Discrete Fourier Transform over Schurian Schemes

Hantao Yu

Department of Mathematics

University of California, San Diego

Supervisor

Kiran Kedlaya, Chris Umans (Caltech)

In partial fulfillment of the requirements for the degree of

Honors Bachelor of Science in Mathematics

June, 2022

Acknowledgements

First of all, I would like to thank Prof. Chris Umans for advising me in this research project. Chris is a wonderful advisor who convinces me to study theoretical computer science in graduate school. His passion and dedication have always been inspiring.

Second of all, I would like to thank Prof. Kiran Kedlaya for teaching me commutative algebra and number theory, and making this thesis possible. I always learn new things when talking to him.

I am grateful to Prof. Shachar Lovett, Prof. Daniel Rogalski, Prof. Steven Sam, and Prof. Jiapeng Zhang for teaching me various topics in mathematics and computer science. I am also grateful to Max Hopkins and Ruth Luo for advising my research projects.

In addition, I want to thank all my friends for their support.

Finally, I would like to thank my parents for their enormous amount of support and encouragement throughout my undergraduate years.

Abstract

The Discrete Fourier Transform (DFT) is a fundamental linear map that can be defined with respect to any finite group. Recent work [1] has obtained fast algorithms for computing the DFT over any finite group of order n, using approximately $O(n^{\omega/2})$ arithmetic operations, where ω is the exponent of matrix multiplication.

In this work, we consider the DFT over more general algebraic objects, *association schemes*, which contain groups as a special case. In this more general setting, even the basic recursive algorithm encounters complications. In this paper we focus on Schurian association schemes, which can be defined by a group G and a subgroup H. We obtain several algorithms for computing the DFT with respect to these schemes. For a Schurian scheme of rank n, each one can potentially have a complexity close to $O(n^{\omega/2})$ (which is optimal if $\omega = 2$), with a multiplicative overhead that depends on structural properties of the G and H. In particular, our algorithms work well in any of the following five situations:

1. $N_G(H)$ has small index.
2. There exists proper subgroup L containing H such that $LN_G(H) = G$.
3. There exists large normal proper subgroup P containing H.
4. There exists normal proper subgroup P and subgroup K, both containing H, such that $PK = G$.
5. There is a large normal subgroup N contained in H.

In order to obtain these algorithms, we work out an explicit description of the irreducible representations of the Schurian association scheme

defined by G and H, in terms of the irreducible representations of G, which may be of independent interest.

Contents

Chapter 1

Introduction

1.1 Background

The DFT dates back to Cooley-Tukey and probably even further. It has many applications in signal processing, fast matrix multiplication, polynomial multiplication etc. When people talk about the DFT they often implicitly mean the DFT with respect to a cyclic group. In fact the DFT makes sense with respect to any finite group. To state it we need the notion of irreducible representations of a the group G. A d dimensional representation is a homomorphism from the group G into the group of invertible $d \times d$ matrices; it is irreducible if there are no invariant subspaces. A finite group G has a finite set of inequivalent irreducible representations, denoted $\mathrm{Irr}(G)$.

Given a vector $\alpha \in \mathbb{C}[G]$, the DFT with respect to G is a linear transform that takes c to

$$\mathrm{DFT}(\alpha) = \sum_{g \in G} c_g \bigoplus_{\rho \in \mathrm{Irr}(G)} \rho(g).$$

It is known that

$$\sum_{\rho \in \mathrm{Irr}(G)} (\dim \rho)^2 = |G|,$$

and thus the trivial algorithm takes $O(|G|^2)$ time because we are summing up $|G|$ matrices and each has $|G|$ nonzero entries. In contrast, the classical FFT algorithm computes a DFT with respect to the cyclic group of order $n = 2^k$ in $O(n \log n)$ time. Efforts beginning in the 1980s [2, 3] have aimed to extend this to

general groups. Recently, [1] achieved $O(|G|^{\omega/2+o(1)})$ time algorithms for all finite groups, where ω is the exponent of matrix multiplication.

The most general setting in which one can describe a DFT is the case of finite dimensional semi-simple algebras over $\mathbb{C}$. Such an object is specified by a basis $b_1, \ldots, b_n$ and structure constants $p_{i,j,k} \in \mathbb{C}$ with multiplication defined by

$$b_i b_j = \sum_{k=1}^{n} p_{i,j,k} b_k.$$

If the algebra A is semi-simple, then it has a finite set of irreducible representations $\mathrm{Irr}(A)$, and it makes sense to compute a DFT with respect to A: given a vector $\alpha \in \mathbb{C}^n$, a DFT with respect to A with its basis $b_1, \ldots, b_n$ is a linear transform that takes α to

$$\sum_{i=1}^{n} \alpha_i \bigoplus_{\rho \in \mathrm{Irr}(A)} \rho(b_i)$$

Notice that in the case of groups the group algebra $\mathbb{C}[G]$ is semi-simple and the natural basis is the basis of group elements, with multiplication according to the group multiplication law. Once again the trivial algorithm takes $O(n^2)$ operations, and we aim to achieve $O(n^{\alpha+o(1)})$ operations, where with α close to one.

Association schemes are a well-studied generalization of groups with an associated semisimple algebra called the *adjacency algebra*, and in this paper we aim to find fast DFT algorithms with respect to this adjacency algebras. We succeed in doing so for certain association schemes called Schurian schemes; these are defined in terms of a pair of groups $H \subseteq G$, and our algorithms are fast when there is an intermediate sequence of subgroups between H and G satisfying certain conditions.

Fast DFT algorithms are typically recursive and in the case of groups, they descend from the full group to subgroups. The analogous strategy for association schemes already encounters significant difficulties. We overcome them in certain cases and in the next section we describe the challenges in more detail, and our ideas for overcoming them.

1.2 Main technique ideas

A fast recursive DFT with respect to a semisimple algebra A, with basis $b_1, \ldots, b_n$, generally works by identifying a closed subalgebra A' spanned by $\{b_i : i \in S\}$ for a set $S \subseteq [n]$. It performs a DFT recursively with respect to A' and then combines these to obtain the final result. In order to combine the A'-DFTs, we need to identify a set of algebra elements ("translations") $t_1, t_2, \ldots, t_k$ with the key property that

$$\{b_i t_j : i \in S, 1 \leq j \leq k\}$$

linearly spans all of A.

In the case of a finite group G, the subalgebra is given by a subgroup H, and the translations can easily be taken to be a set of distinct coset representatives of H in G.

In the present case of association schemes, it is not hard to identify the analog of a subgroup; this is called a *closet set* in the association scheme literature. But given a rank n association scheme and rank k closed set, we don't even know if there exists, in general, a set of $\frac{n}{k}$ translations with the required spanning property (as there are in the group case). In this paper, we find that in the special case of Schurian association schemes defined by $H \subseteq G$, with some additional properties involving intermediate subgroups, there is an explicit description of a small set of translations with the required spanning property (although not, in general, as small as the optimal $\frac{n}{k}$).

Even when one has a set of translations, the recursive algorithm must perform a linear transformation determined by the way in which the

$$\{b_i t_j : i \in S, 1 \leq j \leq k\}$$

linearly span A. A general linear transformation at this step would already cost $O(n^2)$ operations which is as bad as the trivial algorithm, so we need to carefully choose our translations so that this linear transformation can be computed rapidly. In the cases we study in this paper, we are able to show that the support of $b_i t_j$ is at most one, for each i, j, and this implies that the linear transformation can be computed in linear time.

1.3 Main results

To state our main results, we need notation for double-cosets: given a group G and subgroups H, K we denote by $K\backslash G/H$ the set of K, H double cosets in G. We will substitute $H\backslash G/H$ by $G/\!\!/H$ for simplicity.

Our key contributions include a way to reduce a $G/\!\!/H$-DFT to a G-DFT and use it to design our subscheme reductions. Furthermore, we give two recursive algorithms that work in different scenarios. Throughout this paper, we use $\mathrm{DC}(G/\!\!/H)$ to denote the complexity of computing a $G/\!\!/H$-DFT.

Theorem 1.3.1 (Reduction to Group-DFTs)**.** *A* DFT *over* $\mathbb{C}[G/\!\!/H]$ *can be computed using the same number of operations as a* DFT *over* $\mathbb{C}[G]$*.*

Theorem 1.3.1 helps us reduce a $G/\!\!/H$-DFT to a G-DFT. One immediate consequence is that if N is a subgroup of H, then a $G/\!\!/N$-DFT will give us a corresponding $G/\!\!/H$-DFT with no cost. Formally,

Theorem 1.3.2. *If* $N \subset H$*, then*

$$\mathrm{DC}(G/\!\!/H) \leq \mathrm{DC}(G/\!\!/N).$$

However, this reduction does not necessarily give us a proper set of translations as in the group case. We show that in certain scenarios, we could find proper translations to apply our subscheme reductions.

Theorem 1.3.3 (Single Subscheme Reduction)**.** *If* $H < K \triangleleft G$*, then*

$$\mathrm{DC}(G/\!\!/H) \leq |G/K|\mathrm{DC}(K/\!\!/H) + O(|G/K||G/\!\!/H|^{\omega/2+\epsilon})$$

for any $\epsilon > 0$*.*

Our Single Subscheme Reduction works well if there exists a large normal subgroup K containing H. In that case, $|G/K|$ is small so our algorithm is efficient. Some examples include p-groups, where we are guaranteed a subnormal series containing H by keep taking the normalizer and recurse on the subnormal series. However, this is not the case for many G, H, so we design the Double Subscheme

Reduction to tackle the cases when no large normal subgroup K containing H exists.

Theorem 1.3.4 (Double Subscheme Reduction)**.** *If there exists subgroups K, P both containing H such that K is normal in G and $KP = G$, then*

$$\mathrm{DC}(G/\!\!/H) \leq |P/\!\!/H|\cdot\mathrm{DC}(K/\!\!/H) + |K/\!\!/H|\cdot\mathrm{DC}(P/\!\!/H)+ O\left(|G/\!\!/H|^{\omega/2+o(1)}+(|P/\!\!/H||K/\!\!/H|)^{\omega/2+o(1)}\right).$$

For Double Subscheme Reduction, we do not require K to be large. It is effective for groups including the wreath-product of a large simple group with S_n.

Besides subscheme reductions, we find two natural recursive algorithms depending on the normalizer of H. If $N_G(H)$ has small index in G, H is close to being normal, and we show that

Theorem 1.3.5 (First recursive algorithm)**.** *Given $H \triangleleft K < G$,*

$$\mathrm{DC}(G/\!\!/H) \leq |K\backslash G/H||K/H|^{\omega/2+\epsilon}+O(|K\backslash G/H||G/\!\!/H|^{\frac{\omega}{2}+\epsilon})$$

for all $\epsilon > 0$.

When $N_G(H)$ is small (but not H itself), then for certain groups we can find a small index subgroup L such that $LN_G(H) = G$, leading to our second recursive algorithm:

Theorem 1.3.6 (Second recursive algorithm)**.** *If there exists a subgroup L such that $H \subset L$ and $LN_G(H) = G$, then*

$$\mathrm{DC}(G/\!\!/H) \leq |G/L|\mathrm{DC}(L/\!\!/H) + |G/L||G/\!\!/H|^{\omega/2+\epsilon}$$

for all $\epsilon > 0$.

In other words, if L is "orthongonal" to $N_G(H)$, then we could use $L/\!\!/H$ to do the recursive algorithm even when L is not normal in G.

1.4 Related work

Beth[2] and Clausen[3] initiated the research program of trying to obtain fast DFTs with respect to any finite group in 1980s. Initially, fast algorithms for abelian groups and some special classes like supersolvable groups and symmetric groups. The goal is to obtain "near-linear" algorithms, meaning the algorithm takes $O(|G|^{1+o(1)})$ operations. The algorithms of computing DFTs usually involve matrix multiplications, so we usually consider algorithms with $O(|G|^{\omega/2+o(1)})$ for $\epsilon > 0$ operations as good ones as when $\omega = 2$ these algorithms will be near-linear. Baum [4] achieved near-linear algorithms for abelian and supersolvable groups, and Clausen[3] achieved near-linear algorithms for symmetric and alternating groups. Later in 1998, Maslen [5] gave an improvement of computing DFTs over symmetric groups.

Groups with no large subgroups such as $\mathrm{SL}_2(\mathbb{F}_q)$ are the problematic for the basic recursive algorithms. Recently, Hsu and Umans[1] devised the "double subgroup" reduction which handles these cases, and this led to the first improvement on the exponent for general groups. They were able to achieve $O(|G|^{\omega/2+\epsilon})$ for linear groups, and $O(|G|^{\sqrt{2}})$ for all finite groups. In 2019, Umans[6] extended this work to handle all finite groups with exponent $\omega/2$, by devising a more complicated "triple subgroup" reduction, thus concluding the research of DFTs over finite groups. Interestingly these last two results rely on the Classification Theorem to obtain certain structural properties needed for the algorithms, for all finite groups.

Moving beyond groups to association schemes, there has been comparatively little work. In 2016, Maslen, Rockmore, and Wolf [7] computed the DFT over semi-simple algebras including Brauer algebra, BMW algebra and Temperley-Lieb algebra. However, no work has directly target the DFTs over the double coset algebra, i.e. Schurian scheme.

Chapter 2

Association schemes, adjacency algebras, and DFTs

In this section we give some basic definitions and background content on Association schemes and their adjacency algebras.

Association schemes are objects that have been well studied after Bose and Shimamoto introduced the concept in 1952 [8]. They generalize groups, and are defined as follows:

Definition 2.0.1 (Association Scheme)**.** *A* association scheme consists of a set X with a partition of $X \times X$ into $n+1$ binary relations $\mathcal{R} = \{R_0, \ldots, R_n\}$ which satisfy:

1. $R_0 = \{(x, x) : x \in X\}$
2. The *dual relation* of R_i, defined by $\{(x, y) : (y, x) \in R_i\}$ is in $\mathcal{R}$.
3. If $(x, y) \in R_k$, the number of $z \in X$ such that $(x, z) \in R_i$ and $(z, y) \in R_j$ is a constant p_{ijk} called the *intersection number* depending only on i, j, k (and not x, y).

For example, any group G is an association scheme there is a relation R_g for each $g \in G$ defined as

$$R_g = \{(gh, h) : h \in G\}$$

Clearly this satisfies the first axiom; the second one is satisfied because $R_{g^{-1}}$ is the dual of R_g; and the third axiom is routine to check.

A broad class of association schemes are the *Schurian* association schemes, which are defined in terms of groups:

Definition 2.0.2 (Schurian association scheme)**.** *Given a group G with a subgroup H, let X be the set of left-cosets of H in G. Group G acts transitively by left-multiplication on X and one can extend this to an action on $X \times X$ by having G act identically on each copy of X. The orbits of this action form an association scheme, called the* Schurian scheme of G, H*.*

One can define an algebra from any association scheme called the *adjacency algebra*. This is simply the matrix algebra generated by the natural incidence matrices, one for each relation in the scheme; i.e., each relation R_i defines an $|X|\times|X|$ adjacency matrix A_i where

$$A_i(x,y) = \begin{cases} 1, \text{ if } (x,y) \in R_i \\ 0, \text{otherwise} \end{cases}$$

Observe that the intersection numbers p_{ijk} become the structure constants of multiplication in the adjacency algebra:

$$A_i A_j = \sum_{k=0}^{n} p_{ij}^k A_k$$

as a consequence of the third axiom.

2.1 Irreducible representations and the DFT

A representation of an algebra $\mathcal{A}$ is a homomorphism from $\mathcal{A}$ into matrices $\mathbb{C}^{d\times d}$. The dimension of such a representation is d. Such a representation naturally acts on $\mathbb{C}^d$, and it is said to be *irreducible* if there is no invariant subspace under this action. Two representations ρ, τ are equivalent if they have the same dimension and there is a change of basis matrix $T \in \mathbb{C}^{d\times d}$ such that $\rho(x) = T\tau(x)T^{-1}$.

For this work, the key fact is that adjacency algebras of association schemes are *semisimple* which means that each such algebra $\mathcal{A}$ has a finite set of inequivalent irreducible, representations $\rho_1, \ldots, \rho_k$ with $\dim(\mathcal{A}) = \sum_i \dim(\rho_i)^2$.

We now can define the DFT with respect to an association scheme:

Definition 2.1.1. *Let A be an association scheme with relations $R_0, R_1, \dots R_t$ and associated incidence matrices $A_1, \dots, A_t$, and let $\mathrm{Irr}(\mathbb{C}[A])$ be the complete set of inequivalent, irreducible representations of its adjacency algebra $\mathbb{C}[A]$. The associated DFT is the linear map that takes $\alpha \in \mathbb{C}^t$ to*

$$\sum_{i=1}^{t} \alpha_i \bigoplus_{\rho \in \mathrm{Irr}(\mathbb{C}[A])} \rho(A_i)$$

Observe that this map can easily be computed with $O(t^2)$ operations, by simply writing out the big sum for each matrix entry, since $\sum_j \dim(\rho_j)^2 = t+1$. Our goal will be to compute it using closer to $O(t)$ operations, which would be optimal. We present several algorithms that work well in different scenarios in Chapter 4 and Chapter 5.

2.2 Double coset algebras

Our algorithms will make heavy use of the fact that the adjacency algebra of a Schurian association scheme of G, H is isomorphic to the *H, H-double coset algebra*, which we define next.

Definition 2.2.1. *For a group G and subgroups $H, K \subset G$, the (H, K)-double coset of $x \in G$ is*

$$HxK = \{hxk : h \in H, k \in K\}$$

When $H = K$, HxH are called the H-double coset of x. The set of all (H, K)-double cosets is denoted as $H\backslash G/K$, and we use $G/\!\!/H$ as shorthand for $H\backslash G/H$.

Here are some useful properties of double cosets:

1. The double coset HxK is an equivalence class that partitions G. In other words, $x \sim y$ iff $HxK = HyK$.
2. If $H = \{1\}$, then $H\backslash G/K = G/K$.
3. If $H \triangleleft G$, then $HxK = x(HK)$ so $H\backslash G/K = G/(HK)$ (HK is a subgroup of G because H is normal). In particular, if $H < K$, $H\backslash G/K = G/K$.

It is not hard to verify that the product of two double cosets $(HxH) \cdot (HyH)$ (counting group elements in their multiplicities) is an integer linear combination of H, H-double cosets; hence the H, H-double cosets generate an algebra, denoted $\mathbb{C}[G/\!\!/H]$.

The double coset algebra $G/\!\!/H$ forms a Schurian scheme on the set of single G/H cosets where each double coset HgH correspond to a binary relation R_g:

$$(xH, yH) \in R_g \text{ iff } x^{-1}y \in HgH.$$

In other words,

$$R_g = \{(xH, yH) : Hx^{-1}yH = HgH\}.$$

Recall that multiplication is the same as in the algebra: for any $HaH, HbH \in G/\!\!/H$,

$$R_a \cdot R_b = \sum_{R_c \in G/\!\!/H} p_{abc} R_c,$$

where we use p_{abc} as a shorthand for p_{R_a,R_b,R_c} for simplicity.

Note that the group G is an association scheme in this framework by taking H to be the trivial subgroup, and also that if H is normal in G, then this association scheme is isomorphic to the quotient group G/H (because the H, H double cosets coincide with H-cosets in this case). The following is a standard fact that can be verified by writing out the product of two double-cosets:

Proposition 2.2.2. *The adjacency algebra of the Schurian association scheme defined by G, H is isomorphic to the (H, H)-double coset algebra in G.*

For the remainder of this paper, we will explicitly describe our DFTs for Schurian association schemes defined by G, H as DFTs over the H, H-double coset algebra; i.e. given $\alpha \in \mathbb{C}[G/\!\!/H]$, we wish to compute

$$\sum_{b \in G/\!\!/H} \alpha_b \bigoplus_{\rho \in \mathrm{Irr}(\mathbb{C}[G/\!\!/H])} \rho(b).$$

In the group case, any product of $h, k \in G$ will have a single support, namely hk. However, this is not the case in the double coset algebra. For any $HaH, HbH \in$

$G/\!\!/H$,

$$R_a \cdot R_b = \sum_{R_c \in G/\!\!/H} p_{abc} R_c.$$

It is routine to check that $p_{abc} \neq 0$ iff $HcH \subset HaHbH$. In other words, the support of $R_a \cdot R_b$ is all the R_c such that $HcH \subset HaHbH$. Given that, we can show a useful fact that will be used later: if $HaH = aH \subset K$ where H is normal in K, then $R_a \cdot R_b$ will have a single support, namely R_{ab}, for any $R_b \in G/\!\!/H$.

Proposition 2.2.3. *If $HaH \in K/\!\!/H$ where H is normal in K, then $R_a \cdot R_b$ has a single support for any $R_b \in G/\!\!/H$.*

Proof. By our argument above, the support of $R_a \cdot R_b$ is all the R_c such that $HcH \subset HaHbH = HabH$ as $a \in K$. That means R_{ab} is the only support for $R_a \cdot R_b$. □

Chapter 3

Representations of double-coset algebras

In this section we work out a concrete and explicit description of the irreducible representations of double coset algebra $\mathbb{C}[G/\!\!/H]$ in terms of the irreducible representations of G, by choosing a favorable basis.

We are given a group G and a subgroup H. Recall that a d-dimensional representation ρ of G acts naturally on $V = \mathbb{C}^d$. It a basic fact of representation theory, that the *restriction* of ρ to H decomposes as the direct sum of irreps of H. Concretely, this means that there is some basis for V in which the matrix $\rho(h)$ is block diagonal (for all $h \in H$), with block sizes that correspond to the irreps of H occurring in the restriction of ρ to H. One can choose this favorable basis with respect to each irrep $\rho \in \mathrm{Irr}(G)$, and the resulting basis is called an *H-adapted* basis. This will be the basis we choose to work in for this section, and the subsequent sections that depend on it.

In general, an irrep ρ of G acting on V, when restricted to H, will decompose into 0 or more trivial irreps of H, and other irreps of H. We need to single out the part of the direct sum decomposition of V corresponding to the trivial irreps; we will call this V_0. In other words, $V = V_0 \oplus V_1$ and in an H-adapted basis, $\rho(h)$ acts trivially on V_0 for all $h \in H$. When H is clear from context, will always use V_0 to refer to this direct-summand of the vector space.

We have the following lemma regarding H, H-invariant functions:

Lemma 3.0.1. *Let $\alpha \in \mathbb{C}[G]$ be H, H-*invariant *(i.e., $\alpha h = \alpha = h\alpha$ for all $h \in H$)*

and let $\rho \in \mathrm{Irr}(G)$ *act on* $V = V_0 \oplus V_1$. *Then*

$$\rho(\alpha) = \sum_{g \in G} \alpha_g \rho(g)$$

acts by 0 *on* V_1.

Proof. Because α is H, H-invariant, letting $w = \frac{1}{|H|} \cdot \sum_{h \in H} h$, we have

$$\rho(\alpha) = \rho(w\alpha w) = \rho(w)\rho(\alpha)\rho(w).$$

The restriction of ρ to H decomposes into irreps of H. Now, for $\tau \in \mathrm{Irr}(H)$, we have $\tau(w) = 0$ if τ is not the trivial irrep; if it is the trivial irrep, then $\tau(w) = 1$. Since we are working in an H-adapted basis, ρ restricted to H respects the direct sum decomposition of $V = V_0 \oplus V_1$ described above, and we have just argued that it acts by 0 on V_1. □

Recall that ρ respects the direct sum decomposition since we are working in an H-adapted basis, i.e, ρ is block-diagonal. In concrete terms, the above lemma then implies that the block in $\rho(\alpha)$ corresponding to V_1 is zero, so the only non-zeros in $\rho(\alpha)$ are in the block corresponding to V_0.

Let d be the dimension of V_0. We define a map ρ_H from $G/\!\!/H$ into invertible $d \times d$ matrices as follows: given a double coset $b \in G/\!\!/H$, define the H, H-invariant function $w_b = \sum_{g \in b} g$, and then define

$$\rho_H(b) = \rho_0(w_b),$$

where ρ_0 is the part of ρ acting on V_0, the direct summand corresponding to the occurrences of the trivial irrep of H in the restriction of ρ to H. Extend ρ_H linearly to $\mathbb{C}[G/\!\!/H]$.

A key observation is that as one runs through all irreps ρ of G, the ρ_H are exactly the irreps of $\mathbb{C}[G/\!\!/H]$.

Lemma 3.0.2. *Let* $\rho \in \mathrm{Irr}(G)$ *and define the map* ρ_H *from* $\mathbb{C}[G/\!\!/H]$ *into invertible matrices as above. Then* ρ_H *is a representation of* $\mathbb{C}[G/\!\!/H]$.

Proof. We need to show that ρ_H is a homomorphism; for this it suffices to show that

$$\rho_H(b)\rho_H(c) = \rho_H(bc),$$

where b, c are double cosets in $G/\!\!/H$. Letting w_b and w_c be the H, H-invariant functions on G corresponding to the double cosets b and c, as above. By the definition of ρ_H, we have

$$\rho_H(b)\rho_H(c) = \rho_0(w_b)\rho_0(w_c) = \rho_0(w_b w_c).$$

We know that $bc = \sum_{d \in G/\!\!/H} p_{b,c,d} d$ for non-negative integers $p_{b,c,d}$ (the intersection numbers), and thus

$$w_b w_c = \sum_{d \in G/\!\!/H} p_{b,c,d} w_d.$$

Therefore,

$$\rho_0(w_b w_c) = \rho_0\left(\sum_{d \in G/\!\!/H} p_{b,c,d} w_d\right) = \rho_H(bc).$$

□

Also, we show that ρ_H is exactly all the representations of $\mathbb{C}[G/\!\!/H]$. We need a useful lemma:

Lemma 3.0.3. *Let*

$$\mathbb{C}[G]_H = \{\alpha \in \mathbb{C}[G] : \alpha \text{ is } H, H\text{-invariant}\}.$$

Then $\mathbb{C}[G]_H$ is a $\mathbb{C}$-algebra and $\mathbb{C}[G]_H \cong \mathbb{C}[G/\!\!/H]$ as $\mathbb{C}$-algebras. Moreover, the isomorphism $\varphi : \mathbb{C}[G/\!\!/H] \to \mathbb{C}[G]_H$ can be described as follows. For any $\alpha \in \mathbb{C}[G/\!\!/H]$, let $\varphi(\alpha) \in \mathbb{C}[G]_H$ such that for any $g \in G$, $\varphi(\alpha)_g = \alpha_b$ where $g \in b$, $b \in G/\!\!/H$. φ^{-1} is defined reversely.

Proof. It is not hard to check $\mathbb{C}[G]_H$ is a $\mathbb{C}$-algebra. We show that φ is a algebra isomorphism:

1. For any $\alpha, \beta \in \mathbb{C}[G/\!\!/H]$, we need to show $\varphi(\alpha\beta) = \varphi(\alpha)\varphi(\beta)$. Because φ is linear, wlog we can assume $\alpha, \beta \in G/\!\!/H$. Let $x = \varphi(\alpha\beta), y = \varphi(\alpha)\varphi(\beta)$.

Then

$$y_g = \sum_{hk=g} \varphi(\alpha)_h \varphi(\beta)_k$$

and that

$$x_g = p_{\alpha,\beta,b}$$

where $g \in b \in G/\!\!/H$. But $p_{\alpha,\beta,b} = \sum_{hk=g} \varphi(\alpha)_h \varphi(\beta)_k$ comes from the multiplication of double coset algebra.

2. φ is a bijection. If $\varphi(\alpha) = \varphi(\beta)$ for some $\alpha, \beta \in \mathbb{C}[G/\!\!/H]$, then for any $b \in G/\!\!/H$ we have

$$\alpha_b = \varphi(\alpha)_g = \varphi(\beta)_g = \beta_b$$

for any $g \in b$ so $\alpha = \beta$. Thus φ is injective. Furthermore, for any $\eta \in \mathbb{C}[G]_H$ we can construct $\alpha \in \mathbb{C}[G/\!\!/H]$ such that for any $b \in G/\!\!/H$

$$\alpha_b = \eta_g$$

for any $g \in b$. The fact that η is H, H-invariant guarantees this construction is well defined. Then $\varphi(\alpha) = \eta$. Therefore φ is surjective.

□

Lemma 3.0.4. *The set* $\{\rho_H : \rho \in \mathrm{Irr}(G)\}$ *is exactly the set of inequivalent irreducible representations of* $\mathbb{C}[G/\!\!/H]$.

Proof. By Lemma 3.0.2, we know ρ_H is a representation of $\mathbb{C}[G/\!\!/H]$. Under the H-adapted basis, for each ρ_H, $1 \leq i, j \leq \dim(\rho_H)$, we can define $\rho_{H,i,j} : \mathbb{C}[G/\!\!/H] \to \mathbb{C}$ to be

$$\rho_{H,i,j}(b) = [\rho_H(b)]_{ij}.$$

We claim that

$$\{\rho_{H,i,j} : \rho \in \mathrm{Irr}(G), 1 \leq i, j \leq \dim(\rho_H)\}$$

is a basis for all functions from $\mathbb{C}[G/\!\!/H]$ to $\mathbb{C}$ (which is isomorphic to $\mathbb{C}[G/\!\!/H]$). Then this immediately implies every ρ_H is irreducible because otherwise, $\rho_H = \tau \oplus \sigma$ but $\dim(\rho_H)^2 > \dim(\tau)^2 + \dim(\sigma)^2$ so we can replace all $\rho_{H,i,j}$ with $\tau_{i,j}, \sigma_{i,j}$. We just find a new basis with less elements, which is a contradiction.

Because $\mathbb{C}[G/\!\!/H]$ is a semi-simple algebra, and we have found a set of irreducible representations such that

$$\sum_{\rho \in \mathrm{Irr}(G)} \dim(\rho_H)^2 = |G/\!\!/H|.$$

Therefore, $\{\rho_H : \rho \in \mathrm{Irr}(G)\}$ is a complete set of irreducible representations of $\mathbb{C}[G/\!\!/H]$.

Finally we prove our claim. Again let $\varphi : \mathbb{C}[G/\!\!/H] \to \mathbb{C}[G]_H$ be the isomorphism described in Lemma 3.0.3. Given $\rho \in \mathrm{Irr}(G)$, we can define ρ_{ij} as

$$\rho_{ij}(g) = [\rho(g)]_{ij}$$

and extend linearly. We know from representation theory that

$$\{\rho_{ij} : \rho \in \mathrm{Irr}(G), 1 \leq i, j \leq \dim(\rho)\}$$

forms a basis for $\mathbb{C}[G]$. Under H-adapted basis, consider its subset

$$T = \{\rho_{ij} : \rho \in \mathrm{Irr}(G), 1 \leq i, j \leq \dim(\rho_H)\}.$$

For any $g \in G$, we have

$$\rho_{ij}\left(g \sum_{h \in H} h\right) = \left[\rho\left(g \sum_{h \in h} h\right)\right]_{ij} = \left[\rho(g)\rho\left(\sum_{h \in H} h\right)\right]_{ij} = |H|[\rho(g)]_{ij} = |H|\rho_{ij}(g)$$

where the third equality is because under H-adapted basis, we know from Lemma 3.0.1 that $\rho\left(\sum_{h \in H} h\right)$ acts by 0 on V_1. Therefore, for any $g \in G, h \in H$,

$$\rho_{ij}(gh) = \frac{1}{|H|}\rho_{ij}\left(gh \sum_{a \in H} a\right) = \frac{1}{|H|}\rho_{ij}\left(g \sum_{a \in H} a\right) = \rho_{ij}(g).$$

The computation for $\rho_{ij}(hg)$ is similar. Thus ρ_{ij} is H, H-invariant. Furthermore we claim that T forms a basis for $\mathbb{C}[G]_H$. They are linearly independent because T is a subset of $\{\rho_{ij} : \rho \in \mathrm{Irr}(G), 1 \leq i, j \leq \dim(\rho)\}$. We need to show that T

spans $\mathbb{C}[G]_H$. For any $\alpha \in \mathbb{C}[G]_H$, we have

$$\alpha = \sum_{\substack{\rho \in \mathrm{Irr}(G) \\ 1 \le i,j \le \dim(\rho)}} c_{\rho,i,j} \rho_{ij}$$

It suffices to show that $c_{\rho,i,j} = 0$ for all $\rho_{ij} \notin T$. Indeed,

$$c_{\rho,i,j} = \frac{1}{|G|} \sum_{g \in G} \alpha_g \overline{\rho_{ij}(g)}$$

where $\overline{\rho_{ij}}$ is the complex conjugate. One can check for any $g \in G$,

$$\begin{aligned}
\sum_{\substack{\rho \in \mathrm{Irr}(G) \\ 1 \le i,j \le \dim(\rho)}} c_{\rho,i,j} \rho_{ij}(g) &= \frac{1}{|G|} \sum_{\substack{\rho \in \mathrm{Irr}(G) \\ 1 \le i,j \le \dim(\rho)}} \left(\sum_{a \in G} \alpha_a \overline{\rho_{ij}(a)} \right) \rho_{ij}(g) \\
&= \frac{1}{|G|} \sum_{\substack{\rho \in \mathrm{Irr}(G) \\ 1 \le i,j \le \dim(\rho)}} \left(\sum_{a \in G} \alpha_a \overline{\rho_{ij}(a)} \rho_{ij}(g) \right) \\
&= \frac{1}{|G|} (MM^*)_{g,g} \\
&= 1
\end{aligned}$$

where M is the DFT-matrix, which we already know has the nice property that

$$MM^* = |G| I$$

where M^* is the conjugate transpose of M. Therefore, since α is H, H-invariant, Lemma 3.0.1 implies that $c_{\rho,i,j} = 0$ for all $\rho_{ij} \notin T$.

Finally, by Lemma 3.0.3, $\varphi^{-1}(T)$ is a basis for $\mathbb{C}[G/\!\!/H]$. For every $\rho_{ij} \in T, b \in /\!\!/H$,

$$\varphi^{-1}(\rho_{ij})(b) = \rho_{ij}(g) = \frac{1}{|H|} \rho_{ij} \left(\sum_{x \in b} x \right) = \frac{1}{|H|} \rho_{H,i,j}(b).$$

Therefore,

$$\{\rho_{H,i,j} : \rho \in \mathrm{Irr}(G), 1 \le i,j \le \dim(\rho_H)\}$$

is a basis for $\mathbb{C}[G/\!\!/H]$.

□

This concrete description allows us to "find" the DFT with respect to $\mathbb{C}[G/\!\!/H]$ within the DFT with respect to $\mathbb{C}[G]$ (by working in an H-adapted basis and restricting each ρ to ρ_0), and this is the basis of the next two algorithms, which both use the following theorem.

Theorem 3.0.5. *A DFT over $\mathbb{C}[G/\!\!/H]$ can be computed using the same number of operations as a DFT over $\mathbb{C}[G]$.*

Proof. Let φ be the isomorphism. Given $\alpha \in \mathbb{C}[G/\!\!/H]$, we can compute a G-DFT of $\varphi(\alpha)$ using an H-adapted basis:

$$s = \sum_g \varphi(\alpha)_g \bigoplus_{\rho \in \mathrm{Irr}(G)} \rho(g).$$

Noting that $\varphi(\alpha)$ is H, H-invariant and recalling the definition of ρ_0 above, for each ρ we can restrict to ρ_0 and then to ρ_H, and all other entries are guaranteed to be zero, by Lemma 3.0.1.

Moreover these ρ_H are all the irreps of $\mathbb{C}[G/\!\!/H]$, so we have computed

$$\begin{aligned}\sum_{b \in G/\!\!/H} \alpha_b \bigoplus_{\tau \in \mathrm{Irr}(G/\!\!/H)} \tau(b) &= \sum_{b \in G/\!\!/H} \alpha_b \bigoplus_{\rho \in \mathrm{Irr}(G)} \rho_H(b) \\ &= \sum_{b \in G/\!\!/H} \alpha_b \bigoplus_{\rho \in \mathrm{Irr}(G)} \rho_0(w_b) \\ &= \sum_{g \in G} \varphi(\alpha)_g \bigoplus_{\rho \in \mathrm{Irr}(G)} \rho_0(g).\end{aligned}$$

The last term in the equation above is exactly the nonzero part of s by our argument above.

□

Theorem 3.0.5 is a key theorem for our second and third algorithms. We are now able to use group-DFTs to compute Schurian scheme-DFTs.

Before we proceed to the subscheme reductions and recursive algorithms, we show that Theorem 3.0.5 immediately gives us:

Theorem 3.0.6 (Theorem 1.3.2)**.** *If $N \subset H$ are subgroups of G, then*

$$\mathrm{DC}(G/\!\!/H) \leq \mathrm{DC}(G/\!\!/N).$$

Proof. Again let $\varphi : \mathbb{C}[G/\!\!/H] \to \mathbb{C}[G]_H$ be the isomorphism. For any $\alpha \in G/\!\!/H$, $\varphi(\alpha)$ is H, H-invariant, and thus N, N-invariant. Therefore, we could compute a $G/\!\!/N$-DFT with respect to $\varphi(\alpha)$ which is just the $G/\!\!/H$-DFT we want. □

Chapter 4

Subscheme Reductions

4.1 Single Subscheme Reduction

We now introduce our single subscheme reduction, as an analog of the single subgroup reduction in [1]. The recursive algorithm has the exact same idea, but we require our intermediate subgroup K to be normal in G to ensure that all our vectors are H, H-invariant.

Theorem 4.1.1 (Single Subscheme Reduction)**.** *If $H < K \triangleleft G$, then*

$$\mathrm{DFT}(G/\!\!/H) \leq |G/K|\mathrm{DFT}(K/\!\!/H) + O(|G/K||G/\!\!/H|^{\omega/2+\epsilon})$$

for any $\epsilon > 0$.

Proof. By Theorem 3.0.5, to compute a DFT over $G/\!\!/H$ is equivalent to compute a DFT over H, H-invariant $\alpha \in \mathbb{C}[G]$. Let $g_1, \ldots, g_m$ be the coset representatives for G/K, then

$$\alpha = \sum_{g \in G} \alpha_g \cdot g = \sum_{i=1}^{m} \left(\sum_{k \in K} \alpha_{g_i k} k \right) g_i.$$

Because we parititon G by double K cosets (which is actually single coset when K is normal in G), our $\alpha_{g_i k}$ is H, H-invariant. Therefore, for each i, we can compute

$$s_i = \sum_{k \in K} \alpha_{g_i k} \bigoplus_{\sigma \in \mathrm{Irr}(K)} \sigma(k).$$

Lift s_i to $\overline{s_i}$ by repeating each $\sigma \in \mathrm{Irr}(K)$ as many times as it occurs in $\rho \in \mathrm{Irr}(G)$,

and notice that

$$\mathrm{DFT}(\alpha) = \sum_{g \in G} \alpha_g \bigoplus_{\rho \in \mathrm{Irr}(G)} \rho(g) = \sum_{i=1}^{m} \overline{s_i} \cdot \left(\bigoplus_{\rho \in \mathrm{Irr}(G)} \rho(g_i) \right).$$

Thus we can compute m many $K/\!\!/H$-DFTs and m diagonal matrix multiplications and get

$$\mathrm{DFT}(G/\!\!/H) = |G/K| \mathrm{DFT}(K/\!\!/H) + O(|G/K||G/\!\!/H|^{\omega/2+\epsilon})$$

for any $\epsilon > 0$.

□

The single subgroup reduction is costly when $|G/K|$ is large, i.e. there is no big normal subgroup K containing H. However, if K does not contain H, then we would expect KH to be a subgroup strictly larger than K and H.

Just like the single subgroup reduction, our single subscheme reduction is costly if K is small. That is, if G has no large normal subgroups, the single subscheme reduction fails to give a fast algorithm. Our "double subscheme reduction" described in the next section can be considered as an improvement on this problem.

4.2 Double Subscheme Reduction

In this section we provide the analog of the double subgroup reduction described in [1], which is intended to deal with the case when G does not have large normal subgroup.

Given a schurian scheme $G/\!\!/H$ and $H \subset K, P \subset G$ with $P \triangleleft G$, such that $KP = G$, we can compute a $G/\!\!/H$-DFT via DFTs with respect to $K/\!\!/H$ and $P/\!\!/H$.

Theorem 4.2.1. *Let $H \subset K, P \subset G$ be subgroups in G such that $P \triangleleft G$, and let $\alpha \in \mathbb{C}[G]$ be H, H-invariant and supported on KP. Fix a way of writing $g = kp$ for each $g \in KP$. Then, we can compute*

$$\sum_{g=kp \in KP} \alpha_g \bigoplus_{\sigma \in \mathrm{Irr}(K), \tau \in \mathrm{Irr}(P)} \sigma(k) \otimes \tau(p),$$

by performing $|P/\!\!/H|$ many $|K/\!\!/H|$-DFTs and $|K/\!\!/H|$ many $P/\!\!/H$-DFTs.

Proof. We can write

$$\alpha = \sum_{g \in G} \alpha_g \cdot g = \sum_{xP \in G/P} x \left(\sum_{p \in P} \alpha_{xp} \cdot p \right).$$

Similar as before, α_{xp} is H, H-invariant. We can pick the coset representatives x such that they are in K and are further also different double coset representatives for K/H. Indeed, if x_1P, x_2P are different G/P cosets, then $Hx_1H \subset x_1P, Hx_2H \subset x_2P$ must be different $G/\!\!/H$ double cosets and thus must also be different as $K/\!\!/H$ cosets. This also means $|K/\!\!/H| \geq |KP/P| = |K|/|K \cap P|$.

Extend all such x to the coset representatives y of $K/\!\!/H$, and c can be written further as

$$\alpha = \sum_{yH \in K/\!\!/H} y \left(\sum_{p \in P} \alpha_{yp} \cdot p \right).$$

Since α_{yp} is H, H-invariant, we can perform $|K/\!\!/H|$ many $|P/\!\!/H|$-DFTs to compute for each $yH \in K/H$:

$$s_y = \sum_{p \in P} \alpha_{yp} \bigoplus_{\tau \in \mathrm{Irr}(P)} \tau(p).$$

We use the notation $s_y[\tau, u, v]$ to refer to the (u, v) entry of component τ in the direct sum. Then we perform $|P/\!\!/H|$ many $K/\!\!/H$-DFTs to compute for each $\tau \in \mathrm{Irr}(P)$ and $u, v \in [\dim(\tau)]$,

$$t_{\tau,u,v} = \sum_{h \in H} s_h[\tau, u, v] \bigoplus_{\sigma \in \mathrm{Irr}(H)} \sigma(h).$$

Note that $t_{\tau,u,v}[\sigma, u_0, v_0]$ is the $((u_0, u), (v_0, v))$ entry of $\sum_{k,p} c_{xp} \sigma(k) \otimes \tau(p)$, so we are done. □

Lemma 4.2.2 ([1])**.** *If A is an $n_1 \times n_2$ matrix, B is an $n_2 \times n_3$ matrix, and C is an $n_3 \times n_4$ matrix, then the product ABC can be computed by multiplying $A \otimes C^T$ (which is an $n_1 n_4 \times n_2 n_3$ matrix) by B viewed as an $n_2 n_3$-vector.*

Corollary 4.2.3 ([1])**.** *If A and C are as above, and $n_1 = n_2, n_3 = n_4$, and we have several $n_2 \times n_3$ matrices $B_1, \ldots, B_l$, then we can compute AB_iC for all i from*

$A \otimes C^T$, at a cost of

$$O((n_2 n_3)^{\omega-1+\epsilon} \cdot \max\{n_2 n_3, l\})$$

operations for all $\epsilon > 0$.

Lemma 4.2.4 ([1])**.** *There is a linear map*

$$\varphi_{G,K,P} : \prod_{\sigma\in\mathrm{Irr}(K),\tau\in\mathrm{Irr}(P)} \mathbb{C}^{(\dim(\sigma)\dim(\tau))^2} \to \prod_{\rho\in\mathrm{Irr}(G)} \mathbb{C}^{\dim(\rho)^2}$$

that maps $\bigoplus_{\sigma\in\mathrm{Irr}(K),\tau\in\mathrm{Irr}(P)} \sigma(k)\otimes\tau(p)$ to $\bigoplus_{\rho\in\mathrm{Irr}(G)} \rho(kp)$ for all $k \in K, p \in P$. The map $\varphi_{G,K,P}$ can be computed using

$$\sum_{\sigma\in\mathrm{Irr}(K),\tau\in\mathrm{Irr}(P)} O\left((\dim(\sigma)\dim(\tau))^{\omega-1+\epsilon} \cdot \max\left\{\dim(\sigma)\dim(\tau), \sum_{\rho\in\mathrm{Irr}(G)} n_{\sigma,\rho} m_{\tau,\rho}\right\} + \sum_{\rho\in\mathrm{Irr}(G)} O(\dim(\rho)^{\omega+\epsilon})\right)$$

operations for all $\epsilon > 0$.

Lemma 4.2.5 ([1])**.** *For all finite groups G and subgroups K, P, the expression in the previous lemma is upper bounded by $O((|K||P|)^{\omega/2+\epsilon/2} + |G|^{\omega/2+\epsilon/2})$.*

Putting everything together, we get our "Double Subscheme Reduction".

Theorem 4.2.6 (Double Subscheme Reduction)**.** *Let $G/\!\!/H$ be a schurian scheme and $H \subset K, P \subset G$, $P \triangleleft G$ such that $KP = G$. Fix a way of writing $g = kp$ for each $g \in G$. Then*

$$\mathrm{DFT}(G/\!\!/H) \le |P/\!\!/H|\cdot\mathrm{DFT}(K/\!\!/H) + |K/\!\!/H|\cdot\mathrm{DFT}(P/\!\!/H) + O\left(|G/\!\!/H|^{\omega/2+o(1)} + (|P/\!\!/H||K/\!\!/H|)^{\omega/2+o(1)}\right).$$

Proof. For any $\alpha \in \mathbb{C}[G]$ that is H,H-invariant, we first use Theorem 4.2.1 to compute

$$\sum_{g\in G} \alpha_g \bigoplus_{\sigma\in\mathrm{Irr}(K),\tau\in\mathrm{Irr}(P)} \sigma(k)\otimes\tau(p)$$

by performing $|P/\!\!/H|$ many $K/\!\!/H$-DFTs and $|K/\!\!/H|$ many $P/\!\!/H$-DFTs. Now

apply the linear map $\phi_{G,K,P}$ to obtain

$$\sum_{g=kp\in G} \alpha_g \bigoplus_{\rho\in \mathrm{Irr}(G)} \rho(g)$$

with the cost indicated in the previous lemma. □

Chapter 5

Recursive Algorithms

5.1 First Recursive Algorithm

Recall that the main obstacle for our subscheme reduction algorithms is that we could not find appropriate "translation elements". However, Proposition 2.2.3 shows that if $H \triangleleft K$, then we overcome this obstacle as for any $b, c \in G/\!\!/H$, $b \cdot c$ has a single support.

We reduce to computing several DFTs over the group $N_G(H)/H$, and this strategy has small and "overhead" that is proportional to the number of $(N_G(H), H)$-double cosets (which is small, e.g., when the index of $N_G(H)$ in G is small). We first state a useful lemma:

Proposition 5.1.1. *The cost of multiplying two block diagonal matrices of block size $d_1, \ldots, d_n$ is*

$$O\left(\left(\sum_{i=1}^{n} d_i^2\right)^{\omega/2+o(1)}\right)$$

Proof. We know the cost to multiply two $d \times d$ matrices is $d^\omega + o(1)$. Since the product of two block diagonal matrices with block size $d_1, \ldots, d_n$ still has the same

block structure, we are just multiplying each block matrix, so the total cost is

$$\begin{aligned}\sum_{i=1}^{n} d_i^{\omega+o(1)} &\leq \left(\max_{1\leq i\leq n} d_i\right)^{\omega-2+o(1)} \sum_{i=1}^{n} d_i^2 \\ &\leq \left(\sum_{i=1}^{n} d_i^2\right)^{(\omega-2)/2+o(1)} \left(\sum_{i=1}^{n} d_i^2\right) \\ &= \left(\sum_{i=1}^{n} d_i^2\right)^{\omega/2+o(1)}\end{aligned} \tag{5.1}$$

□

In the subscheme reductions, we assume that $K \triangleleft G$ and we were able to overcome the obstacle. Here we show that we could again overcome the obstacle when $H \triangleleft K$:

Lemma 5.1.2. *Let H be a subgroup of group G, define $K = N_G(H)$, and let $t_1, t_2, \ldots t_{|K\backslash G/H|}$ be representative (H,H)-double cosets, one inside each (K,H)-double coset. Given $\alpha \in \mathbb{C}[G/\!\!/H]$, we can compute vectors $\alpha^{(i)} \in \mathbb{C}[K/\!\!/H]$ for which*

$$\alpha = \sum_{u\in(G/\!\!/H)} \alpha_u u = \sum_i \left(\sum_{w\in K/\!\!/H} \alpha_w^{(i)} w\right) \cdot t_i \tag{5.2}$$

using $|K\backslash G/H||K/\!\!/H|$ operations overall.

Proof. By Proposition 2.2.3, we know $w \cdot t_i$ only has a single support, which is in the same K,H-double cosets as t_i, in the double coset. We assign $\alpha_w^{(i)}$ using the following algorithm: for any $i = 1, \ldots, |K\backslash G/H|$,

1. For all $w \in K/\!\!/H$, we "visit" the support u of $w \cdot t_i$.
2. If u has not been visited, assign $\alpha_w^{(i)} = \alpha_u / p^u_{w,t_i}$.
3. Otherwise, assign $\alpha_w^{(i)} = 0$.

We claim that this assignment will make sure equation 5.2 holds.

First observe that for any $w, v \in K/\!\!/H$, $w\cdot t_i$ and $v\cdot t_j$ will have different support for if $i \neq j$. Indeed, the support of $w \cdot t_i$ and $v \cdot t_j$ are in different K,H double cosets which form a partition of G. Thus we can look at each i separately. Also, we show for each $u \in G/\!\!/H$, the coefficient of u in right hand side of equation 5.2

is α_u. Assume that u is in the same K, H-double coset as t_j. We claim there exists $w \in K/\!\!/H$ such that $w \cdot t_j = p^u_{w,t_j} u$ and since $\alpha^{(j)}_w w \cdot t_j = (\alpha_u / p^u_{w,t_j}) p^u_{w,t_j} u = \alpha_u u$, the coefficients will match, and step 2 ensures that we do not have repetitive assignments.

Let $u = Hu_0H, w = Hw_0H, t_j = Ht_0H$. Then the support of $w \cdot t_j$ is Hw_0t_0H. Since u, t_j are in the same K, H-double coset, so are u_0, t_0, which means $u_0 = kt_0h$ for some $k \in K, h \in H$. Let $w_0 = k$ and we have $Hw_0t_0H = Hkt_0H = Hkt_0hH = Hu_0H$. □

We give our first recursive algorithm as follows.

Theorem 5.1.3 (First Recursive Algorithm)**.** *Given a Schurian scheme $G/\!\!/H$,*

$$\mathrm{DC}(G/\!\!/H) \leq |N_G(H)\backslash G/H||N_G(H)/H|^{\omega/2+o(1)} + O(|N_G(H)\backslash G/H||G/\!\!/H|^{\omega/2+o(1)}).$$

Proof. Let $K = N_G(H)$. Since $H \subseteq K$, each (K, H)-double coset in G is the union of (H, H)-double cosets. Let $t_1, t_2, \ldots t_{|K\backslash G/H|}$ be representative (H, H)-double cosets, one inside each (K, H)-double coset. These may be chosen arbitrarily.

Given $\alpha \in \mathbb{C}[G/\!\!/H]$, by Lemma 5.1.2 we can write

$$\alpha = \sum_{u \in (G/\!\!/H)} \alpha_u u = \sum_i \left(\sum_{w \in K/\!\!/H} \alpha^{(i)}_w w \right) \cdot t_i \tag{5.3}$$

for some $\alpha^{(i)} \in \mathbb{C}[K/\!\!/H] \cong \mathbb{C}[K/H]$, and these $\alpha^{(i)}$ vectors can be computed from α in linear time in the output, i.e., using only $|K\backslash G/H||K/\!\!/H|$ operations.

For each i we compute an $(K/\!\!/H)$-DFT to obtain

$$s^{(i)} = \sum_{w \in K/\!\!/H} \alpha^{(i)}_w \bigoplus_{\sigma \in \mathrm{Irr}(\mathbb{C}[K/\!\!/H])} \sigma(w).$$

Let $\overline{s^{(i)}}$ be the lift of $s^{(i)}$ in which we repeat each $\sigma \in \mathrm{Irr}(\mathbb{C}[K/\!\!/H])$ as many times as it occurs in the irreducible representations of $\mathbb{C}[G/\!\!/H]$. Finally, compute

$$\sum_i \overline{s^{(i)}} \cdot \left(\bigoplus_{\rho \in \mathrm{Irr}(\mathbb{C}[G/\!\!/H])} \rho(t_i) \right),$$

by multiplying each $\overline{s^{(i)}}$ on the right and summing the results.

We claim this gives the desired result. We have

$$\begin{aligned}\sum_i \overline{s^{(i)}} \cdot \left(\bigoplus_{\rho \in \mathrm{Irr}(\mathbb{C}[G/\!\!/H])} \rho(t_i) \right) &= \sum_i \left(\sum_{w \in K/\!\!/H} \alpha_w^{(i)} \bigoplus_{\rho \in \mathrm{Irr}(\mathbb{C}[G/\!\!/H])} \rho(w) \right) \cdot \left(\bigoplus_{\rho \in \mathrm{Irr}(\mathbb{C}[G/\!\!/H])} \rho(t_i) \right) \\ &= \sum_i \sum_{w \in K/\!\!/H} \alpha_w^{(i)} \bigoplus_{\rho \in \mathrm{Irr}(\mathbb{C}[G/\!\!/H])} \rho(wt_i) \\ &= \sum_{u \in G/\!\!/H} \alpha_u \bigoplus_{\rho \in \mathrm{Irr}(\mathbb{C}[G/\!\!/H])} \rho(u).\end{aligned}$$

where the first equality applies the definition of $\overline{s^{(i)}}$, the second equality uses that each ρ is a homomorphism, and the final equality follows from Equation (5.3).

Because

$$\sum_{\rho \in \mathrm{Irr}(\mathbb{C}[G/\!\!/H])} (\dim \rho)^2 = |G/\!\!/H|$$

here we are multiplying block diagonal square matrices with $|G/\!\!/H|$ nonzero entries, $|K\backslash G/H|$ many times, which takes $|K\backslash G/H| \cdot O(|G/\!\!/H|^{\frac{\omega}{2}+\epsilon})$ time overall, by Proposition 5.1.1. The Theorem follows. □

This algorithm works well when the number of $N_G(H), H$ double cosets is small, which happens in particular, when the normalizer $N_G(H)$ has small index in G, because

$$|N_G(H)\backslash G/H| \leq \frac{|G|}{|N_G(H)|}.$$

Notice that it is important here for H to be normal in K, otherwise we will not have single support in all multiplications. The general statement could be false when H is not normal in K because there is no guarantee that $|K\backslash G/H||K/\!\!/H| \geq |G/\!\!/H|$, which is a necessary condition to find the coefficients for equation 5.2 to hold.

5.2 Second Recursive Algorithm

For any $\alpha \in \mathbb{C}[G/\!\!/H]$, Theorem 3.0.5 shows that it suffices to compute

$$\sum_{g \in G} \varphi(\alpha)_g \bigoplus_{\rho \in \mathrm{Irr}(G)} \rho_0(g),$$

where $\varphi : \mathbb{C}[G/\!\!/H] \to \mathbb{C}[G]_H$ is the isomorphism. For any intermediate subgroup $H \subset P \subset G$, let $g_1, \ldots, g_m$ be the coset representatives of G/P. Then we can natually express $\varphi(\alpha)$ by

$$\varphi(\alpha) = \sum_{i=1}^{m} \beta_i \cdot g_i$$

where $\beta_i \in \mathbb{C}[P] \subset \mathbb{C}[G]$ and

$$(\beta_i)_p = \varphi(\alpha)_{g_i p}$$

for all $p \in P$. In order to do a recursive algorithm, we need β_i to be H, H-invariant as before. The following lemma shows that this occurs when $g_i \in N_G(H)$ for all i.

Lemma 5.2.1. *Suppose P is an intermediate subgroup of H, G. Given H, H-invariant $\varphi(\alpha) \in \mathbb{C}[G]$, for all $1 \leq i \leq m$, let $\beta_i \in \mathbb{C}[P]$ be defined as*

$$(\beta_i)_p = \varphi(\alpha)_{g_i p}$$

such that $\varphi(\alpha) = \sum_{i=1}^{m} \beta_i \cdot g_i$. Then if $N_G(H)P = G$, β_i is H, H-invariant for all i.

Proof. Because $N_G(H)P = G$, we can pick coset representatives $g_1, \ldots, g_m$ for G/P such that $g_i \in N_G(H)$ for all i. Then for any $a, b \in H$, $1 \leq i \leq m$, we have

$$(\beta_i)_{apb} = \varphi(\alpha)_{g_i apb} = \varphi(\alpha)_{(g_i a x_i^{-1}) g_i pb} = \varphi(\alpha)_{g_i p} = (\beta_i)_p$$

where the middle equality holds as $g_i a g_i^{-1} \in H$. □

We give our second algorithm for computing a $G/\!\!/H$-DFT based on our analysis above:

Theorem 5.2.2 (Second Recursive Algorithm)**.** *Given a Schurian scheme $G/\!\!/H$, if there exists a subgroup P such that $H \subset P$ and $PN_G(H) = G$, then*

$$\mathrm{DC}(G/\!\!/H) \leq |G/P|\mathrm{DC}(P/\!\!/H) + O(|G/P||G/\!\!/H|^{\omega/2+\epsilon})$$

for any $\epsilon > 0$.

Proof. Given $\alpha \in G/\!\!/H$, again it suffices to compute

$$\sum_{g \in G} \varphi(\alpha)_g \bigoplus_{\rho \in \mathrm{Irr}(G)} \rho(g).$$

By the previous lemma, we know we can write

$$\varphi(\alpha) = \sum_{i=1}^{m} \beta_i \cdot g_i$$

where $\beta_i \in \mathbb{C}[P]$ are H, H-invariant. Compute $m = |G/P|$ many $P/\!\!/H$-DFTs and lift them just as in Theorem 5.1.3. □

Chapter 6

Conclusions

In this paper, we give several recursive algorithms to compute the $G/\!\!/H$-DFTs. Our main obstacle is that for any H, H-invariant $\alpha \in \mathbb{C}[G]$, we want to find the appropriate translations $t_1, \ldots, t_m$ such that

$$\alpha = \sum_{i=1}^{m} \beta_i \cdot t_i,$$

where β_i are also H, H-invariant. We overcome this obstacle in several different scenrios, and we aim for solving it for arbitary $G/\!\!/H$.

References

(1) Hsu, C. C.-Y.; Umans, C. A new algorithm for fast generalized DFTs, 2018.

(2) Beth, T., *Verfahren der schnellen Fourier-Transformation*; Teubner: 1984.

(3) Clausen, M. *Theoretical Computer Science* **1989**, *67*, 55–63.

(4) Baum, U. *computational complexity* **1991**, *1*, 235–256.

(5) Maslen, D. K. *Math. Comput.* **1998**, *67*, 1121–1147.

(6) Umans, C. Fast generalized DFTs for all finite groups, 2019.

(7) Maslen, D.; Rockmore, D. N.; Wolff, S. *arXiv preprint arXiv:1609.02634* **2016**, To appear in Journal of Fourier Analysis and Applications.

(8) Bose, R. C.; Shimamoto, T. *Journal of the American Statistical Association* **1952**, *47*, 151–184.

(9) Alman, J.; Williams, V. V. A Refined Laser Method and Faster Matrix Multiplication, 2020.

(10) Cooley, J. W.; Tukey, J. W. *Mathematics of Computation* **1965**, *19*, 297–301.

(11) Bürgisser, P.; Clausen, M.; Shokrollahi, M. A., *Algebraic Complexity Theory*; Grundlehren der mathematischen Wissenschaften, Vol. 315; Springer-Verlag: 1997.

(12) Maslen, D.; Rockmore, D. N.; Wolff, S. *Journal of Fourier Analysis and Applications* **2016**, 1–59.

(13) Lev, A. *Journal of Algebra* **1992**, *152*, 434–438.

(14) Schmidt, J. *Computational Group Theory and the Theory of Groups, II* **2010**, *511*, 185.

(15) Carter, R. W., *Simple groups of Lie type*; John Wiley & Sons: 1989; Vol. 22.

(16) Clausen, M.; Hühne, P. In *Proceedings of the 2017 ACM on International Symposium on Symbolic and Algebraic Computation*, ACM: Kaiserslautern, Germany, 2017, pp 101–108.

(17) Clausen, M.; Baum, U., *Fast Fourier transforms*; Wissenschaftsverlag: 1993.

(18) Lafferty, J. D.; Rockmore, D. *Experiment. Math.* **1992**, *1*, 115–139.

(19) Le Gall, F. In *Proceedings of the 39th international symposium on symbolic and algebraic computation*, 2014, pp 296–303.

(20) Maslen, D. K.; Rockmore, D. N. *Journal of Fourier Analysis and Applications* **2000**, *6*, 349–388.

(21) Rockmore, D. N. *Applied and Computational Harmonic Analysis* **1995**, *2*, 279–292.

(22) Maslen, D. K.; Rockmore, D. N. In *Groups and Computation II*, 1997; Vol. 28, pp 183–287.

(23) Maslen, D.; Rockmore, D. *Journal of the American Mathematical Society* **1997**, *10*, 169–214.

(24) Rockmore, D. N. In *Signals, Systems and Computers, 2002. Conference Record of the Thirty-Sixth Asilomar Conference on*, 2002; Vol. 1, pp 773–777.

(25) Wikipedia List of finite simple groups — Wikipedia, The Free Encyclopedia, [Online; accessed 30-June-2017], 2017.

(26) Rockmore, D. In *Proceedings of the 1995 DIMACS Workshop on Groups and Computation*, 1997, pp 329–369.

(27) Horn, R. A.; Johnson, C. R., *Topics in Matrix Analysis*; Cambridge University Press: 1991.

(28) Hsu, C. C.; Umans, C. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2018, New Orleans, LA, USA, January 7-10, 2018*, ed. by Czumaj, A., SIAM: 2018, pp 1047–1059.

(29) Czumaj, A., Ed., Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2018, New Orleans, LA, USA, January 7-10, 2018, SIAM: 2018.