Chapter 1
Basics of Numerical Analysis

1.1 Introduction

An ever increasing amount of computational work is being relegated to computers, and often we almost blindly assume that the obtained results are correct. At the same time, we wish to accelerate individual computation steps and improve their accuracy. Numerical computations should therefore be approached with a good measure of skepticism. Above all, we should try to understand the meaning of the results and the precision of operations between numerical data.

A prudent choice of appropriate algorithms is essential (see, for example, [1, 2]). In their implementation, we should be aware that the compiler may have its own "will" and has no "clue" about mathematical physics. In order to learn more about the essence of the computation and its natural limitations, we strive to simplify complex operations and restrict the tasks of functions to smaller, well-defined domains. It also makes sense to measure the execution time of programs (see Appendix J): large fluctuations in these well measurable quantities without modifications in running conditions typically point to a poorly designed program or a lack of understanding of the underlying problem.

1.1.1 Finite-Precision Arithmetic

The key models for computation with real numbers in finite precision are the *floating-point* and *fixed-point arithmetic*. A real number x in floating-point arithmetic with base β is represented by the approximation

$$\mathrm{fl}(x) = \pm\left(d_0 + d_1\beta^{-1} + d_2\beta^{-2} + \cdots + d_{p-1}\beta^{-(p-1)}\right) \cdot \beta^e \equiv \pm d_0.d_1 \ldots d_{p-1} \cdot \beta^e,$$

where $\{d_i\}_{i=0}^{p-1}$, $d_i \in \{0, 1, \ldots, \beta - 1\}$, is a set of p integers and the exponent e is within $[e_{\min}, e_{\max}]$. The expression $m = d_0.d_1 \ldots d_{p-1}$ is called the *significand*

S. Širca, M. Horvat, *Computational Methods for Physicists*, Graduate Texts in Physics,
DOI 10.1007/978-3-642-32478-9_1, © Springer-Verlag Berlin Heidelberg 2012

Table 1.1 The smallest and largest exponents and approximate values of some important numbers representable in single- and double-precision floating-point arithmetic in base two, according to the IEEE 754 standard. Only positive values are listed

Precision	Single ("`float`")	Double ("`double`")
$e_{\max}$	127	1023
$e_{\min} = 1 - e_{\max}$	-126	-1022
Smallest normal number	$\approx 1.18 \times 10^{-38}$	$\approx 2.23 \times 10^{-308}$
Largest normal number	$\approx 3.40 \times 10^{38}$	$\approx 1.80 \times 10^{308}$
Smallest representable number	$\approx 1.40 \times 10^{-45}$	$\approx 4.94 \times 10^{-324}$
Machine precision, ε_{M}	$\approx 1.19 \times 10^{-7}$	$\approx 2.22 \times 10^{-16}$
Format size	32 bits	64 bits

or *mantissa*, while $f = 0.d_1 \ldots d_{p-1}$ is its *fractional part*. Here we are mostly interested in binary numbers ($\beta = 2$) which can be described by the fractional part f alone if we introduce two classes of numbers. The first class contains *normal* numbers with $d_0 = 1$; these numbers are represented as $\mathrm{fl}(x) = 1.f \cdot 2^e$, while the number zero is defined separately as $\mathrm{fl}(0) = 1.0 \cdot 2^{e_{\min}-1}$. The second class contains *subnormal* numbers, for which $d_0 = 0$. Subnormal numbers fall in the range between the number zero and the smallest positive normal number $2^{e_{\min}}$. They can be represented in the form $\mathrm{fl}(x) = 0.f \cdot 2^{e_{\min}}$. Data types with single ("`float`") and double ("`double`") precision, as well as algorithms for computation of basic operations between them are defined by the IEEE 754 standard; see Table 1.1, the details in Appendix B, as well as [3, 4].

Computations in fixed-point arithmetic (in which numbers are represented by a *fixed* value of e) are faster than those in floating-point arithmetic, and become relevant when working with a restricted range of values. They becomes useful on very specific architectures where large speed and small memory consumption are crucial (for example, in GPS devices or CNC machining tools). In scientific and engineering work, floating-point arithmetic dominates.

The elementary binary operations between floating-point numbers are addition, subtraction, multiplication, and division. We denote these operations by

$$+ : x \oplus y, \qquad - : x \ominus y, \qquad \times : x \otimes y, \qquad / : x \oslash y.$$

Since floating-point numbers have finite precision, the results of the operations $x + y$, $x - y$, $x \times y$, and x/y, computed with exact values of x and y, are not identical to the results of the corresponding operations in finite-precision arithmetic, $x \oplus y$, $x \ominus y$, $x \otimes y$, and $x \oslash y$. One of the key properties of finite-precision arithmetic is the non-associativity of addition and multiplication,

$$x \oplus (y \oplus z) \neq (x \oplus y) \oplus z, \qquad x \otimes (y \otimes z) \neq (x \otimes y) \otimes z.$$

This has important consequences, as demonstrated by the following examples.

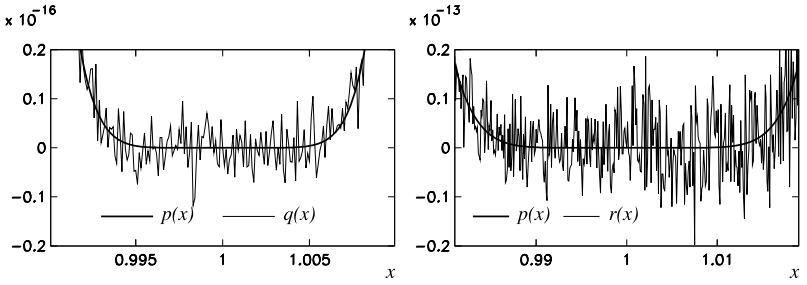


Fig. 1.1 Computation of $(1-x)^8$ in the vicinity of $x=1$ in double-precision floating-point arithmetic. [*Left*] Formula $p(x)=(1-x)^8$ by using `pow(1-x,8)` and formula $q(x)=1-8x+28x^2-\cdots$ by using simple multiplications of x. [*Right*] Computation of $r(x)$ with formula for $q(x)$, but by using `pow` functions for the powers

Example By writing a simple program in C or C++ you can convince yourself that in the case `float x=1e9; float y=-1e9; float z=1;` with the GNU compiler `c++` and option `-O2` you obtain $(\mathtt{x}\oplus\mathtt{y})\oplus\mathtt{z}=1$, while $\mathtt{x}\oplus(\mathtt{y}\oplus\mathtt{z})=0$. (Other compilers may behave differently.)

Example The effects of rounding errors can also be neatly demonstrated [5] by observing the results of three algebraically identical, but numerically different ways of computing the values of the polynomial $(1-x)^8$ in the vicinity of $x=1$. Let us denote the results of the three methods by $p(x)$, $q(x)$, and $r(x)$. We first compute

$$p(x)=(1-x)^8$$

by using the standard power function `pow(1-x, 8)` available in C or C++. The same polynomial can also be expanded as

$$q(x)=1-8x+28x^2-56x^3+70x^4-56x^5+28x^6-8x^7+x^8,$$

which we compute by simple multiplication, for example, $x^3=x\cdot x\cdot x$. Finally, we compute $r(x)$ just like $q(x)$, each term in a row, but evaluate the powers x^n by using the `pow(x, n)` function. Figure 1.1 shows the results of the three methods (be alert to the change of scale on the vertical axis).

Binary operations in floating-point arithmetic are thus performed with errors which depend on the arguments, the type of operation, the programming language, and the compiler. The precision of floating-point arithmetic, called the *machine precision* or *machine epsilon*, and denoted by ε_{M}, is defined as the difference between the representation of the number 1 and the representation of the nearest larger number. In single precision, we have $\varepsilon_{\mathrm{M}}=2^{-23}\approx 1.19\times 10^{-7}$, while in double precision $\varepsilon_{\mathrm{M}}=2^{-52}\approx 2.22\times 10^{-16}$. The precision of the arithmetic can also be defined as the largest ε for which $\mathrm{fl}(1+\varepsilon)=1$, but the value of ε obtained in this manner depends on the rounding method. The IEEE standard prescribes rounding to the

nearest representable result: in this case $\varepsilon \approx \varepsilon_{\mathrm{M}}/2$, which is known as *unit round-off*. For arbitrary data or result of a basic binary operation between normal numbers we have

$$\big|\mathrm{fl}(x) - x\big| \le \frac{\varepsilon_{\mathrm{M}}}{2}|x|, \qquad \big|\mathrm{fl}(x \circ y) - x \circ y\big| \le \frac{\varepsilon_{\mathrm{M}}}{2}|x \circ y|,$$

where $\circ$ denotes a basic binary operation. The floating-point numbers therefore behave as exact values perturbed by a relative error:

$$\mathrm{fl}(x) = x(1+\delta), \quad |\delta| \le \frac{\varepsilon_{\mathrm{M}}}{2}.$$

When operations are performed between such numbers, a seemingly small perturbation δ can greatly increase the error of the result, leading to a *loss of significant digits* (one or more incorrect digits). As an example, we subtract two different numbers $x = \mathrm{fl}(x)(1+\delta_1)$ and $y = \mathrm{fl}(y)(1+\delta_2)$:

$$x - y = \big(\mathrm{fl}(x) - \mathrm{fl}(y)\big)(1+h), \quad h = \frac{\mathrm{fl}(x)\delta_1 - \mathrm{fl}(y)\delta_2}{\mathrm{fl}(x) - \mathrm{fl}(y)}.$$

The relative error is bounded by $|h| \le \varepsilon_{\mathrm{M}} \max\{|x|, |y|\}/|x-y|$ and can become very large if x and y are nearby.

Considering the effects of small perturbations also helps us analyze the relative errors of other basic operations and the precision of more complex algorithms. Special libraries like GMP [7] do allow for computations with almost arbitrary precision, but let us not use this possibility absent-mindedly: do not try to out-smart an unstable algorithm solely by increasing the arithmetic precision!

For a detailed description of how floating- or fixed-point arithmetic behave in typical algorithms, see [4, 8–10]; for details on floating-point data format according to the IEEE 754 standard see Appendix B. We conclude this section by advices for a stable and precise solution of a few seemingly trivial tasks [4, 6].

Branch Statements In programs, we tend to use simple branch statements like

$$\texttt{if } \big(|x_k - x_{k-1}| < \varepsilon_{\mathrm{abs}}\big) \texttt{ then } \ldots, \tag{1.1}$$

where, for example, $\varepsilon_{\mathrm{abs}} = 10^{-6}$. But for $\mathrm{fl}(x_k) = 10^{50}$ the nearest representable number in double precision is $\varepsilon_{\mathrm{M}}\,\mathrm{fl}(x_k) \approx 10^{34}$ away. By using an algorithm which returns x_k in double precision, the condition (1.1) will never be met, except in the trivial case of equal x_k and x_{k-1} (for example, due to rounding to zero). It is much better to use

$$\texttt{if } \big(|x_k - x_{k-1}| < \varepsilon_{\mathrm{abs}} + \varepsilon_{\mathrm{rel}} \max\{x_k, x_{k-1}\}\big) \texttt{ then } \ldots,$$

where $\varepsilon_{\mathrm{rel}} \approx \varepsilon_{\mathrm{M}}$. Similarly, we repeatedly encounter sign-change tests like

$$\texttt{if } (f_k f_{k-1} < 0) \texttt{ then } \ldots. \tag{1.2}$$

If $f_k = 10^{-200}$ and $f_{k-1} = -10^{-200}$, we expect $f_k f_{k-1} = -10^{-400} < 0$, but in double precision we get an underflow $\mathrm{fl}(-10^{-400}) = 0$ and the statement (1.2) goes the wrong way. If $f_k = 10^{200}$ and $f_{k-1} = -10^{200}$, we get an overflow. Let us rather check only the sign of the arguments! In C or C++ this can be accomplished by using the function `template <typename T> int sign(const T & val){ return int((val>0) - (val<0)); }`, and then comparing

$$\texttt{if}\ \big(\texttt{sign}(f_k) \neq \texttt{sign}(f_{k-1})\big)\ \texttt{then}\ \ldots.$$

Roots of the Quadratic Equation The quadratic equation $ax^2 + bx + c = 0$ has the roots

$$x_\pm = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} = \frac{2c}{-b \mp \sqrt{b^2 - 4ac}},$$

related by $x_+ x_- = c/a$. In most cases the absolute value of one of the roots is larger than the other and is computed with a larger relative precision in floating-point arithmetic. Once b and $\sqrt{b^2 - 4ac}$ are known, it is therefore preferable from the numerical viewpoint to first compute the larger root $x_{\max}$ and then use $x_{\min} = c/(ax_{\max})$ to compute the smaller one. In a similar spirit we compute $\sqrt{1+x^2} - 1$ for $|x| \ll 1$ where subtraction and potential loss of significant digits can be avoided by rewriting the expression as

$$\sqrt{1+x^2} - 1 = \frac{(\sqrt{1+x^2} - 1)(\sqrt{1+x^2} + 1)}{\sqrt{1+x^2} + 1} = \frac{x^2}{\sqrt{1+x^2} + 1}.$$

Area of Triangle Heron's formula $S = \sqrt{d(d-a)(d-b)(d-c)}$ for the area of a triangle with sides $a \geq b \geq c$, where $d = (a+b+c)/2$, is very sensitive to round-off errors, in particular when one of the angles is larger than 90° and $a \approx b + c$. In such cases it is advisable to use the following formula which is accurate to $\approx 10\varepsilon_{\mathrm{M}}$:

$$S = \frac{1}{4}\sqrt{\big[a + (b+c)\big]\big[c - (a-b)\big]\big[c + (a-b)\big]\big[a + (b-c)\big]}.$$

Magnitude of Complex Number, Ratio of Complex Numbers The magnitude (absolute value) of a complex number $z = x + \mathrm{i}y$ is $|z| = (x^2 + y^2)^{1/2}$. Squaring and addition of large numbers, which both may lead to overflows, can be avoided by using the formula

$$|z| = \begin{cases} |x|\sqrt{1 + (|y|/|x|)^2}; & 0 < |y| < |x|, \\ |y|\sqrt{1 + (|x|/|y|)^2}; & 0 < |x| < |y|, \\ |x|\sqrt{2}; & \text{otherwise.} \end{cases}$$

We can exploit a similar trick to avoid overflows when computing the ratio of complex numbers $(a + \mathrm{i}b)/(c + \mathrm{i}d) = (ac + bd)/(c^2 + d^2) + \mathrm{i}(bc - ad)/(c^2 + d^2)$. If

this formula is applied directly, overflow may occur even though the correct result is within the allowed range. A safer way to compute the ratio is

$$\frac{a+\mathrm{i}b}{c+\mathrm{i}d} = \begin{cases} \frac{a+b(d/c)}{c+d(d/c)} + \mathrm{i}\frac{b-a(d/c)}{c+d(d/c)}; & |d| < |c|, \\ \frac{b+a(c/d)}{d+c(c/d)} - \mathrm{i}\frac{a-b(c/d)}{d+c(c/d)}; & |d| \geq |c|. \end{cases}$$

Natural Logarithm To compute $\log(1+x)$ at $0 \leq x < 3/4$ we recommend

$$\log(1+x) = \begin{cases} x; & 1 \oplus x = 1, \\ \frac{x\log(1+x)}{(1+x)-1}; & \text{otherwise}, \end{cases} \tag{1.3}$$

which has an error smaller than $5\varepsilon_{\mathrm{M}}$. Such a precise calculation finds its uses in economics for computation of interest rates where wrong results literally cost money. Let us assume we have some funds A and a small interest rate x for a short period of time. After n periods we have $A' = A(1+x)^n$. If $x \ll 1$, errors can accumulate in computing A' for $n \gg 1$. It is preferable to use the formula $A' = A\exp(n\log(1+x))$ and resort to (1.3).

Average of Two Numbers Even a simple expression like the arithmetic mean of two floating-point numbers, $x = (a+b)/2$, may overflow, and one should use $x = a + (b-a)/2$ or $a/2 + b/2$ instead.

1.2 Approximation of Expressions

Approximation is one of the key concepts of numerical analysis [11, 12]. In this book we only refer to approximations of scalar functions and expressions involving operators, and therefore only discuss these two examples.

1.2.1 Optimal (Minimax) and Almost Optimal Approximations

The optimal approximation of degree n of a continuous function f on the interval $[a,b]$ is defined as the polynomial $p_n^\star$ for which $E_n^\star \in \mathbb{R}$ exists such that

$$E_n^\star = \max_{a\leq x\leq b} \left|p_n^\star(x) - f(x)\right| \leq \max_{a\leq x\leq b} \left|p_n(x) - f(x)\right|, \tag{1.4}$$

where p_n is any polynomial of degree n. We are therefore seeking a polynomial $p_n^\star$ that minimizes the maximal error with respect to the function f. Such a polynomial represents an *optimal* or *minimax approximation*. The curve $p_8^\star$ in Fig. 1.2 (left) is the optimal approximation of the function $\mathrm{e}^x \sin(3\pi x)$ by a polynomial of degree eight ($n = 8$), while the corresponding curve in the right panel is the error of the approximation.

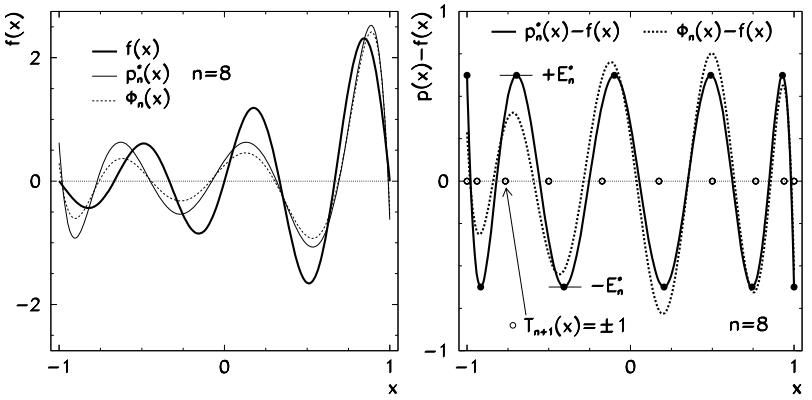


Fig. 1.2 Approximation of $f(x) = \mathrm{e}^x \sin(3\pi x)$ by an optimal polynomial and by a Chebyshev expansion. [*Left*] The function f, the optimal polynomial $p_8^\star$ of degree eight, and the expansion in terms of Chebyshev polynomials $\phi_8(x) = \sum_{k=0}^{8} \hat{\tau}_k T_k(x)$. [*Right*] Error of the approximations. The *symbols* • denote $(n+2)$ characteristic points at which the absolute value of the error $\pm E_n^\star = \pm|p_n^\star - f|$ is maximal, and between which the error changes its sign $(n+1)$-times. The *symbols* ∘ denote the points at which $T_{n+1}(x) = \pm 1$

The optimal polynomial approximation of a continuous function f on $[a, b]$ can be computed by the iterative Remes algorithm (see e.g. [13]). The basis for this procedure is the Borel equi-oscillation theorem which states that a degree-n polynomial $p_n^\star$ is the optimal approximation of f precisely when $(n+2)$ distinct points x_i exist, arranged as $a \le x_0 < x_1 < \cdots < x_{n+1} \le b$, for which

$$p_n^\star(x_i) - f(x_i) = \lambda(-1)^i E_n^\star, \quad i = 0, 1, \ldots, n+1,$$

and where λ has a fixed value of $+1$ or -1. The extremes $\pm E_n^\star$ of the error of the optimal approximation $p_n^\star$ therefore occur at the points x_i with the signs flipping back and forth (Fig. 1.2 (right)). This implies that the coefficients of $p_n(x) = \sum_{j=0}^{n} a_j x^j$ and the parameter E_n must fulfill the system of equations

$$\sum_{j=0}^{n} a_j x_i^j - f(x_i) = (-1)^i E_n, \quad i = 0, 1, \ldots, n+1, \tag{1.5}$$

$$\sum_{j=0}^{n} a_j x_i^j - f(x_i) = \text{extreme}. \tag{1.6}$$

If f is differentiable, we can rewrite the last equation as

$$\sum_{j=1}^{n} j a_j x_i^{j-1} - f'(x_i) = 0. \tag{1.7}$$

The iteration is started with an initial guess for x_i. We show in the following that a very good choice for x_i are the points at which the Chebyshev polynomials T_{n+1} are equal to ± 1, so

$$x_i = \cos\left(\frac{i\pi}{n+1}\right), \quad i = 0, 1, \ldots, n+1.$$

By solving the linear system (1.5) we obtain $n+1$ coefficients a_j and the parameter E_n, which are the first approximations for the coefficients of the optimal polynomial and the maximum error of $p - f$. These parameters are then used to solve the system (1.6) or (1.7): we search for points x_i at which the error $p - f$ has an extreme. This is the most difficult step in the procedure, especially when it needs to be automated. The new points x_i are then again used in (1.5) and we repeat the procedure until the difference between the consecutive values of E_n drops below a desired tolerance. At the end of the iteration the $\{a_j\}_{j=0}^n$ are the coefficients of the optimal polynomial $p_n^\star$, and $E_n = E_n^\star$.

Actually, the power basis $\{1, x, x^2, \ldots\}$ of $p_n^\star$ is a bad choice. The condition number of the matrix corresponding to the system (1.5) deteriorates exponentially with increasing n. Instead of polynomials, rational functions $p(x) = P_n(x)/Q_m(x)$ can be plugged into the Remes procedure, but the system of equations for the $(n+1)+(m+1)$ coefficients and the errors E becomes non-linear; it can be solved by linearization [14]. In the context of high-order optimal approximations, Chebyshev polynomials are also an attractive option [15].

Even without the Remes algorithm, the Chebyshev polynomials lead to an almost identical goal. Instead of searching for the optimal polynomial $p_n^\star$, we may be satisfied by finding an approximation q_n for which

$$\varepsilon = \max_{a \le x \le b} \left| p_n^\star(x) - q_n(x) \right|$$

with some small ε. Such an approximation is called an *almost optimal* or *near-minimax approximation*. Often it is much easier to find than the true optimal approximation. The most important among them is the approximation by Chebyshev polynomials

$$f(x) \approx \phi_n(x) = \sum_{k=0}^{n} \hat{\tau}_k T_k(x), \tag{1.8}$$

where the coefficients $\hat{\tau}_k$ are given in (4.39). We switch between the intervals $[a, b]$ and $[-1, 1]$ (on which Chebyshev polynomials are defined) by the transformations

$$t \mapsto x = 2\frac{t-a}{b-a} - 1, \qquad x \mapsto t = \frac{1-x}{2}\, a + \frac{1+x}{2}\, b.$$

The summation of the series (1.8) can be terminated at the same n that would be used in the optimal approximation. The error due to the series truncation is then determined by the term $\hat{\tau}_{n+1} T_{n+1}(x)$, which has interchanging extreme values of

$\pm\hat{\tau}_{n+1}$ at $(n+2)$ points on $[-1,1]$ and closely resembles the genuine optimal approximation (see Fig. 1.2 (left and right)).

Chebyshev polynomials have many pleasing and useful properties which we exploit heavily in function and signal transformations (Chap. 4) and spectral methods for partial differential equations (Chap. 11). In the context of optimal approximations, an important property of Chebyshev polynomials is their point-wise orthogonality (4.38). There exists a whole class of polynomials which are orthogonal on a discrete set of points, but only Chebyshev polynomials on $[-1,1]$ oscillate uniformly and can be generated by so few numerical operations (see recurrence (4.37)).

The shape of the optimal approximation depends on the norm in which its error is measured. Equation (1.4) defines the *uniform* optimal approximation in the "max"-norm $\|\cdot\|_\infty$. The computation of the optimal approximation for an arbitrary function, in particular if it is given only at specific points, can be cumbersome. The main nuisance is the large sensitivity of the Remes algorithm to individual values $f(x_i)$ that strongly deviate from the average (e.g. outliers in discrete-time signals). In physics we often resort to the optimal approximation in the Euclidean norm (A.2). In this case, the optimal approximation of the function f on $[a,b]$ is a function $p^\star$ for which $E^\star \in \mathbb{R}$ exists such that

$$E^\star = \int_a^b \big[p^\star(x) - f(x)\big]^2 \,\mathrm{d}x \le \int_a^b \big[p(x) - f(x)\big]^2 \,\mathrm{d}x \tag{1.9}$$

for any polynomial p of degree n (compare this expression to (1.4)). The form (1.9) is less sensitive to outliers mentioned above. We are of course referring to the method of least squares which we most often encounter in its discrete form, where we search for

$$\min_{p\in P_n} \sum_i \big[p(x_i) - f(x_i)\big]^2.$$

1.2.2 Rational (Padé) Approximation

Suppose that, on some interval, we wish to effectively approximate a function f possessing a power expansion

$$f(z) = \sum_{k=0}^{\infty} c_k z^k. \tag{1.10}$$

A Padé approximation of f is a rational function with a numerator of degree L and denominator of degree M, determined such that its power expansion matches the series (1.10) up to including the power $L+M$. In other words, if we can find a polynomial P_L of degree L and Q_M of degree M such that

$$f(z) = \frac{P_L(z)}{Q_M(z)} + \mathcal{O}\big(z^{L+M+1}\big), \quad Q_M(0) = 1,$$

then

$$[L/M]_f(z) = \frac{P_L(z)}{Q_M(z)} = \frac{a_0 + a_1 z + a_2 z^2 + \cdots + a_L z^L}{b_0 + b_1 z + b_2 z^2 + \cdots + b_M z^M}, \quad b_0 = 1,$$

defines a Padé approximation of order (L, M) of the function f. The coefficients a_k and b_k can be determined by equating the approximation $[L/M]_f$ with the power series for f and reading off the coefficients of the same powers of x:

$$\begin{aligned}&\big(b_0 + b_1 z + \cdots + b_M z^M\big)(c_0 + c_1 z + \cdots) \\ &\quad = a_0 + a_1 z + \cdots + a_L z^L + \mathcal{O}\big(z^{L+M+1}\big).\end{aligned}$$

By comparing the terms with powers $z^{L+1}, z^{L+2}, \ldots, z^{L+M}$ we obtain a system of equations for the coefficients b_k of the denominator Q_M, while by comparing terms with powers $z^0, z^1, \ldots, z^L$ we obtain explicit equations for the coefficients a_k of the numerator P_L. For example, with $L = M = 3$, we get

$$\begin{pmatrix} c_3 & c_2 & c_1 \\ c_4 & c_3 & c_2 \\ c_5 & c_4 & c_3 \end{pmatrix} \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix} = \begin{pmatrix} -c_4 \\ -c_5 \\ -c_6 \end{pmatrix}, \qquad \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{pmatrix} = \begin{pmatrix} c_0 & 0 & 0 & 0 \\ c_1 & c_0 & 0 & 0 \\ c_2 & c_1 & c_0 & 0 \\ c_3 & c_2 & c_1 & c_0 \end{pmatrix} \begin{pmatrix} 1 \\ b_1 \\ b_2 \\ b_3 \end{pmatrix}.$$

Since $b_0 = 1$, the first line of the equation on the right tells us that the zeroth coefficient of P_L is equal to the zeroth coefficient of the power expansion of the function, $a_0 = c_0$. The bulk of the work is hidden in the matrix system on the left. Large Padé systems of this type are solved by robust algorithms like Gauss elimination with complete pivoting because, in most cases, we are interested in relatively low degrees L and M and because accuracy takes precedence over speed. In the sense of properties mentioned in the following, *diagonal Padé approximations*, in which $L = M$, are the most efficient.

Example (Adapted from [16], p. 4.) The function

$$f(z) = \sqrt{\frac{1 + z/2}{1 + 2z}} = \sum_{k=0}^{\infty} c_k z^k = 1 - \frac{3}{4} z + \frac{39}{32} z^2 - \frac{267}{128} z^3 + \frac{7563}{2048} z^4 - \cdots, \quad (1.11)$$

has the first two diagonal Padé approximations

$$[1/1]_f(z) = \frac{1 + \frac{7}{8} z}{1 + \frac{13}{8} z}, \qquad [2/2]_f(z) = \frac{1 + \frac{17}{8} z + \frac{61}{64} z^2}{1 + \frac{23}{8} z + \frac{121}{64} z^2}.$$

(Compute them!) The comparison of two power expansions and these Padé approximations is shown in Fig. 1.3, revealing a most delightful property: if some power series converges to a function with a convergence radius ρ, that is, for all $|z| < \rho$ and $0 < \rho < \infty$, then an appropriately chosen Padé approximation converges to f

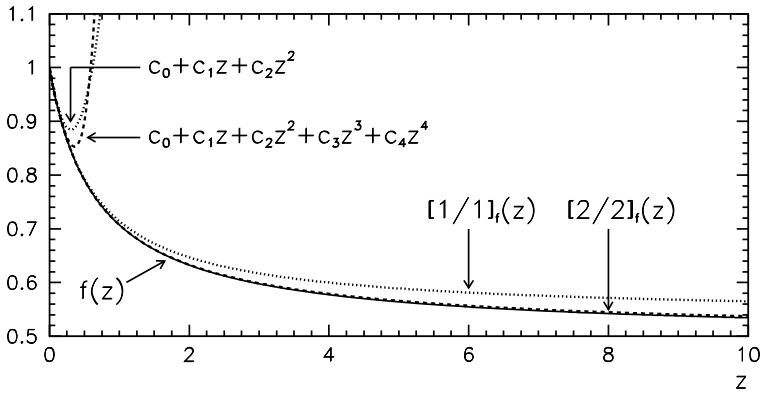


Fig. 1.3 Comparison of power expansions of degree two and four for the function (1.11) and the diagonal Padé approximations $[1/1]_f$ and $[2/2]_f$. The curves for f and $[2/2]_f$ are barely distinguishable in this plot

for all $z \in \mathcal{R}$, where the domain $\mathcal{R}$ is larger than the domain defined by $|z| < \rho$. The function f in example (1.11) has the limit $\lim_{z\to\infty} f(z) = 1/2$ and its power series has a convergence radius of only $\rho = 1/2$. On the other hand, both lowest-order diagonal Padé approximations are stable at infinity. Moreover, when the order of the approximation is increased, the correct limit $1/2$ is approached rapidly: $\lim_{z\to\infty}[1/1]_f(z) = 7/13 \approx 0.5385$ and $\lim_{z\to\infty}[2/2]_f(z) = 61/121 \approx 0.5041$. In other words, the Padé approximation tells us something about how the function behaves outside of the convergence radius of its power series, and ensures better asymptotics.

Example Take the function $f(z) = \mathrm{e}^z$, for which the lowest-order Padé approximations are listed in Table H.1. This table shows the Padé approximation in the context of solving partial differential equations and also summarizes the leading error terms. The Padé approach is also fruitful in approximating the time evolution of Hamiltonian operators: see example (1.17) below.

In certain cases the Padé approximation of a function f does not exist. For example, for $f(z) = 1 + z^2$ we would attempt to compute the Padé coefficients such that $(a_0 + a_1 z)/(b_0 + b_1 z) = 1 + z^2 + \mathcal{O}(z^3)$, and by comparing coefficients with equal powers of z we would get $a_0 = b_0$, $a_1 = b_1$ and $a_0 = 0$. This would lead to $[1/1]_f = (0 + a_1 z)/(0 + b_1 z) = 1 \neq 1 + z^2 + \mathcal{O}(z^3)$. One can encounter problems if the denominator of the Padé approximation is zero when $z = 0$. For details, see [16].

Padé approximations have numerous important transformation and invariance properties, of which *duality* and *unitarity* are the most relevant. Duality connects the Padé approximations for reciprocal functions:

$$g(z) = \big\{f(z)\big\}^{-1} \quad \text{and} \quad f(0) \neq 0 \quad \Leftrightarrow \quad [L/M]_g(z) = \big\{[M/L]_f(z)\big\}^{-1} \; \forall L, M.$$

In physical applications, unitarity is even more important, in particular in the theory of scattering matrices. Assume that $f(z) = \sum_{k=0}^{\infty} c_k z^k$ is unitary, so that $f(z) f^*(z) = 1$, and that $[M/M]_f$ is its diagonal Padé approximation. Then we also have

$$[M/M]_f(z)[M/M]_f^*(z) = 1,$$

where the symbol $*$ denotes complex conjugation of the coefficients of the approximation (not of the argument z). In approximations of time evolution of Hamiltonian operators (see (1.16)), preservation of unitarity is crucial.

1.2.3 Summation of Series by Using Padé Approximations (Wynn's ϵ-Algorithm)

Summation of series is the topic of Sect. 1.4, but here we wish to discuss an important method based on the Padé approximation. We have just witnessed how this approximation can be used to extend the convergence radius of a power series. But this very same approximation can be used to accelerate the summation of a series within its convergence radius. Assume that a series (in the region where it converges) can be approximated by a rational function f which is analytical in this region and has the form

$$f(z) = c_0 + c_1 z + c_2 z^2 + \cdots = \frac{P_L(z)}{Q_M(z)} \equiv [L/M]_f(z),$$

where $P_L(z) = a_0 + a_1 z + \cdots + a_L z^L$ and $Q_M(z) = 1 + b_1 z + \cdots + b_M z^M$. An efficient procedure to compute the diagonal approximations at a given z can be obtained by transformations between the values of $[L/M]_f(z)$ for different L and M. A connection between these approximations can be established which is known as the Wynn's ϵ-algorithm [17]. It can be written as a recurrence,

$$\epsilon(n, m+1) = \epsilon(n+1, m-1) + \frac{1}{\epsilon(n+1, m) - \epsilon(n, m)}. \tag{1.12}$$

We start the recurrence with $\epsilon(n, -1) = 0$ for $\forall n \geq 0$ and $\epsilon(n, 0)$ which contain the partial sums

$$\epsilon(n, 0) = S_n = \sum_{k=0}^{n} c_k z^k, \quad n \geq 0.$$

The initial state of the algorithm is given by the first two columns of the $\epsilon(n, m)$ table:

$$\begin{array}{cccccc}
\epsilon(0,-1)=0 & \underline{\epsilon(0,0)=S_0} & \epsilon(0,1) & \underline{\epsilon(0,2)} & \epsilon(0,3) & \underline{\epsilon(0,4)} & \cdots \\
\epsilon(1,-1)=0 & \epsilon(1,0)=S_1 & \epsilon(1,1) & \epsilon(1,2) & \epsilon(1,3) & \epsilon(1,4) & \cdots \\
\epsilon(2,-1)=0 & \epsilon(2,0)=S_2 & \epsilon(2,1) & \epsilon(2,2) & \epsilon(2,3) & \epsilon(2,4) & \cdots \\
\epsilon(3,-1)=0 & \epsilon(3,0)=S_3 & \epsilon(3,1) & \epsilon(3,2) & \epsilon(3,3) & \epsilon(3,4) & \cdots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots &
\end{array}$$

By repeated use of (1.12) we obtain all other entries, and these are related to Padé approximations of various orders. The diagonal approximations $[p/p]_f(z)$ at a given z can be read off from the underlined entries with even indices m:

$$[p/p]_f(z) = \epsilon(0, 2p), \quad p = 0, 1, \ldots.$$

We stop the algorithm when the value of the denominator of the fraction in (1.12) drops below a prescribed tolerance; since the convergence is very fast, this can be set to $\approx\varepsilon_{\mathrm{M}}$. Some authors recommend the Wynn procedure as the best general algorithm to accelerate the summation of any slowly converging series [18].

As an exercise, compare the extremely slow convergence of the partial sums $S_n = \sum_{k=0}^{n}(-1)^k/(k+1)$ with $\lim_{n\to\infty} S_n = \log 2$ (second column of the Wynn table) to the much faster convergence of the entries $\epsilon(0,0), \epsilon(0,2), \epsilon(0,4), \ldots$ in the first row of the table!

Example The Wynn procedure also enables us to evaluate asymptotic (divergent) series in the sense of Sect. 1.3.2. The exponential integral $f(z) = \mathrm{Ei}(z)$ for large positive z can be represented by the asymptotic series

$$\mathrm{Ei}(z) = \int_z^\infty \frac{\mathrm{e}^{-t}}{t}\,\mathrm{d}t \sim \frac{\mathrm{e}^{-z}}{z}\left[1 - \frac{1!}{z} + \frac{2!}{z^2} - \frac{3!}{z^3} + \cdots\right], \quad z \to \infty. \tag{1.13}$$

The partial sums of (1.13),

$$S_n = \frac{\mathrm{e}^{-z}}{z}\sum_{k=0}^{n}\frac{k!}{(-z)^k}, \quad n \geq 0, \tag{1.14}$$

do not converge for any z, as the upper two curves in Fig. 1.4 (left) indicate for $z = 2$ and $z = 4$. (At even higher z, the minimum of the error would just shift to the right and downwards.) The lower two curves in the same Figure show the error when the series is evaluated by the Wynn algorithm.

Example A physically more convincing use of the Wynn's method for divergent series is described in Problem 1.5.5. There, the function f represents the Coulomb scattering amplitude with a divergent power series in $z = \cos\theta$:

$$f(\theta) = \frac{1}{2\mathrm{i}k}\sum_{l=0}^{\infty}(2l+1)P_l(\cos\theta)\left(\mathrm{e}^{2\mathrm{i}\sigma_l} - 1\right). \tag{1.15}$$

The errors of the partial sums of (1.15) and of the Wynn approximations with respect to the exact amplitude are shown in Fig. 1.4 (right).

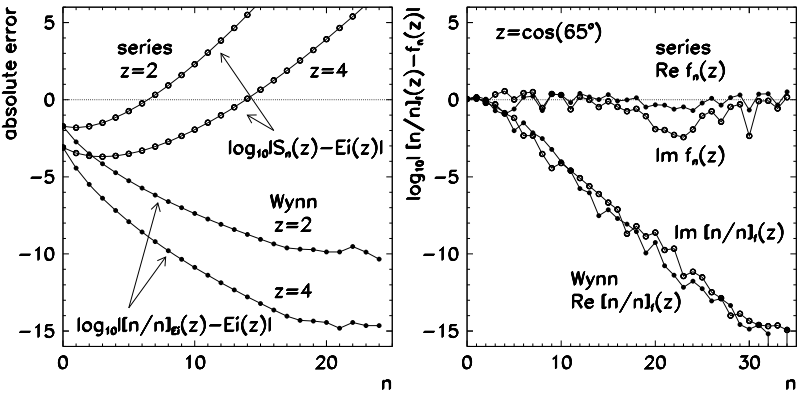


Fig. 1.4 Evaluation of asymptotic series by the Wynn algorithm. [*Left*] Errors of the partial sums (1.14) of (1.13) and of the Padé approximation $[n/n]_{\mathrm{Ei}}$ with respect to the exact values Ei(2) and Ei(4) as a function of the index n. [*Right*] Errors of the partial sums of (1.15) and of $[n/n]_f$ with respect to the exact amplitude (1.71) at $\theta = 65°$

1.2.4 Approximation of the Evolution Operator for a Hamiltonian System

The time evolution of a quantum Hamiltonian system from time t to time $t + \Delta t$ (for example, of a wave-function $\Psi(x, t)$ governed by the non-stationary Schrödinger equation) is given by

$$\Psi(x, t + \Delta t) = \mathrm{e}^{-\mathrm{i}\, H \Delta t/\hbar} \Psi(x, t). \tag{1.16}$$

In difference methods for partial differential equations (Chap. 9) we prefer to approximate the evolution operator $\exp[-\mathrm{i}H\Delta t/\hbar]$ by low-order unitary approximations. If the order of the approximation is too low, the solution can become noisy and meaningless even on short time scales. If H does not depend on time, a good way to improve the accuracy of the difference method in its temporal part is to approximate the exponential operator by a diagonal Padé approximation

$$\mathrm{e}^z = \sum_{k=0}^{\infty} c_k z^k = \frac{1 + a_1 z + \cdots + a_M z^M}{1 + b_1 z + \cdots + b_M z^M} + \mathcal{O}\big(z^{2M+1}\big), \tag{1.17}$$

where $c_k = 1/k!$, and the coefficients a_m and b_m are complex in general. The coefficients at a chosen order M are computed by the procedure described on p. 10. The numerator and the denominator of (1.17) can then be factorized as [16]

$$\mathrm{e}^z = \prod_{s=1}^{M} \left(\frac{1 - z/z_s^{(M)}}{1 + z/z_s^{*(M)}} \right) + \mathcal{O}\big(z^{2M+1}\big), \tag{1.18}$$

where $z_s^{(M)}$ are the zeros of the numerator, and $z_s^{*(M)}$ the zeros of the denominator (which are just their complex conjugates). For example, these zeros up to $M = 3$, in double precision, are

$$\begin{aligned} z_1^{(1)} &= -2, \\ z_{1,2}^{(2)} &= -3 \pm \mathrm{i}\sqrt{3}, \\ z_1^{(3)} &= -4.6443707092521712, \\ z_{2,3}^{(3)} &= -3.6778146453739144 \pm \mathrm{i}3.5087619195674433. \end{aligned}$$

At the lowest order ($M = 1$) the expression (1.18) with $z = -\mathrm{i}H\Delta t/\hbar$ simplifies to

$$\mathrm{e}^{-\mathrm{i}H\Delta t/\hbar} = \frac{1 - \frac{1}{2}\mathrm{i}H\Delta t/\hbar}{1 + \frac{1}{2}\mathrm{i}\,H\Delta t/\hbar} + \mathcal{O}\big(\Delta t^3\big).$$

In the context of (1.16) this can be read as

$$\left[1 + \frac{\mathrm{i}H\Delta t}{2\hbar}\right]\Psi(x, t + \Delta t) = \left[1 - \frac{\mathrm{i}H\Delta t}{2\hbar}\right]\Psi(x, t)$$

(see Problem 9.13.8). At higher orders, we get a product operator

$$\mathrm{e}^{-\mathrm{i}H\Delta t/\hbar} \approx \prod_{s=1}^{M} E_s^{(M)}, \quad E_s^{(M)} = \frac{1 + (\mathrm{i}\,H\Delta t/\hbar)/z_s^{(M)}}{1 - (\mathrm{i}\,H\Delta t/\hbar)/z_s^{*(M)}} \equiv \frac{R_s^{(M)}}{L_s^{(M)}},$$

which allows us to compute the time evolution as

$$\Psi(x, t_{n+1}) = E_M^{(M)} \cdots E_2^{(M)} E_1^{(M)} \Psi(x, t_n), \quad t_{n+1} - t_n = \Delta t.$$

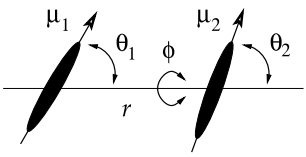
In a practical implementation, this implies a sequence of steps

$$\begin{aligned} L_1^{(M)}\Psi(x, t_{n+1/M}) &= R_1^{(M)}\Psi(x, t_n), \\ L_2^{(M)}\Psi(x, t_{n+2/M}) &= R_2^{(M)}\Psi(x, t_{n+1/M}), \\ &\cdots \quad \cdots \\ L_M^{(M)}\Psi(x, t_{n+1}) &= R_M^{(M)}\Psi(x, t_{n+(M-1)/M}). \end{aligned}$$

Note the symbolic notation: $t_{n+1} = t_n + \Delta t$ for each n, but the time advance in each line ($t_n \to t_{n+1/M}, t_{n+1/M} \to t_{n+2/M}, \ldots$) is not equidistant. If H does not depend on time, the sequence of steps is arbitrary. The band structure of matrices L and R depends on the spatial discretization of H: if it contains the kinetic energy operator $-(\hbar^2/2m)\partial^2/\partial x^2$, which is discretized in the form (9.11) on an equidistant spatial mesh, the matrices are tridiagonal; if it is discretized as in (9.12), they are pentadiagonal. Details can be found in [19]; the methods for time-dependent Hamiltonians are discussed in [20, 21].

1.3 Power and Asymptotic Expansion, Asymptotic Analysis

Power expansion is a tool to describe the behavior of a function in the vicinity of a specific point, while asymptotic expansion and analysis are languages in which we express the dependence of integrals and solutions of differential equations on parameters governing their behavior in the limit of very small or very large values. The basic notation (symbols $\mathcal{O}$, o and $\sim$) is given in Appendix A.1.



Example A nice instance of asymptotic analysis can be found in the study of the system of two freely rotating electric dipoles with an interaction of the form

$$V(r,\Omega)=\frac{\mu_1\mu_2}{4\pi\varepsilon_0 r^3}F(\Omega),\qquad \Omega=(\theta_1,\theta_2,\phi),$$

where $F(\Omega)=-2\cos\theta_1\cos\theta_2+\sin\theta_1\sin\theta_2\cos\phi$. One encounters such systems in studies of orientation of nitroaniline molecules in zeoliths. The relevant quantity is the interaction energy V weighted by the Boltzmann factor $\exp(-V/kT)$ and averaged over all possible orientations of the dipoles Ω,

$$\left\langle V\mathrm{e}^{-V/kT}\right\rangle=\frac{\mu_1\mu_2}{4\pi\varepsilon_0 r^3}\frac{\int \mathrm{d}\Omega\, F(\Omega)\,\mathrm{e}^{\lambda F(\Omega)}}{\int \mathrm{d}\Omega\,\mathrm{e}^{\lambda F(\Omega)}},\qquad \lambda=\frac{\mu_1\mu_2}{4\pi\varepsilon_0 r^3 kT},$$

where $\mathrm{d}\Omega=\sin\theta_1\sin\theta_2\,\mathrm{d}\theta_1\,\mathrm{d}\theta_2\,\mathrm{d}\phi$. By using standard collections of integrals [22, 23] the triple integrals can be reduced to a single integration [24, 25],

$$\left\langle V\mathrm{e}^{-V/kT}\right\rangle=-\frac{\mu_1\mu_2}{4\pi\varepsilon_0 r^3}\frac{\mathrm{d}}{\mathrm{d}\lambda}\log K(\lambda),\qquad K(\lambda)=\frac{8\pi}{\sqrt{3}\,\lambda}\int_1^2\frac{\sinh\lambda x}{\sqrt{x^2-1}}\,\mathrm{d}x.$$

We are interested in the behavior of $K(\lambda)$ for large values of the parameter λ, i.e. at fixed r and low temperatures T. By asymptotic analysis (see p. 23) we get

$$K(\lambda)\sim\frac{4\pi}{3}\frac{\mathrm{e}^{2\lambda}}{\lambda^2}\left(1+\frac{2}{3\lambda}+\frac{1}{\lambda^2}+\frac{22}{9\lambda^3}+\cdots\right) \tag{1.19}$$

or

$$\frac{\mathrm{d}}{\mathrm{d}\lambda}\log K(\lambda)=\frac{1}{K}\frac{\mathrm{d}K}{\mathrm{d}\lambda}\sim 2-\frac{2}{\lambda}-\frac{2}{3\lambda^2}+\cdots,\qquad \lambda\to\infty. \tag{1.20}$$

The terms in the expansion (1.20) have clear physical meanings. The leading term $+2$ does not depend on T and reflects the attraction of the dipoles in the state with $\theta_1=\theta_2=0$ that has the lowest energy, $\langle V\exp(-V/kT)\rangle=-\mu_1\mu_2/2\pi\varepsilon_0 r^3$. The second term (proportional to T) expresses the average potential energy of a pair of anisotropic two-dimensional oscillators [24, 25]. The third term (proportional to T^2) corresponds to the non-harmonic part of the potential.

1.3.1 Power Expansion

Assume that a real function f is at least $(n+1)$-times differentiable in the vicinity of the point x_0. The nth order *power (Taylor) expansion* of f is defined as

$$f(x) = \sum_{k=0}^{n} \frac{f^{(k)}(x_0)}{k!}(x - x_0)^k + R_n(x) \tag{1.21}$$

with the remainder

$$R_n(x) = \int_{x_0}^{x} \frac{f^{(n+1)}(t)}{n!}(x-t)^n \,\mathrm{d}t = \frac{f^{(n+1)}(x^\star)}{(n+1)!}(x - x_0)^{n+1}, \quad x^\star \in [x_0, x].$$

In the last step we have used the mean value theorem [26]: for any continuous function g on the interval $[a, b]$ there exists a point $a \le x^\star \le b$ such that

$$\int_a^b g(x)\,\mathrm{d}x = g\big(x^\star\big)(b-a),$$

and $g(x^\star)$ is the average value of g. If at x_0 all derivatives of f exist and are finite, f can be expanded in the vicinity of x_0 in an infinite series. This series converges on the interval $(x_0 - r, x_0 + r)$ where r is the *convergence radius of the series*. It is given by the formulas (1.55).

The extension of the Taylor series to the complex plane and inclusion of negative powers of the arguments is called the *Laurent expansion*:

$$f(z) = \sum_{k\in\mathbb{Z}} a_k (z - z_0)^k, \quad a_k = \frac{1}{2\pi \mathrm{i}} \oint_\gamma \frac{f(z)}{(z - z_0)^{k+1}}\,\mathrm{d}z.$$

Here γ is an arbitrary closed contour encircling z_0 in the positive sense. According to the coefficients $\{a_{-k}\}_{k\in\mathbb{N}}$ of the negative powers, we have three distinct cases. The function f is *analytic* if all these coefficients are zero; it is *meromorphic* if there exists a minimal N such that $a_{-k} = 0$ for $\forall k \ge N$; if such N does not exist, we say that f has an *essential singularity* at z_0.

1.3.2 Asymptotic Expansion

The asymptotic expansion of a function is a series, the partial sum of which is an approximation of this function in the regime where the parameter describing the asymptotics becomes large or small. The classical example, used by virtually all textbooks (see, for example, [27]), is the gamma function. For small x we have the Laurent expansion

$$\Gamma(x) = \frac{1}{x} - \gamma + \left(\frac{\gamma^2}{2} + \frac{\pi^2}{12}\right)x + \cdots, \quad 0 < |x| < 1, \tag{1.22}$$

where $\gamma \approx 0.577216$ is the Euler constant. The expansion (1.22) converges for all x satisfying $0 < |x| < 1$, and both the left and the right side are functions of x. On the other hand, for large positive x, we have the expansion

$$\Gamma(x) \sim \mathrm{e}^{-x} x^x \sqrt{\frac{2\pi}{x}} \left(1 + \frac{1}{12x} + \frac{1}{288x^2} + \cdots\right), \quad x \to \infty, \tag{1.23}$$

which does not converge for *any* x! The right side of (1.23) represents an *asymptotic expansion* of the function $\Gamma(x)$ in the limit $x \to \infty$ and is not a function of x. It is merely a sequence of approximations for the function. The error due to the truncation of the series at order n and fixed x does *not* go to zero when n is increased, but it does vanish when $x \to \infty$ at fixed n.

An asymptotic series may converge or diverge. Colloquially, an asymptotic series usually means a divergent series. If the series converges, it can be summed up to an arbitrary term. But it makes no sense to sum a divergent asymptotic series; rather, we use such a series to obtain as good as possible an approximation to the function f. We sum the series only until the sequence of partial sums appears to converge, or stop the summation with the term just before the smallest one. (It turns out that in many asymptotic series the truncation error does not exceed the first omitted term.)

For a more general discussion of asymptotics at an arbitrary point x_0 we choose a sequence of functions $\{\phi_k\}_{k=0}^{\infty}$ with the property $\phi_{k+1}(x) = o(\phi_k(x))$ in the limit $x \to x_0$ and $\phi_k(x) \neq 0$ in the neighborhood of x_0, excluding x_0 itself. The sum $\sum_{k=0}^{\infty} c_k \phi_k(x)$ is called the *general asymptotic expansion* of f if

$$f(x) = \sum_{k=0}^{n} c_k \phi_k(x) + o\big(\phi_n(x)\big), \quad x \to x_0. \tag{1.24}$$

The limit point x_0 can be finite or infinite. With the chosen sequence of functions $\{\phi_k\}$, the expansion coefficients in (1.24) are given by

$$c_k = \lim_{x \to x_0} \frac{f(x) - \sum_{m=0}^{k-1} c_m \phi_m(x)}{\phi_k(x)}, \quad k = 0, 1, \ldots, n. \tag{1.25}$$

Example (See [27], p. 30) We seek an asymptotic expansion of the function

$$f(x) = \frac{1}{x} + \mathrm{e}^{-x} \left(1 - \frac{1}{x}\right)^{-1}, \quad x \to \infty, \tag{1.26}$$

based on the sequence of functions $\{\phi_k(x)\}_{k=0}^{\infty} = \{1, x^{-1}, x^{-2}, \ldots\}$. We compute the coefficients c_k of (1.24) by using (1.25): we get $c_0 = \lim_{x\to\infty} f(x)/\phi_0(x) = 0$ and $c_1 = \lim_{x\to\infty}(f(x) - c_0\phi_0(x))/\phi_1(x) = 1$, while $c_k = 0$ for $k \geq 2$. Based on the chosen sequence $\{\phi_k(x)\}_{k=0}^{\infty}$, the function f has the expansion

$$f(x) \sim c_1 \phi_1(x) = \frac{1}{x}, \quad x \to \infty. \tag{1.27}$$

The same function f can have another asymptotic expansion if a different sequence $\{\phi_k\}_{k=0}^{\infty}$ is chosen. If we select $\{\phi_k\}_{k=0}^{\infty} = \{1, x^{-1}, \mathrm{e}^{-x}, \mathrm{e}^{-x}x^{-1}, \mathrm{e}^{-x}x^{-2}, \ldots\}$, we get for the same f as before the coefficients $c_0 = 0$ and $c_k = 1$ for $k \geq 1$, thus

$$f(x) \sim \frac{1}{x} + \sum_{k=1}^{\infty} \mathrm{e}^{-x} x^{1-k}, \quad x \to \infty.$$

By reading this example backwards we realize that different functions may correspond to the same asymptotic expansion. Based on the sequence $\{1, x^{-1}, x^{-2}, \ldots\}$ both (1.26) and $f(x) = 1/x$ have identical expansions, namely (1.27).

1.3.3 Asymptotic Analysis of Integrals by Integration by Parts

For an arbitrary function f and positive $m \in \mathbb{N}$ let f_m (lower case) represent its mth derivative and F_m (upper case) its mth indefinite integral,

$$f_0 = f, \qquad f_m = \frac{\mathrm{d}^m f}{\mathrm{d}x^m}, \qquad \frac{\mathrm{d}F_m}{\mathrm{d}x} = F_{m-1}.$$

Suppose we are interested in the asymptotic behavior of the integral

$$I(\lambda) = \int_a^b g(\lambda, x) h(\lambda, x)\, \mathrm{d}x,$$

where the asymptotics is determined by the parameter λ. Let g be at least n-times differentiable and let h be integrable. By using integration by parts, $I(\lambda)$ can be written as the sum

$$I_n(\lambda) = \sum_{k=0}^{n-1} s_k(\lambda) + R_n(\lambda),$$

where the terms s_k and the remainder R_n are

$$s_k(\lambda) = (-1)^k \big[g_k(\lambda, b) H_{k+1}(\lambda, b) - g_k(\lambda, a) H_{k+1}(\lambda, a) \big],$$

$$R_n(\lambda) = (-1)^n \int_a^b g_n(\lambda, x) H_n(\lambda, x)\, \mathrm{d}x.$$

If g is $(n+1)$-times continuously differentiable, we have $R_n = s_n + R_{n+1}$, which can be used to estimate the value of the remainder in two cases [28].

1. If g and h are real and the products $g_n H_n$ and $g_{n+1} H_{n+1}$ have constant and equal signs on $[a, b]$, then R_n has the same sign as s_n and opposite than R_{n+1}, and $|R_n| \leq |s_n|$. Example:

$$g(\lambda, x) = (1 + \lambda x)^{-1}, \qquad h(x) = \exp(-x),$$

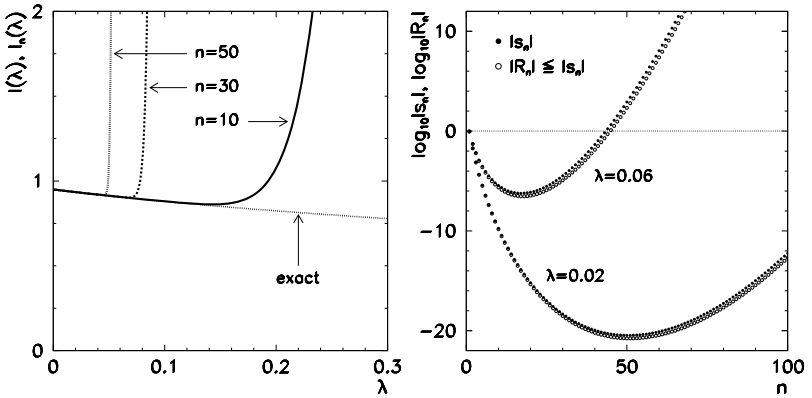


Fig. 1.5 Computation of the $I(\lambda) = \int_a^b \mathrm{e}^{-x}(1+\lambda x)^{-1}\,\mathrm{d}x$ on $[a,b] = [0,3]$ at small λ by asymptotic series. [*Left*] Divergent behavior of partial sums as a function of λ at $n = 10$, 30, and 50. [*Right*] The size of the first omitted term s_n and the remainder R_n as a function of n at $\lambda = 0.02$ and 0.06: we see that $|R_n| \le |s_n|$. (The error $|I(\lambda) - I_n(\lambda)|$ has the same qualitative behavior.) We stop the summation when we reach the minimum in this graph. This point also determines the smallest error one can achieve at a given λ

where $0 \le (-1)^m R_m \le (-1)^m s_m$. In the case $a = 0$ and $b = \infty$ we are dealing with the Euler integral $I(\lambda) = \int_0^\infty \mathrm{e}^{-x}(1+\lambda x)^{-1}\,\mathrm{d}x$, which can be represented as a finite alternating series and the remainder

$$I(\lambda) = \sum_{k=0}^{n} (-1)^k k!\,\lambda^k + (-\lambda)^{n+1}(n+1)! \int_0^\infty \frac{\mathrm{e}^{-x}}{(1+\lambda x)^{n+2}}\,\mathrm{d}x,$$

and which is closely related to the exponential integral Ei (see Fig. 1.5).

2. If g is real, $|H_{n+1}|$ an increasing function of x, and both g_n and g_{n+1} have constant and equal signs on $[a,b]$, or if g is real, $|H_{n+1}|$ is a decreasing function of x, and both g_n and g_{n+1} have constant but opposite signs on $[a,b]$, we have $|R_n| \le 2|s_n|$.

If λ is complex, we rename $x \mapsto x/\lambda$; thus $g(x) = (1+x)^{-1}$ and $h(\lambda, x) = \lambda^{-1}\exp(-x/\lambda)$. Then for $\mathrm{Re}\,\lambda > 0$, the functions g_n, g_{n+1} and H_{n+1} correspond to the second criterion of item 2 above, and we have $|R_n| \le 2|s_n|$. At any rate, the remainder R_n is on the order of the first omitted term, $R_n = \mathcal{O}(s_n)$.

The approach described above is particularly useful when h can be integrated easily, like in the case of $h(\lambda, x) = \exp(\lambda x)$ when $H_m(\lambda, x) = h(x)/\lambda^m$. This is the foundation of the asymptotic expansion of Laplace and Fourier integrals

$$L(\lambda) = \int_a^b \mathrm{e}^{-\lambda x}\phi(x)\,\mathrm{d}x, \qquad F(\lambda) = \int_a^b \mathrm{e}^{\mathrm{i}\lambda x}\phi(x)\,\mathrm{d}x,$$

in the limit $\lambda \to \pm\infty$, which are hard to compute by other means. The asymptotic expansion of the Fourier integral is

$$F_n(\lambda) = \sum_{k=0}^{n-1} \left(\frac{\mathrm{i}}{\lambda}\right)^{k+1} \left[\mathrm{e}^{\mathrm{i}\lambda a}\phi^{(k)}(a) - \mathrm{e}^{\mathrm{i}\lambda b}\phi^{(k)}(b)\right] + R_n(\lambda).$$

The remainder after the truncation of the series to n terms,

$$R_n(\lambda) = \left(\frac{\mathrm{i}}{\lambda}\right)^n \int_a^b \phi^{(n)}(x)\mathrm{e}^{\mathrm{i}\lambda x}\,\mathrm{d}x,$$

has an upper limit when dealing with finite intervals $[a, b]$. By integrating the remainder by parts one more time, we get [28]

$$\left|R_n(\lambda)\right| \le \lambda^{-n-1}\left[\left|\phi^{(n)}(a)\right| + \left|\phi^{(n)}(b)\right| + \int_a^b \left|\phi^{(n+1)}(x)\right|\mathrm{d}x\right] = \mathcal{O}\left(\lambda^{-n-1}\right).$$

1.3.4 Asymptotic Analysis of Integrals by the Laplace Method

Here we analyze integrals of the form

$$I(\lambda) = \int_a^b \phi(x)\mathrm{e}^{-\lambda h(x)}\,\mathrm{d}x \tag{1.28}$$

in the limit of large positive λ, where ϕ and h are real functions of a real variable x. Assume that h has a global minimum at one of the internal points ξ of the interval $[a, b]$, thus $h'(\xi) = 0$ and $h''(\xi) > 0$. Therefore $\exp(-\lambda h(x))$ reaches its maximum at ξ and in its vicinity we may expect the largest contribution to the integral (1.28). We expand h in the Taylor series around ξ,

$$h(x) = h(\xi) + \frac{1}{2}h''(\xi)(x-\xi)^2 + \mathcal{O}\left((x-\xi)^3\right), \tag{1.29}$$

while we take simply $\phi(x) \approx \phi(\xi)$. When these are used in the integral (1.28) and its integration limits are extended to $[-\infty, +\infty]$, we obtain

$$I(\lambda) \approx \int_a^b \phi(\xi)\,\mathrm{e}^{-\lambda[h(\xi)+h''(\xi)(x-\xi)^2/2]}\,\mathrm{d}x \approx \phi(\xi)\,\mathrm{e}^{-\lambda h(\xi)} \int_{-\infty}^{\infty} \mathrm{e}^{-\lambda h''(\xi)x^2/2}\,\mathrm{d}x.$$

This is the Laplace approximation, which is the leading term in the asymptotic expansion

$$I(\lambda) = \mathrm{e}^{-\lambda h(\xi)}\left[\phi(\xi)\sqrt{\frac{2\pi}{\lambda h''(\xi)}} + \mathcal{O}\left(\frac{1}{\lambda}\right)\right], \quad \lambda \to \infty. \tag{1.30}$$

If h reaches its maximum only at $x = a$ and $h'(a) > 0$, the leading term in the asymptotic expansion becomes

$$I(\lambda) = \mathrm{e}^{-\lambda h(a)} \left[\frac{\phi(a)}{h'(a)} \frac{1}{\lambda} + \mathcal{O}\left(\frac{1}{\lambda^2} \right) \right], \quad \lambda \to \infty, \tag{1.31}$$

while if it reaches its minimum only at $x = b$ and $h'(b) < 0$, we have

$$I(\lambda) = \mathrm{e}^{-\lambda h(b)} \left[-\frac{\phi(b)}{h'(b)} \frac{1}{\lambda} + \mathcal{O}\left(\frac{1}{\lambda^2} \right) \right], \quad \lambda \to \infty. \tag{1.32}$$

The asymptotics of the integrals of the form (1.28), where the dominant contributions originate in the minima of h, is given by expressions (1.30)–(1.32) [27]. Similar formulas can be derived for the case where $\exp(\lambda h(x))$ appears in the integral (1.28) instead of $\exp(-\lambda h(x))$.

Example We seek the leading term in the asymptotic expansion of the integral

$$I(\lambda) = \int_0^1 \frac{\exp(-\lambda[1 + x(1-x)])}{\sqrt{x^2+1}} \, \mathrm{d}x, \quad \lambda \to \infty.$$

In this case $h(x) = 1 + x(1-x)$ and $\phi(x) = 1/\sqrt{x^2+1}$. The function h reaches its minimum at both extreme points of the interval, $x = a = 0$ and $x = b = 1$, at which $h(a) = h(b) = 1$, $h'(a) = 1$, $h'(b) = -1$, $\phi(a) = 1$ and $\phi(b) = 1/\sqrt{2}$. The asymptotic expansion is therefore given by the sum of (1.31) and (1.32):

$$I(\lambda) = \mathrm{e}^{-\lambda} \left[\left(1 + \frac{1}{\sqrt{2}} \right) \frac{1}{\lambda} + \mathcal{O}\left(\frac{1}{\lambda^2} \right) \right], \quad \lambda \to \infty.$$

As an exercise, see how the integral behaves in the limit $\lambda \to -\infty$.

For better Laplace approximations, we need a more general expansion [29]. Assume that h has only one minimum on the interval $[a, b]$, at $x = a$; otherwise, we split the whole interval on suitable subintervals. Assume that h in the vicinity of $x = a$ (in limit $x \searrow a$) can be represented as

$$h(x) \sim h(a) + \sum_{s=0}^{\infty} a_s (x-a)^{s+\alpha}, \tag{1.33}$$

where $\alpha \in \mathbb{R}$ and $\alpha > 0$, $a_0 \neq 0$, while ϕ can be represented as

$$\phi(x) \sim \sum_{s=0}^{\infty} b_s (x-a)^{s+\beta-1}, \tag{1.34}$$

where $b_0 \neq 0$ and we require $\mathrm{Re}\, \beta > 0$ for the constant $\beta \in \mathbb{C}$. Let h' and ϕ be continuous around $x = a$ (except perhaps at $x = a$ itself). Under these assumptions, if

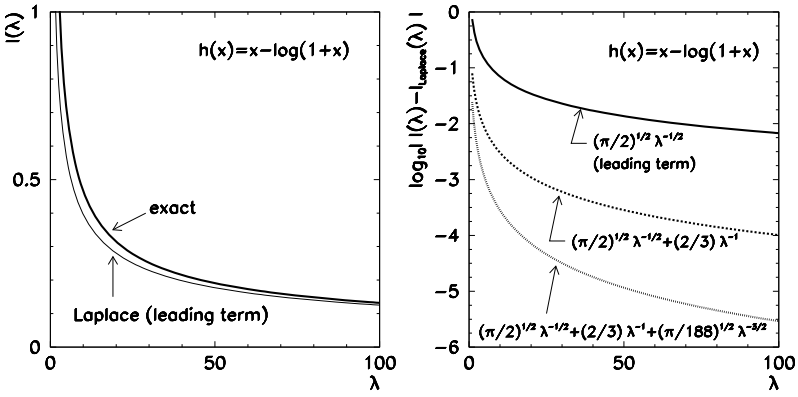


Fig. 1.6 Asymptotic behavior of $I(\lambda) = \int_0^\infty \phi(x)\exp[-\lambda h(x)]\,\mathrm{d}x$, where $\phi(x) = 1$ and $h(x) = x - \log(1+x)$. [*Left*] Exact dependence on λ and the leading Laplace approximation. [*Right*] The error of the approximation. The function h has a minimum at $a = 0$, where it can be expanded as $h(x) = x^2/2 - x^3/3 + x^4/4 - \cdots$. By comparison with (1.33) and (1.34) we get $\alpha = 2$, $a_s = (-1)^s/(s+2)$, $\beta = 1$, $b_0 = 1$, and $b_s = 0$, $s \geq 1$. From here we get the coefficients $c_0 = 1/\sqrt{2}$, $c_1 = 2/3$, and $c_2 = \sqrt{2}/12$ of (1.35)

the integral $I(\lambda)$ absolutely converges for all large enough λ, we have the asymptotic expansion

$$I(\lambda) \sim \mathrm{e}^{-\lambda h(a)} \sum_{s=0}^{\infty} \Gamma\left(\frac{s+\beta}{\alpha}\right) \frac{c_s}{\lambda^{(s+\beta)/\alpha}}, \quad \lambda \to \infty. \tag{1.35}$$

The coefficients c_s in the expansion (1.35) can be expressed by the coefficients a_k and b_k for $k \leq s$. Here we write the first three,

$$\begin{aligned} c_0 &= \frac{b_0}{\alpha a_0^{\beta/\alpha}}, \\ c_1 &= \left[\frac{b_1}{\alpha} - \frac{(\beta+1)a_1 b_0}{\alpha^2 a_0}\right] a_0^{-(\beta+1)/\alpha}, \\ c_2 &= \left[\frac{b_2}{\alpha} - \frac{(\beta+2)a_1 b_1}{\alpha^2 a_0} + \left\{(\alpha+\beta+2)a_1^2 - 2\alpha a_0 a_2\right\}\frac{(\beta+2)b_0}{2\alpha^3 a_0^2}\right] a_0^{-(\beta+2)/\alpha}, \end{aligned} \tag{1.36}$$

while the procedure to compute any c_k can be found in [29] and [30]. The estimate of the error due to the truncation of (1.35) to a finite number of terms is discussed by [31] in Sect. 3.9. The general Laplace method is illustrated in Fig. 1.6 and by the following example.

Example Let us revisit the calculation of the average interaction energy of two electric dipoles and its asymptotic behavior at low temperatures (1.19). We use the

Laplace method to analyze the integral

$$K(\lambda) = \frac{8\pi}{\sqrt{3}\lambda} \int_1^2 \frac{\sinh \lambda x}{\sqrt{x^2-1}}\,\mathrm{d}x = \frac{4\pi}{\sqrt{3}\lambda}\Bigg[\underbrace{\int_1^2 \frac{\mathrm{e}^{\lambda x}}{\sqrt{x^2-1}}\,\mathrm{d}x}_{I_1(\lambda)} - \underbrace{\int_1^2 \frac{\mathrm{e}^{-\lambda x}}{\sqrt{x^2-1}}\,\mathrm{d}x}_{I_2(\lambda)}\Bigg]$$

in the limit $\lambda = \mu_1\mu_2/(4\pi\varepsilon_0 r^3 kT) \to \infty$. By using $x \mapsto -x$ the first term can be rewritten as

$$I_1(\lambda) = \int_1^2 \frac{\mathrm{e}^{\lambda x}}{\sqrt{x^2-1}}\,\mathrm{d}x = \int_{-2}^{-1} \frac{\mathrm{e}^{-\lambda x}}{\sqrt{x^2-1}}\,\mathrm{d}x,$$

so that $h(x) = x$ and $\phi(x) = 1/\sqrt{x^2-1}$. The function h has a minimum at $x = a = -2$, as required by the assumptions for the expansion (1.35). Since h is so simple, its expansion (1.33) has only two terms,

$$h(x) = x = h(a) + \sum_{s=0}^{\infty} a_s (x-a)^{s+\alpha} = -2 + a_0\big(x-(-2)\big)^{0+\alpha} + 0 + 0 + \cdots,$$

from which we read off $\alpha = 1$, $a_0 = 1$, and $a_s = 0$ for $s \geq 1$. We expand ϕ around a in the Taylor series and compare it to the expansion (1.34),

$$\phi(x) = \frac{1}{\sqrt{x^2-1}} = \frac{1}{\sqrt{3}} + \frac{2(x-a)}{3\sqrt{3}} + \frac{(x-a)^2}{2\sqrt{3}} + \cdots = \sum_{s=0}^{\infty} b_s (x-a)^{s+\beta-1},$$

from which we infer $\beta = 1$, $b_0 = 1/\sqrt{3}$, $b_1 = 2/(3\sqrt{3})$, and $b_2 = 1/(2\sqrt{3})$. By using the coefficients a_s and b_s we compute c_s by (1.36) and use them in the expansion (1.35). We get $c_0 = b_0$, $c_1 = b_1$, $c_2 = b_2$. Finally, we have $-\lambda h(a) = 2\lambda$, thus

$$I_1(\lambda) = \mathrm{e}^{2\lambda}\left[\frac{1}{\sqrt{3}\lambda} + \frac{2}{3\sqrt{3}\lambda^2} + \frac{1}{\sqrt{3}\lambda^3} + \cdots\right],$$

from which (1.19) follows. The integral $I_2(\lambda)$ is already in the appropriate form, with the function $h(x) = x$ reaching its minimum at $x = a = 1$. Since $I_2(\lambda)$ behaves like $\mathrm{e}^{-\lambda}$ it is negligible in comparison to $I_1(\lambda)$ in the limit $\lambda \to \infty$.

1.3.5 Stationary-Phase Approximation

The method of *stationary-phase approximation* is used to deduce the asymptotic series for integrals of the form

$$I(\lambda) = \int_a^b \phi(x)\,\mathrm{e}^{\mathrm{i}\lambda h(x)}\,\mathrm{d}x, \tag{1.37}$$

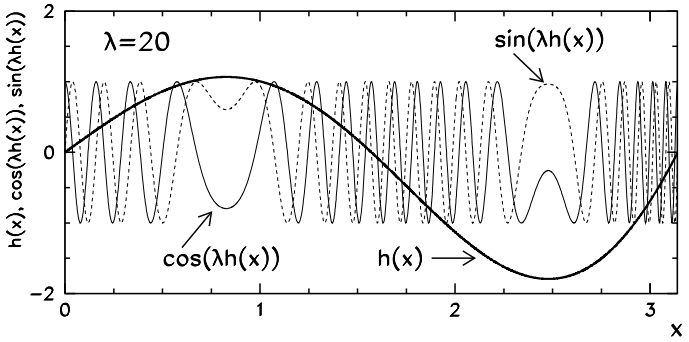


Fig. 1.7 The basic idea of stationary-phase approximation. At large λ the function $\exp(\mathrm{i}\lambda h(x))$ in (1.37) rapidly oscillates around zero. In the regions of large variations of $h(x)$ the contributions to the integral therefore largely cancel out, while there is much less cancellation where the variation is small (near maxima and minima)

where h is a real function of a real variable x. The integrand thus contains the exponential function with an imaginary argument, and we are interested in the behavior of the integral at $|\lambda| \gg 1$. In order to compute $I(\lambda)$ for negative arguments, we use the symmetry $I(\lambda)^\star = I(-\lambda)$.

The approximation can be established by realizing that the leading contribution to $I(\lambda)$ for $\lambda \to \infty$ comes from the integral over the points at which the *phase function h is stationary*, that is, $h'(x) = 0$. Assume that h has only one minimum ($h''(\xi) > 0$) or one maximum ($h''(\xi) < 0$) on the interval $[a, b]$, at ξ. We insert the expansion (1.29) into (1.37) and obtain

$$I(\lambda) \approx \int_a^b \phi(\xi)\, \mathrm{e}^{\mathrm{i}\lambda[h(\xi)+h''(\xi)(x-\xi)^2/2]}\, \mathrm{d}x = \phi(\xi)\, \mathrm{e}^{\mathrm{i}\lambda h(\xi)} \int_a^b \mathrm{e}^{\mathrm{i}\lambda h''(\xi)(x-\xi)^2/2}\, \mathrm{d}x.$$

We extend the integral on the right to the whole real axis and obtain the leading term of the stationary-phase approximation:

$$I(\lambda) \sim \phi(\xi) \sqrt{\frac{2\pi}{\lambda |h''(\xi)|}}\, \exp\left\{\mathrm{i}\left[\lambda h(\xi) + \frac{\pi}{4}\, \mathrm{sign}\big(h''(\xi)\big)\right]\right\}, \tag{1.38}$$

where we have assumed $h''(\xi) \neq 0$ and used $\int_{-\infty}^{\infty} \mathrm{e}^{\mathrm{i}x^2}\, \mathrm{d}x = \sqrt{\pi}\, \mathrm{e}^{\mathrm{i}\pi/4}$.

Example We are interested in the leading asymptotic term of the integral

$$I(\lambda) = \int_0^\pi \phi(x)\, \mathrm{e}^{\mathrm{i}\,\lambda h(x)}\, \mathrm{d}x, \quad h(x) = \sin(2x)\, \mathrm{e}^{x^2/10},\ \phi(x) = \frac{1}{\sqrt{x^2+1}},$$

in the limit $\lambda \to \infty$. On $[0, \pi]$, the function h has a maximum at $\xi_1 \approx 0.8266$ and a minimum at $\xi_2 \approx 2.4776$ (see Fig. 1.7). At these points, $h''(\xi_1) \approx -4.0841$ and $h''(\xi_2) \approx 7.2550$. The asymptotics of the integral is determined by two contributions

of the form (1.38) which, at any $\lambda \gg 1$, are computed for $\xi = \xi_1$ and $\xi = \xi_2$, and then summed. With $\lambda = 20$, for example, we obtain

$$I(20) \approx (-0.0487 + 0.3566\,\mathrm{i}) + (-0.4814 + 0.2781\,\mathrm{i}) = -0.5301 + 0.6347\,\mathrm{i},$$

while $I(20) \approx -0.5290 + 0.6280\,\mathrm{i}$ by precise numerical integration. The leading-order stationary-phase approximation to the complex value of the integral thus leads to the error in the modulus of $\approx$0.5 % and in the phase of $\approx 0.2°$.

Higher terms of the stationary-phase approximation can be obtained by the generalization of this method [29]. Assume that h has a finite number of stationary points (zeros of $h'(x)$) on the integration interval. We split this interval into subintervals such that there is one stationary point at the lower edge of every subinterval. Let $[a, b]$ be such a subinterval, and let h be monotonously increasing on $[a, b]$, thus $h'(x) > 0$ for $x \in [a, b]$ (in the opposite case, we transform $x \mapsto -x$). Assume that h has the form

$$h(x) = h(a) + (x - a)^{\alpha} h_1(x), \quad h_1(a) \neq 0,$$

where h_1 is smooth on $[a, b]$ and $\alpha \geq 1$. Let ϕ have the form

$$\phi(x) = (x - a)^{\beta - 1} \phi_1(x),$$

where ϕ_1 is smooth on $[a, b]$ and $\beta \in (0, 1]$. Then the asymptotic expansion of the integral (1.37) in the limit $\lambda \to \infty$ is

$$I(\lambda) = \mathrm{e}^{\mathrm{i}\lambda h(a)} \left[\sum_{n=0}^{N-1} a_n \left(\frac{\mathrm{i}}{\lambda} \right)^{(n+\beta)/\alpha} \right] + R_N^{(1)} - \mathrm{e}^{\mathrm{i}\lambda h(b)} \left[\sum_{n=0}^{M-1} b_n \left(\frac{\mathrm{i}}{\lambda} \right)^{n+1} \right] + R_M^{(2)},$$

where $R_N^{(1)} = \mathcal{O}(\lambda^{-(N+\beta)/\alpha})$ and $R_M^{(2)} = \mathcal{O}(\lambda^{-M})$ are the remainders due to the truncation of the series (see [29] for details). By introducing new variables $t^{\alpha} = h(x) - h(a)$ the coefficients a_n can be determined as

$$a_n = \frac{1}{\alpha n!} \Gamma\left(\frac{n + \beta}{\alpha} \right) \left(\frac{\mathrm{d}}{\mathrm{d}t} \right)^n \left[\left(\frac{x - a}{t} \right)^{\beta - 1} \phi_1(x) \frac{\mathrm{d}x}{\mathrm{d}t} \right] \Bigg|_{t=0},$$

and the coefficients b_n as

$$b_n = \left(\frac{1}{h'(x)} \frac{\mathrm{d}}{\mathrm{d}x} \right)^n \left[\frac{\phi(x)}{h'(x)} \right] \Bigg|_{x=b}.$$

Only few instances of functions h and ϕ allow for a simple calculation of the coefficients a_n and b_n. We typically let this work be done by programs for symbolic computation like MATHEMATICA [32] (routine `InverseSeries`). In connection to the integration of rapidly oscillating functions see also Sect. E.2.

General integrals along a contour $\mathcal{C}$ in the complex plane

$$I(\lambda) = \int_{\mathcal{C}} g(z)\, \mathrm{e}^{\lambda f(z)}\, \mathrm{d}z, \quad \lambda \to \infty,$$

where f and g are analytic, can be computed by means of the *method of steepest descent* and by the *saddle-point method*, which are both similar to the Laplace method in spirit, but technically more complicated. For further information, we refer the reader to [29] and [33].

1.3.6 Differential Equations with Large Parameters

Asymptotic approaches are also applicable to the analysis of differential equations. For a physicist, second-order homogeneous equations

$$y''(x) + p(x,\lambda)y'(x) + q(x,\lambda)y(x) = 0 \tag{1.39}$$

in the limit $\lambda \to \infty$ may be particularly relevant. By using the ansatz $y(x) = z(x)\exp(-\frac{1}{2}\int p(x,\lambda)\,\mathrm{d}x)$ (1.39) can be put into the standard form

$$z''(x) + h(x,\lambda)z(x) = 0, \quad h(x,\lambda) = q(x,\lambda) - \frac{1}{2}p'(x,\lambda) - \frac{1}{4}p^2(x,\lambda). \tag{1.40}$$

Assume that $h(x,\lambda)$ has the Laurent expansion

$$h(x,\lambda) = \lambda^{2k}\sum_{n=0}^{\infty} h_n(x)\lambda^{-n}, \tag{1.41}$$

where k is a positive integer and $h_0 \neq 0$. By using this expansion, a large class of problems can be treated, in spite of the seemingly restrictive character of the leading term $\sim \lambda^{2k}$ in (1.41). Equation (1.39) has two types of solutions [28].

The First Type of the Solution has the form

$$z(x,\lambda) = A(x,\lambda)\,\mathrm{e}^{S(x,\lambda)}, \tag{1.42}$$

where we have introduced the amplitude function $A(x,\lambda)$ and the action function $S(x,\lambda)$. They are defined as series in the parameter λ:

$$A(x,\lambda) = \sum_{n=0}^{\infty} a_n(x)\lambda^{-n}, \qquad S(x,\lambda) = \lambda^k \sum_{n=0}^{k-1} b_n(x)\lambda^{-n}. \tag{1.43}$$

When we insert (1.42) in (1.40) and collect the terms with powers λ^{2k-n}, we get

$$\left(b_0'\right)^2 + h_0 = 0, \tag{1.44}$$

$$2b_0'b_m' + h_m + \sum_{n=1}^{m-1} b_n'b_{m-n}' = 0, \quad m = 1, 2, \ldots, k-1, \tag{1.45}$$

where $'$ denotes the derivative with respect to x. This is a system of differential equations for the coefficient functions b_n of the action $S(x, \lambda)$. Since $h_0 \neq 0$, squaring in (1.44) implies two possible signs for the derivative of the leading coefficient, $b_0' = \pm\sqrt{-h_0}$. These possibilities correspond to two linearly independent solutions of (1.40), as expected for a second-order equation. We then use the computed b_n in the equations for the coefficient functions a_n:

$$\begin{aligned} 2a_0'b_0' + a_0\left(b_0'' + h_k + \sum_{n=1}^{k-1} b_n'b_{k-n}'\right) &= 0, \\ 2a_n'b_0' + \sum_{m=0}^{n} a_{n-m}A_m + 2\sum_{m=1}^{n} a_{n-m}'b_m' + a_{n-k}'' &= 0, \quad n = 1, 2, \ldots. \end{aligned} \tag{1.46}$$

The functions h_n, a_n, and b_n are zero if the subscripts are outside of their ranges required by (1.41) and (1.43), thus $h_{-n} = a_{-n} = b_{-n} = b_{k-1+n} = 0$ for $\forall n \in \mathbb{N}$. We have also introduced

$$A_m = b_m'' + h_{k+m} + \sum_{l=m+1}^{k-1} b_l'b_{k+m-l}'.$$

The Second Type of the Solution of (1.40) has the form

$$z(x, \lambda) = \mathrm{e}^{Z(x,\lambda)}, \quad Z(x, \lambda) = \lambda^k \sum_{n=0}^{\infty} c_n(x)\lambda^{-n}, \tag{1.47}$$

where k is a positive integer. When the ansatz (1.47) is used in (1.40) and the terms with equal powers of λ are combined, we obtain

$$\left(c_0'\right)^2 + h_0 = 0, \tag{1.48}$$

$$2c_0'c_n' + h_n + \sum_{m=1}^{n-1} c_m'c_{n-m}' = 0, \quad n = 1, 2, \ldots, k-1, \tag{1.49}$$

$$2c_0'c_n' + h_n + \sum_{m=1}^{n-1} c_m'c_{n-m}' + c_{n-k}'' = 0, \quad n = k, k+1, \ldots, \tag{1.50}$$

while the functions c_{-n} vanish for $\forall n \in \mathbb{N}$. By determining the phase of the leading term (the derivative of which has two possible dependencies, $c_0' = \pm\sqrt{-h_0}$) this procedure yields two linearly independent solutions of (1.40).

For subscripts $0 \leq n \leq k-1$ the system of equations (1.44) and (1.45) for the functions b_n is the same as the system (1.48) and (1.49) for the functions c_n, so

$b_n = c_n$ for $0 \le n \le k-1$. By comparing (1.42) to (1.47) it becomes clear that the amplitude function of the first-type solution is just a formal expansion of the remainder of the action of the second-type solution,

$$A(x,\lambda) = \exp\left(\sum_{n=k}^{\infty} c_n(x)\lambda^{k-n}\right),$$

with functions $\{c_n\}_{n=k}^{\infty}$ determined by (1.50).

We have assumed that $h(x,\lambda)$ as a function of λ has a pole of even degree at infinity, $h(x,\lambda) = \mathcal{O}(\lambda^{2k})$. If the pole has an odd degree, $h(x,\lambda) = \mathcal{O}(\lambda^k)$, the solutions (1.42) and (1.47) are not valid. In this case we can introduce a new asymptotic parameter $\lambda' = \lambda^{1/2}$ and use λ' in the formulas derived above.

The procedure described here is called the WKB (Wentzel–Kramers–Brillouin) method. The special case $k = 1$ coincides with the problems of the Schrödinger equation for a particle of mass m in an one-dimensional potential V,

$$-\frac{\hbar^2}{2m}\frac{\partial^2}{\partial x^2}\psi(x) + \big[V(x) - E\big]\psi(x) = 0. \tag{1.51}$$

In the limit of large energies $E \to \infty$ or small Planck constant $\hbar \to 0$ we speak of a *semi-classical* approach. The asymptotic parameter λ then represents high energies $E = \lambda^2$ or the smallness of $\hbar = \lambda^{-1}$. The WKB method is illustrated by the following example adapted from [27]; for details, see [34, 35].

Example The classical example of the WKB method in quantum mechanics is the calculation of the particle's wave-function in a space with linear potential [34]. In (1.51) this means $V(x) = \lambda^2 x$, and it can be rewritten as

$$z''(x) - \lambda^2 x z(x) = 0 \tag{1.52}$$

by a suitable change of variables. This equation is of the form (1.40) with $h(x,\lambda) = -\lambda^2 x$, but let us pretend for a moment that h is still general and has the expansion (1.41). To solve (1.52), we use the ansatz (1.42). First, we determine the leading coefficient of the action, b_0. From (1.44) it follows that

$$b_0'(x) = \pm\sqrt{-h_0(x)}, \qquad b_0''(x) = \mp\frac{h_0'(x)}{2\sqrt{-h_0(x)}}, \qquad b_0(x) = \pm\int_{x^\star}^{x}\sqrt{-h_0(t)}\,\mathrm{d}t.$$

The lower integration limit $x^\star$ will be determined in the following. We compute the coefficient a_0 of the amplitude function (1.43) by using (1.46), $2a_0'b_0' + a_0b_0'' + a_0h_1 = 0$. This is a differential equation for a_0:

$$\frac{\mathrm{d}a_0}{a_0} = \left(-\frac{1}{4}\frac{h_0'}{h_0} \mp \frac{h_1}{2\sqrt{-h_0}}\right)\mathrm{d}x.$$

We use $h_0'/h_0 = (\log h_0)'$, and obtain

$$a_0(x) = \frac{1}{[h_0(x)]^{1/4}} \exp\left\{\mp\frac{1}{2}\int_{x^\star}^{x} \frac{h_1(t)}{\sqrt{-h_0(t)}}\,\mathrm{d}t\right\}.$$

Since $k = 1$, the series (1.43) for $S(x, \lambda)$ contains only the term $\lambda^1 b_0 \lambda^0 = \lambda b_0$. The final structure of the solution to the leading order in λ is therefore simply $z(x, \lambda) = a_0(x)\exp\{\lambda b_0(x)\}$, but its precise form still depends on the sign of the coefficient function h_0 from the expansion (1.41).

The point $x^\star$, in which h_0 has a simple zero, ($h_0(x^\star) = 0$, $h_0'(x^\star) \neq 0$), is called the *turning* or *transition point*, since the physical character of the solution changes at this point. In the region where $h_0 > 0$, the expressions written above yield two linearly independent oscillatory solutions

$$z^{\pm}_{\mathrm{osc}}(x, \lambda) \approx \frac{1}{[h_0(x)]^{1/4}} \exp\left\{\pm\mathrm{i}\,\lambda \int_{x^\star}^{x} \sqrt{h_0(t)}\,\mathrm{d}t \pm \frac{\mathrm{i}}{2}\int_{x^\star}^{x} \frac{h_1(t)}{\sqrt{h_0(t)}}\,\mathrm{d}t\right\},$$

while in the region with $h_0 < 0$ we get exponentially increasing or decreasing solutions

$$z^{\pm}_{\mathrm{exp}}(x, \lambda) \approx \frac{1}{[|h_0(x)|]^{1/4}} \exp\left\{\pm\lambda \int_{x^\star}^{x} \sqrt{|h_0(t)|}\,\mathrm{d}t \mp \frac{1}{2}\int_{x^\star}^{x} \frac{h_1(t)}{\sqrt{|h_0(t)|}}\,\mathrm{d}t\right\}.$$

In order for the WKB analysis to be valid, some authors require that the whole function h, not just its leading term h_0, should have a zero at the turning point. It turns out that it is very hard to formulate an asymptotic analysis of the WKB type if h has a zero in the region being discussed while h_0 does not. The explicit demand that h_0 has a simple zero can thus be understood as a necessary condition for the applicability of the WKB method.

Let us reconsider (1.52). From (1.41) we read off $h_0(x) = -x$ and $h_n(x) = 0$ for $n \geq 1$. In the exponents of $z^{\pm}_{\mathrm{osc}}(x, \lambda)$ and $z^{\pm}_{\mathrm{exp}}(x, \lambda)$ only the first term appears, and the turning point is $x^\star = 0$. In its vicinity the WKB approximation fails (see Fig. 1.8). The solution of (1.52) in the WKB approximation to the left of the turning point ($x < x^\star$ and $h_0(x) > 0$) is a linear combination of the solutions z^{+}_{osc} and z^{-}_{osc}. The solution on the right ($x > x^\star$ and $h_0(x) < 0$) is a linear combination of z^{+}_{exp} and z^{-}_{exp}.

The method described above can be generalized to the case when h has a zero $x^\star$ of degree p. Assume that h is analytic at $x^\star$ and that in the vicinity of $x^\star$, in the limit $\lambda \to \infty$, it has the asymptotic expansion

$$h(x, \lambda) \sim C\,\lambda^{2k}\big(x - x^\star\big)^p, \quad C \in \mathbb{R}. \tag{1.53}$$

By substitution $x - x^\star = |C\,\lambda^{2k}|^{-1/(2+p)}t$ and by using (1.53) we rewrite (1.40) as

$$\frac{\mathrm{d}^2 z(t)}{\mathrm{d}t^2} + s t^p z(t) = 0, \quad s = \mathrm{sign}(C).$$

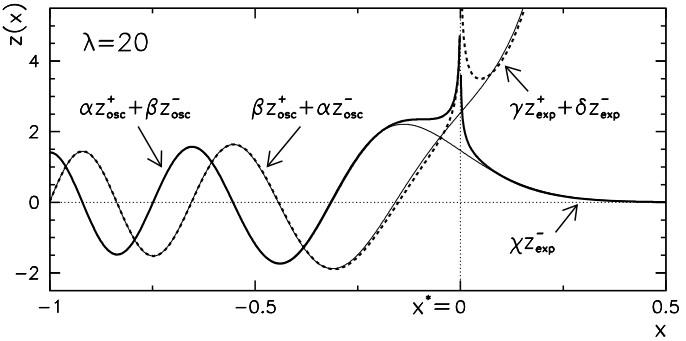


Fig. 1.8 The solution of $z''(x) - \lambda^2 x z(x) = 0$ with $\lambda = 20$ in the WKB approximation. The exact solutions (*thin lines*) are given by the Airy functions Ai and Bi (Problem 1.5.2). The coefficients α, β, γ, δ and χ in the linear combinations $z^{\pm}_{\rm osc}$ and $z^{\pm}_{\rm exp}$ are determined such that the WKB solutions match the exact solutions far from the turning point $x^\star = 0$

This equation is valid near $t = 0$. Its solutions are known [36] and can be expressed in terms of the Bessel functions of the first and second kind [22] as

$$z(t) = \sqrt{t} \begin{cases} C_1 J_{1/2q}(t^q/q) + C_2 Y_{1/2q}(t^q/q); & s = +1, \\ C_1 I_{1/2q}(t^q/q) + C_2 K_{1/2q}(t^q/q); & s = -1, \end{cases}$$

where $q = \frac{1}{2}(p+2)$. In seeking the solutions over a larger region, the constants C_1 and C_2 can be determined such that the solution near the turning point matches the solution far from the turning point in amplitude and phase.

For details on the formulation and use of asymptotic series see [28] and [29]. Connections of asymptotic series to special functions are discussed in the classic work [31].

1.4 Summation of Finite and Infinite Series

Physical quantities are often represented as infinite or finite series

$$S = \sum_{k=0}^{\infty} a_k, \qquad S_n = \sum_{k=0}^{n} a_k, \quad a_k \in \mathbb{R} \text{ or } a_k \in \mathbb{C}.$$

The sum S_n of the first $n+1$ terms of S is the *nth partial sum* of S. The series S *converges* if the sequence $\{S_n\}$ converges, i.e. if for any $\varepsilon > 0$ a $\kappa \in \mathbb{N}$ can be found such that for each $p \in \{0\} \cup \mathbb{N}$ we have $n > \kappa \implies |S_{n+p} - S_n| < \varepsilon$. A convergent sequence converges to a finite limit and this limit is its one and only cluster point. The series S is said to *diverge* if the sequence $\{S_n\}$ has a limit at infinity, has multiple cluster points or has no cluster points at all. Sometimes we carelessly interpret divergence as "convergence" to infinity.

General properties of series and summation methods are treated by the theory of summability [37, 38]. Further reading on modern techniques of symbolic summation of series to closed forms can be found in [39, 40].

1.4.1 Tests of Convergence

Tests of convergence are procedures used to identify sufficient conditions for convergence of infinite series $\sum_{k=0}^{\infty} a_k$. In many tests, we disregard the signs of the terms (if $a_k \in \mathbb{R}$) or their phases (if $a_k \in \mathbb{C}$) and only use their absolute values. This simplification is based on the Cauchy inequality $|\sum_k a_k| \leq \sum_k |a_k|$, from which we infer that a series converges if the corresponding series with absolute values of terms converges (*absolute convergence*). The necessary condition for the convergence of any series is $\lim_{k\to\infty} a_k = 0$.

Comparison Test For a given sequence $\{a_k\}_{k\in\mathbb{N}_0}$, where all $a_k \geq 0$, we find a sequence $\{b_k\}_{k\in\mathbb{N}_0}$. If there exists a $N \in \mathbb{N}_0$ such that $0 \leq a_k \leq b_k$ for all $k > N$ and the series $\sum_k b_k$ converges, the series $\sum_k a_k$ also converges. If at some $N \in \mathbb{N}_0$ we find $0 \leq b_k \leq a_k$ for all $k > N$ and the series $\sum_k b_k$ diverges, then the series $\sum_k a_k$ also diverges.

Quotient and Cauchy Square-Root Test In the quotient test we observe the upper limit of the quotient of the consecutive terms of the series,

$$\rho = \limsup_{k\to\infty} \left| \frac{a_{k+1}}{a_k} \right|,$$

while in the square-root test, we look at the upper limit of the square roots,

$$\rho = \limsup_{k\to\infty} |a_k|^{1/k}.$$

The series (absolutely) converges if $\rho < 1$, and (absolutely) diverges if $\rho > 1$. In the case $\rho = 1$ the test is inconclusive (the series may either converge or diverge).

Integral Test Assume we have a sequence $\{a_k\}_{k\in\mathbb{N}}$ where all $a_k \geq 0$, and there exists a continuous monotonously decreasing function f such that $f(k) = a_k$ for all $k \geq 1$. Then the series $\sum_{k=1}^{\infty} a_k$ and the integral $\int_1^{\infty} f(x)\,\mathrm{d}x$ either both converge or both diverge. In the case of convergence, the difference $R_n = S - S_n = \sum_{k=n+1}^{\infty} a_k$ satisfies $\int_{n+1}^{\infty} f(x)\,\mathrm{d}x \leq R_n \leq \int_n^{\infty} f(x)\,\mathrm{d}x$.

Kummer's and Raabe's Test To perform the Kummer's test, we need a sequence $\{a_k\}_{k\in\mathbb{N}_0}$, $a_k > 0$, and a sequence $\{b_k\}_{k\in\mathbb{N}_0}$, $b_k > 0$, from which we form the limit

$$\rho = \lim_{k\to\infty} \left(b_k \frac{a_k}{a_{k+1}} - b_{k+1} \right).$$

The series $\sum_k a_k$ converges if $\rho > 0$, while it diverges if $\rho < 0$ and the series $\sum_k 1/b_k$ diverges. If $\rho = 0$ the criterion is useless. In the case $b_k = k$ we obtain the Raabe's test, where we observe the limit

$$\rho = \lim_{k\to\infty}\left(k\frac{a_k}{a_{k+1}} - k - 1\right) = \lim_{k\to\infty}\left[k\left(\frac{a_k}{a_{k+1}} - 1\right)\right] - 1.$$

The series $\sum_k a_k$ converges if $\rho > 0$, and diverges if $\rho < 0$. In the case $\rho = 0$ the convergence or divergence cannot be ascertained.

Limit Comparison Test To a sequence $\{a_k\}_{k\in\mathbb{N}_0}$, $a_k > 0$, we find a sequence $\{b_k\}_{k\in\mathbb{N}_0}$, $b_k > 0$, such that the limit $\rho = \lim_{k\to\infty} a_k/b_k$ exists. If ρ is finite and $\rho \neq 0$, then both $\sum_k a_k$ and $\sum_k b_k$ either converge or diverge.

Leibniz's Test for Alternating Series An important class of real series is represented by alternating series $\sum_{k=0}^{\infty}(-1)^k a_k$ where $a_k \geq 0$ (the consecutive terms change signs). If a_k decrease monotonically and $\lim_{k\to\infty} a_k = 0$ holds true, the alternating series converges. The remainder can be bounded as $|S - S_n| \leq a_n$.

Most of the enumerated tests are adapted for analytic work, but very often we can also use them numerically to determine with large certainty whether a given series diverges or converges.

We are also interested in the convergence of the power series

$$\sum_{k=0}^{\infty} a_k(z - z_0)^k, \quad a_k, z, z_0 \in \mathbb{C}, \tag{1.54}$$

which is used to describe functions around a point z_0. Let us consider only absolute convergence and define the largest disk (circular region of points z in the complex plane) $\{z : |z - z_0| \leq r\}$, within which the series (1.54) absolutely converges. The disk radius r is the *convergence radius* and can be computed as

$$r = \left(\limsup_{k\to\infty} |a_k|^{1/k}\right)^{-1} \quad \text{or} \quad r = \limsup_{k\to\infty}\left|\frac{a_k}{a_{k+1}}\right|. \tag{1.55}$$

1.4.2 Summation of Series in Floating-Point Arithmetic

In floating-point arithmetic, the summation of real series $\sum_{k=0}^{n} a_k$ implies rounding errors. In particular for series with $n \to \infty$, precision is of utmost importance. Substantial work has been done in the minimization of summation errors (see [8, 41–43]). Here we list three most widely used summation methods that do not require more than $\mathcal{O}(n)$ of operations.

Simple Recursive Summation Assume that we have the values $\{a_k\}_{k=0}^{n}$ and wish to compute their sum $S = \sum_{k=0}^{n} a_k$. Most obviously, this can be accomplished by computing $\hat{S} = (\cdots(((a_0 \oplus a_1) \oplus a_2) \oplus a_3)\cdots \oplus a_{n-1}) \oplus a_n$, in a loop

Input: real numbers $a_0, a_1, \ldots, a_n$
$\hat{S} = a_0$;
for $k = 1$ **step** 1 **to** n **do**
  $\hat{S} = \hat{S} + a_k$;
end
Output: $\hat{S}$ is the numerical sum of numbers a_k

The deviation of the numerical sum $\hat{S}$ from the exact sum S strongly depends on how a_k are sorted. If they are unsorted, we have

$$|S - \hat{S}| \leq \frac{\varepsilon_{\mathrm{M}}}{2} n \sum_{k=0}^{n} |a_k| + \mathcal{O}\big(\varepsilon_{\mathrm{M}}^2\big), \tag{1.56}$$

where ε_{M} is the arithmetic precision (see p. 2) [42]. In simple summation one can therefore expect a loss of up to $\log_{10} n$ significant digits. The estimate for the upper limit of the error (not the error itself) is smallest when the terms are sorted as $|a_k| \leq |a_{k+1}|$. Sorting requires at least $\mathcal{O}(n \log n)$ additional operations.

Kahan's Algorithm A much better procedure to sum a series, by which the effect of rounding errors is greatly diminished, was proposed by Kahan [44]:

Input: real numbers $a_0, a_1, \ldots, a_n$
$\hat{S} = a_0$;
$c = 0$;
for $k = 1$ **step** 1 **to** n **do**
  $y = a_k - c$;
  $t = \hat{S} + y$;
  $c = (t - \hat{S}) - y$; // do not omit brackets
  $\hat{S} = t$;
end
Output: $\hat{S}$ is the numerical sum of numbers a_k

Algebraically, the value of c is zero, but in finite arithmetic it represents a large part of the lost precision when summing $t = \hat{S} + y$. It is added to the sum in the next step and by doing this, it compensates the rounding error from the previous step. The deviation of the numerical sum from the exact one satisfies

$$|S - \hat{S}| \leq \big(\varepsilon_{\mathrm{M}} + \mathcal{O}\big(n\varepsilon_{\mathrm{M}}^2\big)\big) \sum_{k=0}^{n} |a_k|. \tag{1.57}$$

According to (1.57), Kahan's summation is more precise than simple summation for $n\varepsilon_{\mathrm{M}}/2 \leq 1$. In practice, this applies to even larger n (see Fig. 1.9). In the implementation of the algorithm we should make sure that the compiler does not simplify it, since the essence of its strength is hidden in the rules of floating-point arithmetic. In C and C++ variables should be declared `volatile`.

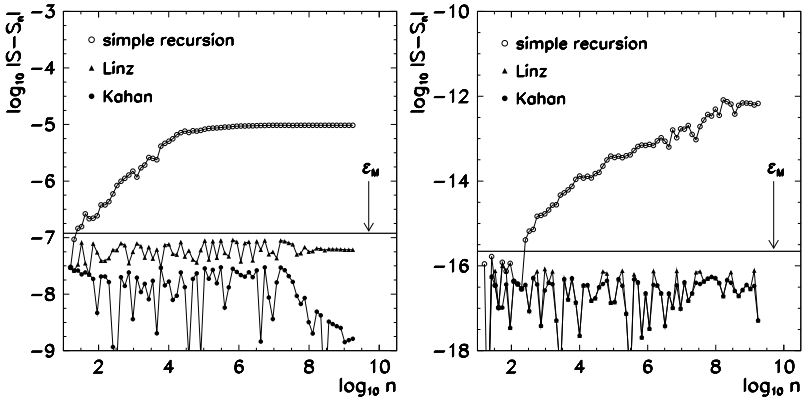


Fig. 1.9 Rounding errors in summing series with many terms. Shown is the absolute error of the numerical partial sums $S_n = \sum_{k=0}^{n} (-1)^k/(k+1)$ with respect to the limiting value $S_\infty = \log 2$. [*Left*] Summation in single-precision arithmetic. [*Right*] Summation in double-precision arithmetic. The horizontal lines correspond to $\varepsilon_M = 1.19 \times 10^{-7}$ (*left*) and $\varepsilon_M = 2.22 \times 10^{-16}$ (*right*)—see p. 2

Recursive Summation of Pairs Summation is an associative and commutative operation between real numbers. Algebraically, the order of summation is thus irrelevant, and this fact is exploited by the Linz procedure [45]. In the first step we sum the consecutive pairs of terms and obtain a new series. In this series we again sum the consecutive pairs and repeat this ($r = \lceil \log_2 n \rceil$)-times, until we are left with only one term, which represents the final sum:

Input: real numbers $a_0, a_1, \ldots, a_{n-1}$, where $n = 2^r$, $r \in \mathbb{N}$
$m = m' = n/2$;
for $k = 0$ **step** 1 **to** $m - 1$ **do**
  $S_{0,k} = a_{2k} + a_{2k+1}$;
end
for $j = 1$ **step** 1 **to** $r - 1$ **do**
  $m' = m'/2$;
  for $k = 0$ **step** 1 **to** $m' - 1$ **do**
    $S_{j,k} = S_{j-1,2k} + S_{j-1,2k+1}$;
  end
end
Output: $\hat{S} = S_{r-1,0}$ is the numerical sum of numbers a_k

Because each term a_k in the sum is touched only r-times, the deviation of the sum $\hat{S}$ from the exact value S is much smaller than in simple recursive summation:

$$|S - \hat{S}| \leq \frac{\varepsilon_M}{2} \log_2 n \sum_{k=0}^{n-1} |a_k|$$

(compare to (1.56)). For the intermediate sums $S_{j,k}$ we need additional computer memory to store $n/2$ real numbers, which is a bit wasteful compared to the simple and Kahan's summation which require only $\mathcal{O}(1)$ of memory. Linz's algorithm can be improved by compensating the numerical error and selecting the pairs in a more intricate manner. For details, consult [46].

In all three methods we specified the upper bounds for $|S - \hat{S}|$; for a given set of numbers $\{a_k\}$, all methods may be equally precise. In general, we recommend Linz's method unless pairs cannot be formed or this does not make much sense (for example, for relatively short series). On the other hand, Kahan's method, which is both simple and precise, never fails to enchant (see Fig. 1.9). For very precise summation, we resort to more sophisticated but slower methods like *distillation algorithms* described in [41, 47, 48].

1.4.3 Acceleration of Convergence

The convergence of the partial sums $S_n = \sum_{k=0}^{n} a_k$ to the limit $S = \lim_{n\to\infty} S_n$ may be slow. By "slow" we mean its leading behavior to be $|S_n - S| = \mathcal{O}(n^{-p})$ (power) or $|S_n - S| = \mathcal{O}((\log n)^{-p})$ (logarithmic) where $p > 0$ is the convergence order. Slow convergence is not desired since it implies large numerical costs and a potential accumulation of rounding errors.

We speak of "fast" convergence when it is better than "slow" according to the definition given above. Ideally, one would like to have exponential (geometric) convergence $|S_n - S| = \mathcal{O}(a^n)$ where $a \in [0, 1)$. In many cases, convergence can be accelerated by transforming the original series into another series that converges more rapidly. In the following, we describe a few basic approaches. A modern introduction to convergence acceleration with excellent examples and many hints can be found in [49]; for a more detailed review, see [50, 51].

Richardson Extrapolation Assume that we already know the order of convergence for a series $S = \sum_{k=0}^{\infty} a_k$, so that for its partial sums $S_n = \sum_{k=0}^{n} a_k$ we have

$$S = S_n + \frac{\alpha}{n^p} + \mathcal{O}\big(n^{-r}\big), \quad r > p > 0.$$

We think of the "value of the series" as being the value of the partial sum plus a correction with the known leading-order behavior α/n^p. By transforming

$$T_n^{(1)} = \frac{2^p S_{2n} - S_n}{2^p - 1} = S_{2n} + \frac{S_{2n} - S_n}{2^p - 1}$$

the term α/n^p can be eliminated and the terms $T_n^{(1)}$ give us a better estimate for the sum, for which we obtain $S = T^{(1)} + \mathcal{O}(n^{-r})$. The very same trick can be repeated—until this makes sense—by forming new sequences,

$$T_n^{(2)} = \frac{2^{p+1} T_{2n}^{(1)} - T_n^{(1)}}{2^{p+1} - 1}, \qquad T_n^{(3)} = \frac{2^{p+2} T_{2n}^{(2)} - T_n^{(2)}}{2^{p+2} - 1}, \quad \cdots.$$

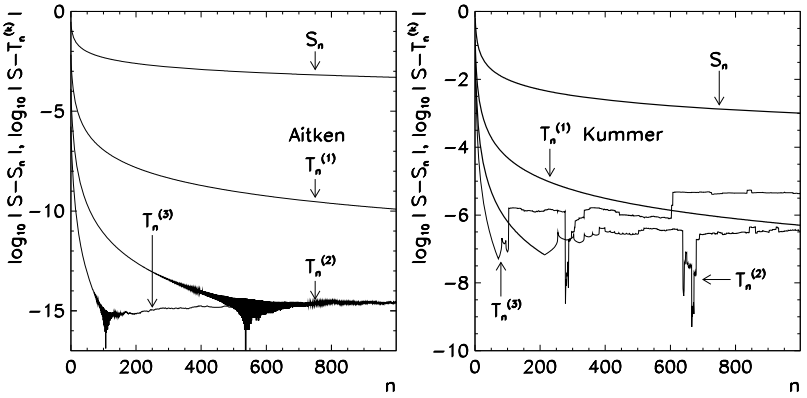


Fig. 1.10 Acceleration of convergence of partial sums. [*Left*] Aitken's method for the sum $S_n = \sum_{k=0}^{n} (-1)^k/(k+1)$ with the limit $S = \lim_{n\to\infty} S_n = \log 2$. Shown is the acceleration of this series with very slow (logarithmic) convergence by three-fold repetition of the Aitken's method. For typical series usually a single step of (1.58) suffices. [*Right*] Kummer's acceleration of the sums $S_n = \sum_{k=1}^{n} 1/k^2$ with the limit $S = \lim_{n\to\infty} S_n = \pi^2/6$ by using the auxiliary series $\sum_{k=1}^{\infty} 1/(k(k+1))$

Richardson's procedure is an example of a *linear extrapolation method* X, in which for partial sums S_n and T_n of two series we have $X(\lambda S_n + \mu T_n) = \lambda X(S_n) + \mu X(T_n)$. It is efficient if the partial sums S_n behave like polynomials in some sequence h_n, that is, $S_n = S + c_1 h_n^{p_1} + c_2 h_n^{p_2} + \cdots$ or $S_{n+1} = S + c_1 h_{n+1}^{p_1} + c_2 h_{n+1}^{p_2} + \cdots$, where the ratio h_{n+1}/h_n is constant. If this condition is not met, linear extrapolation may become inefficient or does not work at all. In such cases we resort to *semi-linear* or *non-linear extrapolation* [49].

Aitken's Method Aitken's method is one of the classical and most widely used ways to accelerate the convergence by non-linear extrapolation. Assume that we have a sequence of partial sums S_n with the limit $S = \lim_{n\to\infty} S_n$. We transform the sequence S_n into a new sequence

$$T_n^{(1)} = S_n - \frac{(S_{n+1} - S_n)^2}{S_{n+2} - 2S_{n+1} + S_n}, \quad n = 0, 1, 2, \ldots, \tag{1.58}$$

where the fraction should be evaluated exactly in the given form in order to minimize rounding errors. (Check that the transformed sequence (1.58) is identical to the column $\epsilon(n, 2)$ of the Wynn's table (1.12).) We repeat the process by using $T_n^{(1)}$ instead of S_n to form yet another, even more accelerated sequence $T_n^{(2)}$, and proceed thus until it continues to make sense as far as the rounding errors are concerned. Figure 1.10 (left) shows the comparison of convergence speeds for the unaccelerated partial sums S_n and the accelerated sequences $T_n^{(1)}$, $T_n^{(2)}$, and $T_n^{(3)}$.

Aitken's method is optimally suited for acceleration of linearly convergent sequences, for which $\lim_{n\to\infty}(S_{n+1} - S)/(S_n - S) = a$ with $-1 \le a < 1$. Such sequences originate in numerous numerical algorithms based on finite differences. In

some cases, we apply Aitken's formula to triplets of partial sums S_{n+p}, S_n, and S_{n-p}, where $p > 1$, because sometimes the geometric convergence of a series only becomes apparent at larger p; see also Sect. 2.3 and [52].

Kummer's Acceleration The basic idea of the Kummer's method of summing a convergent series $S = \sum_k a_k$ is to subtract from it another (auxiliary) convergent series $B = \sum_k b_k$ with the known limit B, such that

$$\lim_{k\to\infty} \frac{a_k}{b_k} = \rho \neq 0.$$

Then the original series can be transformed to

$$T = \sum_k a_k = \rho \sum_k b_k + \sum_k (a_k - \rho b_k) = \rho B + \sum_k \left(1 - \rho \frac{b_k}{a_k}\right) a_k. \tag{1.59}$$

The convergence of the series on the right is faster than the convergence of that on the left since $(1 - \rho b_k/a_k)$ tends to zero when $k \to \infty$. An example is the sum $S = \sum_{k=1}^{\infty} 1/k^2 = \pi^2/6$ from which we subtract $B = \sum_{k=1}^{\infty} 1/(k(k+1)) = 1$, thus $\rho = \lim_{k\to\infty} k(k+1)/k^2 = 1$. We use the terms a_k and b_k, as well as ρ and B, in (1.59), and get the transformed partial sum

$$T_n^{(1)} = 1 + \sum_{k=1}^{\infty} \left(1 - \frac{k^2}{k(k+1)}\right) \frac{1}{k^2},$$

which has a faster convergence than the original series. Again, the procedure can be invoked repeatedly (see [53] and Fig. 1.10 (right)).

1.4.4 Alternating Series

In alternating series the sign of the terms flips periodically,

$$S = a_0 - a_1 + a_2 - a_3 + \cdots = \sum_{k=0}^{\infty} (-1)^k a_k, \qquad S_n = \sum_{k=0}^{n} (-1)^k a_k.$$

In physics such examples can be encountered e.g. in electro-magnetism in problems with oppositely charged particles or currents flowing in opposite directions. An example is the calculation of the electric potential $U(x, y, z)$ of charges of opposite signs lying next to each other at distances a along the x-axis:

$$U(x, y, z) \propto \sum_{k=-\infty}^{\infty} \frac{(-1)^k}{\sqrt{(x + ka)^2 + y^2 + z^2}}.$$

Making a Monotonous Series Alternate For realistic physics problems, the results of series summation may be unpredictable. Simple recursive summation may suffer from large rounding errors. On the other hand, the acceleration of alternating series is typically more efficient than the acceleration of series with exclusively positive (or exclusively negative) terms. A monotonous sequence can be transformed into an alternating one by using the Van Wijngaarden's trick:

$$\sum_{k=0}^{\infty} a_k = \sum_{k=0}^{\infty} (-1)^k b_k, \quad b_k = \sum_{j=0}^{\infty} 2^j a_{2^j(k+1)-1}.$$

Euler's Transformation One of the oldest ways to accelerate the convergence of an alternating sequence by a linear combination of its terms is the *Euler transformation*. We rewrite the original sum $S = \sum_k (-1)^k a_k$ and its partial sum as

$$S = \sum_{k=0}^{\infty} (-1)^k \frac{\Delta^k a_0}{2^{k+1}}, \qquad S_n = \sum_{k=0}^{n} (-1)^k \frac{\Delta^k a_0}{2^{k+1}}, \tag{1.60}$$

where

$$\Delta^k a_0 = (-1)^k \sum_{j=0}^{k} (-1)^j \binom{k}{j} a_j.$$

If there exist $N \in \mathbb{N}$ and $C > 0$ such that $|\Delta^n a_0| \leq C$ for all $n > N$, the series (1.60) converges faster than geometrically with

$$|S - S_n| \leq \frac{C}{2^{n+1}}, \quad n > N.$$

In practical algorithms, we first form the partial sums

$$s_n^{(0)} = \sum_{k=0}^{n} (-1)^k a_k, \quad n = 0, 1, \ldots, N-1,$$

and recursively compute the *partial Euler transforms*

$$s_n^{(j+1)} = \frac{1}{2}\left(s_n^{(j)} + s_{n+1}^{(j)}\right), \quad j = 0, 1, \ldots. \tag{1.61}$$

The values $T_n \equiv s_0^{(n)}$ represent the improved (accelerated) approximations of the partial sums S_n (Fig. 1.11 (left)). The procedure is numerically demanding, since it requires $\mathcal{O}(n^2)$ operations and $\mathcal{O}(n)$ of memory for a complete transformation of a series with n terms. It turns out that the optimally precise results are obtained not by using the transform (1.61) with $j = N-1$ and $n = 0$, but with $j = \lceil 2N/3 \rceil$ and $n = \lceil N/3 \rceil$. An efficient implementation is given in [54].

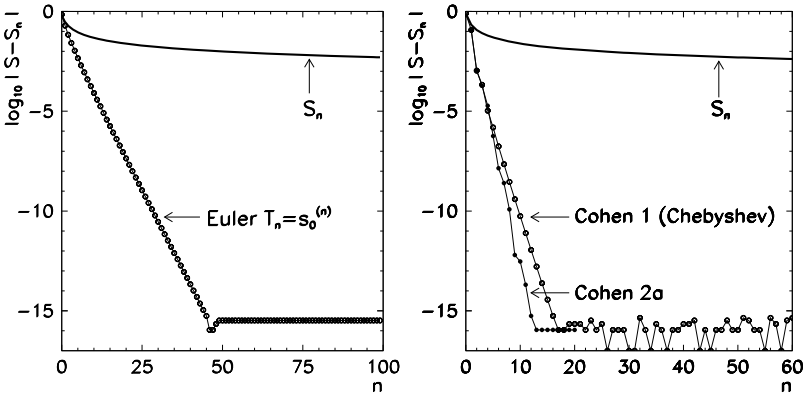


Fig. 1.11 Examples of acceleration of alternating series. Shown are the partial sums $S_n = \sum_{k=0}^{n} (-1)^k a_k$ without and with acceleration. [*Left*] Euler's method (1.61) for $a_k = 1/(k+1)$ (limit $S = \lim_{n\to\infty} S_n = \log 2$). [*Right*] Cohen–Villegas–Zagier's algorithm 1 from p. 42 by using Chebyshev polynomials (1.65) and algorithm 2a in [55] for $a_k = 1/(2k+1)$ (limit $S = \lim_{n\to\infty} S_n = \pi/4$). To compute the partial sum to ≈ 15 significant digits typically less than $\approx$10–20 terms of the accelerated series are required

Generalizing the Euler's Method Euler's transformation can be generalized by using the theory of measures. In this fresh approach to the summation of alternating series [55] we assume that for a series $\sum_{k=0}^{\infty} (-1)^k a_k$ there exists a positive function w such that the series terms a_k are its moments on the interval $[0, 1]$,

$$a_k = \int_0^1 x^k w(x)\,\mathrm{d}x. \tag{1.62}$$

The sum of the series can then be written as

$$S = \sum_{k=0}^{\infty} (-1)^k a_k = \int_0^1 \left(\sum_{k=0}^{\infty} (-1)^k x^k w(x) \right) \mathrm{d}x = \int_0^1 \frac{w(x)}{1+x}\,\mathrm{d}x.$$

(In the final summation formula the weight function does not appear.) In the last step, we have used the identity

$$\sum_{k=0}^{n-1} (-1)^k x^k = \frac{1-(-x)^n}{1+x}, \quad |x| < 1, \tag{1.63}$$

in the limit $n \to \infty$. We now choose a sequence of polynomials $\{P_n\}$, where P_n has a degree n and $P_n(-1) \neq 0$. To the sequence $\{P_n\}$ we assign the numbers

$$S_n = \frac{1}{P_n(-1)} \int_0^1 \frac{P_n(-1) - P_n(x)}{1+x}\, w(x)\,\mathrm{d}x.$$

The numbers S_n are linear combinations of the series terms a_k. This can be seen by inserting the expression for a general polynomial $P_n(x) = \sum_{k=0}^{n} p_k(-x)^k$ into the equation for S_n and observe (1.63) and a_k (1.62). We obtain

$$S_n = \frac{1}{d_n}\sum_{k=0}^{n-1}(-1)^k c_k^{(n)} a_k, \quad d_n = \sum_{k=0}^{n} p_k,\ c_k^{(n)} = \sum_{j=k+1}^{n} p_j.$$

The S_n defined in this way represent the partial sums of S that converge to S when n is increased. The difference between the partial sum S_n and the sum S can be constrained as

$$|S - S_n| \le \frac{1}{|P_n(-1)|}\int_0^1 \frac{|P_n(x)|}{1+x}\, w(x)\,\mathrm{d}x \le \frac{M_n}{|P_n(-1)|}|S|,$$

where $M_n = \sup_{x\in[0,1]}|P_n(x)|$ is the maximum value of the polynomial P_n on $[0, 1]$. The sufficient condition for the convergence of the partial sums S_n to S is therefore $\lim_{n\to\infty} M_n/P_n(-1) = 0$. The authors of [55] recommend to choose a sequence of polynomials $\{P_n\}$ such that $M_n/P_n(-1)$ converges to zero as quickly as possible. The following three choices are the most fruitful.

The first type of the polynomials P_n that may cross one's mind is

$$P_n(x) = (1-x)^n = \sum_{k=0}^{n}\binom{n}{k}(-x)^k, \qquad P_n(-1) = 2^n, \qquad M_n = 1.$$

Namely, the corresponding partial sums are

$$S_n = \frac{1}{2^n}\sum_{k=0}^{n-1}(-1)^k c_k^{(n)} a_k, \quad c_k^{(n)} = \sum_{j=k+1}^{n}\binom{n}{j}, \tag{1.64}$$

and they are identical to the partial sums of the Euler transform (1.60), except for a different subscripting (the sums (1.60) with subscript n are equal to the sums (1.64) with subscript $n + 1$). By this choice we obtain $|S - S_n| \le |S|/2^n$. Faster convergence, $|S - S_n| \le |S|/3^n$, can be obtained by using the polynomials

$$P_n(x) = (1-2x)^n = \sum_{k=0}^{n} 2^k\binom{n}{k}(-x)^k, \qquad P_n(-1) = 3^n, \qquad M_n = 1.$$

Here the partial sums have the form

$$S_n = \frac{1}{3^n}\sum_{k=0}^{n-1}(-1)^k c_k^{(n)} a_k, \quad c_k^{(n)} = \sum_{j=k+1}^{n} 2^j\binom{n}{j}.$$

A third choice is a special family of Chebyshev polynomials, which have other beneficial algebraic properties and are orthogonal. We define these polynomials im-

plicitly by $P_n(\sin^2 t) = \cos(2nt)$ or explicitly by

$$P_n(x) = T_n(1-2x) = \sum_{j=0}^{n} 4^j \frac{n}{n+j} \binom{n+j}{2j} (-x)^j,$$

where $T_n(x) = \cos(n \arccos x)$ are the standard Chebyshev polynomials of degree n on $[-1, 1]$. The polynomials of this sequence are computed by using the recurrence $P_{n+1}(x) = 2(1-2x)P_n(x) - P_{n-1}(x)$, which is initiated by $P_0(x) = 1$ and $P_1(x) = 1 - 2x$. For polynomials chosen in this way, one can show that $P_n(-1) = \frac{1}{2}[(3+\sqrt{8})^n + (3-\sqrt{8})^n]$ and $M_n = 1$. The partial sums

$$S_n = \frac{1}{P_n(-1)} \sum_{k=0}^{n-1} (-1)^k c_k^{(n)} a_k, \quad c_k^{(n)} = \sum_{j=k+1}^{n} 4^j \frac{n}{n+j} \binom{n+j}{2j}, \tag{1.65}$$

converge to the final sum as

$$|S - S_n| \le \frac{2|S|}{(3+\sqrt{8})^n} < \frac{2|S|}{5.828^n},$$

so we need to sum only $n \approx 1.31\,D$ terms for a precision of D significant digits! The coefficients $c_k^{(n)}$ and other constants can be computed iteratively and the whole computation of S_n can be implemented in a very compact algorithm [55]

Input: numbers $a_0, a_1, \ldots, a_{n-1}$ of an alternating series $\sum_{k=0}^{n-1} (-1)^k a_k$
$d = (3+\sqrt{8})^n$; $d = (d + 1/d)/2$;
$b = -1$; $c = -d$; $s = 0$;
for $k = 0$ **step** 1 **to** $n-1$ **do**
  $c = b - c$;
  $s = s + c\,a_k$;
  $b = (k+n)(k-n)b/((k+1/2)(k+1))$;
end
Output: partial sum $S_n = s/d$

This algorithm requires $\mathcal{O}(1)$ of memory and $\mathcal{O}(n)$ of CPU. Similar results can be obtained by using other families of orthogonal polynomials; the paper [55] describes further algorithms in which the coefficients of the partial sums cannot be generated as easily, but yield even faster convergence. For many types of sequences, these algorithms allow us to achieve convergence rates of $|S - S_n| \le |S|/7.89^n$, in some cases even the breath-taking $|S - S_n| \le |S|/17.93^n$. However, they require $\mathcal{O}(n)$ of memory and $\mathcal{O}(n^2)$ of CPU.

1.4.5 Levin's Transformations

Levin's transformations [56] are among the most generally useful, handy, and efficient methods to accelerate the convergence of series by semi-linear extrapolation.

We implement them by using divided differences which are computed recursively:

$$\delta^k f_n = \frac{\delta^{k-1} f_{n+1} - \delta^{k-1} f_n}{t_{n+k} - t_n}, \qquad \delta^0 f_n = f_n,$$

where $t_n = (n + n_0)^{-1}$ and we usually take $n_0 = 0$ or $n_0 = 1$. To compute the extrapolated partial sums we need the partial sums S_n and auxiliary functions ψ which depend on the terms of the sequence and its character (monotonous or alternating). We use the formula

$$S_{k,n} = \delta^k \left(\frac{S_n}{\psi(n)} \right) \left[\delta^k \left(\frac{1}{\psi(n)} \right) \right]^{-1}, \quad k = 1, 2, \ldots \tag{1.66}$$

and take $S_{k,0}$ (with $n_0 = 1$) or $S_{k,1}$ (with $n_0 = 0$) as the extrapolated sum. Levin's transformations differ by the functional forms of ψ. The best known are

$$T:\ \psi(n) = a_n, \qquad U:\ \psi(n) = (n + n_0)a_n, \qquad W:\ \psi(n) = a_n^2/(a_{n+1} - a_n).$$

The T-transformation is best for alternating series in which the partial sums behave as $S_n \sim r^n$, where r is the convergence ratio. The U-transformation works well with monotonous sequences for which $S_n \sim n^{-j}$ applies. The W-transformation can be used in either case regardless of the series type, although it is more sensitive to rounding errors than the U- and T-methods. The U-method is recommended [49] as a reliable way to speed up the summation of any series. The U-transformation, including the extrapolation to the limit and providing the remainder estimates, is implemented in the GSL library [57] in the `gsl_sum_levin_u_accel()` function.

Example Let us sum the slowly converging series $S_n = \sum_{k=0}^{n} (-1)^k/(k+1)$ with the limit $S = \lim_{n\to\infty} S_n = \log 2$. We choose the Levin's T-method and $n_0 = 0$, thus $t_n = n^{-1}$ and $\psi(n) = (-1)^n/(n+1)$. By using (1.66) with $n = 1$ we obtain

$k =$	1	$S_k =$	0.5	$S_{k,1} =$	0.7
	2		0.8333333333333334		0.6923076923076924
	3		0.5833333333333334		0.6932153392330384
	4		0.7833333333333333		0.6931436119116234
	5		0.6166666666666667		0.6931472579962513
	6		0.7595238095238096		0.6931471858853886
	7		0.6345238095238096		0.6931471799439380
	8		0.7456349206349208		0.6931471805844429
	9		0.6456349206349208		0.6931471805603313
	10		0.7365440115440117		0.6931471805598398.

While the partial sums S_k merely hint at convergence, the accelerated sum $S_{k,1}$ at $k = 10$ is already precise to 12 digits. See also Fig. 1.12 (left).

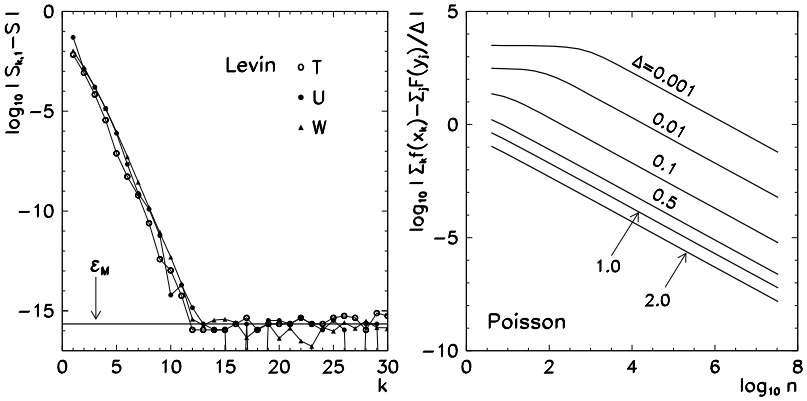


Fig. 1.12 [*Left*] Precision of Levin's methods T, U, and W in accelerating the convergence of the partial sums $S_n = \sum_{k=0}^{n} (-1)^k/(k+1)$ with the limit $S = \lim_{n\to\infty} S_n = \log 2$. [*Right*] Precision of the Poisson's summation formula (1.67) for $f(x) = 1/(1+x^2)$ with different samplings $x_k = k\,\Delta$ on the real axis, where $-n \le \{j,k\} \le n$ and $n \gg 1$

1.4.6 Poisson Summation

Often we encounter sums of function values f at equidistant points on the real axis,

$$S = \sum_{k\in\mathbb{Z}} f(x_k), \quad x_k = k\,\Delta,\ \Delta = x_{k+1} - x_k.$$

Assume that f is differentiable, that it decreases sufficiently fast at infinity, and that its Fourier transform

$$F(y) = \int_{-\infty}^{\infty} f(x)\, \mathrm{e}^{-\mathrm{i}\,2\pi xy}\, \mathrm{d}x$$

exists. The sum of the values $f(x_k)$ and the sum of the transforms $F(y_j)$, computed at $y_j = j/\Delta$, $j \in \mathbb{Z}$, are linked by the Poisson's summation formula (see Fig. 1.12 (right))

$$\sum_{k=-\infty}^{\infty} f(x_k) = \frac{1}{\Delta} \sum_{j=-\infty}^{\infty} F(y_j). \tag{1.67}$$

1.4.7 Borel Summation

Perturbative solutions in classical and quantum mechanics often appear as formally divergent series which, by appropriate means of summation, can yield a conditionally valid final result. Assume we have a sequence $\{a_k\}_{k\in\mathbb{N}_0}$ with the sum $\sum_{k=0}^{\infty} a_k$ that diverges. In the case when all a_k can be explicitly expressed as functions of

the index k, the series can be summed by *Borel resummation*. The original sum is *resummable* in the form

$$S = \lim_{\xi\to\infty} \mathrm{e}^{-\xi} \sum_{n=0}^{\infty} \frac{\xi^n}{n!} S_n, \quad S_n = \sum_{k=0}^{n} a_k, \tag{1.68}$$

if the corresponding limit exists. In practice, the summation parameter ξ is not allowed to go to infinity; rather, we try to locate a range of its values in which the value of S stabilizes when ξ is being increased.

The resummation (1.68) is defined in its differential form. Even more often, we use the integral form, in which the series terms a_k (not the partial sums S_n) are used:

$$S = \int_0^{\infty} \mathrm{e}^{-\xi} \left(\sum_{k=0}^{\infty} \frac{a_k \xi^k}{k!} \right) \mathrm{d}\xi.$$

This form is particularly useful when the function $f(\xi) = \sum_{k=0}^{\infty} a_k \xi^k / k!$ can be written in closed form or is well known in the region where it contributes most significantly to the integral $\int_0^{\infty} f(\xi)\, \mathrm{e}^{-\xi}\, \mathrm{d}\xi$. To do this, we can use the Padé approximation (see Sect. 1.2.2 and Problem 1.5.5).

1.4.8 Abel Summation

Assume that the series $S = \sum_{k=0}^{\infty} a_k$ formally diverges, but that the limit of the expression $S(x) = \sum_{k=0}^{\infty} x^k a_k$ still exists when $x \nearrow 1$. We can also introduce an auxiliary parameter ε such that $x = \mathrm{e}^{-\varepsilon}$ and observe the limit $\varepsilon \searrow 0$. Then the value

$$S_{\mathrm{A}} = \lim_{x\nearrow 1} S(x) = \lim_{x\nearrow 1} \sum_{k=0}^{\infty} x^k a_k = \lim_{\varepsilon\searrow 0} \sum_{k=0}^{\infty} \mathrm{e}^{-\varepsilon k} a_k$$

is called the Abel's generalized sum of the series S. Like with the Borel summation, we introduce an intermediate parameter to regularize a divergent series and then try to sum it in the hope that the generalized limit is finite.

Example The divergent series

$$S = \sum_{k=0}^{\infty} k \cos kr = \mathrm{Re} \sum_{k=0}^{\infty} k\, \mathrm{e}^{\mathrm{i}kr}, \quad 0 < r < 2\pi,$$

has the Abel sum

$$S_{\mathrm{A}} = \mathrm{Re} \lim_{x\nearrow 1} \sum_{k=0}^{\infty} x^k k\, \mathrm{e}^{\mathrm{i}kr} = \mathrm{Re} \lim_{x\nearrow 1} \sum_{k=0}^{\infty} k \left(x\, \mathrm{e}^{\mathrm{i}r} \right)^k = \mathrm{Re} \frac{\mathrm{e}^{\mathrm{i}r}}{(1 - \mathrm{e}^{\mathrm{i}r})^2} = -\frac{1}{4 \sin^2 r/2}.$$

As an exercise, change the parameter $0 < x < 1$ (approach $x \nearrow 1$) and the upper range of the sum, and watch what happens to the values S and S_A.

An analogous method can be used with divergent integrals. If the integral $S = \int_a^\infty f(x)\,\mathrm{d}x$ diverges, its generalized Abel sum is given by

$$S_\mathrm{A} = \lim_{\varepsilon \searrow 0} \int_a^\infty \mathrm{e}^{-\varepsilon x} f(x)\,\mathrm{d}x.$$

Example The divergent integral

$$S = \int_0^\infty \sqrt{x} \cos \alpha x\,\mathrm{d}x, \quad \alpha > 0$$

has the generalized value

$$S_\mathrm{A} = \lim_{\varepsilon \searrow 0} \int_0^\infty \mathrm{e}^{-\varepsilon x} \sqrt{x} \cos \alpha x\,\mathrm{d}x = \mathrm{Re} \lim_{\varepsilon \searrow 0} \int_0^\infty \sqrt{x}\,\mathrm{e}^{-(\varepsilon + \mathrm{i}\alpha)x}\,\mathrm{d}x = -\frac{\sqrt{\pi}}{(2\alpha)^{3/2}}.$$

For exercise, take $\alpha = 1$. Change the upper integration limit in the expressions for S and S_A, and force the limiting parameter ε to approach zero from above. Compare the results to the analytic limit $-\sqrt{\pi}/(2\alpha)^{3/2}$.

1.5 Problems

1.5.1 Integral of the Gauss Distribution

The standard (Gaussian) probability distribution $(1/\sqrt{2\pi})\,\exp(-x^2/2)$ pervades all branches of probability and statistics. Usually we need the probability over an interval, which can be computed by the integral

$$\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x \exp\bigl(-t^2\bigr)\,\mathrm{d}t, \qquad \mathrm{erfc}(x) = 1 - \mathrm{erf}(x). \tag{1.69}$$

Let us restrict the discussion to $x \geq 0$. The erf function monotonously increases and rapidly converges to 1 for large arguments. Its values are tabulated in textbooks, but for general purposes we wish to be able to compute them by exploiting different representations and approximations of the erf and erfc functions. If $\mathrm{erf}(x)$ is known, $\mathrm{erfc}(x)$ is also known (and vice versa), but it is preferable to compute the function having a smaller argument because the error is easier to control. We can switch between calculations of erf and erfc at the point $x \approx 0.4769362762044695$ where $\mathrm{erf}(x) = \mathrm{erfc}(x) = \frac{1}{2}$.

For small x, we can use the power expansion

$$\mathrm{erf}(x) = \frac{2x}{\sqrt{\pi}} \sum_{k=0}^{\infty} (-1)^k \frac{x^{2k}}{k!(2k+1)}.$$

The convergence radius of this series is infinite, but its terms alternate and without acceleration it is not suitable for the computation of $\operatorname{erf}(x)$ at x much larger than 1. For large arguments, $x \gg 1$, the asymptotic expansion is suitable:

$$\operatorname{erfc}(x) \sim \frac{\mathrm{e}^{-x^2}}{x\sqrt{\pi}}\left[1 + \sum_{k=1}^{\infty}(-1)^k \frac{(2k-1)!!}{(2x^2)^k}\right].$$

In the range of x where both the power and the asymptotic expansions provide a poor description of erf, a rational approximation

$$\operatorname{erfc}(x) = \mathrm{e}^{-x^2} \frac{\sum_{k=0}^{7} a_k x^k}{\sum_{k=0}^{7} b_k x^k}\big(1 + \varepsilon(x)\big)$$

may be used. The parameters of this formula are listed in the following table.

k	a_k	b_k
0	1.000000000000013	1.000000000000000
1	1.474885681937094	2.603264849035166
2	1.089127207353042	3.026597029346489
3	0.481934851516365	2.046071816911715
4	0.133025422885837	0.873486411474986
5	0.021627200301105	0.237214006125950
6	0.001630015433745	0.038334123870994
7	−0.000000000566405	0.002889083295887

The relative precision of this parameterization is $\varepsilon(x) < 10^{-14}$ on $x \in [0, 5]$. An elegant, fast, but almost impenetrable implementation of the power expansion of erf and of the computation of erfc by means of tabulated values with a precision of 14 to 16 digits in C can be found in [58]. The algorithms in the GNU C library are also based on rational approximations with many parameters [59, 60].

⊙ Examine the applicability and usefulness of different methods to compute $\operatorname{erf}(x)$. Watch the convergence of the power and asymptotic series. In the latter, sum the terms until the series appears to be converging. Does the convergence improve if Euler's method or Levin's U-transformation is used? By using all three ways of computation, write a program to calculate $\operatorname{erf}(x)$ in double precision on the whole real axis with an error less than 10^{-10}. Try to maximize the speed of the program. Show a comparison table of $\operatorname{erf}(x)$ for $x = 0\,(0.2)\,3$ and $\operatorname{erfc}(x)$ for $x = 3\,(0.5)\,8$.

The integral (1.69) can also be integrated numerically. What would be the required size of the subintervals in the Simpson's formula (Appendix E) in order to achieve a precision that would be comparable to the summation methods used above?

⊕ Write a program to compute the inverse function erf^{-1} with an absolute precision of 10^{-9}. Now that we are in possession of an efficient procedure to compute $\operatorname{erf}(x)$, the inverse can be found by finding the root of the equation

$$F(x) = \operatorname{erf}(x) - y = 0$$

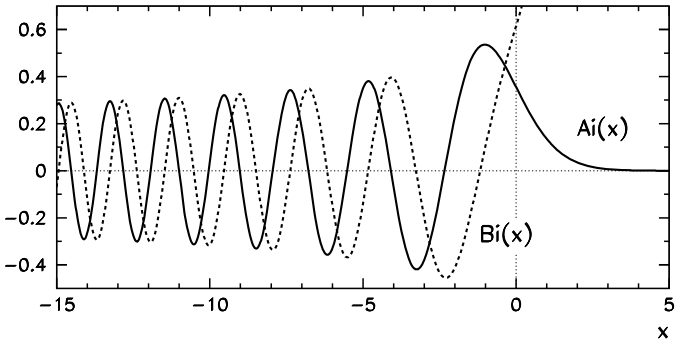


Fig. 1.13 Airy functions Ai and Bi for real arguments. Ai is bound everywhere, while Bi diverges at $x \to \infty$. The zeros of both functions occur only on the negative semi-axis

by using a method to solve non-linear equations. You can use bisection because erf is monotonous, but since the derivative $[\mathrm{erf}(x)]' = 2\mathrm{e}^{-x^2}/\sqrt{\pi}$ is also known, the much faster Newton's method can be used. Compare the computed $\mathrm{erf}^{-1}(y)$ for small values of y to the power expansion [61, 62]

$$\mathrm{erf}^{-1}(y) = \sum_{k=0}^{\infty} \frac{c_k}{2k+1} \left(\frac{\sqrt{\pi}}{2} y \right)^{2k+1}, \quad c_0 = 1,\ c_{k\geq 1} = \sum_{m=0}^{k-1} \frac{c_m c_{k-1-m}}{(m+1)(2m+1)}.$$

1.5.2 Airy Functions

In physics, the Airy functions Ai and Bi (Fig. 1.13) appear in optics and quantum mechanics [63]. They are defined as the independent solutions of the equation

$$y''(x) - xy(x) = 0$$

and have the integral representations

$$\mathrm{Ai}(x) = \frac{1}{\pi} \int_0^{\infty} \cos\bigl(t^3/3 + xt\bigr)\, \mathrm{d}t,$$
$$\mathrm{Bi}(x) = \frac{1}{\pi} \int_0^{\infty} \bigl[\mathrm{e}^{-t^3/3+xt} + \sin\bigl(t^3/3 + xt\bigr)\bigr]\, \mathrm{d}t.$$

For small x the functions Ai and Bi can be expressed by the Maclaurin series

$$\mathrm{Ai}(x) = \alpha f(x) - \beta g(x), \qquad \mathrm{Bi}(x) = \sqrt{3}\bigl[\alpha f(x) + \beta g(x)\bigr],$$

where, at $x = 0$, we have $\alpha = \mathrm{Ai}(0) = \mathrm{Bi}(0)/\sqrt{3} \approx 0.355028053887817239$ and $\beta = -\mathrm{Ai}'(0) = \mathrm{Bi}'(0)/\sqrt{3} \approx 0.258819403792806798$. The series for f and g are

$$f(x) = \sum_{k=0}^{\infty} \left(\frac{1}{3}\right)_k \frac{3^k x^{3k}}{(3k)!}, \qquad g(x) = \sum_{k=0}^{\infty} \left(\frac{2}{3}\right)_k \frac{3^k x^{3k+1}}{(3k+1)!},$$

where $(z)_n = \Gamma(z+n)/\Gamma(z)$ and $(z)_0 = 1$.

For large $|x|$ the Airy functions can be approximated by their asymptotic expansions. By substituting $\xi = \frac{2}{3}|x|^{3/2}$ and by using the asymptotic series

$$L(z) \sim \sum_{s=0}^{\infty} \frac{u_s}{z^s}, \qquad P(z) \sim \sum_{s=0}^{\infty} (-1)^s \frac{u_{2s}}{z^{2s}}, \qquad Q(z) \sim \sum_{s=0}^{\infty} (-1)^s \frac{u_{2s+1}}{z^{2s+1}},$$

with coefficients

$$u_s = \frac{\Gamma(3s + \frac{1}{2})}{54^s s! \Gamma(s + \frac{1}{2})},$$

we get, for large positive x,

$$\mathrm{Ai}(x) \sim \frac{\mathrm{e}^{-\xi}}{2\sqrt{\pi} x^{1/4}} L(-\xi), \qquad \mathrm{Bi}(x) \sim \frac{\mathrm{e}^{\xi}}{\sqrt{\pi} x^{1/4}} L(\xi),$$

while for large negative x we have

$$\mathrm{Ai}(x) \sim \frac{1}{\sqrt{\pi}(-x)^{1/4}} \big[\sin(\xi - \pi/4) Q(\xi) + \cos(\xi - \pi/4) P(\xi)\big],$$

$$\mathrm{Bi}(x) \sim \frac{1}{\sqrt{\pi}(-x)^{1/4}} \big[-\sin(\xi - \pi/4) P(\xi) + \cos(\xi - \pi/4) Q(\xi)\big].$$

⊙ Find an efficient procedure to compute the values of the Airy functions Ai and Bi on the real axis with a precision better than 10^{-10} by using a combination of the Maclaurin series and the asymptotic expansion. When estimating the errors, use programs that are capable of arbitrary-precision computations, e.g. MATHEMATICA.

⊕ The zeros of Ai have an important role in mathematical analysis when one tries to determine the intervals containing zeros of other special functions and orthogonal polynomials [64], as well as in physics in computation of energy spectra of quantum systems [34]. Compute the first hundred zeros $\{a_s\}_{s=1}^{100}$ of the Airy function Ai and the first hundred zeros $\{b_s\}_{s=1}^{100}$ of Bi at $x < 0$, and compare the computed values to the formulas

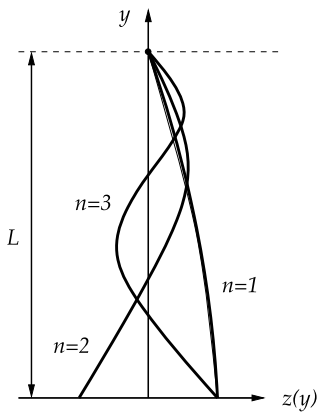
$$a_s = -f\left(\frac{3\pi(4s-1)}{8}\right), \qquad b_s = -f\left(\frac{3\pi(4s-3)}{8}\right), \quad s = 1, 2, \ldots,$$

where f has the asymptotic expansion [22]

$$f(z) \sim z^{2/3}\left(1 + \frac{5}{48}z^{-2} - \frac{5}{36}z^{-4} + \frac{77125}{82944}z^{-6} - \frac{108056875}{6967296}z^{-8} + \cdots\right).$$

1.5.3 Bessel Functions

Solving differential equations in circular or spherical geometry often leads to Bessel functions of the first kind J_ν and second kind Y_ν. A physical example which does not belong to this group but is related to it, is the problem of the oscillations of a freely hanging heavy rope in a constant gravitational field, as shown in the figure below.



A rope of length L is suspended from a ceiling and the origin of the y axis is placed at the free end of the rope. We study small deflections $z(y)$ of the rope in the field of the gravitational acceleration g. Three lowest eigenmodes are shown. The eigenmodes are described by the equation

$$\frac{\mathrm{d}}{\mathrm{d}y}\left(y\,\frac{\mathrm{d}z(y)}{\mathrm{d}y}\right) + \frac{\omega^2}{g}z(y) = 0,$$

where $z(y)$ is the deflection of the rope at y when oscillating with the angular frequency ω. The eigenfrequencies are determined by the equation and the boundary conditions $|z(0)| < \infty$ (deflection bounded at $y = 0$) and $z(L) = 0$ (fixed end at $y = L$). By substitution $t = 2\omega\sqrt{y/g}$ the differential equation can be transformed to $\ddot{z}(t) + \dot{z}(t)/t + z(t) = 0$. The solution of this equation is a linear combination of the Bessel functions J_0 and Y_0. With $\alpha = \sqrt{L/g}$ the boundary conditions become

$$\big|z(t = 0)\big| < \infty, \qquad z(t = 2\omega\alpha) = 0.$$

The second condition eliminates Y_0 which is singular at the origin. The condition for the eigenfrequencies is then

$$\omega_n = \xi_{0,n}/(2\alpha), \quad n = 1, 2, \ldots,$$

where $\xi_{0,n}$ is the nth zero of J_0. The values of $J_0(x)$ for small x can be computed by using the power expansion

$$J_0(x) = \sum_{k=0}^{\infty} \frac{(-1)^k (x/2)^{2k}}{(k!)^2},$$

while at large enough arguments, we use the formula

$$J_0(x) = \sqrt{\frac{2}{\pi x}}\big[P(x)\cos(x-\pi/4) + Q(x)\sin(x-\pi/4)\big], \quad x \to \infty,$$

where $P(x)$ and $Q(x)$ are known in terms of the asymptotic series [65]

$$P(x) \sim \sum_{k=0}^{\infty}(-1)^k \frac{a_{2k}^2}{(2k)!(8x)^{2k}}, \qquad Q(x) \sim \sum_{k=0}^{\infty}(-1)^k \frac{a_{2k+1}^2}{(2k+1)!(8x)^{2k+1}}$$

with coefficients $a_n = (2n-1)!!$. (Note $(-1)!! = 0!! = 1$.) For intermediate x we compute $J_0(x)$ by using Bessel functions of higher orders J_n. In the limit $n \to \infty$ and constant x we have $J_n(x) \sim (x/2)^n/n!$, and for $x \sim 1$ and $n > 2x$ this is a very good approximation. Suppose we wish to calculate $J_0(x)$ at given x in arithmetic with precision ε_{M}. With the asymptotic approximation we determine an even $N \gg 2x$ such that $\varepsilon = J_N(x) \ll \varepsilon_{\mathrm{M}}$. This value of $J_N(x)$ is not normalized. Let us denote J_n temporarily by C_n and, for given x, start the iteration

$$C_{N+1} = 0, \qquad C_N = \varepsilon, \qquad C_{n-1}(x) = \frac{2n}{x}\,C_n(x) - C_{n+1}(x).$$

We obtain $C_0(x)$ which differs by a factor from the true value, $J_0(x) = C_0(x)/A$. The factor A is determined by using the identity $J_0(x) + 2\sum_{k=1}^{\infty} J_{2k}(x) = 1$:

$$A = C_0(x) + 2\sum_{k=1}^{N/2} C_{2k}(x).$$

⊙ Write a procedure to compute J_0 on the real axis to an absolute precision of 10^{-12} and compare it to the result of a tool of higher precision, like MATHEMATICA or MATLAB. Determine the first $N = 10000$ zeros $\{\xi_{0,n}\}_{n=1}^{N}$ of J_0 and compare them to the asymptotic formula [22, 31]

$$\xi_{0,n} \sim \beta + \frac{1}{8\beta} - \frac{31}{384\beta^3} + \frac{3779}{15360\beta^5} - \frac{6277237}{3440640\beta^7} + \cdots, \quad n \to \infty,$$

where $\beta = (n - 1/4)\pi$. Draw the first five eigenmodes of the oscillating rope.

1.5.4 Alternating Series

Some alternating series are literally famous, for example

$$\frac{\pi}{4} = \sum_{k=0}^{\infty}\frac{(-1)^k}{2k+1}, \qquad \log 2 = \sum_{k=0}^{\infty}\frac{(-1)^k}{k+1}, \qquad \frac{\pi^2}{12} = \sum_{k=0}^{\infty}\frac{(-1)^k}{(k+1)^2}.$$

Just as well-known, the natural algorithm has the expansions

$$\log x = \sum_{k=1}^{\infty} \frac{(-1)^{k+1}}{k}(x-1)^k, \quad |x-1| \le 1,\ x \neq 1,$$
$$\log x = 2\sum_{k=0}^{\infty} \frac{1}{2k+1}\left(\frac{x-1}{x+1}\right)^{2k+1}, \quad x > 0, \tag{1.70}$$

that enable us to compute $\log x$ on the whole positive semi-axis.

$\odot$ Compute the sums of these series to a precision of $\varepsilon = 10^{-7}$ by using different methods applicable to alternating series, and study their convergence. (The second series in (1.70) is an exception since it is not alternating.) Compare these methods to simple summation. Make a detailed analysis of the rounding errors first by using single-precision data types (type `float` in C or C++), and then by using double precision (type `double`).

$\bigoplus$ Sum the series given above by using a data type that allows for variable precision (for example, by using the GMP library mentioned in Appendix B.3). Draw a diagram of computational (CPU) times versus the required precision ε in the range $\log_{10}\varepsilon \in [-300, -4]$.

1.5.5 Coulomb Scattering Amplitude and Borel Resummation

An eloquent example [66] of trouble we may face by careless summation of divergent series is the Rutherford scattering amplitude for Coulomb scattering

$$f(\theta) = -\frac{\eta}{2k\sin^2(\theta/2)} \exp\bigl[\mathrm{i}\bigl(2\sigma_0 - \eta\log\sin^2(\theta/2)\bigr)\bigr]. \tag{1.71}$$

We know that the asymptotic expansion of this amplitude is

$$f(\theta) \sim \frac{1}{2\mathrm{i}k}\sum_{l=0}^{\infty}(2l+1)P_l(\cos\theta)\bigl(\mathrm{e}^{2\mathrm{i}\sigma_l} - 1\bigr), \tag{1.72}$$

where σ_l is the phase shift for Coulomb scattering in the partial wave with the orbital angular momentum quantum number l, and P_l is the Legendre polynomial of degree l (see (4.28)). The phase shift in the lth partial wave is

$$\sigma_l = \arg\Gamma(l+1+\mathrm{i}\eta).$$

In the limit $l \to \infty$ the phase shifts behave as $\sigma_l \sim \log l$, while the values $P_l(\cos\theta)$ at fixed θ fade out like $P_l(\cos\theta) \sim l^{-1/2}$. Therefore, for $l \to \infty$, the terms in the sum (1.72) oscillate within an envelope that is proportional to $l^{1/2}$, and the series diverges.

⊙ Sum the series (1.72) directly by summing the terms up to $l_{\max} = 100$ by using $\eta = 1$ and $k = 1\,\mathrm{fm}^{-1}$. Compute the sum for angles $30^\circ \le \theta \le 180^\circ$ in steps of 10° and compare the results to (1.71). Do not calculate the phase shifts by using the gamma function. Rather, use backward recurrence: start with the value of σ_l at $l = l_{\max}$, for which the Stirling approximation applies:

$$\sigma_l \sim \left(l + \frac{1}{2}\right)\beta + \eta \log \alpha - \eta - \frac{\sin\beta}{12\alpha} + \frac{\sin 3\beta}{360\alpha^3} - \frac{\sin 5\beta}{1260\alpha^5} + \frac{\sin 7\beta}{1680\alpha^7} - \frac{\sin 9\beta}{1188\alpha^9} + \cdots,$$

where

$$\alpha = \sqrt{(l+1)^2 + \eta^2}, \qquad \beta = \arctan\left(\frac{\eta}{l+1}\right).$$

Then use the recurrence to compute the phase shifts at lower l:

$$\sigma_l = \sigma_{l+1} - \arctan\left(\frac{\eta}{l+1}\right), \quad l = l_{\max} - 1, l_{\max} - 2, \ldots, 0.$$

Similarly, compute the Legendre polynomials by using the three-term recurrence formula $(l+1)P_{l+1}(x) = (2l+1)xP_l(x) - lP_{l-1}(x)$, which is initialized by $P_0(x) = 1$ and $P_1(1) = x$.

In addition, sum the series by Borel resummation in differential form (1.68). Compare the exact value (1.71) to the numerical one at angles $\theta = 10^\circ$, 60°, 120°, and 150° with the parameter ξ in the range $5 \le \xi \le 100$ in steps of 5.

⊕ Compute the Rutherford sum by applying the Wynn algorithm (1.12). At scattering angles $\theta = 10^\circ$, 60°, 120°, and 150°, calculate the diagonal Padé approximations $[n/n]$ for $0 \le n \le 20$. Stop the recurrence in the algorithm when the denominator of the fraction on the right side of (1.12) becomes equal to zero.

References

1. T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, *Introduction to Algorithms*, 2nd edn. (MIT Press/McGraw-Hill, Cambridge/New York, 2001)
2. D.E. Knuth, *The Art of Computer Programming, Vol. 1: Fundamental Algorithms*, 3rd edn. (Addison-Wesley, Reading, 1997)
3. IEEE Standard 754-2008 for Binary Floating-Point Arithmetic, IEEE 2008; see also http://grouper.ieee.org/groups/754/
4. D. Goldberg, What every computer scientist should know about floating-point arithmetic. ACM Comput. Surv. **23**, 5 (1991). An up-to-date version is accessible as Appendix D of the Sun Microsystems Numerical Computation Guide. http://docs.sun.com/source/806-3568
5. M.H. Holmes, *Introduction to Numerical Methods in Differential Equations* (Springer, New York, 2007) (Example in Appendix A.3.1)
6. D. O'Connor, *Floating Point Arithmetic*. Dublin Area Mathematics Colloquium, March 5, 2005.
7. GNU Multi Precision (GMP), free library for arbitrary precision arithmetic. http://gmplib.org
8. H.J. Wilkinson, *Rounding Errors in Algebraic Processes* (Dover, Mineola, 1994)
9. D.E. Knuth, *The Art of Computer Programming, Vol. 2: Seminumerical Algorithms*, 2nd edn. (Addison-Wesley, Reading, 1980)

10. J. Stoer, R. Bulirsch, *Introduction to Numerical Analysis*, 3rd edn. Texts in Applied Mathematics, vol. 12 (Springer, Berlin, 2002)
11. M.J.D. Powell, *Approximation Theory and Methods* (Cambridge University Press, Cambridge, 1981)
12. C. Hastings, *Approximations for Digital Computers* (Princeton University Press, Princeton, 1955), which is a pedagogical jewel; a seemingly simplistic outward appearance hides a true treasure-trove of ideas yielding one insight followed by another
13. W. Fraser, A survey of methods of computing minimax and near-minimax polynomial approximations for functions of a single variable. J. Assoc. Comput. Mach. **12**, 295 (1965)
14. H.M. Antia, *Numerical Methods for Scientists and Engineers*, 2nd edn. (Birkhäuser, Basel, 2002); see Sects. 9.11 and 9.12
15. R. Pachón, L.N. Trefethen, Barycentric-Remez algorithms for best polynomial approximation in the chebfun system. Oxford University Computing Laboratory, NAG Report No. 08/20
16. G.A. Baker, P. Graves-Morris, *Padé Approximants*, 2nd edn. Encyclopedia of Mathematics and Its Applications, vol. 59 (Cambridge University Press, Cambridge, 1996)
17. P. Wynn, On the convergence and stability of the epsilon algorithm. SIAM J. Numer. Anal. **3**, 91 (1966)
18. P.R. Graves-Morris, D.E. Roberts, A. Salam, The epsilon algorithm and related topics. J. Comput. Appl. Math. **12**, 51 (2000)
19. W. van Dijk, F.M. Toyama, Accurate numerical solutions of the time-dependent Schrödinger equation. Phys. Rev. E **75**, 036707 (2007)
20. K. Kormann, S. Holmgren, O. Karlsson, J. Chem. Phys. **128**, 184101 (2008)
21. C. Lubich, in *Quantum Simulations of Complex Many-Body Systems: From Theory to Algorithms*, ed. by J. Grotendorst, D. Marx, A. Muramatsu. NIC Series, vol. 10 (John von Neumann Institute for Computing, Jülich, 2002), p. 459
22. M. Abramowitz, I.A. Stegun, *Handbook of Mathematical Functions*, 10th edn. (Dover, Mineola, 1972)
23. I.S. Gradshteyn, I.M. Ryzhik, *Tables of Integrals, Series and Products* (Academic Press, New York, 1980)
24. P.C. Abbott, Asymptotic expansion of the Keesom integral. J. Phys. A, Math. Theor. **40**, 8599 (2007)
25. M. Battezzati, V. Magnasco, Asymptotic evaluation of the Keesom integral. J. Phys. A, Math. Theor. **37**, 9677 (2004)
26. T.M. Apostol, *Mathematical Analysis*, 2nd edn. (Addison-Wesley, Reading, 1974)
27. P.D. Miller, *Applied Asymptotic Analysis*. Graduate Studies in Mathematics, vol. 75 (Am. Math. Soc., Providence, 2006)
28. A. Erdélyi, *Asymptotic Expansions* (Dover, New York, 1987)
29. R. Wong, *Asymptotic Approximations of Integrals* (SIAM, Philadelphia, 2001)
30. J. Wojdylo, Computing the coefficients in Laplace's method. SIAM Rev. **48**, 76 (2006). While [29] in Sect. II.1 describes the classical way of computing the coefficients c_s by series inversion, this article discusses a modern explicit method
31. F.J.W. Olver, *Asymptotics and Special Functions* (Peters, Wellesley, 1997)
32. S. Wolfram, Wolfram Mathematica. http://www.wolfram.com
33. N. Bleistein, R.A. Handelsman, *Asymptotic Expansion of Integrals* (Holt, Reinhart and Winston, New York, 1975)
34. L.D. Landau, E.M. Lifshitz, *Course in Theoretical Physics, Vol. 3: Quantum Mechanics*, 3rd edn. (Pergamon, Oxford, 1991)
35. P.M. Morse, H. Feshbach, *Methods of Theoretical Physics*, vol. 1 (McGraw-Hill, Reading, 1953)
36. A.D. Polyanin, V.F. Zaitsev, *Handbook of Exact Solutions for Ordinary Differential Equations*, 2nd edn. (Chapman & Hall/CRC, Boca Raton, 2003), p. 215
37. J. Boos, *Classical and Modern Methods in Summability* (Oxford University Press, Oxford, 2000)

38. T.J.I'a. Bromwich, *An Introduction to the Theory of Infinite Series* (Macmillan, London, 1955). The collection of formulas and hints contained in the book remains indispensable
39. A. Sofo, *Computational Techniques for the Summation of Series* (Kluwer Academic/Plenum, New York, 2003)
40. M. Petkovšek, H. Wilf, D. Zeilberger, $A = B$ (Peters, Wellesley, 1996)
41. D.M. Priest, On properties of floating-point arithmetics: numerical stability and the cost of accurate computations. PhD thesis, University of California at Berkeley (1992)
42. N.J. Higham, The accuracy of floating point summation. SIAM J. Sci. Comput. **14**, 783 (1993)
43. J. Demmel, Y. Hida, Accurate and efficient floating point summation. SIAM J. Sci. Comput. **25**, 1214 (2003)
44. W. Kahan, Further remarks on reducing truncation errors. Commun. ACM **8**, 40 (1965)
45. P. Linz, Accurate floating-point summation. Commun. ACM **13**, 361 (1970)
46. T.O. Espelid, On floating-point summation. SIAM Rev. **37**, 603 (1995)
47. I.J. Anderson, A distillation algorithm for floating-point summation. SIAM J. Sci. Comput. **20**, 1797 (1999)
48. G. Bohlender, Floating point computation of functions with maximum accuracy. IEEE Trans. Comput. **26**, 621 (1977)
49. D. Laurie, Convergence acceleration, in *The SIAM 100-Digit Challenge. A Study in High-Accuracy Numerical Computing*, ed. by F. Bornemann, D. Laurie, S. Wagon, J. Waldvögel (SIAM, Philadelphia, 2004), pp. 227–261
50. C. Brezinski, M.R. Zaglia, *Extrapolation Methods* (North-Holland, Amsterdam, 1991)
51. C. Brezinski, Convergence acceleration during the 20th century. J. Comput. Appl. Math. **122**, 1 (2000)
52. E.J. Weniger, Nonlinear sequence transformations for the acceleration of convergence and the summation of divergent series. Comput. Phys. Rep. **10**, 189 (1989)
53. K. Knopp, *Theory and Application of Infinite Series* (Blackie, London, 1951)
54. http://www.nr.com/webnotes?5. The implementation described here is particularly attractive, as it automatically changes N and J such that the required maximum error is achieved most rapidly
55. H. Cohen, F.R. Villegas, D. Zagier, Convergence acceleration of alternating series. Exp. Math. **9**, 4 (2000)
56. H.H.H. Homeier, Scalar Levin-type sequence transformations. J. Comput. Appl. Math. **122**, 81 (2000)
57. GSL (GNU Scientific Library). http://www.gnu.org/software/gsl
58. G. Marsaglia, Evaluation of the normal distribution. J. Stat. Softw. **11**, 1 (2004)
59. J.F. Hart et al., *Computer Approximations* (Wiley, New York, 1968)
60. W.J. Cody, Rational Chebyshev approximations for the error function. Math. Comput. **23**, 631 (1969)
61. J.R. Philip, The function inverfc θ. Aust. J. Phys. **13**, 13 (1960)
62. L. Carlitz, The inverse of the error function. Pacific J. Math. **13** (1963)
63. O. Vallée, M. Soares, *Airy Functions and Applications to Physics* (Imperial College Press, London, 2004)
64. G. Szegö, *Orthogonal Polynomials* (Am. Math. Soc., Providence, 1939)
65. G.N. Watson, *Theory of Bessel Functions* (Cambridge University Press, Cambridge, 1922)
66. W.R. Gibbs, *Computation in Modern Physics* (World Scientific, Singapore, 1994). Sections 12.4 and 10.4

http://www.springer.com/978-3-642-32477-2

Computational Methods for Physicists
Compendium for Students
Sirca, S.; Horvat, M.
2012, XX, 716 p., Hardcover
ISBN: 978-3-642-32477-2