Computing infinite range integrals of an arbitrary product of Bessel functions

Joris Van Deun and Ronald Cools

Report TW416, February 2005

Katholieke Universiteit Leuven
Department of Computer Science
Celestijnenlaan 200A – B-3001 Heverlee (Belgium)

Computing infinite range integrals of an arbitrary product of Bessel functions

Joris Van Deun and Ronald Cools

Report TW416, February 2005

Department of Computer Science, K.U.Leuven

Abstract

We present an algorithm to compute integrals of the form

$$\int_0^\infty x^m \prod_{i=1}^{k} J_{\nu_i}(a_i x) dx$$

with $J_{\nu_i}(x)$ the Bessel function of the first kind and order ν_i. The parameter m is a real number such that $\sum_i \nu_i + m > -1$ and the coefficients a_i are strictly positive real numbers. The main ingredients in this algorithm are the well-known asymptotic expansion for $J_{\nu_i}(x)$ and the observation that the infinite part of the integral can be approximated using the incomplete Gamma function $\Gamma(a, z)$. Accurate error estimates are included in the algorithm, which is implemented as a Matlab program.

Keywords : Quadrature, Bessel functions
CR Subject Classification : Primary : G.1.4 (Numerical Analysis; Quadrature and Numerical Differentiation) Secondary : G.4 (Mathematical Software)

Computing infinite range integrals of an arbitrary product of Bessel functions

Joris Van Deun and Ronald Cools
Department of Computer Science, Katholieke Universiteit Leuven
Celestijnenlaan 200A, B-3001 Heverlee, Belgium

February 21, 2005

Abstract

We present an algorithm to compute integrals of the form

$$\int_0^\infty x^m \prod_{i=1}^{k} J_{\nu_i}(a_i x) dx$$

with $J_{\nu_i}(x)$ the Bessel function of the first kind and order ν_i. The parameter m is a real number such that $\sum_i \nu_i + m > -1$ and the coefficients a_i are strictly positive real numbers. The main ingredients in this algorithm are the well-known asymptotic expansion for $J_{\nu_i}(x)$ and the observation that the infinite part of the integral can be approximated using the incomplete Gamma function $\Gamma(a, z)$. Accurate error estimates are included in the algorithm, which is implemented as a Matlab program.

1 Introduction

The main motivation for writing this paper, arises from Trefethen's famous 100-Digit Challenge [Tre02]. As a result of this challenge, some of the winning team members joined up to write a book [BLWW04], presenting several solutions to each of the ten original problems, and offering some new problems in Appendix D. Problem 8 of these additional problems, proposed by Folkmar Bornemann, reads as follows,

> **8.** What is the value of
>
> $$\int_0^\infty x J_0(x\sqrt{2}) J_0(x\sqrt{3}) J_0(x\sqrt{5}) J_0(x\sqrt{7}) J_0(x\sqrt{11}) dx,$$
>
> where J_0 denotes the Bessel function of the first kind of order zero?

In fact, the book's webpage contains some references to the literature, where explicit expressions can be found for some of these integrals with three or four factors [Nic20, Son80]. Reference [Wat66] also contains examples of integrals with an arbitrary number of factors.

In this paper we present a numerical method to approximate integrals of the form

$$I(\boldsymbol{a},\boldsymbol{\nu},m) = \int_0^\infty x^m \prod_{i=1}^k J_{\nu_i}(a_i x)dx$$

where the coefficients $a_i \in \mathbf{R}_0^+$ (we use $\boldsymbol{a}$ to denote the vector which contains these coefficients), the orders $\nu_i \in \mathbf{R}$, the power $m \in \mathbf{R}$ and $\sum_{i=1}^k \nu_i + m > -1$ (the last condition assures that a possible singularity in zero is integrable). If the integral exists in the ordinary sense, our algorithm can yield a result which is correct up to the last but one digit. In the case where the integral is only defined in the sense of Abel summability [Sid87], experiments indicate that our method is still valid (although some significant digits may be lost due to cancellation). We will, however, not prove this.

Formula 9.1.35 in [AS64] states that

$$J_\nu(-z) = e^{\nu\pi\mathbf{i}} J_\nu(z),$$

so the case of negative coefficients a_i can always be reduced to the case of positive coefficients, multiplying by a constact factor. We leave it to the reader to do this transformation, so that we are certain that the result is a real number.

The approach we take is as follows. We split the integral $I(\boldsymbol{a},\boldsymbol{\nu},m)$ at the point x_0 to obtain

$$I(\boldsymbol{a},\boldsymbol{\nu},m) = I_1(\boldsymbol{a},\boldsymbol{\nu},m,x_0) + I_2(\boldsymbol{a},\boldsymbol{\nu},m,x_0)$$

where

$$I_1(\boldsymbol{a},\boldsymbol{\nu},m,x_0) = \int_0^{x_0} x^m \prod_{i=1}^k J_{\nu_i}(a_i x)dx$$

and likewise for the infinite part $I_2(\boldsymbol{a},\boldsymbol{\nu},m,x_0)$. The finite part I_1 can be approximated using an appropriate quadrature formula. More details are given in section 4. In the next section we explain how to compute the infinite part.

2 Infinite range approximation

It is well-known, see e.g. [AS64, p. 364] or [Wat66, p. 199], that there exist functions $P(\nu,z)$ and $Q(\nu,z)$ such that

$$J_\nu(x) = \sqrt{\frac{2}{\pi x}}\left[P(\nu,x)\cos\chi - Q(\nu,x)\sin\chi\right] \tag{1}$$

where $\chi = x - (\nu/2 + 1/4)\pi$. For large x the functions $P(\nu,x)$ and $Q(\nu,x)$ admit the following asymptotic expansions,

$$P(\nu,x) \sim \sum_{j=0}^\infty (-1)^j \frac{(\mu-1^2)(\mu-3^2)\dots(\mu-(4j-1)^2)}{(2j)!(8x)^{2j}} = \sum_{j=0}^\infty c_{\nu,j} x^{-2j},$$

$$Q(\nu,x) \sim \sum_{j=0}^\infty (-1)^{j+1} \frac{(\mu-1^2)(\mu-3^2)\dots(\mu-(4j+1)^2)}{(2j+1)!(8x)^{2j+1}} = \sum_{j=0}^\infty d_{\nu,j} x^{-2j-1}.$$

where $\mu = 4\nu^2$. We introduce the notation for the partial sums $P_n(\nu, x) = \sum_{j=0}^{n} c_{\nu,j} x^{-2j}$ and $Q_n(\nu, x) = \sum_{j=0}^{n} d_{\nu,j} x^{-2j-1}$ and put

$$F(\nu, x) = \frac{1-\mathbf{i}}{2\sqrt{\pi x}} \exp\left(-\frac{\pi}{2}\nu\mathbf{i}\right) \left[P(\nu, x) + \mathbf{i}Q(\nu, x)\right],$$
$$F_n(\nu, x) = \frac{1-\mathbf{i}}{2\sqrt{\pi x}} \exp\left(-\frac{\pi}{2}\nu\mathbf{i}\right) \left[P_n(\nu, x) + \mathbf{i}Q_n(\nu, x)\right],$$

where $\mathbf{i}$ is the imaginary unit, $\mathbf{i}^2 = -1$. Then using equation (1) we may write

$$J_\nu(x) = e^{\mathbf{i}x} F(\nu, x) + e^{-\mathbf{i}x} \overline{F(\nu, x)} \sim e^{\mathbf{i}x} F_n(\nu, x) + e^{-\mathbf{i}x} \overline{F_n(\nu, x)} = J_{\nu,n}(x). \quad (2)$$

Note that the part between square brackets in the definition of F_n is a Laurent polynomial of degree $2n + 1$. With the aid of formula (2) the integrand in $I_2(\boldsymbol{a}, \boldsymbol{\nu}, m, x_0)$ can be approximated by

$$x^m \prod_{i=1}^{k} J_{\nu_i}(a_i x) \sim$$
$$2\Re\left\{x^m \sum e^{\mathbf{i}(a_1 \pm a_2 \pm \ldots \pm a_k)x} F_n(\nu_1, a_1 x) \overset{(}{\overline{F_n(\nu_2, a_2 x)}}{}^{)} \ldots \overset{(}{\overline{F_n(\nu_k, a_k x)}}{}^{)}\right\} \quad (3)$$

where the sum is over all 2^{k-1} possible combinations of $a_1 \pm a_2 \pm \ldots \pm a_k$ and a complex conjugate bar only occurs on those factors $F_n(\nu_j, a_j x)$ whose corresponding coefficient a_j appears with a minus sign in the exponent. In each term of this sum, the product following the exponential consists of a Laurent polynomial of degree $k(2n + 1)$ times $x^{-k/2}$. These Laurent polynomials are completely defined by the coefficients $c_{\nu,j}$ and $d_{\nu,j}$ in the expressions for $P_n(\nu, x)$ and $Q_n(\nu, x)$ and by the coefficients $\boldsymbol{a}$. We thus have to compute $2^{k-1}(k(2n + 1) + 1)$ integrals of the form

$$\int_{x_0}^{\infty} e^{\mathbf{i}\eta_i x} x^{m-k/2-j} dx, \quad i = 1, 2, \ldots, 2^{k-1}, \quad j = 0, 1, \ldots, k(2n + 1).$$

This can easily be done, because

$$\int_{x_0}^{\infty} e^{\mathbf{i}\alpha x} x^{\beta} dx = \left(\frac{\mathbf{i}}{\alpha}\right)^{\beta+1} \Gamma(\beta + 1, -\mathbf{i}\alpha x_0), \quad (4)$$

whenever $\alpha \neq 0$, where $\Gamma(\cdot, \cdot)$ denotes the incomplete Gamma function [AS64, p. 260]. This function can be evaluated very efficiently using Legendre's well-known continued fraction expansion, as explained in [KG87, Win03]. When $\alpha = 0$ and $\beta < -1$ we obviously have $\int_{x_0}^{\infty} x^{\beta} dx = -x_0^{\beta+1}/(\beta + 1)$. This case occurs when $m < k/2 - 1$. In the case where $\alpha = 0$ and $m \geq k/2 - 1$, the result $I(\boldsymbol{a}, \boldsymbol{\nu}, m)$ may still be finite, if the coefficients in (3) corresponding to the infinite integrals are zero. However, in this case $I(\boldsymbol{a}, \boldsymbol{\nu}, m)$ can be a discontinuous function of its parameters, which is why we refer to this case as the *discontinuous* case. A detailed analysis of this problem is outside the scope of this article, but from our point of view, the discontinuity essentially arises from the fact that the function $g(\alpha) = \int_{x_0}^{\infty} \sin(\alpha x) x^{\beta} dx$ is discontinuous in $\alpha = 0$ for $\beta \geq -1$. Next we look at *a priori* error estimates for the infinite range approximation.

3 Error analysis for infinite part

It is known [AS64, p. 364] that the remainder after n terms in the expansion of $P(\nu, x)$ does not exceed the $(n+1)$th term in absolute value whenever $n \geq |\nu|/2$. The same is true for $Q(\nu, x)$. We assume that this condition is satisfied. This means that we have the bounds

$$|P(\nu, x) - P_n(\nu, x)| \leq \frac{|c_{\nu,n+1}|}{x^{2n+2}}, \tag{5}$$

$$|Q(\nu, x) - Q_n(\nu, x)| \leq \frac{|d_{\nu,n+1}|}{x^{2n+3}}, \tag{6}$$

which will be the key to the error estimates in this section. To simplify the formulas, let us first introduce some notation. Put

$$f(x) = x^m \prod_{i=1}^{k} J_{\nu_i}(a_i x), \quad f_n(x) = x^m \prod_{i=1}^{k} J_{\nu_i,n}(a_i x)$$

for the integrand and its infinite range approximation. We are interested in obtaining upper bounds for the absolute and relative errors

$$\Delta(x_0, n) = \left| I_2(\boldsymbol{a}, \boldsymbol{\nu}, m, x_0) - \int_{x_0}^{\infty} f_n(x) dx \right|,$$

$$\delta(x_0, n) = \frac{\Delta(x_0, n)}{|I_2(\boldsymbol{a}, \boldsymbol{\nu}, m, x_0)|}.$$

It is clear that

$$J_{\nu,n}(x) = J_\nu(x) - \left[e^{\mathbf{i}x}(F(\nu, x) - F_n(\nu, x)) + e^{-\mathbf{i}x}\overline{(F(\nu, x) - F_n(\nu, x))} \right],$$

so, discarding second order error terms, this yields

$$\prod_{i=1}^{k} J_{\nu_i,n}(a_i x) \approx \prod_{i=1}^{k} J_{\nu_i}(a_i x) - \sum_{j=1}^{k} \prod_{\substack{i=1 \\ i \neq j}}^{k} J_{\nu_i}(a_i x) \left[e^{\mathbf{i}a_j x}(F(\nu_j, a_j x) - F_n(\nu_j, a_j x)) + e^{-\mathbf{i}a_j x}\overline{(F(\nu_j, a_j x) - F_n(\nu_j, a_j x))} \right].$$

Using this equation we can write

$$f(x) - f_n(x) \approx 2\Re \left\{ x^m \sum_{j=1}^{k} \prod_{\substack{i=1 \\ i \neq j}}^{k} J_{\nu_i}(a_i x) e^{\mathbf{i}a_j x}(F(\nu_j, a_j x) - F_n(\nu_j, a_j x)) \right\}.$$

With the definition of $F(\nu, x)$ and $F_n(\nu, x)$ and the fact that for large x

$$|J_\nu(x)| \leq \sqrt{\frac{2}{\pi x}}, \quad x \to \infty,$$

some computations then lead to

$$|f(x) - f_n(x)| \leq x^{m-k/2} \left(\frac{2}{\pi}\right)^{k/2} (a_1 \dots a_k)^{-1/2}$$
$$\sum_{j=1}^{k} (|P(\nu_j, a_j x) - P_n(\nu_j, a_j x)| + |Q(\nu_j, a_j x) - Q_n(\nu_j, a_j x)|).$$

Now use equations (5)–(6) and define

$$A_n = \sum_{j=1}^{k} \left| c_{\nu_j, n} \right| a_j^{-2n}, \quad B_n = \sum_{j=1}^{k} \left| d_{\nu_j, n} \right| a_j^{-2n-1},$$

to obtain

$$|f(x) - f_n(x)| \leq \left(\frac{2}{\pi}\right)^{k/2} (a_1 \dots a_k)^{-1/2} x^{m-k/2-2n-3} (A_{n+1} x + B_{n+1}).$$

Integrating between x_0 and ∞ we finally get the following estimate for the absolute error bound, assuming that $n > (m - k/2 - 2)/2$,

$$\Delta(x_0, n) \lesssim \left(\frac{2}{\pi}\right)^{k/2} (a_1 \dots a_k)^{-1/2} x_0^{m-k/2-2n-2}$$
$$\left(\frac{A_{n+1} x_0}{2n + 1 + k/2 - m} + \frac{B_{n+1}}{2n + 2 + k/2 - m}\right). \quad (7)$$

We use the symbol $\lesssim$ to indicate that, strictly speaking, this is not a real upper bound, but only an *estimate* for an upper bound, obtained by taking into account only first order error terms, as explained before. The numerical experiments indicate, however, that this estimate is completely satisfactory (except in the case of very large orders or the discontinuous case: in those cases the estimate can be too optimistic).

To find an upper bound for the relative error $\delta(x_0, n)$, we need an estimate for the integral $I_2(\boldsymbol{a}, \boldsymbol{\nu}, m, x_0)$. Since we are only interested in orders of magnitude, a very rough estimate will be sufficient. Numerical evidence indicates that

$$|I_2(\boldsymbol{a}, \boldsymbol{\nu}, m, x_0)| \approx \left(\frac{2}{\pi}\right)^{k/2} (a_1 \dots a_k)^{-1/2} \eta^{k/2-m-1} \left|\Gamma(m - k/2 + 1, -\mathbf{i} x_0 \eta)\right|, \quad (8)$$

where $\eta = \sum_{j=1}^{k} a_j$, is a good estimate. This formula is obtained as follows. In (3), approximate the sum by taking 2^{k-1} times the term corresponding to all positive coefficients (i.e. no complex conjugates appear), and only take the first term in the Laurent polynomial following the exponential. This yields

$$f(x) \approx 2^k \Re \left\{ x^{m-k/2} e^{\mathbf{i}\eta x} \left(\frac{1 - \mathbf{i}}{2\sqrt{\pi}}\right)^k (a_1 \dots a_k)^{-1/2} \right\}.$$

Integrating between x_0 and ∞ using (4) and taking the modulus instead of the real part (this yields a simpler formula) then gives (8). Taking only the first term in the asymptotic expansion of the incomplete Gamma function as given by formula 6.5.32 in [AS64], we can simplify this further to

$$|I_2(\boldsymbol{a},\boldsymbol{\nu},m,x_0)| \approx \left(\frac{2}{\pi}\right)^{k/2} (a_1 \dots a_k)^{-1/2} \frac{x_0^{m-k/2}}{\eta}. \tag{9}$$

This is the formula used in the program, since it avoids the computation of the incomplete Gamma function and gives satisfactory results.

4 Finite range integration

To compute the integral $I_1(\boldsymbol{a},\boldsymbol{\nu},m,x_0)$, the idea was to provide a tailor-made numerical quadrature formula. First the interval $[0,x_0]$ is split into several subintervals at equidistant points. The number of intervals is proportional to the number of zeros in the integrand, an estimate obtained from the approximation

$$J_\nu(x) \approx \sqrt{\frac{2}{\pi x}} \cos\left(x - \left(\frac{\nu}{2} + \frac{1}{4}\right)\pi\right).$$

In this way we more or less take into account the number of oscillations in the integrand. However, our experiments indicate that we do not gain any accuracy by splitting at the actual zeros. The subfunction `fri` takes care of this part, and calls the actual quadrature formula, implemented in the function `nqf`. This function contains several hard coded Gauss-Legendre rules, which are capable of integrating almost up to machine precision the integrand on each of the subintervals, for most cases we tested. This special purpose quadrature formula is faster and provides more accurate (i.e. less pessimistic) error estimates than e.g. Matlab's `quad` or Cubpack's `nint` [NC05]. In the exceptional cases where the requested precision is not reached, a warning message is given and the best result is returned. At this point, we could switch to one of the automatic general purpose routines. This code is commented out and can easily be incorporated.

Special care must be taken in the first subinterval $[0,x_1]$ because of a possible algebraic singularity in 0. From formula 9.1.7 in [AS64] it follows that the integrand $f(x)$ satisfies

$$f(x) = x^p \sum_{i=0}^{\infty} \alpha_i x^i, \quad x \to 0,$$

for some coefficients α_i (the exact values are not relevant here) and where $p = \sum_{i=1}^{k} \nu_i + m$. If p is not a positive integer, there is an algebraic singularity in 0 which is difficult to integrate with the hard-coded Gauss-Legendre rules. However, since we know the exact degree of the singularity, we can write a specific extrapolation procedure to solve this problem. Defining

$$I_0 = \int_0^{x_1} f(t)dt, \quad I_x = \int_x^{x_1} f(t)dt$$

it follows that

$$I_x = I_0 - x^p \sum_{i=0}^{\infty} \frac{\alpha_i}{i+p+1} x^i.$$

If we know the values I_{ρ_j} where $\rho_j = \rho^j x_1/2$ for $j = 0, 1, \ldots$ and $\rho < 1$ (e.g. $\rho = 1/4$), then we can extrapolate (by successively eliminating terms in the series expansion) to the value I_0 using an extrapolation scheme which is completely analogous to Richardson extrapolation [Hen77, p. 461–463] (we omit the details). Of course, since $I_{\rho_j} = \int_{\rho_j}^{\rho_{j-1}} f(t)dt + I_{\rho_{j-1}}$, we only have to integrate between successive values of ρ_j, which can easily be done using the subfunction `nqf`. The extrapolation procedure is part of the subfunction `fri`.

5 Implementation issues

The Matlab implementation of this algorithm is given by two functions. The function `igamma.m` is an implementation of the incomplete Gamma function $\Gamma(a, z)$ for arbitrary complex a and z. We cannot use Matlab's `gammainc`, since this only supports real arguments. For the same reason, we cannot use any of the other implementations available from Transactions on Mathematical Software. Our implementation is a straightforward Fortran-to-Matlab conversion of the program from [KG87], which computes the continued fraction approximation using a forward recurrence. In [Win03] a way is described to determine *a priori* the number of terms needed in the continued fraction to reach a certain accuracy, such that one can use the (generally faster) backward recurrence. However, this estimate is only valid asymptotically (for a large number of terms) and our experiments (comparing the result to a multiprecision computation in Maple) indicate that it is not reliable in our case. Therefore, we have not taken the idea of using backward recurrence any further. Of course, if a faster implementation of the incomplete Gamma function is available to the user, it can be used instead of `igamma`.

The actual algorithm is implemented in `besselint.m` and consists of several subfunctions. The Matlab script does not require any special toolbox, so it can easily serve as a template for translations in another language.

Before we can compute the finite and infinite part of the integral, an important issue to be resolved is the automatic determination of the parameters x_0 (the breakpoint) and n (the infinite range approximation order). The contour plot of the function $\delta(x_0, n)$ in figure 1 shows that for each precision there are infinitely many possible (x_0, n)-pairs. We choose these parameters such that they minimize the computational effort. The cost of our algorithm is dominated by the cost of evaluating the Bessel and incomplete Gamma functions. A suitable cost function takes into account the number of evaluations of the incomplete Gamma function compared to the number of evaluations of the Bessel function. This leads to

$$\chi(x_0, n) = 2^k n t_{GJ} + \frac{x_0}{\pi} \frac{N}{2} \sum_{j=1}^{k} a_j \tag{10}$$

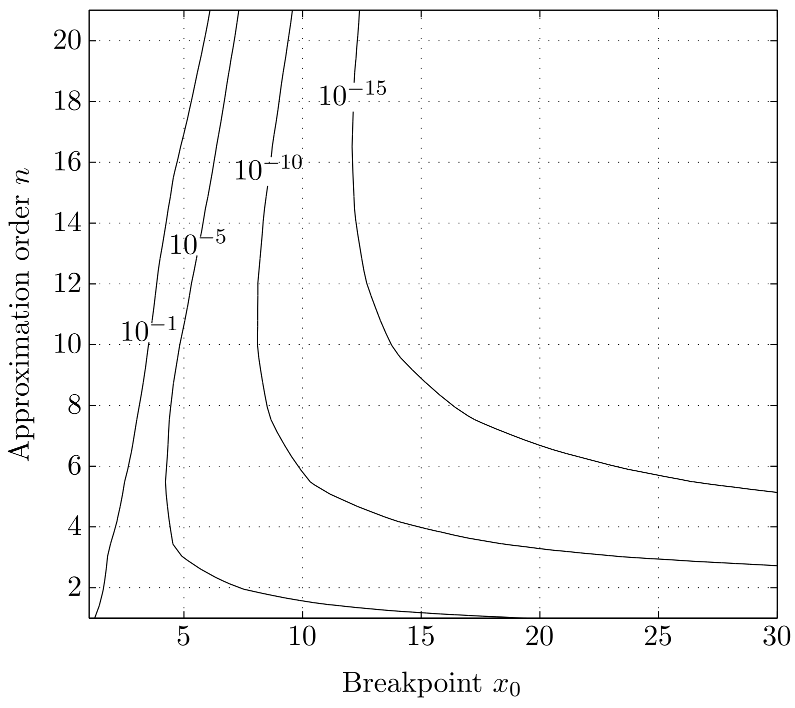

Figure 1: Relative accuracy of infinite range approximation. Parameters are $\boldsymbol{a} = [\sqrt{2}, \sqrt{3}, \sqrt{5}]$, $\boldsymbol{\nu} = [0, 0, 0]$ and $m = 1$.

where t_{GJ} is the relative efficiency of the incomplete Gamma function compared to the Bessel function. We provide a small testprogram `gettGJ` to determine this value automatically. The parameter N is the average number of function evaluations in the numerical quadrature formula *per subinterval*. With our special purpose routine and for full accuracy, this will usually be $N = 15 + 19$ (the 19-point rule is necessary to get an accurate error estimate). This formula was obtained as follows. The number of evaluations of the incomplete Gamma function is $2^{k-1}(k(2n+1)+1)$, but ignoring the fixed cost (independent of n) gives $k2^k n$. For the Bessel function we have kN evaluations per interval, times the number of subintervals. The approximate zeros of the integrand are given by $\pi(i - 1/4 + \nu_j/2)/a_j$ for $j = 1, \ldots, k$ and $i = 1, 2, \ldots$, so the number of subintervals equals $\sum_{j=1}^{k}(x_0 a_j/\pi + 1/4 - \nu_j/2)$ (we ignore possible extrapolation). In our implementation we apply the Gauss-Legendre rules on two consecutive intervals at once. This turns out to give the same accuracy, for half the amount of work. Therefore the number of intervals is divided by 2. Ignoring the fixed cost, adding both terms together and dividing them by k (this does not change the optimum) gives (10). Minimizing this cost function with the constraint that the relative or absolute error must equal the required accuracy, gives the values of x_0 and n which we need. This optimization problem is solved in the subfunction `gop`. In the appendix we explain how this is done.

Once the parameters x_0 and n are determined, the infinite range approximation is computed in the subfunction `ira`. An efficient way to sum over all 2^{k-1} terms, is taking the binary representation of the loop variable to determine the plus and minus signs and the complex conjugate bars in formula (3). To test if a certain combination of the a_i's equals zero,

$$a_1 \pm a_2 \pm \ldots \pm a_k = \sum_{i=1}^{k} \gamma_i a_i = 0,$$

we actually test if $|\sum \gamma_i a_i| \leq \epsilon \sum a_i$, where ϵ is the machine precision. If this condition is satisfied *and* if $m \geq k/2 - 1$, a warning is given stating the assumption $\sum \gamma_i a_i = 0$. Because of the possible discontinuity, the result may be very different depending on whether or not this assumption is satisfied. In fact, examples are known where an infinitesimal change in the coefficients a_i makes the difference between 0 and infinity in the final result (we refer to the next section). In the case $\sum \gamma_i a_i = 0$ (discontinuous or not), we can also improve numerical accuracy, depending on the value of $\sum \gamma_i(\nu_i + 1/2)$. From the formulas for F_n, P_n and Q_n it follows that the asymptotic expansion of the integrand will contain an expression of the form

$$Cx^{m-k/2} \exp\left[-\frac{\pi}{2}\mathbf{i}\sum_{i=1}^{k}\gamma_i\left(\nu_i + \frac{1}{2}\right)\right]\prod_{i=1}^{k}[P_n(\nu_i, a_i x) + \gamma_i \mathbf{i} Q_n(\nu_i, a_i x)]$$

(C is a real constant). The coefficients with an odd index in the product following the exponential are real, the ones with an even index are purely imaginary. Since we are only interested in the real part of the integral, we can ignore all the odd or even coefficients, depending on whether the exponential is real or purely imaginary. This is also exploited in the program.

Finally we mention that the user can also specify a relative and/or absolute error tolerance, and if requested, the program returns error estimates.

6 Experiments

To test our algorithm and its implementation, we compare with some explicitly known formulas. Most of these formulas can be found in [Wat66, Chap. 13] and [Son80]. We present them in the order of an increasing number of factors in the integrand. First of all we have formula (ω_4) from [Son80, p. 39] which states that

$$\int_0^\infty x^m J_\nu(x)dx = 2^m \frac{\Gamma\left(\frac{\nu+m+1}{2}\right)}{\Gamma\left(\frac{\nu-m+1}{2}\right)}, \quad m < 1/2, \quad \nu + m > -1,$$

leading to the very simple case

$$\int_0^\infty J_\nu(x)dx = 1, \quad \nu > -1.$$

Our experiments indicate that this formula is also valid in the sense of Abel summability when $m \geq 1/2$, leading to another simple example

$$\int_0^\infty x J_\nu(x) dx = \nu, \quad \nu > -2.$$

We have not encountered any proof of this formula. Some more examples with diverging integrals, defined only in the sense of Abel summability, can be found in [Sid88]. The following two examples are given,

$$\int_0^\infty x^2 J_0(x) dx = -1, \quad \int_0^\infty x^4 J_0(x) dx = 9.$$

It turns out that our algorithm also provides the correct answer for this type of integrals. Some digits are lost, however, in the first step (finite range integration using a numerical quadrature formula) due to cancellation.

An example of an integrand containing two factors is the Weber-Schafheitlin integral (ω_5) in [Son80, p. 39]. For $\nu > \mu > -1$,

$$\int_0^\infty x^{\mu-\nu+1} J_\mu(bx) J_\nu(ax) dx = \begin{cases} \dfrac{b^\mu}{a^\nu \Gamma(\nu-\mu)} \left(\dfrac{a^2-b^2}{2}\right)^{\nu-\mu-1}, & a > b, \\ 0, & a < b, \end{cases}$$

which again leads to a simple special case

$$\int_0^\infty J_0(bx) J_1(ax) dx = \begin{cases} \dfrac{1}{a}, & a > b, \\ 0, & a < b. \end{cases}$$

Also in this case our experiments indicate that this formula remains valid in the sense of Abel summability, yielding

$$\int_0^\infty x J_\nu(ax) J_\nu(bx) dx = 0, \quad a \neq b.$$

The critical case of the Weber-Schafheitlin integral occurs when $a = b$. An example is

$$\int_0^\infty J_\mu(ax) J_{\mu-1}(ax) dx = \frac{1}{2a}, \quad \mu > 0,$$

which is formula (8) in [Wat66, p. 406]. Comparing this formula to the formulas for $a \neq b$ clearly shows that the Weber-Schafheitlin integral is discontinuous in its arguments.

Next we turn to the case of three factors. Formula (ω_{11}) in [Son80, p. 46] gives

$$\int_0^\infty x^{1-\nu} J_\nu(ax) J_\nu(bx) J_\nu(cx) dx = \frac{2^{\nu-1} \Delta^{2\nu-1}}{(abc)^\nu \Gamma(\nu+1/2) \Gamma(1/2)}, \quad \nu > -1/2,$$

where Δ is the area of the triangle with sides a, b and c, i.e.

$$\Delta = \frac{1}{4}\sqrt{(a+b+c)(a+b-c)(a+c-b)(b+c-a)},$$

whenever these lengths form a triangle. Otherwise, the integral equals zero. The formula is also valid in the case of a *degenerate* triangle, i.e. when $\Delta = 0$. For $\nu = 0$, this yields one of the discontinuous cases we mentioned. An example where the Bessel functions are of different order can be found in [JM72, p. 455],

$$\int_0^\infty x J_\mu(ax) J_\nu(bx) J_\sigma(cx) dx = \frac{1}{6\pi\Delta}[\cos(\mu\beta - \nu\alpha) + \cos(\nu\gamma - \sigma\beta) + \cos(\sigma\alpha - \mu\gamma)]$$

where Δ is as before, $\mu + \nu + \sigma = 0$ and $\mu, \nu, \sigma \in \mathbf{N}$. The values α, β and γ are the exterior angles of the triangle formed by the lengths a, b and c (i.e. α is the exterior angle opposite the side with length a, etc.). When the sides do not form a triangle, the integral equals 0. When they form a degenerate triangle, the integral is infinite.

In [Wat66, Chap. 13] we find some formulas for integrals involving four Bessel functions. The first one is (9) on page 414 and can also be found in [Nic20] and an example is cited on the web page of [BLWW04],

$$\int_0^\infty x \prod_{i=1}^4 J_0(a_i x) dx = \begin{cases} \dfrac{1}{\pi^2 \Box} K\left(\dfrac{\sqrt{a_1 a_2 a_3 a_4}}{\Box}\right) \\ \dfrac{1}{\pi^2 \sqrt{a_1 a_2 a_3 a_4}} K\left(\dfrac{\Box}{\sqrt{a_1 a_2 a_3 a_4}}\right) \end{cases}$$

where $K(k)$ denotes the complete elliptic integral of the first kind of modulus k and that one whose modulus is less than one is to be taken. The symbol $\Box$ denotes the area of the quadrilateral whose sides are a_1, a_2, a_3 and a_4 (whenever they form a quadrilateral),

$$6\Box^2 = \prod_{i=1}^4 (a_1 + a_2 + a_3 + a_4 - 2a_i).$$

Another formula involving four factors is (10) in [Wat66, p. 415],

$$\int_0^\infty x^{1-2\nu} [J_\nu(ax)]^4 dx = \frac{a^{2\nu-2}\Gamma(2\nu)\Gamma(\nu)}{2\pi\Gamma(3\nu)[\Gamma(\nu+1/2)]^2}, \quad \nu > 0.$$

The last two formulas we present have an arbitrary number of Bessel functions. The first one is (8) from [Wat66, p. 413],

$$\int_0^\infty x^{1-\nu k+2\nu} \prod_{i=1}^k J_\nu(a_i x) dx = 0$$

if $a_1 > \sum_{i=2}^k a_i$ (assuming they are arranged in descending order of magnitude) and $\nu > -1$. The second formula for integrals containing an arbitrary number of Bessel functions is (16) on page 419 in the same reference, which states that

$$\int_0^\infty x^{\mu-k\nu-1} J_\mu(bx) \prod_{i=1}^k J_\nu(a_i x) dx = \frac{2^{\mu-1}\Gamma(\mu)}{b^\mu} \prod_{i=1}^k \left(\frac{(a_i/2)^\nu}{\Gamma(\nu+1)}\right)$$

if $b > \sum_{i=1}^{k} a_i$ and $k\nu + (k+1)/2 > \mu > 0$.

All these examples and some additional tests are gathered in the Matlab script `experiments.m`, together with ample comments and a comparison of the requested, predicted and actual accuracy.

Acknowledgements

We would like to thank Dirk Laurie for bringing this topic to our attention during his stay at the Department of Computer Science of the K.U.Leuven in autumn 2004. The last problem in his course on high accuracy numerical computation was almost identical to problem 8 we mentioned in the introduction, but with x^2 instead of x. Both problems are included in the test program. We are also very grateful that he was willing to extensively test several preliminary versions of the program.

We also would like to thank Folkmar Bornemann for pointing out to us references [JM72] and [Wat66].

A Computing the optimal parameters

First we look at the case where the absolute error should not exceed a required accuracy ϵ. The optimization problem at hand is

> Find the values of x_0 and n which minimize the cost function $\chi(x_0, n)$ with the constraint that $\Delta(x_0, n) \leq \epsilon$.

Using the method of Lagrange multipliers, this leads to the system of equations

$$\begin{aligned} \frac{\partial \chi}{\partial x_0} - \lambda \frac{\partial \Delta}{\partial x_0} &= 0 \\ \frac{\partial \chi}{\partial n} - \lambda \frac{\partial \Delta}{\partial n} &= 0 \\ \lambda z &= 0 \\ \Delta(x_0, n) - \epsilon + z^2 &= 0 \end{aligned}$$

where λ is a Langrage multiplier and z an auxiliary variable. With the definition of χ this can further be reduced to

$$\frac{\partial \Delta}{\partial x_0} 2^k t_{GJ} = \frac{\partial \Delta}{\partial n} \frac{N}{2\pi} \sum_{j=1}^{k} a_j, \tag{11}$$

$$\Delta(x_0, n) = \epsilon. \tag{12}$$

The last equation indicates that the constraint is active (which is not surprising since the target function is a plane). In the rest of this section we explain how the solution to this system of two nonlinear equations can be approximated by solving one nonlinear equation in one variable. We approximate equation (11) to obtain a simple relation between x_0 and n, which we can then substitute into equation (12).

Of course, we take the upper bound in (7) as an estimate for the actual error $\Delta(x_0, n)$. Differentiating this estimate with respect to x_0 gives

$$\frac{\partial \Delta}{\partial x_0}(x_0, n) \approx -\left(\frac{2}{\pi}\right)^{k/2} (a_1 \dots a_k)^{-1/2} x_0^{m-k/2-2n-3}(A_{n+1}x_0 + B_{n+1}).$$

In what follows, assume that n is large, so that we have

$$2n + 1 \approx 2n + 2 \approx 2n + 3/2 \tag{13}$$

This yields

$$\frac{\partial \Delta}{\partial x_0}(x_0, n) \approx -\frac{\Delta(x_0, n)}{x_0}(2n + (3 + k)/2 - m). \tag{14}$$

The derivative with respect to n is more complicated. First we need expressions for the coefficients c_n and d_n for arbitrary real n, not just integers. It is not difficult to see that for n large enough,

$$|c_{\nu,n}| = \frac{|\cos(\pi\nu)|\,\Gamma(2n + 1/2 - \nu)\Gamma(2n + 1/2 + \nu)}{\pi 4^n \Gamma(2n + 1)}.$$

Differentiating gives

$$\frac{d\,|c_{\nu,n}|}{dn} = 2\,(\Psi(2n + 1/2 - \nu) + \Psi(2n + 1/2 + \nu) - \Psi(2n + 1) - \log 2)\,|c_{\nu,n}|$$

where Ψ is the Digamma function (the logarithmic derivative of the Gamma function). Taking the asymptotic formula 6.3.18 from [AS64] for large n, we get the approximation

$$\frac{d\,|c_{\nu,n}|}{dn} \approx 2\,|c_{\nu,n}| \log n.$$

Using this formula to differentiate A_n and with the same reasoning applied to $|d_{\nu,n}|$ and B_n we get

$$\frac{dA_n}{dn} \approx 2A_n \log n - 2\sum_{j=1}^{k} \left|c_{\nu_j,n}\right| a_j^{-2n} \log a_j,$$

$$\frac{dB_n}{dn} \approx 2B_n \log n - 2\sum_{j=1}^{k} \left|d_{\nu_j,n}\right| a_j^{-2n-1} \log a_j.$$

Differentiating the estimate for $\Delta(x_0, n)$ with respect to n and using the approximations above and formula (13), some computations yield

$$\frac{\partial \Delta}{\partial n}(x_0, n) \approx -2\Delta(x_0, n)\left(\log x_0 + \frac{1}{2n + (3 + k)/2 - m} - \log(n + 1) + \frac{\sum_{j=1}^{k} |\cos(\pi\nu_j)|\, a_j^{-2n-5/2} \log a_j}{\sum_{j=1}^{k} |\cos(\pi\nu_j)|\, a_j^{-2n-5/2}}\right).$$

For large n, the sums in the last term are dominated by the term corresponding to $\min_j a_j$, so we replace this ratio of sums by $\log \min_j a_j$. However, in the program we normalize the vector $\boldsymbol{a}$ such that $\min_j a_j = 1$, which means that we can just ommit this term from the above expression.

Putting everything together, equation (11) can be approximated by

$$(2n + (3+k)/2 - m)2^k t_{GJ} = \\ 2x_0 \left(\log x_0 + \frac{1}{2n + (3+k)/2 - m} - \log(n+1) \right) \frac{N}{2\pi} \sum_{j=1}^{k} a_j .$$

This transcendental equation can be solved for x_0 with the aid of the Lambert W-function $W(x)$ [CGH+96]. The result is

$$x_0(n) = \frac{\kappa}{2} \frac{2n + (3+k)/2 - m}{W\left[\frac{\kappa}{2} \frac{2n+(3+k)/2-m}{n+1} \exp\left(\frac{1}{2n+(3+k)/2-m} \right) \right]} \tag{15}$$

where we have introduced the constant

$$\kappa = \frac{2^{k+1} t_{GJ} \pi}{N \sum_{j=1}^{k} a_j} .$$

Surprisingly, if we plot $x_0(n)$ using (15), the result looks remarkably linear. If we compute a series development of $x_0(n)/n$ for $n \to \infty$, some computations give

$$\frac{x_0(n)}{n} = \frac{\kappa}{W(\kappa)} + \frac{\kappa}{4W(\kappa)} \left(3 + k - 2m - \frac{1+k-2m}{1+W(\kappa)} \right) \frac{1}{n} + O(n^{-2}). \tag{16}$$

Multiplying by n and taking only the first two terms in this series gives a simple linear relation between x_0 and n. Figure 2 illustrates how well this formula approximates the exact solution of equations (11)–(12). Using the Optimization Toolbox of Matlab, we computed the optimal parameters (x_0, n) for different precision levels. They are indicated with a $'+'$. The solid line corresponds to formula (16). The parameters for this illustration are $\boldsymbol{a} = [1, \sqrt{3/2}, \sqrt{5/2}]$ and $m = 1$.

Replacing x_0 in (12) by its linear approximation, we obtain a nonlinear equation in n, which can easily be solved using standard techniques. The subfunction `gop`, which computes the optimal parameters, contains an implementation of the Dekker-Brent algorithm to find the zero of a nonlinear equation. Two iterations are usually sufficient.

Another implementation issue is the computation of the Lambert W-function. This is not a standard Matlab function. Fortunately, for positive arguments, $W(x)$ can be computed very efficiently using the iterative scheme

$$w_{j+1} = \frac{x e^{-w_j} + w_j^2}{w_j + 1}$$

starting from the initial value $w_0 = 0$ for $x < 2$ or $w_0 = \log(x) - \log(\log(x))$ else. We recall that $W(x)$ satisfies $x = We^W$. The iterative scheme is a Newton

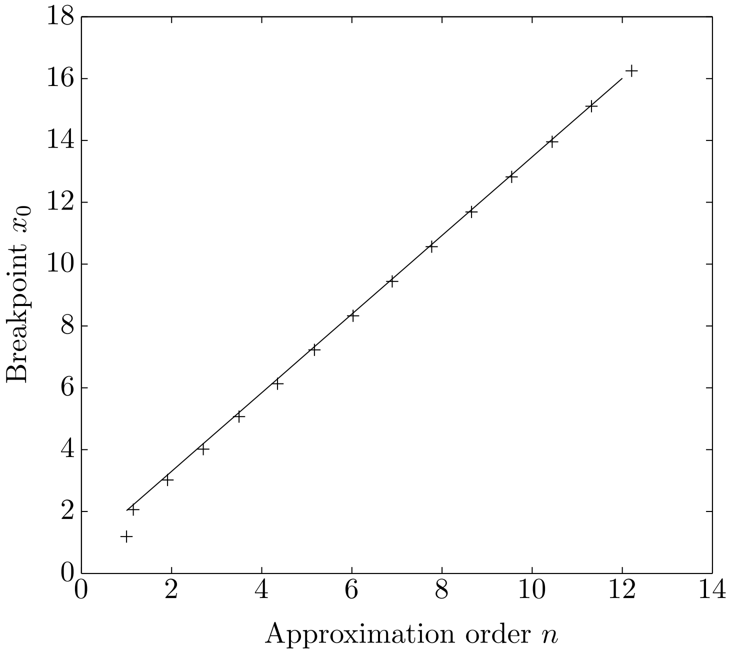

Figure 2: Exact solution of the optimization problem and linear approximation. Parameters are $\boldsymbol{a} = [1, \sqrt{3/2}, \sqrt{5/2}]$, $\boldsymbol{\nu} = [0, 0, 0]$ and $m = 1$.

iteration applied to this equation. The starting value for $x \geq 2$ follows from the asymptotic expansion of $W(x)$. This iteration is also part of `gop`.

Everything which was said before, also applies (with some minor modifications) to the case where the *relative* error should not exceed a certain required precision. The derivation now also uses equation (9) and its derivative with respect to x_0. As a result, the two $(k - 2m)$ terms in (16) disappear, but the rest of the expression remains the same.

Finally we mention that for the case of large orders ν_j, the coefficients in the asymptotic expansion grow very fast with increasing n. In this case, the 'optimal' n may be too large, leading to cancellation and even overflow. This is tested in the function `ira`, and when the coefficients $|c_{\nu,n}|$ or $|d_{\nu,n}|$ exceed a certain treshold (which was experimentally determined), the previous value of n is taken and a new breakpoint x_0 is solved from equation (12) using Newton-iteration.

References

[AS64] M. Abramowitz and I.A. Stegun. *Handbook of Mathematical Functions With Formulas, Graphs, and Mathematical Tables*, volume 55 of *Applied Mathematics Series.* National Bureau of Standards, Washington, D.C., 1964.

[BLWW04] F. Bornemann, D. Laurie, S. Wagon, and J. Waldvogel. *The SIAM 100-Digit Challenge. A Study in High-Accuracy Numerical Computing.* SIAM, Philadelphia, 2004. http://www-m8.ma.tum.de/m3/bornemann/challengebook.

[CGH+96] R.M. Corless, G.H. Gonnet, D.E.G. Hare, D.J. Jeffrey, and D.E. Knuth. On the Lambert W-function. *Advances in Computational Mathematics*, 5:329–359, 1996.

[Hen77] P. Henrici. *Applied and Computational Complex Analysis. Volume 2: Special Functions, Integral Transforms, Asymptotics, Continued Fractions.* John Wiley & Sons, New York, 1977.

[JM72] A. D. Jackson and L. C. Maximon. Integrals of products of Bessel functions. *SIAM J. Math. Anal.*, 3:446–460, 1972.

[KG87] E. Kostlan and D. Gokhman. A program for calculating the incomplete gamma function. Technical report, Dept. of Mathematics, Univ. of California, Berkeley, 1987.

[NC05] D. Nuyens and R. Cools. A multi-variate numerical integration toolbox for Matlab. 2005. In preparation. http://www.wogsymposium.ugent.be.

[Nic20] J.W. Nicholson. Generalisation of a theorem due to Sonine. *Quart. J. Math.*, 48:321–329, 1920.

[Sid87] A. Sidi. Extrapolation methods for divergent oscillatory infinite integrals that are defined in the sense of summability. *J. Comput. Appl. Math.*, 17:105–114, 1987.

[Sid88] A. Sidi. A user-friendly extrapolation method for oscillatory infinite integrals. *Math. Comp.*, 51(183):249–266, 1988.

[Son80] N.J. Sonine. Recherches sur les fonctions cylindriques et le développement des fonctions continues en séries. *Math. Ann.*, 16:1–80, 1880.

[Tre02] L.N. Trefethen. The $100, 100-Digit Challenge. *SIAM News*, 35(6):1–3, 2002.

[Wat66] G.N. Watson. *A treatise on the Theory of Bessel Functions.* Cambridge University Press, 1966.

[Win03] S. Winitzki. Computing the incomplete gamma function to arbitrary precision. In V. Kumar, M.L. Gavrilova, C.J.K. Tan, and P. L'Ecuyer, editors, *Computational Science and Its Applications - ICCSA*, volume 2667 of *Lecture Notes in Computer Science*, pages 790–798. Springer Verlag, 2003.