Rev., 50 (2008), pp. 67–87.
[43] C. W. UEBERHUBER, *Implementation of defect correction methods for stiff differential equations*, Computing, 23 (1979), pp. 205–232.
[44] P. E. ZADUNAISKY, *A method for the estimation of errors propagated in the numerical solution of a system of ordinary differential equations*, in Proc. Int. Astron. Union Symp. no. 25, The Theory of Orbits in the Solar System and in Stellar Systems, G. Contopoulos, ed., Academic Press, London, 1966.
[45] P. E. ZADUNAISKY, *On the estimation of errors propagated in the numerical integration of ordinary differential equations*, Numer. Math., 27 (1976/77), pp. 21–39.