Computing Integrals of Analytic Functions to High Precision

Jörg Waldvogel, Seminar for Applied Mathematics,
Swiss Federal Institute of Technology ETH, CH-8092 Zurich

July, 2008

Abstract

We present a set of numerical quadrature algorithms which typically show exponential convergence for analytic integrands, even in the presence of integrable boundary singularities. The algorithms are based on mapping the integration interval onto the entire real axis, together with suitable transformations of the integrand, preferably to a doubly-exponentially decaying function. The transformed integrals are approximated efficiently by the trapezoidal rule; the approximation error may be analyzed by means of Fourier theory.

This method results in a practicable algorithm for computing analytic integrals to a precision of hundreds – or thousands – of digits. Such high precision may prove meaningful for, e.g., identifying new numbers (defined by integrals) with combinations of known mathematical constants [1, 3]. An elegant, almost fully automated experimental implementation in the language PARI/GP is given.

Key words: Numerical quadrature, analytic functions, singularities, transformation method, doubly-exponential decay, trapezoidal rule.

1 Introduction

Many well-known mathematical constants may be expressed as integrals of simple functions, e.g.

$$\pi = \int_{-\infty}^{\infty} \frac{d\,x}{1+x^2}\,.$$

Such integrals may even be regarded as the definition of the number under consideration. Whereas a few digits of precision (e.g., 16 digits, to be generous) may suffice for most practical applications, it is a legitimate quest to know fundamental mathematical constants to much higher precision.

A second reason for being interested in high-precision approximation of definite integrals is a possible identification of new mathematical constants in terms of more fundamental, previously known numbers. This has recently become feasible on the basis of high-precision numerical computaions alone, by means of so-called *integer relation detection algorithms* [3]. For example, inspired by a problem in the American Mathematical Monthly [2], and using the software [3], D.H. Bailey, Jonathan Borwein and Greg Fee conjectured the previously unknown closed form

$$\int_0^1 \frac{\arctan(\sqrt{x^2+2})}{(x^2+1)\,\sqrt{x^2+2}}\,dx = \frac{5}{96}\,\pi^2\,.$$

Later this result was rigorously established.

The conjectural discovery of previously unknown relationships (and their subsequent proof, of course) is in the same line. Consider, e.g., the number

$$q := (-\frac{1}{4})\,! = \Gamma(\frac{3}{4}) = \int_0^\infty e^{-x}\,x^{-1/4}\,dx\,. \tag{1}$$

Not to a complete surprise, there is a relationship between q and elliptic integrals. Precise numerics and the software [3] yields the conjecture

$$q^2 = \frac{\pi^{3/2}}{2\,\mathrm{K}(\frac{1}{\sqrt{2}})}\,, \tag{2}$$

where K is the complete elliptic integral of first kind (equivalently, this relation may also be written as $q^2 = \sqrt{\pi}\,\mathrm{agm}(1, 1/\sqrt{2})$ with $\mathrm{agm}(a, b)$ being the arithmetic-geometric mean of a and b). This relation is known to specialists.

Finally, we mention that other processes, such as limits or infinite sums, may be the source of new mathematical constants. Often such processes may be fomulated as definite integrals, see, e.g., [5], Chapter 3. Therefore, the envisioned technique involving high-precision evaluation of definite integrals is expected to have a wide range of applications.

The tool for efficiently approximating definite integrals to high precision is numerical quadrature, a topic which has been extensively studieded in the literature on numerical analysis. A comprehensive treatment, together with an 80-page list of references may be found in Davis and Rabinowitz (second edition, 1984) [6].

In this paper we propose a unified numerical technique which is well adapted to the high-precision evaluation of definite integrals of analytic functions. We propose to use monotonically increasing combinations of elementary functions in order to map the integration interval to the entire real axis. According to the nature of possible (integrable!) boundary singularities, further transfomations are appropriately chosen in order to obtain an integrand of double-exponential decay. This transformed integral is finally approximated by means of the trapezoidal rule with an appropriate step $h > 0$.

For a required precision of d digits the length of the support grows only as $O(\log(d)\,)$. With respect to the step size h the discretization error is exponentially small, typically $O(\exp(-c\,\omega)$, sometimes $O(\exp(-c\,\omega/\log(\omega))$ or $O(\exp(-c\,\sqrt{\omega})$ with some $c > 0$, $\omega := 2\,\pi/h$. The suggested procedure is completely flexible with respect to the nature of the boundary singularities. We will conclude with an experimental code in PARI/GP which is able to automatically handle a large suite of test integrals to a precision of d digits, where d may be specified as an arbitrary parameter. The computation time scales as a (small) power of d.

2 Intervals and Singularities

In order to handle one-dimensional integrals in some generality it is necessary to consider three types of integration intervals D: the entire real line $\mathbb{R}$, semi-infinite intervals, and finite intervals. For simplicity, and with no loss of generality, we restrict ourselves to the following standardized versions of those intervals:

$$D = \begin{cases} (-\infty,\, \infty)\,, \\ (0,\, \infty)\,, \\ (0,\, b) \quad \text{with} \quad 0 < b < \infty\,. \end{cases} \tag{3}$$

We now consider the integral

$$C = \int_D f(x)\, dx\,, \tag{4}$$

where f is defined and analytic in the open interval D and is such that the integral (2) over the finite or infinite interval D exists. The reduction to the standard form (2), (1) (e.g. by a translation) is left up to the user.

At this point it is necessary to exclude highy oscillating integrands f. Although the methods of this paper may be applicable in particular situations (e.g. if the integrand is an entire function), more advanced techiques of complex analysis are needed in general for handling such integrands. Without giving precise definitions here we refer to Section 6 for a sketch of this technique.

Since integrable singularities at the finite boundaries are explicitly allowed it is necessary to introduce a rough classification of the possible growth or decay rates of f near an interval boundary. We use $x \longrightarrow +\infty$ and $x \longrightarrow +0$ as models for an infinite and a finite boundary.

We first restrict ourselves to infinite intervals and consider the right boundary in the limit $x \longrightarrow +\infty$. Let $p,\ 0 < p < x$ be a fixed intermediate point and consider the indefinite integral

$$F(x) := \int_p^x f(\xi) d\xi \tag{5}$$

which satisfies the initial-value problem

$$F'(x) = f(x)\,, \quad F(p) = 0\,, \tag{6}$$

where primes denote differentiation with respect to x. The integrand f is *integrable* as $x \longrightarrow +\infty$ if $\lim_{x\to+\infty} F(x)$ exists. A popular, simple integrable model function of this class leads to

$$F(x) = F_0(x) := \text{const} - \frac{1}{\alpha}\, x^{-\alpha}\,, \quad \alpha > 0\,, \tag{7}$$

resulting in the integrable integrand

$$f_0(x) = F_0'(x) := x^{-\alpha-1}\,. \tag{8}$$

Next, we define the family of model integrands $f_n(x)$ of *exponential type* $n,\ n \in \mathbb{Z}$ by Equ. (5) and the recurrence

$$F_{n+1}(x) = F_n(e^x)\,, \quad f_n(x) = F_n'(x)\,, \tag{9}$$

which implies

$$f_{n+1}(x) = e^x \; f_n(e^x)\,. \tag{10}$$

Clearly, for every integer n the integrand $f_n(x)$ is integrable as $x \to +\infty$, whereby the lower limit p has to be chosen appropriately. In particular:

$$\begin{aligned}
F_0(x) =&\ \text{const} - \ \tfrac{1}{\alpha}\ x^{-\alpha}\,, \quad \alpha > 0\,, \\
F_1(x) =&\ \text{const} - \ \tfrac{1}{\alpha}\ e^{-\alpha\, x}\,, \quad F_{-1}(x) = \text{const} - \frac{1}{\alpha}\ (\log x)^{-\alpha}\,, \\
F_2(x) =&\ \text{const} - \ \tfrac{1}{\alpha}\ e^{-\alpha\, e^x}\,, \quad F_{-2}(x) = \text{const} - \frac{1}{\alpha}\ (\log\log x)^{-\alpha}
\end{aligned} \tag{11}$$

and

$$\begin{aligned}
f_0(x) =&\ \ x^{-\alpha-1}\,, \\
f_1(x) =&\ \ e^{-\alpha\, x}\,, \quad f_{-1}(x) = x^{-1}\ (\log x)^{-\alpha-1}\,, \\
f_2(x) =&\ \ e^{x-\alpha\, e^x}\,, \quad f_{-2}(x) = x^{-1}\ (\log x)^{-1}\ (\log\log x)^{-\alpha-1}\,.
\end{aligned} \tag{12}$$

With increasing exponential type n the integrand $f_n(x)$ decays faster as $x \to +\infty$; for strongly negative n the integrand $f_n(x)$ approaches non-integrability. This classification is by far not complete; in fact, even the functions expressible as finite compositions of exponentials, logarithms and shifts form an uncountable set. However, already the five model integrands listed in Equs. (10) cover most of the integrals encountered in the classical literature. As it was first observed in 1974 by Takahasi and Mori [9], analytic integrands of the exponential type $n = 2$, i.e. integrands of the decay rate displayed by f_2 in Equ. (10), referred to as *doubly exponential decay*, allow for particularly efficient numerical quadrature algorithms.

In order to carry over this crude classification to finite boundaries with possibly singular integrands we use the map $x \mapsto t = \frac{1}{x}$ mapping the infinite boundary $x \to +\infty$ to the finite boundary $t \to +0$. By using the substitution $\xi = 1/\tau$ the integral (3) becomes

$$F(x) = F(\frac{1}{t}) = \int_p^{1/t} f(\xi)\, d\xi = \int_t^{1/p} g(\tau)\, d\tau \ \ =: \ G(t)\,, \tag{13}$$

where

$$g(\tau) = f(\frac{1}{\tau})\ \tau^{-2}\,. \tag{14}$$

By applying (12) to the model functions f_n from Equ. (10) and denoting the resulting integrands by $g_n(t)$ we obtain

$$\begin{aligned} g_0(t) &= \quad t^{\alpha-1}\,, \\ g_1(t) &= \quad t^{-2}\,e^{-\alpha/t}\,, \quad g_{-1}(t) = t^{-1}\,(\log\frac{1}{t})^{-\alpha-1}\,, \\ g_2(t) &= \quad t^{-2}\,e^{\frac{1}{t}-\alpha\,e^{1/t}}\,, \quad g_{-2}(t) = t^{-1}\,(\log\frac{1}{t})^{-1}\,(\log\log\frac{1}{t})^{-\alpha-1}\,. \end{aligned} \tag{15}$$

Clearly, $g_n(t)$ are integrands with an integrable boundary singularity as $t \to +0$; we define $g_n(t)$ as the model integrand of exponential type n as $t \to +0$. Note that a finite boundary with a regular integrand is modeled by the exponential type 0, i.e. by $g_0(t)$ with $\alpha = 1$.

Given an interval and an explicitly defined integand f, the model functions in (10), (13) allow to read off the exponential types of the integrand at the two interval bondaries. Note that "algebraic" (as opposed to "exponential") transformations $x \mapsto x^\beta,\ t \mapsto t^\beta,\ \beta > 0$ in Equs. (11) or (13) do not change the exponential type of f_n or g_n.

3 Transformations

Consider now the integral (2) of f over the interval D defined in (1), where the type of D and – if necessary – the value $b \in (0, +\infty)$ are given, and f is analytic and integrable in D. We assume that the exponential types n_l and n_r (both integers) of f at the left and right boundaries are known. In the following we assume $n_l \leq 2,\ n_r \leq 2$ for simplicity. Most of the integrals found in the classical literature will be covered; nevertheless, this restrictioon may easily be overcome.

Following the general concept of transformation methods (see, e.g., [8]) we will use a mapping of the form

$$x = \Phi(t), \quad t \in \mathbb{R} \tag{16}$$

in order to transform the integral (2) to an integral over the entire real line $\mathbb{R}$. Furthermore, we will follow the recommendation of Takahasi and Mori [9] and tune the mapping in such a way that the resulting integrand decays doubly exponentially (i.e. of exponential type 2) on both sides. If D is not the entire real line the mapping (14) must be chosen such that D is the image of $\mathbb{R}$, otherwise (14) must be an isomorphism of $\mathbb{R}$. We thus adopt the strategy of mapping the integration interval D onto $\mathbb{R}$ first; then it suffices to handle integrals over the entire real line. This strategy is exactly opposite to the conventional wisdom of avoiding infinite intervals by mapping them to finite intervals first.

The functions Φ defining the mappings must be defined and monotonically increasing on $\mathbb{R}$. Since we are working with analytic integrands we restrict ourselves to analytic mappings. For practical reasons we choose simple combinations of elementary functions, taking advantage of the highly optimized algorithms for evaluating elementary functions in arbitrary precision which are available in many mathematical software systems.

We suggest to compose the mapping $\Phi: \; t \mapsto x = \Phi(t)$ in three steps E, F, G via the auxiliary variables u and v according to the sequence of mappings

$$t \in \mathbb{R} \overset{E}{\longmapsto} u \in \mathbb{R} \overset{F}{\longmapsto} v \in \mathbb{R} \overset{G}{\longmapsto} x \in D\,. \tag{17}$$

First step. The goal of the outermost mapping, $G : v \mapsto x = G(v)$, is to map the entire v-axis onto D by preserving the exponential types of the integrand at the boundaries of D. By using the definitions

$$\begin{aligned} G_0(v) &:= \; v + \sqrt{1+v^2} = \frac{1}{-v + \sqrt{1+v^2}}\,, \\ G_1(v) &:= \; \frac{G_0(v)}{1+G_0(v)}\,, \end{aligned} \tag{18}$$

where the equivalent second expression in the first line avoids cancellation for $v < 0$, the mapping G will be chosen as

$$x = G(v) = \begin{cases} v & \text{if} \;\; D = (-\infty,\, \infty)\,, \\ G_0(v) & \text{if} \;\; D = (0,\, \infty)\,, \\ b\, G_1(v) & \text{if} \;\; D = (0,\, b)\;; \quad b - x = b\, G_1(-v)\,. \end{cases} \tag{19}$$

The expression for $b-x$ given in the third line allows to compute the distance from the right boundary b in the case of a finite interval D without loss of accuracy.

Second step. For the intermediate mapping, $F : u \mapsto v = F(u)$, we conveniently choose

$$v = F(u) := \sinh(u)\,; \tag{20}$$

every application of F simultaneously increases the exponential type of the integrand at both boundaries by 1. Therefore, the more strongly decaying boundary reaches the exponential type 2 after $m := 2 - \max(n_l, n_r)$ iterations of F. We mention the convenient relations

$$G_0(F(u)) = e^u \,, \quad G_1(F(u)) = \frac{1}{1+e^{-u}} \,, \quad 1 - G_1(F(u)) = \frac{1}{1+e^{u}} \,, \qquad (21)$$

which give simple expressions for the composition of G_0 or G_1 with the first iteration of F.

Third step. For the innermost mapping, $E : t \mapsto u = E(t)$, we suggest to use the definitions

$$E_0(t) := t + e^t \,, \quad E_1(t) := -E_0(-t) = t - e^{-t} \,. \qquad (22)$$

Each application of E_0 increases the right exponential type by 1 and leaves the left exponential type unchanged (and vice versa for E_1). Therefore, the innermost step in the process of composing the mapping Φ is given by

$$u = E(t) = \begin{cases} t & \text{if } \; n_l = n_r \,, \\ [E_0(t)]^n & \text{if } \; n_l > n_r \,, \\ [E_1(t)]^n & \text{if } \; n_l < n_r \,, \end{cases} \qquad (23)$$

where $n := |n_l - n_r|$, and the exponent indicates the number of compositional iterations.

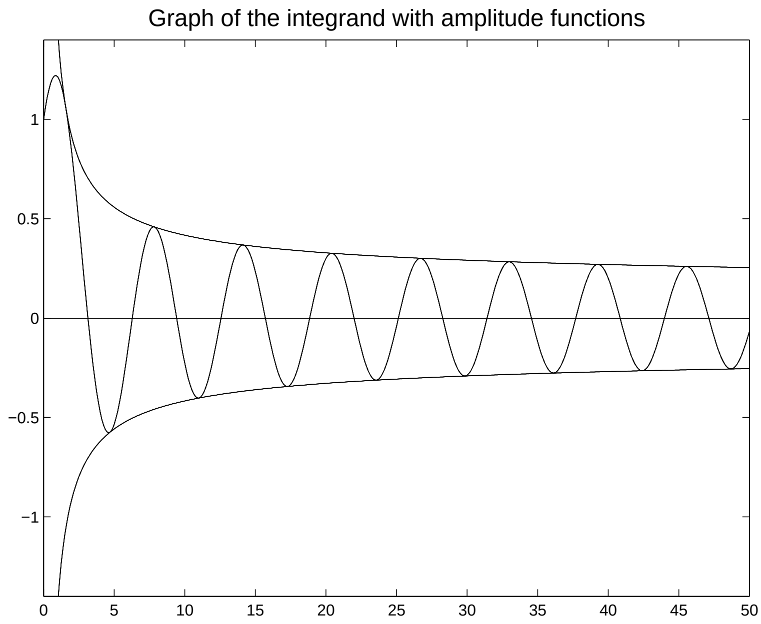

Figure 1: Graph of the integrand in (24) with a slowly decaying amplitude function

4 Highly Oscillatory Analytic Integrands

The best strategy for evaluating integrals of highly oscillatory analytic functions is to *avoid* the regions of oscillatory behaviour as much as possible. For many of the usually encountered oscillatory analytic integrands regions of non-oscillatory behaviour may readily be found; then Cauchy's theorem and calculus of residues as a *paper-and-pensil* preparatory step yields a non-oscillatory integral of the same value.

The efficiency and simplicity of this technique will be demonstrated by means of a constructed example which superficially appears to be rather nasty (cf. the graph in Figure 1): Let

$$I(a,b) := \int_a^b \frac{\sin z}{\log(1+z)}\ dz\ , \tag{24}$$

find $I(0,\,\infty) = \lim_{a\to+0,\ b\to+\infty} I(a,\,b)$. The naive approach is to write $I(0,\,\infty)$

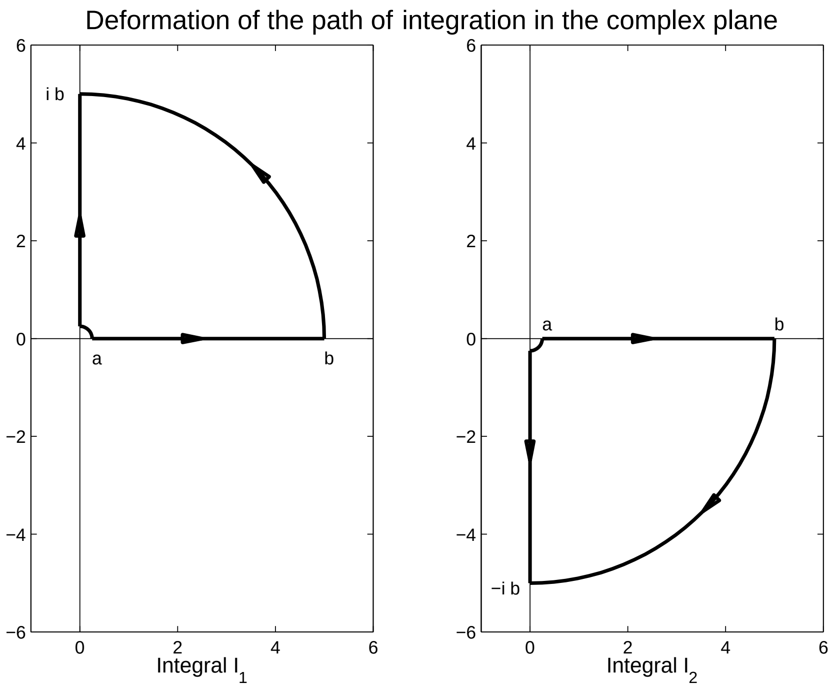

Figure 2: Extension and deformation of the path of integration of the integrals I_1 and I_2

as a series,

$$I(0,\infty) = \sum_{n=0}^{\infty} \int_{n\pi}^{(n+1)\pi} \frac{\sin z}{\log(1+z)}\, dz \ , \tag{25}$$

of terms of alternating signs tending to zero. Therefore, according to Cauchy's theorem (on infinite alternating sums), the series converges (albeit extremely slowly), and the integral $I(0,\infty)$ exists. With luck, extrapolation techniques applied to the partial sums of the series (25) may yield $I(0,\infty)$ to a limited accuracy, see, e.g., [5].

In our more advanced approach we observe that the integrand is non-oscillatory on the imaginary axis $z = i\,y,\ y \in \mathbb{R}$. In order to deform the path of integration to the imaginary axis, while still working with existent integrals, we use Euler's famous identity $e^{iz} = \cos z + i\,\sin z$ for splitting up

$\sin z$ as $\sin z = (e^{iz} - e^{-iz})/2i$. Then we have

$$I(a, b) = I_1(a, b) - I_2(a, b) \tag{26}$$

where

$$I_1(a, b) := \int_a^b \frac{e^{iz}}{2\,i\,\log(1+z)}\,dz\ , \quad I_2(a, b) := \int_a^b \frac{e^{-iz}}{2\,i\,\log(1+z)}\,dz\ . \tag{27}$$

We now consider each term of (26) separately and try to modify the path of integration in such a way that it may be deformed into a subset of the imaginary axis without changing the value of the corresponding integral. In the case of I_1 (see Figure 2, left frame) consider the cicular arc $z = b\ e^{i\varphi}$, $0 < \varphi < \pi/2$ from b to $i\,b$ as an extension of the original path of integration, i.e. of the real interval $[a,\ b]$. By estimating from above the corresponding integral

$$C_1(b) := \int_0^{\pi/2} \frac{b\,e^{i\,(\varphi+z)}}{2\ \log(1+z)}\,d\varphi \quad \text{with} \quad z = b\ e^{i\varphi} \tag{28}$$

over this arc, we will show that $C_1(b)$ vanishes in the limit $b \to \infty$. For $b > 1$ we obtain

$$|C_1(b)| \le \int_0^{\pi/2} \frac{b\ e^{-b\ \sin\varphi}}{2\ \log\ b}\,d\varphi \le \int_0^{\pi/2} \frac{b\ e^{-b\cdot 2\,\varphi/\pi}}{2\ \log\ b}\,d\varphi = \frac{\pi}{4\ \log\ b}\,(1 - e^{-b})\ ; \tag{29}$$

therefore we indeed have $\lim_{b\to\infty} C_1(b) = 0$. It is seen that a similar effect is achieved with the integral I_2 by extending the path of integration by the lower arc $z = b\ e^{i\varphi}$, $0 > \varphi > -\pi/2$ (see Figure 2, right frame). In general, extensions of the path of integration should be attempted in regions of small values of the integrand.

Since both integrands of Equ. (27) are free of singularities in the regions delineated in Figure 2, the integrals over the extended paths from a to $i\,b$ and from a to $-i\,b$, respectively, may be replaced by the integrals over the direct connections along the imginary axis, according to Cauchy's theorem. Hereby the origin must be avoided, as it is indicated by the small circular arcs in Figure 2. Reason: although the original integrand in (24) is analytic in $z = 0$, it is safer to speak of a *removable singularity*; in fact, the removable singularity spawns poles of first order at $z = 0$ in the integrands of I_1 and I_2 of Equ. (27).

Collecting everything, we obtain

$$I(0,\,\infty) = \lim_{\substack{a\to+0\\ b\to+\infty}} \left(\int_a^{i\,b} \frac{e^{iz}}{2\,i\,\log(1+z)}\,dz \;-\; \int_{-a}^{-i\,b} \frac{e^{-iz}}{2\,i\,\log(1+z)}\,dz \right), \quad (30)$$

where the integrations are along the imaginary axis, sparing the origin. By using calculus of residues, the contribution of the origin in the limit $a \to +0$ is found to be

$$\frac{2\pi\,i}{4}\,\mathrm{Res}\,\frac{e^{iz}}{2\,i\,\log(1+z)}\bigg|_{z=0} + \frac{2\pi\,i}{4}\,\mathrm{Res}\,\frac{e^{-iz}}{2\,i\,\log(1+z)}\bigg|_{z=0} = \frac{\pi}{2}\,.$$

Parametrizing the integrals in (30) by $y \in \mathbb{R}$ according to $z = i\,y$ or $z = -i\,y$, respectively, finally yields

$$I(0,\,\infty) = \frac{\pi}{2} + \int_0^{\infty} \mathrm{Re}\,\frac{1}{\log(1+i\,y)}\,e^{-y}\,dy\,. \quad (31)$$

For safely evaluating the integrand we recommend the identity (35) of the following section, which is valid for complex arguments as well. The exponential types of the left and right boundaries are 0 and 1, respectively; the transformations to $\mathbb{R}$ and doubly exponential decay in the new integration variable s, as suggested in Section 3 are therefore

$$y = e^t\,, \quad t = s - e^{-s}\,. \quad (32)$$

As an example, the automatic evaluation of (31) to a requested accuracy of 45 digits by the routines of the next section yields

$$I(0,\,\infty) = \frac{\pi}{2} + 0.470222288352813267454402348362864678273093140. \quad (33)$$

Using step bisection totally takes 27, 50, 92, or 172 evaluations of the integrand, respectively, in order to obtain 6, 12, 23, or 45 correct digits. This nicely reflects exponential convergence, i.e. halving the step doubles the number of correct digits.

5 An Implementation in PARI/GP

Any mathematical software offering arithmetics – including elementary functions – in arbitrary precision may serve as a tool for implementing the algorithm described in the previous sections. We choose PARI/GP [4] because it is available as freeware and because it has a simple and versatile command structure. Furthermore, it executes surprisingly fast.

For a crude checkout of our algorithm we will use a test suite of 25 integrals I_1 through I_{25} over intervals of all three types:

$$I_1 = \int_0^1 dx, \qquad I_2 = \int_0^1 e^x\, dx,$$

$$I_3 = \int_0^1 x^{63}\, dx, \qquad I_4 = \int_0^1 \sin(8\,\pi\, x^2)\, dx,$$

$$I_5 = \int_0^1 \frac{1}{1+\exp(x)}\, dx, \qquad I_6 = \int_0^1 \frac{1}{x+0.5}\, dx,$$

$$I_7 = \int_0^1 \sqrt{12.25-(5\,x-3)^2}\, dx, \qquad I_8 = \int_0^1 \frac{10}{1+(10\,x-4)^2}\, dx,$$

$$I_9 = \int_0^1 \frac{1}{\sqrt{x\,(1-x)}}\, dx, \qquad I_{10} = \int_0^1 \frac{\cos(2\pi x)}{\sqrt{1-x}}\, dx,$$

$$I_{11} = \int_0^1 x^{-3/4}\,(1-x)^{-1/4}\,(3-2\,x)^{-1}\, dx, \qquad I_{12} = \int_0^1 x^{-3/4}\, L(x)^{-3/4}\, dx,$$

$$I_{13} = \int_0^1 x^{0.21}\, \sqrt{L(x)}\, dx,\ \ L(x) := \log\frac{1}{x}\,, \qquad I_{14} = \int_0^1 L(x)^{\sqrt{L(x)}}\, dx,$$

$$I_{15} = \int_0^1 x^{0.6}\, L(x)^{-0.7}\, \cos(2\,L(x))\, dx, \qquad I_{16} = \int_0^\infty \frac{1}{x^2+\exp(4\,x)}\, dx,$$

$$I_{17} = \int_0^\infty \frac{1}{1+x^2+\frac{x^4}{1+exp(-x)}}\, dx, \qquad I_{18} = \int_0^\infty \frac{1}{x^{2/3}+x^{3/2}}\, dx,$$

$$I_{19} = \int_0^\infty e^{-\sqrt{x}}\, dx, \qquad I_{20} = \int_0^\infty \mathrm{Re}\, \frac{e^{-x}}{\log(1+i\,x)}\, dx,$$

$$I_{21} = \int_{-\infty}^\infty \frac{1}{1+x^2+\frac{x^4}{1+exp(-x)}}\, dx, \qquad I_{22} = \int_{-\infty}^\infty \frac{1}{x^2+\exp(4\,x)}\, dx,$$

$$I_{23} = \int_{-\infty}^\infty (1+x^2)^{-5/4}\, dx, \qquad I_{24} = \int_{-\infty}^\infty \exp(-\sqrt{1+x^2})\, dx,$$

$$I_{25} = \int_{-\infty}^\infty \frac{1}{x^2+\frac{1}{\cosh(x)}}\, dx$$

The integrands are analytic in the open intervals; some of them have integrable boundary singularities. For convenience the integrands are chosen as simple expressions in terms of elementary functions; otherwise they are more or less arbitrary. The first eight integrals over the finite interval $(0, 1)$ have regular integrands. However, I_3, I_4, I_8 may challenge the algorithm by a sudden steep slope, oscillatory behaviour, or a narrow spike. In the integrals I_9 through I_{15} over the interval $(0, 1)$ both boundaries carry algebraic or logarithmic singularities, all of exponential type 0. In some of the integrals I_{16} through I_{25} over half-infinite and doubly infinite intervals the finite boundary carries an algebraic singularity (I_{18}, I_{19}). The intricate details of the behaviour of the intgrands at infinity hardly affect the performance of the algorithm.

For coding the original integrands (in any language) a few simple and well-known precautions need to be taken in order to avoid error stops due to overflow and for reducing the numerical effects of round-off errors and of cancellation:

(i) The integrand needs to be evaluated in such a way that no overflow or underflow interrupts occur. The exponential function `exp(x)` is particularly prone to this. Our implementation assumes that `exp(x)` works without an error stop for every $x <$ `maxexp` where `maxexp` > 0 is some large overflow limit. In the underflow region $x < -$`maxexp` we assume `exp(x)` $:= 0$ without producing an error.

In contrast to this, however, the current version of PARI/GP produces an error stop in calls of `exp(x)` with $x < -$`maxexp`. In the code below we use the user-defined function `exp0(x)` instead of `exp(x)` in order to avoid this problem. In the current version of PARI/GP the overflow limit is about `maxexp` $= 372\,130\,000$. In our examples the much less prodigious value of `maxexp` $= 10\,000$ will suffice.

(ii) For integrals over finite intervals $(0, b)$ with boundary singularities the user has to make sure that both distances from the boundaries, x and $x_1 := b - x$, are known accurately, even if one of them is small. Both distances need to be transmitted after having been computed independently as stated in the third line of Equ. (19). The original integrand must be coded by using the transmitted values of the two distances.

(iii) Avoid expressions prone to cancellation, such as $1 - \cos(x)$. Use numerically stable forms instead, mostly stemming from well-known mathematical identities, e.g.

$$1 - \cos(x) = 2\,\sin^2(\frac{x}{2})\ . \tag{34}$$

A less known example is

$$\log(1+x) = \mathrm{asinh}(\mathrm{x}\,\frac{1+\mathrm{x}/2}{1+\mathrm{x}})\ , \tag{35}$$

useful in some of the intgrands of the test suite or, e.g., for accurately computing $(1+1/n)^n$ for large values of n.

Below is the sample of code `fct` for the entire test suite in PARI/GP. Following Remark (ii) above both distances `x` $= x$ and `x1` $= b - x$ from the interval boundaries are transmitted to `fct`, together with the integer variable `ind` defining the specific integrand to be considered. In our code all variables not listed as function parameters are global variables. The codes for the integrands of $I_{16}, I_{17}, I_{21}, I_{22}, I_{25}$ exemplify a technique for avoiding overflow. The codes for the integrands of $I_9, I_{10}, I_{12}, I_{20}$ offer examples of the implementation of Remark (iii) to avoid numerical cancellation. No attempt is made for carrying out these simple precautions automatically. With I_{20} the value reported in (33) of the integral (31) is computed. The function `exp0` defined below is explained in Remark (i) above.

```
{fct(x,x1,ind) =
if(ind== 1, f=1);                if(ind== 2, f=exp0(x));
if(ind== 3, f=x^63);             if(ind== 4, f=sin(8*Pi*x^2));
if(ind== 5, f=1/(1+exp0(x)));    if(ind== 6, f=1/(x+1/2));
if(ind== 7, f=sqrt(49/4-(5*x-3)^2)); if(ind==8,f=10/(1+(10*x-4)^2));
if(ind== 9, f=1/sqrt(x*x1));     if(ind==10,f=cos(2*Pi*x1)/sqrt(x1));
if(ind==11, f=(x*x1)^(-1/4)/(3-2*x)/sqrt(x));
if(ind>=12 & ind<=15, L = if(x<3/4, -log(x), asinh((x1-x1^2/2)/x)));
if(ind==12, f=(x*L)^(-3/4));     if(ind==13, f=x^(21/100)*sqrt(L));
if(ind==14, f=L^sqrt(L));  if(ind==15,f=x^(3/5)*L^(-7/10)*cos(2*L));
if(ind==16, if(x<maxexp, f=1/(x^2+exp0(4*x)), f=0));
if(ind==17, d=1+x^2; if(x>-maxexp, d=d+x^4/(1+exp0(-x))); f=1/d);
if(ind==18, f=1/(x^(2/3)+x^(3/2))); if(ind==19, f=exp0(-sqrt(x)));
if(ind==20, f=exp(-x)*real(1/asinh((I*x-x^2/2)/(1+I*x))));
if(ind==21, if(x<maxexp, f=1/(x^2+exp0(4*x)), f=0));
```

```
if(ind==22, d=1+x^2; if(x>-maxexp, d=d+x^4/(1+exp0(-x))); f=1/d);
if(ind==23, f=(1+x^2)^(-5/4));  if(ind==24, f=exp0(-sqrt(1+x^2)));
if(ind==25, d=x^2; if(abs(x)<maxexp, d=d+1/cosh(x)); f=1/d); f}

{exp0(x) = if(x>-maxexp, exp(x), 0)}
```

The subroutine `Data` defines the interval types, the exponential types of the boundaries and reference values for the exact integrals. The interval length and type are encoded in the single quantity $b :=$ `bvec[ind]`, where a value $b \in (0, \infty)$ codes for the finite integration interval $(0, b)$, $b = \infty$ codes for the half-infinite interval $(0, \infty)$, and $b = 0$ codes for the doubly infinite interval $(-\infty, \infty)$. As a mnemonic, one may think of the interval code b as the sum of the two interval boundaries, where $(-\infty) + (\infty) := 0$. In absence of a representation of ∞ in PARI/GP we arbitrarily use any number $\geq$ `inf` $:= 2^{30}$ to represent ∞.

For simplicity the vectors `left, right` of the exponential types must be defined by the user. An automatic determination of these exponential types from values of the integrands is not difficult, however.

For conveniently assessing the the errors of the final results we endow `Data` with reference values for the test integrals. Only for the integrals $I_{14}, I_{16}, I_{17}, I_{20}, I_{21}, I_{22}, I_{25}$ no explicit expressions could be found; we include 67-digit approximations. For all other integrals the PARI codes for the closed forms may be found in `Data`. Note that the quarter-integer values of the gamma function (I_{12}) and the arithmetic-geometric mean agm (I_{23}) are connected via Equ. (2) and the relation $(-\frac{1}{4})!\,(\frac{1}{4})! = \pi/\sqrt{8}$.

The intgrals I_4 and I_{10} are related to the Fresnel integrals $S(z), C(z)$, see [1], Section 7.3,

$$I_4 = \frac{S(4)}{4}\,, \quad I_{10} = C(2)\,.$$

These, in turn, are connected with the complementary error function erfc via the complex relation [1], 7.3.22,

$$C(x) + i\,S(x) = \frac{1+i}{2}\left(1 - \operatorname{erfc}(\frac{(1-i)}{2}\sqrt{\pi}\,x)\right)\,. \qquad (36)$$

An appropriate algorithm for evaluating erfc to high precision for complex arguments z with $\operatorname{Re} z > 0$ may be based on the continued fraction [1], 7.1.14,

$$\operatorname{erfc}(z) = \frac{e^{-z^2}}{\sqrt{\pi}} \cdot \frac{1\,|}{|\,z} + \frac{\frac{1}{2}\,|}{|\,z} + \frac{1\,|}{|\,z} + \frac{\frac{3}{2}\,|}{|\,z} + \frac{2\,|}{|\,z} + \frac{\frac{5}{2}\,|}{|\,z} + \ldots . \tag{37}$$

The procedure CIS below evaluates Equ. (36) to the current working precision, where for obtaining a convenient stopping criterion forward evaluation of the continued fraction (37) is used.

Of particular interest are

$$\begin{aligned} I_{11} &= \pi\, 2^{1/2}\, 3^{-3/4} \\ I_{15} &= \Gamma(0.3)\, \operatorname{Re}\left((1.6 + 2\,i)^{-0.3}\right) \\ I_{24} &= 2\, K_1(1) \quad \text{(modified Bessel function of 2nd kind)}, \end{aligned}$$

expressions that could not be found by Maple. Mathematica cannot find I_{24} either but correctly finds I_{15}. I_{11} is reported incorrectly by including a superfluous factor of $-i$. Regarding the remarks in Section 1 it is interesting to note that the Inverse Calculator [3] was able to correctly identify 8 among the 16 nontrivial closed forms, based on 20-digit approximations alone.

```
{Data() =
bvec =[1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,inf,inf,inf,inf,inf,0,0,0,0,0];
left =[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0, 0 , 0 , 0 , 0 , 0 ,0,0,0,1,0];
right=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0, 1 , 0 , 0 , 1 , 1 ,0,1,0,1,0];
nb=length(bvec); rpi=sqrt(Pi);
\\67 digits are given below
ex14=2.595132461375600632865453596449932552478419572986913447335431561632;
ex16=0.246187594844969988455735993593299408678383079390865062727850450893;
ex17=0.941290135428197484815318673981177460494439206459001328476727087905 6;
ex20=0.470222288352813267454402348362864678273093140584333920545401161923 5;
ex21=2.352556240156597452006557266749073664089683637842586132781113858813;
ex22=3.160322869748470870443040097285096584359515878397156175719639506290;
ex25=3.474037783744718107096426978387195272872810866563003223677556283784;
exact=[1, exp(1)-1, 1/64, imag(CIS(4)[1])/4, 1/2-log(cosh(1/2)),\
log(3), (sqrt(132)+sqrt(117)+49/2*(asin(6/7)+asin(4/7)))/20,\
Pi-atan(10/23), Pi, real(CIS(2)[1]), Pi*sqrt(2/3/sqrt(3)),\
sqrt(2)*gamma(1/4), sqrt(Pi)*500/1331, ex14,\
gamma(3/10)*real((8/5+2*I)^(-3/10)),ex16,ex17,6/5*Pi*sqrt(2-2/sqrt(5)),\
2, ex20, ex21, ex22, 2*agm(1,sqrt(2)), 2*besselk(1,1), ex25]}
```

```
{CIS(x) = z=rpi/2*(1-I)*x; \\[1] 7.3.22, erfc by 7.1.14, forw.eval.
  u0=[0,1.0]; u=[1.0,z]; q0=u0[1]/u0[2]; q=u[1]/u[2]; n=0;
  while(q!=q0, n=n+1; u1=z*u+n/2*u0; u0=u; u=u1; q0=q; q=u[1]/u[2]);
  [(1+I)/2*(1-exp(-z^2)/rpi*q), n]}
```

The essential operations of our algorithm are carried out in the subroutines `FCT`, `trapez` and `halfstep`. The function `FCT` evaluates the transformed integrand, where the parameters `b`, `L`, `R` specify the interval type and the exponential types of the boundary singularities.

```
{FCT(t,b,L,R,ind) = D=sign(L-R); dt=1;
\\ b: interval type: b=0:(-inf,inf), 0<b<inf:(0,b), b=inf:(0,inf)
\\ L,R = boundary types, restricted to -1 (subpolyn.) .. 1 (exp.)
for(i=1,abs(L-R), eDt=exp(D*t); dt=dt*(1+eDt); t=t+D*eDt);
for(i=1,1-max(L,R), dt=dt*cosh(t); t=sinh(t));
if(b==0, dx=dt*cosh(t); x=sinh(t),
  if(b<inf, e=exp(t)/2; e1=1/4/e; ch=e+e1;
     x=b*e/ch; x1=b*e1/ch; dx=dt*b/2/ch^2,   x=exp(t); dx=dt*x)
); dx*fct(x,x1,ind)}
```

The procedure `trapez` accumulates the trapezoidal sum with step `h` and an arbitrary `offset`. For truncating the trapezoidal sums according to the current working precision the sums are accumulated outwards in both directions, starting from an appropriate inner point, `offset`. Summation is stopped if the addition of two consecutive terms does not change the current sum. Owing to the careful internal handling of guard digits in PARI/GP this direct summation from large to small terms suffices in all cases of the test suite. Backwards summation from small to large absolute values does not significantly improve the accuracy. The procedure returns the trapezoidal sum `S+T` and the number `j+k` of evaluations of the integrand.

```
{trapez(h,offset) = j=0; S00=1.9; S0=1.1; S=0; s=offset+h;
while(S!=S00, j=j+1; s=s-h; S00=S0; S0=S; S=S+h*FCT(s,b,L,R,ind));
                    k=0; T00=1.9; T0=1.1; T=0; t=offset;
while(T!=T00, k=k+1; t=t+h; T00=T0; T0=T; T=T+h*FCT(t,b,L,R,ind));
[S+T, j+k]}
```

Refinement of the trapezoidal sum by step halving and comparison with the exact values is done in `halfstep`. The function call `dig(error)` computes the number of correct digits in a result with a given relative error.

```
{halfstep(tol) =
h=1; sh=0; res=trapez(h,sh); val=res[1]; v0=val+1;
evec=dig(val/ex-1); count=res[2]; cvec=count;
while(abs(val-v0)>tol, h=h/2;
  v0=val; res=trapez(2*h,sh+h); val=(val+res[1])/2; count=count+res[2];
  evec=concat(evec,dig(val/ex-1)); cvec=concat(cvec,count);
)};

{dig(err) = if(err!=0, floor(-log(abs(err))/l10), dec)}
```

The calling program loops through all examples of the test suite and prints the number of correct digits and the accumulated number of function evaluations at every stage. Also, the final approximation of the integral is printed (although not shown in the table of Section 8). For efficiently truncating the step halving algorithm its convergence rate has to be known. For many common integrals, in particular for the integrals in our test suite (except for I_{17} and I_{25}), this convergence rate is $O(\exp(-c/h))$ with some constant $c > 0$ or close to this.

If `halfstep` is stopped with the toleance $\texttt{tol} = 10^{-\texttt{dec}/2}$ the final approximation has just about reached the accuracy of `dec` digits. If the convergence rate is slower, as it appears to be the case in I_{25}, the required accuracy may not quite be reached. The choice of a smaller value of `tol` may help, at the risk of unnecessarily doubling the computational effort in other cases.

The program given below is the actual call for approximating the 25 test integrals of `fct` in 67-digit working precision.

```
\\ Calling program, choose dec
dec=67; default(realprecision,dec); tol=1/10^floor(1/2*dec);
inf=2^30; maxexp=10000; l10=log(10); Data();
for(ind=1,nb, b=bvec[ind]; L=left[ind]; R=right[ind]; ex=exact[ind];\
    halfstep(tol); print(ind,"     ",evec,"     ",cvec); print(val));
```

6 Results

Here we summarize the results of the calling program given at the end of the previous sectiom. With a working precision of `dec` = 67 digits the program executes in 4.1 seconds on a 1.6-MHz processor. In this range the execution time roughly scales as $\texttt{dec}^{2.3}$. An accuracy of 65 or 66 digits is achieved in all cases except for I_{25}, where an additional step bisection would be required in order to compensate for the slower convergence rate with respect to step bisection.

Case	Correct digits per stage	Number of function evaluations
1	[2,6,14,31,66]	[15,29,53, 97,183]
2	[2,6,13,29,62,65]	[15,29,53, 97,183,349]
3	[0,1, 4,13,37,66]	[11,21,37, 66,121,227]
4	[0,0, 1,10,33,66]	[13,25,46, 84,155,294]
5	[2,6,14,31,67]	[15,29,53, 97,183]
6	[2,5,12,24,50,65]	[15,29,53, 97,183,349]
7	[2,5,11,20,40,65]	[15,29,53, 97,183,349]
8	[1,0, 2, 4, 8,17,35,65]	[15,29,53, 97,181,347,674,1323]
9	[4,7,15,32,66]	[15,31,59,109,205]
10	[0,5,12,26,58,65]	[15,30,56,103,194,371]
11	[2,6,12,25,50,65]	[16,31,59,110,209,402]
12	[4,7,15,32,66]	[17,33,63,119,226]
13	[3,6,14,31,65]	[13,27,50, 93,173]
14	[2,5,11,26,56,65]	[15,30,55,101,190,363]
15	[1,3, 7,15,30,59,65]	[15,30,56,103,194,371,721]
16	[2,4,10,22,43,65]	[13,26,47, 86,160,304]
17	[3,5, 9,14,23,38,65]	[14,29,54,100,187,357,694]
18	[3,7,16,33,66]	[17,34,65,122,233]
19	[4,8,18,35,65]	[20,39,73,138,263]
20	[3,6,12,23,45,65]	[15,29,53, 97,182,348]
21	[1,2, 5, 8,15,28,51,65]	[14,29,54,100,187,358,695,1373]
22	[0,0, 2, 4, 8,18,36,65]	[14,27,50, 92,171,325,629,1242]
23	[3,7,15,32,65]	[15,29,55,101,191]
24	[3,7,16,32,66]	[15,31,59,109,205]
25	[1,4, 7,12,20,35,58]	[15,31,59,109,205,393,765]

References

[1] M. Abramowitz, I. Stegun: *Handbook of Mathematical Functions*. Nat. Bureau of Standards, 1964, 1046 pp.

[2] Z. Ahmed: *Definitely an integral*. Amer. Math. Monthly **109**, 2002, 670-671.

[3] David H. Bailey, D. Broadhurst: *Parallel integer relation detection*. To appear in Math. of Comp., 2006.

[4] C. Batut, K. Belabas, D. Bernardi, H. Cohen, M. Olivier: *The software package PARI* (freeware). http://pari.math.u-bordeaux.fr/

[5] Folkmar Bornemann, Dirk Laurie, Stan Wagon, Jörg Waldvogel: *The SIAM 100-Digit Challenge*. SIAM, Philadelphia 2004, 306 pp.

[6] P. J. Davis, P. Rabinowitz: *Methods of Numerical Integration*. Academic Press, San Diego, Second Edition, 1984, 612 pp.

[7] W. Gautschi, J. Waldvogel: *Computing the Hilbert transform of the generalized Laguerre and Hermite weight functions*. BIT Numerical Mathematics **41**, 2001, 490-503.

[8] H.R. Schwarz: *Numerical Analysis*. John Wiley 1989, 517 pp. In particular: *Numerical Quadrature*, 330-350, by J. Waldvogel. Original edition: *Numerische Mathematik*, Teubner 1986, 496 pp.

[9] H. Takahasi, M. Mori: *Double exponential formulas for numerical integration*. Publ. RIMS, Kyoto Univ. **9**, 1974, 721-741.

[10] Jörg Waldvogel: *Fast construction of the Fejér and Clenshaw-Curtis quadrature rules*. BIT Numerical Mathematics **46**, 2006, 195-202.