Max-Planck-Institut für Mathematik in den Naturwissenschaften Leipzig

Computation of Best L^∞ Exponential Sums for $1/x$ by Remez' Algorithm

by

Wolfgang Hackbusch

Preprint no.: 80 2017

Computation of Best L^∞ Exponential Sums for $1/x$ by Remez' Algorithm

Wolfgang Hackbusch

Abstract

The approximation of the function $1/x$ by exponential sums has several interesting applications. It is well known that best approximations with respect to the maximum norm exist. Moreover, the error estimates exhibit exponential decay as the number of terms increases. Here we focus on the computation of the best approximations. In principle, the problem can be solved by the Remez algorithm, however, because of the very sensitive behaviour of the problem the standard approach fails for a larger number of terms. The remedy described in the paper is the use of other independent variables of the exponential sum. We discuss the approximation error of the computed exponential sums up to 63 terms and hint to a webpage containing the corresponding coefficients.

1 Introduction

At the first sight, the problem considered in this paper has an obvious solution. The best approximation of functions as $1/x$ by exponential sums

$$E_k(x) = \sum_{\nu=1}^{k} a_\nu \exp(-b_\nu x) \qquad (x \in \mathbb{R}) \tag{1.1}$$

with respect to the maximum norm is well studied (cf. Braess [3]). Even rather precise error estimates are known (cf. Braess–Hackbusch [4]). The approximation problem can be solved by the Remez algorithm which leads to a system of nonlinear equations. Since there is exactly one solution and the involved functions are analytic, the Newton method should be a perfect solver ensuring quadratic convergence.

This may be true for small values of k, but for larger k the exponential decay of the approximation error e_k has a negative effect. The Remez algorithm requires exponential sums E_k which interpolate the function $1/x$ at $2k$ points (in this case we say that E_k is *feasible*). Since e_k is rather small, tiny perturbations of E_k can lead to a difference $1/x - E_k$ with less than $2k$ zeros. Hence, the subset of feasible exponential sums is a rather small set. Even if an initial value belongs to this set, small corrections may cause divergence. Another difficulty is the fact that the error e_k may be much smaller that $\sqrt{\text{eps}}$ (eps: machine precision). This fact prevents quadratic convergence.

Because of these difficulties[1] the best L^∞ approximation is often replaced by least squares approximations (cf. Evans et al. [7], Kammler [13]). Concerning ill-posedness of this least squares problem we refer to [17]. Furthermore, §2.3 will show that there are applications requiring the maximum norm, while the L^2 norm is insufficient.

In §3 we recall the facts about the best approximation by exponential sums. To apply the Remez algorithm we introduce a crucial 'trick' in §4.3. We use other variables than the coefficients in (1.1). As a consequence, the represented exponential sums are always feasible. The drawback is an increased computational cost, since we apply an outer Newton method involving several inner Newton iterations per outer iteration step. The best approximation refers to an underlying interval $[1, R]$. In §4.6 we describe how to proceed from one R to another R' so that always good initial values are available. Another 'continuation' is the step from k to $k+1$.

[1] In Kammler [12] the author writes: In general, the problem of finding a best uniform approximation ... is quite difficult, and consequently various schemes ... have sometimes been used to produce "good" if not best approximations.

The tables in this contribution show the approximation error in dependence of the parameters k and R. They are part of a large collection of best approximations as described in §2.4. The obtained approximation errors are compared with the theoretical bounds.

2 Definition, Properties, and Applications

2.1 Definition

Exponential sums are expressions of the form (1.1) with $2r$ parameters[2] a_ν and b_ν. Here we discuss the approximation of the function $1/x$ by exponential sums in a positive interval $[a,b] \subset (0,\infty)$ with respect to the maximum norm $\|\cdot\|_{\infty,[a,b]}$. Let E_k^* be the best approximation. To be precise, its coefficients a_ν, b_ν in (1.1) depend on the underlying interval $[a,b]$, i.e., $a_\nu = a_{\nu,[a,b]}$ and $b = b_{\nu,[a,b]}$. The minimal error is denoted by

$$\varepsilon_{[a,b]}(k) = \min_{E_k} \left\| \frac{1}{\cdot} - E_k \right\|_{\infty,[a,b]} = \left\| \frac{1}{\cdot} - E_k^* \right\|_{\infty,[a,b]} .$$

2.2 Properties

Using the map of $x \in [a,b]$ to $x' := x/a \in [1, b/a]$, one finds that the best approximations in $[a,b]$ and $[1,b/a]$ and their approximation errors are related by

$$a_{\nu,[a,b]} := \frac{a_{\nu,[1,R]}}{a}, \quad b_{\nu,[a,b]} := \frac{b_{\nu,[1,R]}}{a}, \quad \varepsilon_{[a,b]}(k) = \frac{\varepsilon_{[1,R]}(k)}{a},$$

where we introduce $R := b/a$. Hence, it is sufficient to study the approximation problem for different values of R and k. It turns out that also the choice

$$R = \infty$$

makes sense. In this case, $[1,R]$ should be read as $[1,R)$.

The optimal exponential sums allow a stable evaluation since the coefficients a_ν, b_ν are positive.

2.3 Applications

The typical property $\exp(x+y) = \exp(x)\exp(y)$ of the exponential function shows that $E_k(\sum_\mu x_\mu) = \sum_{\nu=1}^k a_\nu \prod_\mu \exp(-b_\nu x_\mu)$ is a sum of separable terms.

A direct application is used in the second-order Møller–Plesset theory (MP2), where the energy is a sum of terms $\frac{\dots}{\varepsilon_a+\varepsilon_b-\varepsilon_i-\varepsilon_j}$ (cf. Ayala–Scuseria [1] and [16]). The effect of using E_k^* is demonstrated by a simpler example. Consider $S := \sum_{i,j=1}^N \frac{v_i v_j}{\varepsilon_i+\varepsilon_j}$ with large N, where $\varepsilon_i+\varepsilon_j \in [a,b]$. The computation of S costs $\mathcal{O}(N^2)$ operations. Replacing $\frac{1}{\varepsilon_i+\varepsilon_j}$ by $E_k^* = E_{k,[a,b]}^*$ yields

$$S \approx \sum_{i,j=1}^N v_i v_j E_k(\varepsilon_i+\varepsilon_j) = \sum_{\nu=1}^k a_\nu \left(\sum_{i=1}^N v_i \exp(-b_\nu \varepsilon_i) \right) \left(\sum_{j=1}^N v_j \exp(-b_\nu \varepsilon_j) \right),$$

which can be evaluated by $\mathcal{O}(Nk)$ operations.

The inverse matrix A^{-1} can be approximated by $E_k^*(A)$ with $E_k^* = E_{k,[a,b]}^*$, provided that the spectrum of A is contained in $[a,b]$. If $A = TDT^{-1}$ (D diagonal), the estimate with respect to the spectral matrix is

$$\left\| A^{-1} - E_k^*(A) \right\|_2 \le \|T\|_2 \left\| T^{-1} \right\|_2 \left\| \tfrac{1}{\cdot} - E_k^* \right\|_{\infty,[a,b]} .$$

The maximum norm $\left\| \frac{1}{\cdot} - E_k^* \right\|_{\infty,[a,b]}$ cannot be replaced by a weaker norm as the L^2 norm. The replacement of the inverse A^{-1} by $E_k^*(A)$ is helpful for Kronecker matrices. Let A_i be positive definite matrices. Then

[2] Here we only consider real, in particular positive, coefficients.

the Kronecker matrix $\mathbf{A} := A_1 \otimes I \otimes I + I \otimes A_2 \otimes I + I \otimes I \otimes A_3$ has the approximate inverse

$$\mathbf{B}_k := E_k^*(\mathbf{A}) = \sum_{\nu=1}^{k} a_\nu \exp(-b_\nu A_1) \otimes \exp(-b_\nu A_2) \otimes \exp(-b_\nu A_3)$$

(cf. [10, Prop. 9.34]) with the spectral norm $\left\|\mathbf{A}^{-1} - \mathbf{B}_k\right\|_2 \le \left\|\frac{1}{\cdot} - E_k^*\right\|_{\infty,[a,b]}$, where $[a,b]$ contains the spectrum of $\mathbf{A}$.

2.4 Available Data

This paper contains tables of the approximation errors $\varepsilon_{[1,R]}(k)$ for $1 \le k \le 56$ and various R. The data for $1 \le k \le 7$ are shown in Table 2.1. The parameter R takes all values $R = n \cdot 10^m$ ($n \in \mathbb{N}$, $m \in \mathbb{N}_0 := \mathbb{N} \cup \{0\}$), subject to the following bounds. The largest value is $R = R_k^*$, where R_k^* is explained in §3.3. Larger values of R are uninteresting since $E^*_{k,[1,R]} = E^*_{k,[1,R_k^*]}$ for $R \ge R_k^*$. Besides $R \ge 2$ the lower bound is implicitly given by $\varepsilon_{[1,R]}(k) \approx 10^{-16}$. Such values are available for all $1 \le k \le 56$ (see web page `https://www.mis.mpg.de/scicomp/EXP_SUM/1_x/`). The tables 2.2–2.8 given here contain only results for R being powers of 10. The additional table 2.9 is restricted to $R = \infty$ and $57 \le k \le 63$.

The web page `https://www.mis.mpg.de/scicomp/EXP_SUM/1_x/` contains a complete table (see file 'tabelle'). For each pair (k, R) contained in the table there is a file[3] with the coefficients[4] a_ν, b_ν ($1 \le \nu \le k$). The file contains additional data which are important as input for the computer program. In particular, the points ξ_i ($1 \le i \le 2k$) with $1/\xi_i = E_k^*(\xi_i)$ are given (cf. §4.3).

Approximations of other functions are mentioned in [8].

3 Existence and Error Estimates

3.1 Existence and Equioscillation

The approximation problem is closely related to the interpolation by exponential sums. Because of the nonlinear nature, an interpolating exponential may fail to exist (example: $f(x) = x$ cannot by interpolated by some E_1 at $x = \pm 1$). For sufficient conditions we refer to Braess [3, §VI.3].

Since $f(x) = 1/x$ is *completely monotone* for $x > 0$, i.e., $(-1)^n f^{(n)}(x) > 0$ for all $n \in \mathbb{N}_0$, the unique existence of the best approximation E_k^* is guaranteed (cf. [3, §VII]). Moreover, E_k^* satisfies the *equioscillation* property which is well known for polynomials (cf. de la Vallée Poussin [6, page 85], Süli–Mayers [15, Theorem 8.3]): The error $e_k := \frac{1}{\cdot} - E_k^*$ is extreme at $2k+1$ points μ_i with

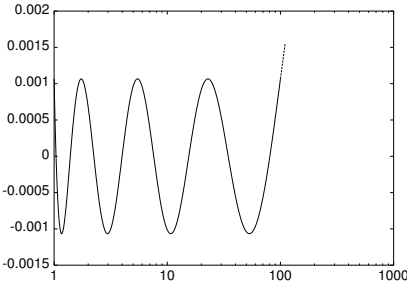

Figure 3.1: Error $e_{4,[1,100]}$

$$1 = \mu_0 < \mu_1 < \ldots < \mu_{2k} \le R$$

and

$$e_k(\mu_i) = (-1)^i \, \varepsilon_{[1,R]}(k). \tag{3.1}$$

Each interval $(\mu_i - 1, \mu_i)$ contains exactly one zero of e_k, i.e., there are $\xi_i \in (\mu_i - 1, \mu_i)$ with $e_k(\xi_i) = 0$. The latter equation states the interpolation property

$$\frac{1}{\xi_i} = E_k^*(\xi_i) \quad \text{for } 1 \le i \le 2k \qquad \text{(cf. Figure 3.1)}.$$

[3] For instance, the file $k12.3E5$ contains the coefficients corresponding to $k = 12$ and $R = 3_{10}5$. The file with the largest value of R (k fixed) yields the best approximation for $[1, \infty)$.

[4] In the file, a_ν and b_ν are called omega[ν] and alpha[ν].

3.2 Error Estimate for Finite R

The precise error estimates involve elliptic integrals (cf. [4, §2]). Estimating these special functions by exponentials yields

$$\varepsilon_{[1,R]}(k) \le 16 \exp\left(-\frac{\pi^2 k}{\log(8R)}\right)$$

(cf. Braess–Hackbusch [4, (2.9)], [5, (29)]). A comparison with the numbers from Tables 2.1–2.9 show that this bound can be improved by $O\left(\frac{1}{\sqrt{R}} \exp(-\frac{\pi^2 k}{\log(6R)})\right)$. All errors computed in §2.4 satisfy

$$0.0134 < \frac{\varepsilon_{[1,R]}(k)}{\frac{1}{\sqrt{R}} \exp\left(-\frac{\pi^2 k}{\log(6R)}\right)} < 12.18.$$

The small ratios occur for small R.

3.3 Case of $R = \infty$

Consider the error $e_{k,[1,R]} := \frac{1}{\cdot} - E^*_{k,[1,R]}$ for varying R. For sufficiently small R, the last extremum $e_{k,[1,R]}(\mu_{2k}) = \varepsilon_{[1,R]}(k)$ is located at $\mu_{2k} = R$ with $\frac{\mathrm{d}}{\mathrm{d}x} e_{k,[1,R]}(\mu_{2k}) > 0$. As R increases, $\frac{\mathrm{d}}{\mathrm{d}x} e_{k,[1,R]}(\mu_{2k})$ decreases until, for a certain $R = R^*_k$, $\frac{\mathrm{d}}{\mathrm{d}x} e_{k,[1,R]}(\mu_{2k}) = 0$ holds. Then $|e_{k,[1,R^*_k]}(x)| \le \varepsilon_{[1,R^*_k]}(k)$ also holds for $R \ge R^*_k$ (cf. Figure 3.2). Hence

$$E^*_{k,[1,R^*_k]} = E^*_{k,[1,R]} = E^*_{k,[1,\infty)} \qquad \text{for all } R^*_k \le R < \infty.$$

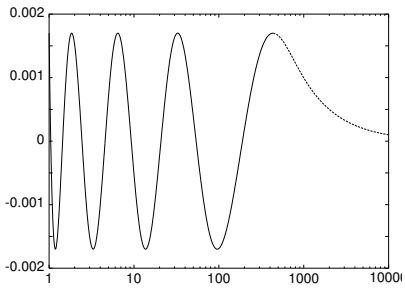

Figure 3.2: Error $e_{4,[1,436.1]} = e_{4,[1,\infty)}$

For $R = \infty$ the error can be proved to be bounded by

$$\varepsilon_{[1,\infty)}(k) \le 16 \exp\left(-\pi\sqrt{k}\right)$$

(cf. [5, (30)]). The values $e_{k,[1,R^*_k]}(k) = e_{k,[1,\infty)}(k)$ shown in the tables behave better. The function $\log(2+k)\,\exp(-\pi\sqrt{2k}\,)$ describing the asymptotic decay is proposed by D. Braess. The approximation errors for $1 \le k \le 63$ satisfy the two-sided inequality

$$6.6\,\log(2+k)\,\exp(-\pi\sqrt{2k}\,) \le \varepsilon_{[1,\infty)}(k) \le 6.9\,\log(2+k)\,\exp(-\pi\sqrt{2k}\,).$$

4 Computation

4.1 Machine Precision

The coefficients of E^*_k given in `https://www.mis.mpg.de/scicomp/EXP_SUM/1_x/` are computed with extended precision (eps = 1_{10}-19). This fact allows us to reach approximations with $\varepsilon_{[1,R]} \approx 10^{-16}$ and better. Of course, using the corresponding coefficients in double precision, the floating-point errors of the function evaluation may be larger than $\varepsilon_{[1,R]}$.

4.2 Remez Algorithm

The equioscillation property yields the necessary $2k$ equations for the $2k$ coefficients. Condition (3.1) implies that

$$e_k(\mu_{i-1}) + e_k(\mu_i) = 0 \qquad \text{for } i = 1, \ldots, 2k. \tag{4.1}$$

This is the basis of the Remez algorithm[5] (cf. Remez [14]). To apply the iterative algorithm, one has to start from a function E_k as in (1.1) such that the difference $e_k = \frac{1}{\cdot} - E_k$ has $2k$ simple zeros ξ_i $(1 \le i \le 2k)$. We call such an exponential sum a *feasible* E_k.

[5] Usually the Remez algorithm is applied to the L^∞ approximation by polynomials or rational functions (cf. Werner [18]). Approximation by nonlinear families is studied by Barrar–Loeb [2]).

Next one has to determine the extrema $e_k(\mu_i)$. The arguments μ_i belong to (ξ_i, ξ_{i+1}) for $1 \le i \le 2k-1$. The extremum $e_k(\mu_0)$ is taken at the boundary: $\mu_0 = 1$. If $R \le R_k^*$, the extremum $e_k(\mu_{2k})$ belongs to $\mu_{2k} = R$. If $R > R_k^*$, μ_{2k} lies in (ξ_{2k}, ∞).

Given the values μ_i, (4.1) represents $2k$ nonlinear equations for the $2k$ coefficients. This yields a new $E_k^{\rm new}$. If $E_k^{\rm new}$ is feasible, one can determine the new μ_i etc.

4.3 Choice of Variables

It seems to be obvious to use the parameters a_ν, b_ν in (1.1) for the computation, i.e., $E_k = E_k(\cdot; \mathbf{a})$, where

$$\mathbf{a} := (a_1, b_1, \ldots, a_k, b_k)\,.$$

Inserting E_k into (4.1) yields $2k$ nonlinear equations $\phi_i(\mathbf{a}) = 0$ $(1 \le i \le 2k)$. In theory, Newton's method should converge to the desired solution E_k^*. However, that does not work in practice. This is why the computation of best L^∞ exponential sums is regarded as hardly solvable.

The cause of the difficulty is explained as follows. Due to the good approximation, $\varepsilon_{[1,R]}(k)$ is small. Assume that there is an iterate E_k with $\|\frac{1}{\cdot} - E_k\|_\infty \le 10^{-13}$. Then a perturbation of E_k by a tiny shift of, e.g., $2 \cdot 10^{-13}$ may yield an $E_k^{\rm new}$ which does not interpolate $\frac{1}{\cdot}$. Hence there are no zeros ξ_i (or at least not $2k$ of them), $E_k^{\rm new}$ is infeasible, and the algorithm cannot be continued.

Instead we introduce other variables describing E_k. Since the interpolating exponential sum is unique, we describe a feasible exponential sum E_k by the interpolation points ξ_i. Using the vector

$$\boldsymbol{\xi} := (\xi_1, \ldots, \xi_{2k}) \quad \text{with } 1 < \xi_1 < \xi_2 < \ldots < \xi_{2k} < R,$$

we can express a feasible E_k by

$$E_k(\cdot) = \hat{E}_k(\cdot; \boldsymbol{\xi}) \quad \text{satisfying } E_k(\xi_i) = 1/\xi_i \text{ for } 1 \le i \le 2k. \tag{4.2}$$

Given a feasible $E_k(\cdot; \mathbf{a})$, we can determine the zeros ξ_i of $\frac{1}{\cdot} - E_k(\cdot; \mathbf{a}) = 0$ and obtain $\boldsymbol{\xi} = \boldsymbol{\xi}(\mathbf{a})$, i.e., $E_k(\cdot; \mathbf{a}) = \hat{E}_k(\cdot; \boldsymbol{\xi}(\mathbf{a}))$. On the other hand, given $\boldsymbol{\xi}$, the interpolating $E_k(\cdot; \mathbf{a})$ can be determined by Newton's method. This yields the inverse mapping $\mathbf{a} = \mathbf{a}(\boldsymbol{\xi})$; i.e., $\hat{E}_k(\cdot; \boldsymbol{\xi}) = E_k(\cdot; \mathbf{a}(\boldsymbol{\xi}))$.

4.4 Computation of $\hat{E}_k(\cdot; \boldsymbol{\xi})$ – Inner Iteration

For the practical computation, one uses the pair $(\boldsymbol{\xi}, \mathbf{a}(\boldsymbol{\xi}))$. If one wants to determine $\hat{E}_k(\cdot; \boldsymbol{\xi}')$ for $\boldsymbol{\xi}'$ close to $\boldsymbol{\xi}$, one has to solve $E_k(\xi_i'; \mathbf{a}') = 1/\xi_i'$ with respect to $\mathbf{a}'$. Here we exploit the fact that the Newton method has a starting value $\mathbf{a} = \mathbf{a}(\boldsymbol{\xi})$ very close to $\mathbf{a}' = \mathbf{a}(\boldsymbol{\xi}')$.

The interpolation is harder to compute if $\xi_i \approx \xi_{i+1}$ are very close. However, the zeros ξ_i of the best approximation E_k^* are well separated (cf. Remark 4.2).

Note that the use of the parameters $\boldsymbol{\xi}$ ensures that $\hat{E}_k(\cdot; \boldsymbol{\xi})$ is a feasible exponential sum. The drawback of this approach is a larger computational work. Instead of the evaluation of $E_k(\cdot; \mathbf{a})$ the Newton method requires several evaluations of exponential sums and their derivatives (which are again exponential sums).

In the following, iterates $\boldsymbol{\xi}^0, \boldsymbol{\xi}^1, \ldots$ of an *outer* iteration will appear. Each $\boldsymbol{\xi}^\nu$ requires an *inner* iteration by the Newton method described above. A standard value of the number of inner iteration steps used in §2.4 is 4.

4.5 Computation of the Extrema

In principle, given $\boldsymbol{\xi}$ and $\mathbf{a} = \mathbf{a}(\boldsymbol{\xi})$, the location μ_i of the extrema can be determined by Newton's method. For the results given in the tables the extrema are computed differently. In each interval $[\xi_{i-1}, \xi_i]$ the values at $x_\nu = \xi_{i-1} + \nu\,(\xi_i - \xi_{i-1})\,/N$ are evaluated for $\nu = 1, \ldots, N-1$. Let ν^* be the index with $x_{\nu^*} = \operatorname{argmax}_\nu |e_k(x_\nu)|$ and let P the cubic interpolation at $x_{\nu-1}, x_\nu, x_{\nu+1}$. The location of the maximum of P is taken as μ_i.

Remark 4.1 *Since the computational work increases with N, only the final data should be performed with a large N. Intermediate steps can be done with small N.*

The values in the tables are based on $N = 1000$.

4.6 Newton's Method for (4.1) and Continuation Principle

4.6.1 The Outer Iteration

The Newton method for solving (4.1) is formulated with respect to the parameters $\boldsymbol{\xi}$. Equation (4.1) becomes $0 = e_k(\mu_{i-1}) + e_k(\mu_i) = \frac{1}{\mu_{i-1}} - \hat{E}_k(\mu_{i-1};\boldsymbol{\xi}) + \frac{1}{\mu_i} - \hat{E}_k(\mu_i;\boldsymbol{\xi}) =: \phi_i(\boldsymbol{\xi})$. The evaluation of $\phi_i(\boldsymbol{\xi})$ is explained in §4.3. The derivatives $\frac{\partial}{\partial \xi_j}\phi_i(\boldsymbol{\xi})$ required by the Newton method are replaced by divided differences.

We call this iteration the *outer* iteration since each evaluation of $\hat{E}_k(\cdot;\boldsymbol{\xi})$ for a new $\boldsymbol{\xi}$ requires inner iteration steps according to §4.4.

If the Newton iteration is not successful, it is replaced by the damped version. The damping parameter should be chosen such that

- the approximation error $\|e_k\|_\infty$ is decreasing,
- the zeros ξ_i of e_k should not come too close.

The reason for the last advice is the fact that the ξ_i values of the best approximation $E^*_{k,[1,R]}(\cdot,\boldsymbol{\xi})$ are well separated. More precisely, the following observation holds.

Remark 4.2 *In a first approximation, the zeros behave as* $\xi_i \approx R^{(i/(2k))^c}$ *with* c *between* 1.2 *and* 1.3.

If it happens that ξ_i and ξ_{i+1} are too close, one should change the positions, e.g., by $\xi_{i+1} := (\xi_i + \xi_{i+2})/2$.

The continuation method explained in §4.6.3 ensures that the Newton method can be started with initial iterates close to the solution.

4.6.2 Start

The process is started by computations for $k = 1$. Consider, e.g., $R = 2$. A simple choice of interpolation points is $\xi_1 = 4/3$ and $\xi_1 = 5/3$. The computation of $\mathbf{a}(\boldsymbol{\xi})$ is harmless. For all initial values $0.1 \le a_1, b_1 \le 5$, the Newton method converges (to $a_1 = 1.831\ldots$, $b_1 = 0.6694\ldots$). After 6 steps of the outer Newton method for solving (4.1), one obtains the best approximation up to machine precision.

Given any best approximation $E^*_{k,[1,R]} = E^*_{k,[1,R]}(\cdot,\boldsymbol{\xi}^*)$ together with the interpolation points $\boldsymbol{\xi}^*$, one can obtain $E^*_{k,[1,R']}$ for other R' according to §4.6.3.

4.6.3 Change of R

In the following, k is fixed. Assume that a best approximation $E^*_{k,[1,R]}$ is already available for some R.

The first task is to compute $E^*_{k,[1,R']}$ for a *larger* $R' > R$. The approximation error of $E^*_{k,[1,R]}$ is $\varepsilon_{[1,R]}(k)$. Take $E_k := E^*_{k,[1,R]}$ as initial value for the outer iteration of §4.6.1 on $[1,R']$. If $R' \le R^*_k$, the maximum in the last subinterval $[\xi_{2k}, R']$ is taken at $x = R'$. If R' is not close enough to R, the maximum $e_k(R')$ may be much larger than $\varepsilon_{[1,R]}(k)$ and the Newton method may fail. In that case one has apply the continuation method: compute $E^*_{k,[1,R_m]}$ for a sequence $R = R_0 < R_1 < \ldots < R_M = R'$, where each R_{m+1} is sufficiently close to R_m.

If it happens that the last extremum is at $\mu_{2k} < R'$, one has obtained the best approximation in $[1,\infty)$ and $\mu_{2k} = R^*_k$ holds (cf. §3.3).

In the case of a *smaller* $R' < R$ the same continuation method can be used. However, the new value R' should be larger than ξ_{2k}. Otherwise, the interval $[1,R']$ contains less than $2k$ values ξ_i of the vector $\boldsymbol{\xi}$ defining the initial value $E^*_{k,[1,R]} = E_k(\cdot,\boldsymbol{\xi})$ and the Newton iteration may fail.

For the intermediate computations with $R = R_0 < R_1 < \ldots < R_M = R'$ one may save computational work by Remark 4.1.

For k large and R small, the points ξ_i are rather close. In this case, the restriction $\xi_{2k} < R' < R$ for the new value R' is very restrictive. Here, another strategy can be applied. The affine[6] map $x \mapsto 1 + \frac{R'-1}{R-1}(x-1)$

[6] An improvement could be a mapping making use of Remark 4.2.

maps $[1, R]$ onto $[1, R']$. Applying this map to $\boldsymbol{\xi}$, one obtains a rough approximation of $\boldsymbol{\xi}'$ corresponding $E^*_{k,[1,R']} = E_k(\cdot, \boldsymbol{\xi}')$. Note that $\xi'_{2k} < R'$ holds. After defining $\boldsymbol{\xi}'$, the coefficients $\mathbf{a}' = \mathbf{a}(\boldsymbol{\xi}')$ have to be computed by the inner iteration of §4.4. If the factor $\frac{R'-1}{R-1}$ is chosen too small, this iteration may fail to converge.

If one follows these hints, the outer Newton iteration even works without damping.

We illustrate these hints by two examples. First we start from the best approximation $E^*_{k,[1,R]}$ for $k = 7$, $R = 1000$. This case is less sensitive, i.e., the Newton iteration behaves rather robust. For a larger $R' \in [R, 10 \cdot R]$ the outer Newton iteration converges. The choice of a smaller R' is restricted by $R' > \xi_{14} = 838$. Much smaller R' can be obtained by the second strategy using new $\boldsymbol{\xi}'$. Convergence is observed for amplification factors $\frac{R'-1}{R-1} \in [0.5, 2]$.

A more sensitive example is $E^*_{k,[1,R]}$ for $k = 50$, $R = 2_{10}8$ since $\varepsilon_{[1,2_{10}8]}(50) = 4.43_{10}$-14. An increase of R leads to convergence as long as $R' \le 3.5_{10}8 = \frac{7}{4}R$. A decrease of R is restricted by $R > \xi_{100} = 1.967_{10}8$. The second strategy is more helpful. Using the factor $\frac{R'-1}{R-1} \in [0.92, 1.1]$, the inner iterations converge and enable the computation for $R' \in [1.84_{10}8,\ 2.2_{10}8]$.

4.6.4 Increasing k

The step $k \mapsto k + 1$ is more delicate since two additional parameters must be created. They must be such that the exponential sum is feasible and has two additional zeros. The difficulty increases with the size of k. As an example we consider the largest k appearing in Table 2.8: $k = 56$. Since $\varepsilon_{[1,R]}(56)$ increases with increasing R, the largest possible R is the best candidate for starting values: $R = R^*_{56} = 7.5_{10}12$. The coefficients of $E^*_{56,[1,R]}$ are

ν	1	2	3	...	56
a_ν	1.65_{10}-12	1.08_{10}-11	4.44_{10}-10	...	5.7
b_ν	5.16_{10}-13	5.68_{10}-12	3.01_{10}-11	...	14.4

The ansatz for $k = 57$ is $E_{57}(x) := a_0 \exp(-b_0 x) + E^*_{56,[1,R]}$, i.e., we do not change the above data. The additional term $a_0 \exp(-b_0 x)$ should be so small that the equioscillation structure is not perturbed. Since $\varepsilon_{[1,R]}(56) \approx 1_{10}$-13, we need $a_0 \le 1_{10}$-13. A look at a_ν, b_ν for $\nu = 1, 2$ shows that they decay by a factor of about 10. This leads to the proposal

$$a_0 = 1_{10}\text{-}13, \qquad b_0 = 5_{10}\text{-}14.$$

$\frac{1}{\cdot} - E^*_{56,[1,R]}$ has the following zeros ξ_i:

i	1	2	3	...	109	110	111	112
ξ_i	1.003	1.026	1.074	...	$159_{10}11$	$3.57_{10}11$	$9.27_{10}11$	$3.26_{10}12$

One observes that the ratios are increasing. Therefore we introduce the two additional zeros

$$\xi_{113} = 1_{10}13, \qquad \xi_{114} = 1_{10}14.$$

Since the new value of R must be larger than ξ_{114}, we choose $R = 1_{10}15$.

The first step are many (damped) Newton iterations for computing the coefficients $\mathbf{a}(\boldsymbol{\xi})$ and E_{57} for the new $\boldsymbol{\xi}$. It turns out that the last maximum is taken at $\mu_{115} = 1.9_{10}14$. Accordingly we choose $R = 1.9_{10}14$.

The first outer iteration requires a damping of the Newton correction by 1/8. The values ξ_i as well as μ_{115} have decreased and $R = \mu_{115} = 5.1_{10}13$ can be chosen. About 4 damped outer Newton steps are needed, before the Newton method works without damping.

The choice of the initial values may be a matter of trial and error.

4.7 Modifications

4.7.1 Wavelet Applications

In [9] an application in quantum mechanics is mentioned which involves exponential sums for $1/\sqrt{x}$. The technique explained in [9] works equally well for the function $1/x$.

Whenever scalar products $\left\langle \frac{1}{\cdot}, \varphi \right\rangle$ with wavelet functions φ appear, we can exploit the following property. Depending on the *vanishing moment* M of φ, $\langle p, \varphi \rangle = 0$ holds for all polynomials p of degree $\leq M-1$. This leads to the following approximation problem. Let $F^*_{k,M}$ be the best L^∞ approximation of $\frac{1}{\cdot}$ in the interval $[1, R]$ within the family of functions

$$\{F_{k,M} = E_k + p : E_k \text{ as in (1.1)}, \, p \text{ polynomial of degree } < M\}\,.$$

Obviously, the best approximation error improves with increasing M. As an example we show the errors for $k = 7$, $R = 10$ (degree -1 means $p = 0$):

polynomial degree	-1	0	1
approximation error	2.344_{10}-8	6.554_{10}-9	1.934_{10}-9

The best approximation $F^*_{k,M}$ can be computed analogously to usual exponential sums. Note that the exponential part E_k in $F^*_{k,M} = E_k + p$ is different from the best approximation E^*_k. After computing $F^*_{k,M} = E_k + p$, the polynomial part p can be omitted, since it is not needed for

$$\left\langle F^*_{k,M}, \varphi \right\rangle = \langle E_k, \varphi \rangle\,.$$

Therefore the computational work for $\langle E_k, \varphi \rangle$ is independent of the degree of p.

4.7.2 Weighted Norm

There may be reasons to prefer approximations with respect to a weighted norm

$$\|f\|_{[a,b],\omega} := \max\{|f(x)\omega(x)| : a \leq x \leq b\}$$

with $\omega > 0$. In principle, one can apply the Remez algorithm after replacing $e_k(\mu_{i-1}) + e_k(\mu_i) = 0$ in (4.1) by $e_k(\mu_{i-1})\omega(\mu_{i-1}) + e_k(\mu_i)\omega(\mu_i) = 0$.

The computation of a best approximation $E^*_{k,[a,b]}$ is simplified by the reduction to $E^*_{k,[1,b/a]}$ (cf. §2.2). In the case of a weight ω, the reduction to $[1, b/a]$ requires that ω is homogeneous:

$$\omega(ax) = \varphi(a)\omega(x).$$

Examples are all powers $\omega(x) = x^\gamma$ ($\gamma < 1$ if $b = \infty$). Then the best approximation $E^*_{k,[1,b/a]}$ in $[1, b/a]$ also determines the best approximation $E^*_{k,[a,b]}$ in $[a, b]$ by

$$E^*_{k,[a,b]}(x) := \frac{1}{a} E^*_{k,[1,b/a]}(x/a)$$

with the error

$$\left\| \tfrac{1}{\cdot} - E^*_{k,[a,b]} \right\|_{[a,b],\omega} = \frac{\varphi(a)}{a} \left\| \tfrac{1}{\cdot} - E^*_{k,[1,b/a]} \right\|_{[1,a/b],\omega}.$$

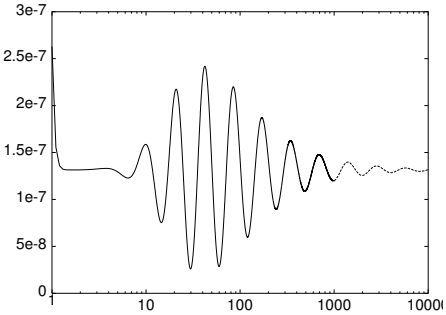

Figure 4.1: Approximation error of the exponential sum for $k = 45$ obtained by sinc quadrature.

4.7.3 Quadrature

Since $\frac{1}{x} = \int_0^\infty \exp(-xt)\mathrm{d}t$, any quadrature formula $Q(f) = \sum_{\nu=1}^k a_\nu f(b_\nu)$ applied to the function $f(\cdot) = \exp(-x\,\cdot)$ yields an exponential sum of the form (1.1).

A particular choice is the sinc quadrature. It requires an integral over $\mathbb{R}$. For instance, the substitution $t = \exp(y)$ yields an integral of the desired form: $\frac{1}{x} = \int_{-\infty}^\infty \exp(-x\exp(y))\exp(y)\mathrm{d}y$. The sinc quadrature is

defined and analysed in [11, §D.4]. The drawback is that the quadrature is not adapted to the fact that the interesting parameters x belong to the interval $[1, R]$. The L^∞ error estimate is of the form $\leq c \exp(-c'\sqrt{k})$ (cf. [4, (3.6)]). Figure 4.1 shows the error e_k for $k = 45$. The error bound is 2.63_{10}-7, while the best approximation – say in $[1, 10^6]$ – is 3.913_{10}-15. It is oscillating, but remains in the positive part, i.e., it is infeasible. Therefore it is *not* possible to use the quadrature result as starting value for the Remez algorithm.

References

[1] P. Y. Ayala and G. E. Scuseria. Linear scaling second-order Moller–Plesset theory in the atomic orbital basis for large molecular systems. *J. Chem. Phys.*, 110:3660, 1999.

[2] R. B. Barrar and H. L. Loeb. On the Remez algorithm for non-linear families. *Numer. Math.*, 15:382–391, 1970.

[3] D. Braess. *Nonlinear Approximation Theory.* Springer, Berlin, 1986.

[4] D. Braess and W. Hackbusch. Approximation of $1/x$ by exponential sums in $[1, \infty)$. *IMA J. Numer. Anal.*, 25:685–697, 2005.

[5] D. Braess and W. Hackbusch. On the efficient computation of high-dimensional integrals and the approximation by exponential sums. In Ronald A. DeVore and Angela Kunoth, editors, *Multiscale, Nonlinear and Adaptive Approximation*, pages 39–74. Springer, Berlin, 2009.

[6] C.-J. de la Vallée Poussin. *Leçons sur l'approximation des fonctions d'une variable réelle.* Gauthier-Villars, Paris, 1919.

[7] J. W. Evans, W. B. Gragg, and Randall J. LeVeque. On least squares exponential sum approximation with positive coefficients. *Math. Comp.*, 34:203–211, 1980.

[8] W. Hackbusch. Entwicklungen nach Exponentialsummen. Techn. Bericht 25, Max-Planck-Institut für Mathematik in den Naturwissenschaften, Leipzig, 2005.

[9] W. Hackbusch. Approximation of $1/|x-y|$ by exponentials for wavelet applications. *Computing*, 76:359–366, 2006.

[10] W. Hackbusch. *Tensor Spaces and Numerical Tensor Calculus*, volume 42 of *SSCM.* Springer, Berlin, 2012.

[11] W. Hackbusch. *Hierarchical Matrices – Algorithms and Analysis*, volume 49 of *SSCM.* Springer, Berlin, 2015.

[12] D. W. Kammler. Chebyshev approximation of completely monotonic functions by sums of exponentials. *SIAM J. Numer. Anal.*, 13:761–774, 1976.

[13] D. W. Kammler. Least squares approximation of completely monotoic functions by sums of exponentials. *SIAM J. Numer. Anal.*, 16:801–818, 1979.

[14] E. J. Remez. Sur un procédé convergent d'approximations successives pour déterminer les polynômes d'approximation. *Compt. Rend. Acad. Sc.*, 198:2063–2065, 1934.

[15] E. Süli and D. F. Mayers. *An Introduction to Numerical Analysis.* Cambridge University Press, Cambridge, 4th printing edition, 2008.

[16] A. Takatsuka, S. Ten-no, and W. Hackbusch. Minimax approximation for the decomposition of energy denominators in Laplace-transformed Møller-Plesset perturbation theories. *J. Chem. Phys.*, 129:044112, 2008.

[17] J. M. Varah. On fitting exponentials by nonlinear least squares. *SIAM J. Sci. Statist. Comput.*, 6:30–44, 1985.

[18] H. Werner. *Vorlesung über Approximationstheorie*, volume 14 of *Lect. Notes Math.* Springer, Berlin, 1966.

R	$k=1$	2	3	4	5	6	7
2E00	2.128E-02	2.080E-04	1.834E-06	1.542E-08	1.261E-10	1.012E-12	8.020E-15
3E00	4.358E-02	1.035E-03	2.223E-05	4.556E-07	9.088E-09	1.780E-10	3.444E-12
4E00	5.960E-02	2.191E-03	7.279E-05	2.311E-06	7.139E-08	2.167E-09	6.498E-11
5E00	7.075E-02	3.437E-03	1.500E-04	6.258E-06	2.543E-07	1.016E-08	4.007E-10
6E00	7.825E-02	4.659E-03	2.463E-04	1.246E-05	6.143E-07	2.976E-08	1.424E-09
7E00	8.288E-02	5.811E-03	3.553E-04	2.079E-05	1.185E-06	6.643E-08	3.677E-09
8E00	8.516E-02	6.878E-03	4.718E-04	3.098E-05	1.982E-06	1.246E-07	7.741E-09
9E00	8.556E-02	7.857E-03	5.924E-04	4.273E-05	3.004E-06	2.076E-07	1.417E-08
1E01		8.752E-03	7.145E-04	5.577E-05	4.243E-06	3.173E-07	2.344E-08
2E01		1.448E-02	1.819E-03	2.169E-04	2.521E-05	2.880E-06	3.252E-07
3E01		1.699E-02	2.627E-03	3.795E-04	5.336E-05	7.379E-06	1.008E-06
4E01		1.784E-02	3.215E-03	5.230E-04	8.266E-05	1.285E-05	1.973E-06
5E01			3.659E-03	6.469E-04	1.110E-04	1.872E-05	3.121E-06
6E01			4.001E-03	7.541E-04	1.377E-04	2.471E-05	4.382E-06
7E01			4.271E-03	8.474E-04	1.626E-04	3.066E-05	5.711E-06
8E01			4.485E-03	9.293E-04	1.858E-04	3.648E-05	7.077E-06
9E01			4.655E-03	1.002E-03	2.074E-04	4.213E-05	8.458E-06
1E02			4.789E-03	1.066E-03	2.274E-04	4.760E-05	9.841E-06
2E02				1.456E-03	3.707E-04	9.217E-05	2.261E-05
3E02				1.628E-03	4.554E-04	1.235E-04	3.297E-05
4E02				1.695E-03	5.117E-04	1.467E-04	4.141E-05
5E02					5.517E-04	1.649E-04	4.842E-05
6E02					5.811E-04	1.795E-04	5.438E-05
7E02					6.031E-04	1.915E-04	5.952E-05
8E02					6.193E-04	2.016E-04	6.401E-05
9E02					6.309E-04	2.102E-04	6.799E-05
1E03					6.385E-04	2.177E-04	7.153E-05
2E03						2.570E-04	9.365E-05
3E03							1.047E-04
4E03							1.110E-04
5E03							1.146E-04
6E03							1.162E-04
∞	8.556E-02	1.785E-02	5.052E-03	1.700E-03	6.428E-04	2.646E-04	1.163E-04
R_k^*	8.667	41.54	146.8	436.1	1154	2807	6373
	$k=1$	2	3	4	5	6	7

Table 2.1: $e_{k,[1,R]}(k)$ for $k = 1, \ldots, 7$ and various R. The errors corresponding to the empty places are those shown for $R = \infty$.

R	$k=8$	9	10	11	12	13	14
1E01	1.716E-09	1.248E-10	9.021E-12	6.492E-13	4.654E-14	3.326E-15	2.371E-16
1E02	2.016E-06	4.103E-07	8.303E-08	1.673E-08	3.357E-09	6.716E-10	1.340E-10
1E03	2.321E-05	7.468E-06	2.389E-06	7.605E-07	2.412E-07	7.623E-08	2.403E-08
1E04	5.271E-05	2.232E-05	9.296E-06	3.844E-06	1.582E-06	6.481E-07	2.648E-07
1E05				6.795E-06	3.379E-06	1.646E-06	7.973E-07
∞	5.392E-05	2.611E-05	1.312E-05	6.807E-06	3.630E-06	1.984E-06	1.108E-06
R_k^*	13749	28387	56502	1.089E+5	2.042E+6	3.737E+5	6.691E+5

Table 2.2: $e_{k,[1,R]}(k)$ for $k = 8, \ldots, 14$ and various R.

R	$k=15$	16	17	18	19	20	21
1E01	1.708E-17						
1E02	2.667E-11	5.298E-12	1.050E-12	2.079E-13	4.110E-14	8.114E-15	1.600E-15
1E03	7.555E-09	2.371E-09	7.426E-10	2.322E-10	7.251E-11	2.261E-11	7.044E-12
1E04	1.079E-07	4.388E-08	1.780E-08	7.213E-09	2.918E-09	1.179E-09	4.755E-10
1E05	3.847E-07	1.850E-07	8.877E-08	4.251E-08	2.032E-08	9.700E-09	4.624E-09
1E06	6.280E-07	3.445E-07	1.867E-07	1.007E-07	5.421E-08	2.911E-08	1.560E-08
1E07						4.679E-08	2.752E-08
∞	6.311E-07	3.659E-07	2.155E-07	1.289E-07	7.811E-08	4.794E-08	2.976E-08
R_k^*	1.175E+6	2.027E+6	3.440E+6	5.753E+6	9.491E+6	1.546E+7	2.491E+7

Table 2.3: $e_{k,[1,R]}(k)$ for $k = 15, \ldots, 21$ and various R.

R	$k=22$	23	24	25	26	27	28
1E02	3.153E-16	6.218E-17					
1E03	2.192E-12	6.813E-13	2.116E-13	6.566E-14	2.036E-14	6.309E-15	1.954E-15
1E04	1.916E-10	7.715E-11	3.103E-11	1.247E-11	5.009E-12	2.010E-12	8.061E-13
1E05	2.201E-09	1.047E-09	4.975E-10	2.362E-10	1.120E-10	5.310E-11	2.515E-11
1E06	8.351E-09	4.464E-09	2.384E-09	1.272E-09	6.777E-10	3.609E-10	1.920E-10
1E07	1.611E-08	9.404E-09	5.481E-09	3.190E-09	1.854E-09	1.076E-09	6.244E-10
1E08				4.802E-09	2.999E-09	1.866E-09	1.159E-09
∞	1.868E-08	1.185E-08	7.583E-09	4.898E-09	3.190E-09	2.094E-09	1.385E-09
R_k^*	3.969E+7	6.258E+7	9.776E+7	1.513E+8	2.325E+8	3.540E+8	5.353E+8

Table 2.4: $e_{k,[1,R]}(k)$ for $k = 22, \ldots, 28$ and various R.

R	$k=29$	30	31	32	33	34	35
1E03	6.048E-16	1.872E-16	6.218E-17				
1E04	3.231E-13	1.294E-13	5.181E-14	2.073E-14	8.292E-15	3.315E-15	1.325E-15
1E05	1.191E-11	5.633E-12	2.663E-12	1.259E-12	5.945E-13	2.807E-13	1.325E-13
1E06	1.021E-10	5.426E-11	2.882E-11	1.530E-11	8.114E-12	4.303E-12	2.281E-12
1E07	3.619E-10	2.096E-10	1.213E-10	7.018E-11	4.057E-11	2.344E-11	1.354E-11
1E08	7.188E-10	4.452E-10	2.755E-10	1.704E-10	1.053E-10	6.499E-11	4.011E-11
1E09		6.162E-10	4.053E-10	2.651E-10	1.730E-10	1.128E-10	7.343E-11
∞	9.227E-10	6.188E-10	4.177E-10	2.837E-10	1.938E-10	1.331E-10	9.194E-11
R_k^*	8.036E+8	1.198E+9	1.775E+9	2.614E+9	3.826E+9	5.569E+9	8.063E+9

Table 2.5: $e_{k,[1,R]}(k)$ for $k = 29, \ldots, 35$ and various R.

R	$k=36$	37	38	39	40	41	42
1E04	5.294E-16	2.117E-16	8.451E-17				
1E05	6.249E-14	2.947E-14	1.389E-14	6.546E-15	3.084E-15	1.453E-15	6.839E-16
1E06	1.208E-12	6.399E-13	3.388E-13	1.793E-13	9.484E-14	5.016E-14	2.652E-14
1E07	7.814E-12	4.509E-12	2.600E-12	1.499E-12	8.641E-13	4.978E-13	2.867E-13
1E08	2.474E-11	1.525E-11	9.396E-12	5.787E-12	3.563E-12	2.193E-12	1.349E-12
1E09	4.777E-11	3.105E-11	2.018E-11	1.310E-11	8.501E-12	5.515E-12	3.576E-12
1E10	6.365E-11	4.353E-11	2.962E-11	2.011E-11	1.364E-11	9.246E-12	6.262E-12
∞	6.382E-11	4.452E-11	3.121E-11	2.197E-11	1.554E-11	1.104E-11	7.869E-12
R_k^*	1.162E+10	1.665E+10	2.375E+10	3.374E+10	4.772E+10	6.719E+10	9.424E+10

Table 2.6: $e_{k,[1,R]}(k)$ for $k = 36, \ldots, 42$ and various R.

R	$k=43$	44	45	46	47	48	49
1E05	3.220E-16	1.516E-16					
1E06	1.402E-14	7.407E-15	3.913E-15	2.067E-15	1.091E-15	5.763E-16	3.042E-16
1E07	1.651E-13	9.503E-14	5.469E-14	3.146E-14	1.810E-14	1.041E-14	5.983E-15
1E08	8.296E-13	5.100E-13	3.135E-13	1.926E-13	1.183E-13	7.266E-14	4.461E-14
1E09	2.318E-12	1.502E-12	9.725E-13	6.297E-13	4.076E-13	2.637E-13	1.706E-13
1E10	4.238E-12	2.867E-12	1.939E-12	1.310E-12	8.852E-13	5.978E-13	4.036E-13
1E11	5.591E-12	3.938E-12	2.766E-12	1.940E-12	1.360E-12	9.524E-13	6.667E-13
∞	5.633E-12	4.047E-12	2.919E-12	2.113E-12	1.534E-12	1.118E-12	8.172E-13
R_k^*	1.316E+11	1.832E+11	2.540E+11	3.509E+11	4.833E+11	6.631E+11	9.074E+11

Table 2.7: $e_{k,[1,R]}(k)$ for $k = 43, \ldots, 49$ and various R.

R	$k=50$	51	52	53	54	55	56
1E06	1.606E-16						
1E07	3.439E-15	1.976E-15	1.136E-15	6.524E-16	3.749E-16	2.154E-16	1.239E-16
1E08	2.739E-14	1.681E-14	1.031E-14	6.327E-15	3.880E-15	2.379E-15	1.459E-15
1E09	1.103E-13	7.135E-14	4.612E-14	2.981E-14	1.926E-14	1.244E-14	8.038E-15
1E10	2.724E-13	1.838E-13	1.240E-13	8.362E-14	5.638E-14	3.800E-14	2.571E-14
1E11	4.664E-13	3.262E-13	2.281E-13	1.594E-13	1.113E-13	7.776E-14	5.429E-14
1E12	5.966E-13	4.321E-13	3.120E-13	2.251E-13	1.622E-13	1.168E-13	8.410E-14
∞	5.992E-13	4.407E-13	3.251E-13	2.405E-13	1.784E-13	1.327E-13	9.897E-14
R_k^*	1.238E+12	1.683E+12	2.281E+12	3.083E+12	4.155E+12	5.587E+12	7.491E+12

Table 2.8: $e_{k,[1,R]}(k)$ for $k = 50, \ldots, 56$ and various R.

R	$k=57$	58	59	60	61	62	63
∞	7.400E-14	5.547E-14	4.168E-14	3.139E-14	3.139E-14	1.793E-14	1.360E-14
R_k^*	1.002E+13	1.337E+13	1.779E+13	2.362E+13	3.064E+13	4.134E+13	5.453E+13

Table 2.9: $e_{k,[1,\infty)}(k)$ for $k = 57, \ldots, 63$.