NORTH-HOLLAND

Fast Polynomial Multiplication and Convolutions Related to the Discrete Cosine Transform

Günter Baszenski
Fachhochschule Dortmund
Fachbereich Nachrichtentechnik
44047 Dortmund, Germany

and

Manfred Tasche
Universität Rostock
Fachbereich Mathematik
18051 Rostock, Germany

Dedicated to Professor L. Berg on the occasion of his 65th birthday

Submitted by Frank Uhlig

ABSTRACT

A classical scheme for multiplying polynomials is given by the Cauchy product formula. Faster methods for computing this product have been developed using circular convolution and fast Fourier transform algorithms. From the numerical point of view the Chebyshev expansion of polynomials is preferred to the monomial form. We develop a direct scheme for multiplication of polynomials in Chebyshev form as well as a fast algorithm using discrete cosine transforms. This approach leads to a new convolution operation and a new type of circulant matrices, both related to the discrete cosine transform. Extensions to bivariate polynomial products are also discussed. © *Elsevier Science Inc., 1997*

1. INTRODUCTION

In order to compute the product of two polynomials of degree n given in monomial form the Cauchy product formula can be used. The number of real

LINEAR ALGEBRA AND ITS APPLICATIONS 252:1–25 (1997)

0024-3795/97/$17.00
SSDI 0024-3795(95)00696-6

multiplications is $\mathcal{O}(n^2)$. Using the fast multiplication of polynomials in monomial form (based on circular convolution and fast Fourier transforms) we can realize this product with only $\mathcal{O}(n \log n)$ real multiplications (see [1, 2]).

From a point of view of numerical analysis polynomials in Chebyshev form are more stable than polynomials in monomial form. Therefore polynomials in Chebyshev form are often used in numerical analysis, especially in spectral and collocation methods (see [3, 4]).

In the following we describe a new method of fast polynomial multiplication in Chebyshev form by using a new convolution related to the type I discrete cosine transform (DCT–I) defined by formulas (3.2)–(3.3). Note that for the integration and differentiation of polynomials in Chebyshev form algorithms of linear complexity are known (see [3, p. 68; 5, p. 134 f]).

Very often the Clenshaw algorithm is used for the evaluation of polynomials in Chebyshev form see [6, 4, 5, p. 134]. This requires $\mathcal{O}(n)$ multiplications for each polynomial value. We propose the simultaneous evaluation of polynomials at the grid $G_N := \{\cos(\mu\pi/N) : \mu = 0, \ldots, N\}$ by means of the DCT–I of length $N + 1$ with N a power of 2. This gives N polynomial values with $\mathcal{O}(N \log N)$ real multiplications.

Note that polynomial interpolation on the grid G_N has good simultaneous approximation properties in the following sense: Let p_N denote the polynomial of degree N which interpolates a given function $f \in C^1[-1, 1]$ at the grid G_N. Then we have (see [7, 8])

$$\|f - p_N\|_\infty \leqslant \left(2 + \frac{2}{\pi}\log N\right)\inf\{\|f - q\|_\infty : q \in \Pi_N\},$$

$$\|f' - p_N'\|_\infty \leqslant (2 + 2\log N)\inf\{\|f' - q\|_\infty : q \in \Pi_{N-1}\}.$$

For an error estimate in a weighted Sobolev norm see [3, p. 295 f].

An important application of polynomial multiplication in the form described here occurs when Galerkin–Petrov methods are used (see [3, 4]).

We remark that our method can be easily adapted to polynomial division which arises in rational approximation (see [4]).

Similar to the known fast multiplication algorithms for polynomials in monomial representation, which lead to circular convolutions and circulant matrices, our approach induces a new type of convolution and circulant matrices related to the DCT–I.

The introduced convolution is a modification of both circular convolution and cross correlation. We give a computation scheme for the convolution based on fast DCT–I algorithms, which we also derive.

The outline of our paper is as follows. In Section 2 we describe a direct multiplication scheme for polynomials in Chebyshev form, which needs $\mathcal{O}(n^2)$ real multiplications for two polynomials of degree n. Based on evaluation and interpolation schemes for polynomials in Chebyshev form on the grid G_N with $N \geqslant 2n + 1$ we derive in Section 3 an identity for the coefficients of the product of two polynomials of degree n involving the DCT–I. If we use fast DCT–I algorithms, then only $\mathcal{O}(n \log n)$ multiplications are necessary to compute the polynomial product. In Section 4 we use this result for the definition of a convolution $\overset{\mathrm{I}}{*}$ related to DCT–I. We discuss the close connection between the convolution $\overset{\mathrm{I}}{*}$, the circular convolution and cross correlation. Note that $\overset{\mathrm{I}}{*}$ has all the usual properties of a product operation. Section 5 is devoted to shifts and circulant matrices induced in a natural way by $\overset{\mathrm{I}}{*}$. We derive properties of these circulant matrices, especially a diagonalization property with respect to DCT–I. In Section 6 we outline a fast algorithm for DCT–I of length $N + 1$, where N is a power of 2. Applying the divide-and-conquer technique the DCT–I of length $N + 1$ can be recursively reduced to DCT–I and type III DCT (see formula (6.2)) of half length. Finally, we extend the previous concept to the multiplication of bivariate polynomials in Chebyshev form.

2. MULTIPLICATION OF POLYNOMIALS IN CHEBYSHEV FORM

We define the Chebyshev polynomials (of first kind) on $I := [-1, 1]$ in the standard way

$$T_k(x) := \cos(k \arccos x) \quad (k \in \mathbb{N}_0, x \in I). \tag{2.1}$$

From the definition we obtain for $x \in I$

$$|T_k(x)| \leqslant 1, \tag{2.2}$$

$$\begin{aligned} T_0(x) &= 1, \qquad T_1(x) = x, \\ T_{k+1}(x) &= 2xT_k(x) - T_{k-1}(x) \qquad (k \in \mathbb{N}). \end{aligned} \tag{2.3}$$

By (2.3) it follows that T_k is a polynomial of exact degree k. Therefore every real polynomial p_n of degree $\leqslant n$ can be represented in the Chebyshev form

$$p_n = \frac{a_0}{2} + \sum_{k=1}^{n} a_k T_k \qquad (a_k \in \mathbb{R}). \tag{2.4}$$

From the numerical point of view the representation (2.4) is more stable than the monomial representation of p_n.

By standard trigonometric identities it is easy to show the relation

$$2T_k T_l = T_{k+l} + T_{|k-l|} \qquad (k, l \in \mathbb{N}_0). \tag{2.5}$$

Note that (2.3) is obtained in the case $l = 1$. Using (2.5), we derive the following multiplication rule for polynomials in Chebyshev form.

PROPOSITION 2.1. *Let $n \in \mathbb{N}$. Let p_n, q_n be given in the Chebyshev form*

$$p_n = \frac{a_0}{2} + \sum_{k=1}^{n} a_k T_k, \qquad q_n = \frac{b_0}{2} + \sum_{k=1}^{n} b_k T_k \tag{2.6}$$

with $a_k, b_k \in \mathbb{R}$ $(k = 0, \ldots, n)$. Then the product $r_{2n} := p_n q_n$ has the Chebyshev form

$$r_{2n} = \frac{c_0}{2} + \sum_{k=1}^{2n} c_k T_k \tag{2.7}$$

with

$$2c_k := \begin{cases} a_0 b_0 + 2\sum_{l=1}^{n} a_l b_l, & k = 0, \\ \sum_{l=0}^{k} a_{k-l} b_l + \sum_{l=1}^{n-k} (a_l b_{l+k} + a_{l+k} b_l), & k = 1, \ldots, n-1, \\ \sum_{l=k-n}^{n} a_{k-l} b_l, & k = n, \ldots, 2n. \end{cases} \tag{2.8}$$

Proof. Using (2.5), we get for $l = 1, \ldots, n$

$$\begin{aligned} 2p_n T_l &= a_0 T_l + \sum_{j=1}^{n} a_j (2T_j T_l) \\ &= a_0 T_l + \sum_{j=1}^{n} a_j T_{j+l} + \sum_{j=1}^{n} a_j T_{|j-l|} \\ &= \sum_{k=l}^{n+l} a_{k-l} T_k + \sum_{k=1}^{l-1} a_{l-k} T_k + a_l + \sum_{k=1}^{n-l} a_{k+l} T_k. \end{aligned}$$

Hence,

$$\begin{aligned} 2p_n b_l T_l = \sum_{k=1}^{n+l} a_{k-l} b_l T_k + \sum_{k=1}^{l-1} a_{l-k} b_l T_k + a_l b_l \\ + \sum_{k=1}^{n-l} a_{k+l} b_l T_k. \end{aligned} \tag{2.9}$$

Moreover we have

$$p_n b_0 = \frac{a_0 b_0}{2} + \sum_{k=1}^{n} a_k b_0 T_k. \tag{2.10}$$

Summing up Eq. (2.9) for $l = 1, \ldots, n$ and (2.10), we obtain

$$\begin{aligned} 2p_n q_n = \left(\frac{a_0 b_0}{2} + \sum_{l=1}^{n} a_l b_l \right) + \sum_{k=1}^{n} b_0 a_k T_k + \sum_{l=1}^{n} \sum_{k=l}^{n+l} a_{k-l} b_l T_k \\ + \sum_{l=2}^{n} \sum_{k=1}^{l-1} a_{l-k} b_l T_k + \sum_{l=1}^{n-1} \sum_{k=1}^{n-l} a_{k+l} b_l T_k. \end{aligned}$$

We change the order of summation in the following way:

$$\sum_{l=1}^{n} \sum_{k=l}^{n+l} = \sum_{k=1}^{n} \sum_{l=1}^{k} + \sum_{k=n+1}^{2n} \sum_{l=k-n}^{n},$$

$$\sum_{l=2}^{n} \sum_{k=1}^{l-1} = \sum_{k=1}^{n-1} \sum_{l=k+1}^{n}, \qquad \sum_{l=1}^{n-1} \sum_{k=1}^{n-l} = \sum_{k=1}^{n-1} \sum_{l=1}^{n-k}.$$

This completes the proof. ■

3. POLYNOMIAL COMPUTATIONS INVOLVING THE DISCRETE COSINE TRANSFORM

In order to evaluate a polynomial p_n given by (2.4) on the grid

$$G_N := \left\{t_\mu^{(N)} := \cos(\mu\pi/N) : \mu = 0,\dots,N\right\} \tag{3.1}$$

with $N \geqslant n + 1$, we have to compute the expressions

$$p_n\left(t_\mu^{(N)}\right) = \frac{a_0}{2} + \sum_{k=1}^{n} a_k \cos\frac{\mu k\pi}{N} \qquad (\mu = 0,\dots,N).$$

This can be written in matrix form as

$$\mathbf{p} = \mathbf{C}_N^{\mathrm{I}}\mathbf{a}, \tag{3.2}$$

where

$$\mathbf{p} := \left(p_n\left(t_\mu^{(N)}\right)\right)_{\mu=0}^{N} \in \mathbb{R}^{N+1},$$

$$\mathbf{a} := (a_k)_{k=0}^{N} \in \mathbb{R}^{N+1} \qquad \left(a_k := 0 \quad (k = n+1,\dots,N)\right), \tag{3.3}$$

$$\mathbf{C}_N^{\mathrm{I}} := \left(\varepsilon_{N,k}\cos\frac{\mu k\pi}{N}\right)_{\mu,k=0}^{N} \in \mathbb{R}^{(N+1)\times(N+1)}$$

with $\varepsilon_{N,0} = \varepsilon_{N,N} := 1/2$ and $\varepsilon_{N,k} := 1$ $(k = 1,\dots,N-1)$. The transformation (3.2) is called *type I discrete cosine transform of length* $N+1$ (DCT–I($N+1$)) (see [2, p. 229]). In order to construct the inverse of C_N^{I}, we need the following

LEMMA 3.1. *For $N \in \mathbb{N}$ and $u \in \mathbb{Z}$ we have*

$$\sum_{k=0}^{N} \varepsilon_{N,k}\cos\frac{ku\pi}{N} = \begin{cases} N, & u \equiv 0 \bmod 2N, \\ 0, & \text{otherwise}. \end{cases} \tag{3.4}$$

Proof. This follows immediately from the known identity [9, p. 71]

$$\sum_{k=0}^{N} \varepsilon_{N,k} \cos(kx) = \sin(Nx) \frac{\cos(x/2)}{2\sin(x/2)} \qquad (x \in \mathbb{R} \setminus 2\pi\mathbb{Z}). \qquad \blacksquare$$

Using the relation

$$2\cos x \cos y = \cos(x-y) + \cos(x+y) \qquad (x, y \in \mathbb{R}), \tag{3.5}$$

we get from (3.4) the following orthogonality relations

$$\frac{2}{N} \varepsilon_{N,l} \sum_{k=0}^{N} \varepsilon_{N,k} \cos\frac{jk\pi}{N} \cos\frac{kl\pi}{N} = \delta_{j,l} \qquad (j, l = 0, \ldots, N),$$

i.e.,

$$\left(\mathbf{C}_N^{\mathrm{I}}\right)^{-1} = \frac{2}{N} \mathbf{C}_N^{\mathrm{I}}. \tag{3.6}$$

Now we can rewrite relation (3.2) as

$$\mathbf{a} = \frac{2}{N} \mathbf{C}_N^{\mathrm{I}} \mathbf{p}. \tag{3.7}$$

This illustrates that the polynomial interpolation problem with knot grid (3.1) can also be solved with DCT–I($N+1$). As we show in Section 6, the DCT–I($N+1$) can be efficiently evaluated by fast algorithms if N is a power of 2.

Applying these ideas, we compute the product of polynomials

$$p_n = \frac{a_0}{2} + \sum_{k=1}^{n} a_k T_k, \qquad q_\nu = \frac{b_0}{2} + \sum_{k=1}^{\nu} b_k T_k \tag{3.8}$$

in the following way. Set $r_{n+\nu} := p_n q_\nu$ with

$$r_{n+\nu} = \frac{c_0}{2} + \sum_{k=1}^{n+\nu} c_k T_k. \tag{3.9}$$

Choose $N \geqslant n + \nu + 1$ (as a power of 2). The polynomial $r_{n+\nu}$ is uniquely determined by

$$r_{n+\nu}\left(t_\mu^{(N)}\right) = p_n\left(t_\mu^{(N)}\right) q_\nu\left(t_\mu^{(N)}\right) \qquad (\mu = 0, \dots, N). \tag{3.10}$$

Introducing the vectors

$$\begin{aligned}
\mathbf{a} &:= (a_0, \dots, a_n, 0, \dots, 0)^{\mathrm{T}} \in \mathbb{R}^{N+1}, \\
\mathbf{b} &:= (b_0, \dots, b_\nu, 0, \dots, 0)^{\mathrm{T}} \in \mathbb{R}^{N+1}, \\
\mathbf{c} &:= (c_0, \dots, c_{n+\nu}, 0, \dots, 0)^{\mathrm{T}} \in \mathbb{R}^{N+1}, \\
\mathbf{p} &:= \left(p_n\left(t_\mu^{(N)}\right)\right)_{\mu=0}^{N} \in \mathbb{R}^{N+1}, \\
\mathbf{q} &:= \left(q_\nu\left(t_\mu^{(N)}\right)\right)_{\mu=0}^{N} \in \mathbb{R}^{N+1}, \\
\mathbf{r} &:= \left(r_{n+\nu}\left(t_\mu^{(N)}\right)\right)_{\mu=0}^{N} \in \mathbb{R}^{N+1},
\end{aligned} \tag{3.11}$$

we have by (3.2)

$$\mathbf{p} = \mathbf{C}_N^{\mathrm{I}} \mathbf{a}, \qquad \mathbf{q} = \mathbf{C}_N^{\mathrm{I}} \mathbf{b}, \qquad \mathbf{r} = \mathbf{C}_N^{\mathrm{I}} \mathbf{c}. \tag{3.12}$$

Furthermore, (3.10) can be written as the componentwise product

$$\mathbf{r} = \mathbf{p} \circ \mathbf{q}. \tag{3.13}$$

Inserting (3.12) into this equation, we have

$$\mathbf{C}_N^{\mathrm{I}} \mathbf{c} = \left(\mathbf{C}_N^{\mathrm{I}} \mathbf{a}\right) \circ \left(\mathbf{C}_N^{\mathrm{I}} \mathbf{b}\right). \tag{3.14}$$

By (3.6), this yields the following result:

PROPOSITION 3.1. *Let $N \geqslant n + \nu + 1$. Let p_n and q_ν be given polynomials of degree n and ν, respectively. Assume that p_n and q_ν are given in Chebyshev form* (3.8). *Then the coefficient vector* $\mathbf{c}$ *of the product $r_{n+\nu} = p_n q_\nu$ can be computed by* DCT–I($N + 1$) *as*

$$\mathbf{c} = \frac{2}{N}\mathbf{C}_N^{\mathrm{I}}\big((\mathbf{C}_N^{\mathrm{I}}\mathbf{a}) \circ (\mathbf{C}_N^{\mathrm{I}}\mathbf{b})\big). \tag{3.15}$$

Comparing with the direct computation in Proposition 2.1, the main advantage of the above transform method is the lower arithmetic complexity. In order to compute a product of two polynomials of degree n by Proposition 2.1 directly, we need $\mathcal{O}(n^2)$ real multiplications. If we use fast DCT–I($N + 1$) algorithms, where $N \geqslant 2n + 1$ is a power of 2, then $\mathcal{O}(N \log N)$ real multiplications are sufficient to compute the product of two polynomials of degree n.

4. CONVOLUTION RELATED TO DCT–I

Briefly we recall the circular convolution property of the discrete Fourier transform of length N (DFT(N)). The Fourier matrix of order N is defined as

$$\mathbf{F}_N := \big(\exp(-2\pi ijk/N)\big)_{j,k=0}^{N-1}.$$

Then the circular convolution property of DFT(N) reads (see [1, 2])

$$\mathbf{F}_N(\mathbf{x} * \mathbf{y}) = (\mathbf{F}_N\mathbf{x}) \circ (\mathbf{F}_N\mathbf{y})$$

or

$$\mathbf{x} * \mathbf{y} = \frac{1}{N}\overline{\mathbf{F}}_N\big((\mathbf{F}_N\mathbf{x}) \circ (\mathbf{F}_N\mathbf{y})\big), \tag{4.1}$$

where $\mathbf{x} := (x_k)_{k=0}^{N-1}, \mathbf{y} := (y_k)_{k=0}^{N-1} \in \mathbb{R}^N$ and

$$\mathbf{x} * \mathbf{y} := \left(\sum_{l=0}^{k} x_l y_{k-l} + \sum_{l=k+1}^{N-1} x_l y_{N+k-l} \right)_{k=0}^{N-1}.$$

Note that (4.1) has the same structure as (3.15). Our aim is to define a new convolution related to DCT–I($N + 1$).

In the following we simplify the right side of (3.15) with arbitrary vectors $\mathbf{a} = (a_j)_{j=0}^{N}, \mathbf{b} = (b_m)_{m=0}^{N} \in \mathbb{R}^{N+1}$. Then the $(l + 1)$st component of (3.15) reads as follows:

$$c_l = \frac{2}{N} \sum_{k=0}^{N} \sum_{j=0}^{N} \sum_{m=0}^{N} \varepsilon_{N,k} \varepsilon_{N,j} \varepsilon_{N,m} a_j b_m \cos\frac{jk\pi}{N} \cos\frac{mk\pi}{N} \cos\frac{kl\pi}{N}.$$

Using

$$\begin{aligned} 4\cos\frac{jk\pi}{N}\cos\frac{mk\pi}{N}\cos\frac{kl\pi}{N} &= \cos\frac{k\pi}{N}(j + m - l) + \cos\frac{k\pi}{N}(j - m + l) \\ &\quad + \cos\frac{k\pi}{N}(-j + m + l) + \cos\frac{k\pi}{N}(j + m + l) \end{aligned}$$

we obtain

$$\begin{aligned} c_l = \frac{1}{2N} \sum_{j=0}^{N} \sum_{m=0}^{N} \varepsilon_{N,j} \varepsilon_{N,m} a_j b_m \sum_{k=0}^{N} \varepsilon_{N,k} \Bigg(&\cos\frac{k\pi}{N}(j + m - l) \\ &+ \cos\frac{k\pi}{N}(j - m + l) + \cos\frac{k\pi}{N}(-j + m + l) \\ &+ \cos\frac{k\pi}{N}(j + m + l) \Bigg). \qquad (4.2) \end{aligned}$$

From (3.4) it follows that for $j, m, l = 0, \ldots, N$ we have

$$\sum_{k=0}^{N} \varepsilon_{N,k} \cos\frac{k\pi}{N}(j+m-l) = \begin{cases} N, & j+m-l=0 \\ & \quad \text{or } j=m=N, l=0, \\ 0, & \text{otherwise,} \end{cases}$$

$$\sum_{k=0}^{N} \varepsilon_{N,k} \cos\frac{k\pi}{N}(j-m+l) = \begin{cases} N, & j-m+l=0 \\ & \quad \text{or } j=l=N, m=0, \\ 0, & \text{otherwise,} \end{cases}$$

$$\sum_{k=0}^{N} \varepsilon_{N,k} \cos\frac{k\pi}{N}(-j+m+l) = \begin{cases} N, & -j+m+l=0 \\ & \quad \text{or } l=m=N, j=0, \\ 0, & \text{otherwise,} \end{cases}$$

$$\sum_{k=0}^{N} \varepsilon_{N,k} \cos\frac{k\pi}{N}(j+m+l) = \begin{cases} N, & j+m+l=2N \\ & \quad \text{or } j=m=l=0, \\ 0, & \text{otherwise.} \end{cases}$$

Inserting the above identities into (4.2) we obtain

$$\begin{aligned} c_l &= \frac{1}{2}\left(\sum_{j=0}^{l} \varepsilon_{N,j}\varepsilon_{N,l-j} a_j b_{l-j} + \sum_{j=0}^{N-l} \varepsilon_{N,j}\varepsilon_{N,j+l} a_j b_{j+l} \right. \\ &\qquad \left. + \sum_{j=l}^{N} \varepsilon_{N,j}\varepsilon_{N,j-l} a_j b_{j-l} + \sum_{j=N-l}^{N} \varepsilon_{N,j}\varepsilon_{N,2N-j-l} a_j b_{2N-j-l} \right) \\ &= \frac{1}{2}\left(\sum_{j=0}^{l-1} a_j b_{l-j} + \sum_{j=1}^{N-l} a_j b_{j+l} + \sum_{j=l}^{N-1} a_j b_{j-l} + \sum_{j=N-l+1}^{N} a_j b_{2N-j-l} \right). \end{aligned} \tag{4.3}$$

Especially, for $l = 0$ and $l = N$ we have

$$c_0 = \sum_{j=0}^{N} \varepsilon_{N,j} a_j b_j, \qquad c_N = \sum_{j=0}^{N} \varepsilon_{N,j} a_j b_{N-j}.$$

Note that c_l $(1 \leqslant l \leqslant N-1)$ possesses the following form

$$c_l = \frac{1}{2} \sum_{(j,m) \in \mathscr{I}_l} a_j b_m$$

with the index set

$$\mathscr{I}_l := \{(j,m) \in \{0,\ldots,N\}^2 : j+m = l \vee j+m = 2N-l \vee m-j = l \vee m-j = -l\}.$$

A graph of $\mathscr{I}_l$ is shown in Fig. 1.

Therefore we define the *convolution related to* DCT–I$(N+1)$ by

$$\mathbf{c} = \mathbf{a} \overset{\mathrm{I}}{*} \mathbf{b} := (c_l)_{l=0}^{N},$$

where c_l is given by (4.3).

By our construction of the convolution related to DCT–I$(N+1)$ we have the following convolution property

$$\mathbf{C}_N^{\mathrm{I}}(\mathbf{a} \overset{\mathrm{I}}{*} \mathbf{b}) = (\mathbf{C}_N^{\mathrm{I}}\mathbf{a}) \circ (\mathbf{C}_N^{\mathrm{I}}\mathbf{b}) \tag{4.4}$$

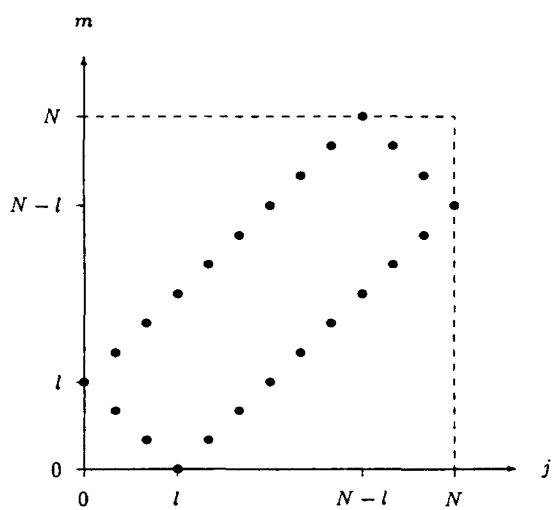

FIG. 1. Index set $\mathscr{I}_l$.

for all $\mathbf{a}, \mathbf{b} \in \mathbb{R}^{N+1}$. Hence, the operation $\overset{\mathrm{I}}{*}$ is commutative, associative, and distributive. It is easily checked that $(2\delta_{0,j})_{j=0}^{N}$ is the unit element of $\overset{\mathrm{I}}{*}$.

If $N \geqslant 2n+1$, then for $\mathbf{a} = (a_0, \ldots, a_n, 0, \ldots, 0)^{\mathrm{T}} \in \mathbb{R}^{N+1}$ and $\mathbf{b} = (b_0, \ldots, b_n, 0, \ldots, 0)^{\mathrm{T}} \in \mathbb{R}^{N+1}$ we obtain

$$\mathbf{c} = \mathbf{a} \overset{\mathrm{I}}{*} \mathbf{b}$$

with $\mathbf{c} = (c_0, \ldots, c_{2n}, 0, \ldots, 0)^{\mathrm{T}} \in \mathbb{R}^{N+1}$, where c_k $(k = 0, \ldots, 2n)$ are given as in Proposition 2.1.

Now we discuss the connection between the convolution $\overset{\mathrm{I}}{*}$ and the circular convolution $*$. Given two vectors $\mathbf{a} = (a_j)_{j=0}^{N}$ and $\mathbf{b} = (b_j)_{j=0}^{N}$, we form the even extensions $\mathbf{x} = (x_j)_{j=0}^{2N-1}$ and $\mathbf{y} = (y_j)_{j=0}^{2N-1}$ of $\mathbf{a}$ and $\mathbf{b}$ by setting $x_0 := a_0$, $y_0 := b_0$, $x_j := x_{2N-j} := a_j$, $y_j := y_{2N-j} := b_j$ $(j = 1, \ldots, N)$. Let $\mathbf{z} := \mathbf{x} * \mathbf{y} = (z_k)_{k=0}^{2N-1}$. Then

$$z_k = \sum_{j=0}^{k} x_j y_{k-j} + \sum_{j=k+1}^{2N-1} x_j y_{2N-j+k} \qquad (k = 0, \ldots, 2N-1).$$

Note that $z_k = z_{2N-k}$ $(k = 1, \ldots, N-1)$. For $k = 0, \ldots, N$, we have

$$\begin{aligned} z_k &= \sum_{j=0}^{k} a_j b_{k-j} + \sum_{j=k+1}^{N} a_j b_{j-k} \\ &\quad + \sum_{j=N+1}^{N+k} a_{2N-j} b_{j-k} + \sum_{j=N+k+1}^{2N-1} a_{2N-j} b_{2N+k-j} \\ &= \sum_{j=0}^{k} a_j b_{k-j} + \sum_{j=k+1}^{N} a_j b_{j-k} \\ &\quad + \sum_{j=N-k}^{N-1} a_j b_{2N-j-k} + \sum_{j=1}^{N-k-1} a_j b_{j+k} \end{aligned}$$

i.e., by (4.3) we have $z_k = 2c_k$ $(k = 0, \ldots, N)$, where

$$\mathbf{c} = (c_k)_{k=0}^{N} = \mathbf{a} \overset{\mathrm{I}}{*} \mathbf{b}.$$

A similar computation shows that the cross correlation $\mathbf{w} = (w_k)_{k=0}^{2N-1} = \operatorname{cor}(\mathbf{x}, \mathbf{y})$ defined by

$$w_k = \sum_{j=0}^{2N-k-1} x_j y_{k+j} + \sum_{j=2N-k}^{2N-1} x_j y_{k+j-2N}$$

equals the circular convolution

$$\operatorname{cor}(\mathbf{x}, \mathbf{y}) = \mathbf{x} * \mathbf{y}.$$

Therefore the operation $\overset{\mathrm{I}}{*}$ is a suitable modification for both the cross correlation and the circular convolution.

5. SHIFTS AND CIRCULANT MATRICES RELATED TO DCT–I

The circular convolution is strongly connected to circular shifts and circulant matrices (see [10]). In the following we introduce suitable modifications of these concepts for the DCT–I setting.

We define $\mathbf{P}_m \in \mathbb{R}^{(N+1)\times(N+1)}$ by

$$\mathbf{P}_0 := \left(\delta_{j,k}\right)_{j,k=0}^{N}, \qquad \mathbf{P}_N := \left(\delta_{j,N-k}\right)_{j,k=0}^{N},$$

and for $m = 1, \ldots, N-1$ by

$$\mathbf{P}_m := \begin{array}{r} \\ \\ \\ m \rightarrow \\ \\ \\ \\ \\ \\ \end{array} \left[\begin{array}{ccccccc} & & \overset{\displaystyle m \atop \downarrow}{1} & & & & \\ & \frac{1}{2} & & \frac{1}{2} & & & \\ \cdot^{\cdot^{\cdot}} & & & & \ddots & & \\ \frac{1}{2} & & & & & \ddots & \\ & \ddots & & & & & \frac{1}{2} \\ & & \ddots & & & \cdot^{\cdot^{\cdot}} & \\ & & & \frac{1}{2} & & \frac{1}{2} & \\ & & & & \underset{\displaystyle \uparrow \atop N-m}{1} & & \end{array} \right] \begin{array}{l} \\ \\ \\ \\ \leftarrow N-m \\ \\ \\ \\ \end{array} \tag{5.1}$$

Blank spaces in the matrix $\mathbf{P}_m$ denote zero entries. These matrices are called *shift matrices related to* DCT–I($N + 1$).

The shift matrices $\mathbf{P}_m$ ($m = 1, \ldots, N-1$) operate as follows on a vector $\mathbf{b} = (b_j)_{j=0}^N \in \mathbb{R}^{N+1}$:

(i) if $1 \leqslant m \leqslant N/2$:

$$(\mathbf{P}_m \mathbf{b})_l = \begin{cases} \frac{1}{2}(b_{m-l} + b_{m+l}), & l = 0, \ldots, m, \\ \frac{1}{2}(b_{l-m} + b_{m+l}), & l = m+1, \ldots, N-m, \\ \frac{1}{2}(b_{2N-m-l} + b_{l-m}), & l = N-m+1, \ldots, N, \end{cases} \quad (5.2)$$

(ii) if $N/2 + 1 \leqslant m \leqslant N-1$:

$$(\mathbf{P}_m \mathbf{b})_l = \begin{cases} \frac{1}{2}(b_{m-l} + b_{m+l}), & l = 0, \ldots, N-m, \\ \frac{1}{2}(b_{m-l} + b_{2N-m-l}), & l = N-m+1, \ldots, m, \\ \frac{1}{2}(b_{l-m} + b_{2N-m-l}), & l = m+1, \ldots, N. \end{cases} \quad (5.3)$$

It can be shown by direct computation with (4.3) that for $m = 0, \ldots, N$:

$$\mathbf{e}_m \overset{\mathrm{I}}{*} \mathbf{b} = \varepsilon_{N,m} \mathbf{P}_m \mathbf{b} \quad (5.4)$$

where $\mathbf{e}_m = (\delta_{m,j})_{j=0}^N \in \mathbb{R}^{N+1}$ are the unit vectors.

Applying the definition (5.1) of $\mathbf{P}_m$ we obtain the following formula for the shifted unit vectors:

$$\mathbf{P}_m \mathbf{e}_j = \tfrac{1}{2}(\mathbf{e}_{|m-j|} + \mathbf{e}_{\min\{m+j, 2N-m-j\}}) \qquad (j, m = 0, \ldots, N).$$

Now we define *circulant matrices related to* DCT–I($N + 1$) by setting for $\mathbf{a} = (a_j)_{j=0}^N \in \mathbb{R}^{N+1}$:

$$\operatorname{circ}_{\mathrm{I}} \mathbf{a} := \sum_{j=0}^{N} \varepsilon_{N,j} a_j \mathbf{P}_j. \quad (5.5)$$

In the case $N = 4$, a circulant matrix related to DCT–I(5) reads as follows:

$$\operatorname{circ}_{\mathrm{I}} \mathbf{a} = \frac{1}{2} \begin{bmatrix} a_0 & 2a_1 & 2a_2 & 2a_3 & a_4 \\ a_1 & a_0 + a_2 & a_1 + a_3 & a_2 + a_4 & a_3 \\ a_2 & a_1 + a_3 & a_0 + a_4 & a_1 + a_3 & a_2 \\ a_3 & a_2 + a_4 & a_1 + a_3 & a_0 + a_2 & a_1 \\ a_4 & 2a_3 & 2a_2 & 2a_1 & a_0 \end{bmatrix}$$

with arbitrary vector $\mathbf{a} = (a_j)_{j=0}^4 \in \mathbb{R}^5$.

Using (5.4), the relation to the convolution $\overset{\mathrm{I}}{*}$ is as follows:

$$\mathbf{a} \overset{\mathrm{I}}{*} \mathbf{b} = \sum_{j=0}^{N} a_j \mathbf{e}_j \overset{\mathrm{I}}{*} \mathbf{b}$$

$$= \sum_{j=0}^{N} \varepsilon_{N,j} a_j \mathbf{P}_j \mathbf{b} = (\mathrm{circ}_{\mathrm{I}} \mathbf{a})\mathbf{b} \tag{5.6}$$

for $\mathbf{a} = (a_j)_{j=0}^{N}, \mathbf{b} \in \mathbb{R}^{N+1}$.

In the following, we describe the main properties of circulant matrices related to DCT−I($N + 1$):

PROPOSITION 5.1. *For* $\mathbf{a}, \mathbf{b} \in \mathbb{R}^{N+1}$, $\alpha \in \mathbb{R}$ *we have*:

(i) $\mathbf{a} \overset{\mathrm{I}}{*} \mathbf{b} = (\mathrm{circ}_I \mathbf{a})\mathbf{b} = (\mathrm{circ}_I \mathbf{b})\mathbf{a}$.
(ii) $\mathrm{circ}_I(\mathbf{a} + \mathbf{b}) = \mathrm{circ}_I \mathbf{a} + \mathrm{circ}_I \mathbf{b}$.
(iii) $\mathrm{circ}_I(\alpha \mathbf{a}) = \alpha\, \mathrm{circ}_I \mathbf{a}$.
(iv) $N\, \mathrm{circ}_I \mathbf{a} = 2\mathbf{C}_N^{\mathrm{I}}(\mathrm{diag}\, \hat{\mathbf{a}})\mathbf{C}_N^{\mathrm{I}}$ *with* $\hat{\mathbf{a}} = (\hat{a}_k)_{k=0}^{N} = \mathbf{C}_N^{\mathrm{I}} \mathbf{a}$.
(v) $\mathrm{circ}_I \mathbf{a}$ *is invertible iff* $\hat{a}_k \neq 0$ *for* $k = 0, \ldots, N$. *Then we have* $N(\mathrm{circ}_I \mathbf{a})^{-1} = 2\mathbf{C}_N^{\mathrm{I}}(\mathrm{diag}\, \hat{\mathbf{a}})^{-1}\mathbf{C}_N^{\mathrm{I}}$.
(vi) $(\mathrm{circ}_I \mathbf{a})(\mathrm{circ}_I \mathbf{b}) = \mathrm{circ}_I(\mathbf{a} \overset{\mathrm{I}}{*} \mathbf{b})$.

Proof. From (5.6) and from the commutativity of $\overset{\mathrm{I}}{*}$ it follows that (i) holds. Using (i) and the bilinearity of $\overset{\mathrm{I}}{*}$ we obtain (ii) and (iii). Next we show the diagonalization property (iv). By (4.4), we have

$$\mathbf{C}_N^{\mathrm{I}}(\mathbf{a} \overset{\mathrm{I}}{*} \mathbf{b}) = (\mathrm{diag}\, \hat{\mathbf{a}})\mathbf{C}_N^{\mathrm{I}} \mathbf{b}.$$

Using (i) we get

$$\mathbf{C}_N^{\mathrm{I}}(\mathrm{circ}_{\mathrm{I}} \mathbf{a})\mathbf{b} = (\mathrm{diag}\, \hat{\mathbf{a}})\mathbf{C}_N^{\mathrm{I}} \mathbf{b}$$

for all $\mathbf{b} \in \mathbb{R}^{N+1}$. Therefore

$$\mathbf{C}_N^{\mathrm{I}}(\mathrm{circ}_{\mathrm{I}} \mathbf{a}) = (\mathrm{diag}\, \hat{\mathbf{a}})\mathbf{C}_N^{\mathrm{I}} \tag{5.7}$$

which is by (3.6) equivalent to (iv). By (iv), $\mathrm{circ}_{\mathrm{I}} \mathbf{a}$ is invertible iff $\mathrm{diag}\, \hat{\mathbf{a}}$ is invertible. This is the case iff all diagonal entries $\hat{a}_k$ are nonzero. By the associativity of $\stackrel{\mathrm{I}}{*}$ we have

$$(\mathbf{a} \stackrel{\mathrm{I}}{*} \mathbf{b}) \stackrel{\mathrm{I}}{*} \mathbf{c} = \mathbf{a} \stackrel{\mathrm{I}}{*} (\mathbf{b} \stackrel{\mathrm{I}}{*} \mathbf{c})$$

for arbitrary $\mathbf{a}, \mathbf{b}, \mathbf{c} \in \mathbb{R}^{N+1}$. Applying (i) we obtain

$$\big(\operatorname{circ}_{\mathrm{I}}(\mathbf{a} \stackrel{\mathrm{I}}{*} \mathbf{b})\big)\mathbf{c} = (\operatorname{circ}_{\mathrm{I}} \mathbf{a})(\operatorname{circ}_{\mathrm{I}} \mathbf{b})\mathbf{c}$$

for all $\mathbf{c} \in \mathbb{R}^{N+1}$; i.e., (vi) is fulfilled. This completes the proof. ∎

Note that all circulant matrices related to DCT–I(N + 1) form a commutative algebra in $\mathbb{R}^{(N+1)\times(N+1)}$.

The shift matrices related to DCT–I(N + 1) have the following properties:

PROPOSITION 5.2. *For $m, n = 0, \ldots, N$ we have*:

(i) $\varepsilon_{N,m} \mathbf{P}_m = \operatorname{circ}_I \mathbf{e}_m$.

(ii) $N \mathbf{P}_m = 2\mathbf{C}_N^{\mathrm{I}}\ (\operatorname{diag}(\cos(mk\pi/N))_{k=0}^{N})\mathbf{C}_N^{\mathrm{I}}$.

(iii) $2\mathbf{P}_m \mathbf{P}_n = \mathbf{P}_{|m-n|} + \mathbf{P}_{\min\{m+n,\, 2N-m-n\}}$.

Proof. The first property follows immediately from (5.5). In order to show (ii), note first that by (3.3) we have

$$\hat{\mathbf{e}}_m := \mathbf{C}_N^{\mathrm{I}} \mathbf{e}_m = \varepsilon_{N,m}\left(\cos\frac{mk\pi}{N}\right)_{k=0}^{N}. \tag{5.8}$$

From property (i) and (5.7), (5.8) we conclude that

$$\begin{aligned}\varepsilon_{N,m}\mathbf{C}_N^{\mathrm{I}}\mathbf{P}_m &= \mathbf{C}_N^{\mathrm{I}}(\operatorname{circ}_{\mathrm{I}}\mathbf{e}_m)\\ &= (\operatorname{diag}\hat{\mathbf{e}}_m)\mathbf{C}_N^{\mathrm{I}}\\ &= \varepsilon_{N,m}\left(\operatorname{diag}\left(\cos\frac{mk\pi}{N}\right)_{k=0}^{N}\right)\mathbf{C}_N^{\mathrm{I}},\end{aligned}$$

i.e., (ii) is valid by (3.6).

From (ii) we obtain

$$2\mathbf{P}_m\mathbf{P}_n = \frac{2}{N}\mathbf{C}_N^{\mathrm{I}}\left(\operatorname{diag}\left(2\cos\frac{km\pi}{N}\cos\frac{nk\pi}{N}\right)_{k=0}^{N}\right)\mathbf{C}_N^{\mathrm{I}}.$$

By (3.5) we have

$$2\cos\frac{mk\pi}{N}\cos\frac{nk\pi}{N} = \cos\frac{l_1 k\pi}{N} + \cos\frac{l_2 k\pi}{N}$$

with $l_1 := |m - n|$ and $l_2 := \min\{m + n, 2N - m - n\}$. Note that $l_1, l_2 \in \{0, \dots, N\}$. This completes the proof of (iii). ■

6. FAST DCT-ALGORITHMS

In order to describe fast DCT algorithms we assume that N is a power of 2, i.e., $N = 2^{t+1}$ with $t \in \mathbb{N} \setminus \{1\}$.

We recall that the DCT–I$(N + 1)$ is a mapping of $\mathbb{R}^{N+1}$ into itself defined by DCT–I$(N + 1)(\mathbf{a}) = \mathbf{C}_N^1 \mathbf{a} = \hat{\mathbf{a}}$ with

$$\hat{a}_j := \sum_{k=0}^{N} \varepsilon_{N,k} a_k \cos\frac{jk\pi}{N} \qquad (j = 0, \dots, N), \tag{6.1}$$

where $\mathbf{a} = (a_k)_{k=0}^N, \hat{\mathbf{a}} = (\hat{a}_j)_{j=0}^N$.

The *type* III DCT *of length* $N_1 = 2^t$ (DCT–III(N_1)) is a mapping of $\mathbb{R}^{N_1}$ into itself defined by

$$\tilde{g}_j := \sum_{k=0}^{N_1 - 1} \varepsilon_{N_1,k} g_k \cos\frac{(2j+1)k\pi}{N} \qquad (j = 0, \dots, N_1 - 1). \tag{6.2}$$

In the following we describe a procedure for computing the DCT–I$(N + 1)$ in real arithmetic. (see [M]). For a computation with discrete Fourier transforms in complex arithmetic see [2, p. 238].

Applying the divide-and-conquer technique we obtain:

PROPOSITION 6.1. *Let $N = 2^{t+1}, N_1 = 2^t$ with $t \in \mathbb{N} \setminus \{1\}$ be given. Then the* DCT–I$(N + 1)$ *of a data vector $\mathbf{a} = (a_k)_{k=0}^N \in \mathbb{R}^{N+1}$ can be computed recursively by the transforms* DCT–I$(N_1 + 1)$ *and* DCT–III(N_1) *as follows*:

For $j = 0, \dots, N_1$ we have

$$\hat{a}_{2j} = \hat{f}_j$$

and for $j = 0, \ldots, N_1 - 1$ *we have*

$$\hat{a}_{2j+1} = \tilde{g}_j$$

with

$$\left(\hat{f}_j\right)_{j=0}^{N_1} = \hat{\mathbf{f}} = \text{DCT-I}(N_1 + 1)(\mathbf{f})$$

$$\left(\tilde{g}_j\right)_{j=0}^{N_1-1} = \tilde{\mathbf{g}} = \text{DCT-III}(N_1)(\mathbf{g})$$

where

$$\mathbf{f} = (a_l + a_{N-l})_{l=0}^{N_1},$$

$$\mathbf{g} = (a_l - a_{N-l})_{l=0}^{N_1-1}.$$

Proof. For $j = 0, \ldots, N_1$ we obtain from (6.1) and

$$\cos\frac{j(N-l)\pi}{N_1} = \cos\frac{jl\pi}{N_1} \qquad (l = 0, \ldots, N_1 - 1)$$

that

$$\begin{aligned} \hat{a}_{2j} &= \sum_{l=0}^{N_1-1} \varepsilon_{N_1,l} a_l \cos\frac{jl\pi}{N_1} + (-1)^j a_{N_1} \\ &\quad + \sum_{l=0}^{N_1-1} \varepsilon_{N_1,l} a_{N-l} \cos\frac{j(N-l)\pi}{N_1} \\ &= \sum_{l=0}^{N_1} \varepsilon_{N_1,l}(a_l + a_{N-l}) \cos\frac{jl\pi}{N_1}. \end{aligned}$$

For $j = 0, \ldots, N_1 - 1$ we conclude similarly using (6.1) and

$$\cos\frac{(2j+1)(N-l)\pi}{N} = -\cos\frac{(2j+1)l\pi}{N}$$

that

$$\hat{a}_{2j+1} = \sum_{l=0}^{N_1-1} \varepsilon_{N_1,l}(a_l - a_{N-l})\cos\frac{(2j+1)l\pi}{N}.$$

This completes the proof. ∎

In the following we describe a fast algorithm for DCT−III(N_1) in real arithmetic (see [11]). A complex algorithm of (6.2) based on the discrete Fourier transform can be found in [2, p. 142 f].

First we set $h_l := \varepsilon_{N_1,l}g_l$ ($l = 0,\ldots,N_1-1$). Substituting

$$\tilde{h}_j := \begin{cases} \tilde{g}_{2j} & j = 0,\ldots,N_2-1, \\ \tilde{g}_{N-2j-1} & j = N_2,\ldots,N_1-1 \end{cases}$$

with $N_2 = N_1/2$, we obtain the *simplified* DCT−III(N_1):

$$\tilde{h}_j = \sum_{l=0}^{N_1-1} h_l\cos\frac{(4j+1)l\pi}{N} \qquad (j = 0,\ldots,N_1-1). \tag{6.3}$$

Using the divide-and-conquer technique, we form

$$\tilde{h}_j = \sum_{n=0}^{N_2-1} h_n\cos\frac{(4j+1)n\pi}{N} + \sum_{n=0}^{N_2-1} h_{N_2+n}\cos\frac{(4j+1)(N_2+n)\pi}{N}.$$

Hence we obtain by (3.5)

$$\tilde{h}_j = \left(h_0 + \sum_{n=1}^{N_2-1}(h_n - h_{N_1-n})\cos\frac{(4j+1)n\pi}{N}\right)$$

$$+ 2\cos\frac{(4j+1)\pi}{4}\left(\frac{1}{2}h_{N_2} + \sum_{n=1}^{N_2-1} h_{N_2+n}\cos\frac{(4j+1)(N_2+n)\pi}{N}\right).$$

Now the factors $\cos(4j+1)(\pi/4)$ $(j = 0, \ldots, N_1 - 1)$ possess only two different values, namely

$$\cos\frac{(4j+1)\pi}{4} = (-1)^{j_0}\frac{\sqrt{2}}{2}$$

for $j = \sum_{\tau=0}^{t-1} j_\tau 2^\tau$ $(j_\tau \in \{0,1\}, \tau = 0, \ldots, t-1)$.

This leads to the first step of our radix 2 algorithm:

ALGORITHM 6.1. *Simplified* DCT–III(N_1)

Input: $N_1 = 2^t$ $(t \in \mathbb{N} \setminus \{1\})$, $h_j \in \mathbb{R}$ $(j = 0, \ldots, N_1 - 1)$.
Step 1: *For* $i_0 = 0, 1$ *compute*

$$h_0^{I_1} := h_0 + (-1)^{i_0} h_{N_1} \frac{\sqrt{2}}{2},$$

$$h_n^{I_1} := h_n - h_{N_1 - n} + (-1)^{i_0} h_{N_2 + n} \sqrt{2} \quad (n = 1, \ldots, N_2 - 1)$$

with $I_1 := (i_0)$.

After step 1 we obtain

$$\tilde{h}_j = \sum_{n=0}^{N_2 - 1} h_n^{I_1} \cos\frac{(4j+1)n\pi}{N} \tag{6.4}$$

for all $j \in \{0, \ldots, N_1 - 1\}$ with $j_0 = i_0$. So we have divided our original sum (6.3) of length N_1 into two sums of half length N_2. If we apply the same technique repeatedly then we get for $\tau = 2, \ldots, t$:

Step τ: *For all* $i_0, \ldots, i_{\tau-1} = 0, 1$ *compute*

$$h_0^{I_\tau} := h_0^{I_{\tau-1}} + (-1)^{i_{\tau-1}} h_{N_\tau}^{I_{\tau-1}} \gamma(I_{\tau-1}),$$

$$h_n^{I_\tau} := h_n^{I_{\tau-1}} - h_{N_\tau - n}^{I_{\tau-1}} + 2(-1)^{i_{\tau-1}} h_{N_{\tau+1}+n}^{I_{\tau-1}} \gamma(I_{\tau-1})$$

$$(n = 1, \ldots, N_{\tau+1} - 1)$$

with

$$I_\tau := (i_{\tau-1}, \ldots, i_0), \qquad (I_{\tau-1})_2 := \sum_{k=0}^{\tau-2} i_k 2^k,$$

$$\gamma(I_{\tau-1}) := \cos(4(I_{\tau-1})_2 + 1)\pi/2^{\tau+1}.$$

After step t we have

Output: $\tilde{h}_j := h_0^{I_t}$ *with* $j = (I_t)_2$.

7. MULTIPLICATION OF BIVARIATE POLYNOMIALS IN CHEBYSHEV FORM

Now we extend the previous concept to bivariate polynomials. Let $p_{n,n}$ be a given bivariate polynomial in Chebyshev form

$$p_{n,n}(x, y) = \sum_{k=0}^{n} \sum_{l=0}^{n} \varepsilon_{N,k} \varepsilon_{N,l} a_{k,l} T_k(x) T_l(y) \tag{7.1}$$

where $N \geqslant n + 1$.

The values of $p_{n,n}$ at the bivariate grid $G_N \times G_N$, where G_N is defined by (3.1) can be written in matrix form as

$$\mathbf{P} = \mathbf{C}_N^1 \mathbf{A} \left(\mathbf{C}_N^1\right)^{\mathrm{T}} \tag{7.2}$$

with

$$\mathbf{P} := \left(p_{n,n}\left(t_\mu^{(N)}, t_\nu^{(N)}\right)\right)_{\mu,\nu=0}^{N}, \qquad \mathbf{A} := (a_{k,l})_{k,l=0}^{N} \tag{7.3}$$

where $a_{k,l} := 0$ for $k > n$ or $l > n$. The matrix $\mathbf{C}_N^1$ is given in (3.3). Note that the right-hand side of (7.2) is the *bivariate* DCT–I *of size* $(N + 1) \times (N + 1)$ (DCT–I($(N + 1) \times (N + 1)$)) of $\mathbf{A}$.

For given values of $p_{n,n}$ at $G_N \times G_N$ the Chebyshev coefficients $a_{k,l}$ can be computed by

$$\mathbf{A} = \frac{4}{N^2} \mathbf{C}_N^1 \mathbf{P} \left(\mathbf{C}_N^1\right)^{\mathrm{T}}. \tag{7.4}$$

This method of solving Eq. (7.2) is also used in the tensor product method of interpolation [12, p. 341 f]. Formula (7.4) can be efficiently computed by performing row and column transforms of **P** with a fast DCT–I$(N+1)$ algorithm if N is chosen as a power of 2. Another solution based on the Clenshaw algorithm has been proposed in [13]. Our method based on the DCT – I$(N+1)$ gives a lower operation count, though.

In the following we develop a method to compute the polynomial product

$$r_{2n,2n}(x, y) = p_{n,n}(x, y) q_{n,n}(x, y)$$

where $p_{n,n}$ is given in Chebyshev form (7.1) and where

$$q_{n,n}(x, y) = \sum_{k=0}^{n} \sum_{l=0}^{n} \varepsilon_{N,k} \varepsilon_{N,l} b_{k,l} T_k(x) T_l(y) \qquad (b_{k,l} \in \mathbb{R}) \quad (7.5)$$

$$r_{2n,2n}(x, y) = \sum_{k=0}^{2n} \sum_{l=0}^{2n} \varepsilon_{N,k} \varepsilon_{N,l} c_{k,l} T_k(x) T_l(y) \qquad (c_{k,l} \in \mathbb{R}) \quad (7.6)$$

with $N \geqslant 2n+1$. The polynomial $r_{2n,2n}$ is uniquely determined by

$$r_{2n,2n}\left(t_\mu^{(N)}, t_\nu^{(N)}\right) = p_{n,n}\left(t_\mu^{(N)}, t_\nu^{(N)}\right) q_{n,n}\left(t_\mu^{(N)}, t_\nu^{(N)}\right) \qquad (\mu, \nu = 0, \dots, N). \quad (7.7)$$

We want to write these relations as a matrix equation. Let **A**, **P** be as in (7.3) and

$$\mathbf{B} := (b_{k,l})_{k,l=0}^{N}, \qquad \mathbf{C} := (c_{k,l})_{k,l=0}^{N}$$

where we set $b_{k,l} := 0$ if $k > n$ or $l > n$ and $c_{k,l} := 0$ if $k > 2n$ or $l > 2n$. Let

$$\mathbf{Q} := \left(q_{n,n}\left(t_\mu^{(N)}, t_\nu^{(N)}\right)\right)_{\mu,\nu=0}^{N}, \qquad \mathbf{R} := \left(r_{2n,2n}\left(t_\mu^{(N)}, t_\nu^{(N)}\right)\right)_{\mu,\nu=0}^{N}.$$

Then we have

$$\mathbf{P} = \mathbf{C}_N^1 \mathbf{A} (\mathbf{C}_N^1)^{\mathrm{T}}, \qquad \mathbf{Q} = \mathbf{C}_N^1 \mathbf{B} (\mathbf{C}_N^1)^{\mathrm{T}}, \qquad \mathbf{R} = \mathbf{C}_N^1 \mathbf{C} (\mathbf{C}_N^1)^{\mathrm{T}} \quad (7.8)$$

and by (7.7)

$$\mathbf{R} = \mathbf{P} \circ \mathbf{Q} \quad (7.9)$$

where $\circ$ denotes the elementwise product of matrices. Inserting (7.8) into (7.9) we get

$$\mathbf{C}_N^1 \mathbf{C} (\mathbf{C}_N^1)^{\mathrm{T}} = \left(\mathbf{C}_N^1 \mathbf{A} (\mathbf{C}_N^1)^{\mathrm{T}}\right) \circ \left(\mathbf{C}_N^1 \mathbf{B} (\mathbf{C}_N^1)^{\mathrm{T}}\right).$$

By (3.6) we obtain the following result:

PROPOSITION 7.1. *Let $N \geqslant 2n + 1$. Let $p_{n,n}$ and $q_{n,n}$ be given bivariate polynomials in Chebyshev form* (7.1) *and* (7.5), *respectively. Then the product $r_{2n,2n} = p_{n,n} q_{n,n}$ possesses the Chebyshev form* (7.6) *with coefficients $c_{k,l}$ given by*

$$\mathbf{C} = \frac{4}{N^2} \mathbf{C}_N^1 \left(\left(\mathbf{C}_N^1 \mathbf{A} (\mathbf{C}_N^1)^{\mathrm{T}}\right) \circ \left(\mathbf{C}_N^1 \mathbf{B} (\mathbf{C}_N^1)^{\mathrm{T}}\right) \right) (\mathbf{C}_N^1)^{\mathrm{T}}. \quad (7.10)$$

Note that (7.10) can be used as a definition for a bivariate convolution $\mathbf{C} = \mathbf{A} \overset{\mathrm{I}}{*} \mathbf{B}$ related to the bivariate DCT–I($(N + 1) \times (N + 1)$).

8. CONCLUSION

Polynomials of high degree restricted on $[-1, 1]$ can be numerically stable evaluated if they are represented in Chebyshev form. In this setting the problem naturally arises to form products of such polynomials. Standard trigonometric identities yield formulas for the product of two Chebyshev polynomials and thus of Chebyshev expansions. We have given an alternative approach using fast algorithms of discrete cosine transforms which improves significantly the operation count for computing polynomial products. The approach yields a variant of the convolution operation and of circulant matrices, whose properties we have investigated. As we have shown the approach can be easily extended to cover bivariate polynomials.

The authors are grateful to the referees for helpful comments and suggestions. The research of the second author was supported by the Deutsche Forschungsgemeinschaft.

REFERENCES

1 H. J. Nussbaumer, *Fast Fourier Transform and Convolution Algorithms*, Springer, Berlin, 1981.
2 C. Van Loan, *Computational Frameworks for the Fast Fourier Transform*, SIAM, Philadelphia, 1992.
3 C. Canuto, M. Y. Hussaini, A. Quarteroni, T. A. Zang, *Spectral Methods in Fluid Dynamics*, Springer, New York, 1988.
4 S. Paszkowski, *Numerical Applications of Chebyshev Polynomials and Series*, Nauka, Moscow, 1983 [in Russian].
5 R. Schaback, H. Werner, *Numerische Mathematik*, Springer, Berlin, 1992.
6 C. W. Clenshaw, A note on the summation of Chebyshev series, *Math. Comp.* 9:118–120 (1955).
7 V. K. Dzjadyk V. V. Ivanov, On asymptotics and estimates for the uniform norms of the Lagrange interpolation polynomials corresponding to the Chebyshev nodal points, *Anal. Math.* 9:85–97 (1983).
8 R. Haverkamp, Approximationsfehler der Ableitungen von Interpolations polynomen, *J. Approx. Theory* 30:180–196 (1980).
9 G. P. Tolstov, *Fourier Series*, Dover, New York, 1962.
10 P. J. Davis, *Circulant Matrices*, Wiley, New York, 1979.
11 G. Steidl, Fast radix-p discrete cosine transform, *Algebra in Engrg. Comm. Comput.* 3:39–46 (1992).
12 C. de Boor, *A Practical Guide to Splines*, Springer, New York, 1978.
13 N. K. Basu, On double Chebyshev series approximation, *SIAM J. Numer. Anal.* 10:496–505 (1973).

Received 14 September 1994; final manuscript accepted 1 August 1995