THE ALGEBRAIC APPROACH TO THE DISCRETE COSINE AND SINE TRANSFORMS AND THEIR FAST ALGORITHMS*

MARKUS PÜSCHEL† AND JOSÉ M. F. MOURA†

Abstract. It is known that the discrete Fourier transform (DFT) used in digital signal processing can be characterized in the framework of representation theory of algebras, namely as the decomposition matrix for the regular module $\mathbb{C}[Z_n] = \mathbb{C}[x]/(x^n - 1)$. This characterization provides deep insight on the DFT and can be used to derive and understand the structure of its fast algorithms. In this paper we present an algebraic characterization of the important class of discrete cosine and sine transforms as decomposition matrices of certain regular modules associated to four series of Chebyshev polynomials. Then we derive most of their known algorithms by pure algebraic means. We identify the mathematical principle behind each algorithm and give insight into its structure. Our results show that the connection between algebra and digital signal processing is stronger than previously understood.

Key words. Discrete cosine transform, DCT, discrete sine transform, DST, discrete trigonometric transform, discrete Fourier transform, DFT, FFT, polynomial transform, fast algorithm, Chebyshev polynomial, algebra representation, group representation, symmetry

AMS subject classifications. Primary 42C05, 42C10, 33C80, 33C90, 65T50, 65T99; Secondary 15A23, 62-07

1. Introduction.

Many algorithms in digital signal processing are based on the use of linear discrete signal transforms. Mathematically, such a transform is a matrix-vector multiplication $a \mapsto M \cdot a$, where $a \in \mathbb{F}^n$ is the sampled signal, and $M \in \mathbb{F}^{n \times n}$ is the transform over some base field $\mathbb{F}$. We will only consider $\mathbb{F} = \mathbb{C}$. Crucial for the applicability of a signal transform M is the existence of fast algorithms that allow its computation with $O(n \log n)$ operations (or better) compared to $O(n^2)$ arising from a direct matrix-vector multiplication. The problem of finding these algorithms for different transforms has been a major research topic leading to a vast number of publications in signal processing and mathematics.

In this paper we present an algebraic approach to the class of the 16 trigonometric transforms in the framework of algebra representation theory. Then we use algebraic methods to derive most of their known fast algorithms. Our results give insight into the structure and the existence of these algorithms and extend the relationship between signal processing and algebra that is currently mainly restricted to the discrete Fourier transform.

1.1. Transforms and Algorithms.

Probably the most famous example of a signal transform is the discrete Fourier transform (DFT), which is used in harmonic analysis to decompose a signal into its frequency components. Important algorithms for the DFT include the "fast Fourier transform" (FFT) found by Cooley/Tukey [11] (originally due to Gauß [23]), Rader's algorithm for prime size [39], Winograd's algorithms [54], and several others. An overview on DFT algorithms is, for example, in [49].

Another important class of transforms consists of the 8 different types of discrete cosine and sine transforms (DCTs and DSTs), respectively, also called discrete trigonometric transforms (DTTs). Their applications include image and video compression, [40]. Important algorithms for the trigonometric transforms were developed

*This work was supported by NSF through award 9988296.

†Dept. of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, {pueschel,moura}@ece.cmu.edu.

by Chen, Smith and Fralick [7], Wang [52], Yip and Rao [56, 57], Vetterli and Nussbaumer [50], Lee [28], Feig [20], Chan and Ho [6], Steidl and Tasche [46], and Feig and Winograd [21].

Each of these algorithms has been derived through insightful manipulation of the transform matrix entries. The algorithms are highly structured, a property that can be used to write them as sparse factorizations of the transform matrix in a very concise way using mathematical operators. As examples, the Cooley/Tukey FFT can be written as

$$\mathrm{DFT}_{mn} = (\mathrm{DFT}_m \otimes \mathrm{I}_m) \cdot D \cdot (\mathrm{I}_m \otimes \mathrm{DFT}_n) \cdot P, \tag{1.1}$$

and an example for an algorithm for the DCT-2 is

$$\text{DCT-2}_{2n} = Q \cdot (\text{DCT-2}_n \oplus \text{DCT-4}_n) \cdot B. \tag{1.2}$$

The notation will be explained in §2; the matrices D, P, Q, B are all sparse. Both algorithms are of recursive nature.

1.2. The Algebraic Characterization of the DFT. It is well-known that a DFT (of size n) can be introduced in strict algebraic terms as the decomposition matrix for the group algebra $\mathbb{C}[Z_n]$ of a cyclic group Z_n with n elements,

$$\mathrm{DFT}_n : \ \mathbb{C}[Z_n] \to \mathbb{C} \oplus \ldots \oplus \mathbb{C}, \tag{1.3}$$

or, equivalently, as the decomposition matrix for the algebra $\mathbb{C}[x]/(x^n - 1)$,

$$\mathrm{DFT}_n : \ \mathbb{C}[x]/(x^n - 1) \to \mathbb{C}[x]/(x - \omega_n^0) \oplus \ldots \oplus \mathbb{C}[x]/(x - \omega_n^{n-1}), \tag{1.4}$$

with respect to appropriate bases in each case. These decompositions are instantiations of the Wedderburn theorem for the semi-simple algebra $\mathbb{C}[Z_n] \cong \mathbb{C}[x]/(x^n - 1)$. Equations (1.3) and (1.4) show that the DFT is indeed an algebraic object and thus provides a deep understanding of its use in signal processing. Furthermore, (1.3) and (1.4) can be used to easily derive and explain the structure of fast DFT algorithms by algebraic constructions rather than by manipulation of the DFT entries. As an example, (1.1) arises from a stepwise decomposition of $\mathbb{C}[Z_n]$ as has been shown by Auslander, Feig, and Winograd [2] and Beth [3].

Given the algebraic characterization of the DFT, we naturally obtain the following question: Is it possible to generalize (1.3) or (1.4) to capture a larger class of signal transforms in an algebraic framework? And, in the affirmative case: How do we use the algebraic characterization to derive and explain their fast algorithms?

1.3. Beyond the DFT. Depending on the interpretation of the DFT in (1.3) and (1.4) there have been two threads of generalization.

First, (1.3) has been generalized to arbitrary finite groups $G \neq Z_n$ leading to the area of "Fourier analysis on groups" that provides a rich class of transforms and the theory to derive their fast algorithms. Examples for important results in this field include the work of Beth [4], Clausen [9], Diaconis and Rockmore [13], and Rockmore [43]. A nice overview on this area can be found in [10] and in the survey articles [29] and [44]. However, with few exceptions, the Fourier transforms on groups do not correspond to the transforms actually used in signal processing. This problem initiated a further generalization by Minkwitz [32, 31] to include $\mathbb{C}[G]$-modules that afford an arbitrary permutation representation. A matrix that decomposes such a

module was said to have "symmetry". Minkwitz discovered that the DCT (type 3) possesses such a symmetry and derived, by pure algebraic means, a fast algorithm. The approach was further generalized by Egner and Püschel to include monomial representations. A decomposition theory [37, 36] and tools to analyze a matrix for symmetry and automatically derive a factorization [19, 15, 17] were developed. In [16] these tools were successfully applied to several signal transforms. Among the DTTs, the DCT and DST of type 3 and 4 exhibited symmetry and could be decomposed by these techniques.

Second, the generalization of (1.4) to arbitrary polynomials $p(x) \neq x^n - 1$ and arbitrary bases of $\mathbb{C}[x]/p(x)$ leads to the class of "polynomial transforms". If p is arbitrary, and $(1, x, \ldots, x^{n-1})$ is chosen as a basis, one obtains a Vandermonde matrix, which is known to have a sparse factorization, e.g., [5]. Driscoll, Healy, and Rockmore [14] developed a fast algorithm for the case of arbitrary (separable) polynomials p and bases consisting of orthogonal polynomial sequences. Independently, Potts, Steidl, and Tasche provide a numerical stable version of this algorithm [35]. In this paper the DCT of type 1 is recognized as a polynomial transform. Steidl and Tasche [46] also recognized the DCT of type 3 as a polynomial transform and used this property to derive a fast algorithm. In a different context, the DCTs and DSTs of types 1–4 have been related to polynomial transforms, in some cases after appropriate normalization [26].

Taken together, we encounter the following situation with respect to the DTTs.

1. There are 16 types of DTTs and a large number of publications on their fast algorithms.

2. In signal processing the DTTs are characterized as eigenmatrices of certain linear time-invariant processes with given boundary conditions [33, 47].

3. Four DTTs have been shown to exhibit group symmetries, and, in each case, an algorithm has been derived by algebraic means.

4. Two DTTs have been shown to be polynomial transforms (note that this property is not equivalent to 3.). In one case this has been used to derive a fast algorithm. Further 6 DTTs have been recognized as polynomial transforms after suitable normalization.

This sets the framework for the results presented in this paper.

1.4. The Algebraic Characterization of the DTTs. In this paper we present the algebraic characterization of the DTTs. This shows that, like the DFT, the DTTs are algebraic objects. Then we use the algebraic framework to derive, and explain, most of the fast DTT algorithms known in the literature. The results extend our previous preliminary work [38].

In particular we present the following:

1. A complete algebraic characterization of all 16 DTTs as scaled polynomial transforms (a generalization of polynomial transforms to be defined) arising from polynomial algebras $A = \mathbb{C}[x]/p(x)$ and A-modules of the form $f \cdot A$, where f is a scaling function. The construction of these modules follows the defining signal processing properties of the DTTs as eigenmatrices of certain linear time-invariant processes with given boundary conditions. Thus, our construction relates the domain of signal processing with the domain of algebra representation theory. As polynomials, four series of Chebyshev polynomials will naturally come into play.

2. A comprehensive overview on existing fast algorithms and their derivation by pure algebraic means, i.e., by manipulating modules and algebras rather than matrix entries. The algorithms are divided into classes depending on the mathematical prin-

ciple that accounts for their existence. Examples, based on a direct manipulation of the A-module M and polynomial $p(x)$ associated to a DTT, include: (1) translation of a DTT into another DTT by a sparse base change in M; (2) recursive decomposition based on a factorization of p; and (3) recursive decomposition based on a decomposition of p. We continue our investigation by deriving a striking property of the DTTs. The characterization of the DTTs as scaled polynomial transforms, i.e., in a framework of polynomial algebras and their representations, leads in a natural way to group symmetry properties, i.e., properties in the framework of groups and their representations. We will identify two ways in which group symmetries come into play (1) by extending the A-module M to a suitable $\mathbb{C}[G]$-module, where G is a finite group; and (2) through certain subgroups of the automorphism group of A. These symmetry properties lead to algorithms that are structurally different from the ones obtained by direct derivation (see above). All techniques used for the derivation of fast DTT algorithms are potentially more generally applicable.

Taken together, our results provide a comprehensive framework that puts previous results on the DTTs into a common context, thus tying the knot between their signal processing properties, their algebraic properties, and the structure of their fast algorithms.

1.5. Organization. The paper is divided into 2 parts. Part I (§2 - §6) provides the mathematical framework and establishes the algebraic interpretation of the discrete trigonometric transforms (DTTs). Part II (§7 - §10) uses different algebraic methods to derive and explain most of the known fast algorithms for the DTTs.

Part I. In §2 we briefly describe the notation and mathematical concepts we use. Polynomial transforms and scaled polynomial transforms are introduced in §3 together with their module-theoretic interpretation. In §4 we present a generalization of Chebyshev polynomials with particular attention to four special series and their properties. The 16 types of DTTs are introduced in §5 together with their defining signal processing properties. In §6 we construct for each DTT, using its signal processing properties, an associated module, which reveals that the DTTs are scaled polynomial transforms.

Part II. In §7 we present general methods to obtain fast algorithms for polynomial transforms and discuss results known from the literature. In §8 we use the algebraic properties of the DTTs to derive and understand several known fast algorithms for the DTTs. Other classes of algorithms for the DTTs are explained in §9 and are based on group representation symmetries. In §10 we will briefly discuss algorithms that are not covered by the previous methods.

2. Notation and Terminology. We will use the following notation and mathematical background.

Matrices: An $(n \times n)$-matrix with entry $a_{k,\ell}$ at row k and column ℓ is written as $[a_{k,\ell}]$. In most cases we provide the index range of k, ℓ as subscript. We denote by $A \oplus B = \left[\begin{smallmatrix} A & \\ & B \end{smallmatrix}\right]$ the direct sum of A and B. If $A = [a_{k,\ell}]$ then $A \otimes B = [a_{k,\ell} \cdot B]$ denotes the tensor or Kronecker product of A and B. The conjugation is written as $A^B = B^{-1} \cdot A \cdot B$. A monomial matrix has exactly one non-zero entry in every row and column. If σ is a permutation (usually written in cycle notation), we will denote a corresponding $(n \times n)$-matrix as $[\sigma, n]$, which has ones at positions $(i, \sigma(i))$. The special case $\sigma : i \mapsto ki \bmod n - 1$, $i = 0 \dots n-2$, and $n-1 \mapsto n-1$, for $k \mid n$, is called "stride permutation", and we write $[\sigma, n] = \mathrm{L}^n_k$. A diagonal matrix is written as $\mathrm{diag}(L)$, where L is the list of diagonal elements. A monomial matrix is denoted

by $[\sigma, L] = [\sigma, |L|] \cdot \mathrm{diag}(L)$.

Polynomials: Polynomials are denoted by lower case letters, $p(x)$, $q(x)$, etc. We will often drop the argument for convenience. A polynomial is called separable, if its zeros are pairwise distinct, i.e., if $\deg(p) = n$ then

$$p(x) = \prod_{k=0}^{n-1} (x - \alpha_k), \quad \alpha_i \neq \alpha_j, \text{for } i \neq j.$$

Algorithms: If B is a $(n \times n)$-matrix, we mean "by a fast algorithm for B" a fast algorithm for computing the matrix vector product $x \mapsto B \cdot x$. Algorithms are given by factorizations, $B = B_1 \cdots B_k$, where all B_i are sparse. If we refer to the arithmetic cost of an algorithm or the arithmetic complexity of matrices B, we mean the number of additions and multiplications different from $1, -1$ (cf. [5]).

Algebras and Modules: We assume the reader is familiar with the basic theory of algebras and modules. Examples for introductory books on this topic are [12, 25]. All algebras in this paper are $\mathbb{C}$-algebras. In particular we will consider the polynomial algebra $\mathbb{C}[x]$ and factor algebras $\mathbb{C}[x]/p(x)$, where p is a separable polynomial, and group algebras $\mathbb{C}[G]$, where G is a finite group. Each of the algebras $A = \mathbb{C}[x]/p(x)$ or $A = \mathbb{C}[G]$ is of finite dimension and semi-simple, i.e., every finite dimensional (left or right) A-module, can be decomposed into a direct sum of irreducible submodules,

$$M \cong M_1 \oplus \ldots \oplus M_k,$$

which is called the *Wedderburn decomposition* of M. If bases in M and M_i, $i = 1 \ldots k$, are chosen, then this isomorphism can be expressed by a matrix, which we will refer to as a *Wedderburn matrix*. The module M is usually a left module unless otherwise stated. If M has dimension n as $\mathbb{C}$ vector space, and a basis is chosen, then M affords a matrix representation of A, i.e., a homomorphism

$$\phi : \ A \to \mathbb{C}^{n \times n}.$$

The Wedderburn decomposition of M is equivalent to a decomposition of ϕ into a direct sum of irreducible representations. In the special case where $M \cong A$ (as A-modules), M is called the *regular* A-module and a corresponding representation is also called regular.

The *annihilator* of M in A is defined as

$$\mathrm{ann}_A(M) = \{a \in A \mid a \cdot m = 0, \text{for all } m \in M\};$$

it is a two-sided ideal in A. If M is an A-module, then M is also a $A/\mathrm{ann}_A(M)$-module.

3. Polynomial Algebras, Modules, and Transforms. In this section we introduce polynomial transforms and their algebraic interpretation as decomposition matrices of polynomial algebras. Then we extend the definition to the more general class of "scaled" polynomial transforms. This extension will enable us to capture all trigonometric transforms in an algebraic framework.

3.1. Polynomial Transforms. Let

$$p(x) = \prod_{k=0}^{n-1} (x - \alpha_k)$$

be a separable polynomial. Then the algebra $A = \mathbb{C}[x]/p(x)$ is semi-simple and the Wedderburn decomposition of (the regular module) $M = A$ is given by the Chinese remainder theorem (CRT) as

$$\mathbb{C}[x]/p(x) \cong \bigoplus_{k=0}^{n-1} \mathbb{C}[x]/(x - \alpha_k). \tag{3.1}$$

We want to represent the isomorphism in (3.1) by a matrix.

DEFINITION 3.1 (Polynomial Transform). *Let $b = (p_0, \ldots, p_{n-1})$ be a basis of polynomials in $\mathbb{C}[x]/p(x)$, p separable, $\alpha = (\alpha_0, \ldots, \alpha_{n-1})$ the vector of zeros of p, and assume that the one-dimensional algebras $\mathbb{C}[x]/(x - \alpha_k)$ have the base vector $1 = x^0$, respectively. With these choices, the isomorphism* (3.1) *is given by the $(n \times n)$-matrix*

$$\mathcal{P}_{b,\alpha} = [p_\ell(\alpha_k)]_{k,\ell=0\ldots n-1}, \tag{3.2}$$

where k is the row index. The Wedderburn matrix $\mathcal{P}_{b,\alpha}$ is called the polynomial transform *w.r.t. the polynomials b and the sample points α (note that the order of base polynomials and sample points matters).*

The polynomial transform $\mathcal{P}_{b,\alpha}$ can also be characterized via the representation ϕ afforded by the module $A = M$ with basis b. This is the subject of the following lemma.

LEMMA 3.2. *We use previous notation. Let $p(x) = \prod_{k=0}^{n-1}(x - \alpha_k)$ be separable, $A = \mathbb{C}[x]/p(x)$ and $M = A$ the (regular) left module with basis $b = (p_0, p_1, \ldots, p_{n-1})$ and polynomial transform $\mathcal{P}_{b,a}$. Let ϕ be the corresponding representation of A. Then*

(i) *$\mathcal{P}_{b,\alpha}^{-1}$ decomposes ϕ into a direct sum of irreducible representations. More precisely,*

$$\mathcal{P}_{b,\alpha} \cdot \phi(q(x)) \cdot \mathcal{P}_{b,\alpha}^{-1} = \operatorname{diag}(q(\alpha_0), \ldots, q(\alpha_{n-1})), \quad \text{for } q(x) \in A.$$

All such decomposition matrices are given by $\mathcal{P}_{b,\alpha}^{-1} \cdot D$, where D is diagonal and invertible.

(ii) *$\mathcal{P}_{b,\alpha}^{T}$ decomposes ϕ^T into a direct sum of irreducible representations. More precisely,*

$$(\mathcal{P}_{b,\alpha}^{T})^{-1} \cdot \phi^T(q(x)) \cdot \mathcal{P}_{b,\alpha}^{T} = \operatorname{diag}(q(\alpha_0), \ldots, q(\alpha_{n-1})), \quad \text{for } q(x) \in A.$$

All such decomposition matrices are given by $\mathcal{P}_{b,\alpha}^{T} \cdot D$, where D is diagonal and invertible.

Proof. The matrix $\mathcal{P}_{b,\alpha}$ expresses the basis b of $M = A$ in the basis of the decomposed module $M' = \bigoplus_{k=0}^{n-1} \mathbb{C}[x]/(x - \alpha_k)$. Thus, the representation ρ afforded by M' is given by $\rho = \phi^{\mathcal{P}_{b,\alpha}^{-1}}$. Since all the $\mathbb{C}[x]/(x - \alpha_k)$ are submodules (of dimension 1) of M, ρ is diagonal. The projection of $q(x) \in A$ onto $\mathbb{C}[x]/(x - \alpha_k)$ is just the evaluation $q(\alpha_k)$. Since for $q(x) = x$ all eigenvalues of $\phi(x)$ are distinct, all decomposition matrices are given by $\mathcal{P}_{b,\alpha}^{-1} \cdot D$, D diagonal (and invertible). This shows (i). (ii) follows from (i) by transposition. □

Remark. The representation ϕ^T arises from the *right* regular module A.

EXAMPLE 3.3 (Vandermonde matrix). Let $A = M = \mathbb{C}[x]/p(x)$, with separable p as above. We consider the special case $b = (x^0, x^1, \ldots, x^{n-1})$. Then the polynomial transform

$$\mathcal{P}_{b,\alpha} = \left[\alpha_k^\ell\right]_{k,\ell=0\ldots n-1},$$

is precisely the transpose of a Vandermonde matrix [5].

Next, we construct the regular representation ϕ of A with respect to the basis b. Since A is cyclic (generated by the polynomial x), it is sufficient to determine the image of $x \in A$ under ϕ. Let $p(x) = \sum_{i=0}^{n} \eta_i \cdot x^i$. Then

$$x \cdot x^i = x^{i+1}, \quad i = 0 \dots n-2, \text{ and}$$
$$x \cdot x^{n-1} = x^n \equiv \sum_{i=0}^{n-1} -\eta_i \cdot x^i \bmod p(x).$$

Thus we obtain

$$\phi(x) = \begin{bmatrix} 0 & & & & -\eta_0 \\ 1 & 0 & & & -\eta_1 \\ & \ddots & \ddots & & \vdots \\ & & 1 & 0 & -\eta_{n-2} \\ & & & 1 & -\eta_{n-1} \end{bmatrix},$$

which is the transpose of the companion matrix of p. Using Lemma 3.2,

$$\phi(x)^{\mathcal{P}_{b,\alpha}^{-1}} = (\phi(x)^T)^{\mathcal{P}_{b,\alpha}^{T}} = \operatorname{diag}(\alpha_0, \dots, \alpha_{n-1}).$$

EXAMPLE 3.4 (discrete Fourier transform). We continue Example 3.3 by requiring that also $p(x) = x^n - 1$, which implies that $\alpha_k = e^{2\pi i k/n}$, $k = 0 \dots n-1$. In this case the transposed Vandermonde matrix coincides with the discrete Fourier transform (DFT) of size n,

$$\mathrm{DFT}_n = \left[e^{2\pi i k\ell/n} \right]_{k,\ell=0\dots n-1}.$$

This identifies the DFT as a polynomial transform. The corresponding representation ϕ maps x to the cyclic shift,

$$\phi(x) = \begin{bmatrix} 0 & & & 1 \\ 1 & 0 & & 0 \\ & \ddots & \ddots & \vdots \\ & & 1 & 0 \end{bmatrix},$$

and, by Lemma 3.2 and since DFT_n is symmetric,

$$(\phi(x)^T)^{\mathrm{DFT}(n)} = \operatorname{diag}_{k=0}^{n-1}(e^{2\pi i k/n}).$$

3.2. Scaled Polynomial Transforms. In §3.1 we introduced polynomial transforms as Wedderburn matrices of regular A-modules M, where $A = M = \mathbb{C}[x]$. To capture all discrete trigonometric transforms (DTTs) in an algebraic framework, we have to generalize slightly the notion of polynomial transforms. In short, we will consider *scaled* polynomial transforms. These arise when the polynomials p_ℓ in (3.2) are replaced by $f \cdot p_\ell$, where f is a complex-valued function. Every scaled polynomial transform is associated to a regular module $M \cong A$, where M can be $\neq A$. We start with the definition.

DEFINITION 3.5 (Scaled Polynomial Transforms). *Let $\mathbb{C}[x]/p(x)$, b, and α as in Definition 3.1. Further, let f be a complex-valued function satisfying $f(\alpha_k) \neq 0$, $k = 0 \dots n-1$. We define the* scaled *polynomial transform with respect to the scaling function f, basis b, and sample points α as*

$$\mathcal{P}_{f\cdot b,\alpha} = [(f \cdot p_\ell)(\alpha_k)]_{k,\ell=0\dots n-1}. \tag{3.3}$$

We can associate a regular module to a scaled polynomial transform $\mathcal{P}_{f\cdot b,\alpha}$ in the following way. The vector space $f \cdot \mathbb{C}[x] = \{f \cdot q \mid q \in \mathbb{C}[x]\}$ naturally becomes a $\mathbb{C}[x]$-module by defining

$$r \cdot (f \cdot q) = f \cdot rq, \quad \text{for } r \in \mathbb{C}[x].$$

Let p be a separable polynomial with zeros $\alpha = (\alpha_0, \dots, \alpha_{n-1})$, and $A = \mathbb{C}[x]/p$. Then $\mathbb{C}[x] \cdot (f \cdot p)$ is a $\mathbb{C}[x]$-submodule of $f \cdot \mathbb{C}[x]$, and we can construct the factor module $M = f \cdot \mathbb{C}[x]/(\mathbb{C}[x] \cdot (f \cdot p)) = f \cdot \mathbb{C}[x]/(f \cdot p)$. We will briefly write $M = f \cdot A$. Its $\mathrm{ann}_{\mathbb{C}[x]}(M) = \mathbb{C}[x] \cdot p$, and thus M is an A-module and if $b = (p_0, \dots, p_{n-1})$ is a basis of A then $f \cdot b = (f \cdot p_0, \dots, f \cdot p_{n-1})$ is a basis of M.

We summarize the properties of the module $M = f \cdot A$ and the scaled polynomial transform $\mathcal{P}_{f\cdot b,\alpha}$ in the following lemma.

LEMMA 3.6. *Let $A = \mathbb{C}[x]/p(x)$, $b = (p_0, \dots, p_{n-1})$ a basis of A, and p a separable polynomial with zeros $\alpha = (\alpha_0, \dots, \alpha_{n-1})$. Assume that f is defined as above, and that $f(\alpha_k) \neq 0$, $k = 0 \dots n-1$. Further let $M = f \cdot A$ with basis $f \cdot b$ as defined above. Then the following holds.*

(i) *M is a regular A-module.*

(ii) *The (regular) representation ϕ of A afforded by M and $f \cdot b$ is equal to the (regular) representation of A afforded by A and b.*

(iii) $\mathcal{P}_{f\cdot b,\alpha} = \mathrm{diag}(f(\alpha_0), \dots, f(\alpha_{n-1})) \cdot \mathcal{P}_{b,\alpha}$.

(iv) *The representation ϕ is diagonalized by $\mathcal{P}^{-1}_{f\cdot b,\alpha}$, the representation ϕ^T is diagonalized by $\mathcal{P}^{T}_{f\cdot b,\alpha}$.*

Proof. (i) follows since $f \not\equiv 0$, $p_i \mapsto f \cdot p_i$, $i = 0 \dots n-1$ defines an isomorphism $A \to f \cdot A$. (ii) obvious. (iii) follows straight from the definitions in (3.2) and (3.3). (iv) follows from (iii) and Lemma 3.2. □

Remark. The scaled polynomial transform $\mathcal{P}_{f\cdot b,\alpha}$ is not the Wedderburn matrix of the module $f \cdot A$ with basis $f \cdot b$. The mapping $f \cdot p_k \mapsto p_k$, $k = 0 \dots n-1$, defines an A module isomorphism between $f \cdot A$ and A. The corresponding matrix (w.r.t. the bases $f \cdot b$ and b) is the identity. Hence $f \cdot A$ and A have the same Wedderburn matrix $\mathcal{P}_{b,\alpha}$.

4. Chebyshev Polynomials.

In this section we introduce a general class of Chebyshev polynomials and their properties that we will use throughout this paper. We start with the classical cases.

4.1. The Classical Cases.

The classical Chebyshev polynomials (of the first kind) T_n are given by the three-term recurrence

$$T_0(x) = 1,\ T_1(x) = x, \quad T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x),\ n \geq 2. \tag{4.1}$$

T_n is a polynomial of degree n and can be written in a closed form as

$$T_n(x) = \cos n\theta,\ \cos\theta = x, \quad \text{for } x \in (-1, 1). \tag{4.2}$$

TABLE 4.1
T_n and U_n for $-2 \le n \le 3$.

n	-2	-1	0	1	2	3
T_n	$2x^2-1$	x	1	x	$2x^2-1$	$4x^3-3x$
U_n	-1	0	1	$2x$	$4x^2-1$	$8x^3-4x$

The recurrence formula in (4.1) is symmetric and can also be run in the other direction to obtain Chebyshev polynomials with negative n. Doing this, we obtain the symmetry property $T_{-n} = T_n$ as can also be seen from (4.2). The sequence $\{T_n \mid n \ge 0\}$ is orthogonal on the interval $(-1, 1)$ w.r.t. the weight function $w(x) = (1-x^2)^{-1/2}$, i.e.,

$$\int_{-1}^{1} T_n(x)T_m(x)w(x)dx = 0, \quad \text{for } n \neq m.$$

From the closed form (4.2) for T_n, we also readily read off its zeros as

$$\cos\frac{(k+1/2)\pi}{n}, \quad k = 0 \dots n-1.$$

Using recurrence (4.1) with changed initial conditions yields the Chebyshev polynomials U_n of the second kind

$$U_0(x) = 1,\ U_1(x) = 2x, \quad U_n(x) = 2xU_{n-1}(x) - U_{n-2}(x),\ n \ge 2.$$

The closed form of U_n is given by

$$U_n(x) = \frac{\sin(n+1)\theta}{\sin\theta},\ \cos\theta = x, \quad \text{for } x \in (-1, 1),$$

and we get $U_{-1} = 0$ and the symmetry $U_{-n-2} = -U_n$. For $-2 \le n \le 3$ the polynomials T_n, U_n are shown in Table 4.1.

A thorough introduction to Chebyshev polynomials and orthogonal polynomials in general can be found in the books of Chihara , Szegö, and Rivlin [8, 48, 42].

4.2. Generalized Chebyshev Polynomials. Now we consider the set $\mathcal{C}$ of all polynomial sequences $\{P_n \mid n \in \mathbb{Z}\}$ that satisfy the three-term recurrence

$$P_n(x) = 2xP_{n-1}(x) - P_{n-2}(x). \tag{4.3}$$

We will refer to each such sequence as Chebyshev polynomials.

LEMMA 4.1. *Let $\{P_n \mid n \in \mathbb{Z}\}$ be a sequence of Chebyshev polynomials (we drop the argument x for simplicity). Then the following holds*

(i) $P_n = P_1 \cdot U_{n-1} - P_0 \cdot U_{n-2}$.
(ii) $T_k \cdot P_n = (P_{n+k} + P_{n-k})/2$.

Proof. Clearly, a sequence P_n is uniquely determined by its initial conditions P_0 and P_1. If P_0, P_1 give rise to the sequence P_n and Q is any polynomial, then $Q \cdot P_0, Q \cdot P_1$ give rise to the sequence $Q \cdot P_n$. If further P'_0, P'_1 gives rise to the sequence P'_n, then $P_0 + P'_0, P_1 + P'_1$ gives rise to $P_n + P'_n$.

(i) First we consider the initial polynomials $0, 1$ and $1, 0$ and obtain (cf. Table 4.1)

$$\begin{aligned} P_0 = 1,\ P_1 = 0: &\quad P_n = -U_{n-2} \\ P_0 = 0,\ P_1 = 1: &\quad P_n = U_{n-1}. \end{aligned}$$

TABLE 4.2
Four series of Chebyshev polynomials. The range for the zeros is $k = 0 \dots n-1$.

	$n = 0, 1$	closed form	symmetry	zeros	weight $w(x)$
T_n	$1, x$	$\cos(n\theta)$	$T_{-n} = T_n$	$\cos \frac{(k+\frac{1}{2})\pi}{n}$	$\frac{1}{(1-x^2)^{1/2}}$
U_n	$1, 2x$	$\frac{\sin(n+1)\theta}{\sin\theta}$	$U_{-n} = -U_{n-2}$	$\cos \frac{(k+1)\pi}{n+1}$	$(1-x^2)^{1/2}$
V_n	$1, 2x-1$	$\frac{\cos(n+\frac{1}{2})\theta}{\cos\frac{1}{2}\theta}$	$V_{-n} = V_{n-1}$	$\cos \frac{(k+\frac{1}{2})\pi}{n+\frac{1}{2}}$	$\frac{(1+x)^{1/2}}{(1-x)^{1/2}}$
W_n	$1, 2x+1$	$\frac{\sin(n+\frac{1}{2})\theta}{\sin\frac{1}{2}\theta}$	$W_{-n} = -W_{n-1}$	$\cos \frac{(k+1)\pi}{n+\frac{1}{2}}$	$\frac{(1-x)^{1/2}}{(1+x)^{1/2}}$

With the previous remark this shows (i).

(ii) Induction on k. For $k = 0$ it is trivial, for $k = 1$ this is the defining recurrence (4.3) for P_n. Further we have

$$\begin{aligned} T_{k+1} \cdot P_n &= (2xT_k - T_{k-1}) \cdot P_n \\ &= 2x \cdot (P_{n+k} + P_{n-k})/2 - (P_{n+k-1} + P_{n-k+1})/2 \quad \text{(ind. hyp.)} \\ &= (P_{n+k+1} + P_{n-k-1})/2. \end{aligned}$$

In the last step we used (4.3). □

Remarks. (1) Both assertions in Lemma 4.1 remain valid, when the polynomials P_n and hence P_0, P_1 are allowed to be arbitrary complex-valued functions. (2) Lemma 4.1 shows that $\mathcal{C}$ is a free $\mathbb{C}[x]$-module of rank 2.

We are particularly interested in four polynomial sequences, T_n,U_n,V_n,W_n, in $\mathcal{C}$ arising from different initial conditions. The sequences T_n and U_n are the Chebyshev polynomials of the first and second kind as introduced above. All four sequences can be written in a closed form, have simple symmetry properties, and are orthogonal on $(-1, 1)$ w.r.t. some weight function $w(x)$. The zeros in all cases can be obtained from the closed form. These properties are summarized in Table 4.2. The results on V_n and W_n can be found in [8, p. 37,39].

Later we will need the following arithmetic properties of the Chebyshev polynomials.

LEMMA 4.2. *The following holds for all $m, n \in \mathbb{Z}$.*

(i) $T_{mn} = T_n(T_m) = T_m(T_n)$.
(ii) $U_{mn-1} = U_{m-1}(T_n)U_{n-1}$.
(iii) $U_{2m} = V_m \cdot W_m$.
(iv) $W_n(x) = (-1)^n V_n(-x)$.
(v) $T_n(1) = 1$.

Proof. Follows from the closed form of the polynomials (Table 4.2) and trigonometric identities. □

We conclude this section by stating an interesting property of the four types of Chebyshev polynomials introduced. Let P_n be any of T_n, U_n, V_n, W_n. Then, using well-known trigonometric identities, $P_n - P_{n-2}, P_n - P_{n-1}, P_n + P_{n+1}$ can again be expressed using these polynomials. In particular, this allows us to determine their zeros using Table 4.2. The complete set of identities is given in Table 4.3. The second

TABLE 4.3
Identities among the four series of Chebyshev polynomials; P_n has to be replaced by T_n, U_n, V_n, W_n to obtain rows $1, 2, 3, 4$, respectively.

	$P_n - P_{n-2}$	P_n	$P_n - P_{n-1}$	$P_n + P_{n-1}$
T_n	$2(x^2-1)U_{n-2}$	T_n	$(x-1)W_{n-1}$	$(x+1)V_{n-1}$
U_n	$2T_n$	U_n	V_n	W_n
V_n	$2(x-1)W_{n-1}$	V_n	$2(x-1)U_{n-1}$	$2T_n$
W_n	$2(x+1)V_{n-1}$	W_n	$2T_n$	$2(x+1)U_{n-1}$

column is trivial and introduced to make the table comply with later investigations. As an example, row 2, column 1 shows that $U_n - U_{n-2} = 2T_n$.

Remarks. (1) In a few places in the literature the four series of Chebyshev polynomials occur together, e.g., in [30]. (2) If we use the close forms of T_n, U_n to extend their definitions to rational $n \in \mathbb{Q}$, we can write $V_n = T_{n+1/2}/T_{1/2}$ and $W_n = U_{n-1/2}/U_{-1/2}$. (3) For a complete overview on the factorization of T_n and U_n over $\mathbb{Q}$ see [41].

5. The 16 types of DTTs. The first discrete cosine transform was introduced by Ahmed, Natarajan, and Rao [1]. The complete set of all 8 types of discrete cosine transforms (DCTs) and discrete sine transforms (DSTs), respectively, was first presented by Wang and Hunt [53]. We will refer to them sometimes as discrete trigonometric transforms (DTTs). Each of the transforms is given by an $(n \times n)$-matrix M, $n \geq 0$, which multiplies to a signal vector a from the left, $a \mapsto M \cdot a$. As examples, we will use the symbol DCT-2 to refer to a DCT of type 2, DST-7_n to refer to a DST of type 7 and size n. If an arbitrary trigonometric transform is addressed we will write DTT or DTT$_n$. In this notation, the first DCT introduced was of type 2.

Table 5.1 gives the definitions of all 16 types of DCTs and DSTs, by stating the respective entry at position (k, ℓ), where k is the row index, for $k, \ell = 0 \dots n-1$. As can be seen, all entries are pure cosines or sines of the form $\cos r\pi$ or $\sin r\pi$, where r is some rational number. Thus, all entries are elements in a suitable cyclotomic field over $\mathbb{Q}$. The definitions given in Table 5.1 are the *unscaled* versions of the DCTs and DSTs, which will be considered in this paper. The *scaled* versions of the DCTs and DSTs are orthonormal and arise from the unscaled versions by multiplying in some cases the first and/or last row and/or column by $1/\sqrt{2}$, which makes the matrix orthogonal. In addition, the entire matrix is multiplied by a factor to achieve orthonormality. As an example, the orthonormal version of the DCT-2 has entries

$$\sqrt{\frac{2}{n}} \cdot c_k \cdot \cos k\left(\ell + \frac{1}{2}\right)\frac{\pi}{n}, \quad k, \ell = 0 \dots n-1,$$

where $c_k = \sqrt{1/2}$ for $k = 0$ and $c_k = 1$ elsewhere. For the convenience of the reader, the scaled, orthonormal versions of the DCTs and DSTs are given in Table A.1 in the appendix. Note that the set of all DTTs is closed under matrix transposition. From Table 5.1 it is easily seen that the DCT and the DST of types 1, 4, 5, and 8 are symmetric, and that types 2 and 3, and types 6 and 7 are converted into each other by transposition, respectively.

All 16 DCTs and DSTs arise as eigenmatrices of certain tridiagonal matrices

TABLE 5.1
8 types of DCTs and DSTs, given for size n. The entry at row k and column ℓ is given for $k, \ell = 0 \dots n-1$.

	DCTs	DSTs
type 1	$\cos k\ell \frac{\pi}{n-1}$	$\sin(k+1)(\ell+1)\frac{\pi}{n+1}$
type 2	$\cos k(\ell+\frac{1}{2})\frac{\pi}{n}$	$\sin(k+1)(\ell+\frac{1}{2})\frac{\pi}{n}$
type 3	$\cos(k+\frac{1}{2})\ell\frac{\pi}{n}$	$\sin(k+\frac{1}{2})(\ell+1)\frac{\pi}{n}$
type 4	$\cos(k+\frac{1}{2})(\ell+\frac{1}{2})\frac{\pi}{n}$	$\sin(k+\frac{1}{2})(\ell+\frac{1}{2})\frac{\pi}{n}$
type 5	$\cos k\ell \frac{\pi}{n-\frac{1}{2}}$	$\sin(k+1)(\ell+1)\frac{\pi}{n+\frac{1}{2}}$
type 6	$\cos k(\ell+\frac{1}{2})\frac{\pi}{n-\frac{1}{2}}$	$\sin(k+1)(\ell+\frac{1}{2})\frac{\pi}{n+\frac{1}{2}}$
type 7	$\cos(k+\frac{1}{2})\ell\frac{\pi}{n-\frac{1}{2}}$	$\sin(k+\frac{1}{2})(\ell+1)\frac{\pi}{n+\frac{1}{2}}$
type 8	$\cos(k+\frac{1}{2})(\ell+\frac{1}{2})\frac{\pi}{n+\frac{1}{2}}$	$\sin(k+\frac{1}{2})(\ell+\frac{1}{2})\frac{\pi}{n-\frac{1}{2}}$

[47, 45] of size $(n \times n)$, which can be chosen of the form

$$B(\beta_1,\beta_2,\beta_3,\beta_4) = \frac{1}{2}\cdot \begin{bmatrix} \beta_1 & \beta_2 & & & & & \\ 1 & 0 & 1 & & & & \\ & 1 & 0 & 1 & & & \\ & & \cdot & \cdot & \cdot & & \\ & & & 1 & 0 & 1 & \\ & & & & \beta_3 & \beta_4 & \end{bmatrix}. \tag{5.1}$$

The internal structure of $B(\beta_1,\beta_2,\beta_3,\beta_4)$ corresponds to the equation

$$a_k = \frac{1}{2}(a_{k-1} + a_{k+1}), \quad 1 \le k \le n-2. \tag{5.2}$$

The entries β_1, β_2 are determined by a choice of *left boundary conditions (b.c.)* that determine how a_{-1} is chosen in (5.2) for $k = 0$. The 4 left b.c. considered are $a_{-1} = a_1$, $a_{-1} = 0$, $a_{-1} = a_0$, $a_{-1} = -a_0$. For example, the choice $a_{-1} = a_1$ leads to $\beta_1 = 0, \beta_2 = 2$. Similarly, the entries β_3, β_4 are determined by right b.c. arising from the choice of a_n in (5.2) for $k = n-1$. The right b.c. are the mirrored versions of the left b.c.: $a_n = a_{n-2}$, $a_n = 0$, $a_n = a_{n-1}$, $a_n = -a_{n-1}$. The complete set of values $\beta_1, \beta_2, \beta_3, \beta_4$ for all 16 possible combinations of b.c. is given in Table 5.2. If b.c., and thus values $\beta_1, \beta_2, \beta_3, \beta_4$ are chosen, and $a = (a_0, \dots, a_{n-1})^T$, then (5.2), $k = 0 \dots n-1$, can be written as

$$a = B(\beta_1,\beta_2,\beta_3,\beta_4) \cdot a.$$

Remarks. (1) The matrices $B(\cdot)$ in (5.1) correspond to linear time-invariant processes with boundary conditions [33, 47]. (2) The b.c. $a_{-1} = 0$ and $a_{-1} = -a_0$ are the discrete versions of Dirichlet b.c.; $a_{-1} = a_1$ and $a_{-1} = a_0$ are the discrete versions of Neumann b.c. Analogously for the right b.c. [33, 47].

The 16 DTTs correspond to these different choices of boundary conditions as shown in Table 5.3 [47]. The relationship is as follows. If numbers $\beta_1, \beta_2, \beta_3, \beta_4$ (and hence left and right b.c.) are chosen from row k and row ℓ, respectively, of Table 5.2 $(k, \ell = 1 \dots 4)$, then the corresponding matrix $B(\beta_1,\beta_2,\beta_3,\beta_4)$ is diagonalized by *the transpose* of the DTT of size n given in row k and column ℓ of Table 5.3.

TABLE 5.2
The values $\beta_1, \beta_2, \beta_3, \beta_4$ from (5.1) for the 4 respective choices of left b.c. and right b.c.

left b.c.	β_1	β_2
$a_{-1} = a_1$	0	2
$a_{-1} = 0$	0	1
$a_{-1} = a_0$	1	1
$a_{-1} = -a_0$	−1	1

right b.c.	β_3	β_4
$a_n = a_{n-2}$	2	0
$a_n = 0$	1	0
$a_n = a_{n-1}$	1	1
$a_n = -a_{n-1}$	1	−1

TABLE 5.3
The left and right boundary conditions associated with the DCTs and DSTs.

	$a_n = a_{n-2}$	$a_n = 0$	$a_n = a_{n-1}$	$a_n = -a_{n-1}$
$a_{-1} = a_1$	DCT-1	DCT-3	DCT-5	DCT-7
$a_{-1} = 0$	DST-3	DST-1	DST-7	DST-5
$a_{-1} = a_0$	DCT-6	DCT-8	DCT-2	DCT-4
$a_{-1} = -a_0$	DST-8	DST-6	DST-4	DST-2

EXAMPLE 5.1. As an example we choose left b.c. $a_{-1} = a_0$ and right b.c. $a_n = a_{n-1}$ and obtain $\beta_1 = \beta_2 = \beta_3 = \beta_4 = 1$. The $(n \times n)$-matrix

$$B(1,1,1,1) = \frac{1}{2} \cdot \begin{bmatrix} 1 & 1 & & & & \\ 1 & 0 & 1 & & & \\ & \cdot & \cdot & \cdot & & \\ & & 1 & 0 & 1 & \\ & & & 1 & 1 & \end{bmatrix}. \tag{5.3}$$

is diagonalized by $\text{DCT-2}_n^T = \text{DCT-3}_n$, i.e., $B(1,1,1,1)^{\text{DCT-3}_n}$ is diagonal.

Remarks. (1) The DTTs of type 5-8 are also called "odd" DTTs of type 1-4, respectively. (2) Reference [47] considers the matrices $2\,\mathrm{I} - 2B(\cdot)$, rather than the matrices $B(\cdot)$, which leads to equivalent diagonalization properties. Also the definition of the DTTs is transposed to our definition. We chose the original [53] and commonly used definition.

6. The Algebraic Characterization of the DTTs. In this section we will show that all 16 DTTs are scaled polynomial transforms (see §5) by constructing the corresponding modules and bases. To connect, for a given DTT, its diagonalization property, i.e., the associated matrix $B(\beta_1, \beta_2, \beta_3, \beta_4)$ (cf. §5), with the algebra/module framework, we will construct a module with basis b that affords a representation ϕ, such that

$$\phi^T(x) = B(\beta_1, \beta_2, \beta_3, \beta_4).$$

In other words, the operation of x (via multiplication) on b is reflected by the matrix $B(\beta_1, \beta_2, \beta_3, \beta_4)$. Lemma 3.6, (iv), will establish the correspondence between the DTT and the module constructed this way.

The construction of the module and its base is a three step procedure. Assume a DTT and an associated matrix $B(\cdot)$ is given.

1. *Internal structure* (§6.1): Determine a sequence of polynomials that yields the internal structure of $B(\cdot)$, i.e., the $\ldots, \frac{1}{2}, 0, \frac{1}{2}, \ldots$ in each column. This will bring into play generalized Chebyshev polynomials in a natural way.

2. *Left boundary conditions* (§6.2): Fix the left b.c. This corresponds to fixing the initial conditions for the Chebyshev polynomials, i.e., the choice of a particular sequence of Chebyshev polynomials.

3. *Right boundary conditions* (§6.3) Fix the right b.c. This corresponds to choosing the appropriate polynomial p for the module (and the algebra) $\mathbb{C}[x]/p$.

6.1. Internal Structure. First we will consider n-dimensional modules that carry the structure given in (5.2). Rewriting (4.3) in a slightly different form as

$$x \cdot P_k = \frac{1}{2}(P_{k-1} + P_{k+1}) \tag{6.1}$$

shows that this is afforded by any regular module $A = \mathbb{C}[x]/p$, $\deg(p) = n$, if we choose the basis $b = (P_0, \ldots, P_{n-1})$ where the P_k are generalized Chebyshev polynomials. In other words, the image of x under the representation ϕ afforded by A with basis b will have an internal structure similar to the matrices given in (5.1).

6.2. Left Boundary Conditions. The 4 left b.c. associated with the DTTs are (see Table 5.3)

$$a_{-1} = a_1,\ a_{-1} = 0,\ a_{-1} = a_0,\ a_{-1} = -a_0. \tag{6.2}$$

They apply in the boundary case $k = 0$ in (5.2). An equivalent behavior is obtained in (6.1) if we choose the 4 special sequences of Chebyshev polynomials, T_k, U_k, V_k, W_k introduced in Table 4.2. The symmetry properties of these polynomials (cf. Table 4.2) correspond to the left b.c. in (6.2),

$$T_{-1} = T_1,\ U_{-1} = 0,\ V_{-1} = V_0,\ W_{-1} = -W_0,$$

respectively. As an example, *every* regular module $\mathbb{C}[x]/p$ with basis $(T_0, \ldots, T_{n-1})$ carries the left b.c. $a_{-1} = a_1$.

6.3. Right Boundary Conditions. The 4 right b.c. associated with the DTTs mirror the left b.c. (see Table 5.3)

$$a_n = a_{n-2},\ a_n = 0,\ a_n = a_{n-1},\ a_n = -a_{n-1}. \tag{6.3}$$

The right b.c. are determined by the choice of p in $\mathbb{C}[x]/p$. As an example, to introduce the right b.c. $a_n = a_{n-2}$, we choose $p = P_n - P_{n-2}$ where $P \in \{T, U, V, W\}$. Thus the choices of p corresponding to (6.3) are

$$P_n - P_{n-2}, P_n, P_n - P_{n-1}, P_n + P_{n-1}, \tag{6.4}$$

respectively. To determine the zeros of p in these cases, and hence the decomposition of A and its associated decomposing polynomial transform, we need to consult Table 4.3, which covers all cases in (6.4) for $P \in \{T, U, V, W, \}$.

6.4. Summary. Before we state the interpretation of the DTTs as scaled polynomial transforms, it is perhaps instructive to consider an example.

EXAMPLE 6.1 (DST-3). We choose the left b.c. $a_{-1} = 0$, which leads to the choice of the basis $b = (U_0, \ldots, U_{n-1})$. As right b.c. we choose $a_n = a_{n-2}$, which leads to $p = U_n - U_{n-2} = 2T_n$ using Table 4.3. The decomposition of the regular module $A = \mathbb{C}[x]/T_n$ (the 2 can be dropped) is determined by the zeros of T_n, which are $\alpha = (\cos \frac{1}{2}\pi/n, \ldots, \cos(n - \frac{1}{2})\pi/n)$ (cf. Table 4.2), i.e.,

$$A = \mathbb{C}[x]/(U_n - U_{n-2}) = \mathbb{C}[x]/T_n = \bigoplus_{k=0}^{n-1} \mathbb{C}[x]/(x - \cos(k + \frac{1}{2})\pi/n).$$

The decomposing polynomial transform is given by

$$\begin{aligned}\mathcal{P}_{b,\alpha} &= [U_\ell(\cos(k+1/2)\pi/n)]_{k,\ell=0\dots n-1} \\ &= \left[\frac{\sin(\ell+1)(k+1/2)\pi/n}{\sin(k+1/2)\pi/n}\right]_{k,\ell=0\dots n-1} \\ &= \operatorname{diag}_{k=0}^{n-1}\left(\frac{1}{\sin(k+1/2)\pi/n}\right)\cdot \text{DST-3}_n,\end{aligned}$$

which shows that DST-3_n is the scaled polynomial transform

$$\text{DST-3}_n = \mathcal{P}_{f\cdot b,\alpha}, \quad f = \sin\theta,$$

associated with the module $f \cdot A$ with basis $f \cdot b$.

Next we construct the representation ϕ afforded by A with basis b. By construction, we have $x \cdot U_0 = \frac{1}{2}U_1$, $x \cdot U_\ell = \frac{1}{2}(U_{\ell-1} + U_{\ell+1})$ for $\ell = 1 \dots n-2$, and $x \cdot U_{n-1} = U_{n-2}$ (in A). We get

$$\phi(x) = \frac{1}{2}\cdot\begin{bmatrix} 0 & 1 & & & \\ 1 & 0 & 1 & & \\ & \cdot & \cdot & \cdot & \\ & & 1 & 0 & 2 \\ & & & 1 & 0 \end{bmatrix}.$$

Lemma 3.6 shows that $\phi(x)^T$ is diagonalized by DST-3_n^T = DST-2_n, namely

$$\phi^T(x)^{\text{DST-2}_n} = \operatorname{diag}(\cos\frac{1}{2}\pi/n, \dots, \cos(n-\frac{1}{2})\pi/n).$$

Using the notation from §5, $\phi(x)^T = B(0,1,2,0)$, which corresponds to the DST-3 (see Table 5.2 and 5.3) as desired.

Remarks. (1) It is intriguing that the left and right b.c. are seemingly handled differently (initial conditions versus factor polynomial). In §8.1 we will see that this construction can be reversed. (2) Note that the boundary conditions corresponding to the *left* module constructed affect the first and last *column* of the *left* representation $\phi(x)$. Lemma 3.2 shows that the *right* representation ϕ^T is decomposed by the transpose of the corresponding DTT, which complies with the fact that the b.c. affect the first and last *row* in the matrices $B(\cdot)$ (cf. §5). (3) The polynomial defining the right b.c. in Example 6.1 can be written in two ways, $U_n - U_{n-2} = 2T_n$ (cf. Table 4.3). The left form determines the b.c.; the right form provides the decomposition of $\mathbb{C}[x]/T_n$, which corresponds to the zeros of T_n.

The complete correspondence between DTTs and modules is given in Theorem 6.2. To provide a convenient overview, and because we will repeatedly use it in the following, we have combined Table 4.3, Table 5.3, and the respective scaling functions into Table 6.1.

THEOREM 6.2. *Define the 4 scaling functions $f_1 = 1$, $f_2 = \sin\theta$, $f_3 = \cos\frac{1}{2}\theta$, and $f_4 = \sin\frac{1}{2}\theta$, with $\cos\theta = x$. Choose a combination of left and right boundary conditions with index i, j from Table 6.1, $i, j = 1 \dots 4$, and let* DTT_n *be the corresponding discrete trigonometric transform. Denote the polynomial below the DTT in Table 6.1 by Q_n and its zeros by $\alpha = (\alpha_0, \dots, \alpha_{n-1})$. Choose a basis of $A = \mathbb{C}[x]/Q_n$ as $b = (P_0, \dots, P_{n-1})$ where $P = T, U, V, W$ for $i = 1, 2, 3, 4$, respectively. Then*

TABLE 6.1
Overview on the DTTs and associated modules. The left b.c. and right b.c. are in the first column (value of a_{-1}) and row, respectively. A given DTT_n is associated to the module $f \cdot \mathbb{C}[x]/Q_n$, where Q_n is given below the DTT and the scaling function f in the second column. The basis of $\mathbb{C}[x]/Q_n$ is given in the third column.

			$a_n - a_{n-2}$	a_n	$a_n - a_{n-1}$	$a_n + a_{n-1}$
a_1	1	T_ℓ	DCT-1 $2(x^2-1)U_{n-2}$	DCT-3 T_n	DCT-5 $(x-1)W_{n-1}$	DCT-7 $(x+1)V_{n-1}$
0	$\sin\theta$	U_ℓ	DST-3 $2T_n$	DST-1 U_n	DST-7 V_n	DST-5 W_n
a_0	$\cos\frac{1}{2}\theta$	V_ℓ	DCT-6 $2(x-1)W_{n-1}$	DCT-8 V_n	DCT-2 $2(x-1)U_{n-1}$	DCT-4 $2T_n$
$-a_0$	$\sin\frac{1}{2}\theta$	W_ℓ	DST-8 $2(x+1)V_{n-1}$	DST-6 W_n	DST-4 $2T_n$	DST-2 $2(x+1)U_{n-1}$

(i) DTT_n *is the scaled polynomial transform*

$$\mathrm{DTT}_n = \mathcal{P}_{f_i \cdot b, \alpha}$$

associated with the module $f_i \cdot A$ and basis $f_i \cdot b$.

(ii) *If ϕ is the representation afforded by A with b then $\phi(x)^T$ is the matrix $B(\cdot)$ in* (5.1) *given by the left and right b.c. chosen.*

(iii) *The matrix $\phi(x)^T$ is diagonalized by DTT_n^T, namely*

$$(\mathrm{DTT}_n^T)^{-1} \cdot \phi(x)^T \cdot \mathrm{DTT}_n^T = \mathrm{diag}(\alpha_0, \ldots, \alpha_{n-1}),$$

which implies that DTT_n^T is a decomposition matrix for the (right) regular representation ϕ^T of A.

Proof. By computations completely analogous to Example 6.1 for all 16 cases. □

Remarks. (1) Theorem 6.2 shows that a DTT is a polynomial transform (i.e., not scaled) iff it appears in the first row of Table 6.1. For the DCT-1 and the DCT-3 this has been recognized in [35] and [46], respectively. (2) The sparse matrices $B(\cdot)$ occur as images of $T_1 = x$ under the (right) representation ϕ^T of the respective module. Using Lemma 4.1, (ii), one can compute the images $\phi^T(T_k)$, $k = 0 \ldots n-1$, which all turn out to be sparse. This makes $(T_0, \ldots, T_{n-1})$ a natural choice of basis in the algebra (not the module) A in all 16 cases. The image $\phi^T(a)$ (or $\phi(a)$) of a generic element $a = \sum a_k T_k \in A$ has a structure that is usually referred to as Toeplitz + Hankel.

With the algebraic characterization of the DTTs given in Theorem 6.2 we are now in the position to derive and explain many of their fast algorithms known from the literature. This is the subject of the remaining sections.

7. Fast Algorithms for Polynomial Transforms. Fast algorithms for the matrix-vector multiplication with a polynomial transforms, $z \mapsto \mathcal{P}_{b,\alpha} \cdot z$ or, equivalently, sparse factorizations of $\mathcal{P}_{b,\alpha}$, have been subject of several papers. In [46] the DCT-3 and the real and imaginary part of the DFT are recognized as polynomial transforms, which is used for their factorization. In [14] and in [35] an $O(n \log^2 n)$ algorithm is derived for the case that b is an arbitrary sequence of orthogonal polynomials and α a list of arbitrary (distinct) evaluation points. Using this result in

combination with Theorem 6.2 shows that the complexity of computing a DTT_n is $O(n \log^2 n)$. The fast DTT algorithms known from the literature, however, and the following discussion show that the complexity is indeed $O(n \log n)$.

In this section we will present two general techniques that can be used to factor a polynomial transform $\mathcal{P}_{b,\alpha}$ associated to the regular module $\mathbb{C}[x]/p$. They apply in the cases

1. $p(x) = q(x) \cdot r(x)$ (p factors)
2. $p(x) = q(r(x))$ (p decomposes)

An important question is to know when the resulting matrix factors are sparse yielding a fast algorithm. We note that it is also possible to factor $\mathcal{P}_{b,\alpha}$, if

3. $p(x) = q(x) \otimes r(x)$ (p is a tensor product),

but we omit this case since it does not apply to the DTTs. Because of Lemma 3.6 the problems of finding fast algorithms for $\mathcal{P}_{b,\alpha}$ and $\mathcal{P}_{f \cdot b,\alpha}$ are equivalent.

Throughout this section, p is a separable polynomial with zero vector α.

7.1. Direct Sum. One straightforward way of obtaining a fast polynomial transform is by splitting the polynomial p recursively using the fact that, if $p = q \cdot r$,

$$\mathbb{C}[x]/p \cong \mathbb{C}[x]/q \oplus \mathbb{C}[x]/r. \tag{7.1}$$

This reduces the problem of computing one polynomial transform to the computation of 2 smaller polynomial transforms.

LEMMA 7.1. *Let $p = q \cdot r$ and assume p, q, r have the zero vectors α, β, γ, respectively. Let further b, c, d be bases of $\mathbb{C}[x]/p$, $\mathbb{C}[x]/q$, $\mathbb{C}[x]/r$, respectively. Then*

$$\mathcal{P}_{b,\alpha} = P \cdot (\mathcal{P}_{c,\beta} \oplus \mathcal{P}_{d,\gamma}) \cdot B,$$

where B is the base change matrix $b \to (c, d)$ (concatenation) corresponding to (7.1) *and P is a permutation matrix mapping $(\beta, \gamma) \mapsto \alpha$.*

Proof. Follows from the definition of B and P. □

Clearly, the decomposition in Lemma 7.1 is useful for a fast algorithm only if B is sparse or has itself a fast algorithm. As an example, the fast algorithm for the Vandermonde matrix relies on the fact that in this case B has a Toeplitz structure, which permits its computation with $O(n \log n)$ arithmetic operations [14, 34].

7.2. Decomposition. A more interesting factorization of a polynomial transform can be derived if p *decomposes* into 2 polynomials, $p(x) = q(r(x))$. We will need the following lemma.

LEMMA 7.2. *Let p be separable and of degree n with zeros $\alpha_0, \ldots, \alpha_{n-1}$. Assume $p(x) = q(r(x))$ with q of degree k and r of degree ℓ. Then for each zero β of q there are precisely ℓ zeros α_m of p such that $r(\alpha_m) = \beta$.*

Proof. Let α_m be a zero of p. Then $0 = p(\alpha_m) = q(r(\alpha_m))$. Thus r maps the $n = k\ell$ zeros of p to the k zeros of q. If β is one of the k zeros of q, then the equation $r(\alpha_m) = \beta$ has maximal $\deg(r) = \ell$ solutions α_m, thus it has precisely ℓ solutions. □

As in Lemma 7.2 let the degrees of p, q, r be n, k, ℓ, respectively, $n = k\ell$. We choose bases $c = (q_0, \ldots, q_{k-1})$ of $\mathbb{C}[x]/q$ and $d = (r_0, \ldots, r_{\ell-1})$ of $\mathbb{C}[x]/r$. Then

$$\begin{aligned} b' = (\ & r_0 \cdot q_0(r), \ldots, r_0 \cdot q_{k-1}(r), \\ & r_1 \cdot q_0(r), \ldots, r_1 \cdot q_{k-1}(r), \\ & \ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ & r_{\ell-1} \cdot q_0(r), \ldots, r_{\ell-1} \cdot q_{k-1}(r)\ ) \end{aligned}$$

is a basis of $\mathbb{C}[x]/p$. Using the shorter notation $p_{j,i,m} = (r_j \cdot q_i(r))(\alpha_m)$, the corresponding polynomial transform is given by

$$\mathcal{P}_{b',\alpha} = \begin{bmatrix} p_{0,0,0} & \cdots & p_{0,k-1,0} & \cdots & p_{\ell-1,0,0} & \cdots & p_{\ell-1,k-1,0} \\ p_{0,0,1} & \cdots & p_{0,k-1,1} & \cdots & p_{\ell-1,0,1} & \cdots & p_{\ell-1,k-1,1} \\ \cdots & & & & & & \cdots \\ p_{0,0,n-1} & \cdots & p_{0,k-1,n-1} & \cdots & p_{\ell-1,0,n-1} & \cdots & p_{\ell-1,k-1,n-1} \end{bmatrix}.$$

Because of Lemma 7.2, for each i, the n numbers $q_i(r(\alpha_m))$, $m = 0 \dots n-1$, will divide into k groups of ℓ equals. We permute α into α' with a permutation P such that $r(\alpha_{i+jk}) = \beta_i$, $i = 0 \dots k-1$, $j = 0 \dots \ell-1$, i.e.,

$$\mathcal{P}_{b',\alpha'} = P \cdot \mathcal{P}_{b',\alpha}.$$

Now $\mathcal{P}_{b',\alpha'}$ reveals the following block structure

$$\begin{aligned} \mathcal{P}_{b',\alpha'} &= [D_{h,j} \cdot \mathcal{P}_{c,\beta}]_{h,j=0\dots\ell-1}, \quad \text{with} \\ D_{h,j} &= \operatorname{diag}(r_j(\alpha'_{hk}), r_j(\alpha'_{hk+1}), \dots, r_j(\alpha'_{hk+k-1})). \end{aligned}$$

Thus we can write $\mathcal{P}_{b',\alpha'}$ as

$$\mathcal{P}_{b',\alpha'} = [D_{h,j}]_{h,j=0\dots\ell-1} \cdot (\mathrm{I}_\ell \otimes \mathcal{P}_{c,\beta}).$$

Since $D_{h,j}$ is diagonal, $h, j = 0 \dots \ell-1$, the matrix $[D_{h,j}]$ consists of k $(\ell \times \ell)$ blocks at stride k. Thus,

$$[D_{h,j}]^{\mathrm{L}^n_\ell}$$

is a direct sum of $(\ell \times \ell)$-matrices, which turn out to be again polynomial transforms. Using $(\mathrm{L}^n_\ell)^{-1} = \mathrm{L}^n_k$, we get the following theorem.

THEOREM 7.3. *We use previous notation. Then*

$$\mathcal{P}_{b,\alpha} = P \cdot \left(\bigoplus_{i=0}^{k-1} \mathcal{P}_{d,\overline{\alpha}_i} \right)^{\mathrm{L}^n_k} \cdot (\mathrm{I}_\ell \otimes \mathcal{P}_{c,\beta}) \cdot B,$$

where B is the matrix giving the base change $b \to b'$, P is a permutation matrix, and

$$\overline{\alpha}_i = (\alpha'_{0\cdot k+i}, \alpha'_{1\cdot k+i}, \dots, \alpha'_{(\ell-1)\cdot k+i}).$$

As in Lemma 7.1, the value of this factorization for deriving a fast algorithm for $\mathcal{P}_{b,\alpha}$ depends on the base change matrix B.

Theorem 7.3 can be interpreted as a generalization of the Cooley/Tukey FFT as we will see in the next example.

EXAMPLE 7.4 (FFT, size 4). We consider the case $p(x) = x^4 - 1 = (x^2)^2 - 1$, i.e., $q(x) = x^2 - 1$, and $r(x) = x^2$. As bases we choose $b = (1, x, x^2, x^3)$and $c = d = (1, x)$. The zeros of p are $\alpha = (1, i, -1, -i)$ and the zeros of q are $\beta = (1, -1)$. This is the situation of Example 3.4, $\mathcal{P}_{b,\alpha} = \mathrm{DFT}_4$, $\mathcal{P}_{c,\beta} = \mathrm{DFT}_2$. Since $r(1) = r(-1)$ and $r(i) = r(-i)$, it is $\alpha' = \alpha$. Further, $b' = (1, x^2, x, x^3)$ and thus $B = [(2, 3), 4] = \mathrm{L}^4_2$. It remains to compute $\mathcal{P}_{d,\overline{\alpha}_0}, \mathcal{P}_{d,\overline{\alpha}_1}$:

$$\mathcal{P}_{d,\overline{\alpha}_0} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \mathrm{DFT}_2, \text{ and } \mathcal{P}_{d,\overline{\alpha}_1} = \begin{bmatrix} 1 & i \\ 1 & -i \end{bmatrix} = \mathrm{DFT}_2 \cdot \operatorname{diag}(1, i).$$

As a result we get the FFT of size 4,

$$\mathrm{DFT}_4 = (\mathrm{DFT}_2 \otimes \mathrm{I}_2) \cdot \mathrm{diag}(1,1,1,i) \cdot (\mathrm{I}_1 \otimes \mathrm{DFT}_2) \cdot \mathrm{L}^4_2 .$$

Remark. It is worth giving an algebraic interpretation of Theorem 7.3 using the notation from above. Since $p(x) = q(r(x))$, $A' = \mathbb{C}[r(x)]/p(x) = \mathbb{C}[y]/q(y)$ $(y = r(x))$ is a subalgebra of $A = C[x]/p(x)$. We have

$$A \cong r_0 \cdot A' \oplus \ldots \oplus r_{\ell-1} \cdot A'$$

as vector spaces, i.e., $d = (r_0, \ldots, r_{\ell-1})$ is a transversal of A' in A. In a similar way as it is done for $\mathbb{C}[G]$-modules (G a group) [12, pp. 73], we can construct the induced module

$$A \otimes_{A'} A' = (r_0 \otimes A') \oplus \ldots \oplus (r_{\ell-1} \otimes A'),$$

which has the basis b'. The modules A and $A \otimes'_A A'$ are isomorphic with the base change given by the matrix B. Thus, Theorem 7.3 (for polynomial algebras) is the equivalent of Theorem 3.33 (for group algebras of solvable groups) in [37]. They coincide for the case $\mathbb{C}[Z_n] \cong \mathbb{C}[x]/(x^n - 1)$ (Z_n = cyclic group of order n), where they yield the Cooley/Tukey FFT (cf. Example 7.4).

8. Fast DTTs via Decomposition of Polynomial Transforms. In this section we derive and explain several different recursive algorithms for the DTTs directly from their algebraic interpretation. In contrast to the derivations given in the literature, we do not manipulate matrix entries; rather, we obtain the algorithm directly from the underlying modules. This makes the derivation simpler, more transparent, and provides a mathematically satisfying insight into the structure of the algorithm.

The algorithms presented in this section can be loosely grouped into the following categories.

1. *Translation* (§8.1): A DTT is translated into another DTT using sparse matrices. Two different methods are identified.

2. *Direct Sum* (§8.2): A DTT is decomposed into the direct sum of smaller DTTs using sparse matrices. These algorithms are due to Lemma 7.1.

3. *Reduction* (§8.3): A DTT is decomposed into smaller DTTs of the same type using sparse matrices. These algorithms are due to Theorem 7.3.

It is important to note that we can always derive, from any given fast algorithm, new fast algorithms by straightforward operations like symbolic transposition or inversion, since these are compatible with $\otimes$ and $\oplus$. As an example, a factorization like

$$\text{DCT-2}_n = P \cdot (\text{DCT-2}_{n/2} \oplus \text{DCT-4}_{n/2}) \cdot B$$

can be transposed to yield

$$\text{DCT-3}_n = B^T \cdot (\text{DCT-3}_{n/2} \oplus \text{DCT-4}_{n/2}) \cdot P^T,$$

since $\text{DCT-2}^T = \text{DCT-3}$, and DCT-4 is symmetric. Moreover, it is always possible to locally manipulate these formula expressions. As an example, let Q and R be permutations. Then

$$Q \cdot (\mathrm{I}_n \otimes \mathrm{DFT}_2) \cdot R = (QP^{-1}) \cdot (\mathrm{I}_n \otimes \mathrm{DFT}_2) \cdot (PR)$$

for any permutation P permuting (2×2)-blocks ($n!$ such P). We will consider algorithms that can be transformed into each other using manipulations of this kind as "algebraically equivalent". The comparisons between the algorithms we derive here and the algorithms from the literature have to be understood "modulo" this equivalence, though, in many cases, the comparison will be exact.

8.1. Translation between DTTs. In this section we will discuss and derive sparse relationships between the different types of DTTs. We say that DTT_n and DTT'_n are in sparse relationship, if DTT_n can be derived from DTT'_n using $O(n)$ operations. An example of a sparse relationship is the equation

$$\mathrm{DTT}_n = B_n \cdot \mathrm{DTT}'_n \cdot C_n,$$

where B_n, C_n are sparse matrices ($O(n)$ entries). These relationships are important for fast algorithms. If a fast algorithm for DTT_n is given, and DTT_n and DTT'_n are in sparse relationship, then we obtain a fast algorithm for DTT'_n, and vice-versa.

Investigating Table 6.1 we observe that transform pairs at transposed positions (i, j) and (j, i), $i, j = 1 \dots 4$, have the same associated algebra (i.e., the same polynomial Q_n). As an example, DCT-5 and DCT-6 both arise from $\mathbb{C}[x]/(x-1)W_{n-1}$ with different bases. This leads to the concept of duality introduced in the next definition.

Definition 8.1 (Duality). *We call a pair* DTT_n, DTT'_n *dual to each other if the left b.c. of* DTT_n *correspond to the right b.c. of* DTT'_n*, and vice-versa. Equivalently,* DTT_n *and* DTT'_n *appear in transposed positions* (i, j) *and* (j, i) *in Table 6.1. If* $i = j$ *we call* $\mathrm{DTT}_n = \mathrm{DTT}'_n$ *self-dual.*

We show how this duality can be used to derive a sparse relationship between the transforms.

In §6 we derived a module for a given pair of b.c. by fixing (1) a base sequence of Chebyshev polynomials P_n depending on the left b.c., and (2) depending on the right b.c., a polynomial p in $\mathbb{C}[x]/p$. Since the recursion formula (4.3) for Chebyshev polynomials is symmetric, this can be done in a reverse way. We illustrate this with the pair DCT-3 and DST-3. The DST-3 has the left b.c. $a_{-1} = 0$ that fixes the base sequence U_ℓ, and it has the right b.c. $a_n = a_{n-2}$ that is fixed by $p = U_n - U_{n-2} = 0$. In alternative, we can realize the same b.c. by the sequence T_ℓ, $\ell = -n+1 \dots 0$. Now the right b.c. are given by $a_{-1} = a_1$, i.e., $T_1 = T_{-1}$, which corresponds to $U_n - U_{n-2} = 0$. The left b.c. are fixed by $p = T_{-n} = T_n = 0$. The correspondence between the forward U_ℓ and the backward T_ℓ is as follows

$$\begin{array}{c|ccc|c} 0 = U_{-1} & U_0, & \dots, & U_{n-1} & U_n = U_{n-2} \\ 0 = T_{-n} & T_{-(n-1)}, & \dots, & T_0 & T_{-1} = T_1, \end{array}$$

where the vertical lines indicate the boundaries. In other words, using $T_{-\ell} = T_\ell$, the two bases $(U_0, \dots, U_{n-1})$ and $(T_{n-1}, \dots, T_0)$ afford identical representations of $A = \mathbb{C}[x]/T_n$. Thus, the corresponding polynomial transforms must be scaled versions of each other. And indeed, if α_k denotes the zeros of T_n, we get, using basic trigonometric identities,

$$\begin{aligned} T_{n-1-\ell}(\alpha_k) &= \cos(n-1-\ell)(k+\frac{1}{2})\pi/n \\ &= (-1)^k \cdot \sin(\ell+1)(k+\frac{1}{2})\pi/n, \end{aligned}$$

and thus, using the definition of DST-3 and DCT-3 (Table 5.1),

$$\mathrm{diag}_{k=0}^{n-1}((-1)^k) \cdot \text{DST-3}_n = \text{DCT-3}_n \cdot \mathrm{J}_n, \tag{8.1}$$

where J_n denotes the opposite identity, i.e., the permutation matrix exchanging $i \leftrightarrow n-i$, $i = 0 \dots n-1$. Similar computations for all pairs of dual transforms yield the following result.

THEOREM 8.2 (Translation by Duality). *Let* DTT_n *and* DTT'_n *be a pair of dual transforms. Then*

$$\mathrm{diag}_{k=0}^{n-1}((-1)^k) \cdot \mathrm{DTT}_n = \mathrm{DTT}'_n \cdot \mathrm{J}_n,$$

In particular, dual DTTs have the same arithmetic complexity.

A second class of sparse relationships can be obtained in certain cases by appropriate base changes and will be explained in the following. Going back to Table 6.1, we see that the 16 DTTs are partitioned into 4 groups of 4 transforms each depending on the polynomial Q_n, which is essentially equal to one of the Chebyshev polynomials T_n, U_n, V_n, W_n. For example, on the main-diagonal in Table 6.1 are all DTTs in the "U-group", which are exactly the self-dual ones. Each of the other groups consists of two pairs of dual DTTs, respectively. For each 2 DTTs within the same group the corresponding algebra $\mathbb{C}[x]/Q_n$ is basically equal. The difference is in the basis chosen in the module. Thus, it is possible to derive a sparse relationship by performing an appropriate base change. We will illustrate this in the following 2 examples.

EXAMPLE 8.3 (DCT-3 and DST-3). We consider again the pair DCT-3_n and DST-3_n. Using Table 6.1 we see that both transforms correspond to the same algebra, but with different bases,

$$\begin{aligned} \text{DCT-3}_n &\leftrightarrow \mathbb{C}[x]/T_n,\ b &&= (T_0, \dots, T_{n-1}), \\ \text{DST-3}_n &\leftrightarrow \mathbb{C}[x]/T_n,\ b' &&= (U_0, \dots, U_{n-1}), \end{aligned}$$

and that DCT-$3_n = [T_\ell(\alpha_k)]$, and DST-$3_n = D \cdot [U_\ell(\alpha_k)]$, where α_k are the zeros of T_n and $D = \mathrm{diag}_{k=0}^{n-1}(\sin(k+\frac{1}{2})\pi/n)$ arises from the scaling function. To compute the base change matrix B for $b \to b'$, we use that $T_\ell = \frac{1}{2}(U_\ell - U_{\ell-2})$ (2nd row, 1st column in Table 4.3), and get

$$B = \frac{1}{2} \cdot \begin{bmatrix} 2 & 0 & -1 & & & & \\ & 1 & 0 & -1 & & & \\ & & \cdot & \cdot & \cdot & & \\ & & & 1 & 0 & -1 & \\ & & & & 1 & 0 & \\ & & & & & 1 & \end{bmatrix}.$$

Thus, $[T_\ell(\alpha_k)] = [U_\ell(\alpha_k)] \cdot B$ and hence

$$D \cdot \text{DCT-3}_n = \text{DST-3}_n \cdot B.$$

Note that this relationship is different from the one arising from the duality of DCT-3_n and DST-3_n (Theorem 8.2).

EXAMPLE 8.4 (DCT-1 and DST-2). We will translate a DCT-1_{n+1} into a DST-2_n. Using again Table 6.1 we get as associated algebras and bases

$$\begin{aligned} \text{DCT-1}_{n+1} &\leftrightarrow \mathbb{C}[x]/(x^2-1)U_{n-1},\ b &&= (T_0, \dots, T_n), \\ \text{DST-2}_n &\leftrightarrow \mathbb{C}[x]/(x+1)U_{n-1},\ b' &&= (W_0, \dots, W_{n-1}). \end{aligned}$$

Note that we have to choose size $n+1$ and n, respectively, to obtain comparable algebras. We have DCT-$1_{n+1} = [T_\ell(\alpha_k)]$ and DST-$2_n = D \cdot [W_\ell(\alpha_k)]$, where $\alpha_k =$

$\cos k\pi/n$, $k = 0 \dots n$ are the zeros of $(x^2-1)U_{n-1}$ (2nd row in Table 4.2). For the DST-2_n, $\alpha_0 = 1$ is skipped. The scaling matrix is $D = \mathrm{diag}_{k=0}^{n-1}(\sin(k+1)\pi/2n)$ (Table 6.2). We compute the base change matrix B for

$$\mathbb{C}[x]/(x^2-1)U_{n-1} \cong \mathbb{C}[x]/(x-1) \oplus \mathbb{C}[x]/(x+1)U_{n-1}.$$

The bases are $b, (1), b'$, respectively. Using $T_\ell = \frac{1}{2}(W_\ell - W_{\ell-1})$ (4th row, 3rd column in Table 4.3) and $T_n = \frac{1}{2}(W_n - W_{n-1}) \equiv -W_{n-1} \bmod (x+1)U_{n-1}$ (because, again from Table 4.3, $(x+1)U_{n-1} = \frac{1}{2}(W_n + W_{n-1})$) we get

$$B = \frac{1}{2} \cdot \begin{bmatrix} 2 & 2 & 2 & \cdot & \cdot & 2 \\ 2 & -1 & & & & \\ & 1 & -1 & & & \\ & & \cdot & \cdot & & \\ & & & \cdot & -1 & \\ & & & & 1 & -2 \end{bmatrix}.$$

The $1's$ in the first row are due to $T_\ell(1) = 1$ (Lemma 4.2). We get $[T_\ell(\alpha_k)] = (\mathrm{I}_1 \oplus [W_\ell(\alpha_k)]) \cdot B$ and hence

$$(\mathrm{I}_1 \oplus D) \cdot \text{DCT-1}_{n+1} = (\mathrm{I}_1 \oplus \text{DST-2}_n) \cdot B.$$

We obtain the following theorem.

THEOREM 8.5 (Translation by Base Change). *All DTTs of type 1-4 are in sparse relationship. All DTTs of type 5-8 are in sparse relationship.*

Proof. Similar computations as in Examples 8.3 and 8.4 show that all DTTs of type 1 and 2 (the "U-group") are in sparse relationship, and that all DTTs of type 3 and 4 (the "T-group") are in sparse relationship. By transposition we obtain sparse relationship for DTTs of type 2 and 4 and thus for all DTTs of type 1-4, which is the first assertion. The other statement is proved analogously. □

Of particular importance is the translation between a DCT-4 and DCT-2, which, together with Theorem 8.6, yields a fast algorithm for the DCT-2 [28].

Remarks. (1) Aside from Definition 8.1 there is another, more obvious, form of duality among the DTTs: DTT and DTT$'$ are dual if $\text{DTT}^T = \text{DTT}'$. Currently, we have no algebraic explanation for this duality. (2) Note that "sparse relationship" does not define an equivalence relation. Every two matrices (of the same size) can be converted into each other using a (long enough) sequence of sparse matrices.

8.2. Direct Sum: Fast Algorithms via Polynomial Factorization. In this section we will derive recursive algorithms for all DTTs in the U-group, i.e., the DCT and DST of type 1 and 2 The algorithms are based on the rational factorization of the polynomials U_n given in Lemma 4.2, (ii) and (iii).

As an example we will consider a DCT-2_n where $n = 2m$. Consulting Table 6.1 we get as corresponding algebra $\mathbb{C}[x]/(x-1)U_{n-1}$ with basis $b = (V_0, \dots, V_{n-1})$. Lemma 4.2, (ii) gives the factorization $U_{2m-1} = 2 \cdot U_{m-1} \cdot T_m$, which leads to the isomorphism

$$\mathbb{C}[x]/(x-1)U_{2m-1} \cong \mathbb{C}[x]/(x-1)U_{m-1} \oplus \mathbb{C}[x]/T_m. \tag{8.2}$$

For the summands we choose bases b, b', b', respectively, $b' = (V_0, \dots, V_{m-1})$. The zeros of $(x-1)U_{n-1}$ are $\cos k\pi/n$, $k = 0 \dots n-1$. Thus, the first summand in

(8.2) collects the zeros with even k, and the second summand the zeros with odd k (cf. Table 4.2). Now the decomposition of DCT-2_n follows Lemma 7.1. To compute the base change matrix B in (8.2) we need

$$V_{m+k} \equiv V_{m-k-1} \bmod (x-1)U_{m-1}, \text{ and}$$
$$V_{m+k} \equiv -V_{m-k-1} \bmod T_m,$$

which can be shown by induction using $(x-1)U_{m-1} = V_m - V_{m-1}$, $T_m = V_m + V_{m+1}$, and (4.3). We get

$$B = \begin{bmatrix} 1 & & & & 1 \\ & . & & . & \\ & & 1 & 1 & \\ 1 & & & & -1 \\ & . & & . & \\ & & 1 & -1 & \end{bmatrix} = \begin{bmatrix} \mathrm{I}_m & \mathrm{J}_m \\ \mathrm{I}_m & -\mathrm{J}_m \end{bmatrix} = (\mathrm{DFT}_2 \otimes \mathrm{I}_m)(\mathrm{I}_m \oplus \mathrm{J}_m).$$

The summands in (8.2) are decomposed recursively using a DCT-2_m and DCT-4_m, respectively. The resulting one-dimensional summands are permuted in canonical order using the stride permutation L^n_m (see §2). Since DCT-2 and DCT-4 have the same scaling function, we get

$$\text{DCT-2}_{2m} = \mathrm{L}^{2m}_m \cdot (\text{DCT-2}_m \oplus \text{DCT-4}_m) \cdot B.$$

Besides DCT-2, similar derivations can be performed on the three remaining transforms in the U-group DCT-1, DST-1, and DST-2 using $U_{2m-1} = 2U_{m-1}T_m$ and using $U_{2m} = V_m W_m$. The complete set of identities can be stated using two types of block matrices and two types of permutation matrices. The block matrices give the base change,

$$B_{2m} = \begin{bmatrix} \mathrm{I}_m & \mathrm{J}_m \\ \mathrm{I}_m & -\mathrm{J}_m \end{bmatrix}, \quad B_{2m+1} = \begin{bmatrix} \mathrm{I}_m & 0 & \mathrm{J}_m \\ 0 & 1 & 0 \\ \mathrm{I}_m & 0 & -\mathrm{J}_m \end{bmatrix},$$

and the permutation matrices give the reordering of the irreducible modules,

$$P_{2m} = \mathrm{L}^{2m}_m,$$
$$P_{2m+1} : \ i \to (m+1)i \bmod 2m+1, \quad i = 0 \dots 2m.$$

THEOREM 8.6. *The following recursive algorithms for DTTs are based on the rational factorization $U_{2m-1} = 2 \cdot U_{m-1} \cdot T_m$. We also indicate where they first appeared in the literature (to our best knowledge).*

(i) DCT-$1_{2m+1} = P_{2m+1} \cdot (\text{DCT-1}_{m+1} \oplus \text{DCT-3}_m) \cdot B_{2m+1}$, [27].
(ii) DST-$1_{2m-1} = P_{2m-1} \cdot (\text{DST-3}_m \oplus \text{DST-1}_{m-1}) \cdot B_{2m-1}$, [55].
(iii) DCT-$2_{2m} = P_{2m} \cdot (\text{DCT-2}_m \oplus \text{DCT-4}_m) \cdot B_{2m}$, [7].
(iv) DST-$2_{2m} = P_{2m} \cdot (\text{DST-4}_m \oplus \text{DST-2}_m) \cdot B_{2m}$, [52].

Theorem 8.6 is complemented by the decompositions in the following theorem. We did not find these in the literature.

THEOREM 8.7. *The following recursive algorithms for DTTs are based on the rational factorization $U_{2m} = V_m W_m$.*

(i) DCT-$1_{2m} = P_{2m} \cdot (\text{DCT-5}_m \oplus \text{DCT-7}_m) \cdot B_{2m}$.
(ii) DST-$1_{2m} = P_{2m} \cdot (\text{DST-7}_m \oplus \text{DST-5}_m) \cdot B_{2m}$.

(iii) $\text{DCT-2}_{2m+1} = P_{2m+1} \cdot (\text{DCT-6}_{m+1} \oplus \text{DCT-8}_m) \cdot B_{2m+1}$.
(iv) $\text{DST-2}_{2m+1} = P_{2m+1} \cdot (\text{DST-8}_{m+1} \oplus \text{DST-6}_m) \cdot B_{2m+1}$.

Remarks. (1) Transposition of the decompositions in Theorems 8.6 and 8.7 yields algorithms for the DTTs of type 3. (2) Theorem 8.6 reveals why the DCT-1 and the DST-1 are usually considered at sizes $2^k + 1$ and $2^k - 1$, respectively. The available algorithms are more efficient since there are no simple sparse factorizations of the DTTs of types 5-8. (3) Theorem 8.5 and Theorem 8.6 combined give a complete set of algorithms for the DTTs of type 1-4, of 2-power size (for type 1 the size differs by 1, see remark (2)). (4) It is possible to derive algorithms for the more general case $n = k\ell$, using the factorization in Lemma 4.2, (ii).

8.3. Reduction: Fast Algorithms via Polynomial Decomposition. In this section we derive algorithms based on the decomposition of the polynomial T_n in Lemma 4.2, (i). This decomposition property allows the decomposition of all *DTTs* in the T-group, i.e, DCT and DST of type 3 and 4, using Theorem 7.3.

As an example we will consider a DCT-3_n, where $n = 2m$. Using Table 6.1 we get the corresponding algebra $\mathbb{C}[x]/T_n$ with basis $b = (T_0, \ldots, T_{n-1})$. We use the decomposition $T_{2m} = T_m(T_2)$ (Lemma 4.2). Following Theorem 7.3 and its proof, we choose bases $c = (T_0, \ldots, T_{m-1})$ and $d = (T_0, T_1)$ of $\mathbb{C}[x]/T_m$ and $\mathbb{C}[x]/T_2$, respectively. We get the new basis

$$b' = (T_0, T_2, \ldots, T_{2m-2}, T_1, (T_1 + T_3)/2, \ldots, (T_{2m-3} + T_{2m-1})/2).$$

Thus, the base change $b' \to b$ is given by

$$B = \begin{bmatrix} 1 & & & 0 & & & & \\ 0 & 0 & & 1 & \frac{1}{2} & & & \\ 0 & 1 & & 0 & 0 & & & \\ 0 & 0 & & 0 & \frac{1}{2} & \frac{1}{2} & & \\ & \vdots & & & & & \vdots & \\ & & & & & & & \frac{1}{2} \\ & & 1 & & & & & 0 \\ & & 0 & & & & & \frac{1}{2} \end{bmatrix},$$

and the base change $b \to b'$ by B^{-1}. The zeros of T_n are $\alpha_k = \cos(k + \frac{1}{2})\pi/n$, $\alpha_k = -\alpha_{n-1-k}$, and $T_2(\alpha_k) = T_2(\alpha_{n-1-k})$. Thus, the permutation P in Theorem 7.3 is $P = (\mathrm{I}_m \oplus \mathrm{J}_m)$. Further,

$$\mathcal{P}_{d,\overline{\alpha}_i} = \begin{bmatrix} T_0(\alpha_i) & T_1(\alpha_i) \\ T_0(\alpha_{n-1-i}) & T_1(\alpha_{n-1-i}) \end{bmatrix} = \begin{bmatrix} 1 & \alpha_i \\ 1 & -\alpha_i \end{bmatrix} = \mathrm{DFT}_2 \cdot \mathrm{diag}(1, \alpha_i),$$

for $i = 0 \ldots m$. This can be used to derive $(\bigoplus_{i=0}^{m-1} \mathcal{P}_{d,\overline{\alpha}_i})^{\mathrm{L}_m^{2m}} = (\mathrm{DFT}_2 \otimes \mathrm{I}_m) \cdot (\mathrm{I}_m \oplus \mathrm{diag}_{i=0}^{m-1}(\alpha_i))$ and we get

$$\text{DCT-3}_{2m} = P \cdot (\mathrm{DFT}_2 \otimes \mathrm{I}_m) \cdot (\mathrm{I}_m \oplus \mathrm{diag}_{i=0}^{m-1}(\alpha_i)) \cdot (\mathrm{I}_2 \otimes \text{DCT-3}_m) \cdot B^{-1}.$$

Further simplification can be achieved by writing $B = C \cdot (\mathrm{I}_m \oplus \frac{1}{2}\,\mathrm{I}_m)$ and observing that $\mathrm{I}_m \oplus 2\,\mathrm{I}_m$ commutes with $\mathrm{I}_2 \otimes \text{DCT-3}_m$. For simplicity we set $D = \mathrm{diag}_{i=0}^{m-1}(\alpha_i)$ and get

$$\text{DCT-3}_{2m} = P \cdot (\mathrm{DFT}_2 \otimes \mathrm{I}_m) \cdot (\mathrm{I}_m \oplus 2D) \cdot (\mathrm{I}_2 \otimes \text{DCT-3}_m) \cdot C^{-1}. \tag{8.3}$$

Equation (8.3) is also a good example to study the effect of transposition and inversion to derive new algorithms from known ones. Transposition of (8.3) is straightforward and yields

$$\text{DCT-2}_{2m} = C^{-T} \cdot (\mathrm{I}_2 \otimes \text{DCT-2}_m) \cdot (\mathrm{I}_m \oplus 2D) \cdot (\mathrm{DFT}_2 \otimes \mathrm{I}_m) \cdot P. \tag{8.4}$$

For the inversion of (8.3) we need $\text{DCT-3}_n^{-1} = \frac{2}{n} \cdot \mathrm{diag}(\frac{1}{2}, 1, \ldots, 1) \cdot \text{DCT-2}_n$. After simplifications we get

$$\text{DCT-2}_{2m} = C_1 \cdot (\mathrm{I}_2 \otimes \text{DCT-2}_m) \cdot (\mathrm{I}_m \oplus (2D)^{-1}) \cdot (\mathrm{DFT}_2 \otimes \mathrm{I}_m) \cdot P, \tag{8.5}$$

where C_1 arises from C by setting the entry 2 at position $(2, m+1)$ to 1. C_1 incurs only additions. Transposing (8.5) (or, equivalently, inverting (8.4)) yields again an algorithm for DCT-3.

$$\text{DCT-3}_{2m} = P \cdot (\mathrm{DFT}_2 \otimes \mathrm{I}_m) \cdot (\mathrm{I}_m \oplus (2D)^{-1}) \cdot (\mathrm{I}_2 \otimes \text{DCT-3}_m) \cdot C_1^T. \tag{8.6}$$

Equations (8.5) and (8.6) are very similar to (8.4) and (8.3), respectively, with the difference that inverting the entries in the diagonal (middle factor) saves one multiplication by 2 in the base change matrix (C_1 vs. C). More crucial, the additions in C_1 and C_1^T can be performed in parallel (i.e., the critical path has length 1), which does not hold for C^{-1} and C^{-T}.

Each of the equations (8.3)–(8.6) occurs in the literature. The references are: (8.3) [24], (8.4) [24], (8.5) [57] (transposed definition of DTTs), (8.6) [28] and [56] (transposed definition of DTTs).

Note that (8.4) can also be obtained by first applying Theorem 8.6, (iii), and then translating the resulting DCT-4 using Theorem 8.5.

Similar computations for the other DTTs in the T-group yield the following result.

THEOREM 8.8. *Let $n = 2m$. All DTTs in the T-group have a fast recursive algorithm of the form*

$$\mathrm{DTT}_{2m} = P \cdot (\mathrm{DFT}_2 \otimes \mathrm{I}_m) \cdot (\mathrm{I}_m \oplus D) \cdot (\mathrm{I}_2 \otimes \mathrm{DTT}_m) \cdot B,$$

where P is a permutation matrix, D is diagonal, and B is sparse. This factorization is based on $T_{2m} = T_m(T_2)$ and the concrete form of P and B can be obtained using Theorem 7.3.

For the DST-3 the factorization can be also found in [56]. For DCT-4 and DST-4, the factorizations do not appear in literature. They are less efficient with respect to arithmetic cost.

Remark. It is possible to derive a recursive algorithm based on $T_{k\ell} = T_n(T_m)$ using Theorem 7.3. The problem for larger m is the further decomposition of the occurring matrices $\mathcal{P}_{d,\overline{\alpha}_i}$ in Theorem 7.3.

9. Fast DTTs via Group Symmetries. In this section we will derive fast DTT algorithms that are based on "group symmetries" in the sense defined below. In the cases where they occur, these symmetries are a direct consequence of the DTT properties in Theorem 6.2. We will identify two ways in which group symmetries might come into play.

1. *Extension* (§9.2): By extension to a group algebra of the algebra $A = \mathbb{C}[x]/p$ associated to a DTT.

2. *Automorphisms* (§9.3): By subgroups of the automorphism group of A,

These symmetries lead to algorithms that are substantially different from the ones derived in §8.

For the convenience of the reader, we overview briefly group symmetry based matrix factorization. We take in the following a "representation" approach, instead of the equivalent "module-with-basis" point of view.

9.1. Group Symmetry based Matrix Factorization. Matrix factorization based on group symmetries has its origin in [31, 32] and was generalized in [15, 36, 37, 19] to the form presented here. In [18] the technique was successfully applied to several discrete signal transforms, which initiated the research presented in this paper. Due to space limitations we can only give a brief overview and refer to these references for further details.

In the following G is a finite solvable group. All representations of G (or, equivalently $\mathbb{C}[G]$) in the following arise from *right* G-modules. The entire approach is based on the following definition of symmetry.

DEFINITION 9.1. *Let B be an arbitrary complex matrix. A pair (ϕ_1, ϕ_2) of representations of G is called a* symmetry *of B, if*

$$\phi_1(g) \cdot B = B \cdot \phi_2(g), \quad \text{for } g \in G.$$

G is called a symmetry group *of B.*

If B has a symmetry, we can factor B according to Figure 9.1. We choose matrices A_1, A_2 that decompose ϕ_1, ϕ_2, respectively, into a direct sum of irreducible representations ρ_1 and ρ_2. Then we compute the matrix

$$D = A_1^{-1} \cdot B \cdot A_2$$

so that the diagram commutes. We obtain the factorization

$$B = A_1 \cdot D \cdot A_2^{-1}.$$

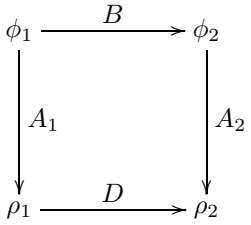

FIG. 9.1. *Factorization of the matrix M with symmetry (ϕ_1, ϕ_2)*

The matrix D is sparse since it is the conjugating matrix for two reduced representations ρ_1, ρ_2 (a consequence of Schur's lemma [12]). This means that the factorization of B is useful as a fast algorithm for B if the A_i are sparse or can themselves be written as products of sparse matrices. This is possible in at least the following cases: (1) ϕ_i is a permuted direct sum of irreducible representations, i.e., $\phi_i = \rho_i^P$, where P is a permutation matrix. In this case we call ϕ_i of type "irred". It is $A_i = P^{-1}$. (2) ϕ_i is monomial. (A representation is monomial if all its images are monomial matrices, i.e., have exactly one non-zero entry in each row and column.) In this case we call ϕ_i

TABLE 9.1
Types of symmetry that can be used for factorizing B (irrs = irreducible representations).

mon-mon symmetry	ϕ_1 and ϕ_2 monomial
mon-irred symmetry	ϕ_1 monomial, ϕ_2 permuted direct sum of irrs
irred-mon symmetry	ϕ_2 monomial, ϕ_1 permuted direct sum of irrs

being of type "mon". The decomposition matrix A_i can be determined as a product of sparse matrices using the algorithm in [37]. Briefly sketched, this algorithm translates the monomial representation into an induction that is decomposed stepwise along a composition series using certain recursion formulas similar to Theorem 7.3. We call ϕ_i of type "mon".

Depending on the types of the ϕ_i, we obtain the 3 types of symmetry shown in Table 9.1. We omitted the type "irred-irred" since it requires B to be already sparse.

Algorithms for finding symmetry [19] and the algorithm [37] for the stepwise decomposition of monomial representations have been implemented as part of the GAP [22] share package AREP [17, 16] for constructive group representation theory. Thus, AREP can find these factorizations automatically and can be used as a discover tool for sparse matrix factorizations, i.e., fast algorithms.

In the remainder of this section, we will show that mon-irred symmetries as well as mon-mon symmetries occur among the DTTs and how these symmetries can be derived. We will also discuss the structure of the resulting algorithms.

9.2. Algorithms by Extension to Group Algebras. In this section we will show which DTTs possess a mon-irred symmetry that can be used for deriving fast algorithms. In [18] exactly 4 DTTs exhibited a mon-irred symmetry with dihedral symmetry groups in all cases. The transforms were the DCT and DST of type 3 and 4. We will now explain and derive these symmetries. Note that we will deal with right representations (arising from right modules) to comply with the symmetry definition 9.1. A right representation is the transpose of a left representation.

We start with the general case of a scaled polynomial transform. As usual, let b be a basis of $A = \mathbb{C}[x]/p$ and let α be the zero vector of p. Further let f be a scaling function. If ϕ is the *right* representation afforded by the regular module A (or, equivalently, $f \cdot A$), then, by Lemma 3.6,

$$\phi \cdot \mathcal{P}^T_{f\cdot b,\alpha} = \mathcal{P}^T_{f\cdot b,\alpha} \cdot \rho,$$

where ρ is a direct sum of one-dimensional irreducible representations of A. If ϕ can be extended to a representation $\overline{\phi}$ of a group algebra $\mathbb{C}[G]$ of a finite group G, then ρ extends to a permuted direct sum of irreducible representations of $\mathbb{C}[G]$ (on extension, the one-dimensional irreducibles in ρ—not necessarily adjacent ones—may fuse to irreducibles of $\mathbb{C}[G]$ of larger dimension). In other words, $\mathcal{P}^T_{f\cdot b,\alpha}$ decomposes $\overline{\phi}$, up to a permutation. We obtain the following result.

LEMMA 9.2. *We use previous notation. If the right regular representation ϕ of $A = \mathbb{C}[x]/p$ can be extended to a representation $\overline{\phi}$ of a group algebra $\mathbb{C}[G]$, where G is finite, then*

$$\overline{\phi} \cdot \mathcal{P}^T_{f\cdot b,\alpha} = \mathcal{P}^T_{f\cdot b,\alpha} \cdot \overline{\rho},$$

where $\overline{\rho}$ is a permuted direct sum of irreducible representations of $\mathbb{C}[G]$. If in particular

$\overline{\phi}$ is monomial, then $\mathcal{P}^T_{f\cdot b,\alpha}$ has a mon-irred symmetry, and $\mathcal{P}_{f\cdot b,\alpha}$ has an irred-mon symmetry, both with symmetry group G.

Now we will apply Lemma 9.2 to determine which DTTs possess a mon-irred symmetry. Consider a fixed DTT with associated regular representation ϕ. The representation ϕ can be extended to a monomial representation, iff all images $\phi(q)$, $q \in A$ can be written as a linear combination of monomial matrices. Since A is cyclic, it is sufficient to consider the images of the generator $\phi(x)$, which is given by the corresponding matrix $B(\cdot)$ in (5.1).

THEOREM 9.3. *The 4 transforms* DCT_n *and* DST_n *of type 3 and 4, $n \geq 0$, are the only DTTs that have a mon-irred symmetry (ϕ, ρ). Denote with $D_{2k} = \langle \sigma, \tau \mid \sigma^2 = \tau^2 = (\sigma\tau)^k = 1 \rangle$ the dihedral group with $2k$ elements. Further, let, for even n,*

$$\pi_1 = (1,2)(3,4)\dots(n-1,n), \text{ and } \pi_2 = (2,3)(4,5)\dots(n-2,n-1),$$

and for odd n,

$$\pi_1 = (1,2)(3,4)\dots(n-2,n-1), \text{ and } \pi_2 = (2,3)(4,5)\dots(n-1,n)$$

(viewed as permutation on $\{1,\dots,n\}$). The symmetry group for DCT-3$_n$ *and* DST-3$_n$ *is D_{2n}, for* DCT-4$_n$ *and* DST-4$_n$ *is D_{4n}. The respective monomial representation ϕ is given for even n by*

$$\begin{array}{ll}
\text{DCT-3}_n: & \sigma \mapsto [\pi_1, n],\ \tau \mapsto [\pi_2, n], \\
\text{DST-3}_n: & \sigma \mapsto [\pi_1, n],\ \tau \mapsto [\pi_2, (-1,1,\dots,1,-1)] \\
\text{DCT-4}_n: & \sigma \mapsto [\pi_1, n],\ \tau \mapsto [\pi_2, (1,\dots,1,-1)], \\
\text{DST-4}_n: & \sigma \mapsto [\pi_1, n],\ \tau \mapsto [\pi_2, (-1,1\dots,1,)],
\end{array}$$

and for odd n by

$$\begin{array}{ll}
\text{DCT-3}_n: & \sigma \mapsto [\pi_1, n],\ \tau \mapsto [\pi_2, n], \\
\text{DST-3}_n: & \sigma \mapsto [\pi_1, (1,\dots,1,-1)],\ \tau \mapsto [\pi_2, (-1,1,\dots,1)] \\
\text{DCT-4}_n: & \sigma \mapsto [\pi_1, (1,\dots,1,-1)],\ \tau \mapsto [\pi_2, n], \\
\text{DST-4}_n: & \sigma \mapsto [\pi_1, n],\ \tau \mapsto [\pi_2, (-1,1\dots,1,)].
\end{array}$$

Proof. For all 16 DTTs and their associated representations ϕ, we have to consider the matrices $\phi(x) = B(\beta_1, \beta_2, \beta_3, \beta_4)$, with β_i as given in Table 5.2. Because of its structure, $B(\cdot)$ can be written as a linear combination of monomial matrices, iff it can be written as the sum of 2 monomial matrices. Assume $\beta_1 = 0$. Writing $B(0,\dots)$ as sum of 2 monomial matrices M_1, M_2 requires that both, M_1 and M_2 have an entry $\neq 0$ at position $(1,2)$. Since the entry $(3,2)$ of $B(0,\dots)$ is also $\neq 0$, this decomposition is not possible. Analogously, a decomposition is not possible, if $\beta_4 = 0$. In the remaining 4 cases the decomposition is possible and yields the desired results. We will give one case as an example. It is readily verified that

$$B(1,1,1,1) = [\pi_1, n] + [\pi_2, n].$$

The permutations π_1, π_2 are involutions and hence generate a dihedral group D_{2m}. The number m is the order of the product $\pi_1\pi_2$, here n. By Theorem 6.2 and Table 5.2, $B(1,1,1,1)$ is diagonalized by DCT-3$_n$, which proves the result. The other 3 cases can be treated analogously. □

Remark. Theorem 9.3 explains the symmetries found in [18].

We want to briefly sketch the decomposition procedure for a DCT-4. For full details we refer the reader to [18, 19, 37].

EXAMPLE 9.4 (DCT-4). We consider a DCT-4 of size $n = 2^k$. By Theorem 9.3, the matrix $B = \text{DCT-4}_{2^k}$ has a mon-irred symmetry (ϕ, ρ) with dihedral symmetry group $D_{2^{k+2}}$. We follow Figure 9.1. The decomposition algorithm will decompose ϕ stepwise along the composition series

$$D_{2^{k+2}} \geq D_{2^{k+1}} \geq \cdots \geq D_{2^2},$$

using a recursion formula for the induction of representations. Note that the last representation of D_{2^2} is decomposed since it is of dimension 1. This gives rise to a factorized decomposition matrix A_1 of ϕ. The representation ρ is a permuted direct sum of irreducible representations and can thus be decomposed by a permutation matrix A_2. The correction matrix D is computed as $D = A_1^{-1} \cdot B \cdot A_2$ to yield

$$\text{DCT-4}_{2^k} = A_1 \cdot D \cdot A_2^{-1}.$$

As an example, we give a factorization of a DCT-4_8 as it is automatically found by AREP,

$$\begin{array}{rcl} \text{DCT-4}_8 & = & [(1,2,8)(3,6,5),(1,-1,1,1,1,-1,1,1)]\cdot \\ & & (\mathrm{I}_2 \otimes ((\mathrm{I}_2 \oplus \frac{1}{\sqrt{2}} \cdot \mathrm{DFT}_2) \cdot [(3,4),4] \cdot (\mathrm{DFT}_2 \otimes \mathrm{I}_2)))\cdot \\ & & [(1,3)(2,4)(5,7)(6,8),8] \cdot (\mathrm{I}_4 \oplus \mathrm{R}_{\frac{15}{8}\pi} \oplus \mathrm{R}_{\frac{11}{8}\pi})\cdot \\ & & (\mathrm{DFT}_2 \otimes \mathrm{I}_4) \cdot [(3,5,7)(4,6,8),8]\cdot \\ & & \frac{1}{2} \cdot (\mathrm{R}_{\frac{31}{32}\pi} \oplus \mathrm{R}_{\frac{19}{32}\pi} \oplus \mathrm{R}_{\frac{27}{32}\pi} \oplus \mathrm{R}_{\frac{23}{32}\pi})\cdot \\ & & [(1,8,5,6,3,2)(4,7),8]. \end{array} \tag{9.1}$$

The (factorized) matrix A_1 is given in lines 1-4, the matrix D in line 5, and the matrix A_2^{-1} in line 6 (the last line).

We observe that the factorization in (9.1) contains rotation matrices

$$R_a = \begin{bmatrix} \cos(a) & \sin(a) \\ -\sin(a) & \cos(a) \end{bmatrix},$$

which do not occur in the algorithms derived in §8. The general (arbitrary $n = 2^k$) version of this algorithm can be found in [7] (corrected in [51, 52]). Combining this algorithm with Theorem 8.6, (iii) yields a factorization of DCT-2, and thus, by transposition, of DCT-3, into rotation matrices [7]. The obtained algorithm coincides with the one derived from the mon-irred symmetry of the DCT-3.

Note that the algorithms arising from a mon-irred symmetry occur only in an iterative form in the literature, i.e., the transform matrix is completely factorized (as in (9.1)) and not into transforms of smaller size. The reason is in the decomposition procedure (cf. Figure 9.1), since not B, but A_1 is decomposed recursively.

Remark. It is striking that, e.g., the algorithm for a DCT-3_{2^k} arising from its mon-irred symmetry and the algorithm from Theorem 8.8 have precisely the same arithmetic cost [7, 28, 56].

9.3. Algorithms from Automorphism Groups. In §9.2 we showed how, in certain cases, a mon-irred symmetry of a DTT can be derived from its interpretation as a (scaled) polynomial transform. In the following we will show that the—completely different—type of mon-mon symmetry also occurs among the DTTs. This type of

symmetry, if present, arises from the automorphism group of the associated algebra. All modules in this section will be right modules.

We introduce the following notation. Let $A = \mathbb{C}[x]/p$. Automorphisms of A will be denoted by letters g, h. We multiply automorphisms from left to right, i.e., in gh, g is applied before h. This complies with applying automorphisms from the right, i.e., if $q \in A$, we write q^g for the image of q under g. If ϕ is a representation of A, and g an automorphism, then $\phi^g : \ q \mapsto \phi(q^g)$ defines another representation of A. As suggested by this notation, $(\phi^g)^h = \phi^{gh}$.

A possible source of a mon-mon symmetry of a polynomial transform $\mathcal{P}_{b,\alpha}$ is described in the following theorem.

THEOREM 9.5. *Let $A = \mathbb{C}[x]/p$ be a regular module with basis b. The polynomial p is separable and has zeros $\alpha = (\alpha_0, \ldots, \alpha_{n-1})$. Denote by ϕ the (right regular) representation afforded by A and b. Assume that A has a group G of automorphisms with the property that for each $g \in G$ there exists a monomial matrix M_g with*

$$\phi^g = \phi^{{M_g}^{-1}}. \tag{9.2}$$

Then $\mathcal{P}^T_{b,\alpha}$ has a mon-mon symmetry (χ, ψ) with symmetry group $\overline{G} \cong \langle M_g \mid g \in G \rangle$. It is $G \cong \overline{G}/N$, where $N \trianglelefteq \overline{G}$ denotes the normal subgroup defined by

$$\begin{aligned} g' \in N &\Leftrightarrow \phi(q) \cdot \chi(g') = \chi(g') \cdot \phi(q), \quad \text{for all } q \in A \\ &\Leftrightarrow \chi(g') \in \phi(A). \end{aligned}$$

If D is any invertible diagonal matrix, then $(D \cdot \mathcal{P}_{b,\alpha})^T = \mathcal{P}^T_{b,\alpha} \cdot D$ has the same mon-mon symmetry as $\mathcal{P}^T_{b,\alpha}$.

Proof. First we note that the set $S = \{M_g \mid g \in G\}$ is not a group since for every g there are (if any) many possible choices for M_g, e.g., all $a \cdot M_g$, where $a \in \mathbb{C}$. Conversely, every $M_g \in S$ uniquely defines an automorphism of A, since $\phi^g = \phi^h$, and ϕ faithful, implies $g = h$. Now we reverse the situation by defining a mapping $\gamma : \ S \to G$, $M_g \mapsto g$. Let $\overline{G} = \langle S \rangle$ (the group generated by S). Then γ can be extended to a homomorphism $\overline{\gamma} : \ \overline{G} \to G$, since, for $M, M' \in S$, and using (9.2),

$$\phi^{\overline{\gamma}(MM')} = \phi^{(MM')^{-1}} = \left(\phi^{{M'}^{-1}}\right)^{M^{-1}} = \left(\phi^{\overline{\gamma}(M')}\right)^{M^{-1}} = \left(\phi^{M^{-1}}\right)^{\overline{\gamma}(M')} = \phi^{\overline{\gamma}(M)\overline{\gamma}(M')}.$$

By definition, $\overline{\gamma}$ is surjective, and the kernel of $\overline{\gamma}$ is given by $N = \{M \mid \phi = \phi^M\}$, and thus $G \cong \overline{G}/N$. Since $M \in N$ implies that M commutes with each $\phi(q)$, $q \in A$, $M \in \phi(A)$. Viewing $\overline{G}$ as a monomial representation χ of itself shows all assertions on $\overline{G}$.

It remains to show that $\mathcal{P}^T_{b,\alpha}$ has a mon-mon symmetry (χ, ψ). To this end we choose an arbitrary monomial matrix $M = \chi(M)$ in $\overline{G}$. The representation ϕ is decomposed by $\mathcal{P}^T_{b,\alpha}$ into a direct sum of irreducible representations ρ (cf. Lemma 3.2). Thus, $\phi^{\overline{\gamma}(M)}$ is also decomposed by $\mathcal{P}^T_{b,\alpha}$ into a direct sum of irreducible representations ρ'. Following Figure 9.2 there is a unique matrix M' such that

$$M \cdot \mathcal{P}^T_{b,\alpha} = \mathcal{P}^T_{b,\alpha} \cdot M'.$$

Since M' conjugates ρ' onto ρ, it is monomial. Setting $\psi(M) = M'$ defines a monomial representation of $\overline{G}$, and shows that $\mathcal{P}^T_{b,\alpha}$ has the mon-mon symmetry (χ, ψ).

If D is any invertible diagonal matrix, then $\rho^D = \rho$ and ${\rho'}^D = \rho'$, since all irreducible summands of ρ, ρ' are of dimension 1. Thus, we can replace $\mathcal{P}^T_{b,\alpha}$ by

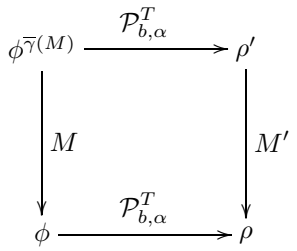

FIG. 9.2. *Constructing a mon-mon symmetry for* $\mathcal{P}^T_{b,\alpha}$.

$\mathcal{P}^T_{b,\alpha} \cdot D$ in Figure 9.2 obtaining the same mon-mon symmetry. This completes the proof. □

Remarks. (1) $\mathcal{P}^T_{b,\alpha}$ has the mon-mon symmetry (χ, ψ) iff $\mathcal{P}_{b,\alpha}$ has the mon-mon symmetry (ψ^T, χ^T). (2) The last assertion in Theorem 9.5 shows that we can apply it to scaled polynomial transforms, and thus to the DTTs.

In the remainder of this section, we will use Theorem 9.5 to derive the mon-mon symmetries for the DTTs whose transposes are in the T-group, i.e., those with associated algebra $\mathbb{C}[x]/T_n$ (cf. Table 6.1) for the special case where $n = 2^m$ is a 2-power. A complete investigation of all DTTs and sizes would exceed the space available.

First we need a suitable group G of automorphisms of $A = \mathbb{C}[x]/T_n$.

LEMMA 9.6. *Let* $n = 2^m$ *and* $A = \mathbb{C}[x]/T_n$. *Each mapping*

$$g_k : \ T_1 \mapsto T_k, \ \textit{and}\ g_{-k} : \ T_1 \mapsto -T_k, \quad 1 \le k \le n, \ k \ \textit{odd},$$

defines an automorphism of the algebra A. *The set* G_n *of all such* $g_{\pm k}$ *is a cyclic group of order* n.

Proof. Before we start the proof we investigate the sequence T_k, $k \ge 0$ in A. The following two equations allow the reduction of each T_k modulo T_n.

$$\begin{array}{lll} 0 \equiv T_n T_{n-k} = \frac{1}{2}(T_{2n-k} + T_k) & \Rightarrow & T_k \equiv -T_{2n-k}, \quad \text{and} \\ 0 \equiv T_n T_{n+k} = \frac{1}{2}(T_{2n+k} + T_k) & \Rightarrow & T_k \equiv -T_{2n+k}. \end{array} \tag{9.3}$$

The latter equation also shows that $T_k \equiv T_{k+4n}$, i.e., the sequence T_k, $k \ge 1$ has period $4n$ (in A). Using (9.3) we can compute the reduced T_k, $k = 0 \dots 4n-1$, as

$$T_0 \dots T_{n-1} \mid 0 \ -T_{n-1} \dots -T_1 \mid -T_0 \dots -T_{n-1} \mid 0 \ T_{n-1} \dots T_1 \mid, \tag{9.4}$$

where the vertical lines indicate the reflection points at multiples of n.

Now we start the proof of Lemma 9.6. Let $n = 2^m$. We will repeatedly use that T_n is an even function, and that T_k, k odd, is an odd function. Also note that $g_{\pm k}$ maps $T_\ell = T_\ell(T_1) \mapsto T_\ell(\pm T_k)$.

(1) $g_{\pm k}$ is a homomorphism, since $T_n(\pm T_k) = T_n(T_k) = T_k(T_n) \equiv 0$ (Lemma 4.2, (i)), i.e., the defining equation $T_n = 0$ in A is preserved. (2) $g_{\pm k}$ is invertible; G_n is a group. Let g_k be given, k odd. We choose an ℓ with $k\ell \equiv 1 \bmod 4n$. The mapping $T_1 \mapsto T_\ell$ inverts g_k, since $T_{k\ell} \equiv T_1$ (see beginning of this proof). Similarly, $T_1 \mapsto -T_\ell$ inverts g_{-k}. Using (9.3), we can reduce $T_\ell \equiv T_{\ell'}$ or $\equiv -T_{\ell'}$ for a suitable odd $\ell' < n$. This shows that G_n is closed under inversion. Also $g_{\pm k} g_{\pm \ell} : \ T_1 \mapsto \pm T_\ell(\pm T_k) = \mp T_{k\ell}$,

which can be reduced analogously. Thus, G_n is a group. (3) G_n is cyclic. For $n = 2$, g_{-1} has order 2. For $n = 4$, g_3 has order 4 ($T_3(T_3) = T_9 \equiv -T_1$). For $n > 4$ we show that g_5 has order n. Observing (9.4), we get that g_5^e is the identity, iff $5^e \equiv \pm 1 \bmod 4n$. Since 5^e is never $\equiv -1$, and 5 has order n mod $4n$ ($n = 2^m$) we get the assumption. □

Now we will use the group G_n of automorphisms (Lemma 9.6) and Theorem 9.5 to derive mon-mon symmetries for all DTTs whose inverses are in the T-group.

THEOREM 9.7. *Let $n = 2^m \geq 4$ and G_n as defined in Lemma 9.6. The transforms* DCT-2_n, DST-2_n, DCT-4_n, DST-4_n *have a mon-mon symmetry (χ, ψ) with non-zero matrix entries ± 1 arising from the group of automorphisms G_n of $\mathbb{C}[x]/T_n$ (cf. Theorem 9.5). Denote with $Z_n = \langle \sigma \mid \sigma^n = 1 \rangle$ the cyclic group of order n. The symmetry group for* DCT-2_n, DST-2_n *is Z_n, for* DCT-4_n, DST-4_n *it is Z_{2n}. The respective monomial representation χ is given by*

$$\begin{array}{ll} \text{DCT-}2_n: & \sigma \mapsto (T_i \mapsto T_{ki} \bmod T_n)^{-1}, \\ \text{DST-}2_n: & \sigma \mapsto (U_i \mapsto U_{k-1+ki} \bmod T_n)^{-1}, \\ \text{DCT-}4_n: & \sigma \mapsto (V_i \mapsto V_{(k-1)/2+ki} \bmod T_n)^{-1}, \\ \text{DST-}4_n: & \sigma \mapsto (W_i \mapsto W_{(k-1)/2+ki} \bmod T_n)^{-1}, \end{array} \tag{9.5}$$

where $i = 0 \ldots n-1$, and $k = 3$ for $n = 4$, and $k = 5$ for $n \geq 8$.

Proof. Let $g_k \in G_n$, i.e., $T_1^{g_k} = T_k$. We consider the first case DCT-2_n = DCT-3_n^T with associated algebra $A = \mathbb{C}[x]/T_n$ and $b = (T_0, \ldots, T_{n-1})$, i.e., DCT-$2_n$ decomposes the right regular representation of A (Theorem 6.2). Following Theorem 9.5 we have to find a monomial base change matrix $M_{g_k} : b \to b'$, such that T_1 operates on b as $T_1^{g_k} = T_k$ on b'. This is afforded by $b' = (T_{k\cdot 0}, T_{k\cdot 1}, \ldots, T_{k\cdot n-1})$, since, using Lemma 4.1 (ii),

T_1 on b			T_k on b'		
$T_1 \cdot T_0$	=	T_1	$T_1 \cdot T_{k\cdot 0}$	=	$T_{k\cdot 1}$
$T_1 \cdot T_i$	=	$(T_{i-1} + T_{i+1})/2$	$T_1 \cdot T_{ki}$	=	$(T_{k(i-1)} + T_{k(i+1)})/2$
$T_1 \cdot T_{n-1}$	=	$T_{n-1}/2$	$T_1 \cdot T_{k(n-1)}$	=	$T_{k(n-1)}/2$

where $i = 2 \ldots n-2$, and in the last line we used $T_n \equiv 0$, and thus $T_{kn} = T_k(T_n) \equiv 0$, since T_k is an odd function. The base change $b \to b'$ is given by the matrix $M_k : T_i \mapsto T_{ki}$, $i = 0 \ldots n-1$, and thus

$$\phi^{g_k} = \phi^{M_k}.$$

(Note that we consider right representations, where ϕ is conjugated into $\phi^{M^{-1}}$ by a base change with matrix M.) As in the proof of Lemma 9.6, we see that M_k is monomial, since every T_{ki} can be reduced to a suitable $\pm T_\ell$ mod T_n, $0 \leq \ell \leq n-1$. Theorem 9.5 establishes a mon-mon symmetry for (χ, ψ). It remains to show that the symmetry group is cyclic of order n. To this end we need the sequence of T_ℓ, $\ell \geq 0$, reduced mod T_n, given in the first row of Table 9.2. We see that $M_k^e = \mathrm{I}_n$ iff $T_{k^e i} \equiv T_i \bmod T_n$ ($i = 0 \ldots n-1$) iff $k^e \equiv \pm 1 \bmod 4n$. As in the proof of Lemma 9.6, this shows that the maximum order $e = n$ is obtained for $k = 3$, if $n = 4$, and $k = 5$ for $n \geq 8$.

The proof of the other three cases is analogous. The base b is replaced by $b = (P_0, \ldots, P_{n-1})$, where $P = U, V, W$, respectively.

The respective base change $b \to b'$ given by the matrix M_k corresponding to the automorphism g_k is given in lines 2-4 of (9.5) (without the inversion). The operation of T_k on b' can again be established using Lemma 4.1 (ii). To determine the order of

TABLE 9.2

For $P \in \{T, U, V, W\}$ the sequence of polynomials $P_0, \ldots, P_{4n-1}$ (i.e., one period) reduced mod T_n. The vertical lines indicate multiples of n.

$$\begin{array}{ll}
\text{DCT-3}: & T_0 \ \ldots\ T_{n-1}|0\ -T_{n-1} \ldots\ -T_1|-T_0\ \ldots\ -T_{n-1}|0\ T_{n-1} \ldots\ T_1| \\
\text{DST-3}: & U_0 \ \ldots\ U_{n-1}|U_{n-2}\ \ldots\ U_0\ 0|-U_0\ \ldots\ -U_{n-1}|-U_{n-2}\ \ldots -U_0\ 0| \\
\text{DCT-4}: & V_0 \ \ldots\ V_{n-1}|-V_{n-1}\ \ldots\ -V_0|-V_0\ \ldots\ -V_{n-1}|V_{n-1}\ \ldots\ V_0| \\
\text{DST-4}: & W_0 \ldots W_{n-1}|W_{n-1}\ \ldots\ W_0|-W_0 \ldots -W_{n-1}|-W_{n-1} \ldots\ -W_0|
\end{array}$$

M_k we need, in each case, the sequence of P_ℓ reduced mod T_n given in Table 9.2 (each of these sequences has period $4n$). Surprisingly, it turns out that for the DCT-4 and DST-4, the symmetry group is factor of 2 larger than G_n. We consider the example DCT-4_n. Denote with $V_{a_{e,i}}$ the image of V_i under M_k^e, $i = 0 \ldots n-1$. We get the recurrence, and its solution

$$a_{0,i} = i, \quad a_{e,i} = (k-1)/2 + k \cdot a_{e-1,i}, \quad \Rightarrow \quad a_{e,i} = (k^e - 1)/2 + ik^e.$$

Using the third row of Table 9.2, we get

$$\begin{aligned}
& V_{(k^e-1)/2+ik^e} \equiv V_i \bmod T_n \quad (i = 0 \ldots n-1) \\
\Leftrightarrow\ & (k^e - 1)/2 + ik^e \equiv i \text{ or } -i-1 \bmod 4n \quad (i = 0 \ldots n-1) \\
\Leftrightarrow\ & k^e(2i+1) \equiv \pm(2i+1) \bmod 8n \quad (i = 0 \ldots n-1) \\
\Leftrightarrow\ & k^e \equiv \pm 1 \bmod 8n,
\end{aligned}$$

which shows that the maximum order $e = 2n$ is obtained for $k = 3$, if $n = 4$, and $k = 5$ for $n \geq 8$. □

We conclude this section with a small example.

EXAMPLE 9.8 (DCT-4, size 4). *Using Theorem 9.7, the* DCT-4_4 *has a mon-mon symmetry (χ, ψ) with a cyclic symmetry group $Z_8 = \langle\sigma\rangle$. The image $\chi(\sigma)$ is determined by the inverse of $V_i \mapsto V_{1+3i} \bmod T_4$, $i = 0 \ldots 3$. Using Table 9.2 we get $V_4 \equiv -V_3$, $V_7 \equiv -V_0$, $V_{10} \equiv -V_2$, and thus*

$$\chi(\sigma) = \begin{bmatrix} 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & -1 & 0 & 0 \end{bmatrix}, \quad \psi(\sigma) = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \end{bmatrix}.$$

The matrix $\psi(\sigma)$ was computed using AREP. Both matrices have order 8.

The mon-mon symmetry of the DCT-4_4 given in Example 9.8 has been used in [21] to derive a fast algorithm (the symmetry is stated in a different way), and, using Theorem 8.6 (iii), as fast algorithm for the DCT-2_8. The derivation essentially follows Figure 9.1, but the two monomial representations ϕ_1, ϕ_2 are decomposed over $\mathbb{Q}$. This concentrates all non-rational operations in the correction matrix D.

Remark. Using AREP we have verified (up to a certain size) that all 16 types of DTTs possess mon-mon symmetries for every size n.

10. Other Fast Algorithms. The algebraic methods presented in §8 and 9 explain most of the algorithms from the literature. There is one class of algorithms, however, that can not be explained by the methods presented so far. We will briefly discuss these algorithms to make this paper a comprehensive overview on DTT algorithms.

In short, it is possible to compute DTTs by embedding the transform matrix into a larger transform that can be computed efficiently. As an example we consider the first algorithm proposed for the DCT-$2_n = [\cos k(\ell+\frac{1}{2})\pi/n]$ [1]. If we define the DFT by

$$\mathrm{DFT}_n = [e^{2\pi ikl/n}]_{k,\ell=0\ldots n-1},$$

we can readily derive

$$\mathrm{re}\left(\mathrm{diag}_{k=0}^{2n-1}(e^{\pi ik/2n}) \cdot \mathrm{DFT}_{2n}\right) = [\cos k(\ell+\frac{1}{2})\pi/n]_{k,\ell=0\ldots 2n-1},$$

where $\mathrm{re}(M)$ denotes the real part of the matrix M. This shows that a DCT-2_n can be computed by padding an input vector x of length n with n zeros, followed by multiplying with a scaled DFT of size $2n$. The first n entries contain the result.

Similar constructions allow the computation of each DTT via a DFT of appropriate length. This shows that the arithmetic complexity of each DTT_n is $O(n \log n)$, independent of the size n. In particular, this includes the DTTs of type 5-8, for which no other algorithms exist in literature.

Embeddings into other transforms are also possible. For example, Theorem 8.7 allows to embed a DTT of type 5-8 into a DTT of type 1 or 2.

11. Summary. We gave a complete characterization of all 16 types of DTTs as scaled polynomial transforms corresponding to appropriate A-modules M with basis b, where $A = \mathbb{C}[x]/p(x)$, $M = f \cdot A$ with a scaling function f, and b is a sequence of Chebyshev polynomials (Theorem 6.2). Every DTT is uniquely determined by this algebraic property.

Then we used the algebraic characterization to derive by algebraic means most of the fast DTT algorithms known in the literature, and identified the mathematical principles behind each algorithm. In particular we derived

1. *Algorithms by direct manipulation/decomposition of M* (§8): (1) Translation between DTTs by duality (Theorem 8.2); (2) Translation between DTTs by base change (Theorem 8.5); (3) Decomposition by polynomial factorization (Theorems 8.6 and 8.7); (4) Decomposition by polynomial decomposition (Theorem 8.8).
2. *Algorithms by group symmetries* (§9): (1) Decomposition by mon-irred symmetry (Theorem 9.3); (2) decomposition by mon-mon symmetry (Theorem 9.5).
3. *Algorithms by embedding* (§10).

Our results show that the connection between digital signal processing and the representation theory of algebras goes clearly beyond the DFT. The question that remains is to what extent this connection can be extended to include other transforms and their fast algorithms and how this connection can be exploited for applications in signal processing. We want to conclude with this question: *To what extent is signal processing algebraic?*

Appendix. Orthonormal DCTs and DSTs.

Table A.1 gives the orthonormal versions of the 16 DTTs.

REFERENCES

[1] N. Ahmed, T. Natarajan, and K. R. Rao, *Discrete Cosine Transform*, IEEE Trans. on Computers, C-23 (1974), pp. 90–93.

[2] L. Auslander, E. Feig, and S. Winograd, *Abelian Semi-simple Algebras and Algorithms for the Discrete Fourier Transform*, Advances in Applied Mathematics, 5 (1984), pp. 31–55.

TABLE A.1

Definition of the orthonormal versions of the DCTs and DSTs; $a_{k,l}$ is the entry at row k and column l of the respective unscaled DTT as given in Table 5.1. All matrices have size $(n \times n)$ with row index $k = 0 \dots n-1$ and column index $\ell = 0 \dots n-1$. The row/column scaling factors are given by: $c_i = 1/\sqrt{2}$ for $i = 0$ and $= 1$ else; $d_i = 1/\sqrt{2}$ for $i = n-1$ and $= 1$ else.

	DCTs	DSTs
type 1	$\sqrt{\frac{2}{n-1}} \cdot c_k c_\ell d_k d_\ell \cdot a_{k,l}$	$\sqrt{\frac{2}{n+1}} \cdot a_{k,l}$
type 2	$\sqrt{\frac{2}{n}} \cdot c_k \cdot a_{k,l}$	$\sqrt{\frac{2}{n}} \cdot c_k \cdot a_{k,l}$
type 3	$\sqrt{\frac{2}{n}} \cdot c_\ell \cdot a_{k,l}$	$\sqrt{\frac{2}{n}} \cdot c_\ell \cdot a_{k,l}$
type 4	$\sqrt{\frac{2}{n}} \cdot a_{k,l}$	$\sqrt{\frac{2}{n}} \cdot a_{k,l}$
type 5	$\sqrt{\frac{2}{n-1/2}} \cdot c_k c_\ell \cdot a_{k,l}$	$\sqrt{\frac{2}{n+1/2}} \cdot a_{k,l}$
type 6	$\sqrt{\frac{2}{n-1/2}} \cdot c_k d_\ell \cdot a_{k,l}$	$\sqrt{\frac{2}{n+1/2}} \cdot a_{k,l}$
type 7	$\sqrt{\frac{2}{n-1/2}} \cdot d_k c_\ell \cdot a_{k,l}$	$\sqrt{\frac{2}{n+1/2}} \cdot a_{k,l}$
type 8	$\sqrt{\frac{2}{n+1/2}} \cdot a_{k,l}$	$\sqrt{\frac{2}{n-1/2}} \cdot d_k d_\ell \cdot a_{k,l}$

[3] T. BETH, *Verfahren der Schnellen Fouriertransformation*, Teubner, 1984.

[4] ———, *On the computational complexity of the general discrete Fourier transform*, Theoretical Computer Science, 51 (1987), pp. 331–339.

[5] P. BÜRGISSER, M. CLAUSEN, AND M. A. SHOKROLLAHI, *Algebraic Complexity Theory*, Springer, 1997.

[6] S. CHAN AND K. HO, *Direct Methods for computing discrete sinusoidal transforms*, IEE Proceedings, 137 (1990), pp. 433–442.

[7] W.-H. CHEN, C. SMITH, AND S. FRALICK, *A Fast Computational Algorithm for the Discrete Cosine Transform*, IEEE Trans. on Communications, COM-25 (1977), pp. 1004–1009.

[8] T. S. CHIHARA, *An Introduction to Orthogonal Polynomials*, Gordon and Breach, 1978.

[9] M. CLAUSEN, *Beiträge zum Entwurf schneller Spektraltransformationen (Habilitationsschrift)*, Univ. Karlsruhe, 1988.

[10] M. CLAUSEN AND U. BAUM, *Fast Fourier Transforms*, BI-Wiss.-Verl., 1993.

[11] J. W. COOLEY AND J. W. TUKEY, *An Algorithm for the Machine Calculation of Complex Fourier Series*, Math. of Computation, 19 (1965), pp. 297–301.

[12] W. C. CURTIS AND I. REINER, *Representation Theory of Finite Groups*, Interscience, 1962.

[13] P. DIACONIS AND D. ROCKMORE, *Efficient computation of the Fourier transform on finite groups*, Amer. Math. Soc., 3(2) (1990), pp. 297–332.

[14] J. R. DRISCOLL, M. HEALY JR., AND D. N. ROCKMORE, *Fast Discrete Polynomial Transforms with Applications to Data Analysis for Distance Transitive Graphs*, SIAM Journal Computation, 26 (1997), pp. 1066–1099.

[15] S. EGNER, *Zur Algorithmischen Zerlegungstheorie Linearer Transformationen mit Symmetrie*, PhD thesis, Universität Karlsruhe, Informatik, 1997.

[16] S. EGNER, J. JOHNSON, D. PADUA, M. PÜSCHEL, AND J. XIONG, *Automatic Derivation and Implementation of Signal Processing Algorithms*, ACM SIGSAM Bulletin Communications in Computer Algebra, 35 (2001), pp. 1–19.

[17] S. EGNER AND M. PÜSCHEL, *AREP—Constructive Representation Theory and Fast Signal Transforms*, GAP share package, 1998. `http://www.ece.cmu.edu/~smart/arep/arep.html`.

[18] ———, *Automatic Generation of Fast Discrete Signal Transforms*, IEEE Trans. on Signal Processing, 49 (2001), pp. 1992–2002.

[19] ———, *Symmetry-Based Matrix Factorization*, Journal of Symbolic Computation, (2002). To appear.

[20] E. FEIG, *A fast scaled-DCT algorithm*, in SPIE Proceedings, vol. 1244, 1990, pp. 2–13.

[21] E. FEIG AND S. WINOGRAD, *Fast Algorithms for the Discrete Cosine Transform*, IEEE Trans. on Signal Processing, 40 (1992), pp. 2174–2193.

[22] THE GAP TEAM, *GAP – Groups, Algorithms, and Programming*, University of St. Andrews,

Scotland, 1997. http://www-gap.dcs.st-and.ac.uk/∼gap/.
[23] M. Heideman, D. Johnson, and C. Burrus, *Gauss and the History of the Fast Fourier Transform*, Archive for History of Exact Sciences, 34 (1985), pp. 265–277.
[24] H. Hou, *A Fast Recursive Algorithm For Computing the Discrete Cosine Transform*, IEEE Trans. on Acoustics, Speech, and Signal Processing, ASSP-35 (1987), pp. 1455–1461.
[25] N. Jacobson, *Basic Algebra II*, W. H. Freeman and Co., 1980.
[26] T. Kailath and V. Olshevsky, *Displacement structure approach to discrete trigonometric transform based preconditioners of G.Strang and T.Chan type*, Calcolo, 33 (1996), pp. 191–208.
[27] H. Kitajima, *A Symmetric Cosine Transform*, IEEE Trans. on Computers, C-29 (1980), pp. 317–323.
[28] B. Lee, *A New Algorithm to Compute the Discrete Cosine Transform*, IEEE Trans. on Acoustics, Speech, and Signal Processing, ASSP-32 (1984), pp. 1243–1245.
[29] D. Maslen and D. Rockmore, *Generalized FFTs – A survey of some recent results*, in Proceedings of IMACS Workshop in Groups and Computation, vol. 28, 1995, pp. 182–238.
[30] J. C. Mason, *Chebyshev polynomials of the second, third and fourth kind in approximation, indefinite integration, and integral transforms*, Journal of Computational and Applied Mathematics, 49 (1993), pp. 169–178.
[31] T. Minkwitz, *Algorithmensynthese für lineare Systeme mit Symmetrie*, PhD thesis, Universität Karlsruhe, Informatik, 1993.
[32] ———, *Algorithms Explained by Symmetry*, Lecture Notes on Computer Science, 900 (1995), pp. 157–167.
[33] J. M. F. Moura and M. G. S. Bruno, *DCT/DST and Gauss-Markov Fields: Conditions for Equivalence*, IEEE Trans. on Signal Processing, 46 (1998), pp. 2571–2574.
[34] D. Potts and G. Steidl, *Optimal trigonometric preconditioners for nonsymmetric Toeplitz system*, Linear Algebra Applications, 281 (1998), pp. 265–292.
[35] D. Potts, G. Steidl, and M. Tasche, *Fast Algorithms for Discrete Polynomial Transforms*, Mathematics of Computation, 67 (1998), pp. 1577–1590.
[36] M. Püschel, *Konstruktive Darstellungstheorie und Algorithmengenerierung*, PhD thesis, Universität Karlsruhe, Informatik, 1998. Also available in English as Tech. Rep. Drexel-MCS-1999-1, Drexel University, Philadelphia.
[37] ———, *Decomposing Monomial Representations of Solvable Groups*, Journal of Symbolic Computation, 34 (2002), pp. 561–596.
[38] M. Püschel and J. M. F. Moura, *The Discrete Trigonometric Transforms and Their Fast Algorithms: An Algebraic Symmetry Approach*, in Proc. 10th IEEE DSP Workshop, 2002.
[39] C. M. Rader, *Discrete Fourier Transforms When the Number of Data Samples is Prime*, Proceedings of the IEEE, 56 (1968), pp. 1107–1108.
[40] K. R. Rao and P. Yip, *Discrete Cosine Transform: Algorithms, Advantages, Applications*, Academic Press, 1990.
[41] M. O. Rayes, V. Trevisan, and P. S. Wang, *Factorization of Chbeyshev Polynomials*, Tech. Report ICM-199802-0001, Kent State University, 1998.
[42] T. J. Rivlin, *The Chebyshev Polynomials*, Wiley Interscience, 1974.
[43] D. Rockmore, *Efficient computation of Fourier inversion for finite groups*, Assoc. Comp. Mach., 41(1) (1994), pp. 31–66.
[44] ———, *Some applications of generalized FFT's*, in Proceedings of DIMACS Workshop in Groups and Computation, vol. 28, 1995, pp. 329–370.
[45] V. Sánchez, P. García, A. M. Peinado, J. C. Segura, and A. J. Rubio, *Diagonalizing Properties of the Discrete Cosine Transforms*, IEEE Trans. on Signal Processing, 43 (1995), pp. 2631–2641.
[46] G. Steidl and M. Tasche, *A Polynomial Approach to Fast Algorithms for Discrete Fourier-Cosine and Fourier-Sine Transforms*, Mathematics of Computation, 56 (1991), pp. 281–296.
[47] G. Strang, *The Discrete Cosine Transform*, SIAM Review, 41 (1999), pp. 135–147.
[48] G. Szegö, *Orthogonal Polynomials*, Amer. Math. Soc. Colloq. Publ., 3rd ed., 1967.
[49] R. Tolimieri, M. An, and C. Lu, *Algorithms for Discrete Fourier Transforms and Convolution*, Springer, 2nd ed., 1997.
[50] M. Vetterli and H. Nussbaumer, *Simple FFT and DCT Algorithms with reduced Number of Operations*, Signal Processing, 6 (1984), pp. 267–278.
[51] Z. Wang, *Reconsideration of "A Fast Computational Algorithm for the Discrete Cosine Transform*, IEEE Trans. on Communications, COM-31 (1983), pp. 121–123.
[52] ———, *Fast Algorithms for the Discrete W Transform and for the Discrete Fourier Transform*, IEEE Trans. on Acoustics, Speech, and Signal Processing, ASSP-32 (1984), pp. 803–816.

[53] Z. Wang and B. Hunt, *The Discrete W Transform*, Applied Mathematics and Computation, 16 (1985), pp. 19–48.
[54] S. Winograd, *Arithmetic Complexity of Computation*, Siam, 1980.
[55] P. Yip and K. Rao, *A Fast Computational Algorithm for the Discrete Sine Transform*, IEEE Trans. on Communications, COM-28 (1980), pp. 304–307.
[56] ———, *Fast Decimation-In-Time Algorithms for a Family of Discrete Sine and Cosine Transforms*, Circuits, Systems, and Signal Processing, 3 (1984), pp. 387–408.
[57] ———, *The Decimation-In-Frequency Algorithms for a Family of Discrete Sine and Cosine Transforms*, Circuits, Systems, and Signal Processing, 7 (1988), pp. 3–19.