J. Symbolic Computation (2002) **11**, 1–000

Decomposing Monomial Representations of Solvable Groups

MARKUS PÜSCHEL[†]

Department of Electrical and Computer Engineering, Carnegie Mellon University

(Received 8 February 2002)

We present an efficient algorithm which decomposes a monomial representation of a solvable group G into its irreducible components. In contradistinction to other approaches we also compute the decomposition matrix A in the form of a product of highly structured, sparse matrices. This factorization can be viewed as a fast algorithm for the multiplication with A. In the special case of a regular representation we obtain a fast Fourier transform for G. Our algorithm is based on a *constructive* representation theory that we develop. The term "constructive" signifies that concrete matrix representations are considered and manipulated and not equivalence classes of representations as it is done in approaches that are based on characters. Thus, we present some well-known theorems in a constructively refined form and derive some new results on decomposition matrices. The decomposition algorithm has been implemented in the GAP share package AREP. One application of the algorithm is the automatic generation of fast algorithms for discrete linear signal transforms.

1. Introduction

The most important computational components in modern digital signal processing are discrete linear signal transforms, mathematically given by a multiplication of a vector (the sampled signal) by a matrix (the transform). A crucial property of linear signal transforms is the existence of fast algorithms, which reduce the asymptotic cost for computing the transform and thus make it useful for performance demanding signal processing applications.

The study of the structure and the derivation of these fast algorithms is the motivation for the work presented in this paper: The constructive decomposition of monomial group representations. To explain the connection between signal transforms and group representation theory, we have to go further back, starting with the discrete Fourier transform (DFT), the workhorse in signal processing, and arguably one of the most important tools used across scientific disciplines.

[†] This work was supported in part by NSF through award 9988296.

0747–7171/90/000000 + 00 \$03.00/0

1.1. Fourier Transforms

The (complex) DFT used in signal processing decomposes a given signal $x \in \mathbb{C}^n$ into a linear combination of exponential functions. The corresponding coefficients are obtained by multiplying x by the matrix

$$\mathrm{DFT}_n = [\omega_n^{k\ell} \mid k, \ell = 0 \ldots n-1], \quad \omega_n = e^{2\pi i/n}.$$

Algebraically, the DFT_n can be viewed as the decomposition of the group algebra $\mathbb{C}[\mathsf{Z}_n]$ of a cyclic group Z_n into simple algebras isomorphic to $\mathbb{C}$,

$$\mathrm{DFT}_n : \ \mathbb{C}[\mathsf{Z}_n] \to \mathbb{C} \oplus \ldots \oplus \mathbb{C},$$

with suitable bases chosen. Equivalently, the DFT_n can be interpreted as a decomposition matrix of a regular representation (a certain permutation representation) of Z_n. This decomposition is an instantiation of Wedderburn's theorem for semi-simple algebras. The algebraic interpretation provides deep insight into the DFT and has been used to derive and understand the structure of its fast algorithms. In particular, Auslander *et al.* (1984) and, independently, Beth (1984) showed that the most famous fast Fourier transform (FFT) algorithm, originally discovered by Gauß (Heideman *et al.* 1985), and rediscovered by Cooley and Tukey (1965), is obtained by a stepwise decomposition of $\mathbb{C}[\mathsf{Z}_n]$ along a composition series of Z_n. The stepwise decomposition determines a factorization of the DFT_n into a product of structured sparse matrices. This factorization is the fast algorithm. Using FFT algorithms, the DFT_n can be computed in $O(n \log(n))$ arithmetic operations compared to $O(n^2)$ required by direct evaluation.

Generalization to arbitrary finite, or even more general classes of groups G created a new area of research, referred to as "Fourier analysis on groups", that is concerned with fast algorithms for decomposing the group algebra $\mathbb{C}[G]$. In contrast to most of the classical work on group representations, e.g., Curtis and Reiner (1962), the algorithmic nature of this research makes it inherently "constructive", which means the choice of bases is crucial. In other words, concrete matrix representations are considered and not equivalence classes of matrix representations, for which characters are the appropriate objects to compute with.

Important work in the field of Fourier transforms on groups includes Beth (1984), Clausen (1988), Baum and Clausen (1994), Diaconis and Rockmore (1990), Rockmore (1990), and Maslen and Rockmore (2000). For an introduction to the area we refer the reader to the book of Clausen and Baum (1993) or the survey article by Maslen and Rockmore (1995). Applications of Fourier transforms on groups can be found in Rockmore (1995) or Terras (1999).

Unfortunately, most of the linear transforms used in signal processing can not be captured as Fourier transforms on groups.

1.2. Signal Transforms and Symmetry-Based Matrix Factorization

To extend the connection between signal transforms and group representation theory, it was necessary to leave the domain of regular representations. The "symmetry-based" matrix factorization introduced by Minkwitz (1993, 1995) and further developed by Egner and Püschel (2001, 2002) shows that decomposition matrices of non-regular permutation representations and, more general, monomial representations occur as signal transforms. The monomial representation associated with the transform is called "symmetry". A fast algorithm for the transform, given as sparse factorization, can be constructed, as in the case of the DFT, by a stepwise decomposition of this representation along a chain of normal subgroups (provided the group is solvable). Thus, the symmetry-based matrix factorization of a given matrix M consists of the following two high-level steps:

1. Find the symmetry of M.
2. Decompose the symmetry.

The first step is a combinatorial search problem and treated in Egner and Püschel (2002). The second step requires the stepwise decomposition of monomial representations and is subject of this paper. Both steps have been implemented in the GAP library AREP, and, taken together, provide a powerful tool to automatically factor a matrix into a product of sparse matrices, with potential applications beyond the domain of signal transforms.

1.3. Decomposing Monomial Representations

The application to signal transforms, as sketched above, was the original motivation for considering the problem solved in this paper:

> Given a monomial representation μ of a solvable group. We want to decompose μ into a direct sum of irreducible representations. *In parallel*, we want to compute the corresponding decomposition matrix *as a product of highly structured sparse matrices.*

Note that this problem embraces the construction of fast Fourier transforms arising from the special case of regular representations μ. Our approach has its root in Minkwitz (1993) and is a continuation of the work begun in the authors thesis (Püschel 1998).

In order to solve the problem we are required to deal with concrete *matrix representations* and not with *equivalence classes of matrix representations* as it is done in standard books. In this spirit we refine some well-known theorems and use them to derive results on the structure of decomposition matrices. The results are, as far as possible, presented as symbolic manipulations of structured representations and matrices—the necessary form for an implementation in a symbolic computation environment.

All results of this paper have been implemented in the GAP share package AREP (Egner and Püschel 1998) which provides the data types and infrastructure for symbolic computation with structured matrix representations and matrices and in particular includes the decomposition algorithm for monomial representations of solvable groups.

1.4. Organization

The paper is organized as follows. Section 2 introduces the notation and constructions from representation theory we are going to use. Section 3 is the mathematical foundation of this paper, which we refer to as "constructive representation theory" since we exclusively compute with matrix representations. First we present theorems that allow us to manipulate and compute with inductions of representations. Then we apply these theorems to monomial representations. After a short investigation of intertwining spaces we present the main mathematical results of this paper on the structure of decomposition matrices. These results form the basis for the decomposition algorithm for monomial representations of solvable groups, which is presented in detail in Section 4 including an example and some runtime measurements. The impatient reader can find a sketched version of the decomposition algorithm on page 26. The paper is concluded by Section 5 where we very briefly survey the library AREP, which contains an implementation of the decomposition algorithm.

2. Notation

A *representation* ϕ of degree n over a field $\mathbb{K}$ is a homomorphism of a group G into the group $\mathsf{GL}_n(\mathbb{K})$ of invertible $(n \times n)$-matrices over $\mathbb{K}$. Throughout the paper the group G is finite, $\mathbb{K}$ is a *splitting field* for ϕ (which is guaranteed, if G contains an eth root of unity, where e is the exponent of G), and the characteristic of $\mathbb{K}$ does not divide the group order $|G|$ (Maschke condition). In this case, every representation ϕ can be decomposed with an invertible matrix $A \in \mathsf{GL}_n(\mathbb{K})$ into a direct sum of *irreducible* representations ρ_i (Maschke's and Wedderburn's theorem):

$$\phi^A = (g \mapsto A^{-1} \cdot \phi(g) \cdot A) = \bigoplus_{i=1}^{r} (\underbrace{\rho_i \oplus \ldots \oplus \rho_i}_{n_i}), \quad \text{where } \rho_i \not\cong \rho_j \text{ for } i \neq j.$$

Every ρ_i is called an irreducible component of ϕ, and every $\rho_i \oplus \ldots \oplus \rho_i$ is called a *homogeneous* (or *isotypic*) component of ϕ. We say that ρ_i has multiplicity n_i in ϕ. A representation is called a *permutation representation* if all images are permutation matrices. A representation is called *monomial* if all images are monomial matrices. A matrix is called monomial if it contains exactly one nonzero entry in every row and column. Monomial matrices are always invertible.

If ϕ is a representation over $\mathbb{K}$ of degree n, then G operates on the vector space $V = \mathbb{K}^n$ via ϕ by $v \cdot g = v \cdot \phi(g),\ v \in V,\ g \in G$, making V a right $\mathbb{K}[G]$-module. We call V the *representation space* of ϕ. The decomposition of a representation ϕ into irreducible or homogeneous components corresponds to the decomposition of the representation space V into irreducible or homogeneous components, respectively. Whenever possible, we present results in terms of matrix representations, avoiding the terminology of modules. We use the following conventions for notation.

Matrices. Matrices are denoted by letters $A, B, M, P, \ldots$. A matrix with entries $a_{i,j}$ is written as $[a_{i,j} \mid i = 1 \ldots n,\ j = 1 \ldots m]$ or more simply as $[a_{i,j}]_{i,j}$. A diagonal matrix is written as $\mathrm{diag}(x_1, \ldots, x_n)$, a permutation matrix as $[\sigma, n] = [\delta_{i^\sigma j} \mid i, j = 1 \ldots n]$, where σ is a permutation and n the matrix size. $[\sigma, (x_1, \ldots, x_n)] = [\sigma, n] \cdot \mathrm{diag}(x_1, \ldots, x_n)$ is used to represent a monomial matrix. A primitive nth root of unity is denoted by ω_n, $\mathbf{1}_n$ is the identity matrix of degree n, $\mathbf{0}_n$ the (square) all-zero matrix of degree n and $\mathrm{DFT}_n = [\omega_n^{ij} \mid i, j = 0 \ldots n-1]$ the *discrete Fourier transform* of degree n. The direct sum of matrices A, B is written as $A \oplus B$ and the tensor or Kronecker product as $A \otimes B$ (A determines the coarse structure). A matrix M is called *block-permuted*, if $M = P \cdot (B_1 \oplus \ldots \oplus B_k) \cdot Q$ with permutation matrices P and Q.

Sets and Lists. A set is written in the usual way as $\{t_1, \ldots, t_n\}$ and a list (i.e., an ordered set) as $(t_1, \ldots, t_n)$. Correspondingly, we denote with "$\cup$" the union of sets or the concatenation of lists, respectively.

Groups. Groups are denoted by letters $G, H, N, K, \ldots$. The set of right cosets of H in G is written as $H \backslash G$. If $H \trianglelefteq G$ is a normal subgroup we also write G/H. *Transversals* (systems of right coset representatives) are denoted by $T, S, \ldots$ and are lists. Group elements are written by lower case letters $g, h, x, y, s, t, \ldots$, $\mathsf{E} = \{1\}$ denotes the trivial group, and Z_n a cyclic group of order n.

Representations. Representations are denoted by lower case Greek letters $\phi, \psi, \rho, \ldots$; μ is a monomial representation and λ is a representation of degree 1. Sometimes we put the group represented in the subscript as in ϕ_G. We let $1_G : \ g \mapsto 1$ denote the trivial representation (of degree 1) of G. The degree of ϕ is denoted by $\deg(\phi)$ and the character of ϕ is denoted as χ_ϕ.

Constructions for representations. We use the following set of constructions for representations. Let ϕ_G be a representation of G and $A \in \mathsf{GL}_n(\mathbb{K})$. Then the *conjugated representation* is defined by $\phi_G^A = g \mapsto A^{-1} \cdot \phi_G(g) \cdot A$. Equivalently, we often write

$$\phi_G \xrightarrow{A} \phi_G^A.$$

We denote the *direct sum* of representations ϕ_G, ψ_G by $\phi_G \oplus \psi_G = g \mapsto \phi_G(g) \oplus \psi_G(g)$. In particular we write $\phi_G^n = \phi_G \oplus \ldots \oplus \phi_G$ (n summands). The *inner tensor product* of representations ϕ_G, ψ_G of the same group G is denoted by $\phi_G \otimes \psi_G = g \mapsto \phi_G(g) \otimes \psi_G(g)$ and is again a representation of G. In contrast, the *outer tensor product* of representations ϕ_G of G and ψ_H of H is written as $\phi_G \# \psi_H = (g, h) \mapsto \phi_G(g) \otimes \psi_H(h)$ and is a representation of the direct product $G \times H$. The *linear multiple* of ϕ_G with a representation λ_G of degree 1 is written as $\lambda_G \cdot \phi_G = g \mapsto \lambda_G(g) \cdot \phi_G(g)$. It is a special case of an inner tensor product. The *restriction* of ϕ_G to a subgroup $H \leq G$ is denoted by $\phi_G \downarrow H = h \mapsto \phi_G(h)$. The *extension* of a representation ϕ_H to a supergroup $G \geq H$ is denoted by $\overline{\phi}_H$. Note that the extension of a representation does not exist in general.

We define the *inner conjugate* of a representation ϕ_H with an element $t \in G$ of a supergroup $G \geq H$ by $\phi_H^t = g \mapsto \phi_H(tgt^{-1})$. ϕ_H^t is a representation of the conjugated subgroup $H^t = t^{-1}Ht$. If in particular H is normal in G then the inner conjugate of any representation of H is again a representation of H, however, in general, not equivalent to the original one. The definition of the inner conjugate implies the rule

$$\left(\phi_H^t\right)^s = \phi_H^{ts} = g \mapsto \phi_H(tsgs^{-1}t^{-1}),$$

i.e., g *first* is conjugated by the inverse of the *outer* exponent.

The most important construction in this paper is the *induction*. Let $H \leq G$ be a subgroup with representation ϕ_H and $T = (t_1, \ldots, t_n)$ a transversal of $H \backslash G$ of length $n = (G : H)$. Then the induction of ϕ_H to G with transversal T is defined by

$$\begin{aligned} \phi_H \uparrow_T G \quad &= \quad g \mapsto \left[\dot{\phi}_H(t_i \cdot g \cdot t_j^{-1}) \mid i, j \in \{1, \ldots, n\}\right], \quad \text{with} \\ & \dot{\phi}_H(x) = \begin{cases} \phi_H(x), & x \in H \\ \mathbf{0}_{\deg(\phi_H)}, & \text{else} \end{cases} . \end{aligned}$$

Since the equivalence class of $\phi_H \uparrow_T G$ is independent of the choice of the transversal T we omit T when calculating only up to equivalence. If in particular ϕ_H is of degree 1, then the induction is monomial. If even $\phi_H = 1_H$, then the induction is a permutation representation. A *regular representation* of a group is the special case of a permutation representation given by any induction $1_{\mathsf{E}} \uparrow G$.

3. Constructive Representation Theory

This section contains the mathematical foundation of the decomposition algorithm. Section 3.1 presents theorems on the interaction of induction and other representation theoretic constructions (direct sum, conjugation etc.). This section serves as a "toolkit" throughout this paper. In Section 3.2 we investigate monomial representations, showing that they are essentially direct sum of inductions which allows us to apply the former results. After the short Section 3.3 on intertwining spaces, we derive formulas for decomposition matrices in Section 3.4, which contains the most important mathematical results in this paper. Note that most of the results in this section are presented as symbolic manipulation of structured representations and matrices. In this form they can readily be implemented and serve as building blocks for doing symbolic computation with representations. We refer to Section 5 for the actual implementation in the library AREP.

3.1. Induction

In this section we present theorems that explain the interaction of the induction with other constructions for representations (cf. Section 2).

3.1.1. Change of Transversal

It is known that the equivalence class of an induction of a representation ϕ of $H \leq G$ is independent of the choice of transversal, i.e.,

$$\phi \uparrow_T G \cong \phi \uparrow_{T'} G.$$

We determine the conjugating matrix, corresponding to the pair (T, T'), which establishes equality. Let ϕ be a representation of H, $n = (G : H)$ and $T = (t_1, \ldots, t_n)$ an arbitrary transversal of $H \backslash G$. First we consider two particular cases of a change of transversal.

Change of coset representatives in T leads to the transversal

$$T' = (h_1 t_1, \ldots, h_n t_n),\ h_i \in H,$$

and has the following effect on the induction:

$$(\phi \uparrow_{T'} G)(x) = \left[\dot{\phi}(t'_i x {t'_j}^{-1})\right]_{i,j} = \left[\dot{\phi}(h_i t_i x t_j^{-1} h_j^{-1})\right]_{i,j} = \left[\dot{\phi}(t_i x {t_j}^{-1})\right]_{i,j}^{D},$$

where $D = \bigoplus_{i=1}^n \phi(h_i^{-1})$ is a block diagonal matrix with blocks of size $\deg(\phi)$.

Permutation of T with $\sigma \in \mathsf{S}_n$ leads to the transversal

$$T' = T^\sigma = (t_{1^{\sigma^{-1}}}, \ldots, t_{n^{\sigma^{-1}}})$$

and the induction with T' can be calculated as

$$\begin{aligned}
(\phi \uparrow_{T'} G)(x) &= \left[\dot{\phi}(t'_i x {t'_j}^{-1})\right]_{i,j} \\
&= \left[\dot{\phi}(t_{i^{\sigma^{-1}}} x t_{j^{\sigma^{-1}}}^{-1})\right]_{i,j} \\
&= \left([\sigma^{-1}, n] \otimes \mathbf{1}_{\deg(\phi)}\right) \cdot \left[\dot{\phi}(t_i x t_j^{-1})\right]_{i,j} \cdot \left([\sigma, n] \otimes \mathbf{1}_{\deg(\phi)}\right) \\
&= (\phi \uparrow_T G)(x)^{[\sigma,n] \otimes \mathbf{1}_{\deg(\phi)}}.
\end{aligned}$$

The general case of change of transversal can be composed from these two particular cases.

Theorem 3.1 *Let $H \leq G$ be a subgroup and ϕ a representation of H and let $T = (t_1, \ldots, t_n)$ and $T' = (t'_1, \ldots, t'_n)$ be two transversals of $H\backslash G$. Assume that σ is the permutation in S_n mapping the cosets $(Ht_1, \ldots, Ht_n)$ on the cosets $(Ht'_1, \ldots, Ht'_n)$. Then*

$$(\phi \uparrow_T G)^M = (\phi \uparrow_{T'} G), \quad \textit{with } M = ([\sigma, n] \otimes \mathbf{1}_{\deg(\phi)}) \cdot \bigoplus_{i=1}^{n} \phi\left(t_{i^{\sigma^{-1}}} \cdot {t'}_i^{-1}\right).$$

We call M the matrix corresponding to the change of transversal $T \to T'$.

Proof. We have $\phi \uparrow_{T^\sigma} G = (\phi \uparrow_T G)^{[\sigma,n]\otimes \mathbf{1}_{\deg(\phi)}}$ according to the calculation above. The transition from T^σ to T' is only a change of coset representatives and hence $\phi \uparrow_{T'} G = (\phi \uparrow_{T^\sigma} G)^D$ with $D = \bigoplus_{i=1}^n \phi(t_{i^{\sigma^{-1}}} \cdot {t'}_i^{-1})$ and the result follows. □

The change of transversal is the most important basic routine for the computation with inductions. The following theorems explain the interaction of the induction with other operations. In most of the cases equality is obtained by choosing a specific transversal. This makes Theorem 3.1 a central tool for manipulating inductions.

3.1.2. Double Induction

Induction is a transitive operation. If ϕ is a representation of H and $H \leq K \leq G$ then

$$\phi \uparrow G \cong (\phi \uparrow K) \uparrow G.$$

Equality is established by appropriate choice of the transversal.

Theorem 3.2 *Let $H \leq K \leq G$ be groups and ϕ a representation of H. Suppose $T = (t_1, \ldots, t_n)$ and $S = (s_1, \ldots, s_m)$ are transversals of $H\backslash K$ and $K\backslash G$ respectively. Then*

$$\phi \uparrow_{TS} G = (\phi \uparrow_T K) \uparrow_S G,$$

where $TS = (t_1s_1, \ldots, t_ns_1, t_1s_2, \ldots, t_ns_2, \ldots, t_1s_m, \ldots, t_ns_m)$ denotes the complex product of the transversals S and T.

Proof. TS is a transversal of $H\backslash G$ and

$$\begin{aligned}
(\phi \uparrow_T K) \uparrow_S G &= \left(x \mapsto \left[\dot{\phi}(t_i x t_j^{-1})\right]_{i,j}\right) \uparrow_S G \\
&= x \mapsto \left[\dot{\phi}(t_i s_k x s_\ell^{-1} t_j^{-1})\right]_{(k,i),(\ell,j)} \\
&= \phi \uparrow_{TS} G,
\end{aligned}$$

which proves the result. □

The previous theorem allows us to decompose a given induction into small steps along a chain of subgroups by switching to a suitable transversal using Theorem 3.1.

3.1.3. Direct Sum

Induction is additive, i.e., if ϕ_1 and ϕ_2, are representations of $H \leq G$ then

$$(\phi_1 \oplus \phi_2) \uparrow G \cong (\phi_1 \uparrow G) \oplus (\phi_2 \uparrow G).$$

Equality can is achieved by a permutation matrix.

Theorem 3.3 *Let $H \leq G$ be a subgroup with representations ϕ_1 and ϕ_2 of degrees d_1 and d_2 respectively. Assume T is a transversal of $H\backslash G$ of length n. For brevity let $d = d_1 + d_2$. Denote with σ the permutation mapping the list*

$$\bigcup_{k=0}^{n-1} (k \cdot d + 1, \ldots, k \cdot d + d_1) \;\cup\; \bigcup_{k=0}^{n-1} (k \cdot d + d_1 + 1, \ldots, (k+1) \cdot d)$$

elementwise onto $(1, \ldots, n \cdot d)$. Then

$$((\phi_1 \oplus \phi_2) \uparrow_T G)^{[\sigma, n \cdot d]} = (\phi_1 \uparrow_T G) \oplus (\phi_2 \uparrow_T G).$$

Proof. The induced representation is of degree $n \cdot d$. The first concatenation corresponds to the indices of the basis vectors of the representation space of $\phi_1 \uparrow G$ in $(\phi_1 \oplus \phi_2) \uparrow G$, and similarly for the second concatenation. The corresponding change of basis decomposes the representation. $\square$

3.1.4. Conjugation

Induction of equivalent representations ϕ and $\psi = \phi^A$ leads to equivalent representations. The conjugating matrix can be stated explicitly.

Theorem 3.4 *Let $H \leq G$ be a subgroup of index n with representation ϕ over $\mathbb{K}$ of degree d and T a transversal of $H\backslash G$. Assume $A \in \mathsf{GL}_d(\mathbb{K})$, then*

$$(\phi^A \uparrow_T G) = (\phi \uparrow_T G)^{(\mathbf{1}_n \otimes A)}.$$

Proof. For $x \in G$ we have

$$(\phi^A \uparrow_T G)(x) \;=\; \left[\dot{\phi^A}(t_i x t_j^{-1})\right]_{i,j} = \left[A^{-1} \cdot \dot{\phi}(t_i x t_j^{-1}) \cdot A\right]_{i,j} = (\phi \uparrow_T G)(x)^{(\mathbf{1}_d \otimes A)},$$

as desired. $\square$

In the case in which A is a decomposition matrix for ϕ, we can apply Theorem 3.3 to compute a permutation matrix P such that $(\mathbf{1}_d \otimes A) \cdot P$ decomposes $\phi \uparrow G$ into a direct sum (however, in general not into irreducibles).

3.1.5. Outer Tensor Product

Assume G_1, G_2 are groups and $H_1 \leq G_1$, $H_2 \leq G_2$ subgroups with representations ϕ_1, ϕ_2. A well-known theorem (e.g., Curtis and Reiner (1962), p. 316) states

$$(\phi_1 \uparrow G_1) \,\#\, (\phi_2 \uparrow G_2) \cong (\phi_1 \,\#\, \phi_2) \uparrow (G_1 \times G_2).$$

As before we obtain equality by an appropriate choice of a transversal. The following theorem is the basis for the constructive decomposition of a monomial representation into an outer tensor product.

Theorem 3.5 *Let $H_1 \leq G_1$ and $H_2 \leq G_2$ be subgroups with representations ϕ_1 and ϕ_2. Assume $T_1 = (t_1^{(1)}, \ldots, t_n^{(1)})$ and $T_2 = (t_1^{(2)}, \ldots, t_m^{(2)})$ are transversals of $H_1\backslash G_1$ and $H_2\backslash G_2$ respectively. Then*

$$(\phi_1 \uparrow_{T_1} G_1) \,\#\, (\phi_2 \uparrow_{T_2} G_2) = (\phi_1 \,\#\, \phi_2) \uparrow_{T_1 \times T_2} (G_1 \times G_2),$$

where $T_1 \times T_2 = \left((t_1^{(1)}, t_1^{(2)}), (t_1^{(1)}, t_2^{(2)}), \ldots\right)$ *denotes the Cartesian product of the lists* T_1 *and* T_2.

PROOF. $T_1 \times T_2$ is a transversal of $(H_1 \times H_2)\backslash(G_1 \times G_2)$ and

$$\begin{aligned}
& ((\phi_1 \uparrow_{T_1} G_1) \#(\phi_2 \uparrow_{T_2} G_2))\,(x_1, x_2) \\
= & \left[\dot{\phi}_1(t_i^{(1)} x_1 {t_j^{(1)}}^{-1})\right]_{i,j} \otimes \left[\dot{\phi}_2(t_k^{(2)} x_2 {t_\ell^{(2)}}^{-1})\right]_{k,\ell} \\
= & \left[(\phi_1 \mathbin{\dot{\#}} \phi_2)\left((t_i^{(1)} x_1 {t_j^{(1)}}^{-1}), (t_k^{(2)} x_2 {t_\ell^{(2)}}^{-1})\right)\right]_{(i,k),(j,\ell)} \\
= & ((\phi_1 \# \phi_2) \uparrow_{T_1 \times T_2} (G_1 \times G_2))\,(x_1, x_2),
\end{aligned}$$

as desired. □

If the products $H_1 \times H_2$ and $G_1 \times G_2$ in Theorem 3.5 are inner direct products then $T_1 \times T_2 = T_1 T_2$ is just the pointwise product of the transversals. In order to decompose an induction into an outer tensor product, however, it is not sufficient that the represented group is a direct product. In Section 3.2 we present a necessary and sufficient criterion for the existence of such a decomposition in the particular case of a monomial representation.

3.1.6. Inner Conjugation

The induction of a representation ϕ of H to a supergroup G can be expressed as an induction of an inner conjugate ϕ^s which is a representation of H^s.

Theorem 3.6 *Let* $H \leq G$ *be a subgroup,* $s \in G$, ϕ *a representation of* H, *and* T *a transversal of* $H\backslash G$. *Then*

$$\phi \uparrow_T G = \phi^s \uparrow_{s^{-1}T} G.$$

PROOF. Let $T = (t_1, \ldots, t_n)$ and $x \in G$. Then

$$(\phi \uparrow_T G)(x) = \left[\dot{\phi}(t_i x t_j^{-1})\right]_{i,j} = \left[\dot{\phi^s}(s^{-1} t_i x t_j^{-1} s)\right]_{i,j} = (\phi^s \uparrow_{s^{-1}T} G)(x).$$

Note that $s^{-1}T$ is a transversal of $H^s\backslash G$. □

In particular we have $\phi \uparrow G \cong \phi^s \uparrow G$. Next, we consider the inner conjugate of an induction.

Theorem 3.7 *Let* $H \leq K \leq G$ *be subgroups,* ϕ *a representation of* H, T *a transversal of* $H\backslash K$, *and* $s \in G$. *Then*

$$(\phi \uparrow_T K)^s = \phi^s \uparrow_{T^s} K^s,$$

where $T^s = (s^{-1}t_1 s, \ldots, s^{-1}t_n s)$ *denotes the element-wise conjugate of the transversal* $T = (t_1, \ldots, t_n)$ *by* s.

PROOF. With $x \in K^s$ we observe

$$\begin{aligned}
(\phi \uparrow_T K)^s(x) &= (\phi \uparrow_T K)(sxs^{-1}) = \left[\dot{\phi}(t_i s x s^{-1} t_j^{-1})\right]_{i,j} = \left[\dot{\phi^s}(t_i^s x (t_j^s)^{-1})\right]_{i,j} \\
&= (\phi^s \uparrow_{T^s} K^s)(x)
\end{aligned}$$

as desired. □

3.1.7. Restriction

Mackey's subgroup theorem Curtis and Reiner (1962), p. 324 provides the partial decomposition of the restriction of an induction to an arbitrary subgroup. Assume $H, K \leq G$ are subgroups and ϕ is a representation of H. Then

$$(\phi \uparrow G) \downarrow K \cong \bigoplus_{s \in S} (\phi^s \downarrow (H^s \cap K)) \uparrow K$$

where S is a system of representatives of the double cosets $H \backslash G / K = \{HgK \mid g \in G\}$ We give transversals for the inductions in order to establish equality.

Theorem 3.8 (Mackey) *Let $H, K \leq G$ be subgroups, ϕ a representation of H and $S = (s_1, \ldots, s_n)$ a system of representatives of the double cosets $H \backslash G / K$. Assume $T_i = (t_{i,1}, \ldots, t_{i,r_i})$, $i = 1 \ldots n$ are transversals of $(H^{s_i} \cap K) \backslash K$. Then the concatenation*

$$T = \bigcup_{i=1}^{n} s_i T_i \text{ is a transversal of } H \backslash G$$

and

$$(\phi \uparrow_T G) \downarrow K = \bigoplus_{i=1}^{n} (\phi^{s_i} \downarrow (H^{s_i} \cap K)) \uparrow_{T_i} K.$$

Proof. First we show that T is a transversal. T has the right length since $(G : H) = \sum_{i=1}^{n} (K : (H^{s_i} \cap K))$. Assume further that $x, y \in T, x \neq y$. Then one of the following two cases applies. (1) $x, y \in T_i$ for a suitable i, and hence x, y are of the form $x = s_i t_{i,j},\ y = s_i t_{i,k},\ j \neq k$, and we get

$$xy^{-1} \in H \Leftrightarrow s_i t_{i,j} {t_{i,k}}^{-1} s_i^{-1} \in H \Leftrightarrow t_{i,j} {t_{i,k}}^{-1} \in H^{s_i}$$

which contradicts the fact that T_i is a transversal of $(H^{s_i} \cap K) \backslash K$. (2) x, y are of the form $x = s_i t_{i,k},\ y = s_j t_{j,\ell}$, with $i \neq j$. We get

$$xy^{-1} \in H \Leftrightarrow s_i t_{i,k} {t_{j,\ell}}^{-1} s_j^{-1} \in H \Leftrightarrow H s_i \underbrace{t_{i,k} {t_{j,\ell}}^{-1}}_{\in K} = H s_j,$$

which contradicts the fact that s_i, s_j are elements of different double cosets.

To proof the second assertion let $x \in K$. We derive

$$\begin{aligned}
(\phi \uparrow_T G)(x) &= \left[\dot{\phi}(t_i x t_j^{-1}) \mid i, j \in \{1, \ldots, \sum_{k=1}^{n} r_k\} \right] \\
&= \bigoplus_{k=1}^{n} \left[\dot{\phi}(s_k t_{k,i} x {t_{k,j}}^{-1} s_k^{-1}) \mid i, j \in \{1, \ldots, r_k\} \right] \\
&= \bigoplus_{k=1}^{n} \left[\dot{\phi^{s_k}}(t_{k,i} x {t_{k,j}}^{-1}) \mid i, j \in \{1, \ldots, r_k\} \right] \\
&= \bigoplus_{k=1}^{n} ((\phi^{s_k} \downarrow (H^{s_k} \cap K)) \uparrow_{T_k} K)(x).
\end{aligned}$$

This completes the proof. □

In the particular case that $\phi = 1_H$ is the trivial representation, i.e., $\phi \uparrow G$ is a permutation representation, Mackey's theorem yields exactly the decomposition of $(1_H \uparrow G) \downarrow K$ into its transitive constituents. The following two cases will be used later.

Corollary 3.9 *If $N \trianglelefteq G$, ϕ a representation of N, and T a transversal of G/N then*

$$(\phi \uparrow_T G) \downarrow N = \bigoplus_{t \in T} \phi^t.$$

Proof. This follows from applying Theorem 3.8 using $N \backslash G / N = G/N$. □

Corollary 3.10 *Let $H \leq G$, $N \trianglelefteq G$ with $HN = G$, ϕ a representation of H and T a transversal of $(N \cap H) \backslash N$. Then T is also a transversal of $H \backslash G$ and*

$$(\phi \uparrow_T G) \downarrow N = (\phi \downarrow (N \cap H)) \uparrow_T N.$$

Proof. Since $H \backslash G / N = HN \backslash G = G \backslash G$ there is only one double coset with representative 1. The rest follows from Theorem 3.8. □

The following theorem computes the induction of a restriction.

Theorem 3.11 *Let $H \leq G$ be a subgroup, ϕ a representation of G and $T = (t_1, \ldots, t_n)$ a transversal of $H \backslash G$. Then*

$$((\phi \downarrow H) \uparrow_T G)^D = (1_H \uparrow_T G) \otimes \phi, \text{ with } D = \bigoplus_{t \in T} \phi(t).$$

Proof. For $x \in G$ we have

$$\begin{aligned} ((\phi \downarrow H) \uparrow_T G)^D (x) &= \left[\phi(t_i)^{-1} (\phi \dot{\downarrow} H)(t_i x t_j^{-1}) \phi(t_j)\right]_{i,j} \\ &= \left[\dot{1}_H (t_i x t_j^{-1}) \cdot \phi(x)\right]_{i,j} \\ &= (1_H \uparrow_T G)(x) \otimes \phi(x). \end{aligned}$$

Notice that in the second equality the block $\phi(t_i)^{-1} \cdot (\phi \dot{\downarrow} H)(t_i x t_j^{-1}) \cdot \phi(t_j)$ is equal to $\phi(x)$ if $t_i x t_j^{-1} \in H$, and else equal to the all-zero matrix. □

3.1.8. Kernel

The kernel of an induction can be computed as follows.

Theorem 3.12 *Let $H \leq G$ be a subgroup with representation ϕ. Then the kernel of the induction,* $\ker(\phi \uparrow G)$, *does not depend on the transversal and can be computed as*

$$\ker(\phi \uparrow G) = \mathrm{core}_G(\ker(\phi)),$$

where $\mathrm{core}_G(K) = \bigcap_{g \in G} K^g$, *which is called* the core of K in G, *denotes the largest normal subgroup of G contained in $K \leq G$. In particular we have* $\ker(1_H \uparrow G) = \mathrm{core}(H)$.

Proof. We choose an arbitrary transversal T of $H\backslash G$ and set $d = \deg(\phi)$. Then $x \in \ker(\phi \uparrow_T G) \Leftrightarrow x^{t^{-1}} \in H$ and $\phi(x^{t^{-1}}) = \mathbf{1}_d$ for all $t \in T \Leftrightarrow x^{g^{-1}} \in H$ and $\phi(x^{g^{-1}}) = \mathbf{1}_d$ for all $g \in G \Leftrightarrow x \in \ker(\phi)^g$ for all $g \in G$, as desired. □

3.2. Monomial Representations

Monomial representations are a natural generalization of permutation representations. A representation $\phi: \ G \to \mathsf{GL}_n(\mathbb{K})$ of a group G is called monomial if every image $\phi(g),\ g \in G$ is a monomial matrix, i.e., $\phi(g)$ contains in every row and column exactly one nonzero entry. While the set of all permutation matrices in $\mathsf{GL}_n(\mathbb{K})$ is finite (of size $n!$) the same does not hold any longer for the set of monomial matrices (if $|\mathbb{K}| = \infty$), not even for the subset of those of finite order.

Note that questions concerning monomial representations cannot easily be reduced to permutation representations. E.g., monomial representations of degree > 1 can be irreducible whereas the same is impossible for permutation representations. Furthermore, there is a class of groups (so-called M-groups, Curtis and Reiner (1962), pp. 357) with the property that every representation is equivalent to a monomial one. One example for M-groups is given by supersolvable groups. G is supersolvable if it has a composition series that is even a chief series. This property can be exploited to very efficiently construct a complete set of monomial irreducible representations and a fast Fourier transform for G (Baum and Clausen 1994).

In the following we present the constructive results concerning monomial representations which we need for their decomposition. We show that a monomial representation essentially is a direct sum of induction of representations of degree 1 (cf. Theorem 3.15 and Theorem 3.16), hence the results on inductions in the last section become applicable.

First we generalize some notions concerning permutation representations to monomial representations. For this purpose we associate with every monomial representation μ a unique permutation representation in the following way.

Definition 3.13 Let μ be a monomial representation. The *underlying permutation representation* of μ, denoted by $\hat{\mu}$, is obtained by substituting all nonzero entries by 1 in the images of μ.

Using $\hat{\mu}$ allows us to transfer many concepts for permutation representations to monomial representations. Standard books on permutation representations and permutation groups include Wielandt (1964) and Dixon and Mortimer (1996).

Definition 3.14 A monomial representation μ of a group G is called transitive, if $\hat{\mu}$ is transitive. The orbits of μ on $\{1, \ldots, \deg(\mu)\}$ are defined as the orbits of $\hat{\mu}$ on this set. The stabilizer $\mathrm{stab}_\mu(i)$ of a point i under μ is the stabilizer of i under $\hat{\mu}$.

3.2.1. Orbit Decomposition

Analogous to a permutation representation also a monomial representation can be decomposed by a permutation into its transitive constituents corresponding to its orbits. For a decomposition into irreducible representations we can thus restrict ourselves to the transitive case which is investigated in the next paragraphs. We remind the reader that $\phi \xrightarrow{A} \psi$ signifies $\phi^A = \psi$. The following theorem is immediate.

Theorem 3.15 *Let μ be a monomial representation of degree n of a group G. Assume the orbits of μ on $\{1,\ldots,n\}$ are given as the lists $O_1,\ldots,O_k$. Suppose σ is the permutation mapping $L = (\ell_1,\ldots,\ell_n) = O_1 \cup \cdots \cup O_k$ onto $(1,\ldots,n)$, i.e., $\ell_i^\sigma = i,\ i = 1\ldots n$. Then*

$$\mu \xrightarrow{[\sigma,n]} \bigoplus_{i=1}^{k} \mu_i,$$

where μ_i are transitive monomial representations.

3.2.2. Decomposition into an Induction

Every transitive monomial representation is equivalent to an induction of a representation λ of degree 1 of a subgroup H. We present a constructive proof.

Theorem 3.16 *Let μ be a transitive monomial representation of a group G over a field $\mathbb{K}$ with representation space $V = \langle v_1,\ldots,v_n\rangle$, i.e.,*

$$v_i \cdot \mu(g) = v_{i^{\hat{\mu}(g)}} \cdot a_i(g)$$

where $a_i(g) \in \mathbb{K}$ for all $g \in G$. Assume $H = \mathrm{stab}_\mu(1)$ and $T = (t_1,\ldots,t_n)$ is a transversal of H in G. Then there exists a representation λ of H of degree 1 such that

$$\mu \xrightarrow{D} \lambda \uparrow_T G, \quad \textit{where } D = \mathrm{diag}(a_1(t_i)^{-1} \mid i = 1\ldots n).$$

Proof. Assume $H = \mathrm{stab}_\mu(1)$ denotes the stabilizer of 1 under μ. Since μ is transitive we have $(G:H) = \deg(\mu) = n$. Let $T = (t_1,\ldots,t_n)$ be a transversal of $H\backslash G$ with $1^{\hat{\mu}(t_i)} = i$. For $h \in H$ we observe $v_1\mu(h) = v_1 a_1(h)$ and define by $\lambda:\ h \mapsto a_1(h)$ a representation λ of H of degree 1. The representation space of the induced representation then is given by $V^G = \langle v_1 \otimes t_i \mid i = 1\ldots n\rangle$. Setting $t_i g = h_i t_{i'},\ h_i \in H$ we deduce

$$(v_1 \otimes t_i)(\lambda \uparrow_T G)(g) = v_1\lambda(h_i) \otimes t_{i'} = (v_1 \otimes t_{i'})a_1(h_i).$$

We define $w_i = v_1\mu(t_i) = v_i a_1(t_i)$ for $i = 1\ldots n$ and calculate

$$w_i \cdot \mu(g) = v_1 \cdot \mu(t_i g) = v_1 \cdot \mu(h_i t_{i'}) = v_1 \cdot a_1(h_i)\mu(t_{i'}) = w_{i'} \cdot a_1(h_i).$$

Hence the change of basis $v_i \to a_1(t_i)v_i,\ i = 1\ldots n$, describes the transformation of μ to $\lambda \uparrow_T G$. The corresponding matrix is

$$D = \mathrm{diag}(a_1(t_i)^{-1} \mid i = 1\ldots n),$$

since G operates from the *right*. $\square$

The case of a permutation representation deserves special attention: here μ is even equal to an induction.

Corollary 3.17 *If, under the conditions of Theorem 3.16, μ is even a permutation representation and $H = \mathrm{stab}_\mu(1)$ then*

$$\mu = 1_H \uparrow_T G$$

for every transversal $T = (t_1,\ldots,t_n)$ of $H\backslash G$ with the property $1^{\mu(t_i)} = i,\ i = 1\ldots n$.

PROOF. Since all entries in the images of μ are 1 we obtain 1_H as the representation from which μ is induced. The correction matrix D hence degenerates to the identity. $\square$

On the uniqueness of this decomposition we prove the following theorem.

Theorem 3.18 *Let λ_i be a representation of $H_i \leq G$ of degree 1 and T_i a transversal of $H_i \backslash G$ for $i = 1, 2$. Suppose*

$$\lambda_1 \uparrow_{T_1} G = \lambda_2 \uparrow_{T_2} G,$$

then H_1 and H_2 are conjugated subgroups in G and λ_1, λ_2 are inner conjugated representations.

PROOF. Let $\mu = \lambda_1 \uparrow_{T_1} G = \lambda_2 \uparrow_{T_2} G$. First, we apply Theorem 3.8 to get $(\lambda_1 \uparrow_{T_1} G) \downarrow H_1 = \ldots \oplus \lambda_1 \oplus \ldots$, where the summand λ_1 arises from the double coset $H_1 \backslash G / H_1$. Hence H_1 stabilizes a point under μ and so does H_2. Since μ is transitive it follows that $H_1 = H_2^s$ for a certain $s \in G$. Using Theorem 3.6 we get $\mu = \lambda_2 \uparrow_{T_2} G = \lambda_2^s \uparrow_{s^{-1}T_2} G$ where λ_2^s is a representation of H_1. We consider the transversal elements $u \in T_1$ and $v \in s^{-1}T_2$ which both are in H_1, without loss of generality at the common position j. Then, for $x \in H_1$,

$$\lambda_1(uxu^{-1}) = \lambda_2^s(vxv^{-1}) \Leftrightarrow \lambda_1(x) = \lambda_2^s(x),$$

and hence λ_1 and λ_2 are inner conjugates. $\square$

The following lemma is obvious.

Lemma 3.19 *Let $H \leq G$ and λ a representation of H of degree 1. If $\mu = \lambda \uparrow_T G$, then $\hat{\mu} = 1_H \uparrow_T G$.*

3.2.3. DECOMPOSITION INTO AN OUTER TENSOR PRODUCT

In this paragraph we prove a necessary and sufficient criterion which determines whether a transitive monomial representation can be decomposed (using a monomial matrix) into a conjugated, outer tensor product of monomial representations. The criterion has been found by Minkwitz for the special case of a permutation representation and is presented here for the monomial case with a shorter proof.

Theorem 3.20 *Let μ be a transitive monomial representation of a group $G = N_1 \times N_2$, which is the direct product of N_1 and N_2. Then μ is equivalent by a monomial matrix M to an outer tensor product of two representations μ_1 of N_1 and μ_2 of N_2 (which necessarily are also monomial and transitive),*

$$\mu \xrightarrow{M} \mu_1 \# \mu_2,$$

if and only if

$$|H| = |H \cap N_1| \cdot |H \cap N_2|.$$

Assume $\mu \xrightarrow{D} \lambda \uparrow_T G$ with a representation λ of degree 1 of H (using Theorem 3.16) then

$$\mu \xrightarrow{DM} ((\lambda \downarrow H \cap N_1) \uparrow_{T_1} N_1) \# ((\lambda \downarrow H \cap N_2) \uparrow_{T_2} N_2),$$

where T_i is a transversal of $(H \cap N_i)\backslash N_i$ for $i = 1,2$ and M is a monomial matrix corresponding to the change of transversals $T \to T_1T_2$ (see Theorem 3.1).

Proof. Suppose $\mu \xrightarrow{M} \mu_1 \# \mu_2$ and M is monomial. Assume $\mu^D = \lambda \uparrow_T G$ with a representation λ of H, and $\mu_i^{D_i} = \lambda_i \uparrow_{T_i} N_i$, where λ_i is a representation of H_i, $i = 1,2$. Then $(\lambda \uparrow_T G)^{D^{-1}M} = ((\lambda_1 \uparrow_{T_1} N_1) \# (\lambda_2 \uparrow_{T_2} N_2))^{D_1^{-1} \otimes D_2^{-1}}$. Switching to the underlying permutation representation on both sides yields (cf. Lemma 3.19, Theorem 3.5) $(1_H \uparrow_T G)^P = 1_{H_1H_2} \uparrow_{T_1T_2} G$ (H_1H_2 is a direct product) where P is the underlying permutation matrix of M. Theorem 3.1 allows us to write $(1_H \uparrow_T G)^P = 1_H \uparrow_{T'} G$ with an appropriate transversal T'. Using Theorem 3.18 we get that H and H_1H_2 are conjugated in G, by say $x = x_1x_2$, $x_i \in N_i$. We get $H = (H_1H_2)^{x_1x_2} = H_1^{x_1}H_2^{x_2}$, which is again a direct product and $|H \cap N_i| = |H_i^{x_i}| = |H_i|$, $i = 1,2$ and hence $|H| = |H_1| \cdot |H_2| = |H \cap N_1| \cdot |H \cap N_2|$ as desired.

Assume now $H_i = H \cap N_i,\ i = 1,2$ and $|H| = |H_1| \cdot |H_2|$. Because of $H_i \trianglelefteq H,\ i = 1,2$ and $H_1 \cap H_2 = \{1\}$ we get $H = H_1 \times H_2 = H_1H_2$ and hence

$$\begin{aligned} \lambda \uparrow_T G &= ((\lambda \downarrow H_1) \# (\lambda \downarrow H_2)) \uparrow_T G \\ &\xrightarrow{M} ((\lambda \downarrow H_1) \uparrow_{T_1} N_1) \# ((\lambda \downarrow H_2) \uparrow_{T_2} N_2)\,, \end{aligned}$$

where the first equality holds since λ is equal to the outer tensor product of the restrictions to the factors (because it is irreducible and of degree 1, Dornhoff (1971), p. 54). The second equality holds because of Theorem 3.5, where M denotes the conjugating monomial matrix corresponding to the change of transversals $T \to T_1T_2$. We get

$$\mu \xrightarrow{DM} \lambda \uparrow_{T_1T_2} G = ((\lambda \downarrow H_1) \uparrow_{T_1} N_1) \# ((\lambda \downarrow H_2) \uparrow_{T_2} N_2)$$

according to Theorem 3.5 and DM is monomial as required. □

Corollary 3.21 *If ϕ is a regular representation or a representation of degree 1 then ϕ decomposes into an outer tensor product exactly as the group decomposes into a direct product.*

Proof. If ϕ is regular, we have $|S| = 1$, if $\deg(\phi) = 1$ then $S = G$. In both cases the condition $|S| = |S \cap N_1| \cdot |S \cap N_2|$ of Theorem 3.20 is satisfied. □

3.2.4. Abelian Groups

The representation theory of abelian groups is classical and well understood (Curtis and Reiner 1962, pp. 34). The purpose of this section is to classify the monomial representations of abelian groups, which provides an efficient way for their decomposition.

First we recall in the following lemma the relationship between representations of a group G and representations of a factor group G/N.

Lemma 3.22 *Let $N \trianglelefteq G$ be a normal subgroup. Then the representations of G/N correspond bijectively to those representations of G for which N is contained in the kernel.*

Proof. Let $\kappa:\ G \to G/N,\ g \mapsto gN$ denote the canonical homomorphism. Assume ϕ is a representation of G/N, then the composition $\phi \circ \kappa$ is a representation of G containing

N in the kernel. If vice-versa ϕ is a representation of G satisfying $N \leq \ker(\phi)$ then $gN \mapsto \phi(g)$ is a (well-defined) representation of G/N. $\square$

When convenient, we identify representations of G/N with the corresponding representation of G. The representation theory of abelian groups is very simple since all irreducibles have degree 1. This implies that any representation ϕ of a subgroup $H \leq G$ has an extension $\overline{\phi}$ to G (follows from Theorem 3.31). We will show in Lemma 3.32 how the extension can be done constructively.

Now we can classify monomial representations of abelian groups.

Theorem 3.23 *Let μ be a transitive monomial representation of an abelian group G with decomposition $\mu^D = \lambda \uparrow_T G$ according to Theorem 3.16, where λ is a representation of $N \leq G$ with extension $\overline{\lambda}$ to G. Then*

$$\mu^{DD_1} = \overline{\lambda} \cdot (1_N \uparrow_T G) \text{ with } D_1 = \operatorname{diag}(\overline{\lambda}(t) \mid t \in T) \text{ and } \overline{\lambda} \downarrow N = \lambda.$$

In particular μ is equivalent (by a diagonal matrix) to the product of a representation of degree 1 and a regular representation of a factor group of G. Thus the irreducible components of μ are pairwise different.

Proof. λ can be extended to a representation $\overline{\lambda}$ of G. Using Theorem 3.11 we get

$$\mu^{DD_1} = (\lambda \uparrow_T G)^{D_1} = \big((\overline{\lambda} \downarrow N) \uparrow_T G\big)^{D_1} = (1_N \uparrow_T G) \otimes \overline{\lambda} = \overline{\lambda} \cdot (1_N \uparrow_T G).$$

By Lemma 3.22, $1_N \uparrow_T G$ is a regular representation of the abelian group G/N and hence contains pairwise different irreducibles and thus the same holds for μ. $\square$

Theorem 3.23 shows that, for an abelian group G, the decomposition problem reduces to the special case of a regular representation. Using Corollary 3.21 we can decompose the latter into regular representations of cyclic groups of prime power order. This can even be done without computing the (many) normal subgroups of G.

3.3. Intertwining Space

Definition 3.24 Assume ϕ, ψ are representations of the group G over the field $\mathbb{K}$ with degrees $\deg(\phi) = n,\ \deg(\psi) = m$ respectively. Then the vector space

$$\mathsf{Int}(\phi, \psi) = \{A \in \mathbb{K}^{n \times m} \mid \forall g \in G :\ \phi(g) \cdot A = A \cdot \psi(g)\}$$

is called the *intertwining space* of ϕ and ψ. Further we denote by

$$\langle \phi, \psi \rangle = \dim(\mathsf{Int}(\phi, \psi))$$

the dimension of the intertwining space or *intertwining number* of ϕ and ψ.

If $\mathbb{K}$ is of characteristic 0, then the intertwining number of two representations is nothing but the scalar product of the corresponding characters (see Curtis and Reiner (1981), p. 212) justifying the notation above. The intertwining number depends only on the equivalence classes of the arguments. Since we consider matrix representations in this paper we need some results on the structure of the intertwining space.

Theorem 3.25 *Let $\phi, \phi_1, \phi_2 \ldots, \psi, \psi_1, \psi_2 \ldots$ be representations over $\mathbb{K}$ of the group G with degrees $\deg(\phi) = n, \deg(\psi) = m$. Then the following holds:*

(i) For $A \in \mathsf{GL}_n(\mathbb{K}), B \in GL_m(\mathbb{K})$*:* $\quad \mathsf{Int}(\phi^A, \psi^B) = A^{-1} \cdot \mathsf{Int}(\phi, \psi) \cdot B.$

(ii) $\mathsf{Int}(\phi_1 \oplus \phi_2, \psi_1 \oplus \psi_2) = \left\{ \begin{bmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{bmatrix} \mid A_{i,j} \in \mathsf{Int}(\phi_i, \psi_j) \right\}.$

(iii) (Schur's Lemma) If ϕ, ψ *are irreducible of degree* n*, then*

$$\mathsf{Int}(\phi, \psi) = \begin{cases} \mathbb{K} \cdot A, \text{ for an } A \in \mathsf{GL}_n(\mathbb{K}), & \phi \cong \psi \\ \mathbf{0}_n, & \phi \not\cong \psi \end{cases}.$$

(iv) Assume $\phi = (\mathbf{1}_{n_1} \otimes \phi_1) \oplus \ldots \oplus (\mathbf{1}_{n_k} \otimes \phi_k)$ *and* $\psi = (\mathbf{1}_{m_1} \otimes \phi_1) \oplus \ldots \oplus (\mathbf{1}_{m_k} \otimes \phi_k)$*,* $n_i, m_j \geq 1$*, are two completely decomposed representations with irreducible, pairwise different* ϕ_i*, then*

$$\mathsf{Int}(\phi, \psi) = (\mathbb{K}^{n_1 \times m_1} \otimes \mathbf{1}_{\deg(\phi_1)}) \oplus \ldots \oplus (\mathbb{K}^{n_k \times m_k} \otimes \mathbf{1}_{\deg(\phi_k)}).$$

Hence every matrix in $\mathsf{Int}(\phi, \psi)$ *is block-permuted with its structure determined by the homogeneous components of* ϕ *and* ψ*.*

PROOF. (i) and (ii) are straightforward. For (iii) let ϕ, ψ be irreducible of degree n. If $\phi \not\cong \psi$ then $\langle \phi, \psi \rangle = 0$, and hence $\mathsf{Int}(\phi, \psi) = \{\mathbf{0}_n\}$. If $\phi \cong \psi$ then there exists an invertible matrix A satisfying $\phi^A = \psi$ which generates the intertwining space because of $\langle \phi, \psi \rangle = 1$. (iv) follows from (ii) and (iii). □

Further results on the intertwining space of inductions and restrictions can be found in Püschel (1998).

Computing the intertwining space is expensive in the general case, however it plays an important role in constructive representation theory. As an example, it can be used to determine a conjugating matrix for two arbitrary, equivalent representations. The computation requires the solution of a system of linear equations.

Assume ϕ, ψ are representations of the same group $G = \langle g_1, \ldots, g_n \rangle$. Obviously, a matrix $A = [a_{i,j}]$ lies in $\mathsf{Int}(\phi, \psi) \subset \mathbb{K}^{\deg(\phi) \times \deg(\psi)}$ if and only if the equations

$$\phi(g_i) \cdot A = A \cdot \psi(g_i) \Leftrightarrow \phi(g_i) \cdot A - A \cdot \psi(g_i) = \mathbf{0}_{\deg(\phi), \deg(\psi)}, \quad i = 1 \ldots n,$$

are satisfied. Thus we obtain for every generator g the following $\deg(\phi) \cdot \deg(\psi)$ equations in the same number of unknowns:

$$\sum_{i=1}^{\deg(\phi)} \phi(g)_{k,i} \cdot a_{i,\ell} - \sum_{j=1}^{\deg(\psi)} a_{k,j} \cdot \psi(g)_{j,\ell} = 0,$$

for $k = 1 \ldots \deg(\phi),\ \ell = 1 \ldots \deg(\psi)$. We restate this in the following theorem.

Theorem 3.26 *The intertwining space of two representations* ϕ, ψ *of the group* G *generated by* $\{g_1, \ldots, g_n\}$ *can be computed by solving a system of* $n \cdot \deg(\phi) \cdot \deg(\psi)$ *linear equations in* $\deg(\phi) \cdot \deg(\psi)$ *unknowns.*

The system of equations is sparse if the representations ϕ, ψ are sparse.

3.4. Decomposition Matrices

Let ϕ be a representation of the group G. We refer to a decomposition matrix of ϕ as any matrix A decomposing ϕ into a direct sum of irreducible representations, i.e.,

$$\phi^A = \bigoplus_{i=1}^{n} \rho_i, \ \rho_i \text{ irreducible for } i = 1 \dots n.$$

Equivalently, this can be stated using the intertwining space.

Definition 3.27 Let ϕ be a representation of a group G and ρ an arbitrary decomposition of ϕ into irreducibles ρ_i, i.e., $\phi \cong \rho = \bigoplus_{i=1}^{n} \rho_i$. We call every invertible matrix $A \in \mathsf{Int}(\phi, \rho)$ a *decomposition matrix* for ϕ.

For the decomposition of the character, i.e., the equivalence class, of a representation, decomposition matrices are not of importance—their mere existence is sufficient. For this reason they rarely appear in the literature. In this paper, decomposition matrices are the central objects of interest.

In the following we derive the main results of this paper that provide formulas for decomposition matrices corresponding to construction for representations like outer tensor product, induction, and extension. Combining these results yields the algorithm for recursively constructing a decomposition matrix for a monomial representation of a solvable group. The decomposition algorithm is presented in Section 4.

3.4.1. Cyclic Groups

Theorem 3.28 *Let $G = \mathsf{Z}_n = \langle x \mid x^n = 1 \rangle$ be the cyclic group of order n with regular representation $\phi: \ x \mapsto [(1, \dots, n), n]$. Then ϕ is decomposed by the matrix $\mathrm{DFT}_n = [\omega_n^{ij} \mid i, j \in \{0, \dots, n-1\}]$ into $\bigoplus_{i=0}^{n-1} \lambda_i$, where $\lambda_i: \ x \mapsto \omega_n^i$.*

Proof. It is sufficient to show that the jth column of DFT_n, $j = 0 \dots n-1$, is an eigenvector of $[(1, \dots, n), n]$ with eigenvalue ω_n^j.

$$\begin{aligned}
[(1, \dots, n), n] \cdot (\omega_n^{0 \cdot j}, \dots, \omega_n^{(n-1) \cdot j})^T &= (\omega_n^{1 \cdot j}, \dots, \omega_n^{(n-1) \cdot j}, \omega_n^{0 \cdot j})^T \\
&= \omega_n^j \cdot (\omega_n^{0 \cdot j}, \dots, \omega_n^{(n-1) \cdot j})^T,
\end{aligned}$$

as desired. □

The basic building blocks for solvable groups are cyclic groups of prime order p. We will see that correspondingly the matrices DFT_p, for a prime p, are the basic building blocks for decomposition matrices of inductions.

3.4.2. Direct Sum

The decomposition of a direct sum is obvious.

Theorem 3.29 *Let ϕ_1, ϕ_2 be representations of G with decomposition matrices A_1, A_2. Then $A_1 \oplus A_2$ is a decomposition matrix for $\phi_1 \oplus \phi_2$.*

3.4.3. Outer Tensor Product

Theorem 3.30 *Let ϕ_1, ϕ_2 be representations of N_1, N_2 with decomposition matrices A_1, A_2, respectively. Then there exists a permutation matrix P such that $(A_1 \otimes A_2) \cdot P$ is a decomposition matrix for $\phi_1 \# \phi_2$.*

Proof. This follows from the distributivity of "#" and the fact that the outer tensor product of two irreducible representations again is irreducible. The computation of P is an easy combinatorial task, which we omit here. □

Note that the corresponding statement does not hold for the inner tensor product. If ϕ, ψ are irreducible representations then $\phi \otimes \psi$ is, in general, not irreducible. Furthermore, the decomposition of $\phi \otimes \psi$ is difficult and occurs frequently in physics where it is known as the *Clebsch-Gordan-Problem*.

3.4.4. Induction

The transitivity of induction (Theorem 3.2),

$$\phi_H \uparrow_{TS} G = (\phi_H \uparrow_T K) \uparrow_S G,$$

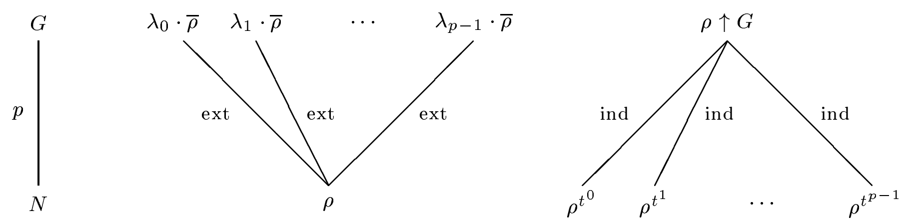
provides an immediate idea for the stepwise decomposition of an induction. First, determine a maximal subgroup K between H and G. Second, decompose the lower induction by recursion (divide) and derive a decomposition of the upper induction (conquer). The conquer step requires the answers to the following two questions:

1. How do the irreducible components of $\phi_H \uparrow G$ arise from those of $\phi_H \uparrow K$?
2. Given a decomposition matrix of $\phi_H \uparrow K$. How do we compute a decomposition matrix of $\phi_H \uparrow G$?

Unfortunately, the answers to these questions do not exist in general. In the case that $K \trianglelefteq G$ (and hence of prime index), Clifford's theorem provides an exact answer to question 1. An answer to question 2 is presented in Theorem 3.33. Clifford's theorem in its general form (Curtis and Reiner 1962, p. 345) deals with the restriction of an irreducible representation of a group G to a normal subgroup $N \trianglelefteq G$ of arbitrary index. For the requirements of this paper we only consider the case where the index $(G : N)$ is prime and present the theorem in a constructive form. In different forms, the theorem can be found in a more general setting in Rockmore (1990) and in Baum and Clausen (1994). In both cases it was used for the construction of fast Fourier transforms.

Theorem 3.31 (Clifford) *Let $N \overset{p}{\trianglelefteq} G$ be a normal subgroup of prime index p, $T = (t^0, t^1, \ldots, t^{p-1})$ a transversal of G/N and ρ an irreducible representation of N. Then exactly one of the two following cases applies:*

1. *(cf. Figure 1, Case 1) $\rho \cong \rho^{t^i}$ for $i = 0 \ldots p-1$. Then ρ has exactly p pairwise inequivalent extensions to G. Assume $\overline{\rho}$ is one of these and $\lambda_i : \ t \mapsto \omega_p^i$ a representation of G/N, $i = 0 \ldots p-1$, then all extensions are given by $\lambda_i \cdot \overline{\rho}$, $i = 0 \ldots p-1$.*



Case 1: $\rho \cong \rho^{t^i}$ Case 2: $\rho \not\cong \rho^{t^i}$

Figure 1. Clifford's Theorem.

The induction decomposes into irreducibles according to

$$(\rho \uparrow_T G)^A = \bigoplus_{i=0}^{p-1} \lambda_i \cdot \overline{\rho},$$

where $A = \mathrm{diag}(\overline{\rho}(t)^i \mid i = 0 \ldots p-1) \cdot (\mathrm{DFT}_p \otimes \mathbf{1}_{n/p})$.

2. *(cf. Figure 1, Case 2)* $\rho \not\cong \rho^{t^i}$ *for* $i = 0 \ldots p-1$. *Then the induction* $\rho \uparrow G$ *is irreducible,*

$$(\rho \uparrow_T G) \downarrow N = \bigoplus_{i=0}^{p-1} \rho^{t^i}$$

and

$$\rho^{t^i} \uparrow_T G = (\rho \uparrow_T G)^B,$$

where $B = \left([(1,\ldots,p)^{-i}, p] \otimes \mathbf{1}_{\deg(\rho)}\right) \cdot \left(\mathbf{1}_{(p-i)\cdot\deg(\rho)} \oplus (\mathbf{1}_i \otimes \rho(t^p))\right)$.

Proof. The proof can be found, e.g., in Clausen and Baum (1993), pp. 88. We prove only the three equations. Equation in 1.: Using Theorem 3.11 leads to

$$(\rho \uparrow_T G)^D = (1_N \uparrow_T G) \otimes \overline{\rho},\ D = \mathrm{diag}(\overline{\rho}(t)^i \mid i = 0 \ldots p-1).$$

The representation $1_N \uparrow_T G$ is a regular representation $t \mapsto [(1,\ldots,p), p]$ of $G/N \cong \mathsf{Z}_p$ and is decomposed by DFT_p into p representations $t \mapsto \omega_p^i$, $i = 0 \ldots p-1$, of degree 1 which gives the first equation.

First equation in 2.: Follows from Corollary 3.9. Second equation in 2.: By Theorem 3.6 we have $\rho^{t^i} \uparrow_T G = \rho \uparrow_{t^iT} G$. Multiplication of T with t^i permutes the cosets according to $\sigma = (1,\ldots,p)^{-i}$. The transition from T^σ to t^iT,

$$T^\sigma = (t^i,\ldots,t^{(p-1)},t^0,\ldots,t^{(i-1)}) \to t^iT = (t^i,\ldots,t^{(p-1)},t^p,\ldots,t^{(p-i+1)}),$$

is equivalent to a multiplication of the last i transversal elements by t^p. Using Theorem 3.1 gives the result. $\square$

In the case $\rho \cong \rho^t$, ρ has an extension to G. This can for instance be calculated using the Extension Formula of Minkwitz (Minkwitz 1996, Clausen 1997). The formula requires the determination of an extending character and the evaluation of ρ for all $h \in H$. In the particular situation in Clifford's theorem, however, there is also another method which can be found in Rockmore (1990) and is based on the following lemma.

Lemma 3.32 *Assume the situation of Theorem 3.31. In the case $\rho \cong \rho^t$ putting $\rho(t) = A$ defines an extension of ρ to G if and only if $A \in \mathsf{Int}(\rho^t, \rho)$ and $A^p = \rho(t^p)$.*

If the degree of ρ is small and $|N|$ is large, then this lemma provides a more efficient way to compute the extension than Minkwitz' formula. Another advantage is that no extending character has to be computed. Note that $\mathsf{Int}(\rho^t, \rho)$ is of dimension 1, hence any generator differs from $\rho(t)$ only by a scalar multiple.

Clifford's theorem shows how the irreducible representations of G arise from those of N. Accordingly the following theorem shows how a decomposition matrix of a representation ϕ of N gives rise to a decomposition matrix of $\phi \uparrow G$.

Theorem 3.33 *Let $N \overset{p}{\trianglelefteq} G$ be a normal subgroup of prime index p with transversal $T = (t^0, t^1, \ldots, t^{(p-1)})$. Assume ϕ is a representation of N of degree n with decomposition matrix A such that $\phi^A = \bigoplus_{i=1}^k \rho_i$, where $\rho_1, \ldots, \rho_j$ are exactly those among the ρ_i which have an extension $\overline{\rho}_i$ to G (Theorem 3.31, Case 1). Denote by $d = \deg(\rho_1) + \ldots + \deg(\rho_j)$ the entire degree of the extensible ρ_i and set $\overline{\rho} = \overline{\rho}_1 \oplus \ldots \oplus \overline{\rho}_j$. Then there exists a permutation matrix P such that*

$$M = (\mathbf{1}_p \otimes A) \cdot P \cdot \left(\bigoplus_{t \in T} \overline{\rho}(t) \oplus \mathbf{1}_{p(n-d)} \right) \cdot \left((\mathrm{DFT}_p \otimes \mathbf{1}_d) \oplus \mathbf{1}_{p(n-d)} \right)$$

is a decomposition matrix of $\phi \uparrow_T G$. If we denote by $\lambda_i : \ t \mapsto \omega_p^i, \ i = 0 \ldots p-1$, the p one-dimensional representations of G/N then

$$(\phi \uparrow_T G)^M = \bigoplus_{i=0}^{p-1} \bigoplus_{\ell=1}^{j} \lambda_i \cdot \overline{\rho}_\ell \oplus \bigoplus_{i=j+1}^{k} \rho \uparrow_T G$$

is the corresponding decomposition into irreducibles.

Proof. By Theorem 3.4 we get

$$(\phi \uparrow_T G)^{(\mathbf{1}_p \otimes A)} = \phi^A \uparrow_T G = \left(\bigoplus_{i=1}^{k} \rho_i \right) \uparrow_T G.$$

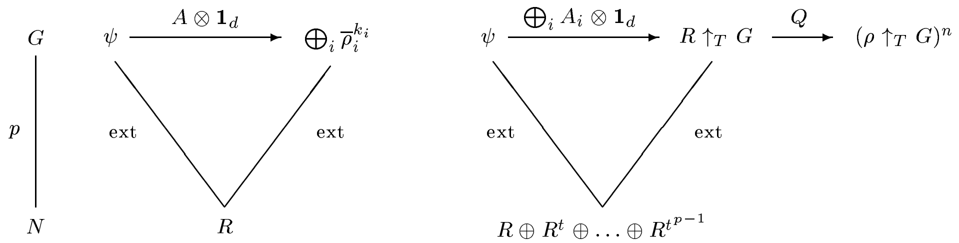
We use Theorem 3.3 with the block decomposition $\rho_1 \oplus \ldots \oplus \rho_j, \rho_{j+1}, \ldots, \rho_k$ of ρ to compute a permutation matrix P such that

$$(\phi \uparrow_T G)^{(\mathbf{1}_p \otimes A) \cdot P} = \rho \uparrow_T G \oplus \rho_{j+1} \uparrow_T G \oplus \ldots \oplus \rho_k \uparrow_T G.$$

Since ρ has an extension $\overline{\rho}$ to G we get by Theorem 3.11

$$(\rho \uparrow_T G)^{\bigoplus_{t \in T} \overline{\rho}(t)} = (1_N \uparrow_T G) \otimes \overline{\rho}.$$

The representation $(1_N \uparrow_T G)$ is decomposed by DFT_p into $\bigoplus_{i=0}^{p-1} \lambda_i$ where $\lambda_i : \ t \mapsto \omega_p^i$. Thus $\bigoplus_{t \in T} \overline{\rho}(t) \cdot (\mathrm{DFT}_p \otimes \mathbf{1}_d)$ is a decomposition matrix for $\rho \uparrow_T G$ with corresponding decomposition $\bigoplus_{i=0}^{p-1} \bigoplus_{\ell=1}^{j} \lambda_i \cdot \overline{\rho}_\ell$. The inductions of the $\rho_i, \ i = j+1 \ldots k$, are already irreducible which completes the proof. □



Case 1: $R = \rho^n$, ρ extends to G Case 2: $R = \rho^n$, $\rho \uparrow_T G$ is irreducible

Figure 2. Decomposing an extension using Theorem 3.34.

3.4.5. Extension

In the last section we have seen that it is possible to derive a decomposition of $\phi_N \uparrow G$ from a decomposition of ϕ_N, if $N \overset{p}{\trianglelefteq} G$. In this section we investigate, how to derive in this situation a decomposition of an extension $\overline{\phi_N}$ (if it exists) from a decomposition of ϕ_N. The result gives us a second recursive decomposition method which allows us to decompose every monomial representation of a solvable group.

Theorem 3.34 *Let $N \overset{p}{\trianglelefteq} G$ be a normal subgroup of prime index p with transversal $T = (t^0, t^1, \ldots, t^{p-1})$ and representation ϕ a over the field $\mathbb{K}$. Assume that ϕ has an extension $\overline{\phi}$ to G. Further let A decompose ϕ such that equivalent irreducibles are equal and adjacent, $\phi^A = \bigoplus_{i=1}^k R_i$, where $R_i = \rho_i^{n_i}$ is a homogeneous component of multiplicity n_i. We denote $d_i = \deg(\rho_i)$. Furthermore, we require that whenever $R_i \cong R_j^{t^\ell}$ then even $R_i = R_j^{t^\ell}$ and that these components are adjacent, ordered according to $R_i, R_i^t, \ldots, R_i^{t^{p-1}}$. Then there exist invertible matrices $A_i \in \mathbb{K}^{n_i \times n_i}$ and a permutation matrix P such that*

$$M = A \cdot \left(\bigoplus_{i=1}^{k} A_i \otimes \mathbf{1}_{d_i} \right) \cdot P$$

is a decomposition matrix of $\overline{\phi}$.

Proof. Let ρ be one of the ρ_i with $\rho \ncong \rho^t$. According to Theorem 3.31, Case 2, the direct sum $\rho \oplus \rho^t \oplus \ldots \oplus \rho^{t^{p-1}}$ extends to $\rho \uparrow_T G$. Since ϕ has an extension to G, the multiplicity of $\rho, \rho^t, \ldots, \rho^{t^{p-1}}$ in ϕ is equal.

Now we investigate, how far $\overline{\phi}$ is decomposed by A. Obviously, $\overline{\phi}^A$ is an extension of ϕ^A. Extensions of inequivalent extensible ρ_i cannot be equivalent. Also the extension of an extensible ρ_i and the induction of a non-extensible ρ_j cannot be equivalent (follows from Theorem 3.31). Hence, $\overline{\phi} \xrightarrow{A} \bigoplus_{j=1}^{\ell} \psi_j$ and for each $\psi = \psi_j$, either $\psi \downarrow N = R$ for an extensible homogeneous component R, or $\psi \downarrow N = R \oplus R^t \oplus \ldots \oplus R^{t^{p-1}}$ for a non-extensible homogeneous component R.

The remaining task is to decompose the blocks $\psi = \psi_j$ which we do next for both cases (cf. Figure 2).

Case 1: $\psi \downarrow N = R = \rho^n$ for an extensible ρ. Then ρ has p pairwise inequivalent extensions $\overline{\rho}_i$, $i = 1 \ldots p$ and hence ψ decomposes as $\bigoplus_{i=1}^p \overline{\rho}_i^{k_i}$ with certain $k_i \geq 0$, $\sum_{i=1}^p k_i =$

n. For the corresponding decomposition matrix B we have $B \in \mathsf{Int}(\psi, \bigoplus_{i=1}^{p} \overline{\rho}_i^{k_i}) \leq \mathsf{Int}(R, R) = \mathbb{K}^{n \times n} \otimes \mathbf{1}_d$, $d = \deg(\rho)$, according to Theorem 3.25, iv). Hence $B = A \otimes \mathbf{1}_d$ with an invertible matrix A.

Case 2: $\psi \downarrow N = R \oplus R^t \oplus \ldots \oplus R^{t^{p-1}} = R'$, $R = \rho^n$ and $\rho \uparrow_T G$ is irreducible. The direct sum R' can also be extended by $R \uparrow_T G$ (Corollary 3.9) since all the R^{t^i} have the common multiplicity n (see above). For the corresponding conjugation matrix we have $B \in \mathsf{Int}(\psi, R \uparrow_T G) \leq \mathsf{Int}(R', R') = \bigoplus_{i=1}^{p} \mathbb{K}^{n \times n} \otimes \mathbf{1}_d$, $d = \deg(\rho)$. Hence $B = \bigoplus_{i=1}^{p} A_i \otimes \mathbf{1}_d$ with invertible matrices A_i. $R \uparrow_T G$ is decomposed into $(\rho \uparrow_T G)^n$ by a permutation matrix Q (using Theorem 3.3).

Altogether we get that $A \cdot (\bigoplus_{i=1}^{k} A_i \otimes \mathbf{1}_{d_i}) \cdot P$ is a decomposition matrix for $\overline{\phi}$, where P is the permutation matrix arising from the direct sum of the matrices Q in Case 2. This completes the proof. □

Note that the condition $R_i \cong R_j^{t^\ell} \Rightarrow R_i = R_j^{t^\ell}$ in Theorem 3.34 can be satisfied by a block diagonal matrix, which can be computed block by block using Theorem 3.26. The requirements concerning the ordering of irreducibles can be established by a suitable permutation matrix.

The problem in Theorem 3.34 from an algorithmic point of view is the efficient computation of the matrices A_i. In the proof, not the matrices A_i but the matrices $A_i \otimes \mathbf{1}_{d_i}$ have been determined which requires the (expensive) computation of the intertwining space of representations which are a factor of d_i larger than the matrices A_i in which we are actually interested. In the following we present efficient methods for the computation of the A_i for both cases (see Figure 2) considered in the proof of Theorem 3.34, following the notation used there. Case 1 is solved in Theorem 3.37 and Case 2 in Theorem 3.38.

Case 1. $\psi \downarrow N = R = \rho^n$ for an extensible ρ. Let $B = A \otimes \mathbf{1}_d$ be the decomposition matrix with $\psi \xrightarrow{B} \bigoplus_{i=1}^{p} \overline{\rho}_i^{k_i}$. Rather than computing B we want to compute directly A (dealing with smaller matrices). The following definition provides us with the appropriate "shrink operator".

Definition 3.35 Let $n, d \geq 1$. We define the *partial trace operator* T_d as

$$T_d: \ \mathbb{K}^{nd \times nd} \to \mathbb{K}^{n \times n}, \ M = [M_{i,j}] \mapsto [\mathrm{tr}(M_{i,j})],$$

where $[M_{i,j}]$ is a division of M into $d \times d$ matrices and $\mathrm{tr}(\cdot)$ denotes the ordinary trace.

We use the following properties of the operator T_d.

Lemma 3.36 *Let $M \in \mathbb{K}^{nd \times nd}$ and $A \in \mathbb{K}^{n \times n}$. Then*

(i) $T_d(M \cdot (A \otimes \mathbf{1}_d)) = T_d(M) \cdot A$,

(ii) $T_d((A \otimes \mathbf{1}_d) \cdot M) = A \cdot T_d(M)$,

(iii) $T_d(M^{A \otimes \mathbf{1}_d}) = T_d(M)^A$.

Proof. We prove only (i), the other statements are shown analogously. Let $A = [a_{i,j}]$ and $M = [M_{i,j}]$ with $d \times d$ matrices $M_{i,j}$, where $i, j = 1 \ldots n$. We get $M \cdot (A \otimes \mathbf{1}_d) = [\sum_{k=1}^{n} M_{i,k} \cdot a_{k,j}]$. Applying T_d yields, using the linearity of the trace, $[\sum_{k=1}^{n} \mathrm{tr}(M_{i,k}) \cdot a_{k,j}] = T_d(M) \cdot A$ as desired. □

The following theorem allows us to efficiently compute the matrix $A \otimes \mathbf{1}_d$ in Case 1 (cf. Figure 2).

Theorem 3.37 *We use previous notation. Let $\psi \downarrow N = R = \rho^n$ for an extensible ρ, $d = \deg(\rho)$, and $\overline{R} = \bigoplus_{i=1}^p \overline{\rho}_i^{k_i}$ a decomposition of ψ into irreducibles. $\overline{\rho}_i$ are pairwise inequivalent extensions of ρ. Then there exists $g \in G \setminus N$ (set difference) satisfying $\mathrm{tr}(\overline{\rho}_1(g)) \neq 0$ and if $A \in \{A \mid T_d(\psi(g)) \cdot A = A \cdot T_d(\overline{R})\}$ and A is invertible, then $\psi \xrightarrow{A\otimes \mathbf{1}_d} \overline{R}$.*

Proof. We have

$$\begin{aligned}\mathsf{Int}(\psi, \overline{R}) &= \{B \mid \psi(g) \cdot B = B \cdot \overline{R}(g), \text{ for all } g \in G\} \\ &= \{A \mid \psi(g) \cdot (A \otimes \mathbf{1}_d) = (A \otimes \mathbf{1}_d) \cdot \overline{R}(g), \text{ for all } g \in G \setminus N\} \\ &= \{A \mid \psi(g) \cdot (A \otimes \mathbf{1}_d) = (A \otimes \mathbf{1}_d) \cdot \overline{R}(g), \text{ for one } g \in G \setminus N\}.\end{aligned}$$

The first equality holds, since a matrix of structure $B = A \otimes \mathbf{1}_d$ guarantees that B is in the intertwining space of the restrictions to N. The second equality holds since $(G : N)$ is prime. We denote $V_g = \{A \mid \psi(g) \cdot (A \otimes \mathbf{1}_d) = (A \otimes \mathbf{1}_d) \cdot \overline{R}(g)\}$. Mapping the equation defining V_g with T_d yields $W_g = \{A \mid T_d(\psi(g)) \cdot A = A \cdot T_d(\overline{R}(g))\}$ according to Lemma 3.36, (i) and (ii). We have to show that $V_g = W_g$. Obviously $V_g \leq W_g$ and $\dim(V_g) = \sum_{i=1}^p k_i^2$. $\psi(g)$ and $\overline{R}$ are conjugated by a matrix $A \otimes \mathbf{1}_d$, hence $T_d(\psi(g))$ and $T_d(\overline{R})$ are conjugated by A (Lemma 3.36, (iii)). Thus $W_g' = \{A \mid T_d(\overline{R}(g)) \cdot A = A \cdot T_d(\overline{R}(g))\}$ has the same dimension as W_g. Evaluating T_d yields $T_d(\overline{R}(g)) = \bigoplus_{i=1}^p \mathrm{tr}(\overline{\rho}_1(g)) \cdot a_i \cdot \mathbf{1}_{k_i}$ with pairwise different $a_i \neq 0$ (the a_i are all powers of ω_p, Theorem 3.31). Since $\mathrm{tr}(\overline{\rho}_1(g)) \neq 0$ by assumption, we get $\dim(W_G') = \sum_{i=1}^p k_i^2$ and hence $\dim(W_g') = \dim(W_g) = \dim(V_g)$ as desired.

It remains to show that a $g \in G \setminus N$ with $\mathrm{tr}(\overline{\rho}_1) \neq 0$ exists. Assume $\mathrm{tr}(\overline{\rho}_1(g)) \neq 0$ on all conjugacy classes, which are not in N and let $\overline{\rho}_i$ be another extension of ρ. Then $\overline{\rho}_i = \lambda \cdot \overline{\rho}_1$ with a certain representation of G/N of degree 1 (Theorem 3.31). Hence $\overline{\rho}_1 \cong \overline{\rho}_i$ which is a contradiction to the fact that ρ has p inequivalent extensions. □

Case 2. The following theorem deals with Case 2 (cf. Figure 2), showing that the decomposition matrix $\bigoplus_{i=1}^p A_i \otimes \mathbf{1}_d$ can be determined without computing an intertwining space.

Theorem 3.38 *We use previous notation. Let $\psi \downarrow N = R \oplus R^t \oplus \ldots \oplus R^{t^{p-1}} = R'$, $R = \rho^n$, $d = \deg(\rho)$ and suppose that $\rho \uparrow_T G$ is irreducible. We evaluate ψ at the transversal $T = (t^0, t^1, \ldots, t^{p-1})$ and consider each image to be represented as a $p \times p$ matrix of $nd \times nd$ matrices: $\psi(t^i) = [M_{k,\ell}^{(i)}]_{k,\ell}$. Then each $M_{1,i+1}^{(i)}$, $i = 0 \ldots p-1$, has the form $A_i \otimes \mathbf{1}_d$. Furthermore, $\psi \xrightarrow{A} R \uparrow_T G$ with $A = \bigoplus_{i=1}^p A_i^{-1} \otimes \mathbf{1}_d$.*

Proof. As in the beginning of the proof of Theorem 3.37 we have that a matrix $C = \bigoplus_{i=1}^p C_i \otimes \mathbf{1}_d \in \mathsf{Int}(R \uparrow_T G, \psi)$ if and only if $(R \uparrow_T G)(g) \cdot C = C \cdot \psi(g)$ for one $g \in G \setminus N$, e.g. for $g = t$.

Let $R \uparrow_T G \xrightarrow{C} \psi$ with C given as above. We compute

$$(R \uparrow_T G)(t) = \begin{bmatrix} \mathbf{0}_{nd} & \mathbf{1}_n \otimes \mathbf{1}_d & & \\ & & \ddots & \\ & & & \mathbf{1}_n \otimes \mathbf{1}_d \\ \mathbf{1}_n \otimes \rho(t^p) & & & \mathbf{0}_{nd} \end{bmatrix},$$

where omitted blocks are $= \mathbf{0}_{nd}$. In general, the matrix $(R \uparrow_T G)(t^i)$ has in the first block row at position $i+1$ the matrix $\mathbf{1}_{nd}$ and $\mathbf{0}_{nd}$ else ($i = 0 \ldots p-1$). Correspondingly, $\psi(t^i) = (R \uparrow_T G)(t^i)^C$ has the matrix $C_1^{-1}C_{i+1} \otimes \mathbf{1}_d$ at position $(1, i+1)$, $i = 0 \ldots p-1$. We set $A_j = C_1^{-1}C_j$ and $A' = \bigoplus_{j=1}^{p} A_j \otimes \mathbf{1}_d$. It remains to show that $R \uparrow_T G \xrightarrow{A'} \psi$ and hence $\psi \xrightarrow{A} R \uparrow_T G$ with $A = A'^{-1}$. Because of the remark at the beginning of this proof, it is sufficient to show it for the image on t. It is $A' = (1_p \otimes C_1^{-1} \otimes \mathbf{1}_d) \cdot C$. Since the first factor of A' leaves $(R \uparrow_T G)(t)$ (given above) invariant by conjugation, $(R \uparrow_T G)(t)^{A'} = (R \uparrow_T G)(t)^C = \psi(t)$, which completes the proof. □

Note that the determination of the conjugating matrix A in Theorem 3.38 is similar to the determination of the conjugating diagonal matrix D in Theorem 3.16.

4. Decomposing Monomial Representations

In this section we present and explain in detail the algorithm for decomposing an arbitrary monomial representation μ of a solvable group (as in the entire paper, we assume that μ is given over a splitting field and that the Maschke condition is satisfied). The The key property of the algorithm is that, in addition to the irreducible representations contained in μ, also the corresponding decomposition matrix A is computed as a product of highly structured, sparse matrices. This factorization can be regarded as a fast algorithm for the multiplication with A. For the special case of a regular representation of a group G we hence obtain a fast Fourier transform for G. A nice feature of the algorithm is the fact that the character table of a group never needs to be computed.

For the algorithm we use a stronger definition of decomposition as in Definition 3.27. Let μ be a representation of a group G. Decomposing μ means computing a matrix A such that

$$\mu^A = \bigoplus_{i=1}^{k} \rho_i, \quad \text{where } \rho_i \text{ is irreducible for } i = 1 \ldots k.$$

Furthermore, we require equivalent irreducibles to be equal, i.e., $\rho_i \cong \rho_j \Rightarrow \rho_i = \rho_j$, and adjacent, and all irreducibles shall be (partially) ordered with respect to their degrees.

4.1. The Algorithm

Algorithm 4.1 Input: a transitive monomial representation μ of degree n of a solvable group G. Output: a decomposition matrix A of μ, given as a product of highly structured sparse matrices, and irreducible representations ρ_i of G, such that

$$\mu^A = \bigoplus_{i=1}^{m} \rho_i^{k_i}, \text{ where } \rho_i \text{ is irreducible for } i = 1 \ldots m,$$

and the following conditions hold

1. $i \neq j \Rightarrow \rho_i \not\cong \rho_j$.
2. The ρ_i are ordered by degree.

For the convenience of the reader we first give a rough sketch of the recursive algorithm and give the detailed version afterwards. In the following, D, P, M denote a diagonal matrix, a permutation matrix, and a monomial matrix, respectively.

Sketched:

Case 1: μ is not faithful.

μ induces a representation μ' of $G/\ker(\phi)$. Recurse with μ'.

Case 2: μ is irreducible.

There is nothing to do.

Case 3: μ is not transitive.

Decompose $\mu \xrightarrow{P} \mu_1 \oplus \ldots \oplus \mu_k$ into its transitive components μ_i. Recurse with the μ_i.

Case 4: μ is a monomial representation of an abelian group.

Decompose $\mu \xrightarrow{D} \lambda_G \cdot (1_H \uparrow_T G)$ into a regular representation. Recurse with $1_H \uparrow_T G$.

Case 5: μ is a conjugated outer tensor product.

Decompose $\mu \xrightarrow{M} \mu_1 \# \ldots \# \mu_k$. Recurse with the μ_i.

Case 6: $\mu^D = \lambda_H \uparrow_T G$ and it exists N with $H \leq N \overset{p}{\trianglelefteq} G$ (induction recursion).

Decompose $\lambda_H \uparrow_T G \xrightarrow{M} (\lambda_H \uparrow_{T_1} N) \uparrow_{T_2} G$ into a double induction. Recurse with $\lambda_H \uparrow_{T_1} N$.

Case 7: $\mu^D = \lambda_H \uparrow_T G$ and it exists N with $H \not\leq N \overset{p}{\trianglelefteq} G$ (switch recursion).

Decompose the restriction $(\lambda_H \uparrow_T G) \downarrow N \xrightarrow{M} \lambda_{H \cap N} \uparrow_{T'} N$ into an induction. Recurse with $\lambda_{H \cap N} \uparrow_{T'} N$.

Detailed:

Case 1: μ is not faithful.

1. Compute the kernel K. If μ is transitive, decompose $\mu \xrightarrow{D} \lambda_H \uparrow_T G$ using Theorem 3.16 and use Theorem 3.12.
2. Construct a faithful representation μ' of G/K and decompose $\mu' \xrightarrow{A} \bigoplus_{j=1}^{m} \rho'_j$ by recursion. We represent G/K (which is solvable) as an ag group to speed up computation (cf. GAP 3 manual GAP (1997), pp. 522).
3. Translate every irreducible ρ'_j of G/K into an irreducible ρ_j of G.

Case 2: μ is irreducible.

The irreducibility is tested with the character of μ ($\langle \chi_\mu, \chi_\mu \rangle \overset{?}{=} 1$). $A = \mathbf{1}_n$ is a decomposition matrix with decomposition μ.

Case 3: μ is not transitive.

1. Decompose $\mu \xrightarrow{P_1} \bigoplus_{i=1}^{\ell} \mu_i$ using Theorem 3.15. The μ_i are transitive and P_1 is a permutation matrix.

2. Decompose $\mu_i \xrightarrow{A_i} \bigoplus_{j=1}^{m_i} \rho_{i,j}^{k_{i,j}}$ for $i = 1 \ldots \ell$ by recursion.
3. Compute a block diagonal matrix D which conjugates equivalent irreducibles of different μ_i to be equal. This is done by solving a system of linear equations according to Theorem 3.26. Note that equivalent irreducibles of the same μ_i are already equal. The blocks in D correspond (in the coarsest case) to the degrees of the $\rho_{i,j}$.
4. Determine a permutation matrix P_2 which sorts the $\rho_{i,j}$ according to their degrees such that equals are adjacent.

$A = P_1 \cdot \left(\bigoplus_{i=1}^{\ell} A_i \right) \cdot D \cdot P_2$ decomposes μ.

Case 4: μ is a monomial representation of an abelian group.

1. Decompose $\mu \xrightarrow{D} \lambda_G \cdot (1_H \uparrow_T G)$ using Theorem 3.23.
2. Decompose $1_H \uparrow_T G \xrightarrow{A} \bigoplus_{j=1}^{m} \rho_j$ by recursion.

$D \cdot A$ decomposes μ with decomposition $\bigoplus_{j=1}^{m} \lambda_G \cdot \rho_j$.

Case 5: μ is a conjugated outer tensor product.

1. Decompose $\mu \xrightarrow{M} \mu_1 \# \ldots \# \mu_\ell$ using Theorem 3.30. If G is abelian, then μ is regular (because of Cases 4 and 1) and we can use Corollary 3.21. M is monomial.
2. Decompose $\mu_i \xrightarrow{A_i} \bigoplus_{j=1}^{m_i} \rho_{i,j}^{k_{i,j}}$ for $i = 1 \ldots \ell$ by recursion.
3. Determine a permutation matrix P, such that

$$\begin{array}{cl} & \left(\bigoplus_{j_1=1}^{m_1} \rho_{1,j_1}^{k_{1,j_1}} \# \ldots \# \bigoplus_{j_\ell=1}^{m_\ell} \rho_{\ell,j_\ell}^{k_{\ell,j_\ell}} \right) \\ \xrightarrow{P} & \bigoplus_{j_1=1}^{m_1} \cdots \bigoplus_{j_\ell=1}^{m_\ell} (\rho_{1,j_1} \# \ldots \# \rho_{\ell,j_\ell})^{k_{1,j_1} \cdots k_{\ell,j_\ell}} . \end{array}$$

   P is computed from the degrees of the $\rho_{i,j}$.

$A = M \cdot \left(\bigotimes_{i=1}^{k} A_i \right) \cdot P$ is a decomposition matrix for μ and

$$\mu^A = \bigoplus_{j_1=1}^{m_1} \cdots \bigoplus_{j_\ell=1}^{m_\ell} (\rho_{1,j_1} \# \ldots \# \rho_{\ell,j_\ell})^{k_{1,j_1} \cdots k_{\ell,j_\ell}} .$$

Case 6: If $\mu^D = \lambda_H \uparrow_T G$ and it exists N with $H \le N \overset{p}{\trianglelefteq} G$ (induction recursion).

1. Determine N by building the normal closure $\overline{H}$ of H in G and computing a composition series of $G/\overline{H}$.
2. Decompose $\lambda_H \uparrow_T G \xrightarrow{M} (\lambda_H \uparrow_{T_1} N) \uparrow_{T_2} G$ using Theorem 3.1 such that T_2 has the form $T_2 = (t^0, t^1, \ldots, t^{p-1})$. M is monomial.
3. Decompose $(\lambda_H \uparrow_{T_1} N) \xrightarrow{B} \bigoplus_{i=1}^{m} \rho_i^{k_i}$ by recursion.
4. Determine which of the ρ_i have an extension to G (cf. Theorem 3.31) which is equivalent to $\rho_i^t \cong \rho_i$. We decide this by computing the permutation π_t arising from t permuting the conjugacy classes of N (by conjugation). Then $\rho_i^t \cong \rho_i$ if and only if the list of character values of ρ_i is invariant under π_t. We denote the extensible representations by σ_ℓ, $\ell = 1 \ldots r$ and their direct sum by σ.

5. Conjugate the non-extensible ρ_i such that the following holds: If $\rho_i \cong \rho_j^{t^k}$ then even equality holds. This is done by using Theorem 3.26 and gives rise to a block diagonal matrix D_1. We denote a (complete) direct sum of inner conjugate non-extensibles by $\tau_\ell = \rho \oplus \ldots \oplus \rho \oplus \rho^t \oplus \ldots \oplus \rho^t \oplus \ldots \oplus \rho^{t^{p-1}} \oplus \ldots \oplus \rho^{t^{p-1}}$ for $\ell = 1 \ldots s$ (Note that the multiplicities of the ρ^{t^k} are not equal in general).
6. Compute a permutation matrix P_1 such that $(\lambda_H \uparrow_{T_1} N) \xrightarrow{B \cdot D_1 \cdot P_1} \sigma \oplus \tau_1 \oplus \ldots \oplus \tau_m$. We apply Theorem 3.33 to obtain a decomposition matrix A' for $(\lambda_H \uparrow_{T_1} N) \uparrow_{T_2} G$.

$$\begin{aligned} A' \quad = \quad & (\mathbf{1}_p \otimes B \cdot D_1 \cdot P_1) \cdot P \cdot \left(\bigoplus_{i=1}^p \overline{\sigma}(t)^i \oplus \mathbf{1}_{p(n-d)}\right) \cdot \\ & \left((\mathrm{DFT}_p \otimes \mathbf{1}_d) \oplus \mathbf{1}_{p(n-d)}\right), \end{aligned}$$

where $d = \deg(\sigma)$ and the extension of σ is computed by extending the its summands σ_ℓ with Lemma 3.32. The corresponding decomposition into irreducibles can easily be computed using Theorem 3.33.
7. Consider a summand ρ of τ_ℓ. We conjugate each $\rho^{t^k} \uparrow_{T_2} G$ onto $\rho \uparrow_{T_2} G$ using Theorem 3.31, Case 2. Altogether this gives rise to a block diagonal matrix D_2 of size $p(n-d)$. Now equivalent irreducibles are equal and we sort them by degree with a permutation matrix P_2.

$A = D \cdot A' \cdot (\mathbf{1}_{pd} \oplus D_2) \cdot P_2$ is a decomposition matrix for μ with decomposition

$$\mu^A = \left(\bigoplus_{i=0}^{p-1} \bigoplus_{\ell=1}^{r} \lambda_i \cdot \overline{\sigma}_\ell \oplus \bigoplus_{\ell=1}^{s} (\tau_{\ell,1} \uparrow_{T_2} G)^{n_\ell}\right)^{P_2},$$

where $\lambda_i = (t \mapsto \omega_p^i)$ and n_l is the number of summands of τ_ℓ.

Case 7: $\mu^D = \lambda_H \uparrow_T G$ and it exists N with $H \not\leq N \overset{p}{\trianglelefteq} G$ (switch recursion).

1. Decompose $\mu \xrightarrow{D} \lambda_H \uparrow_T G$ using Theorem 3.16.
2. Determine a normal subgroup $N \overset{p}{\trianglelefteq} G$ using a composition series of G. It is $H \not\leq N$ since Case 4 did not apply and hence $G = HN$.
3. Decompose $(\lambda_H \uparrow_T G) \downarrow N \xrightarrow{M} (\lambda_{H \cap N} \uparrow_{T_1} N)$ by Corollary 3.10 ($\lambda_{H \cap N} = \lambda_H \downarrow H \cap N$) and $\lambda_{H \cap N} \uparrow_{T_1} N \xrightarrow{B} \bigoplus_{i=1}^m R_i$, $R_i = \rho_i^{k_i}$ by recursion.
4. Determine which of the ρ_i have an extension to G (cf. Theorem 3.31) which is equivalent to $\rho_i^t \cong \rho_i$. We decide this by computing the permutation π_t arising from t permuting the conjugacy classes of N (through conjugation). Then $\rho_i^t \cong \rho_i$ if and only if the list of character values of ρ_i is invariant under π_t.
5. Conjugate the non-extensible ρ_i such that $\rho_i \cong \rho_j^{t^k}$ implies equality. This is done by using Theorem 3.26 and gives rise to a block diagonal matrix D_1.
6. Compute a permutation matrix P that orders the homogeneous components $R = R_i$ such that inner conjugate non-extensible components are ordered adjacently as $R, R^t, \ldots, R^{t^{p-1}}$.
7. Decompose $\lambda_H \uparrow_{T_1} G \xrightarrow{B \cdot D_1 \cdot P_1} \bigoplus_{j=1}^{m'} \psi_j$. For each j, either (1) $\psi_j \downarrow N = R$, $R = \rho^n$ for an extensible ρ or (2) $\psi_j \downarrow N = R \oplus R^t \oplus \ldots \oplus R^{t^{p-1}}$, $R = \rho^n$ for a non-extensible $\rho = \rho_i$ (cf. Theorem 3.34).
8. Case (1), $\psi_i \downarrow N = R$. Extend $R = \rho^n$ by $\bigoplus_{j=1}^p \overline{\rho}_j^{\ell_j} \cong \psi_i$ using Lemma 3.32. The multiplicities ℓ_i can be determined from the character of ψ_i. Decompose ψ_i with $A_i = C_i \otimes \mathbf{1}_{\deg(\rho)}$ into $\bigoplus_{j=1}^p \overline{\rho}_j^{\ell_j}$ using Theorem 3.37. Set $Q_i = \mathbf{1}_{\deg(R)}$.

9. Case (2), $\psi_i \downarrow N = R \oplus R^t \oplus \ldots \oplus R^{t^{p-1}}$, $R = \rho^n$. Decompose ψ_i with $A_i = (\bigoplus_{j=1}^{n} C_j \otimes \mathbf{1}_{\deg(\rho)}) \cdot Q_i$ into $(\rho \uparrow_T G)^n$ using Theorem 3.38. Q_i is a permutation matrix.
10. Order the irreducibles by degree with a permutation matrix P_2.

$M \cdot B \cdot D_1 \cdot P_1 \cdot \left(\bigoplus_{i=1}^{m'} A_i\right) \cdot Q \cdot P_2$ is a decomposition matrix for μ, where Q is the direct sum of the Q_i.

The first thing to note on the algorithm is that the essential steps are given by the Cases 2, 3, 6, and 7. Case 3 reduces to the transitive case in which $\mu \xrightarrow{D} \lambda_H \uparrow G$ can be written as an induction. Since G is solvable, we now find a normal subgroup $N \trianglelefteq G$ of prime index p with either $H \leq N$ and use induction recursion (Case 6) to recurse, or $H \not\leq N$ and use switch recursion (Case 7) to recurse. Induction recursion reduces the degree of the representation and the size of the group, switch recursion reduces only the size of the group. Hence, invoking only these four cases, the algorithm terminates.

Decomposing into an outer tensor product, if possible, yields a simpler decomposition matrix, however requires the computation of all normal subgroups (in the non-abelian case). In the actual implementation, this case can be deactivated on calling the function. In Section 4.3) we show the influence of this on runtime. Reduction to a faithful representation (Case 1) speeds up decomposition by restricting to the smallest possible group represented by a given representation. Abelian groups have a large number of subgroups, which makes the decomposition into an outer tensor product inefficient. Case 4, together with Case 1, reduces the abelian case to regular representations, which decompose into an outer tensor product, as the group into a direct product (Corollary 3.21). The latter decomposition can be done efficiently (cf. the GAP 3 function `IndependentGeneratorsAbelianPermGroup`). The correctness of the algorithm follows from the theorems on which it is based and we get:

Theorem 4.1 *Algorithm 4.1 terminates and is correct.*

Note that by far the most expensive part of the algorithm is the switch recursion, Case 7, because the “conquer part” requires to perform a conjugation (in Step 7). In all other cases, the irreducibles as well as the decomposition matrix are determined by mere construction, dealing only with small matrices (compared to the degree of the representation). Switch recursion is needed for the decomposition of $\lambda_H \uparrow G$ if and only if H is not subnormal in G, i.e., H is not contained in any composition series of G.

The algorithm is implemented in the function `DecompositionMonRep` contained in the package AREP (cf. Section 5).

4.2. An Example

As an example we consider the group $G = \mathsf{SL}(2,3)$, which is a semidirect product of $\mathsf{Z}_3 = \langle r \mid r^3 = 1\rangle$ with the quaternion group $\mathsf{Q}_8 = \langle s, t \mid s^4 = t^4 = 1,\ s^t = s^{-1}\rangle$ defined by $\mathsf{SL}(2,3) = \langle r, s, t \mid s^r = t^{-1}, t^r = st\rangle$. Let $\mu = 1_{\mathsf{E}} \uparrow_T G$ be the regular representation with transversal

$$\begin{aligned} T \quad = \quad & (1, r, r^2, r^2s, s, rs, rt, r^2t, t, sr, s^2t, rsr, rs^2, \\ & r^2s^2, s^2, rtr, st, tr, rts, ts, sr^2, s^3, tr^2, str). \end{aligned}$$

Decomposing μ with Algorithm 4.1 leads to Case 6 (induction recursion). We work out this case following its seven steps as explained in Algorithm 4.1.

1. The trivial subgroup E is the normal closure of itself and we compute $N = \mathsf{Q}_8$ as the only normal subgroup of prime index, $p = 3$.

2. Using Theorem 3.2 we decompose μ into a double induction $1_{\mathsf{E}} \uparrow_T G \xrightarrow{M} (1_{\mathsf{E}} \uparrow_{T_1} N) \uparrow_{T_2} G$ with transversals $T_1 = (1, s, t, s^2t, s^2, st, ts, s^3)$, $T_2 = (1, r, r^2)$ and conjugating matrix

$$M = [(2, 9, 3, 17, 6, 15, 5)(4, 20, 7, 16, 24, 14, 21, 18, 11)(8, 22)(12, 23, 19), 24].$$

3. The lower induction is decomposed recursively (not shown here) as $(1_{\mathsf{E}} \uparrow_{T_1} N) \xrightarrow{B} \rho_1 \oplus \rho_2 \oplus \rho_3 \oplus \rho_4 \oplus \rho_5^2$ with decomposition matrix

$$\begin{array}{rcl} B & = & [(2, 5, 3)(6, 8, 7), 8] \cdot \\ & & (\mathbf{1}_2 \otimes ((\mathrm{DFT}_2 \otimes \mathbf{1}_2) \cdot \mathrm{diag}(1, 1, 1, \omega_4) \cdot (\mathbf{1}_2 \otimes \mathrm{DFT}_2))) \cdot \\ & & [(3, 5)(4, 8, 7, 6), (1, 1, 1, 1, 1, 1, -1, 1)] \cdot \\ & & ((\mathrm{DFT}_2 \otimes \mathbf{1}_2) \oplus \mathbf{1}_4) \cdot [(2, 4), 8], \end{array}$$

and irreducible components $\rho_1 = 1_N$, $\rho_2 : \ s \mapsto -1,\ t \mapsto -1$, $\rho_3 : \ s \mapsto -1,\ t \mapsto 1$, $\rho_4 : \ s \mapsto 1,\ t \mapsto -1$, and $\rho_5 : \ s \mapsto [(1, 2), (-1, 1)], t \mapsto \mathrm{diag}(\omega_4, -\omega_4)$.

4. A system of representatives of the conjugacy classes of N is given by $C = (1, s, t, s^2, st)$. The permutation induced by r on C is $\pi_r = (2, 3, 5)$. Let χ_i denote the character of ρ_i given by values on C. We have $\chi_1 = (1, 1, 1, 1, 1)$, $\chi_2 = (1, -1, -1, 1, 1)$, $\chi_3 = (1, -1, 1, 1, -1)$, $\chi_4 = (1, 1, -1, 1, -1)$, $\chi_5 = (2, 0, 0, -2, 0)$, and it is easily seen that χ_1 and χ_5 are invariant under π_r and hence have an extension to G. We set $\sigma = \rho_1 \oplus \rho_5^2$.

5. ρ_2, ρ_3, ρ_4 are inner conjugates with $\rho_4 = \rho_2^t, \rho_3 = \rho_2^{t^2}$. Equality holds since they are of degree 1. Thus D_1 is the identity.

6. We have to permute the irreducibles into the order $\rho_1 \oplus \rho_5^2 \oplus \rho_2 \oplus \rho_4 \oplus \rho_3$, which is accomplished by $P_1 = [(2, 6, 3, 8, 5)(4, 7), 8]$. The permutation

$$P = [(6, 16, 23, 21, 15, 20, 14, 17, 11, 8, 22, 18, 12, 9)(7, 19, 13, 10), 24]$$

maps $(\sigma \oplus \rho_2 \oplus \rho_4 \oplus \rho_3) \uparrow_{T_2} G$ onto the direct sum of the inductions $\sigma \uparrow_{T_2} G \oplus \rho_2 \uparrow_{T_2} G \oplus \rho_4 \uparrow_{T_2} G \oplus \rho_3 \uparrow_{T_2} G$ (Theorem 3.3). An extension of ρ_1 is given by $\overline{\rho}_1 = 1_G$, an extension of ρ_5 by

$$\overline{\rho}_5(r) = \begin{bmatrix} -1/2 - 1/2 \cdot \omega_4 & 1/2 + 1/2 \cdot \omega_4 \\ -1/2 + 1/2 \cdot \omega_4 & -1/2 + 1/2 \cdot \omega_4 \end{bmatrix}$$

which determines $\overline{\sigma} = \overline{\rho}_1 \oplus \overline{\rho}_5^2$ and hence the matrix A'. The 3 irreducible representations arising from the factor group $G/N \cong \mathsf{Z}_3$ are given by $\lambda_i : \ r \mapsto \omega_3^i$, $i = 0, 1, 2$.

7. We conjugate $\rho_2^{t^i} \uparrow_{T_2} G$ onto $\rho_2 \uparrow_{T_2} G$, $i = 1, 2$ using Theorem 3.31, Case 2, which gives rise to $D_2 = [(19, 20, 21)(22, 24, 23), 24]$ (note that $\rho_2(r^3) = 1$). The irreducibles are already sorted by degree.

After simplifications we obtain (with $M = \overline{\rho}_5(r)$)

$$\begin{aligned} A \;=\; & [(2,9,6,15,3,17,5,4,24,13,11,8,21,20,7,10,12,23,22)(14,19,16,18),24]\cdot \\ & (\mathbf{1}_3 \otimes ((\mathbf{1}_2 \otimes ((\mathrm{DFT}_2 \otimes \mathbf{1}_2) \cdot \mathrm{diag}(1,1,1,\omega_4) \cdot (\mathbf{1}_2 \otimes \mathrm{DFT}_2)))\cdot \\ & [(3,5)(4,8,7,6),(1,1,1,1,1,1,-1,1)] \cdot ((\mathrm{DFT}_2 \otimes \mathbf{1}_2) \oplus \mathbf{1}_4)))\cdot \\ & [(2,20,18,19,23,14,8,5)(3,24,15,9,6)(4,16,10,21,12,17,11,22,13,7),24]\cdot \\ & (\mathbf{1}_6 \oplus M \oplus M \oplus \mathbf{1}_1 \oplus M^2 \oplus M^2 \oplus \mathbf{1}_9) \cdot ((\mathrm{DFT}_3 \otimes \mathbf{1}_5) \oplus \mathbf{1}_9)\cdot \\ & [(2,12,4,14,6,3,13,5,15,7,8,9,10,11),24] \end{aligned}$$

as a decomposition matrix for μ (and hence a Fourier transform for G) with corresponding decomposition

$$\mu \xrightarrow{A} 1_G \oplus \lambda_1 \oplus \lambda_2 \oplus \overline{\rho}_5^2 \oplus (\lambda_1 \cdot \overline{\rho}_5)^2 \oplus (\lambda_2 \cdot \overline{\rho}_5)^2 \oplus (\rho_2 \uparrow_{T_2} G)^3.$$

$G = \mathsf{SL}(2,3)$ is the smallest group which is not an M-group, i.e., it has an irreducible representation which cannot be conjugated to be monomial. A Fourier transform for $\mathsf{SL}(2,3)$ can also be found in Maslen and Rockmore (2000) where a different approach—based on double coset factorizations—is used and applied to certain classes of non-solvable groups.

Decomposing μ with the implemented function `DecompositionMonRep` (cf. Section 5) takes 0.4 seconds CPU time on an Athlon, 1100 MHz, running Linux. The decomposition matrix A is generated in exactly the presented form. We present further timings in the next section.

4.3. Runtimes

We implemented Algorithm 4.1 as part of the GAP package AREP (cf. Section 5) in the function `DecompositionMonRep`. GAP (1997) is a computer algebra system that provides a high level language, data types, and a library targeted for symbolic computation with groups.

Determining the cost of Algorithm 4.1 is a difficult task, since it frequently uses high level subroutines, which, in the actual implementation, are provided by GAP. Examples include the computation of the normal closure of a subgroup, a composition series, the stabilizer of a point, the character of a representation, or evaluating homomorphisms. The cost of these routines is usually not provided in the GAP manual. For this reason we restrict ourselves to providing results that illustrate the algorithm's asymptotic runtime behavior and the efficiency of our approach.

Since we want to consider a variety of groups, we use the only common monomial representation (apart from the trivial one) among groups—the regular representation—for our experiment, which implies the construction of a fast Fourier transform. Note that this excludes the use of the "switch recursion" (Algorithm 4.1, Case 7). The computing platform is an Athlon, 1100 MHz, running Linux.

Before we state the runtime results we make the following important remarks.

• Our implementation is in GAP, not in a lower level language like C or Fortran. This naturally impacts the runtimes by one or two order of magnitudes. On the other hand, GAP allows us to readily consider arbitrary groups, which are provided in a GAP data base. In fact, to our best knowledge, we are not aware of experimental results on fast Fourier transform construction that consider such a broad range of groups as shown below.

• The algorithm is not optimized for regular representations, which constitute a very special case, or for special classes of groups, and it makes no assumptions on the base field of the monomial representation to be decomposed (as long as GAP permits the representation).

• The implementation works for every computer representation of the group (e.g., power commutator or permutation representation) that is permitted by GAP.

• The algorithm not only computes the irreducible components for the given monomial representations, but also the decomposition matrix in a structured form. Other approaches that generate Fourier transforms usually only create a complete set of suitable irreducible representations, which implies a fast algorithm for the Fourier transform. Arbitrary monomial representations require a different approach. Generating a complete set of irreducible representations can also be done with AREP and is faster than decomposing a corresponding regular representation.

• The decomposition matrix is simplified during construction using certain rules. As an example, the matrix A in the example in Section 4.2, Step 7, has been generated automatically in precisely the presented form (including the conversion to Latex). The simplification is included in the runtime.

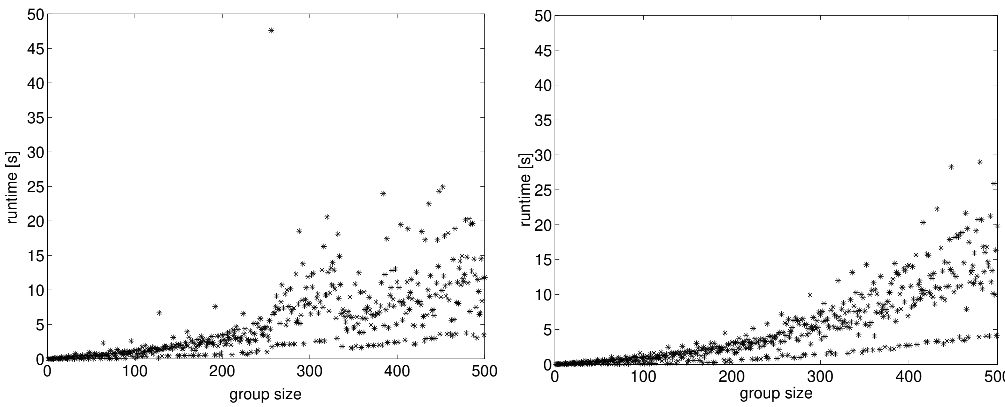
• The series of normal subgroups used for the decomposition is constructed step by step in order to make the algorithm fully recursive. In other words, the algorithms takes as input only the monomial representation. Nothing is precomputed.

Taken together, the design goal was to create an implementation that works under general conditions and can be easily used in the GAP/AREP environment. We proceed with experimental results that illustrate the efficiency of the implementation. As explained above, we decompose regular representations of solvable groups.

In a first experiment, we consider for each $n = 1 \dots 500$ all solvable groups of size n, or a random sample of 100, if there are more than 100, and determine the average runtime for decomposing their regular representation. The result is presented in Figure 3 a). The abscissa carries the group size n, and the ordinate the average runtime in seconds. The outliers of the mainstream correspond to numbers n with a large number of prime factors, e.g. 16, 32, 64, 128, or 48, 96, 192, 384, and are due to Case 5 in Algorithm 4.1, which computes the set of all normal subgroups. As mentioned at the end of Section 4.1, this case can be disabled in the implementation (using a flag), which leads to the somewhat smoother Figure 3 b). The vein below the main stream corresponds to prime numbers n. In this case only one group, Z_n, exists, whose regular representation is handled efficiently.

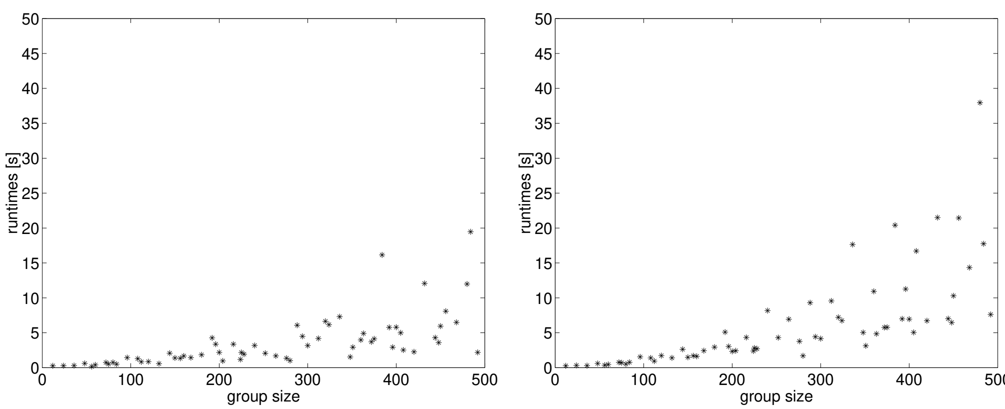
In a second experiment, we repeat the same procedure but restricted to solvable groups G, which are not supersolvable (G is supersolvable if G has a composition series that is even a chief series). Constructing Fourier transforms for these groups (in the sense of creating a complete set of suitable irreducible representations) is, with current methods, more expensive: $O(n^2 \log(n)p)$ (p the largest prime factor of n, Clausen and Müller (2002)) versus $O(n \log^2(n))$ for supersolvable groups (Baum and Clausen (1994), corrected in Clausen and Müller (2002)). These groups only exist for 63 sizes $n \in \{1 \dots 500\}$. In contrast to the previous experiment, we consider, for each n, *all* these groups, independent of their number. The average runtimes for each n are shown in Figure 4, the maximum runtimes are given in Figure 5. In Figure 5 a) we omitted a runtime of 610 seconds for a group of size $n = 384$. As mentioned before, this runtime is due to computing a large number of normal subgroups. Note that the scales in Figure 3 and Figure 4 are equal, whereas Figure 5 has a larger range on the ordinate. Comparing Figure 4 a) and b) shows that using the outer tensor decomposition method can also speed up the decomposition.

Finally, we give an idea of the asymptotic behavior. Figure 6 a) shows the runtimes of



a) with outer tensor product b) without outer tensor product

Figure 3. Average runtime (sec) for decomposing a regular representation of a solvable group of size n.



a) with outer tensor product b) without outer tensor product

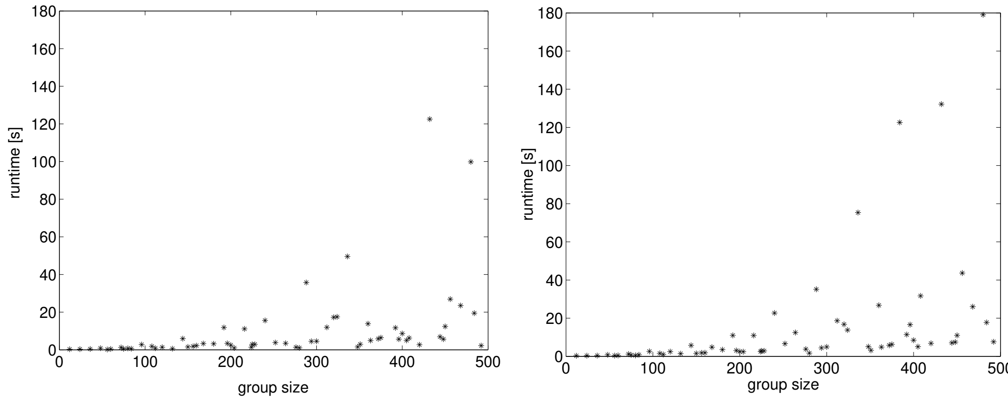
Figure 4. Average runtime (sec) for decomposing a regular representation of a solvable, but not supersolvable, group size of n.

Figure 3 a) divided by n^2, and Figure 6 b) shows the runtimes of Figure 5 b) divided by $n^2 \log(n)p$ (motivated by the upper bound given in Clausen and Müller (2002)), where p is the largest prime factor of n.

We conclude that our algorithm (and its implementation) provides an efficient tool for decomposing monomial representations.

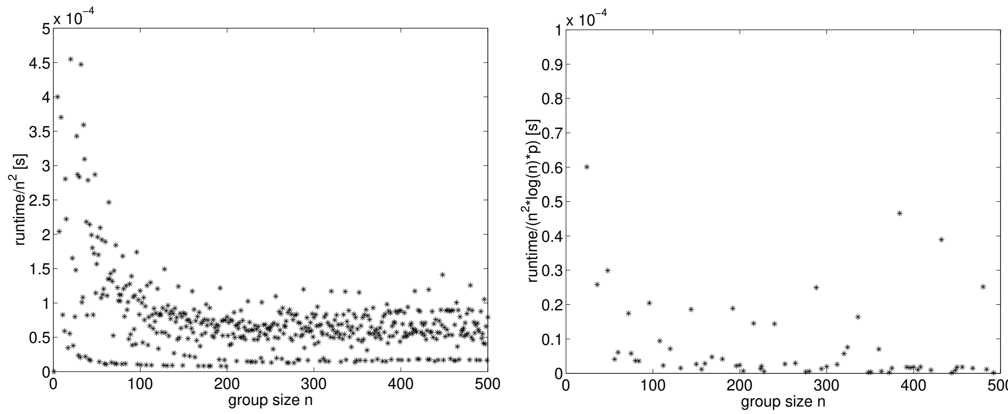
5. AREP—a Package for Constructive Representation Theory

The results of this paper and in particular the algorithm for decomposing monomial representations of solvable groups (cf. Section 4) have been realized in the package AREP (1998) created by Sebastian Egner and the author. AREP is implemented in the language GAP v3.4.4 (1997), a computer algebra system specialized on computational group theory, and has been accepted as a GAP share package. AREP also has been integrated into the SPIRAL system (Moura *et al.* 1998), a library generator for signal processing transforms.



a) with outer tensor product b) without outer tensor product

Figure 5. Maximum runtime (sec) for decomposing a regular representation of a solvable, but not supersolvable, group size of n.



a) Figure 3 b) normalized by n^2 b) Figure 5 b) normalized by $n^2 \log(n)p$

Figure 6. Normalized average runtime (sec) for decomposing a regular representation of size n (left) and normalized maximum runtime for a solvable, but not supersolvable, group of size n (right).

The connection to SPIRAL allows the user to translate the algorithms (e.g., fast Fourier transforms) generated by AREP into C or Fortran code (Egner *et al.* 2001).

The goal of AREP is to provide the data types, infrastructure, and functions for the efficient symbolic computation with structured matrix representations. We briefly survey the main components of AREP.

The main components of AREP are the recursive data types `ARep` and `AMat`, which represent structured representations and matrices, respectively, and the necessary infrastructure for their manipulation. Using this platform, several algorithms, including Algorithm 4.1 have been implemented and are also part of AREP. In the following we briefly survey the main ideas used to implement AREP.

An `ARep` is a GAP record representing a matrix representation. The record contains a number of fields which uniquely characterize a matrix representation, e.g. degree, characteristic, and the represented group always have to be present. There are a number of elementary constructors that allow the user to create an `ARep`, e.g., by specifying the images on a set of generators of the represented group (`ARepByImages`). Furthermore, there are constructors building a higher structured `ARep` from given `AReps` (e.g. `DirectSumARep`, `InductionARep`). The idea is not to immediately evaluate such a construction, but to build an `ARep` representing it. E.g., an `ARep` representing a direct sum has a field `summands` containing the list of summands. Conversion to an (unstructured) matrix representation is performed by calling the appropriate function. On the other side there are functions converting an unstructured, e.g., monomial `ARep`, into a highly structured `ARep`, e.g., a conjugated induction of a representation of degree 1 (cf. Theorem 3.16), which is mathematical *identical* to the original one. Permutation and monomial representations has been given special attention in the package since they are efficiently to store and to compute with and they were the central object of our interest. The decomposition algorithm in Section 4 is realized in the function `DecompositionMonRep` which takes a monomial representation and returns a conjugated direct sum of irreducibles, which is mathematical identical to the input. The highly structured decomposition matrix is represented by an `AMat`, which we explain next.

The data type `AMat` has been created according to the same principle as `ARep`, as a GAP record representing a matrix. Again, there are elementary constructors to create an `AMat`, e.g., `AMatPerm` takes a permutation, a degree, and a characteristic and builds an `AMat` which efficiently represents a permutation matrix. Higher constructors build the product, direct sum, tensor product, etc., of `AMats` and are not evaluated until the appropriate function is invoked. Thus, `AMat` allows the user to construct structured matrices which are efficiently stored and easier to handle than the (mathematical identical) represented matrices, e.g., determinant, trace and inverse can be computed efficiently by using well-known mathematical rules.

For a description of further capabilities of AREP, in particular symmetry-based matrix factorization, we refer the reader to Egner and Püschel (2002) and the AREP manual and web page (Egner and Püschel 1998).

6. Acknowledgements

The main part of the work presented in this paper was developed when the author was with Prof. Thomas Beth at the Institute of Algorithms and Cognitive Systems at the University of Karlsruhe, Germany. The author would like to thank Prof. Thomas Beth for his full support which made this work possible. The author is particularly indebted to Dr. Sebastian Egner (Philips Research Laboratories, Eindhoven) who brought the author's attention to this topic and without whom neither AREP nor this paper would exist. Thanks also to Prof. Jeremy Johnson (Drexel University) for helpful discussions and proofreading.

References

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

Baum, U., Clausen, M. (1994). Computing Irreducible Representations of Supersolvable Groups. *Mathematics of Computation*, 63(207):351–359.

Beth, T. (1984). *Verfahren der Schnellen Fouriertransformation*. Teubner.

Clausen, M. (1988). *Beiträge zum Entwurf schneller Spektraltransformationen (Habilitationsschrift)*. University of Karlsruhe.

Clausen, M. (1997). A Direct Proof of Minkwitz's Extension Theorem. *AAECC*, 8:305–306.

Clausen, M., Baum, U. (1993). *Fast Fourier Transforms*. BI-Wiss.-Verl.

Clausen, M., Müller, M. (2002). Generating Fast Fourier Transforms of Solvable Groups. *Journal of Symbolic Computation special issue on "Computer Algebra and Signal Processing"*. To appear.

Cooley, J. W., Tukey, J. W. (1965). An Algorithm for the Machine Calculation of Complex Fourier Series. *Mathematics of Computation*, 19:297–301.

Curtis, W. C., Reiner, I. (1962). *Representation Theory of Finite Groups*. Interscience.

Curtis, W. C., Reiner, I. (1981). *Methods of Representation Theory*, volume 1. Interscience.

Diaconis, P., Rockmore, D. (1990). Efficient computation of the Fourier transform on finite groups . *Journal AMS*, 3(2):297–332.

Dixon, J., Mortimer, B. (1996). *Permutation Groups*. Springer.

Dornhoff, L. (1971). *Group Representation Theory*. Pure and Applied Mathematics. Dekker New York.

Egner, S., Johnson, J., Padua, D., Püschel, M., Xiong, J. (2001). Automatic Derivation and Implementation of Signal Processing Algorithms. *ACM SIGSAM Bulletin Communications in Computer Algebra*, 35(2):1–19.

Egner, S., Püschel, M. (1998). *AREP – A Package for Constructive Representation Theory and Fast Signal Transforms*. GAP share package. http://www.ece.cmu.edu/∼smart/arep/arep.html.

Egner, S., Püschel, M. (2001). Automatic Generation of Fast Discrete Signal Transforms. *IEEE Transactions on Signal Processing*, 49(9):1992–2002.

Egner, S., Püschel, M. (2002). Symmetry-Based Matrix Factorization. *Journal of Symbolic Computation special issue on "Computer Algebra and Signal Processing"*. To appear.

GAP (1997). *GAP – Groups, Algorithms, and Programming*. The GAP Team, School of Mathematical and Computational Sciences, U. St. Andrews, Scotland. http://www-gap.dcs.st-and.ac.uk/∼gap/.

Heideman, M., Johnson, D., Burrus, C. (1985). Gauss and the History of the Fast Fourier Transform. *Archive for History of Exact Sciences*, 34:265–277.

Maslen, D., Rockmore, D. (1995). Generalized FFTs – A survey of some recent results. In *Proceedings of IMACS Workshop in Groups and Computation*, volume 28, pages 182–238.

Maslen, D., Rockmore, D. (2000). Double coset decompositions and computational harmonic analysis on groups. *Journal of Fourier Analysis and Applications*, 6(4):349–388.

Minkwitz, T. (1993). *Algorithmensynthese für lineare Systeme mit Symmetrie*. PhD thesis, Universität Karlsruhe, Informatik.

Minkwitz, T. (1995). Algorithms Explained by Symmetry. *Lecture Notes on Computer Science*, 900:157–167.

Minkwitz, T. (1996). Extension of Irreducible Representations. *AAECC*, 7:391–399.

Moura, J. M. F., Johnson, J., Johnson, R. W., Padua, D., Prasanna, V., Püschel, M., Veloso, M. M. (1998). SPIRAL: A Generator for Platform-Adapted Libraries of Signal Processing Algorithms. http://www.ece.cmu.edu/∼spiral/.

Püschel, M. (1998). *Konstruktive Darstellungstheorie und Algorithmengenerierung*. PhD thesis, Universität Karlsruhe, Informatik. Also available in English as Tech. Rep. Drexel-MCS-1999-1, Drexel University, Philadelphia.

Rockmore, D. (1990). Fast Fourier Analysis for Abelian Group Extensions. *Advances in Applied Mathematics*, 11:164–204.

Rockmore, D. (1995). Some applications of generalized FFT's. In *Proceedings of DIMACS Workshop in Groups and Computation*, volume 28, pages 329–370.

Terras, A. (1999). *Fourier Analysis on Finite Groups and Applications*. Cambridge University Press.

Wielandt, H. (1964). *Finite Permutation Groups*. Academic Press.