APPLICATIONES MATHEMATICAE
25,4 (1999), pp. 301–311

P. KELLER (Wrocław)

INDEFINITE INTEGRATION OF OSCILLATORY FUNCTIONS

Abstract. A simple and fast algorithm is presented for evaluating the indefinite integral of an oscillatory function $\int_x^y f(t)e^{i\omega t}\,dt$, $-1 \le x < y \le 1$, $\omega \ne 0$, where the Chebyshev series expansion of the function f is known. The final solution, expressed as a finite Chebyshev series, is obtained by solving a second-order linear difference equation. Because of the nature of the equation special algorithms have to be used to find a satisfactory approximation to the integral.

1. Introduction. We present an algorithm for computing the indefinite integral of the form

$$I(\omega, x, y) = \int_x^y f(t)e^{i\omega t}\,dt, \qquad -1 \le x < y \le 1,\ \omega \ne 0, \tag{1.1}$$

where f is a smooth function given in terms of Chebyshev polynomials of the first kind,

$$f = \sum_{k=0}^{\infty}{}' a_k T_k, \tag{1.2}$$

(the prime denotes a sum where the first term is halved). In practice, the function f is approximated by a finite series,

$$f \approx f_N = \sum_{k=0}^{N}{}' a_k T_k, \tag{1.3}$$

1991 *Mathematics Subject Classification*: 65D30, 65Q05.
Key words and phrases: indefinite integration, oscillatory function, second-order linear difference equation.

where the polynomial f_N satisfies

$$\max_{-1\le x<y\le 1}\left|\int_x^y [f(t)-f_N(t)]e^{i\omega t}\,dt\right|\le\delta \tag{1.4}$$

for a given error tolerance $\delta>0$ (cf. [2], [4], [6]).

In [3] (see also [4]), it was shown that the integral (1.1) may be expressed as

$$I(\omega,x,y)=\frac{e^{i\omega y}F(y)-e^{i\omega x}F(x)}{i\omega}, \tag{1.5}$$

where the function F is also expanded in the Chebyshev series,

$$F={\sum_{k=0}^{\infty}}' d_kT_k, \tag{1.6}$$

and the coefficients d_k satisfy the difference equation

$$d_{k-1}+\frac{2k}{i\omega}d_k-d_{k+1}=a_{k-1}-a_{k+1},\qquad k=1,2,\dots \tag{1.7}$$

If f is of the form (1.3) then $a_k=0$ for all $k>N$. Consequently, the solution d_k of (1.7) may be obtained recursively as follows:

$$\begin{aligned} &d_{N+2}=d_{N+1}=0,\\ &d_{k-1}=a_{k-1}-a_{k+1}-\frac{2k}{i\omega}d_k+d_{k+1},\qquad k=N+1,N,\dots,1. \end{aligned} \tag{1.8}$$

Unfortunately, if $N>|\omega|$ the above algorithm is unstable [1], so that other techniques have to be used.

In Section 2, we present a fast and accurate method for solving the difference equation (1.7) in the case $N>|\omega|$, when neither forward nor backward recursion can be used. The algorithm is based on the idea of Olver [7], namely a tridiagonal system of linear equations is solved to obtain the desired solution.

In Section 3, we discuss two algorithms developed by Hasegawa and Torii ([3], [5]) for solving the difference equation (1.7). We also present a simplified proof of one of those methods.

Numerical experiments that test the efficiency of the present algorithm and of the two algorithms of Hasegawa and Torii are reported in Section 4.

2. The algorithm. Before we present the algorithm for evaluating the indefinite integral (1.1) we observe [3] that equality (1.5) holds for any solution F of the differential equation

$$\frac{F'}{i\omega}+F=f. \tag{2.1}$$

If we use the known identity $2kd_k = d'_{k-1} - d'_{k+1}$ (see, e.g., [8], p. 124) where d'_k's are the Chebyshev coefficients of F', we easily verify that the differential equation (2.1) is equivalent to the difference equation (1.7). Therefore, any solution of (1.7) satisfying $\sum_{k=0}^{\infty} |d_k| < \infty$ may be used in (1.6) to obtain the expansion (1.5) for the indefinite integral (1.1).

We now assume that f in (1.1) is replaced by the polynomial f_N defined in (1.3), and that $|\omega| < N$. In that case the backward recursion algorithm (1.8) cannot be used to compute the exact solution of (1.7). We will show, however, that the sequence $\{d_k\}_{k=1}^{M}$ $(M > N)$ obtained using our method is an exact solution of the difference equation (1.7) with a_k's replaced by slightly different values $\widehat{a}_k$. Moreover, if we define

$$\widehat{f}_M = {\sum_{k=0}^{M}}' \widehat{a}_k T_k,$$

then $\max_{-1 \le t \le 1} |f_N(t) - \widehat{f}_M(t)| < \varepsilon$, where ε is a given positive number.

As we have noticed, we are looking for any solution of (1.7) satisfying $\sum_{k=0}^{\infty} |d_k| < \infty$. Define $m := \lfloor |\omega| \rfloor$, and set $d_m = 0$. If we find the value d_{m+1} then the values d_k, $k = m-1, m-2, \ldots, 0$, may be computed in a stable way directly from (1.7) using the backward recursion algorithm, because $k < |\omega|$ for all $k < m$. Thus the only problem is to find d_k for $k > m$.

As we defined $d_m = 0$, the equations (1.7) for $k = m+1, m+2, \ldots$ take the form of an infinite system of linear equations

$$\begin{aligned} -i\mu_{m+1}d_{m+1} - d_{m+2} \qquad\qquad\qquad\qquad &= \alpha_{m+1}, \\ d_{m+1} - i\mu_{m+2}d_{m+2} - d_{m+3} \qquad\qquad &= \alpha_{m+2}, \\ d_{m+2} - i\mu_{m+3}d_{m+3} - d_{m+4} &= \alpha_{m+3}, \\ \ddots \qquad\qquad &\;\;\vdots \end{aligned} \tag{2.2}$$

where we have set $\mu_j = 2j/\omega$, $\alpha_j = a_{j-1} - a_{j+1}$ for $j = m+1, m+2, \ldots$ The matrix of this system is diagonally dominant. Therefore, if we set $d_{M+1} = 0$ for some $M > N$, then Gaussian elimination without pivoting may be used to compute d_k for $k = m+1, m+2, \ldots, M$.

We now show how to determine the integer $M > N$ in order to obtain a satisfactory approximation to the indefinite integral (1.1). We need the following lemmas.

LEMMA 2.1. *Let $f_N = \sum_{k=0}^{\prime N} a_k T_k$ and let the sequence $d_0, d_1, \ldots$ be a solution of the difference equation*

$$d_{k-1} - i\mu_k d_k - d_{k+1} = \alpha_k, \tag{2.3}$$

where $\alpha_k = a_{k-1} - a_{k+1}$, and where we assume that $a_k = 0$ for $k > N$. Then for any integer $M \ge N+1$ the sequence $d_0, d_1, \ldots, d_M, 0, 0, \ldots$ is a solution *of the difference equation*

$$d_{k-1} - i\mu_k d_k - d_{k+1} = \begin{cases} \alpha_k & \text{for } 1 \le k \le N, \\ \alpha_M + d_{M+1} & \text{for } k = M, \\ d_M & \text{for } k = M+1, \\ 0 & \text{for } k > N,\ k \ne M, M+1. \end{cases} \tag{2.4}$$

In this case, the function $e^{i\omega t} F_M(t)/i\omega$, *where* $F_M(t) = \sum_{k=0}^{\prime M} d_k T_k(t)$, *is the exact indefinite integral for the function* $\widehat{f}_M(t) e^{i\omega t}$, *where*

$$\widehat{f}_M(t) = f_N(t) + \sum_{k=0}^{M} \xi_k T_k(t) \tag{2.5}$$

with

$$\xi_k = \begin{cases} d_M & \text{for } k = M, M-2, \ldots, \\ d_{M+1} & \text{for } k = M-1, M-3, \ldots \end{cases}$$

Moreover,

$$\max_{-1 \le t \le 1} |f_N(t) - \widehat{f}_M(t)| < \frac{|d_M| + |d_{M+1}|}{2}(M+2). \tag{2.6}$$

P r o o f. Equation (2.4) follows immediately from (2.3). Equality (2.5) is obtained if we recall that $\alpha_k = a_{k-1} - a_{k+1}$. Inequality (2.6) is a simple consequence of the fact that $\max_{-1 \le t \le 1} |T_k(t)| = 1$ for all $k \ge 0$. ∎

LEMMA 2.2. *Let* $\mu_j = 2j/\omega$ *for* $j = m+1, m+2, \ldots,$ $m = \lfloor |\omega| \rfloor$, $\omega \ne 0$ *and*

$$\widetilde{\mu}_{m+1} = \mu_{m+1}, \qquad \widetilde{\mu}_j = \mu_j - 1/\widetilde{\mu}_{j-1} \qquad \text{for } j > m+1. \tag{2.7}$$

Then for all $j \ge m+1$ *we have*

$$|\widetilde{\mu}_j| > 1 + |2(j-m-1)/\omega|. \tag{2.8}$$

P r o o f. Inequality (2.8) holds for $j = m+1$. Suppose (2.8) is true for some $j \ge m+1$. Then

$$\begin{aligned} |\widetilde{\mu}_{j+1}| &\ge |2(j+1)/\omega| - |1/\widetilde{\mu}_j| > |2(j+1)/\omega| - 1 \\ &= 2(j+1-m-1)/|\omega| + 2(m+1)/|\omega| - 1 \\ &> |2(j+1-m-1)/\omega| + 1. \ ∎ \end{aligned}$$

LEMMA 2.3. *Suppose that for* $M \ge N+1$, $m = \lfloor |\omega| \rfloor$, $\omega \ne 0$ *we are given a system of linear equations*

$$\begin{aligned} &-i\mu_{m+1} d^M_{m+1} - d^M_{m+2} = \alpha_{m+1}, \\ &d^M_{j-1} - i\mu_j d^M_j - d^M_{j+1} = \alpha_j && \text{for } j = m+2, \ldots, N+1, \\ &d^M_{j-1} - i\mu_j d^M_j - d^M_{j+1} = 0 && \text{for } j = N+2, \ldots, M, \\ &\qquad d^M_M - i\mu_M d^M_{M+1} = 0, \end{aligned} \tag{2.9}$$

where $\mu_j = 2j/\omega$, $j \geq m+1$, *and* $\alpha_{m+1}, \ldots, \alpha_{N+1}$ *are given real numbers. Then*

$$\tag{2.10} \begin{aligned} d_{M+1}^M &= \frac{\widetilde{\alpha}_{M+1}}{-i\widetilde{\mu}_{M+1}}, \\ d_j^M &= \frac{1}{-i\widetilde{\mu}_j}(\widetilde{\alpha}_j + d_{j+1}^M) \qquad \text{for } j = M, M-1, \ldots, m+1, \end{aligned}$$

where $\widetilde{\mu}_j$ *are defined in* (2.7), *and*

$$\tag{2.11} \widetilde{\alpha}_{j+1} = \begin{cases} \alpha_{m+1} & \text{for } j = m, \\ \alpha_{j+1} + \dfrac{\widetilde{\alpha}_j}{i\widetilde{\mu}_j} & \text{for } j = m+1, \ldots, N, \\ \dfrac{\widetilde{\alpha}_j}{i\widetilde{\mu}_j} & \text{for } j \geq N+1. \end{cases}$$

Moreover, *for any* $\varepsilon > 0$ *there exists an integer* M *such that*

$$\tag{2.12} |d_M^M| + |d_{M+1}^M| < \frac{\varepsilon}{M+2}.$$

Proof. The formulas (2.10) and (2.11) follow immediately if we apply Gaussian elimination without pivoting to the system (2.9). Using (2.11), we have

$$|\widetilde{\alpha}_{M+1}| = |\widetilde{\alpha}_{N+1}| \left|\frac{1}{\widetilde{\mu}_{N+2}}\right| \left|\frac{1}{\widetilde{\mu}_{N+3}}\right| \cdots \left|\frac{1}{\widetilde{\mu}_{M+1}}\right|,$$

which together with (2.8) and

$$\tag{2.13} |d_M^M| + |d_{M+1}^M| < 3|\widetilde{\alpha}_{M+1}|$$

gives (2.12). ■

We may now formulate the algorithm for computing the Chebyshev expansion

$$\tag{2.14} F_M = {\sum_{k=0}^{M}}' d_k T_k$$

for the indefinite integral $e^{i\omega t} F_M(t)/(i\omega)$ in the case $N > |\omega|$.

Algorithm 2.4. Let $\varepsilon > 0$ be the error tolerance for the integral $\int_x^y f_N(t) e^{i\omega t}\, dt$. Set $\mu_j = 2j/\omega$ and $\alpha_j = a_{j-1} - a_{j+1}$ for $j = 1, \ldots, N+1$.

Step 1. Set $m := \lfloor |\omega| \rfloor$ and $d_m := 0$.

Step 2. Perform Gaussian elimination according to the formulas (2.7) and (2.11) until $|\widetilde{\alpha}_{j+1}| \leq \varepsilon/(3(j+2))$ for some $j > N+1$, and set $M := j$.

Step 3. Compute d_j for $j = M, M-1, \ldots, m+1$ using (2.10).

Step 4. Compute d_k for $k = m-1, m-2, \ldots, 0$ using the recurrence relation (1.7) with initial values d_{m+1} and d_m.

Indeed, for any $-1 \le x < y \le 1$, we have from Lemma 2.1 and (2.13),

$$\frac{e^{i\omega y}F_M(y) - e^{i\omega x}F_M(x)}{i\omega} = \int_x^y \widehat{f}_M(t)e^{i\omega t}\,dt,$$

where $\max_{-1\le t\le 1} |f_N(t) - \widehat{f}_M(t)| < \varepsilon/2$. Having in mind that $|e^{i\omega t}| = 1$, we obtain

$$\left|\int_x^y f_N(t)e^{i\omega t}\,dt - \frac{e^{i\omega y}F_M(y) - e^{i\omega x}F_M(x)}{i\omega}\right| < \varepsilon$$

for all $-1 \le x < y \le 1$.

If the function f is of the form (1.2), i.e., there exists no N such that $a_k = 0$ for all $k > N$, then we cannot use the error estimate (2.6) which we have employed in the second step of our algorithm. However, in that case we may use the idea of Hasegawa and Torii [5] to select the value of M so as to achieve the desired accuracy of the computed integral.

If we assume that the exact solution $\{d_k\}$ decreases rapidly as k increases and that the computed values d_i^{M-1} are very close to d_i, then we may expect the following:

$$\max_{-1\le t\le 1}\Big|{\sum_{k=0}^{\infty}}' d_kT_k(t) - {\sum_{k=0}^{M}}' d_k^{M-1}T_k(t)\Big| \approx \max_{-1\le t\le 1}\Big|\sum_{k=M+1}^{\infty} d_kT_k(t)\Big| \le \sum_{k=M+1}^{\infty} |d_k| \le c|d_M^{M-1}|$$

for some positive constant c. As in [5], we set $c = 10$. If the value of M in the second step of our algorithm is selected to satisfy

$$|d_M^{M-1}| \le |\omega|\frac{\varepsilon}{2c} \tag{2.15}$$

for a given $\varepsilon > 0$, then from (1.5) we obtain

$$\left|\int_x^y f(t)e^{i\omega t}\,dt - \frac{e^{i\omega y}F_M(y) - e^{i\omega x}F_M(x)}{i\omega}\right| \le \varepsilon$$

for all $-1 \le x < y \le 1$, where $F_M = {\sum_{k=0}^{M}}' d_k^{M-1}T_k$.

3. Remarks on the algorithms of Hasegawa and Torii. We now discuss two algorithms developed by Hasegawa and Torii [3], [4] for evaluating the indefinite integral (1.1) in the case $N > |\omega|$. We consider more carefully the first method because it gives an exact theoretical solution of the difference equation (1.7) for f of the form (1.3). The method was originally developed for the fixed value $x = -1$, but as we show, this limitation is not necessary.

The first algorithm [3] expresses the coefficients d_k in (1.7) in the form

$$d_k = s_k + \lambda I_k, \qquad k = 0, 1, \ldots,$$

where $I_k = I_k(-i\omega)$, the modified Bessel function of the first kind, is the minimal solution of the homogeneous difference equation (1.7) and s_k, $k = 0, 1, \ldots, N$, is a given particular solution of (1.7). The sequence d_k is truncated at $k = M$, where the integer M is selected to achieve the desired truncation error tolerance.

Define $m := \lfloor |\omega| \rfloor$ and consider the system of linear equations

$$\begin{aligned}
&\frac{2(m+1)}{i\omega}\gamma_{m+1} - \gamma_{m+2} &&= a_m - a_{m+2},\\
&\gamma_{m+1} + \frac{2(m+2)}{i\omega}\gamma_{m+2} - \gamma_{m+3} &&= a_{m+1} - a_{m+3},\\
&\qquad\ddots &&\quad\vdots\\
&\gamma_{N-1} + \frac{2N}{i\omega}\gamma_N - \gamma_{N+1} &&= a_{N-1},\\
&\gamma_N + \left(\frac{2(N+1)}{i\omega} - \frac{I_{N+2}}{I_{N+1}}\right)\gamma_{N+1} &&= a_N,
\end{aligned} \tag{3.1}$$

with unknowns γ_k, $k = m+1, \ldots, N+1$.

LEMMA 3.1. *Suppose the values γ_k for $k = m+1, \ldots, N+1$ form the solution of the system* (3.1), $\gamma_m = 0$, *and γ_k for $k = m-1, \ldots, 0$ are obtained from the relation*

$$\gamma_{k-1} + \frac{2k}{i\omega}\gamma_k - \gamma_{k+1} = a_{k-1} - a_{k+1}. \tag{3.2}$$

Then the values

$$d_k = \begin{cases} \gamma_k & \text{for } k = 0, 1, \ldots, N+1,\\ \dfrac{\gamma_{N+1}}{I_{N+1}} I_k & \text{for } k = N+2, N+3, \ldots, \end{cases} \tag{3.3}$$

satisfy the difference equation (1.7) *for all $k \ge 0$. In particular*, *the values*

$$s_k = d_k - \frac{\gamma_{N+1}}{I_{N+1}} I_k, \qquad k = 0, 1, \ldots, N, \tag{3.4}$$

form the solution of the difference equation (1.7) *with initial condition $s_{N+2} = s_{N+1} = 0$.*

Proof. Substitute (3.3) in (1.7). ∎

The solution s_k cannot be computed effectively using equalities (3.4). The values $\frac{\gamma_{N+1}}{I_{N+1}} I_k$ are very large for small k while very close to d_k for k near N. This would either cause the floating-point overflow or at least very large roundoff errors.

Therefore, we have to use the solution d_k and approximate the infinite sum in (1.6) with

$$F_M = \sum_{k=0}^{M}{}' d_k T_k, \tag{3.5}$$

where M is chosen so as to achieve the desired accuracy of approximation to the integral (1.1). Hasegawa and Torii suggest using the fact that $|I_k|$ decreases as $|I_k| \sim (|\omega|/2)^k/k!$ as k increases. This implies (see [3]) that

$$\sum_{k=M+1}^{\infty} |I_k| < \frac{2}{(M+1)!}\left(\frac{|\omega|}{2}\right)^{M+1},$$

and the truncation error in (3.5) satisfies

$$\left|\frac{\gamma_{N+1}}{I_{N+1}}\right| \sum_{k=M+1}^{\infty} |I_k| < \frac{2}{(M+1)!}\left|\frac{\gamma_{N+1}}{I_{N+1}}\right|\left|\frac{\omega}{2}\right|^{M+1}.$$

From (1.5) we conclude that if $\varepsilon > 0$ is the desired error tolerance for the integral (1.1) for all $-1 \le x < y \le 1$, then M must be selected such that

$$\frac{2}{(M+1)!}\left|\frac{\gamma_{N+1}}{I_{N+1}}\right|\left|\frac{\omega}{2}\right|^{M} < \varepsilon. \tag{3.6}$$

Special care must be taken when computing the left-hand side of (3.6). The problem is that the values $|\omega/2|^{k+1}/(k+1)!$ may increase very rapidly causing the floating-point overflow. Also, if N is large compared to $|\omega|$ then I_{N+1} may be zero for the computer arithmetic. In that case, M is difficult to estimate. For example, if $N = 500$ and $|\omega| \le 80$ then $|I_{N+1}| < 10^{-330}$.

The second algorithm of Hasegawa and Torii [5] assumes that $x = -1$ and expresses the indefinite integral (1.1) as follows:

$$\int_{-1}^{y} f(t)e^{i\omega t}\,dt = \frac{e^{i\omega y}F(y)}{i\omega},$$

where $F(y)$ is of the form (1.6). If we set $y = -1$ then we obtain an additional normalizing condition for the values d_k,

$$F(-1) = \sum_{i=0}^{\infty}{}' (-1)^i d_i = 0. \tag{3.7}$$

This equality together with (1.7) form an infinite system of linear equations

$$
\begin{aligned}
d_0 + \frac{2}{i\omega} d_1 - d_2 \qquad\qquad &= a_0 - a_2,\\
d_1 + \frac{4}{i\omega} d_2 - d_3 \qquad &= a_1 - a_3,\\
\ddots \qquad\qquad &\quad \vdots\\
\sum_{i=0}^{\infty}{}' (-1)^i d_i \qquad &= 0, \qquad (3.8)\\
\ddots \qquad &\quad \vdots\\
d_{k-1} + \frac{2k}{i\omega} d_k - d_{k+1} &= a_{k-1} - a_{k+1},\\
&\quad \vdots
\end{aligned}
$$

where the condition (3.7) is the mth equation of the system for $m = \lfloor |\omega| \rfloor$. The finite approximation d_i^M, $i = 0, 1, \ldots, M$ ($M > m$), to the exact solution of (3.8) is obtained using the LU decomposition algorithm with rank-one updating technique (see [5] for details). The selection of M is based on the computed value $|d_M^M|$ under some additional assumptions on the solution $\{d_k\}$ as described at the end of Section 2 of the present paper (cf. [5]).

Observe that the solution of the system (3.8) is also a solution of the difference equation (1.7). Therefore it may be used to compute the values of (1.1) for any $-1 \le x < y \le 1$ using the equality (1.5).

4. Numerical results. Here we present the numerical results obtained for the following integrals:

(a) $\int_x^y \exp(-t) e^{i\omega t}\, dt$,

(b) $\int_x^y \exp(10 - (10t - 1)^2) e^{i\omega t}\, dt$,

(c) $\int_x^y \left(\sqrt{t} + \frac{1}{\sqrt{8}}\right)^{-1} e^{i\omega t}\, dt$,

(d) $\int_x^y \tan\left(\frac{\pi t}{2.01}\right) e^{i\omega t}\, dt$.

For a given error tolerance $\delta > 0$ the function f is approximated by a polynomial f_N of the form (1.3) that satisfies the inequality (1.4). Then the approximation $\frac{1}{i\omega}(e^{i\omega y}F_M(y) - e^{i\omega x}F_M(x))$, $F_M = \sum_{k=0}^{\prime M} d_k T_k$, to the indefinite integral $\int_x^y f_N(t)e^{i\omega t}\,dt$ is computed using the presented methods for the error tolerance $\varepsilon = 0.1\delta$.

We have implemented two versions of the present algorithm: K1 that uses the error estimate (2.6) and K2 that uses the error estimate (2.15). These methods were compared to the first (HT1) and second (HT2) algorithm of Hasegawa and Torii described in Section 3.

TABLE 1. Comparison of efficiency of the algorithms

				K1	K2	HT1	HT2	K1	K2	HT1	HT2
integral	ω	δ	N	M				time			
(a)	5	10^{-12}	16	23	22	20	26	1.0	1.0	1.1	2.3
	10			24	21	20	35		0.9	1.1	2.2
	15			17	17	17	43		0.9	1.5	2.4
(b)	25	10^{-10}	128	130	118	129	119	1.0	1.0	1.3	2.4
	75			131	120	132	122		1.0	1.4	2.3
	125			137	127	164	129		0.9	1.6	2.2
(c)	50	10^{-10}	256	258	207	257	202	1.0	1.0	1.3	2.3
	150			260	211	269	222		0.9	1.4	2.4
	250			272	252	335	304		0.9	1.6	2.9
(d)	100	10^{-15}	512	515	350	513	351	1.0	0.8	1.3	2.2
	300			517	350	544	382		0.7	1.4	2.2
	500			533	502	676	595		0.9	1.7	3.3

In Table 1 we compare the values of M obtained by each algorithm and the time used to compute d_k, $k = 0, 1, \ldots, M$. For all four test integrals and all test values of ω, x and y the actual errors produced by each algorithm were always less than the desired error tolerance. In general, the smallest errors were produced by the algorithms K1 and HT1.

TABLE 2. Behaviour of the solutions $\{d_k\}$ for the integral (a), $\omega = 15$

k	$\lvert d_k\rvert$ from K1	$\lvert d_k\rvert$ from HT1	$\lvert d_k\rvert$ from HT2
0	$1.26326174169\cdot10^{+0}$	$1.26326174169\cdot10^{+0}$	$1.23420763177\cdot10^{+0}$
6	$4.48777052839\cdot10^{-5}$	$4.48777052839\cdot10^{-5}$	$1.11822980215\cdot10^{+0}$
12	$1.03685038259\cdot10^{-12}$	$1.03685071786\cdot10^{-12}$	$1.28379920236\cdot10^{+0}$
17	$1.61506568016\cdot10^{-17}$	$1.74260978738\cdot10^{-17}$	$3.60887250171\cdot10^{-1}$
18			$1.87829420472\cdot10^{-1}$
19			$8.99033589614\cdot10^{-2}$
25			$2.74466911423\cdot10^{-4}$
31			$1.44859576475\cdot10^{-7}$
37			$2.07574744222\cdot10^{-11}$
43			$1.00805176249\cdot10^{-15}$

Table 2 compares the values $|d_k|$ obtained for the integral (a) with $\omega = 15$. We can observe that the solutions $\{d_k\}$ obtained by K1 and HT1 are nearly identical while the solution computed by HT2 may decrease much slower. This is caused by the additional normalizing condition (3.7) that has to be satisfied by the solution $\{d_k\}$ obtained with the HT2 method.

From the numerical experiments one may conclude that the algorithm K2 is superior to the other three as it is very fast and the corresponding value of M is relatively small. However, we have to remember that the error estimation in K2 and HT2 assumes some additional properties of the solution.

Acknowledgements. I would like to thank Professor S. Lewanowicz for drawing my attention to this problem and for his valuable comments on the paper.

References

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

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

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

[4] —, —, *Application of a modified FFT to product type integration*, ibid. 38 (1991), 157–168.

[5] —, —, *An algorithm for nondominant solutions of linear second-order inhomogeneous difference equations*, Math. Comp. 64 (1995), 1199–1204.

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

[7] F. W. J. Olver, *Numerical solution of second-order linear difference equations*, J. Res. Nat. Bur. Standards 71 (B) (1967), 111–129.

[8] S. Paszkowski, *Numerical Applications of Chebyshev Polynomials and Chebyshev Series*, PWN, Warszawa, 1975 (in Polish).

Paweł Keller
Institute of Computer Science
University of Wrocław
Przesmyckiego 20
51-151 Wrocław, Poland
E-mail: Pawel.Keller@ii.uni.wroc.pl

Received on 30.4.1997