An automatic integration procedure for infinite range integrals involving oscillatory kernels

Takemitsu Hasegawa

Department of Information Science, Faculty of Engineering, Fukui University, Fukui, 910, Japan
E-mail: hasegawa@agauss.fuis.fukui-u.ac.jp

Avram Sidi

Computer Science Department, Technion – Israel Institute of Technology, Haifa 32000, Israel
E-mail: csssidi@technion.bitnet

Received 28 April 1995; revised 18 March 1996
Communicated by G. Mühlbach

Let the real functions $K(x)$ and $L(x)$ be such that $M(x) = K(x) + \mathrm{i}L(x) = \mathrm{e}^{\mathrm{i}x}g(x)$, where $g(x)$ is infinitely differentiable for all large x and is non-oscillatory at infinity. We develop an efficient automatic quadrature procedure for numerically computing the integrals $\int_a^\infty K(\omega t)f(t)\,\mathrm{d}t$ and $\int_a^\infty L(\omega t)f(t)\,\mathrm{d}t$, where the function $f(t)$ is smooth and non-oscillatory at infinity. One such example for which we also provide numerical results is that for which $K(x) = J_\nu(x)$ and $L(x) = Y_\nu(x)$, where $J_\nu(x)$ and $Y_\nu(x)$ are the Bessel functions of order ν. The procedure involves the use of an automatic scheme for Fourier integrals and the modified W-transformation which is used for computing oscillatory infinite integrals.

Keywords: Automatic integration, infinite oscillatory integral, Bessel function, Hankel transform, extrapolation, modified W-transformation, acceleration, Chebyshev expansion, FFT.

AMS subject classification: 65D30, 41A55, 65B05.

1. Introduction

Let the real functions $K(x)$ and $L(x)$ be such that

$$M(x) = K(x) + \mathrm{i}L(x) = \mathrm{e}^{\mathrm{i}x}g(x), \tag{1.1}$$

where the (in general complex) function $g(x)$ is infinitely differentiable for all large x and is non-oscillatory at infinity. There are many examples of such functions that arise in scientific applications. For instance,

(i) $K(x) = J_\nu(x)$ and $L(x) = Y_\nu(x)$, where $J_\nu(x)$ and $Y_\nu(x)$ are the Bessel functions of order ν of the first and second kinds, respectively,

(ii) $K(x) = \int_x^\infty J_\nu(t)\,\mathrm{d}t$ and $L(x) = \int_x^\infty Y_\nu(t)\,\mathrm{d}t$,

(iii) $K(x) = \int_x^\infty (\cos t/t)\,\mathrm{d}t$ and $L(x) = \int_x^\infty (\sin t/t)\,\mathrm{d}t$ that are related to the cosine and sine integrals respectively,

(iv) $K(x) = \int_x^\infty (\cos t/\sqrt{t})\,\mathrm{d}t$ and $L(x) = \int_x^\infty (\sin t/\sqrt{t})\,\mathrm{d}t$ that are related to the Fresnel integrals,

are but a few.

In the present work we present a general framework within which one can develop an automatic quadrature scheme for the numerical computation of infinite range integrals of the form

$$\int_a^\infty K(\omega t) f(t)\,\mathrm{d}t \quad \text{or} \quad \int_a^\infty L(\omega t) f(t)\,\mathrm{d}t, \quad a \geqslant 0,\ \omega > 0, \tag{1.2}$$

where $f(t)$ is a real function that is infinitely differentiable for all large t and is assumed to be non-oscillatory at infinity. The Hankel transforms

$$H_\nu[f;\omega] = \int_0^\infty tJ_\nu(\omega t)\, f(t)\,\mathrm{d}t,$$

form an important subclass of this family of integrals.

The problem of evaluation of oscillatory integrals of various sorts has been considered in the papers by Longman [25, 26], Gray and Atchison [14], Levin and Sidi [23], Sidi [35–38], Piessens and Haegemans [34], Espelid and Overholt [8], whose methods can be used to compute also integrals of the form (1.2). Methods for computing the Hankel transform specifically have been considered by Linz [24], Piessens and Branders [31, 32], Anderson [1], Lund [29], Lyness and Hines [30] and Sidi [35]. Recently, an automatic quadrature procedure for integrals of the form $\int_0^\infty \mathrm{e}^{\mathrm{i}\omega t} f(t)\,\mathrm{d}t$ has been given by Hasegawa and Torii [16], which tries to minimize the number of function evaluations for a given required accuracy level.

In this paper we combine the modified W-transformation (mW) of Sidi [38] with the approach taken in Hasegawa and Torii [16] to devise an automatic quadrature scheme for the integrals in (1.2) with $K(x)$ and $L(x)$ as defined in (1.1). We note that the mW-transformation has been demonstrated to be a very efficient and user-friendly method for coping with a large class of oscillatory infinite range integrals. To the best of our knowledge, there is no automatic quadrature scheme that treats the integrals mentioned above. There is a "semi-automatic" approach in QUADPACK [33] that can be used for Fourier and Hankel transform, but it is not as efficient as the procedure of the present work. We shall say more on this in section 7.

2. Description of the method

We shall restrict ourselves to the evaluation of the integral

$$Q(\omega) = \int_a^\infty K(\omega t) f(t) \mathrm{d}t. \tag{2.1}$$

By our assumption that $f(t)$ is a real function, we have

$$Q(\omega) = \Re \int_a^\infty M(\omega t) f(t)\,\mathrm{d}t. \tag{2.2}$$

For all the functions $K(x)$ and $L(x)$ that were mentioned in the first paragraph of section 1 it turns out that a polynomial approximation is provided for them in a finite interval $[0, c]$, and, in the interval $[c, \infty)$, the function $g(x)$ of (1.1) is approximated by a polynomial in $1/x$. Normally these approximations are obtained by truncating the appropriate Chebyshev polynomial expansions. For example, for the functions $J_\nu(x)$ and $Y_\nu(x)$, Luke [28, p. 322] gives the expansions

$$J_\nu(x) = \left(\frac{x}{8}\right)^\nu \sum_{n=0}^{\infty} a_n^{(\nu)}\, T_{2n}\left(\frac{x}{8}\right), \quad |x| \leqslant 8,\ \nu = 0, 1,$$

and

$$J_\nu(x) + \mathrm{i}Y_\nu(x) = \mathrm{e}^{\mathrm{i}x} \frac{(-1)^\nu - \mathrm{i}}{\sqrt{\pi x}} \sum_{n=0}^{\infty} c_n^{(\nu)}\, T_n^*\left(\frac{5}{x}\right), \quad x \geqslant 5,\ \nu = 0, 1,$$

where $T_k(x)$ and $T_k^*(x)$ are, respectively, the Chebyshev and shifted Chebyshev polynomials of order k. What matters to us is the fact that these approximations in the above mentioned intervals are known. We do not care how they are given.

We now subdivide the interval in (2.2) in the form

$$Q(\omega) = Q_1(\omega) + Q_2(\omega), \tag{2.3}$$

where

$$Q_1(\omega) = \begin{cases} \int_a^{c/\omega} K(\omega t)\, f(t)\,\mathrm{d}t & \text{if } a < c/\omega, \\ 0 & \text{if } a \geqslant c/\omega, \end{cases} \tag{2.4a}$$

$$Q_2(\omega) = \Re \int_{\max(a, c/\omega)}^\infty M(\omega t)\, f(t)\,\mathrm{d}t. \tag{2.4b}$$

2.1. Computation of $Q_1(\omega)$

For the computation of $Q_1(\omega)$ when $a < c/\omega$ we use the Clenshaw and Curtis (CC) method [4] along with a modified FFT due to Hasegawa et al. [21], since the integrand $K(\omega t) f(t)$ is smooth in the interval $[a, c/\omega]$. For values of ω that are very small the interval $[a, c/\omega]$ may become too large for the integral $Q_1(\omega)$ to be evaluated at once. In such a case it may be advisable to break this interval into several smaller subintervals, and apply the CC method to each subinterval separately.

2.2. Computation of $Q_2(\omega)$

From (1.1) and (2.4b) we have

$$Q_2(\omega) = \Re \int_d^\infty \mathrm{e}^{\mathrm{i}\omega t}\, g(\omega t)\, f(t)\, \mathrm{d}t,$$

where $d = \max(a, c/\omega)$. We now have to evaluate this Fourier integral efficiently. The method that we use for this purpose is the mW-transformation of [38].

We start by letting

$$x_0 = \frac{\pi}{\omega}\left(\left\lfloor \frac{\omega d}{\pi} \right\rfloor + 1\right), \qquad x_l = x_0 + \frac{l\pi}{\omega}, \quad l = 1, 2, \ldots .$$

Here x_0 is simply the first zero of $\sin \omega t$ that is greater than d, and x_l is the lth zero following x_0. We next compute numerically the finite range integrals $F(x_l)$, where

$$F(x_l) = \int_d^{x_l} \mathrm{e}^{\mathrm{i}\omega t} g(\omega t)\, f(t)\, \mathrm{d}t, \quad l = 0, 1, 2, \ldots . \tag{2.5}$$

We finally compute a two-dimensional array of approximations $W_n^{(j)}$ to the integral $\int_d^\infty \mathrm{e}^{\mathrm{i}\omega t} g(\omega t) f(t)\, \mathrm{d}t$ by solving the linear system

$$F(x_l) = W_n^{(j)} + \psi(x_l) \sum_{i=0}^{n} \frac{\beta_i}{x_l^i}, \quad j \leqslant l \leqslant j + n + 1, \tag{2.6}$$

where

$$\psi(x_l) = \int_{x_l}^{x_{l+1}} \mathrm{e}^{\mathrm{i}\omega t} g(\omega t)\, f(t)\, \mathrm{d}t = F(x_{l+1}) - F(x_l), \quad l = 0, 1, \ldots, \tag{2.7}$$

and the β_i serve as the additional unknowns. In addition, we define $F(x_{-1}) = 0$ in (2.7). The solution of the linear system in (2.6) can be achieved in a very efficient manner by the W-algorithm [37], which is given below:

- set $M_{-1}^{(s)} = F(x_s)/\psi(x_s)$ and $N_{-1}^{(s)} = 1/\psi(x_s)$, $\quad s = 0, 1, \ldots;$
- for $s = 0, 1, \ldots$ and $p = 0, 1, \ldots$ compute

$$M_p^{(s)} = \left(M_{p-1}^{(s)} - M_{p-1}^{(s+1)}\right) / \left(x_s^{-1} - x_{s+p+1}^{-1}\right),$$

$$N_p^{(s)} = \left(N_{p-1}^{(s)} - N_{p-1}^{(s+1)}\right) / \left(x_s^{-1} - x_{s+p+1}^{-1}\right),$$

  and set $W_p^{(s)} = M_p^{(s)} / N_p^{(s)}$.

As has been shown in [38], the sequences $\{W_n^{(j)}\}_{n=0}^\infty$ for fixed j have very favorable convergence properties. Therefore, we choose to consider only the sequences $\{W_n^{(0)}\}_{n=0}^\infty$. Furthermore, $|W_{n+1}^{(0)} - W_n^{(0)}|$ may provide an error estimate for the approximation $W_{n+1}^{(0)}$, which is probably the best of the approximations $W_p^{(j)}$, $j + p = n + 1$. For more details the reader is referred to the original papers [37, 38]. Once the best $W_n^{(j)}$ has been computed, $Q_2(\omega)$ is approximated by $\Re W_n^{(0)}$.

The problem that remains is that of computing the $\psi(x_j)$ that were defined in (2.7) to the level of accuracy prescribed by the user. This is the topic of the next section.

3. Computation of the finite oscillatory integrals

As mentioned in the previous section, the mW-transformation requires the sequence of the finite integrals $\psi(x_l)$, $l=-1,0,1,\ldots$ given in (2.7) or, equivalently, the integrals $F(x_l)$ defined by (2.5). The computation of the $\psi(x_l)$ can be performed accurately by an appropriate quadrature rule such as Gaussian formula. If the integrand function $f(t)$ is smooth, however, it might be more efficient to devise a quadrature method for computing several, say r, integrals $\psi(x_{s+1}),\ldots,\psi(x_{s+r})$ or the indefinite integral $\int_{x_s}^{x} \mathrm{e}^{\mathrm{i}\omega t} g(\omega t) f(t)\,\mathrm{d}t$ where we take $x=x_{s+i}$ $(i=1,2,\ldots,r)$, at a time for an arbitrary integer s.

Indeed, for a positive integer m and a non-negative integer μ, define $s=m+\mu r$ and subdivide the integration interval $[c/\omega, x_{s+l}]$ for the integral $F(x_{s+l})$, $0<l\leqslant r$, into $\mu+1$ subintervals K_q $(q=-1,0,1,\ldots,\mu-1)$, and an extra one $(x_s, x_{s+l}]$ as follows:

$$[c/\omega, x_{s+l}] = \left(\bigcup_{q=-1}^{\mu-1} K_q\right) \cup\ (x_s, x_{s+l}],$$

where we take $K_q = (x_{m+qr}, x_{m+qr+r}]$ $(q=0,1,\ldots)$ and in particular $K_{-1} = [c/\omega, x_m]$; the appropriate values of m and r are determined later. Then, we have

$$F(x_{s+l}) = \sum_{q=-1}^{\mu-1} F(K_q) + \int_{x_s}^{x_{s+l}} \mathrm{e}^{\mathrm{i}\omega t} g(\omega t) f(t)\,\mathrm{d}t,$$

$$1\leqslant l\leqslant r,\ \ s=m+\mu r,\ \ \mu=0,1,\ldots, \tag{3.1}$$

where $F(K_q)$ is defined by

$$F(K_q) = \int_{t\in K_q} \mathrm{e}^{\mathrm{i}\omega t} g(\omega t) f(t)\,\mathrm{d}t = \int_{x_{m+qr}}^{x_{m+qr+r}} \mathrm{e}^{\mathrm{i}\omega t} g(\omega t) f(t)\,\mathrm{d}t.$$

The knowledge of the indefinite integrals $\int_{x_s}^{x} \mathrm{e}^{\mathrm{i}\omega t} g(\omega t) f(t)\,\mathrm{d}t$, where $s=m+\mu r$ and $x\in K_\mu$ $(\mu=0,1,\ldots)$, could enable the efficient evaluation of each integral in the right-hand side of (3.1).

To this end, here we briefly describe an automatic quadrature method given in [16, 18] with some modifications, to approximate the indefinite integral $\int_{\alpha}^{x} \mathrm{e}^{\mathrm{i}\omega t} g(\omega t) f(t)\,\mathrm{d}t$, for $\alpha\leqslant x\leqslant\beta$, where for example we set $\alpha=x_s$ and $\beta=x_{s+r}$ for obtaining the integral on $[x_s, x_{s+r}]$. Let $\phi\colon[\alpha,\beta]\to[-1,1]$ be a linear function defined by

$$\phi(t) = (2t-\beta-\alpha)/(\beta-\alpha), \qquad \phi(\alpha)=-1, \qquad \phi(\beta)=1, \tag{3.2}$$

and approximate the non-oscillatory part $g(\omega t)\,f(t)$ in the integral $\int_{\alpha}^{x} \mathrm{e}^{\mathrm{i}\omega t} g(\omega t) f(t)\,\mathrm{d}t$ by a sum $P_N(t)$ of the Chebyshev polynomials $T_k(t)$:

$$P_N(t) = p_N\big(\phi(t)\big) \equiv {\sum_{k=0}^{N}}' a_k^N T_k\big(\phi(t)\big), \quad \alpha\leqslant t\leqslant\beta, \tag{3.3}$$

where the prime denotes the summation whose first term is halved. Then, defining $W = (\beta - \alpha)/2$ and $T = (\beta + \alpha)/2$ we have

$$\int_\alpha^x e^{i\omega t} g(\omega t) f(t)\,dt \sim \int_\alpha^x e^{i\omega t} P_N(t)\,dt = W \exp(i\omega T) I\big(\omega W, \phi(x); p_N\big), \qquad (3.4)$$

where $I(\omega, x; p)$ is defined by

$$I(\omega, x; p) = \int_{-1}^x e^{i\omega t} p(t)\,dt, \quad -1 \leqslant x \leqslant 1. \qquad (3.5)$$

It is efficient to choose r to be a larger positive integer so long as $f(t)$ is a sufficiently smooth function on the interval $[\alpha, \beta]$, whence one can expect that the truncated Chebyshev series (3.3) converges rapidly as N increases, since $g(t)$ is a smooth function, too. Several numerical experiments suggest that the near optimum choice of the integer r depends on the tolerance ε_2 for the integral $Q_2(\omega)$ (2.4b) to minimize the total number of function evaluations required to satisfy ε_2. Let $M = [-\log_{10} \varepsilon_2]$, then in view of the observation that the mW-transformation converges so rapidly for slowly convergent integrals that $M+2$ finite integrals $\psi(x_i)$, $-1 \leqslant i \leqslant M$, might be sufficient to achieve the accuracy ε_2, we determine empirically for (3.1) that $m = 2$ and $r = 3 + 0.7M$. It remains an open problem to determine the optimum values of m and r depending on the required accuracy ε_2 and the class of the given function $f(t)$.

Now we proceed to evaluate the indefinite integral in the right of (3.4) or $I(\omega, x; p)$ given by (3.5) with $p(t)$ replaced by $p_N(t)$ in (3.3). We will see that one can use an auxiliary function $H(x)$ to write the integral $\int_{-1}^x e^{i\omega t} p_N(t)\,dt$ as follows:

$$I(\omega, x; p_N) \equiv \int_{-1}^x e^{i\omega t} p_N(t)\,dt = \frac{e^{i\omega x} H(x)}{i\omega}, \quad -1 \leqslant x \leqslant 1. \qquad (3.6)$$

Differentiating both sides of (3.6) with respect to x, yields the first order differential equation

$$\frac{1}{i\omega} \frac{dH(x)}{dx} + H(x) = p_N(x), \qquad (3.7)$$

which is further integrated from -1 to x to give

$$\frac{H(x) - H(-1)}{i\omega} + \int_{-1}^x H(t)\,dt = \int_{-1}^x p_N(t)\,dt. \qquad (3.8)$$

To solve (3.8), we expand $H(t)$ in terms of Chebyshev polynomials:

$$H(t) = \sum_{k=0}^{\infty}{}' b_k\, T_k(t). \qquad (3.9)$$

Substitute (3.3) and (3.9) into (3.8) and use the relation:

$$2\int T_k(t)\,\mathrm{d}t = \frac{T_{k+1}(t)}{k+1} - \frac{T_{k-1}(t)}{k-1} + \text{const}, \quad k \geqslant 2.$$

Then, comparing the coefficients of the Chebyshev polynomials, we have

$$b_{k-1} + \frac{2k}{\mathrm{i}\omega} b_k - b_{k+1} = a_{k-1}^N - a_{k+1}^N, \quad 1 \leqslant k. \tag{3.10}$$

For convenience, we set $a_k^N \equiv 0$ $(k > N)$. We have omitted the dependence of the coefficients b_k in (3.9) on ω as well as the coefficients $\{a_k^N\}$ of $p_N(t)$.

To make use of the three-term recurrence relations (3.10) to evaluate the coefficients b_k in (3.9) in a numerically stable manner (see, e.g., Gautschi [10] and Lozier [27] for the numerical stability of the recurrence relations), we need another equation, (3.11) below,

$$\sum_{k=0}^{N}{}' (-1)^k b_k = 0, \tag{3.11}$$

which can be derived from (3.9) and the condition for (3.6) that $H_N(-1) = 0$.

Now, we can obtain b_k by solving a system of linear equations [19, 27] of a coefficient matrix A derived from (3.10) and (3.11), which we describe in the following. Let K be a large integer greater than N. Further, let $M = \lfloor|\omega|\rfloor$ and for $0 < M < K$ let A be a $(K+1) \times (K+1)$-matrix defined by

$$A = \begin{pmatrix}
1 & d_0 & -1 & 0 & \cdots & & & & \cdots & 0 \\
0 & 1 & d_1 & -1 & 0 & \cdots & & & \cdots & 0 \\
\vdots & & \ddots & \ddots & \ddots & & & & & \vdots \\
0 & \cdots & 0 & 1 & d_{M-1} & -1 & 0 & & \cdots & 0 \\
\lambda_0 & \lambda_1 & \cdots & \lambda_{M-1} & \lambda_M & \lambda_{M+1} & \lambda_{M+2} & \cdots & \cdots & \lambda_K \\
0 & \cdots & & 0 & 1 & d_M & -1 & 0 & \cdots & 0 \\
0 & \cdots & & & 0 & 1 & d_{M+1} & -1 & \cdots & 0 \\
\vdots & & & & & & \ddots & \ddots & \ddots & \vdots \\
0 & \cdots & & & & \cdots & 0 & 1 & d_{K-2} & -1 \\
0 & \cdots & & & & & \cdots & 0 & 1 & d_{K-1}
\end{pmatrix}, \tag{3.12}$$

where $d_{k-1} = 2k/(\mathrm{i}\omega)$ $(k = 1, 2, \ldots, K)$, $\lambda_0 = 1/2$ and $\lambda_k = (-1)^k$ $(k = 1, 2, \ldots, K)$. When $M = 0$ or $M \geqslant K$, A is a matrix having $\lambda_0, \ldots, \lambda_K$ on the first row or on the last one, respectively. Further, let $\mathbf{a}^T = (a_0, a_1, \ldots, a_K)$ and let $a_k = a_k^N - a_{k+2}^N$ $(k = 0, 1, \ldots, M-1)$, $a_M = 0$ and $a_{k+1} = a_k^N - a_{k+2}^N$ $(k = M, M+1, \ldots, K-1)$, where we take $a_0^N/2$ instead of a_0^N and note that $a_k^N \equiv 0$ $(k > N)$. Then the coefficients b_k $(k = 0, 1, \ldots, K)$ can be stably and efficiently computed from the solution of the linear system of equations $A\mathbf{b} = \mathbf{a}$ by LU decomposition without pivoting, see appendix A for more details of an algorithm for the LU decomposition. See also Cash [3] for the LU factorization of matrices derived from recurrence relations.

4. Chebyshev series expansion and the FFT

We here describe how to construct the sequence of the polynomials $\{p_N\}$ of (3.3) using a modified FFT [21], which can be efficiently used in the CC method [4] and its extension for evaluating the integrals $Q_2(\omega)$ (2.4b) to the required accuracy. It suffices to consider only the indefinite integral $I(\omega, x; f)$ (3.5) on the interval $[-1, 1]$ and its approximation $I_N \equiv I(\omega, x; p_N)$, since an arbitrary finite interval can be easily transformed into $[-1, 1]$ by a linear function such as $\phi(t)$ (3.2).

Incidentally, an automatic quadrature of non-adaptive type is generally constructed from the sequence of the approximations $\{I_N\}$ converging to the integral, having an adequate method of error estimation, until a stopping criterion is satisfied. It is a usual and simple way to double the degree N of $p_N(t)$ (3.3) for generating the sequence $\{I_N\}$. In order to make an automatic quadrature efficient, however, it is advantageous to have more chances of checking the stopping criterion than doubling N. Hasegawa et al. [21] showed that the degree N of $p_N(t)$ can be allowed to take the form 3×2^n and 5×2^n as well as 2^n.

Here and henceforth we assume that N is a power of 2, 2^n $(n = 2, 3, \ldots)$, unless otherwise stated. Now, we outline the iterative procedure [21] for computing the sequence of the truncated Chebyshev series, $\{p_N, p_{5N/4}, p_{3N/2}\}$, $N = 2^n$, $n = 2, 3, \ldots$, until a stopping criterion described in section 5 is satisfied.

Let $t_j^N = \cos(\pi j/N)$ $(0 \leqslant j \leqslant N)$ be the zeros of the polynomial $w_{N+1}(t)$ defined by

$$w_{N+1}(t) = T_{N+1}(t) - T_{N-1}(t) = 2(t^2 - 1)\, U_{N-1}(t), \tag{4.1}$$

where $U_{N-1}(t)$ denotes the Chebyshev polynomial of the second kind defined by $U_{N-1}(t) = \sin(N\theta)/\sin\theta$, $t = \cos\theta$. Then, the coefficients a_k^N of $p_N(t)$ (3.3) are determined [4] so that $p_N(t)$ interpolates $f(t)$ at the abscissae t_j^N, and consequently a_k^N is represented in the form:

$$a_k^N = \frac{2}{N} \sum_{j=0}^{N}{}'' f\left(\cos\frac{\pi j}{N}\right) \cos\frac{\pi k j}{N}, \quad 0 \leqslant k \leqslant N. \tag{4.2}$$

The right-hand side of (4.2) is known to be efficiently computed by means of the FFT for real data [12].

For integer $\sigma = 2$ and 4, let $\{v_j^{N/\sigma}\}$ $(0 \leqslant j < N/\sigma)$ be a subset of the zeros of $T_N(t)$, in particular, be chosen to agree with a set consisting of the N/σ zeros of $T_{N/\sigma}(t) - \cos 3\pi/(2\sigma)$. Then, we represent the polynomials $p_{N+N/\sigma}(t)$ $(\sigma = 2, 4)$ interpolating $f(t)$ at the nodes $v_j^{N/\sigma}$, $0 \leqslant j < N/\sigma$ $(\sigma = 2, 4)$, as well as at the zeros of $w_{N+1}(t)$ (4.1) in the Newton form:

$$\begin{aligned} p_{N+N/\sigma}(t) - p_N(t) &= -w_{N+1}(t) \sum_{k=1}^{N/\sigma} B_k^{N/\sigma}\, U_{k-1}(t) \\ &= \sum_{k=1}^{N/\sigma} B_k^{N/\sigma} \{T_{N-k}(t) - T_{N+k}(t)\}. \end{aligned} \tag{4.3}$$

The coefficients $\{B_k^{N/\sigma}\}$ are determined to satisfy the condition

$$f(v_j^{N/\sigma}) = p_{N+N/\sigma}(v_j^{N/\sigma}), \quad 0 \leqslant j < N/\sigma,\ \sigma = 2,4,$$

and the FFT [21] is used for efficiently evaluating the coefficients $B_k^{N/\sigma}$. We note that the set of $N/4$ abscissae $\{v_j^{N/4}\}$ $(0 \leqslant j < N/4)$ for $p_{5N/4}(t) - p_N(t)$ is contained in the $N/2$ abscissae $\{v_j^{N/2}\}$ $(0 \leqslant j < N/2)$ for $p_{3N/2}(t) - p_N(t)$, which is also included in the set of the N zeros of $T_N(t)$ $(= w_{2N+1}(t)/\{2w_{N+1}(t)\})$ for $p_{2N}(t) - p_N(t)$. This fact allows the iterative algorithm to compute the sequence $\{p_{3m}, p_{4m}, p_{5m}\}$ $(m = 2^n,\ n = 1,2,\ldots)$ using the FFT, see [21] for details.

5. Error estimate

We estimate the truncation errors of the approximations $I(\omega, x; p_{N+mN/4})$ $(m = 0,1,2)$ to the integral $I(\omega,x,f)$ (3.5), where $p_{N+mN/4}(t)$ $(m = 0,1,2)$ are given by (3.3) and (4.3).

Let ε_ρ denote the ellipse in the complex plane $z = x+\mathrm{i}y$ with foci $(-1,0)$, $(1,0)$ and semimajor axis $a = (\rho+\rho^{-1})/2$ and semiminor axis $b = (\rho-\rho^{-1})/2$ for a constant $\rho > 1$. Assume that $f(z)$ is single-valued and analytic inside and on ε_ρ. Then, the error of the interpolating polynomials $p_N(t)$ and $p_{N+N/\sigma}(t)$ $(\sigma = 2,4)$, can be expressed in terms of the contour integral [7, 17]:

$$f(t) - p_N(t) = \frac{1}{2\pi \mathrm{i}} \oint_{\varepsilon_\rho} \frac{w_{N+1}(t)\, f(z)\,\mathrm{d}z}{(z-t)\, w_{N+1}(z)}, \tag{5.1}$$

$$f(t) - p_{N+N/\sigma}(t) = \frac{1}{2\pi \mathrm{i}} \oint_{\varepsilon_\rho} \frac{w_{N+1}(t)\{T_{N/\sigma}(t) - \cos(3\pi/(2\sigma))\}\, f(z)\,\mathrm{d}z}{(z-t)\, w_{N+1}(z)\{T_{N/\sigma}(z) - \cos(3\pi/(2\sigma))\}}, \quad \sigma = 2,4, \tag{5.2}$$

respectively. Now for $t \in [-1,1]$

$$\frac{1}{z-t} = \frac{2}{\pi} \sum_{k=0}^{\infty}{}' \, \widetilde{U}_k(z)\, T_k(t), \tag{5.3}$$

where $\widetilde{U}_k(z)$ is the Chebyshev function of the second kind defined by

$$\widetilde{U}_k(z) = \int_{-1}^{1} \frac{T_k(t)\,\mathrm{d}t}{(z-t)\sqrt{1-t^2}} = \frac{\pi}{\sqrt{z^2-1}\, u^k} = \frac{2\pi}{(u-u^{-1})\, u^k}, \tag{5.4}$$

$u = z + \sqrt{z^2-1}$ and $|u| > 1$ for $z \notin [-1,1]$ (see [11, 20]). Using (5.3) in (5.1) and (5.2) enables us to expand in terms of Chebyshev polynomials, the errors for the interpolating polynomials $p_N(t)$ and $p_{N+N/\sigma}(t)$ as follows:

$$f(t) - p_N(t) = w_{N+1}(t) \sum_{k=0}^{\infty}{}' \, V_k^N(f)\, T_k(t), \tag{5.5}$$

$$\begin{aligned} & f(t) - p_{N+N/\sigma}(t) \\ & \quad = w_{N+1}(t)\left\{T_{N/\sigma}(t) - \cos\frac{3\pi}{2\sigma}\right\}\sum_{k=0}^{\infty}{}' V_k^{N+N/\sigma}(f)\, T_k(t), \quad \sigma = 2,4, \end{aligned} \tag{5.6}$$

where the coefficients $V_k^N(f)$ and $V_k^{N+N/\sigma}(f)$ are given by

$$V_k^N(f) = \frac{1}{\pi^2 \mathrm{i}}\oint_{\varepsilon_\rho}\frac{\widetilde{U}_k(z)\ f(z)\,\mathrm{d}z}{w_{N+1}(z)}, \quad k \geqslant 0, \tag{5.7}$$

$$V_k^{N+N/\sigma}(f) = \frac{1}{\pi^2\,\mathrm{i}}\oint_{\varepsilon_\rho}\frac{\widetilde{U}_k(z)\ f(z)\,\mathrm{d}z}{w_{N+1}(z)\{T_{N/\sigma}(z) - \cos(3\pi/(2\sigma))\}}, \quad k \geqslant 0,\ \sigma = 2,4, \tag{5.8}$$

respectively.

Using (5.5) in (3.5) yields the error for the approximate integral $I(\omega, x; p_N)$

$$I(\omega, x; f) - I(\omega, x; p_N) = I(\omega, x; f - p_N) = \sum_{k=0}^{\infty}{}'\ V_k^N(f)\ \Omega_k^N(x), \tag{5.9}$$

where $\Omega_k^N(x)$ is defined by

$$\Omega_k^N(x) = \int_{-1}^{x} \mathrm{e}^{\mathrm{i}\omega t}\ w_{N+1}(t)\ T_k(t)\,\mathrm{d}t, \tag{5.10}$$

and further, $\Omega_k^N(x)$ can be bounded by $|\Omega_k^N(x)| \leqslant 4$, independently of N, k, ω and x for $|x| \leqslant 1$. Similarly, we have the error for the approximation $I(\omega, x; p_{N+N/\sigma})$ depending on the interpolating polynomial $p_{N+N/\sigma}(t)$ (4.3):

$$\begin{aligned} & I(\omega, x; f) - I(\omega, x; p_{N+N/\sigma}) = I(\omega, x; f - p_{N+N/\sigma}) \\ & \quad = \frac{1}{2}\sum_{k=0}^{\infty}{}'\left\{\Omega_k^{N+N/\sigma}(x) + \Omega_k^{N-N/\sigma}(x) - 2\,\Omega_k^N(x)\,\cos\frac{3\pi}{2\sigma}\right\} V_k^{N+N/\sigma}(f). \end{aligned} \tag{5.11}$$

Suppose that $f(z)$ is a meromorphic function which has M simple poles at the points z_m $(m = 1, 2, \ldots, M)$ outside ε_ρ with residue $\mathrm{Res} f(z_m)$. Then, performing the contour integral of (5.7) yields

$$V_k^N(f) = \frac{1}{\pi^2 \mathrm{i}}\oint_E \frac{\widetilde{U}_k(z)\ f(z)\,\mathrm{d}z}{w_{N+1}(z)} = -\frac{2}{\pi}\sum_{m=1}^{M}\frac{\mathrm{Res} f(z_m)\ \widetilde{U}_k(z_m)}{w_{N+1}(z_m)}, \quad k \geqslant 0, \tag{5.12}$$

where E is an ellipse having foci at ± 1 such that the poles z_m $(m = 1, 2, \ldots, M)$ are in E and no other singularity of $f(z)$ exists. Now, noting that $T_k(z) = (u^k + u^{-k})/2$

for complex $z = (u+u^{-1})/2 \notin [-1,1]$ where $|u| > 1$, we have from (4.1) $w_{N+1}(z) = \sqrt{z^2-1}(u^N - u^{-N})$, which is combined with (5.4) to yield

$$\frac{\widetilde{U}_k(z)}{w_{N+1}(z)} = \frac{\pi}{z^2-1}\frac{1}{u^k\,(u^N - u^{-N})}. \tag{5.13}$$

The most dominant term in the right of (5.12) is obtained for the poles z_j for which

$$\left|z_j + \sqrt{z_j^2 - 1}\right| = \min_{1\leqslant m\leqslant M}\left|z_m + \sqrt{z_m^2-1}\right| \equiv r > 1.$$

If we assume that there is only one such z_j, we have $V_k^N(f) \sim V_0^N(f)u_j^{-k}$ for sufficiently large N, where $u_j = z_j + \sqrt{z_j^2 - 1}$. This fact and (5.9) permit us to estimate the error:

$$\begin{aligned}\left|I(\omega,x;f-p_N)\right| &\leqslant 4\sum_{k=0}^{\infty}{}'\left|V_k^N(f)\right| \sim 4\left|V_0^N(f)\right|\sum_{k=0}^{\infty}{}' r^{-k}\\ &= 4\left|V_0^N(f)\right|\frac{r+1}{2\,(r-1)},\end{aligned} \tag{5.14}$$

Now, we wish to estimate $|V_0^N(f)|$ in terms of the available coefficients a_k^N of $p_N(t)$. Elliott [7] gives

$$a_k^N = \frac{2}{\pi \mathrm{i}}\oint_{\varepsilon_\rho}\frac{T_{N-k}(z)\,f(z)}{w_{N+1}(z)}\,\mathrm{d}z, \quad 0 \leqslant k \leqslant N.$$

Performing the contour integral and comparing the result with (5.12) gives the relations $|V_0^N| \sim |a_N^N| r/(r^2-1)$ and $|a_k^N| \sim r|a_{k+1}^N|$, unless the poles z_m are close to the ranges $[-1,1]$ on the real axis. From these relations and (5.14), we have the estimate R_N of the truncation error $|I(\omega,x;f-p_N)|$

$$R_N = \frac{4(|a_N^N|/2)r}{(r-1)^2}. \tag{5.15}$$

The constant r may be estimated from the asymptotic behavior of $\{a_k^N\}$ [21].

Next, we wish to estimate the error (5.11) in terms of the computed $B_k^{N/\sigma}$, which is expressed in the contour integral [17]:

$$B_k^{N/\sigma} = \frac{-1}{\pi \mathrm{i}}\oint_{\varepsilon_\rho}\frac{T_{N/\sigma-k}(z)f(z)\,\mathrm{d}z}{w_{N+1}(z)\{T_{N/\sigma}(z) - \cos(3\pi/(2\sigma))\}}, \tag{5.16}$$
$$1 \leqslant k \leqslant N/\sigma,\ \sigma = 2,4,$$

where the right-hand side of (5.16) is multiplied by $1/2$ when $k = N/\sigma$. Performing the contour integrals in (5.8) and (5.16) and comparing the both results yield estimates

$$|V_0^{N+N/\sigma}| \sim \frac{4|B_{N/\sigma}^{N/\sigma}|\, r}{r^2 - 1}, \qquad |V_k^{N+N/\sigma}| = \mathrm{O}\big(r^{-k-N-N/\sigma}\big)$$

and $|B_k^{N/\sigma}| \sim r|B_{k+1}^{N/\sigma}|$. Using these relations and (5.8) in (5.11) yields estimates $R_{N+N/\sigma}$ of the truncation errors for the approximates $I(\omega, x; p_{N+N/\sigma})$ ($\sigma = 2, 4$) as follows:

$$R_{N+N/\sigma} = \frac{8(1 + |\cos(3\pi/(2\sigma))|)\ |B_{N/\sigma}^{N/\sigma}| r}{(r-1)^2}. \tag{5.17}$$

The relations (5.15) and (5.17) indicate that the errors are estimated independently of the value of ω. Thus, the errors for the quadrature rules $I(0, 1; p_{N+mN/4})$ ($m = 0, 1, 2$), to the non-oscillatory integral $I(0, 1; f) = \int_{-1}^{1} f(t)\,\mathrm{d}t$ can also be estimated by (5.15) and (5.17), respectively. In the next section we will make use of the error estimations (5.15) and (5.17) to derive the stopping criterion in the automatic quadrature for $Q(\omega)$.

6. Stopping criterion

The efficiency of an automatic quadrature scheme depends on an adequate stopping criterion based on an error estimate as well as on the use of appropriate quadrature rules.

We remember from (2.3) that the integral $Q(\omega)$ is divided into the two integrals $Q_1(\omega)$ on $[a, c/\omega]$ and $Q_2(\omega)$ on $[c/\omega, \infty)$. We want to approximate both integrals to assigned tolerances ε_1 and ε_2, respectively, so as to attain the overall accuracy $\varepsilon = \varepsilon_1 + \varepsilon_2$ for $Q(\omega)$ by using the CC method and its extension described in sections 2, 3 and 5, with as small a number of function evaluations as possible. Now, we have to determine the adequate values of ε_1 and ε_2 for the integrals $Q_1(\omega)$ and $Q_2(\omega)$, respectively. The result of numerical experiments suggests to choose $\varepsilon_1 = \varepsilon/20$ and $\varepsilon_2 = 19\varepsilon/20$, see [9] and [22, p. 173] for a detailed discussion on a more general topic, the software interface problem.

Further, we have seen that the infinite integral $Q_2(\omega)$ (2.4b) can be efficiently approximated by using the approximations to the finite integrals $F(x_i)$ ($i = -1, 0, 1, \ldots$) (2.5) or $F(x_{s+l})$ in (3.1) along with the mW-transformation. The next question is how to assign the tolerance to each $F(K_q)$ ($q = -1, 0, 1, \ldots$) in (3.1) on the interval K_q. It may in general be difficult to know at the outset how many integrals $F(K_q)$ ($q = -1, 0, 1, \ldots$) are required in the mW-transformation to attain the assigned accuracy ε_2 for $Q_2(\omega)$. Numerical experiments, however, suggest that since the mW-transformation can transform a large class of convergent infinite oscillatory integrals into very quickly convergent ones, two or (at most) three intervals K_q ($q = -1, 0$ or 1)

(note that we have determined the K_q depending on ε_2) are sufficient to obtain the tolerance ε_2.

From the observation above we empirically determine the tolerance to each integral $F(K_q)$ on the K_q ($q = -1, 0, 1, \ldots$) as follows. Assume that $f(x)$ in $Q_2(\omega)$ is a smooth function of slow convergence at infinity, and that only three intervals K_q ($q = -1, 0, 1$) are enough. Then we assign the tolerance $\varepsilon_2/3$ to each integral $F(K_q)$ ($q = -1, 0, 1$).

If we have no knowledge of how many finite integrals $F(K_q)$ are required in the mW-transformation, an alternative and most conservative method of assigning the tolerances may be to assign $\varepsilon_2/2^{q+2}$ to $F(K_q)$ ($q = -1, 0, 1, \ldots, \infty$), to obtain the required accuracy $\varepsilon_2 = \sum_{q=-1}^{\infty}(\varepsilon_2/2^{q+2})$ for the infinite integral $Q_2(\omega)$ (2.4b). In this case, the automatic quadrature could be more reliable but certainly less efficient, and to make matters worse we might fail to obtain the convergent result because of a more stringent tolerance for bigger q of K_q.

We conclude this section by summarizing our stopping criteria. For the integral $Q_1(\omega)$, define $\varphi(t) = f(t)K(\omega t)$ and $\Phi(x) = \varphi((c/\omega - a)x/2 + (c/\omega + a)/2)$, where $-1 \leqslant x \leqslant 1$. Further, approximate $\Phi(x)$ by the truncated Chebyshev series $p_N(x)$ (3.3) (or $p_{N+N/\sigma}(x)$ (4.3)). If the error estimate R_N (5.15) (or $R_{N+N/\sigma}$ (5.17)) is less than or equal to $2\varepsilon_1/(c/\omega - a)$, then we accept the approximation using the $p_N(x)$ (or $p_{N+N/\sigma}(x)$).

For the integrals $F(K_q)$ ($q = -1, 0, 1$), define $\varphi(t) = f(t)g(\omega t)$ and further, by using the $\varphi(t)$, define $\Phi_q(x) = \varphi((x_{m+qr+r} - x_{m+qr})x/2 + (x_{m+qr+r} + x_{m+qr})/2)$ for $q = 0, 1$, and $\Phi_q(x) = \varphi((x_m - c/\omega)x/2 + (x_m + c/\omega)/2)$ for $q = -1$. Then, approximate the $\Phi_q(x)$ by the polynomial $p_N(x)$ (or $p_{N+N/\sigma}(x)$) on the interval $[-1, 1]$. If the error estimate R_N (or $R_{N+N/\sigma}$) is less than or equal to $2(\varepsilon_2/3)/(x_{m+qr+r} - x_{m+qr})$ for $q = 0, 1$, and $2(\varepsilon_2/3)/(x_m - c/\omega)$ for $q = -1$, then we accept the corresponding approximation.

7. Numerical examples

Here we compute the following integrals [13, pp. 682, 686 and 712] of $J_0(\omega x)$ and $J_1(\omega x)$, having a parameter a for a variety of ω-values to illustrate the performance of the present automatic quadrature,

$$\int_0^{\infty} J_0(\omega x)\frac{x}{(x^2+a^2)^{1/2}}\,\mathrm{d}x = \frac{\mathrm{e}^{-a\omega}}{\omega}, \qquad a = 1,\ 1/8, \tag{A}$$

$$\int_0^{\infty} J_0(\omega x)\frac{x}{(x^2+a^2)^{3/2}}\,\mathrm{d}x = \frac{\mathrm{e}^{-a\omega}}{a}, \qquad a = 1,\ 1/8, \tag{B}$$

$$\int_0^{\infty} J_0(\omega x)\,\mathrm{e}^{-ax}\,\mathrm{d}x = \frac{1}{(a^2+\omega^2)^{1/2}}, \qquad a = 1,\ 4, \tag{C}$$

$$\int_0^{\infty} J_0(\omega x)x\,\mathrm{e}^{-ax}\,\mathrm{d}x = \frac{a}{(a^2+\omega^2)^{3/2}}, \qquad a = 1,\ 4, \tag{D}$$

$$\int_0^\infty J_1(\omega x)\frac{x^2}{(x^2+a^2)^{3/2}}\,\mathrm{d}x = \mathrm{e}^{-a\omega}, \quad a = 1,\ 1/8, \tag{E}$$

$$\int_0^\infty J_1(\omega x)\frac{x^2}{(x^2+a^2)^{5/2}}\,\mathrm{d}x = \frac{\omega\,\mathrm{e}^{-a\omega}}{3a}, \quad a = 1,\ 1/8, \tag{F}$$

$$\int_0^\infty J_1(\omega x)\,\mathrm{e}^{-ax}\,\mathrm{d}x = \frac{(a^2+\omega^2)^{1/2}-a}{\omega(a^2+\omega^2)^{1/2}}, \quad a = 1,\ 4, \tag{G}$$

$$\int_0^\infty J_1(\omega x)x\,\mathrm{e}^{-ax}\,\mathrm{d}x = \frac{\omega}{(a^2+\omega^2)^{3/2}}, \quad a = 1,\ 4. \tag{H}$$

In tables 1 and 2 we show the numbers of function evaluations, required to achieve the requested accuracies, $\varepsilon_a = 10^{-6}$ and 10^{-12}, and the actual errors, for the integrals (A)–(D) of $J_0(\omega x)$ and the integrals (E)–(H) of $J_1(\omega x)$. The numbers of the

Table 1
Performances of the present method for the integrals $\int_0^\infty J_0(\omega x)f(x)\,\mathrm{d}x$, where: (A) $f(x) = x/(x^2+a^2)^{1/2}$, (B) $f(x) = x/(x^2+a^2)^{3/2}$, (C) $f(x) = \exp(-ax)$ and (D) $f(x) = x\exp(-ax)$. The numbers N of function evaluations required to satisfy the requested tolerances $\varepsilon_a = 10^{-6}$ and 10^{-12} are listed in the fifth and eighth columns. The numbers M of half periods in the interval $[c/\omega, \infty)$ used in the mW-transformation due to Sidi are given in the seventh and last columns.

				$\varepsilon_a = 10^{-6}$			$\varepsilon_a = 10^{-12}$		
$f(x)$	a	ω	Integral	N	error	M	N	error	M
(A)	1	1	$3.678794411714423 \times 10^{-1}$	37	7×10^{-8}	7	87	1×10^{-13}	11
		5	$1.347589399817093 \times 10^{-3}$	39	2×10^{-8}	7	71	5×10^{-15}	12
		9	$1.371220045407551 \times 10^{-5}$	33	1×10^{-8}	7	59	2×10^{-16}	13
	1/8	1	$8.824969025845955 \times 10^{-1}$	83	1×10^{-9}	7	171	4×10^{-14}	11
		5	$1.070522857037980 \times 10^{-1}$	51	1×10^{-8}	6	83	5×10^{-14}	10
		9	$3.607249637314997 \times 10^{-2}$	35	2×10^{-8}	6	83	7×10^{-14}	10
(B)	1	1	$3.678794411714423 \times 10^{-1}$	49	5×10^{-9}	6	91	3×10^{-14}	10
		5	$6.737946999085467 \times 10^{-3}$	37	6×10^{-8}	6	71	7×10^{-14}	11
		9	$1.234098040866796 \times 10^{-4}$	35	7×10^{-9}	7	71	6×10^{-14}	12
	1/8	1	7.059975220676764	121	8×10^{-10}	6	215	3×10^{-14}	10
		5	4.282091428151922	57	6×10^{-10}	6	119	1×10^{-13}	10
		9	2.597219738866798	53	6×10^{-8}	6	103	2×10^{-13}	11
(C)	1	1	0.7071067811865476	37	2×10^{-7}	4	67	9×10^{-16}	7
		5	0.1961161351381840	33	2×10^{-8}	5	51	1×10^{-14}	9
		9	0.1104315260748465	31	2×10^{-9}	6	45	3×10^{-16}	10
	4	1	0.2425356250363330	35	6×10^{-10}	4	59	3×10^{-16}	4
		5	0.1561737618886061	35	8×10^{-8}	4	71	8×10^{-17}	7
		9	0.1015346165133619	33	1×10^{-8}	5	59	7×10^{-17}	8
(D)	1	1	$3.535533905932738 \times 10^{-1}$	39	8×10^{-10}	4	75	6×10^{-17}	7
		5	$7.542928274545540 \times 10^{-3}$	33	4×10^{-9}	6	51	4×10^{-15}	9
		9	$1.346725927742031 \times 10^{-3}$	33	1×10^{-9}	6	45	1×10^{-14}	9
	4	1	$5.706720589090188 \times 10^{-2}$	39	3×10^{-11}	4	59	2×10^{-17}	4
		5	$1.523646457449815 \times 10^{-2}$	33	8×10^{-9}	4	67	2×10^{-16}	7
		9	$4.186994495396367 \times 10^{-3}$	33	7×10^{-9}	5	59	4×10^{-16}	8

Table 2
Performances of the present method for the integrals $\int_0^\infty J_1(\omega x)f(x)\,dx$, where: (E) $f(x) = x^2/(x^2 + a^2)^{3/2}$, (F) $f(x) = x^2/(x^2 + a^2)^{5/2}$, (G) $f(x) = \exp(-ax)$ and (H) $f(x) = x\exp(-ax)$. The nulls in the column of the error indicate that the approximations achieve the accuracy of the roundoff error level of the double precision.

				$\varepsilon_a = 10^{-6}$			$\varepsilon_a = 10^{-12}$		
$f(x)$	a	ω	Integral	N	error	M	N	error	M
(E)	1	1	$3.678794411714423 \times 10^{-1}$	55	3×10^{-8}	6	95	2×10^{-15}	11
		5	$6.737946999085467 \times 10^{-3}$	39	2×10^{-8}	7	71	1×10^{-14}	12
		9	$1.234098040866796 \times 10^{-4}$	37	5×10^{-8}	7	67	2×10^{-14}	13
	1/8	1	0.8824969025845955	89	2×10^{-8}	6	215	1×10^{-15}	11
		5	0.5352614285189903	57	3×10^{-8}	6	99	1×10^{-13}	11
		9	0.3246524673583497	47	3×10^{-8}	6	87	4×10^{-14}	11
(F)	1	1	$1.226264803904808 \times 10^{-1}$	53	8×10^{-9}	5	119	5×10^{-14}	9
		5	$1.122991166514245 \times 10^{-2}$	37	7×10^{-9}	6	79	4×10^{-14}	11
		9	$3.702294122600387 \times 10^{-4}$	39	6×10^{-9}	7	71	3×10^{-14}	12
	1/8	1	2.353325073558921	103	5×10^{-9}	5	183	4×10^{-14}	9
		5	7.136819046919870	95	8×10^{-9}	6	135	4×10^{-15}	11
		9	7.791659216600394	63	3×10^{-8}	6	103	1×10^{-14}	11
(G)	1	1	$2.928932188134525 \times 10^{-1}$	33	2×10^{-8}	4	71	0	7
		5	$1.607767729723632 \times 10^{-1}$	33	1×10^{-8}	5	51	2×10^{-15}	9
		9	$9.884094154723928 \times 10^{-2}$	35	6×10^{-11}	6	45	6×10^{-15}	9
	4	1	$2.985749985466811 \times 10^{-2}$	39	3×10^{-12}	4	51	5×10^{-14}	4
		5	$7.506099048911515 \times 10^{-2}$	35	1×10^{-8}	4	67	8×10^{-17}	7
		9	$6.598461488295027 \times 10^{-2}$	33	2×10^{-9}	5	59	1×10^{-17}	8
(H)	1	1	$3.535533905932737 \times 10^{-1}$	39	4×10^{-9}	5	75	0	8
		5	$3.771464137272770 \times 10^{-2}$	37	2×10^{-10}	6	51	4×10^{-15}	9
		9	$1.212053334967828 \times 10^{-2}$	37	2×10^{-8}	5	45	3×10^{-14}	9
	4	1	$1.426680147272547 \times 10^{-2}$	43	2×10^{-12}	4	59	6×10^{-17}	4
		5	$1.904558071812269 \times 10^{-2}$	37	2×10^{-9}	5	71	4×10^{-18}	8
		9	$9.420737614641827 \times 10^{-3}$	37	9×10^{-10}	5	59	5×10^{-16}	8

integrals $\psi(x_l)$ (2.7) on the half periods of the oscillation in the interval $[5/\omega, \infty)$, used in the mW-transformation, are also listed in the columns headed "M". Tables 1 and 2 experimentally verify the note given in section 3, i.e., the mW-transformation converges so rapidly that $[-\log_{10}\varepsilon_2] + 2$ integrals $\psi(x_i)$ are sufficient to obtain the required accuracy ε_a $(= 20\varepsilon_2/19)$.

It is hard to find out automatic quadratures existing for evaluating the Bessel function integral (2.1), where we set $K(\omega t) = J_\nu(\omega t)$ $(\nu = 0, 1)$, to compare the results computed by using the present scheme. However, an example program in QUADPACK [33, p. 118] manages to compute the integral

$$\int_0^\infty \frac{J_0(x)(1 - e^{-x})}{x\log(1 + \sqrt{2})}\,dx = 1$$

by using the routines DQAGS, DQEXT (ε-algorithm) and ZEROJN (l-positive zero of Bessel function $J_n(x)$). The numbers of function evaluations required in QUADPACK and the present method to obtain the accuracy 10^{-12} are 399 and 71, respectively.

The computation is performed in double precision arithmetic.

Appendix A

For a $(K+1) \times (K+1)$-matrix A defined by (3.12) we show here how to perform the LU decomposition to solve $A\mathbf{x} = \mathbf{b}$ for a given vector $\mathbf{b}^T = (b_0, \dots, b_K)$ with increasing K until satisfactory approximation is obtained [15, 19].

Let B be the same $(K+1) \times (K+1)$-matrix as A, but having elements λ_k $(k = M+2, \dots, K)$ replaced with zeros. Further, let $\mathbf{e}$ be a unit vector,

$$\mathbf{e}^T = (e_0, \dots, e_K) \quad \text{where } e_k = 0 \ (k = 0, \dots, K, \ k \neq M) \text{ and } e_M = 1, \tag{A.1}$$

and let $\mathbf{r}$ be a vector defined by

$$\mathbf{r}^T = (0, \dots, 0, \lambda_{M+2}, \dots, \lambda_K). \tag{A.2}$$

Suppose that we have LU factorization $LU = B$, then we have

$$A = B + \mathbf{e}\,\mathbf{r}^T = LU + \mathbf{e}\,\mathbf{r}^T, \tag{A.3}$$

where L and U are lower and upper triangular matrices, respectively, given by

$$L = \begin{pmatrix} 1 & & & & & & & \\ 0 & 1 & & & & & & \\ \vdots & \ddots & \ddots & & & & & \\ \mu_0 & \cdots & \mu_{M-1} & 1 & & & & \\ 0 & \cdots & 0 & l_M & 1 & & & \\ 0 & \cdots & & 0 & l_{M+1} & 1 & & \\ \vdots & & & & \ddots & \ddots & \ddots & \\ 0 & \cdots & & & \cdots & 0 & l_{K-1} & 1 \end{pmatrix}, \tag{A.4}$$

$$U = \begin{pmatrix} 1 & d_0 & -1 & 0 & \cdots & & & 0 \\ & \ddots & \ddots & \ddots & \ddots & & & \vdots \\ & & 1 & d_{M-1} & -1 & 0 & \cdots & 0 \\ & & & u_M & \zeta & 0 & \cdots & 0 \\ & & & & u_{M+1} & -1 & \ddots & \vdots \\ & & & & & \ddots & \ddots & 0 \\ & & & & & & u_{K-1} & -1 \\ & & & & & & & u_K \end{pmatrix}. \tag{A.5}$$

The values of $\mu_0, \dots, \mu_{M-1}$, $l_M, \dots, l_{K-1}$, $u_M, \dots, u_K$, and ζ in (A.4) and (A.5) will be given later in this appendix.

Using these L and U in (A.3), we have for $A\mathbf{x}=\mathbf{b}$

$$\left(I+L^{-1}\,\mathbf{e}\,\mathbf{r}^T\,U^{-1}\right)U\,\mathbf{x}=L^{-1}\,\mathbf{b}. \tag{A.6}$$

Now let $\mathbf{c}$, $\mathbf{h}$, $\mathbf{v}$ and $\mathbf{y}$ be vectors defined by $\mathbf{c}=L^{-1}\mathbf{b}$, $\mathbf{h}=L^{-1}\mathbf{e}$, $\mathbf{v}^T=\mathbf{r}^TU^{-1}$ and $\mathbf{y}=U\mathbf{x}$, respectively. Then it follows from (A.6) that

$$\mathbf{y}+\mathbf{h}\mathbf{v}^T\mathbf{y}=\mathbf{c},$$

which can be solved for $\mathbf{y}$ as follows,

$$\mathbf{y}=\mathbf{c}-\mathbf{h}\left\{\mathbf{v}^T\mathbf{c}/\left(1+\mathbf{v}^T\mathbf{h}\right)\right\}. \tag{A.7}$$

In summary, the linear system of equations $A\mathbf{x}\equiv(LU+\mathbf{e}\mathbf{r}^T)\mathbf{x}=\mathbf{b}$ can be solved in the following algorithm:

Let A, L and U be $(K+1)\times(K+1)$-matrices defined by (3.12), (A.4) and (A.5), respectively. Further, let $\mathbf{e}$ and $\mathbf{r}$ be vectors defined by (A.1) and (A.2), respectively. Then,

- solve $L\mathbf{h}=\mathbf{e}$ and $U^T\mathbf{v}=\mathbf{r}$ for $\mathbf{h}$ and $\mathbf{v}$, respectively, by the process of forward substitution,
- for a given $(K+1)$-vector $\mathbf{b}$, solve $L\mathbf{c}=\mathbf{b}$ similarly to obtain $\mathbf{c}$,
- evaluate the right hand side of (A.7) for $\mathbf{y}$,
- the solution $\mathbf{x}$ of $A\mathbf{x}=\mathbf{b}$ follows if we solve $U\mathbf{x}=\mathbf{y}$ by back substitution after the appropriate value of K has been determined in such a way as described later.

It remains to show how to compute the non-trivial elements in L and U given by (A.4) and (A.5), respectively. Let $\mathbf{t}^T=(\mu_0,\dots,\mu_{M-1})$, and $\mathbf{q}^T=(\lambda_0,\dots,\lambda_{M-1})$, where μ_k and λ_k are elements in the $(M+1)$th rows of L (A.4) and A (3.12), respectively. Let U_M be an $M\times M$-upper triangular matrix defined by

$$U_M=\begin{pmatrix} 1 & d_0 & -1 & 0 & \dots & 0 \\ 0 & 1 & d_1 & -1 & \ddots & \vdots \\ \vdots & & \ddots & \ddots & \ddots & 0 \\ & & \ddots & 1 & d_{M-3} & -1 \\ \vdots & & & 0 & 1 & d_{M-2} \\ 0 & \dots & & \dots & 0 & 1 \end{pmatrix}.$$

Then we can see from (3.12) and (A.3) that $\mathbf{t}^T U_M=\mathbf{q}^T$, i.e., $U_M^T\mathbf{t}=\mathbf{q}$, which is easily solved for $\mathbf{t}$ by forward substitution. Finally, by using the values of $\mathbf{t}^T=(\mu_0,\dots,\mu_{M-1})$ obtained above, the elements ζ, l_k $(k=M,\dots,K-1)$ and u_k $(k=M,\dots,K)$ in L and U, are computed as follows:

- $\zeta=\lambda_{M+1}+\mu_{M-1}$, $\qquad u_M=\lambda_M+\mu_{M-2}-d_{M-1}\,\mu_{M-1}$,

- with the starting values $l_M = 1/u_M$, $u_{M+1} = d_M - l_M\,\zeta$ do for $k = M+1$ to $K-1$
$$l_k = 1/u_k, \qquad u_{k+1} = d_k + l_k.$$

The value of K, for which the function $H(t)$ given by (3.9) is approximated by

$$H(t) \approx \sum_{k=0}^{K} x_k\, T_k(t),$$

to the accuracy that the computer can achieve, might be determined by checking that x_K ($= y_K/u_K$), obtained before starting the back substitution to solve $U\mathbf{x} = \mathbf{y}$, is smaller in magnitude than the roundoff error level of the computer.

We remark that the above algorithm for the solution of $A\mathbf{x} = \mathbf{b}$ can be executed with $8K - 5M - 1$ multiplications (and divisions).

References

[1] W. L. Anderson, Fast Hankel transform using related and lagged convolutions, ACM Trans. Math. Software 8 (1982) 344–368.

[2] M. Branders and R. Piessens, An extension of Clenshaw–Curtis quadrature, J. Comput. Appl. Math. 1 (1975) 55–65.

[3] J. R. Cash, A note on the numerical solution of linear recurrence relations, Numer. Math. 34 (1980) 371–386.

[4] C. W. Clenshaw and A. R. Curtis, A method for numerical integration on an automatic computer, Numer. Math. 2 (1960) 197–205.

[5] B. Davies, *Integral Transforms and Their Applications* (Springer, New York, 2nd ed., 1985).

[6] P. J. Davis and P. Rabinowitz, *Methods of Numerical Integration* (Academic Press, Orlando, FL, 2nd ed., 1984).

[7] D. Elliott, Truncation errors in two Chebyshev series approximations, Math. Comp. 19 (1965) 234–248.

[8] T. O. Espelid and K. J. Overholt, DQAINF: an algorithm for automatic integration of infinite oscillating tails, Numer. Algorithms 8 (1994) 83–101.

[9] F. N. Fritsch, D. K. Kahaner and J. N. Lyness, Double integration using one-dimensional adaptive quadrature routines: a software interface problem, ACM Trans. Math. Software 7 (1981) 46–75.

[10] W. Gautschi, Computational aspect of three-term recurrence relations, SIAM Rev. 9 (1967) 24–82.

[11] W. Gautschi and R. S. Varga, Error bounds for Gaussian quadrature of analytic functions, SIAM J. Numer. Anal. 20 (1983) 1170–1186.

[12] W. M. Gentleman, Implementing Clenshaw–Curtis quadrature II. Computing the cosine transformation, Comm. ACM 15 (1972) 343–346.

[13] I. S. Gradshteyn and I. M. Ryzhik, *Table of Integrals, Series, and Products*, translated by A. Jeffrey (Academic Press, New York, 1980).

[14] H. L. Gray and T. A. Atchison, Nonlinear transformations related to the evaluation of improper integrals I, SIAM J. Numer. Anal. 4 (1967) 363–371.

[15] T. Hasegawa and T. Torii, Stable algorithm for the minimal solution of second order linear difference equations, J. Inform. Process. 23 (1982) 583–590 (in Japanese).

[16] T. Hasegawa and T. Torii, Indefinite integration of oscillatory functions by the Chebyshev series expansion, J. Comput. Appl. Math. 17 (1987) 21–29.

[17] T. Hasegawa and T. Torii, An automatic quadrature for Cauchy principal value integrals, Math. Comp. 56 (1991) 741–754.

[18] T. Hasegawa and T. Torii, Application of a modified FFT to product type integration, J. Comput. Appl. Math. 38 (1991) 157–168.

[19] T. Hasegawa and T. Torii, An algorithm for nondominant solutions of linear second-order inhomogeneous difference equations, Math. Comp. 64 (1995) 1199–1214.

[20] T. Hasegawa, T. Torii and I. Ninomiya, Generalized Chebyshev interpolation and its application to automatic quadrature, Math. Comp. 41 (1983) 537–553.

[21] T. Hasegawa, T. Torii and H. Sugiura, An algorithm based on the FFT for a generalized Chebyshev interpolation, Math. Comp. 54 (1990) 195–210.

[22] D. K. Kahaner, C. Moler and S. Nash, *Numerical Methods and Software* (Prentice-Hall, Englewood Cliffs, NJ, 1989).

[23] D. Levin and A. Sidi, Two new classes of nonlinear transformations for accelerating the convergence of infinite integrals and series, Appl. Math. Comput. 9 (1981) 204–215.

[24] P. Linz, A method for computing Bessel function integrals, Math. Comp. 26 (1972) 509–513.

[25] I. M. Longman, Tables for rapid and accurate numerical evaluation of certain infinite integrals involving Bessel functions, MTAC 11 (1957) 166–180.

[26] I. M. Longman, A method for the numerical evaluation of finite integrals of oscillatory functions, Math. Comp. 14 (1960) 53–59.

[27] D. W. Lozier, Numerical solution of linear difference equations, Report NBSIR 80-1976, NBS, Washington (1980).

[28] Y. L. Luke, *Mathematical Functions and Their Approximations* (Academic Press, New York, 1975).

[29] J. Lund, Bessel transforms and rational extrapolation, Numer. Math. 47 (1985) 1–14.

[30] J. Lyness and G. Hines, Algorithm 639, to integrate some infinite oscillating tails, ACM Trans. Math. Software 12 (March 1986) 24–25.

[31] R. Piessens and M. Branders, Approximation for Bessel functions and their application in the computation of Hankel transforms, Comput. Math. Appl. 8 (1982) 305–311.

[32] R. Piessens and M. Branders, Modified Clenshaw–Curtis method for the computation of Bessel function integrals, BIT 23 (1983) 370–381.

[33] R. Piessens, E. deDoncker-Kapenga, C. W. Überhuber and D. K. Kahaner, *QUADPACK, A Subroutine Package for Automatic Integration* (Springer, Berlin, 1983).

[34] R. Piessens and A. Haegemans, Algorithm 22, algorithm for the automatic integration of highly oscillatory functions, Computing 13 (1974) 183–193.

[35] A. Sidi, Extrapolation methods for oscillatory infinite integrals, J. Inst. Math. Appl. 26 (1980) 1–20.

[36] A. Sidi, The numerical evaluation of very oscillatory infinite integrals by extrapolation, Math. Comp. 38 (1982) 517–529.

[37] A. Sidi, An algorithm for a special case of a generalization of the Richardson extrapolation process, Numer. Math. 38 (1982) 299–307.

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