MATHEMATICS OF COMPUTATION
VOLUME 54, NUMBER 189
JANUARY 1990, PAGES 195–210

AN ALGORITHM BASED ON THE FFT FOR A GENERALIZED CHEBYSHEV INTERPOLATION

TAKEMITSU HASEGAWA, TATSUO TORII, AND HIROSHI SUGIURA

ABSTRACT. An algorithm for a generalized Chebyshev interpolation procedure, increasing the number of sample points more moderately than doubling, is presented. The FFT for a real sequence is incorporated into the algorithm to enhance its efficiency. Numerical comparison with other existing algorithms is given.

1. INTRODUCTION

We extend the iterative algorithms due to Gentleman [12, 13] and Branders and Piessens [1] for computing the sequence $\{p_N(t)\}$ of the truncated Chebyshev series

$$p_N(t) = \sum_{k=0}^{N}{}'' a_k^N T_k(t), \qquad -1 \le t \le 1, \tag{1.1}$$

interpolating a given function $f(t)$ on $[-1, 1]$, where $f(t)$ is assumed to be sufficiently smooth. In (1.1), $T_k(t)$ is the Chebyshev polynomial of the first kind, and double prime denotes the summation in which the first and the last term is halved.

It is well known that for a well-behaved function $f(t)$ the truncated Chebyshev series (1.1) enables us to construct efficient automatic quadratures for the so-called product integral [1, 6, 15, 23, 24, 25]

$$Q(f, K) = \int_{-1}^{1} K(t) f(t)\, dt, \tag{1.2}$$

where $K(t)$ is some singular or badly-behaved function. To be specific, the approximation (1.1) yields an integration rule $Q_N(f, K)$ to $Q(f, K)$,

$$Q_N(f, K) = \sum_{k=0}^{N}{}'' a_k^N Q(T_k, K), \tag{1.3}$$

where the modified moment $Q(T_k, K)$ can be computed for various useful singular functions $K(t)$ by means of recurrence relations [19, 20, 21]. If $K(t) = 1$,

Received May 4, 1988; revised October 28, 1988.
1980 *Mathematics Subject Classification* (1985 *Revision*). Primary 65D05, 65D30; Secondary 41A55, 42A15.
Key words and phrases. Chebyshev interpolation, approximate integration, FFT.

$Q_N(f, 1)$ reduces to the Clenshaw-Curtis method [5] (henceforth abbreviated to CC method).

Gentleman [12, 13] proposed the use of the Fast Fourier Transform (FFT) to efficiently compute the Chebyshev coefficients a_k^N in (1.1) and incorporated it into a program of automatic quadrature by the CC method, where in general, by doubling N, the computation can be repeated, using previously computed results until an error criterion is satisfied. In the Gentleman scheme, however, N is chosen as $N = 2 \times 3^n$, $n = 1, 2, \dots$, rather than $N = 2^n$, because the program is simpler. In either case, tripling or doubling [1] of N increases the number of function evaluations quickly [22] and is rather expensive when the number of abscissae required is high.

Bulirsch [4] made use of the sequence $N = 3 \times 2^n$ as well as 2^n, $n = 1, 2, \dots$, in the Romberg integration scheme to enhance the efficiency or economy of automatic quadrature [10]. In this paper we increase N more moderately as follows:

$$(1.4) \qquad N = 3, 4, 5, \dots, 3 \times 2^n, 4 \times 2^n, 5 \times 2^n, \dots, \qquad n = 1, 2, \dots.$$

The aim of this paper is to present an algorithm for recursively generating a sequence of $p_N(t)$ (1.1) by increasing N as in (1.4) and by using the FFT. We choose abscissae $\{t_j^N\}$ for interpolating $f(t)$ so that in particular for the integral $Q(f, 1)$ with $K(t) = 1$, the sequence $\{p_N(t)\}$ yields a sequence of interpolatory quadrature rules $Q_N(f, 1) = \int_{-1}^{1} p_N(t)\,dt = \Sigma w_j^N f(t_j^N)$ having positive weights w_j^N. This is important to guarantee the numerical stability and convergence of quadrature rules [9, p. 189].

To this end, we make a slight modification in the sequence of abscissae $\{\cos 2\pi\alpha_j\}$ proposed in [14] to interpolate $f(t)$ on the open interval $(-1, 1)$. We define a sequence β_j ($j = -1, 0, 1, \dots$) such that $\beta_{-1} = 0$, $\beta_0 = 1/2$ and β_j ($j \geq 1$) satisfies the same recurrence relation as that for α_j given in [14, equation (1.1)] except for the starting value $\beta_1 = 3/4$ instead of $1/4$. Then the abscissae t_j are given by

$$t_j = \cos 2\pi\beta_j, \qquad j = -1, 0, 1, 2, \dots.$$

Note that all the properties concerning the sequence α_j in [14] also hold for the sequence β_j ($j \geq 1$) and $\beta_l - \alpha_l = 2^{-m}$ for an m-bit integer l.

The approximation $p_N(t)$ (1.1) is an interpolating polynomial of degree N satisfying

$$(1.5) \qquad p_N(t_j) = f(t_j), \qquad j = -1, 0, 1, \dots, N-1.$$

Let $N = 2^n$, $n = 2, 3, \dots$; then, as shown in [14], the set of the first $N+1$ abscissae t_j, $j = -1, 0, \dots, N-1$, coincides with $\{\cos \pi j/N\}$ $(0 \leq j \leq N)$ used in the CC method, so that we have [5]

$$(1.6) \qquad a_k^N = \frac{2}{N} \sum_{j=0}^{N}{}'' f(\cos \pi j/N) \cos \pi k j/N, \qquad 0 \leq k \leq N.$$

The interpolating polynomials $p_{5N/4}(t)$ and $p_{3N/2}(t)$ of degrees $5N/4$ and $3N/2$ have the form

$$p_{5N/4}(t) = p_N(t) + \sum_{k=1}^{N/4} b_k^N \{T_{N-k}(t) - T_{N+k}(t)\}, \tag{1.7}$$

$$p_{3N/2}(t) = p_N(t) + \sum_{k=1}^{N/2} B_k^N \{T_{N-k}(t) - T_{N+k}(t)\}. \tag{1.8}$$

In §2 we will prove the following theorem.

Theorem 1.1. *Let* $N = 2^n$, $n = 2, 3, \dots$, *and define* δ_k *and* γ_k *by*

$$\delta_k = \frac{4}{N} \sum_{j=0}^{N/4-1} f(\cos\xi_j) e^{-ik\xi_j}, \qquad 0 \le k \le N/4, \tag{1.9}$$

$$\gamma_k = \frac{2}{N} \sum_{j=0}^{N/2-1} f(\cos\eta_j) e^{-ik\eta_j}, \qquad 0 \le k \le N/2, \tag{1.10}$$

where ξ_j *and* η_j *are defined by*

$$\xi_j = 8\pi(j + \beta_4)/N, \qquad \eta_j = 4\pi(j + \beta_2)/N. \tag{1.11}$$

Then we have

$$\begin{aligned} b_{N/4-k}^N = {} & 2\cos\pi\beta_1 \cos\pi\beta_2 \{2\Re\delta_k - a_k^N - \cos\pi\beta_2(a_{N/4-k}^N + a_{N/4+k}^N)\} \\ & - \cos\pi\beta_2(a_{N/2-k}^N + a_{N/2+k}^N) \\ & - (a_{3N/4-k}^N + a_{3N/4+k}^N)/2, \qquad 0 \le k < N/4, \end{aligned} \tag{1.12}$$

$$\begin{aligned} B_{N/2-k}^N = {} & \cos\pi\beta_1 (2\Re\gamma_k - a_k^N) \\ & - (a_{N/2-k}^N + a_{N/2+k}^N)/2, \qquad 0 \le k < N/2, \end{aligned} \tag{1.13}$$

where, when $k = 0$, *the right-hand sides of* (1.12) *and* (1.13) *are to be halved.*

In §3 the FFT technique for real data [2, 26] is shown to be helpful in successively evaluating the discrete Fourier coefficients $\{\delta_k\}$ of length $N/4$ (1.9), followed by $\{\gamma_k\}$ of length $N/2$ (1.10), and followed by $\{a_k^{2N}\}$ (1.1) of length $2N + 1$. Section 4 discusses error estimates for the interpolation polynomials $p_N(t)$, $p_{5N/4}(t)$ and $p_{3N/2}(t)$, respectively. An application to automatic quadrature, and numerical results, are given in §5.

2. Proof of Theorem 1.1

We first prove (1.12). Setting $t = \cos\theta$ in (1.1) and (1.7), we have from (1.7)

$$p_{5N/4}(\cos\theta) = \sum_{k=0}^{N}{}'' a_k^N \cos k\theta + 2\sin N\theta \sum_{k=1}^{N/4} b_k^N \sin k\theta. \tag{2.1}$$

It can be easily seen from (2.4) in [14] that

$$\prod_{j=N}^{5N/4-1}(t-t_j)=2^{1-N/4}(T_{N/4}(t)-\cos 2\pi\beta_4)=\prod_{j=1}^{N/4}(t-\cos\xi_j)$$

for the integer $N=2^n$ $(n=2,3,\dots)$, where ξ_j is defined by (1.11). Thus, the coefficients b_k^N in (2.1) are determined from the condition

$$p_{5N/4}(\cos\xi_j)=f(\cos\xi_j),\qquad 0\le j<N/4. \tag{2.2}$$

Let the formal sine expansion of the right-hand side of (2.2) be

$$f(\cos\xi_j)=\sum_{k=0}^{N/4}{}''d_k\sin k\xi_j, \tag{2.3}$$

where d_k is given [14, equation (3.15)] by

$$\begin{aligned} d_k&=2\Re\delta_{N/4-k}/\sin 2\pi\beta_4, \qquad 0<k\le N/4,\\ d_0&=0. \end{aligned} \tag{2.4}$$

Then from (2.1), (2.2) and (2.3) it follows that

$$\sum_{k=0}^{N/4}{}''d_k\sin k\xi_j=\sum_{k=0}^{N}{}''a_k^N\cos k\xi_j+2\sin 2\pi\beta_1\sum_{k=1}^{N/4}b_k^N\sin k\xi_j. \tag{2.5}$$

Making use of the relations $\cos(N-k)\xi_j=-\sin k\xi_j$ and

$$\cos k\xi_j=\cos 2\pi\beta_4\sin k\xi_j+\sin(N/4-k)\xi_j/\sin 2\pi\beta_4,$$

$$\cos(N/2\pm k)\xi_j=\cos 2\pi\beta_2\cos k\xi_j\mp\sin 2\pi\beta_2\sin k\xi_j,$$

and using the orthogonality of the sine functions in (2.5) and (2.4), proves (1.12). We can prove (1.13) similarly, in fact more easily, but we omit the details.

3. FFT WITH SYMMETRIES

A thorough presentation of the FFT exploiting various symmetry relations is given in Swarztrauber [26]. Here we reformulate some of the algorithms to make them suitable for our applications. It is convenient to introduce a general offset trapezoidal rule [7, 16] (or generalized midpoint rule) $M_\alpha^N(X)$ for a periodic function $X(t)$ with period 2π. Define $X_{j+\alpha}^N$ with a shift parameter α as follows:

$$X_{j+\alpha}^N=X(2\pi(j+\alpha)/N),\qquad 0\le\alpha\le 1. \tag{3.1}$$

Then $M_\alpha^N(X)$, an approximation to $\int_0^{2\pi}X(t)\,dt$, is given by

$$M_\alpha^N(X)=\frac{2\pi}{N}\sum_{j=0}^{N-1}X_{j+\alpha}^N. \tag{3.2}$$

The special cases $\alpha = 0$ and $1/2$ of (3.2) coincide with the trapezoidal rule and the midpoint rule, respectively.

Periodicity in $X(t)$ gives rise to periodicity in $M_\alpha^N(X)$ with respect to α,

$$M_{\alpha+1}^N(X) = M_\alpha^N(X). \tag{3.3}$$

The general offset trapezoidal rule $M_{2\alpha}^{2N}(X)$ with $2N$ abscissae is easily expressible in terms of $M_\alpha^N(X)$ and $M_{\alpha+1/2}^N(X)$, both having N abscissae,

$$M_{2\alpha}^{2N}(X) = \{M_\alpha^N(X) + M_{\alpha+1/2}^N(X)\}/2. \tag{3.4}$$

This relation will play an important role in the FFT algorithms to be developed.

Definition 3.1. For a periodic function $X(t)$ with period 2π, the generalized discrete Fourier transform $A_{k,\alpha}^N$ with a shift parameter α is defined by

$$\begin{aligned} A_{k,\alpha}^N &= \frac{1}{2\pi} M_\alpha^N(e^{-ikt}X(t)) \\ &= \frac{1}{N}\sum_{j=0}^{N-1} X_{j+\alpha}^N \exp\{-2\pi i k(j+\alpha)/N\}, \qquad k = 0, \pm 1, \pm 2, \cdots, \end{aligned} \tag{3.5}$$

where $X_{j+\alpha}^N$ is defined by (3.1).

Lemma 3.2. *Let $X(t)$ be a periodic complex function with period 2π, that is, $X_{N+j+\alpha}^N = X_{j+\alpha}^N$. Then we have*

$$A_{N+k,\alpha}^N = e^{-2\pi i\alpha} A_{k,\alpha}^N, \tag{3.6}$$

$$A_{k,\alpha+1}^N = A_{k,\alpha}^N. \tag{3.7}$$

Proof. The proof follows trivially from the definition (3.5). □

The relation (3.4) gives a splitting algorithm for $A_{k,2\alpha}^{2N}$ (3.5):

$$A_{k,2\alpha}^{2N} = (A_{k,\alpha}^N + A_{k,\alpha+1/2}^N)/2, \qquad 0 \le k < N, \tag{3.8}$$

$$A_{N+k,2\alpha}^{2N} = (A_{k,\alpha}^N - A_{k,\alpha+1/2}^N)e^{-2\pi i\alpha}/2, \qquad 0 \le k < N. \tag{3.9}$$

If N is a power of 2, $N = 2^n$, the iteration of this splitting algorithm constitutes a modified version of the FFT of Gentleman-Sande type [3, p. 155; 11] for a complex function $X(t)$.

Lemma 3.3. *Let $X(t)$ be a real-valued function with period 2π, that is, $X_{j+\alpha}^N = \overline{X}_{j+\alpha}^N$. Then we have*

$$A_{N-k,\alpha}^N = e^{-2\pi i\alpha}\overline{A}_{k,\alpha}^N, \tag{3.10}$$

where $\overline{X}$ denotes the complex conjugate of X.

Proof. The proof is a trivial consequence of the definition (3.5) for $A_{k,\alpha}^N$. □

Corollary. *Let* $\alpha = 0, 1/2, 1/4$ *or* $3/4$; *then we have* (A) $A^N_{N-k,0} = \overline{A}^N_{k,0}$, (B) $A^N_{N-k,1/2} = -\overline{A}^N_{k,1/2}$, (C) $A^N_{N-k,1/4} = -i\overline{A}^N_{k,1/4}$ *or* (D) $A^N_{N-k,3/4} = i\overline{A}^N_{k,3/4}$, *respectively.*

Equation (3.10) indicates that it suffices to compute half of the N transforms $A^N_{k,\alpha}$. Consequently, the amount of computation and storage can be halved in comparison with that for complex $X(t)$. The splitting algorithm incorporating this saving consists of (3.8) with $0 \le k \le N/2$ and

$$A^{2N}_{N-k,2\alpha} = (\overline{A}^N_{k,\alpha} - \overline{A}^N_{k,\alpha+1/2})e^{-2\pi i\alpha}/2, \qquad 0 \le k < N/2. \tag{3.11}$$

Here we conveniently restrict the fraction α in (3.5) to any element of the sequence $\{\beta_j\}$, say, β_q for arbitrary positive integer q, to formulate the FFT of the real-valued data $X^N_{j+\alpha} = X(2\pi(j+\alpha)/N)$ $(0 \le j < N)$ in a form suitable for our applications. Then it can be seen from (2.3) in [14] that (3.5) may be rewritten as

$$A^N_{k,\beta_q} = \frac{1}{N}\sum_{j=0}^{N-1} X(2\pi\beta_{qN+j})\exp(-2\pi ik\beta_{qN+j}), \qquad 0 \le k < N, \tag{3.12}$$

for which an FFT algorithm is given in the following theorem.

Theorem 3.4 (FFT for a real sequence). *Let* $N = 2^n$, $n = 1, 2, \dots,$ *and* $X(t)$ *be a real-valued function. Calculate* $Y^l(k)$ *for* $l = 1, 2, \dots, n$ *by the following recurrence relations with the starting values* $Y^0(j) = X(2\pi\beta_{qN+j})$, $0 \le j < N$:

$$\begin{aligned} Y^l(k+j2^l) &= Y^{l-1}(k+j2^l) + Y^{l-1}(k+2^{l-1}+j2^l), \\ &\qquad\qquad 0 \le k \le 2^{l-2}, \end{aligned} \tag{3.13}$$

$$\begin{aligned} Y^l(-k+2^{l-1}+j2^l) &= \{\overline{Y}^{l-1}(k+j2^l) - \overline{Y}^{l-1}(k+2^{l-1}+j2^l)\} \\ &\quad \times \exp(-\pi i\beta_{q2^{n-l}+j}), \\ &\qquad\qquad 0 \le k < 2^{l-2},\ 0 \le j < 2^{n-l}. \end{aligned} \tag{3.14}$$

Then we have for A^N_{k,β_q} *in* (3.12)

$$A^N_{k,\beta_q} = Y^n(k)/N, \qquad 0 \le k < N, \tag{3.15}$$

where we make use of the relation (3.10) *to obtain* $Y^n(N-k)$ *for* $0 < k < N/2$ *by*

$$Y^n(N-k) = \overline{Y}^n(k)\exp(-2\pi i\beta_q). \tag{3.16}$$

Remark. If we set $L = 2^l$, $M = N/L$ and $\Gamma = \beta_{qM+jL}$, we see that $Y^l(k+jL)$ corresponds to $A^L_{k,\Gamma}$.

In implementing the above FFT on a computer, $N+1$ real-valued storages $V(k)$, $k = 0, 1, \cdots, N$, are sufficient to carry out the recursions (3.13) and (3.14) in place. Specifically, $\Re Y^l(k+j2^l)$, $0 \le k < 2^{l-1}$, are stored in $V(k+j2^l)$ and $\Im Y^l(k+j2^l)$, $0 \le k < 2^{l-1}$, in $V(2^l - k + j2^l)$, while $Y^l(2^{l-1}+j2^l)\exp(\pi i\beta_{q\cdot 2^{n-l}+j})$, which is real-valued, is stored in $V(2^{l-1}+j2^l)$. In the final step, $l = n$ of (3.13) and (3.14), the contents of $V(k)$ are as follows:

$$
\begin{aligned}
&\Re Y^n(k) = V(k), \qquad 0 \le k < 2^{n-1},\\
&\Re Y^n(2^{n-1}) = V(2^{n-1})\cos\pi\beta_q,\\
&\Re Y^n(2^n - k) = V(k)\cos 2\pi\beta_q - V(2^n-k)\sin 2\pi\beta_q,\\
&\hspace{20em} 0 \le k < 2^{n-1},\\
&\Im Y^n(k) = V(2^n - k), \qquad 0 < k < 2^{n-1},\\
&\Im Y^n(2^{n-1}) = -V(2^{n-1})\sin\pi\beta_q,\\
&\Im Y^n(2^n - k) = -V(k)\sin 2\pi\beta_q - V(2^n-k)\cos 2\pi\beta_q,\\
&\hspace{20em} 0 < k < 2^{n-1}.
\end{aligned}
\tag{3.17}
$$

Note that no unscrambling is necessary for the result of the FFT (3.15) because the input sequence $Y^0(j) = X(2\pi\beta_{qN+j})$ has been generated in the bit-reversed order.

Lemma 3.5. *Let $X(t)$ be a real and even function, that is, $X^N_{j+\alpha} = X^N_{N-j-\alpha}$. Then*

$$A^N_{k,1-\alpha} = \overline{A}^N_{k,\alpha}. \tag{3.18}$$

Proof. From (3.5) and the assumption of the lemma, we have

$$
\begin{aligned}
A^N_{k,1-\alpha} &= \frac{1}{N}\sum_{j=0}^{N-1} X^N_{j+1-\alpha}\exp\{-2\pi i k(j+1-\alpha)/N\}\\
&= \frac{1}{N}\sum_{j=0}^{N-1} X^N_{N-j-\alpha}\exp\{-2\pi i k(N-j-\alpha)/N\}\\
&= \frac{1}{N}\sum_{j=0}^{N-1} X^N_{j+\alpha}\exp\{2\pi i(j+\alpha)/N\} \doteq \overline{A}^N_{k,\alpha}. \qquad \square
\end{aligned}
$$

Corollary. *Both $A^{2N}_{k,0}$ and $A^{2N}_{k,1/2}$ are real-valued and are given by*

$$A^{2N}_{k,0} = \frac{1}{N}\sum_{j=0}^{N}{}'' X^{2N}_j \cos\frac{\pi k j}{N}, \qquad 0 \le k \le N, \tag{3.19}$$

$$A^{2N}_{k,1/2} = \frac{1}{N}\sum_{j=0}^{N-1} X^{2N}_{j+1/2}\cos\frac{\pi k}{N}\left(j+\frac{1}{2}\right), \qquad 0 \le k < N. \tag{3.20}$$

When $f(\cos t)$ is taken as $X(t)$ in (3.19), comparison of (1.6) with (3.19) gives the well-known relation [13] for the Chebyshev coefficients a^N_k:

$$a^N_k = 2A^{2N}_{k,0}, \qquad 0 \le k \le N. \tag{3.21}$$

Equations (3.8) and (3.11) yield a well-known splitting algorithm for a_k^N $(= 2A_{k,0}^{2N})$,

$$A_{k,0}^{2N} = (A_{k,0}^N + A_{k,1/2}^N)/2, \qquad 0 \le k \le N/2, \tag{3.22}$$

$$A_{N-k,0}^{2N} = (A_{k,0}^N - A_{k,1/2}^N)/2, \qquad 0 \le k < N/2, \tag{3.23}$$

where $A_{N/2,1/2}^N = 0$ from (3.10) and (3.18). Swarztrauber [26] referred to a real sequence $X_j^N = X_{N+j}^N$ as being R symmetric, to a real even sequence $X_j^N = X_{N-j}^N$ as being E (even) symmetric, and to a sequence $X_j^N = X_{N-j-1}^N$ as being QE symmetric (quarter-wave even symmetric). If we use this terminology and in (3.20) take into account the fact that $Z_j \equiv X_{j+1/2}^{2N}$ is QE symmetric, $Z_{2N-j-1} = Z_j$, equations (3.22) and (3.23) imply that an E symmetric sequence splits into E and QE symmetric sequences both of half the length.

Further, it can be shown from (3.8) and (3.18) that the transform $A_{k,1/2}^N$, a QE symmetric sequence in (3.22) and (3.23), agrees with the real part of the transform $A_{k,1/4}^{N/2}$ $(= \overline{A}_{k,3/4}^{N/2})$, an R symmetric sequence, that is,

$$A_{k,1/2}^N = \Re A_{k,3/4}^{N/2} = \Re A_{k,1/4}^{N/2}, \qquad 0 \le k < N/2. \tag{3.24}$$

As will be seen, the transform $A_{k,1/4}^{N/2}$ relies on the abscissae $\cos 2\pi\alpha_j$ $(1 \le j \le N/2)$ given in [14], while $A_{k,3/4}^{N/2}$ relies on $\cos 2\pi\beta_j$ $(1 \le j \le N/2)$. It will be shown elsewhere that positive quadrature rules [9, p. 189] of closed type can be constructed based on the abscissae $\cos 2\pi\beta_j$, whereas the abscissae $\cos 2\pi\alpha_j$ $(1 \le j \le N-1)$ yield positive quadrature rules of open type and degree $N-2$, with N given by (1.4). This fact makes the transform $\{A_{k,3/4}^{N/2}\}$ preferable to the alternative $\{A_{k,1/4}^{N/2}\}$.

Lemma 3.6. *Take $f(\cos t)$ as a real periodic function $X(t)$ in* (3.5) *and let δ_k and γ_k be defined by* (1.9) *and* (1.10), *respectively. Then*

$$\delta_k = A_{k,3/16}^{N/4} = A_{k,\beta_4}^{N/4}, \qquad 0 \le k < N/4, \tag{3.25}$$

$$\gamma_k = A_{k,3/8}^{N/2} = A_{k,\beta_2}^{N/2}, \qquad 0 \le k < N/2. \tag{3.26}$$

Proof. The proof follows trivially from the definitions of δ_k and γ_k. □

Figure 1 illustrates how the transform $A_{k,0}^{16}$ with E symmetry, which corresponds to the Chebyshev coefficient a_k^8, successively splits into the transforms of smaller length with their own symmetries, until it reaches the original function values $A_{0,\beta_j}^1 = f(\cos 2\pi\beta_j)$, $j = -1, 0, \cdots, 7$.

Let $N = 2^n$ $(n = 3, 4, \ldots)$ and suppose that the Chebyshev coefficients $\{a_k^N\}$, $0 \le k < N$, of the interpolating polynomial $p_N(t)$ (1.1) are given. We now show the process of successively getting $p_{5N/4}(t)$ (1.7), then $p_{3N/2}(t)$ (1.8),

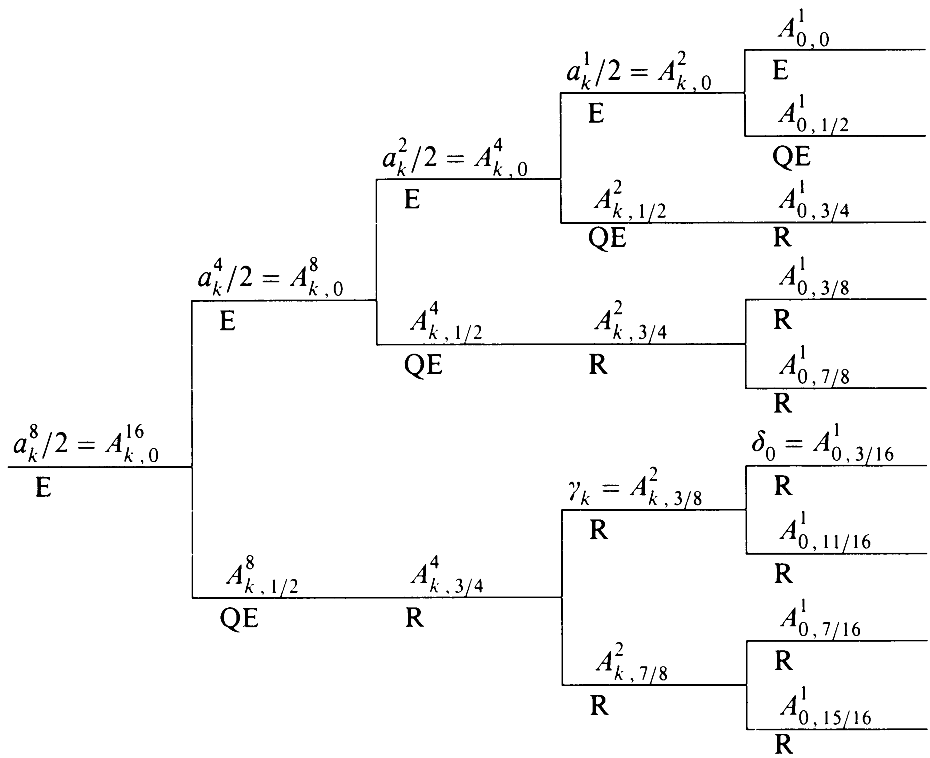

FIGURE 1

Splitting procedure for the Fourier transform $A^{16}_{k,0}$ of a real even function $f(\cos t)$. The A^1_{0,β_j} ($\beta_{-1} = 0$, $\beta_0 = 1/2$, $\beta_1 = 3/4$, $\beta_2 = 3/8, \cdots$) agree with the input data $f(\cos 2\pi\beta_j)$, $j = -1, 0, 1, \cdots, 7$.

then the polynomial $p_{2N}(t)$ of double the order $2N$, until a stopping criterion is satisfied.

Step 1. *Construction of* $p_{5N/4}(t)$. Compute δ_k $(= A^{N/4}_{k,\beta_4})$ defined by (1.9) by using the FFT for a real sequence described in Theorem 3.4 and check a stopping criterion based on an error bound which may be estimated by computing the last two or three coefficients b^N_{N-2}, b^N_{N-1} and b^N_N in (1.12) as described in §4 below. If the stopping criterion is satisfied, exit from this step to stop the process after computing the remaining $\{b^N_k\}$ $(1 \le k \le N-3)$ given by (1.12). Otherwise, proceed to Step 2 without computing $\{b^N_k\}$.

Step 2. *Construction of* $p_{3N/2}(t)$. Compute $A^{N/4}_{k,11/16}$, $0 \le k < N/4$, by using the FFT for a real sequence (Theorem 3.4), and combine δ_k obtained in Step 1 with $A^{N/4}_{k,11/16}$ by the algorithm of (3.8) and (3.11) to calculate γ_k $(= A^{N/2}_{k,\beta_2})$, $0 \le k < N/2$. Similarly as in Step 1, check a stopping criterion. If the criterion is satisfied, compute $\{B^N_k\}$ given by (1.13) and exit from this step to stop the process. Otherwise, go to Step 3.

Step 3. *Construction of* $p_{2N}(t)$. Use the FFT for a real sequence to compute $\{A_{k,7/8}^{N/2}\}$, which is combined with $\{\gamma_k\}$ obtained in Step 2 to yield $\{A_{k,3/4}^{N}\}$ by the algorithm of (3.8) and (3.11). Finally, use (3.22) and (3.23) to compute a_k^{2N} $(=2A_{k,0}^{4N})$ from a_k^N and $A_{k,1/2}^{2N}$ $(=\Re A_{k,3/4}^N)$ obtained previously.

It should be noted that the steps in 1, 2 and 3 for computing δ_k, γ_k and a_k^{2N} can be regarded as parts constituting the algorithm of the FFT of larger length. Consequently, when $p_{5N/4}(t)$ or $p_{3N/2}(t)$ is the polynomial satisfying a stopping criterion, its Chebyshev coefficients can be evaluated with the same amount of computation as required for the FFT, namely $O(N\log_2 N)$.

Lemma 3.7. *Let* $X(t)$ *be a real and odd function with period* 2π, *that is,* $X_{N-j-\alpha}^N = -X_{j+\alpha}^N$, *and* $A_{k,\alpha}^N$ *be defined by* (3.5). *Then*

$$A_{k,1-\alpha}^N = -\overline{A}_{k,\alpha}^N. \tag{3.27}$$

Proof. Equation (3.27) is easily established along the lines of the proof of Lemma 3.5. □

Corollary. *Both* $A_{k,0}^{2N}$ *and* $A_{k,1/2}^{2N}$ *are strictly imaginary and are given by*

$$A_{k,0}^{2N} = -\frac{i}{N}\sum_{j=0}^{N-1} X_j^{2N}\sin\frac{\pi kj}{N}, \qquad 0<k<N, \tag{3.28}$$

$$A_{k,1/2}^{2N} = -\frac{i}{N}\sum_{j=0}^{N-1} X_{j+1/2}^{2N}\sin\frac{\pi k}{N}\left(j+\frac{1}{2}\right), \qquad 0<k<N. \tag{3.29}$$

The splitting algorithm for $A_{k,0}^{2N}$ is the same as (3.22) and (3.23) except for $A_{0,1/2}^N = 0$. Swarztrauber [26] referred to real sequences $X_j^N = -X_{N-j}^N$ and $X_j^N = -X_{N-j-1}^N$ as being O (odd) symmetric and QO (quarter-wave odd) symmetric, respectively. Noting that $Z_j \equiv X_{j+1/2}^{2N}$ in (3.29) is QO symmetric, $Z_{2N-j-1} = -Z_j$, we can see from (3.22) and (3.23) that an O symmetric sequence splits into O and QO symmetric sequences both of half the length. Further, from (3.8) and (3.27), the transform $A_{k,1/2}^{2N}$, a QO symmetric sequence, can be shown to agree in magnitude with the imaginary part of the transform $A_{k,1/4}^N$ $(=-\overline{A}_{k,3/4}^N)$, an R symmetric sequence, as follows:

$$A_{k,1/2}^{2N} = i\Im A_{k,1/4}^N = i\Im A_{k,3/4}^N, \qquad 0<k<N. \tag{3.30}$$

4. Error estimates

We now derive estimates for the differences between $f(t)$ and the approximate polynomials $p_N(t)$, $p_{5N/4}(t)$ and $p_{3N/2}(t)$ defined by (1.1), (1.7) and (1.8), respectively.

Substituting the function $f(t)$, expanded in terms of the Chebyshev polynomials,

$$f(t) = \sum_{k=0}^{\infty}{}' a_k T_k(t), \tag{4.1}$$

into (1.6) establishes the (aliasing) formula [8]

$$a_k^N = a_k + \sum_{m=1}^{\infty}(a_{2mN+k} + a_{2mN-k}), \qquad 0 \le k \le N. \tag{4.2}$$

In (4.1) the prime denotes the summation whose first term is halved. From (1.1), (4.1) and (4.2) it follows [8] that

$$\max_{-1\le t\le 1} |p_N(t) - f(t)| \le 2 \sum_{k=N+1}^{\infty} |a_k|. \tag{4.3}$$

Lemma 4.1. *Let* N *be a power of* 2, $N = 2^n$, *and* δ_k *be defined by* (1.9). *Then*

$$2\Re\delta_k = a_k + \sum_{m=1}^{\infty}(a_{mN/4+k} + a_{mN/4-k}) \cos \pi m \beta_2, \qquad 0 \le k < N/4. \tag{4.4}$$

Proof. Verification of (4.4) consists of inserting (4.1) into (1.9) and using the orthogonality of the cosine function. □

We have from (1.7), (4.1) and (4.2)

$$\begin{aligned} p_{5N/4}(t) - f(t) &= p_N(t) - f(t) + \sum_{k=1}^{N/4} b_k^N \{T_{N-k}(t) - T_{N+k}(t)\} \\ &= \sum_{k=0}^{N/4-1}{}' \sum_{m=1}^{\infty}(a_{2mN+k} + a_{2mN-k})T_k(t) \\ &\quad + \sum_{k=1}^{N/4} \left\{ b_k^N + \sum_{m=1}^{\infty}(a_{2mN+N-k} + a_{2mN-N+k}) \right\} T_{N-k}(t) \\ &\quad + \sum_{m=1}^{\infty} a_{(2m+1)N} T_N(t) - \sum_{k=1}^{N/4}(a_{N+k} + b_k^N)T_{N+k}(t) \\ &\quad - \sum_{k=N+N/4+1}^{\infty} a_k T_k(t). \end{aligned} \tag{4.5}$$

Let $G_m^N(j, k)$ be defined by

$$\begin{aligned} G_m^N(j, k) = a_{2mN+jN/4+k} &+ a_{2mN+jN/4-k} \\ &+ a_{2mN-jN/4+k} + a_{2mN-jN/4-k}. \end{aligned} \tag{4.6}$$

Then $2\Re\delta_k$ in (4.4) can be rewritten as follows:

$$\begin{aligned} 2\Re\delta_k &= a_k + \sum_{j=1}^{3}(a_{jN/4+k} + a_{jN/4-k})\cos j\pi\beta_2 \\ &\quad + \sum_{m=1}^{\infty}(-1)^m \sum_{j=0}^{3}{}' G_m^N(j, k)\cos j\pi\beta_2, \qquad 0 < k \le N/4. \end{aligned} \tag{4.7}$$

We find from (1.12), (4.2) and (4.7) that

$$\begin{aligned} b_{N/4-k}^N = -\sum_{m=1}^{\infty}\Big\{ & G_{2m-1}^N(0, k)\cos\pi\beta_1\cos\pi\beta_2 \\ & + G_{2m-1}^N(1, k)\left(\frac{1}{2} + \cos\pi\beta_1\right) \\ & + G_{2m-1}^N(2, k)\cos\pi\beta_2 + G_{2m-1}^N(3, k)\Big\}, \\ & \qquad 0 \le k < N/4. \end{aligned} \tag{4.8}$$

Substituting (4.8) into the rightmost side of (4.5) we find

$$\begin{aligned} \max_{-1\le t\le 1}|p_{5N/4}(t) - f(t)| &\le 2(2 + |\cos\pi\beta_2|)\sum_{k=5N/4+1}^{\infty}|a_k| \\ &\sim 4.77\sum_{k=5N/4+1}^{\infty}|a_k|, \end{aligned} \tag{4.9}$$

where $\beta_2 = 3/8$. In a similar way we find for $p_{3N/2}(t)$

$$\begin{aligned} \max_{-1\le t\le 1}|p_{3N/2}(t) - f(t)| &\le 4(1 + |\cos\pi\beta_1|)\sum_{k=3N/2+1}^{\infty}|a_k| \\ &= 2(2 + \sqrt{2})\sum_{k=3N/2+1}^{\infty}|a_k|. \end{aligned} \tag{4.10}$$

It can be observed from (4.3), (4.9) and (4.10) that the numerical factors in the error estimates for the approximate polynomials $p_{5N/4}(t)$ and $p_{3N/2}(t)$ are three to four times as large as the one for $p_N(t)$ based on the sample points $\cos\pi j/N$, $j = 0, 1, \ldots, N$, used in the CC method. The coefficients $|a_k|$ in (4.3), (4.9) and (4.10) may be estimated by observing the asymptotic behaviors of $|a_k^N|$, $|B_k^N|$ and $|b_k^N|$ [8, 14, 17].

5. Automatic quadrature and numerical results

This section compares the numerical performance of an automatic quadrature routine based on our results with the performance of GCCINT [1] and CCQUAD [13] for the definite integral $Q(f, 1) = \int_{-1}^{1} f(x)\,dx$.

5.1. *Stopping criterion.* O'Hara and Smith [17], and subsequently Oliver [18], give a practical method for the error estimation in the CC rule. We incorporate the method due to Oliver with minor simplifications and extensions.

For $N = 2^n$ $(n = 2, 3, \ldots)$ Oliver sets

$$K = \max(|a_N^N / 2a_{N-2}^N|, |a_{N-2}^N / a_{N-4}^N|, |a_{N-4}^N / a_{N-6}^N|). \tag{5.1}$$

If $K \leq K_N(16)$, where $K_N(\sigma)$ is tabulated in [18] for $\sigma = 2, 4, 8, 16$ and $N = 2^n$ $(n = 2, 3, \ldots, 7)$, then an error estimate E_N for the approximation $Q_N(f, 1)$ is given by

$$E_N = \frac{16\sigma N}{(N^2-1)(N^2-9)} |a_{N-4}^N| K^3, \tag{5.2}$$

where σ is the smallest of the numbers $2, 4, 8, 16$ such that $K \leq K_N(\sigma)$.

In (5.1) we note that K is an estimate of the rate of convergence of the Chebyshev coefficient a_k in (4.1). On the other hand, the aliasing formula (4.2) indicates that a_{N-i}^N $(0 < i \leq N)$ is a better approximation to a_{N-i} for larger values of i except for a_N^N. Therefore, we replace the second and third terms in the right-hand side of (5.1) by a single term $|a_{N-6}^N / a_{N-8}^N|$ for $N \geq 8$. Further, for simplicity, we neglect the cases $\sigma = 2$ and 8 in (5.2).

If $K_N(16) < K$, the Chebyshev series (1.1) converges slowly. We set $e_N = |Q_N(f, 1) - Q_{N/2}(f, 1)|$ and take $E_N = e_N K^{N/2}$ as an error estimate for $Q_N(f, 1)$ if $K_N(16) < K \leq 0.9$, where the choice of the constant 0.9 has been empirically determined. For $K > 0.9$, we take $E_N = e_N$.

For the error estimates $E_{5N/4}$ and $E_{3N/2}$ of the approximate integrals $Q_{5N/4}(f, 1)$ and $Q_{3N/2}(f, 1)$, respectively, we set

$$E_{5N/4} = \frac{16\sigma N}{(N^2-1)(N^2-9)} |b_{N/4}^N| K, \tag{5.3}$$

$$E_{3N/2} = \frac{16\sigma N}{(N^2-1)(N^2-9)} |B_{N/4}^N| K, \tag{5.4}$$

if $K \leq K_N(\sigma)$. We take $E_{5N/4} = E_N \cdot K^{N/8}$ and $E_{3N/2} = E_N \cdot K^{N/4}$ if $K_N(16) < K \leq 0.9$. For $K > 0.9$, we set $\kappa_N = e_N / e_{N/2}$ and take $E_{5N/4} = e_N \kappa_N^{1/4}$ and $E_{3N/2} = e_N \kappa_N^{1/2}$. Finally, we use $E_{5N/4}$ and $E_{3N/2}$ multiplied by $2 + |\cos \pi \beta_2|$ and $2 + \sqrt{2}$, respectively, to take into account the differences between the approximations $p_{5N/4}(t)$ and $p_{3N/2}(t)$ and $f(t)$ as shown in (4.9) and (4.10).

5.2. *Numerical results.* We give numerical results for the integral $\int_{-1}^{1} f(x)\, dx$, where

(1) $\quad f(x) = (x^2 + a^2)^{-1}, \qquad a = 1, 1/8,$

(2) $\quad f(x) = (1 - a^2)/(1 - 2ax + a^2), \qquad a = 1/2, 7/8,$

(3) $\quad f(x) = (1 + x)^{a/2}, \qquad a = 3, 1.$

Figures 2 and 3 illustrate the number N of functional evaluations required to satisfy the requested tolerance ε_a. Table 1 compares the execution time, the

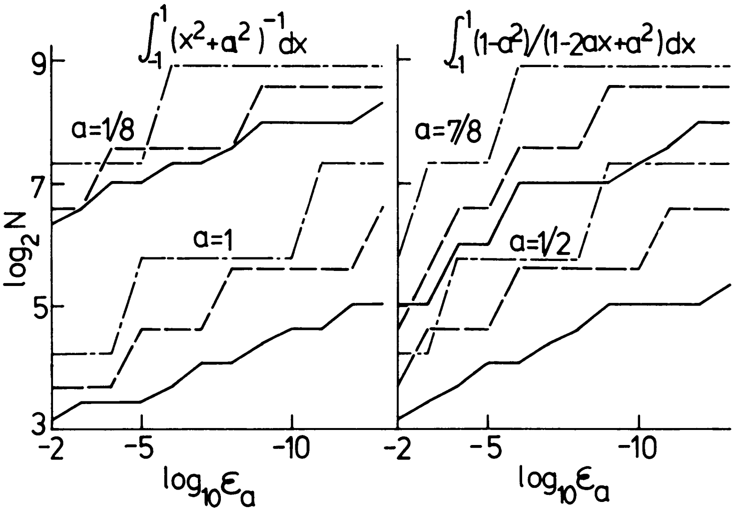

FIGURE 2

Comparison of the number N of functional evaluations required to satisfy the requested tolerance ε_a for $\int_{-1}^{1}(x^2+a^2)^{-1}\,dx$ and $\int_{-1}^{1}(1-a^2)/(1-2ax+a^2)\,dx$. Solid curves, equally and unequally dashed curves represent results based on the present method, the method of Branders and Piessens [1], and the method of Gentleman [13], respectively.

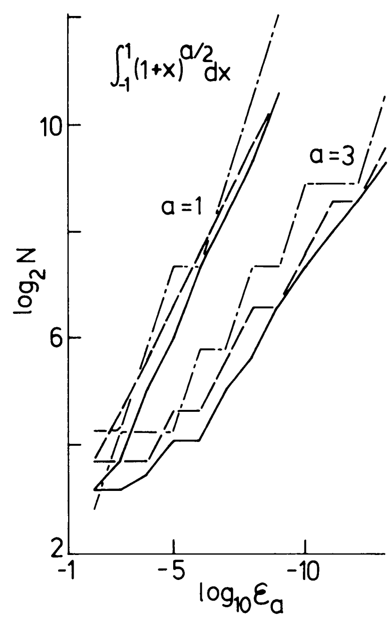

FIGURE 3

Comparison of the number N of functional evaluations for $\int_{-1}^{1}(1+x)^{a/2}\,dx$.

actual error, as well as the N required for the tolerance ε_a for the problem (2) with $a = 3/4$.

Table 1 suggests that all schemes examined perform the computations in execution times proportional to the number of abscissae used. Specifically, CCQUAD and GCCINT take almost the same execution time per sample point, while the present method takes approximately two thirds of that.

TABLE 1

Comparison of the performance of the present method with GCCINT due to Branders and Piessens [1] and CCQUAD due to Gentleman [13] for $\int_{-1}^{1}(1-a^2)/(1-2ax+a^2)\,dx$, $a = 3/4$. The time is given in msec.

	present method			Branders			Gentleman		
ε_a	N	time	error	N	time	error	N	time	error
10^{-2}	17	10	3×10^{-4}	13	8	1×10^{-3}	19	14	1×10^{-4}
10^{-4}	33	18	4×10^{-7}	49	36	9×10^{-10}	163	140	4×10^{-16}
10^{-6}	41	24	9×10^{-8}	97	78	1×10^{-15}	163	140	4×10^{-16}
10^{-8}	65	36	4×10^{-12}	97	78	1×10^{-15}	163	140	4×10^{-16}
10^{-10}	65	36	4×10^{-12}	193	160	4×10^{-16}	487	458	2×10^{-16}
10^{-12}	81	48	8×10^{-14}	193	160	4×10^{-16}	487	458	2×10^{-16}

The positivity of the weights w_j^N of the quadrature rules $Q_N(f, 1)$ depending on the abscissae t_j will be proved elsewhere. The FORTRAN program implementing the present scheme will also appear elsewhere.

The computation was carried out in double-precision arithmetic (about 16 significant digits) on the MELCOM COSMO 700–II computer at Fukui University.

BIBLIOGRAPHY

1. M. Branders and R. Piessens, *An extension of Clenshaw-Curtis quadrature*, J. Comput. Appl. Math. **1** (1975), 55–65.
2. W. L. Briggs, *Further symmetries of in-place FFTs*, SIAM J. Sci. Statist. Comput. **8** (1987), 644–654.
3. E. O. Brigham, *The Fast Fourier Transform and its applications*, Prentice-Hall, Englewood Cliffs, NJ, 1988.
4. R. Bulirsch, *Bemerkung zur Romberg-Integration*, Numer. Math. **6** (1964), 6–16.
5. C. W. Clenshaw and A. R. Curtis, *A method for numerical integration on an automatic computer*, Numer. Math. **2** (1960), 197–205.
6. P. J. Davis and P. Rabinowitz, *Methods of numerical integration*, 2nd ed., Academic Press, Orlando, FL, 1984.
7. E. deDoncker-Kapenga, *Asymptotic expansions and their applications in numerical integration*, Numerical Integration, Recent Developments, Software and Applications (P. Keast and G. Fairweather, eds.), Reidel, Dordrecht, 1987, pp. 141–151.
8. D. Elliott, *Truncation errors in two Chebyshev series approximations*, Math. Comp. **19** (1965), 234–248.
9. H. Engels, *Numerical quadrature and cubature*, Academic Press, London, 1980.

10. G. Fairweather and P. Keast, *An investigation of Romberg quadrature*, ACM Trans. Math. Software **4** (1978), 316–322.

11. W. M. Gentleman and G. Sande, *Fast Fourier transform for fun and profit*, 1966, Fall Joint Computer Conference, AFIPS Conference Proceedings, vol. 29, 1966, pp. 563–578.

12. W. M. Gentleman, *Implementing Clenshaw-Curtis quadrature, I. Methodology and experience*, Comm. ACM **15** (1972), 337–342.

13. ——, *Implementing Clenshaw-Curtis quadrature, II. Computing the cosine transformation*, Comm. ACM **15** (1972), 343–346.

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

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

16. J. N. Lyness, *The calculation of Fourier coefficients by the Möbius inversion of the Poisson summation formula, Part I. Functions whose early derivatives are continuous*, Math. Comp. **24** (1970), 101–135.

17. H. O'Hara and F. J. Smith, *Error estimation in the Clenshaw-Curtis quadrature formula*, Comput. J. **11** (1968), 213–219.

18. J. Oliver, *Doubly-adaptive Clenshaw–Curtis quadrature method*, Comput. J. **15** (1972), 141–147.

19. R. Piessens and M. Branders, *Numerical solution of integral equations of mathematical physics using Chebyshev polynomials*, J. Comput. Phys. **21** (1976), 178–196.

20. R. Piessens, E. deDoncker-Kapenga, C. W. Überhuber and D. K. Kahaner, *QUADPACK, A subroutine package for automatic integration*, Springer-Verlag, Berlin, 1983.

21. R. Piessens, *Modified Clenshaw-Curtis integration and application to numerical computation of integral transforms*, Numerical Integration, Recent Developments, Software and Applications (P. Keast and G. Fairweather, eds.), Reidel, Dordrecht, 1987, pp. 35–51.

22. I. Robinson, *A comparison of numerical integration programs*, J. Comput. Appl. Math. **5** (1979), 207–223.

23. I. H. Sloan and W. E. Smith, *Product-integration with the Clenshaw-Curtis and related points. Convergence properties*, Numer. Math. **30** (1978), 415–428.

24. ——, *Product integration with the Clenshaw-Curtis points: implementation and error estimates*, Numer. Math. **34** (1980), 387–401.

25. ——, *Properties of interpolatory product integration rules*, SIAM J. Numer. Anal. **19** (1982), 427–442.

26. P. N. Swarztrauber, *Symmetric FFTs*, Math. Comp. **47** (1986), 323–346.

Department of Information Science, Faculty of Engineering, Fukui University, Fukui 910 Japan

Department of Information Engineering, Faculty of Engineering, Nagoya University, Nagoya 464 Japan

Department of Information Engineering, Faculty of Engineering, Nagoya University, Nagoya 464 Japan