RECENT PROGRESS AND APPLICATIONS IN GROUP FFTS

Daniel N. Rockmore
Dartmouth College
Hanover, NH 03755
rockmore@cs.dartmouth,edu

Abstract

The Cooley-Tukey FFT can be interpreted as an algorithm for the efficient computation of the Fourier transform for finite cyclic groups, a compact group (the circle), or the non-compact group of the real line. These are all commutative instances of a "Group FFT." We give a brief survey of some recent progress made in the direction of noncommutative generalizations and their applications.

Keywords: Fast Fourier transform, discrete Fourier transform, sampling, Gel'fand-Tsetlin bases.

1. Introduction

The *Fast Fourier Transform* or *FFT* is an efficient algorithm to compute the *Discrete Fourier Transform* (*DFT*). This is a linear transformation, specifically realized in terms of the the $(n \times n)$ *DFT matrix*:

$$\widehat{f} = \left(e^{2\pi ijk/n}\right)_{jk} f, \tag{1}$$

which takes a vector of samples realized as a function $f \in \mathbb{C}^n$, and returns a collection of Fourier coefficients $\widehat{f} \in \mathbb{C}^n$.

The DFT plays a crucial role in a wide range of applied activities, principally in the analysis of *time series* data. These natural quantifications of temporal phenomena presumably owe their origins to the observations of the first priestly mathematicians who made it their work to chart the course of heavenly bodies in the construction of the first calendars. Other early examples would include the analysis of the time course of temperatures, rainfall, and various meteorological data, possibly in the service of the study of the time series of agriculture (crop yields, etc.). Closer to our hearts and heads are the time series of health that are our EKGs and EEGs, which are perhaps, in turn, influenced by the seemingly random walk that are the time series which reflect the health of the financial markets.

One approach to time series analysis is to view these phenomena as well-explained as the superposition of basic periodic phenomena: weather as a combination of diurnal and annual effects, blood pressure or hormone levels tracking some invisible and evolved pacemaker. Herein the DFT is that transformation of the data that teases out the periodicities, taking the discrete or discretized signal and transforming it to a representation in terms of weighted frequencies.

Direct calculation of a DFT of length n is effected by the multiplication of the $n \times n$ DFT matrix with a vector of length n and so requires n^2 operations. When n is large even this quadratic calculation is too large. The great success of the FFT is the reduction in complexity to $O(n \log_2 n)$ operations (with implicit small constant).

1.1 Brief History of the Classical FFT

The original FFT was indeed due to Gauss, and has the astronomical (although not religious) origins indicated above. Its story provides another proof that necessity is indeed often the mother of invention. Gauss was confronted with the computationally daunting problem of interpolating—by hand!—the periodic orbit of the asteroid Ceres, which had suddenly gone missing. Gauss determined a means of building the interpolation on n points from two interpolations of $n/2$ points, and in so doing discovered the basic step in what is now the standard divide-and-conquer efficient algorithm. His discovery languished for centuries (cloaked in Latin and hidden away in little known writings) while a few centuries later it was rediscovered by (among others) Danielson and Lanczos in the service of crystallography but, surely most famously, by Cooley and Tukey [12] in the mid 1960s, this time not in the service of the discovery of missing heavenly bodies, but instead for the detection of hidden nuclear tests in the Soviet Union, as well as stealthy Soviet nuclear submarines. For full histories see [22].

The technical motivations for Cooley and Tukey's rediscovery were

(1) The efficient computation of the power spectrum of time series (especially sampled time series of very long length, so equivalently, the calculation of high frequency contributions) and

(2) Efficient filtering (smoothing).

Item (2) is equivalent to the efficient computation of (cyclic) *convolution.* For vectors f and g of length n this is defined as

$$f \star g(x) = \sum_{y=0}^{n-1} f(x-y)g(y)$$

where the arguments are interpreted as integers mod n. (*Linear convolution* is that which corresponds to the generation of a vector of length $2n$ from f by interpreting the samples $f(i), g(k)$ as coefficients in a polynomial of degree $n-1$, and then asking for the coefficient of x^k in the product. Note that this could be obtained by computing cyclic convolution of f and g *zero-padded* to vectors of length $2n$.)

Note that direct computation of the convolution requires n^2 operations. The identity

$$\widehat{f \star g}(k) = \widehat{f}(k)\widehat{g}(k) \tag{2}$$

shows how application of the DFT permits the filtering of f to be performed directly in the frequency domain via the assignation of a particular frequency profile for g. When $\widehat{g}$ takes only values zero and one, it has the form of a *bandpass filter*, and if the ones are restricted to a subsequence of indices, this nonzero interval is the *passband. Lowpass* filters restrict the passband to an initial segment and a terminal segment for *highpass* filters.

The FFT enables fast convolution via the algorithm

$$f, g \longrightarrow \widehat{f}, \widehat{g} \longrightarrow \widehat{f} \bigodot \widehat{g} \longrightarrow f \star g$$

where $\bigodot$ is meant to indicate pointwise multiplication of the two vectors it separates. Note that the last step is accomplished via an **inverse** FFT, so that in total, the algorithm requires three FFTs and a single n point pointwise multiplication for a total of $O(n \log n)$ operations.

1.2 Group Theoretic Interpretations

"The FFT" is actually a family of algorithms, all designed to compute efficiently the DFT (1). This linear transformation can be cast as a particular instance of any of a variety of mathematical operations, but the focus in this chapter is a group theoretic, indeed, representation theoretic point of view. Within this, there are at least three different interpretations, corresponding to either the case of finite, compact, or non-compact groups. We summarize these below, for each of them presents its own challenges for generalization.

(1) **Finite Groups**— In this setting we view f as a function on the cyclic group of order n, C_n (isomorphic to $\mathbb{Z}/n\mathbb{Z}$). The FFT is the

efficient change of basis algorithm that takes a function written in terms of the basis of delta functions and re-expresses it in terms of the basis of sampled exponentials,

$$\left\{\sum_{x\in C_n} f(x)\delta_x\right\} \longrightarrow \left\{\sum \widehat{f}(k)e_k\right\}$$

where $e_k(j) = e^{2\pi ijk/n}$.

(2) **Compact Groups**— In this case, the vector f is viewed as samples of a function on the circle S^1 (i.e., samples of a periodic function). Any such function has a Fourier expansion, $f(t) = \sum_{\ell\in\mathbb{Z}} \widehat{f}(\ell)e^{-2\pi i\ell t}$ where the Fourier coefficients are computed by an integral

$$\widehat{f}(\ell) = \int_0^1 f(e^{2\pi it})e^{2\pi i\ell t}dt.$$

In general, the FFT can be used to compute efficiently an approximation to these Fourier coefficients, but in the interesting *bandlimited* case, in which the function's Fourier expansion is finite (i.e., there exists $B \geq 0$ such that $\widehat{f}(\ell) = 0$ for $\ell \geq B$), there is an exact *quadrature* or *sampling* rule that provides an exact formula for the (potentially) nonzero Fourier coefficients in terms of a DFT of length $2B + 1$.

(3) **Non-compact Groups** – In this last case, we view our discrete set of samples as arising from a complex-valued function f defined on the real line $\mathbb{R}$. Once again, the Fourier transform is a linear transformation from time (or space) to frequency, this time given as the integral operator (for each x),

$$\widehat{f}(x) = \int_{\mathbb{R}} f(y)e^{-2\pi iyx}dy.$$

As in the compact case the DFT might be used to approximate this integral, and once again there is a bandlimited theory (i.e., the case in which the Fourier transform only has finite support). In this case, the function f is determined by its equispaced samples along the entire real line (i.e., so-called "Shannon sampling"). Consequently, the FFT provides a means for an efficient and quantifiable approximation to the computation of f's frequency content.

In summary, the FFT makes possible the efficient analysis of: (1) discrete periodic data viewed as a function on the discrete circle, that is, a cyclic group of finite order (C_n); (2) continuous periodic data, viewed as a function on the circle; and (3) continuous data, viewed as a function on the line.

1.3 Noncommutative generalizations

The groups C_n, S^1, and $\mathbb{R}$ are all *commutative* groups, i.e., the law we use to combine them obeys a commutative rule: $x + y = y + x$. Each of the above commutative group theoretic interpretations has, over the past generation, found generalization to the *noncommutative* setting, and the purpose of this chapter is to provide a window into this work.

Abstractly, a group is simply a set closed under some associative multiplication rule such that there is an identity element, and to each element there is an inverse. Classically, these arose as the symmetries of roots of polynomials, i.e., those arithmetic transformations that leave invariant a given polynomial, and from this they grew to encompass the notion of symmetry throughout mathematics and physics. They are in general noncommutative, i.e., usually $xy \neq yx$ (think matrices!). As indicated above, they come in at least three general flavors - the three in which we are interested: *Finite*, *Compact* and *Non-compact*.

(1) **Finite groups**—The most familiar commutative examples are the aforementioned cyclic groups, C_n, while of the noncommutative examples, the symmetric groups, S_n, the group of permutations of n elements, commonly realized as the group of all card shuffles of a deck of size n is perhaps the most familiar.

(2) **Compact groups**—Standard examples come from the matrix groups whose entries are bounded in size. The orthogonal groups $O(n)$ and the special orthogonal groups $SO(n)$ (also called *rotation groups*—symmetries of the $n-1$-dimensional sphere), and their complex analogues—unitary groups $U(n)$ and special unitary groups $SU(n)$, are examples. With their length-preserving properties they are effectively the symmetries of space.

(3) **Non-compact groups**—The invertible complex or real matrices, $GL_n(\mathbb{C})$ or $GL_n(\mathbb{R})$ are well-known examples, and within these, the *Euclidean motion groups* are particularly useful. These are (for any n) the matrices of the form $\begin{pmatrix} a & b \\ 0 & 1 \end{pmatrix}$ where $a \in SO(n)$ and $b \in \mathbb{R}^n$. These occur naturally as symmetries of n-dimensional affine space.

1.3.1 Noncommutative DFTs.

Given a complex-valued function defined on a group G, its Fourier decomposition (analysis) is meant to be a rewriting in terms of a basis of functions that are nicely adapted to translation via group elements. It is in this sense a symmetry-guided decomposition.

In the commutative case we have eigenfunctions of translation: If $e(x) = e^{2\pi ix}$, then

$$T_y e(x) = e(x - y) = e^{-2\pi iy} e(x)$$

where T_y indicates the translation operator.

In the noncommutative case there are no simultaneous eigenfunctions for all translation operators. This is both the source of frustration as well as the spur to art for the theory and application of *the representation theory of noncommutative groups.* This forces us to look for the next best thing which is closure of some linear space under the translation action. That is, a basis of functions $e_k(x)$ on the group that have the property

$$T_y e_k(x) = e_k(xy^{-1}) = \sum_{\ell} T_y(k, \ell) e_\ell(x).$$

In this way we see that the eigenfunctions are naturally replaced by functions that act like, and bear the name of *matrix elements* $T_y(k, \ell)$, and it is essentially these functions which replace the sampled exponentials that create frequency space in the commutative case. When grouped together they give *matrix representations* of the group and comprise what is called *the dual* of the group (denoted $\widehat{G}$). Their study is the subject of *group representation theory.*

So in general we have, for any function f defined on a finite group G, the notion of a Fourier expansion

$$f(x) = \sum_{\rho \in \widehat{G}} c_\rho \sum_{k,\ell} \widehat{f}(k, \ell) T_x(k, \ell) \tag{3}$$

where c_ρ is some constant depending on an irreducible representation ρ, the $\widehat{f}(k, \ell)$ are the Fourier coefficients, and the matrix elements (which depend on x) now span the analogue of frequency space. The Fourier transform computes these Fourier coefficients, and it amounts to computing the discrete inner product of the function with the new basis of irreducible matrix elements.

Should G be compact, the sum is infinite (in analogy with the sum over the integers in the case of the circle) while if G is non-compact, this sum is in general some sort of integral (cf. [9] for pointers to basic representation theory references).

This new basis effects convolution in a manner akin to the commutative case:

$$\widehat{f\star g}(k,\ell)=\sum_{m}\widehat{f}(k,m)\widehat{g}(m,\ell) \tag{4}$$

where $f\star g(x)=\sum_{y\in G}f(xy^{-1})g(y)$.

1.4 Organization of this chapter

The majority of what follows focuses on the case of finite groups, for most of the progress has been in this area. This is the content of the next section. Included are generalizations of both the Cooley-Tukey FFT (decimation in time) in the guise of *separation of variables* group FFTs, as well as the Gentleman-Sande FFT (decimation in frequency). We also touch upon the large body of recent work devoted to the development of *quantum (finite group) FFTs*. Section three is devoted to compact group FFTs, almost exclusively compact Lie groups, while Section four discusses recent work in the difficult, but tremendously useful, noncompact case. Indeed, this raises the issue of both the utility and applicability of these algorithms, for while the abstract development of algorithms has epistemological value, it is even of greater interest when motivated by and subsequently applied to real problems. With this in mind, each section contains some indication and discussion of applications, and indeed, we hope that this chapter might inspire many new uses.

2. Finite group FFTs

As mentioned, when G is finite and commutative, the number of operations required is bounded above by $O(|G|\log|G|)$. For arbitrary finite groups G, upper bounds of $O(|G|\log|G|)$ remain the holy grail in group FFT research. Implicit in the big-O notation is the idea that a family of groups is under consideration, with the size of the individual groups going to infinity. In 1978, A. Willsky provided the first noncommutative example by showing that certain metabelian groups had an $O(|G|\log|G|)$ Fourier transform algorithm [55].

Two of the most important algorithms in the commutative case are the Cooley-Tukey FFT and the Gentleman-Sande FFT, the former often described as decimation in time, while the latter as decimation in frequency, their similarity reflected in a natural isomorphism between the group and its dual that exists in the finite commutative case. In this section we describe in some detail the separation of variables approach [35] which generalizes the former, and an isotypic projection algorithm [32] which generalizes the latter.

2.1 Applications

While the applications of Fourier analysis on commutative groups is now legion (see the Introduction in [8] for a truly mind-boggling list!), for finite noncommutative groups the list is still short, but constantly growing.

To date, Fourier analysis on the symmetric group S_n seems to have found the most applicability. It has has been proposed and used to analyze ranked data. In this setting respondents are asked to rank a collection of n objects. As a result, each participant in effect chooses a permutation of the initially ordered list of objects. The counts of respondents choosing particular rankings then gives rise to a function on S_n for which Fourier analysis provides a natural generalization of the usual spectral analysis applied to a time series. Diaconis has used this to study voting data (cf. [14] for a discussion of this example, as well as others). More recently, Lafferty has applied this to the development of conditional probability models to analyze some partially ranked data [27].

In communications, Fourier analysis on finite matrix groups, $SL_2(p)$, the group of two-by-two matrices with determinant one with entries in a finite field, has made possible new developments in the area of low density parity check (LDPC) codes [28], and also proved instrumental in the construction of expander graphs that provide models for networks with high connectivity but relatively small numbers of links.

2.2 Cooley-Tukey revisited

The separation of variables approach generalizes the *decimation in time* FFT, which is essentially the guts of the Cooley-Tukey FFT.

Assuming that $n = pq$ (not necessarily prime), then decimation in time refers to the factorization of our time index ℓ as

$$\ell = \ell_1 q + \ell_2 \quad (0 \leq \ell_1 < p, 0 \leq \ell_2 < q) \tag{5}$$

which is coupled with a corresponding factorization of the frequency index k as

$$k = k_1 + k_2 p \quad (0 \leq k_1 < p, 0 \leq k_2 < q) \tag{6}$$

so that

$$\widehat{f}(k) = \sum_{\ell_2} e^{2\pi i \ell_2 k} \sum_{\ell_1} e^{2\pi i \ell_1 k_1/p} f(\ell_1, \ell_2) \tag{7}$$

where $f(\ell_1, \ell_2) = f(\ell_1 q + \ell_2)$.

Notice that this rewrites a “one-dimensional” computation as a “two-dimensional” computation. The FFT organizes the calculation into two stages:

- Stage 1: For all k_1, ℓ_2 compute

$$\tilde{f}(k_1, \ell_2) = \sum_{\ell_1} e^{2\pi i \ell_1 k_1/p} f(\ell_1, \ell_2). \tag{8}$$

This requires at most pq^2 operations.

- Stage 2: For all k_1, k_2 compute

$$\widehat{f}(k_1, k_2) = \sum_{\ell_2} e^{2\pi i \ell_2 (k_1 + k_2 p)} \tilde{f}(k_1, \ell_2). \tag{9}$$

This requires at most $p^2 q$ operations.

In toto, this gives an algorithm which requires $pq(p+q)$ operations, rather than $(pq)^2$, providing savings as long as factorization is possible.

This approach generalizes nicely. Decimation in time is naturally replaced by *group factorization*, (first generally observed by Beth [5]), but the concomitant factorization of the dual (frequency) requires a little work. For this the machinery of *Bratteli diagrams* has proved to be of immense utility. For illustration we'll revisit Cooley-Tukey in this setting.

Bratteli diagram for $\mathbb{Z}/6\mathbb{Z} > 2\mathbb{Z}/6\mathbb{Z} > 1$.

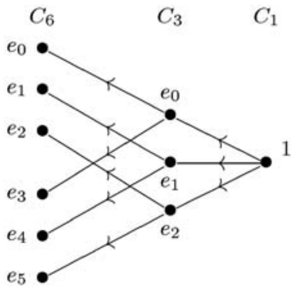

Figure 1. The Bratteli diagram for $C_6 > C_3 > C_1$.

In brief, the diagram above reflects a chain of subgroups $C_6 > C_3 > 1$, the nodes correspond to representations (frequencies) and one node is connected to another if when restricted to that subgroup it gives that corresponding representation. For example, evaluation of e_5 on the multiples of 2 (which comprise the copy of C_3 in C_6) is equivalent to simply evaluating e_2 on C_3.

A full path in the Bratteli diagram is now a frequency, and the factorization (6) gives a labeling of the legs that make up the path. Stages

1 and 2 above can now be reinterpreted diagrammatically. Stage 1 requires the computation over all *initial paths* k_1 and *subgroup elements* ℓ_1, while Stage 2 becomes a computation over all coset representatives ℓ_2 and *full paths* (k_1, k_2).

Separation of variables

As described in [35] in the noncommutative case separation of variables takes on a general form that requires more elaborate Bratteli diagrams. Once again, the initial data is a chain of subgroups $G_n > \ldots, > G_1 > G_0 = \{1\}$, but the nodes at level i now correspond to *matrix representations* of G_i. A node η at level i is connected to ρ at level $i+1$ by a number of arrows equal to the multiplicity of η in $\rho|G_i$.

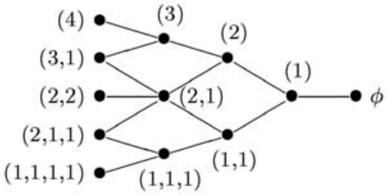

Figure 2. The Bratteli diagram for $S_4 > S_3 > S_2 > 1$.

For example, matrix representations for the group S_n correspond to partitions of n, hence the labeling in the Bratteli diagram for the subgroup chain $S_4 > S_3 > S_2 > 1$ in Figure 2. Notice that the partition $(2,2)$ in Figure 2 reveals that full paths are no longer uniquely described by their endpoints.

The arrows from η to ρ correspond to mutually orthogonal G_i-equivariant maps of a given irreducible vector space. In this way each full path in the diagram corresponds to a basis vector of an irreducible representation of G. Bases indexed in this fashion are called *Gel'fand-Tsetlin bases.*

Formally, this creates an isomorphism between the *path algebra* of the Bratteli diagram and the chain of semisimple algebras defined by the succession of group algebra inclusions $\mathbb{C}[G_i] \hookrightarrow \mathbb{C}[G_{i+1}]$. In this way the group algebra $\mathbb{C}[G_n]$ is realized as a *multimatrix algebra* (see e.g. [20]).

Matrix elements are now indexed by pairs of paths with a common endpoint. The beauty of the Bratteli diagram formalism lies in the convenient characterization it gives for all types of structured matrices which can arise through the use of Gel'fand-Tsetlin bases.

To begin, consider $a \in G_i \leq G_n$. According to the above explanation, the entries of $\rho(a)$ are indexed by pairs of paths from 1 to ρ in the corresponding Bratteli diagram. Since $a \in G_i$, the matrix entry $\rho_{uv}(a)$

can be nonzero only when paths u and v intersect at level i, i.e., at G_i, and agree from level i to level n. In this case the matrix coefficient $\rho_{vw}(a)$ is independent of the subpath from level i to n. This is precisely the diagrammatic realization of a block diagonal matrix with certain equal sub-blocks. It is for this reason that these are also sometimes called *adapted bases.*

For another example, consider the situation in which $a \in G_n$ centralizes G_j. Using the path algebra formalism, it is not too difficult to show that in this case $\rho_{uv}(a)$ can only be nonzero when u and v agree from level 0 to level j, then vary freely until they necessarily meet at ρ at level n. Here the matrix coefficient depends only upon the pairs of paths between levels j and n.

Finally, any factorization of the group, say into elements in subgroups of the chain as well as their centralizers, then gives a factorization of representations into sums of products of matrix elements, which by the previous discussion are only nonzero in case very particular *compatibility relations* are satisfied among the corresponding sets of contributing paths. Complexity estimates are then computed in terms of counts of compatible diagrams, and also indicate a freedom of choice among a range of possible orders of evaluation, over which the complexity estimates may vary (see e.g. [29, 36] for the case of the symmetric group). The full formalism [29, 34] phrases all of this in the language of bilinear maps and bears some resemblance to the fundamental FFT work of Winograd [56].

2.2.1 State of the art.

This separation of variables approach and its even more elaborate successors have been responsible for the fastest known algorithms for almost all classes of finite groups, including the symmetric groups [29] and their wreath products [35]. These are among the classes of groups for which $O|G|\log^c|G|$ Fourier transform algorithms are known. Other examples include the supersolvable groups [3], while the algorithms for finite matrix groups and Lie groups of finite type still have room for improvement [33, 36].

2.2.2 Finite group quantum FFTs.

By now there are many books and surveys available as introductions to quantum computing. Suffice it to say that the problem of computing a Fourier transform on a finite group in the quantum setting looks formally much like the classical setting. Using the usual bracket notation, over an arbitrary finite group G, this analogously refers to the transformation taking the state

$$\sum_{z\in G} f(z)\,|z\rangle \qquad \text{tothestate} \qquad \sum_{\omega\in\hat{G}} \hat{f}(\omega)_{ij}\,|\omega, i, j\rangle\,,$$

where $f : G \to \mathbb{C}$ is a function with $\|f\|_2 = 1$ and $\hat{f}(\omega)_{ij}$ denotes the i, jth entry of the Fourier transform at the representation ω. The collection $\{|v\rangle\}_v$ represent a set of basis vectors for the Hilbert space in question.

To date, the main applications of quantum computing, or more precisely, the advantages attributed to quantum computing, have relied on the use of commutative Fourier analysis for the discovery of hidden periodicities. This is similar in spirit to the applied motivations behind the implementation of classical Fourier analysis, tasked to the revelation of the periodicities whose superposition comprise the Fourier representation of a given time series.

In the quantum setting "hidden periodicity" refers to the existence of a subgroup H in a given commutative group G such that, for a particular function f defined on G, f is invariant under translation by the hidden subgroup, or equivalently, f is constant on cosets of some nontrivial subgroup H. For example, in Shor's famous quantum factoring algorithm [51] G is the cyclic group $\mathbb{Z}_n^*$ where n is the number we wish to factor, $f(x) = r^x \bmod n$ for a random $r < n$, H is the subgroup of $\mathbb{Z}_n^*$ of index order (r). His quantum solution to the discrete log problem uses $\mathbb{Z}_n \times \mathbb{Z}_n$ for G. In Simon's algorithm for the "XOR-mask" oracle problem [52] G is $\mathbb{Z}_2^n$ with H given by a subgroup of order 2^{n-1}.

Interest in *noncommutative* HSPs derives from the relation to the elusive *graph isomorphism problem*: given undirected graphs A and B, determine if they are related by a simple permutation of the vertices (which preserves the connectivity relations). It would be sufficient to solve efficiently the HSP over the permutation group S_n in order to have an efficient quantum algorithm for graph automorphism (see, e.g., Jozsa [24] for a review). This was the impetus behind the development of the first noncommutative quantum FFT [4] and is, to a large degree, the reason that the noncommutative HSP has remained such an active area of quantum algorithms research.

Most (if not all) quantum algorithms take advantage of a certain quantum parallelism by which the register (at any time a superposition of a collection of states—i.e., a particular vector in the Hilbert space) is updated via application of *local unitary transformations* which are, generally speaking, the tensor product of identity matrices with unitary matrices of bounded size. Many of these can be applied simultaneously, in essence glued together to form a single *quantum gate*, and the full transform is then effected via the application of some sequence of such gates. The efficiency of any algorithm is then measured in terms of the *quantum circuit depth.*

The various sparse factorization FFTs are, in spirit, ready-made for quantum implementations and underly efficient quantum implementations for [23] as well as some solvable groups [44]. A recent quantum adaptation of the separation of variables approach [39] provides a rederivation of Beals's original work, as well extensions to those classes of groups whose classical FFTs benefited from this framework.

2.2.3 Sparse structured factorizations. In [17], sparse representation theoretic factorizations are put to work in helping to find factorizations of given linear transformations. In this work the goal is to describe a matrix as an element of the algebra of intertwining operators between two matrix representations. Having accomplished this, if the representations have sparse factorizations in general (e.g., of the type used in the separation of variables sorts of algorithms), then these can in turn be used to realize a sparse factorization of the original intertwining element. The paper [17] discusses optimal applications of this approach for signal transforms such as the DFT, various DCTs (Discrete Cosine Transforms) and Discrete Hartley Transforms. This approach has been partially automated and is contained in the software library AREP (Abstract REPresentations) [43], which is in turn a part of the very interesting SPIRAL Project [40, 45], a multi-university effort directed at the automatic generation of platform-optimized software.

2.3 Projection-based generalizations of the FFT

The approaches explained above rely on what is commonly known as *decimation in time*, a recursive (or, depending on your point of view, iterative) traversal of the spatial (i.e., group) domain. Decimation in time often goes by the name of *subsampling.*

An alternative, or perhaps more precisely, dual formulation is to instead recurse through the range, iteratively constructing the frequency content of the original data through successive projections which build out increasingly finer orthogonal decompositions. This is the philosophy behind *decimation in frequency*, originally due to Gentleman and Sande [19].

This idea has also found generalization in the context of computing *isotypic decompositions* of a function defined on a group or its homogeneous space. This generalization hinges on the observation that through a judicious choice of group elements and their representing matrices it can be possible to find a collection of projection operators whose application can be scheduled in such a way so as to effect the requisite projections.

2.3.1 Nested projections: The Gentleman-Sande FFT.

A DFT of length n effects the projection of the data f onto the n distinct eigenvectors of the DFT operator given by the sampled exponentials. Equivalently, it is also the projection onto the eigenvectors of the *cyclic shift operator* $T_1^{(n)}$ acting on n-space via $\left(T_1^{(n)} f\right)(j) = f(\overline{j+1})$, where $\overline{j+1}$ indicates that the index is to be interpreted $\pmod n$. The DFT eigenvectors are precisely the basis which diagonalizes the shift operator—i.e., they are also the eigenvectors for $T_1^{(n)}$.

Of course, the operator $T_1^{(n)}$ commutes with any of its powers. Suppose now that $n = pq$. Note that under the action of $T_p^{(n)} = \left(T^{(n)}\right)_1^p$, the vector space $V = \mathbb{C}^n$ decomposes into p orthogonal and $T_p^{(n)}$-invariant q-dimensional subspaces $V_j = \operatorname{span}\{\delta_j, \delta_{j+p}, \ldots, \delta_{j+(q-1)p}\}$, where δ_ℓ denotes the standard ℓth basis vector. It is clear that the action of $T_p^{(n)}$ on any V_j is equivalent to the action of $T_1^{(q)}$ on $\mathbb{C}^q$, thus when restricted to the space V_j it is diagonalizable with eigenvectors and eigenvalues corresponding to the DFT of length q.

Thus, we see that the operator $T_p^{(n)}$ has only q distinct eigenvalues on V, one eigenspace W_j for each character of $\mathbb{Z}_q$, and by symmetry, each of these is of dimension p, and as an eigenspace is $T_p^{(n)}$-invariant. Furthermore, since $T_p^{(n)}$ commutes with $T_1^{(n)}$, there is a basis of simultaneous eigenvectors. Thus

$$W_j = W_j^0 \oplus \cdots \oplus W_j^{p-1}.$$

Note that the original DFT of length n is thus the projection of f onto the W_j^k. This suggests the following algorithm for computing the DFT:

- Stage 1: For $j = 0, \ldots, q-1$, compute $f^{(j)}$, the projection of f onto W_j.
- Stage 2: For each j and each k, compute the projection of $f^{(j)}$ onto W_j^k.

This particular fast Fourier transform is known as the Gentleman-Sande, or *decimation in frequency*, FFT (see [19]).

2.3.2 Gentleman-Sande for finite groups.

The above discussion reveals that the decimation in frequency FFT can be viewed as a sequence of projections onto *isotypic subspaces*. In the commutative case these are the individual eigenspaces. For an arbitrary representation this is the decomposition into invariant subspaces each of which has an irreducible decomposition into copies of a single irreducible subspace. Thus we can attempt to generalize Gentleman-Sande by finding a collection of operators and computing (in some order) projections onto their eigenspaces of a collection of simultaneously diagonalizable linear transformations [32].

For example, suppose that $L(X)$ has three isotypic subspaces V_1, V_2, and V_3. Thus $L(X) = V_1 \oplus V_2 \oplus V_3$ and each $f \in L(X)$ may be written uniquely as $f = f_1 + f_2 + f_3$ where $f_i \in V_i$. Additionally, suppose that T and T' are diagonalizable linear transformations on $L(X)$ such that the eigenspaces of T are $V_1 \oplus V_2$ and V_3, and the eigenspaces of T' are V_1 and $V_2 \oplus V_3$. We may therefore compute the f_i by first projecting f onto the eigenspaces of T to compute $f_1 + f_2$ and f_3, and then projecting both $f_1 + f_2$ and f_3 onto the eigenspaces of T' to compute f_1, f_2 and f_3. Note that each computation is done with respect to a fixed basis of $L(X)$. This process of decomposing $L(X) = V_1 \oplus V_2 \oplus V_3$ is illustrated in Figure 3.

$$\begin{tikzcd}
& & V_1 \oplus V_2 \oplus V_3 \arrow[dl] \arrow[dr] & \\
& V_1 \oplus V_2 \arrow[dl] \arrow[dr] & & V_3 \arrow[d] \\
V_1 & & V_2 & V_3
\end{tikzcd}$$

Figure 3. Decomposing $L(X) = V_1 \oplus V_2 \oplus V_3$ using T and T'.

We call the pair $\{T, T'\}$ *a separating set for* $L(X)$ because it allows us to *separate* a representation into its isotypic components.

From this point of view, the Gentleman-Sande FFT first computes a projection of the data onto the isotypic decomposition corresponding to the subgroup generated by $T_p^{(n)}$ (isomorphic to $\mathbb{Z}_q$) and then further decomposing each of these according to the decomposition of $T_1^{(n)}$ (isomorphic to the fill group G). Thus $\{T_p^{(n)}, T_1^{(n)}\}$ is a separating set for $\mathbb{Z}_{pq}$.

More generally, suppose now that $\{T_1, \ldots, T_k\}$ is a collection of diagonalizable linear transformations on a vector space V whose eigenspaces are direct sums of the isotypic subspaces of V. For each isotypic subspace V_i, let $c_i = (\mu_{i1}, \ldots, \mu_{ik})$ be the k-tuple of eigenvalues where, for $1 \leq j \leq k$, μ_{ij} is the eigenvalue of T_j associated to V_i. If $c_i \neq c_{i'}$ whenever $V_i \neq V_{i'}$, then we say that $\{T_1, \ldots, T_k\}$ is a *separating set* for V.

The existence of a separating set $\{T_1, \ldots, T_k\}$ for V means that the computation of the isotypic projections of $v \in V$ can be achieved through a series of eigenspace projections as follows:

Stage 1 Compute the projections of v onto each eigenspace for T_1.

Stage 2 For each $i > 1$, iteratively compute the projections of the projections previously computed for T_{i-1} onto each of the eigenspaces of T_i.

It is not difficult to see that the computed projections at stage k are precisely the isotypic projections of the vector v.

We may easily find separating sets for V by looking to the conjugacy classes $C_1, \ldots, C_h$ of G. In particular, if $T_j = \sum_{c \in C_j} \rho(c)$ is the *class sum* of C_j (with respect to ρ) and $\mu_{ij} = |C_j|\chi_i(C_j)/d_i$, then the class sum T_j is a diagonalizable linear transformation on V whose eigenspaces are direct sums of isotypic subspaces, and μ_{ij} is the eigenvalue of T_j that is associated to the isotypic subspace V_i ([32]).

The complete collection of class sums forms a separating set of V. We may, however, be able to find much smaller separating sets than the complete collection of class sums. For example, the Gentleman-Sande FFT uses approximately $\log n$ of the n conjugacy classes (since the group is commutative each element forms a conjugacy class). Other specific examples where this gives a savings include the homogeneous spaces formed from distance transitive graphs and their symmetry groups as well as quotients of the symmetric group [32].

The efficiency of this approach depends on an efficient eigenspace projection method. Since the separating sets we use consist of real symmetric matrices, in [32] *Lanczos iteration* is used. This is an algorithm that may be used to efficiently compute the eigenspace projections of a real symmetric matrix when, as in all of our examples, it has relatively few eigenspaces and when it may be applied efficiently to arbitrary vectors, either directly or through a given subroutine (see, e.g., [42]). Implicit in this iterated projection are notions of *multiresolution analysis.* See [18] for recent group theoretic interpretations of this.

2.4 Open questions for finite group FFTs

Other groups for which highly improved (but not $O(|G| \log^c |G|)$) algorithms have been discovered include the matrix groups over finite fields, and more generally, the Lie groups of finite type. See [37] for pointers to the literature. There is much work to be done finding new classes of groups which admit fast transforms, and improving on the above results. The ultimate goal is to settle or make progress on the following conjecture:

Conjecture. *There exist constants c_1 and c_2 such that for any finite group G, there is a complete set of irreducible matrix representations for which the Fourier transform of any complex function on G may be computed in fewer than $c_1|G|\log^{c_2}|G|$ scalar operations.*

Perhaps progress toward this goal will require new techniques. Indeed, it does seem as though the separation of variables approach has been pushed almost as far as it can go. One place to look, and indeed one of the more intriguing open questions in the development of FFT techniques, is for generalizations of those commutative FFT methods which are used for groups of prime order: Rader's prime FFT [46] and the "chirp-z transform" (the "chirp" here refers to radar chirp) [6, 47].

Both of these algorithms use an idea that rewrites the DFT (of prime length p) at nonzero frequencies in terms of a convolution of length $p-1$ (which, since it is composite, can be computed efficiently using other FFT methods) while computing the DFT at the zero frequency directly. Rader's prime FFT uses a generator g of $\mathbb{Z}/p\mathbb{Z}^\times$, a cyclic group (under multiplication) of order $p-1$, to write $\widehat{f}(g^{-b})$ as

$$\hat{f}(g^{-b}) = f(0) + \sum_{a=0}^{p-2} f(g^a)e^{2\pi i g^{a-b}/p}. \tag{10}$$

The summation in (10) has the form of a convolution on $\mathbb{Z}/(p-1)\mathbb{Z}$, of the sequence $f'(a) = f(g^a)$, with the function $z(a) = exp^{2\pi i g^a/p}$.

Rabiner et al. [47] (see also [6]) make the change of variables $jk = (j^2 + k^2 - (j-k)^2)/2$ to obtain

$$\hat{f}(k) = \omega^{k^2/2} \sum_{j=0}^{N-1} \left(f(j)\omega^{j^2/2}\right) \omega^{(j-k)^2/2}.$$

This is a non-cyclic convolution of the sequence $\left(f(j)\omega^{j^2/2}\right)$ with the sequence $\left(\omega^{-j^2/2}\right)$, and may be performed using a cyclic convolution of any length $M \geq 2N$. Note that this gives an approach which rewrites the DFT in terms of a convolution that does not depend on N being prime. This method is commonly known as the *chirp-z transform.*

The discovery of noncommutative generalizations of these ideas would be very, very interesting.

3. FFTs for compact groups

The DFT and FFT also have a natural extension to (continuous) compact groups as well. The terminology "discrete Fourier transform" derives from the fact that the algorithm was originally designed to compute the (possibly approximate) Fourier transform of a continuous signal from a discrete collection of sample values.

Under the simplifying assumption of periodicity a continuous function may be interpreted as a function on the unit circle, and compact commutative group, S^1. Any such function f has a *Fourier expansion* defined as

$$f(t) = \sum_{l \in \mathbf{Z}} \widehat{f}(l) e^{-2\pi i l t} \tag{11}$$

where

$$\widehat{f}(l) = \langle f, e_l \rangle = \int_0^1 f(t) e^{2\pi i l t} dt. \tag{12}$$

If $\widehat{f}(l) = 0$ for $|l| \geq N$, then f is *band-limited* with *band-limit* N and there is a *quadrature rule* or *sampling theory* for f, meaning that the Fourier coefficients of any such f can be computed as a summation using only a finite set of samples. Thus,

$$\widehat{f}(l) = \sum_{k=0}^{2N-2} \frac{1}{(2N-1)} f\left(\frac{k}{2N-1}\right) e^{2\pi i k l/(2N-1)} \tag{13}$$

where the factor $\frac{1}{2N-1}$ should be viewed as a (constant) weight function with support at the equispaced points $\{\frac{k}{2N-1}\}_{k=0}^{2N-2}$ (where the circle and unit interval have been identified). The FFT then efficiently computes these Fourier coefficients.

A more general framework capable of encompassing all continuous compact groups and their quotients is easily stated: the irreducible representations of a compact group G are all finite-dimensional, and any square-integrable function f (with respect to Haar measure) has an expansion in terms of irreducible matrix elements

$$f = \sum_{\lambda \in \Lambda} \sum_{j,k=1}^{d_\lambda} \widehat{f}(\lambda)_{jk} T^\lambda_{jk} \tag{14}$$

where Λ is some countable set, T^λ denotes an irreducible representation of degree $d_\lambda < \infty$, and the implied convergence is in the mean. The Fourier coefficients $\{\widehat{f}(\lambda)_{jk}\}$ are computed by integrals

$$\widehat{f}(\lambda)_{jk} = d_\lambda \langle f, T^\lambda_{jk} \rangle = d_\lambda \int_G f(x) T^\lambda_{jk}(x) dx \tag{15}$$

where dx denotes (the translation-invariant) Haar measure.

In turn, a general FFT schema then requires a formulation of the notion of *bandwidth*, accompanied by a corresponding *sampling theory*, and an algorithmic component for the efficient evaluation of the quadrature, or *FFT*.

3.1 Applications

To date, it is the group of rotations in three space, $SO(3)$, where most of the applications for FFTs on continuous, noncommutative compact groups have been found. Its representation theory is effectively that of the theory of *spherical harmonics*. One large source of applications comes from the climate modeling community (see e.g. [53]) where spherical harmonics are used for spectral methods approaches to solving the relevant PDEs in spherical geometry. Further applications are to be found outside the atmosphere, as spherical harmonics expansions of the CMB are the source of new information about its fine scale inhomogeneities which hope to provide new information about the shape of space and origins of the universe.

Most recently, FFTs for $SO(3)$, as applied to the development of fast convolution algorithms on $SO(3)$ [25], have been used to develop search algorithms for shape databases.

3.2 FFTs for compact groups—the work of Maslen

The first FFT for a noncommutative and continuous compact group was the efficient spherical harmonic expansion algorithm discovered by J. Driscoll and D. Healy [15]. In this case, the Fourier expansion of a function on the 2-sphere, viewed as a function on $SO(3)/SO(2)$ (with $SO(2)$ identified with the rotations that leave the north pole fixed) has a natural notion of bandwidth given by the degree of the spherical harmonic. A sampling rule on the 2-sphere, equiangular in both latitude and longitude, gives a quadrature rule, and a function of bandwidth B (and $O(B^2)$ Fourier coefficients) requires $O(B^2)$ points. The story is completed with a fast algorithm ($O(N^{3/2}\log^2 N)$ operations for $N = B^2$) that uses the three-term recurrence satisfied by the Legendre functions to produce a divide and conquer algorithm for its efficient evaluation.

Some years later, this work was extended to the full compact setting by D. Maslen: a general notion of bandwidth consistent with the commutative and spherical notions [31], a sampling rule [30], and finally an FFT which also relies on three term recurrence relations satisfied by related orthogonal polynomial systems. What follows is a brief summary of this work

There is a natural definition of band-limited in the compact case, encompassing those functions whose Fourier expansion has only a finite number of terms. The simplest version of Maslen's theory is as follows:

Definition 1. *Let $\mathcal{R}$ denote a complete set of irreducible representations of a compact group G. A* system of band-limits on G *is a decomposition of $\mathcal{R} = \cup_{b\geq 0}\mathcal{R}_b$ such that*

- *[1] $|\mathcal{R}_b| < \infty$ for all $b \geq 0$;*
- *[2] $b_1 \leq b_2$ implies that $\mathcal{R}_{b_1} \subseteq \mathcal{R}_{b_2}$;*
- *[3] $\mathcal{R}_{b_1} \otimes \mathcal{R}_{b_2} \subseteq span_{\mathbf{Z}}\mathcal{R}_{b_1+b_2}$.*

Let $\{\mathcal{R}_b\}_{b\geq 0}$ be a system of band-limits on G and $f \in L^2(G)$. f is band-limited with band-limit *b if $\widehat{f}(T^{\lambda}_{jk}) = 0$ for all $\lambda \notin \mathcal{R}_b$.*

The case $G = S^1$ provides the classical example. If $\mathcal{R}_b = \{\chi_j : |j| \leq b\}$ where $\chi_j(z) = z^j$, then $\chi_j \otimes \chi_k = \chi_{j+k}$ and the corresponding notion of band-limited (as per Definition 1) coincides with the usual notion.

For a noncommutative example, consider $G = SO(3)$. In this case the irreducible representations of G are indexed by the non-negative integers with V_λ the unique irreducible representation of dimension $2\lambda + 1$. Let $\mathcal{R}_b = \{V_\lambda : \lambda \leq b\}$. The *Clebsch-Gordon relations*

$$V_{\lambda_1} \otimes V_{\lambda_2} = \sum_{j=|\lambda_1-\lambda_2|}^{\lambda_1+\lambda_2} V_j \tag{16}$$

imply that this is a system of band-limits for $SO(3)$. When restricted to the quotient $S^2 \cong SO(3)/SO(2)$, band-limits are described in terms of the highest order spherical harmonics that appear in a given expansion.

Maslen's most general setting for a notion of band-limit develops a theory of band-limited elements for any *filtered module* over a *filtered algebra*. In the case of a connected compact Lie group G, $\mathcal{R}_s$ is defined to be the set of all matrix elements that come from representations whose highest weight is at most s. For the matrix groups $SU(r+1), Sp(r), SO(2r+1)$, it is possible to choose a norm for which $\mathcal{R}_1$ is the span of all matrix representations with highest weight given by a fundamentally analytically integral dominant weight or zero. Bandwidth is thus defined in terms of lengths of factorizations in sums of products of such elements expressing a given matrix element [31].

The importance of developing the band-limited theory is that in this setting there exists a sampling theory or quadrature rule that allows the

Fourier coefficients to be computed exactly as finite sums. The following is the content of [30], once the notion of bandlimit is arranged.

THEOREM 1 *Let G be compact with a system of band-limits $\{\mathcal{R}_b\}_b$. For any band-limit b, there exists a finite set of points $X_b \subset G$ such that for any function $f \in L^2(G)$ of band-limit b,*

$$\widehat{f}(T^{\lambda}_{jk}) = \sum_{x \in X_b} f(x) T^{\lambda}_{jk}(x) w(x) \tag{17}$$

for all $\lambda \in \mathcal{R}_b$ and some weight function w on X_b.

Theorem 1 reduces the integrals (15) to summations, so that efficient algorithms can now be designed to perform the computations (17). For the classical groups $U(n), SU(n), Sp(n)$, a system of band-limits $\mathcal{R}^n_b$ is chosen with respect to a particular norm on the dual of the associated Cartan subalgebra. Such a norm $\| \cdot \|$ (assuming that it is invariant under taking duals, and $\|\alpha\| \leq \|\beta\| + \|\gamma\|$ for α occurring in $\beta \otimes \gamma$) defines a notion of band-limit given by all α with norm less than a fixed b. The associated sampling sets X^n_b are contained in certain one-parameter subgroups.

Implicit here are certain *discrete special function transforms*, which can often be reduced to certain *discrete polynomial transforms*

$$\widehat{f}(P_j) = \sum_{k=0}^{N-1} f(k) P_j(x_k) w_k \tag{18}$$

where $P_0, \ldots, P_{N-1}$ are a set of linearly independent polynomials with complex coefficients, $\{x_0, \ldots, x_{N-1}\}$ are a set of N distinct complex points and $\{w_0, \ldots, w_{N-1}\}$ is a set of positive weights. The case of the DFT comes from choosing equispaced roots of unity for sample points, equal weights of one, and $P_j(x) = x^j$. Direct calculation of all the $\widehat{f}(P_j)$ clearly requires N^2 operations.

If the P_j make up a family of *orthogonal polynomials*, then fast algorithms exist to speed the calculation. Here the idea is to use the *three-term recurrence* satisfied by these polynomials to create a divide-and-conquer algorithm which reduces transforms of degree n to sums of transforms of degree less than n, ultimately providing an $O(n \log^2 n)$ algorithm. (See [16] and references therein.)

By using these sorts of complexity estimates, together with a sampling theory and a careful organization of the calculation (using the diagrammatic techniques explained above), Maslen is able to derive efficient algorithms for all the classical groups.

Theorem 2 *([31], Theorem 5)*
Assume $n \geq 2$.

(i) For $U(n)$, $T_{X_b^n}(\mathcal{R}_b^n) \leq O(b^{dim U(n)+3n-3})$

(ii) For $SU(n)$, $T_{X_b^n}(\mathcal{R}_b^n) \leq O(b^{dim SU(n)+3n-2})$

(iii) For $Sp(n)$, $T_{X_b^n}(\mathcal{R}_b^n) \leq O(b^{dim Sp(n)+6n-6})$

where $T_{X_b^n}(\mathcal{R}_b^n)$ *denotes the number of operations needed for the particular sample set* X_b^n *and representations* $\mathcal{R}_b^n$ *for the associated group.*

3.3 Approximate techniques.

In the bandlimited case, Maslen's techniques are exact, in the sense that if computed in exact arithmetic, they yield exact answers. Of course, in any actual implementation, errors are introduced and the utility of an algorithm will depend highly on its numerical stability.

There are also "approximate methods", approximate in the sense that they guarantee a certain specified approximation to the exact answer that depends on the running time of the algorithm. For computing Fourier transforms at non-equispaced frequencies, as well as spherical harmonic expansions, the fast *multipole method* and its variants are used [21]. Multipole-based approaches efficiently compute these quantities approximately, in such a way that the running time increases by a factor of $\log(\frac{1}{\epsilon})$ where ϵ denotes the precision of the approximation. Another approach is via the use of *quasi-classical frequency estimates* for the relevant transforms [38]. It would be interesting to generalize these sorts of techniques to compact groups and their quotients.

3.4 Open question

Maslen's work effectively creates uniform sampling grids with concomitant quadrature rules, but it may be possible that some applications may require *nonuniform grids*. In the commutative case, examples include applications in medical imaging and other forms of noninvasive testing. Noncommutative examples might include astrophysical, weather, and climate data. The corresponding measurements are rarely equidistributed (in particular, there are many large uninhabited regions in which the data is never taken) and, in fact, these two variable expansions generally use grids which evenly sample in one direction, but use Legendre points in the other [41, 53]. For example, as applied to the analysis of the cosmic microwave background, this is meant to provide a sampling that is sparse at "the center," which corresponds to avoiding our own galaxy. It would seem to be of great interest to push forward the wealth of work done in the commutative setting (see e.g. [1] and the many examples therein).

4. Noncompact groups

Much of modern signal processing relies on the understanding and implementation of Fourier analysis for $L^2(\mathbf{R})$, i.e., the noncompact commutative group $\mathbf{R}$. It is only fairly recently that noncommutative, noncompact examples have begun to attract significant attention.

In this area some of the most exciting work is being done by G. Chirikjian and his collaborators. They have been concerned primarily with the Euclidean motion group $SE(n)$. Recall that the motion groups are given as semidirect products of $\mathbb{R}^n$ with the rotation groups $SO(n)$, realized as $n+1$ by $n+1$ matrices of the form

$$\begin{pmatrix} A & v \\ \mathbf{0}_n & 1 \end{pmatrix}$$

where $A \in SO(n)$, $v \in \mathbb{R}^n$ and $\mathbf{0}_n$ denotes the all zero row vector of length n. This provides an algebraic mechanism for gluing together the group of additive translations with rotations.

Their motivation comes from a diverse collection of applications, ranging among robotics, molecular modeling and pattern matching. Applications to robotics come from the problem of *workspace determination* for *discretely actuated manipulators.* A standard example is a robot arm and a standard problem in motion planning is to determine the set of reachable configurations, as well as to plan a path to move from one configuration to another. The configuration of the end of the arm can be described with two parameters: a position in space (a vector in $\mathbb{R}^n$ for $n = 2$ or 3) as well as an orientation (an element in $SO(n)$), i.e., an element of $SE(n)$. One sort of design paradigm is to build a robot arm as an assembly of a sequence of basic modules, so that the arm takes on a worm-like or cillial form. Any single basic unit will have some finite discrete set of reachable states, defining a discrete probability density in $SE(n)$. This is the *workspace* of a single unit. The *workspace* of the full arm (defined as the linked assembly of m of these basic units) is then given as the m-fold convolution of the fundamental workspace. This is called the *workspace density.* Applications to polymer science are analogous, with similar modeling considerations used to describe the motion of a given end of a polymer (such as DNA) relative to its other end. These are but two examples. For details, as well as other applications see [9] and the many references therein.

Just as the classical FFT provides efficient computation of convolutions on the line or circle, so does an FFT for $SE(n)$ allow for efficient convolution in this setting, replacing direct convolution by FFTs, matrix multiplications and inverse FFTs.

In a collection of papers (see [9]) Chirikjian and Kyatkin create a computational framework for working with the representation theory of $SE(3)$ acting on $\mathbb{R}^3$. The matrix elements in this case are known and involve spherical harmonics, half-integer Bessel functions, glued together with Clebsch-Gordan coefficients. Explicitly, they find themselves in the position of having to compute (for a function $f(\mathbf{r}, R)$ of compact support on $SE(3)$)

$$\begin{aligned}&\hat{f}^s_{\ell',m';\ell,m}(p) = \\ &\quad \int_{\mathbf{u}\in S^2}\int_{\mathbf{r}\in\mathbb{R}^3}\int_{R\in SO(3)} f(\mathbf{r}, R) h^s_{\ell,m}(\mathbf{u}) \times \\ &\quad e^{ip\mathbf{u}\cdot\mathbf{r}} \sum_{n=-\ell'}^{\ell'} \overline{U^{\ell'}_{nm'}(R) h^s_{\ell' n}(\mathbf{u})} d\mathbf{u} d^3 r dR. \end{aligned} \tag{19}$$

where the h's and U's are defined in terms of generalized Legendre functions.

Computation is now effected via a host of discretizations on $\mathbb{R}^3$, $SO(3)$, S^2, and the dual index p, as well as some assumptions on the number of harmonics used to describe f. The exponent $p\mathbf{u}$ implies the need to convert from a rectangular to a polar $\mathbb{R}^3$ grid and so there is also an interpolation (through splines) used. The complexity of the final separation of variables style algorithm is then given by gluing together all the appropriate fast algorithms (FFTs, fast interpolation, fast spherical harmonic expansions, fast Legendre expansions).

The details of this analysis are found in [26]. This paper also contains an analogous discussion for $SE(2)$ as well as the *discrete motion groups* defined as the semidirect product of translations ($\mathbb{R}^3$) with any of the (finite number of) finite subgroups of $SO(3)$.

4.1 Open questions

To date, the techniques used here are approximate in nature and interesting open problems abound. Possibilities include the formulation of natural sampling (regular and irregular), band-limiting and time-frequency theories. The exploration of other special cases beyond the special Euclidean groups, such as semisimple Lie groups (see [2] for a beautifully written succinct survey of the Harish-Chandra theory), is also intriguing. “Fast Fourier transforms on semisimple Lie groups” has a nice ring to it!

Acknowledgments

This work was completed with partial support from the NSF under grants CCR-0219717 and F49620-00-1-0280.

References

[1] A. Aldroubi and K. Grochenig. Non-uniform sampling and reconstruction shift-invariant spaces. *SIAM Rev.*, **43** No. 4 (2001), pp. 585–620.

[2] J. Arthur, *Harmonic analysis and group representations*, Notices. Amer. Math. Soc., **47**(1) (2000), 26–34.

[3] U. Baum. Existence and efficient construction of fast Fourier transforms for supersolvable groups. *Comput. Complexity* **1** (1991), 235–256.

[4] Robert Beals. Quantum computation of Fourier transforms over symmetric groups. In ACM, editor, *Proceedings of the twenty-ninth annual ACM Symposium on the Theory of Computing: El Paso, Texas, May 4–6, 1997*, pages 48–53, New York, NY, USA, 1997. ACM Press.

[5] T. Beth. *Verfahren der schnellen Fourier-Transformation.* B. G. Teubner, Stuttgart, 1984.

[6] L. Bluestein, *A linear filtering approach to the computation of the discrete Fourier transform*, IEEE Trans. **AU-18** (1970), 451-455.

[7] O. Bratteli. Inductive limits of finite dimensional C^*-algebras. *Trans. Amer. Math. Soc.* **171** (1972), 195–234.

[8] O. Brigham. *The Fast Fourier Transform and its Applications*. Prentice Hall, NJ, 1988.

[9] G. S. Chirikjian and A. B. Kyatkin. *Engineering applications of noncommutative harmonic analysis*, CRC Press, FL, 2000.

[10] M. Clausen and U. Baum. *Fast Fourier transforms*, Bibliographisches Institut, Mannheim, 1993.

[11] W. Cochran, et. al., What is the fast Fourier transform?, *IEEE Transactions on Audio and Electroacoustics* Vol. AU-15, No. 2 (1967), 45–55.

[12] J. W. Cooley and J. W. Tukey. An algorithm for machine calculation of complex Fourier series, *Math. Comp.*, **19** (1965), 297–301.

[13] P. Diaconis. A generalization of spectral analysis with application to ranked data, *Ann. Statist.*, **17**(3), (1989), 949–979.

[14] P. Diaconis. *Group representations in probability and statistics*, IMS, Hayward, CA, 1988.

[15] J. R. Driscoll and D. Healy. Computing Fourier transforms and convolutions on the 2-sphere. *Proc. 34th IEEE FOCS*, (1989) pp. 344–349 (extended abstract); *Adv. in Appl. Math.* **15** (1994), 202–250.

[16] J. Driscoll and D. Healy and D. Rockmore. Fast discrete polynomial transforms with applications to data analysis for distance transitive graphs, *SIAM J. Comput.*, **26**, No. 4, (1997), 1066–1099.

[17] S. Egner and M. Püschel. Symmetry-based matrix factorization. *J. Symb. Comp*, to appear.

[18] R. Foote. An algebraic approach to multiresolution analysis, (2003), submitted for publication.

[19] W. Gentleman and G. Sande. Fast Fourier transform for fun and profit, *Proc. AFIPS, Joint Computer Conference*, **29**, (1966), 563-578.

[20] F. Goodman, P. de al Harpe, and V. F. R. Jones, *Coxeter graphs and towers of algebras*, Springer-Verlag, New York, 1989.

[21] L. Greengard and V. Rokhlin. A fast algorithm for particle simulations. *J. Comput. Phys.* **73** (1987) 325–348.

[22] M. T. Heideman, D. H. Johnson and C. S. Burrus. Gauss and the history of the fast Fourier transform, *Archive for History of Exact Sciences*, **34** (1985), no. 3, 265–277.

[23] Peter Höyer. Efficient quantum transforms. Technical Report quant-ph/9702028, Quantum Physics e-Print Archive, 1997.

[24] Richard Jozsa. Quantum factoring, discrete logarithms and the hidden subgroup problem. *Computing, Science, and Engineering*, **3**(2), pp. 34–43, (2001).

[25] M. Kazhdan, T. Funkhouser and S. Rusinkiewicz. Rotation invariant spherical harmonic representation of 3D-shape descriptors. *Symposium on Geometry Processing* June, (2003) pp. 167–175.

[26] A. B. Kyatkin and G. S. Chirikjian. Algorithms for fast convolutions on motion groups. *App. Comp. Harm. Anal.* **9**, 220–241 (2000).

[27] G. Lebanon and J. Lafferty. Cranking: Combining rankings using conditional probability models on permutations, in *Machine Learning: Proceedings of the Nineteenth International Conference*, San Mateo, CA: Morgan Kaufmann, 2002.

[28] J. Lafferty and D. N. Rockmore. Codes and iterative decoding on algebraic expander graphs, in *Proceedings of International Symposium on Information Theory and its Application, Honolulu, Hawaii*, 2000.

[29] D. K. Maslen. The efficient computation of Fourier transforms on the symmetric group, *Math. Comp.* **67**(223) (1998), 1121–1147.

[30] D. K. Maslen. Sampling of functions and sections for compact groups. in *Modern Signal Processing*, D. N. Rockmore and D. M. Healy, eds., Cambridge University Press, to appear.

[31] D. K. Maslen. Efficient computation of Fourier transforms on compact groups, *J. Fourier Anal. Appl.* **4**(1) (1998), 19–52.

[32] D. K. Maslen, M. Orrison, and D.N. Rockmore. Computing Isotypic Projections with the Lanczos Iteration. *SIAM J. Matrix Analysis*, to appear.

[33] D. K. Maslen and D. N. Rockmore. Separation of variables and the computation of Fourier transforms on finite groups. I. *J. Amer. Math. Soc.* **10** (1997), no. 1, 169–214.

[34] D. K. Maslen and D. N. Rockmore. Separation of variables and the computation of Fourier transforms on finite groups. II. In preparation.

[35] D. K. Maslen and D. N. Rockmore. The Cooley-Tukey FFT and group theory, *Notices Amer. Math. Soc* **48** (2001), no. 10, 1151–1160.

[36] D. Maslen and D. N. Rockmore. Double coset decompositions and computational harmonic analysis on groups.. *J. Fourier Anal. Appl.* **6**(4), 2000, pp. 349–388.

[37] D. Maslen and D. N. Rockmore. Generalized FFTs—a survey of some recent results, in *Groups and computation, II (New Brunswick, NJ, 1995)*, Amer. Math. Soc., Providence, RI, 1997, pp. 183–237.

[38] M. P. Mohlenkamp. A fast transform for spherical harmonics, *J. Fourier Anal. Appl.*, **5** no. 2–3 (1999), 159–184.

[39] C. Moore, D. N. Rockmore, and A. Russell. Generic Quantum Fourier Transforms. Technical Report quant-ph/0304064, Quantum Physics e-print Archive, 2003.

[40] J.M.F. Moura, J. Johnson, R.W. Johnson, D. Padua, V. Prasanna, M. Püschel, and M.M. Veloso, SPIRAL: Automatic Library Generation and Platform-Adaptation for DSP Algorithms, 1998, http://www.ece.cmu.edu/∼spiral/

[41] S.P. Oh, D.N. Spergel, and G. Hinshaw. An Efficient Technique to Determine the Power Spectrum from the Cosmic Microwave Background Sky Maps. *Astrophysical Journal*, **510** (1999), 551–563.

[42] B. Parlett. *The symmetric eigenvalue problem.* Prentice-Hall Inc., Englewood Cliffs, N.J. 1980.

[43] M. Püschel, S. Egner, and T. Beth. In *Computer Algebra Handbook, Foundations, Applications, Systems.* Eds. J. Grabmeier, E. Kaltofen, V. Weispfenning, Springer 2003, pp. 461-462.

[44] M. Püschel, M. Rötteler, and T. Beth. Fast quantum Fourier transforms for a class of non-abelian groups. (*Proc. AAECC 99*, LNCS 1719, Springer-Verlag, pp. 148-159.

[45] M. Püschel, B. Singer, J. Xiong, J.M.F. Moura, J. Johnson, D. Padua, M.M. Veloso, and R.W. Johnson, SPIRAL. A Generator for Platform-Adapted Libraries of Signal Processing Algorithms. *J. of High Performance Computing and Applications*, accepted for publication.

[46] C. Rader. Discrete Fourier transforms when the number of data samples is prime, *IEEE Proc.* **56** (1968), 1107–1108.

[47] L. Rabiner, R. Schafer, and C. Rader, *The chirp-z transform and its applications*, Bell System Tech. J. **48** (1969), 1249-1292.

[48] D. N. Rockmore. Some applications of generalized FFTs (An appendix w/D. Healy), in *Proceedings of the DIMACS Workshop on Groups and Computation, June 7-10, 1995.* Eds. L. Finkelstein and W. Kantor, (1997), pp. 329–369

[49] D. N. Rockmore. Fast Fourier transforms for wreath products, *Appl. Comput. Harmon. Anal.*, **2**, No. 3 (1995), 279–292.

[50] J.-P. Serre. *Linear representations of finite groups*, Springer-Verlag, New York, 1977.

[51] P. W. Shor, Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer, *SIAM J. Computing* **26** (1997), 1484–1509.

[52] Daniel R. Simon. On the power of quantum computation. *SIAM Journal on Computing*, 26(5):1474–1483, October 1997.

[53] W. F. Spotz and P. N. Swarztrauber. A Performance Comparison of Associated Legendre Projections. *Journal of Computational Physics*, 168(2), (2001) 339-355.

[54] L. Trefethen and D. Bau III. *Numerical linear algebra.* Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1997.

[55] A. Willsky. On the algebraic structure of certain partially observable finite-state Markov processes. *Inform. Contr.* **38**, 179–212 (1978).

[56] S. Winograd, *Arithmetic complexity of computations*, CBMS-NSF Regional Conference Series in Applied Mathematics, Vol. 33. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, Pa., 1980.