Symmetric FFT's; a general approach

Hans Munthe-Kaas
Division of Mathematical Sciences
Norwegian Institute of Technology
Trondheim, Norway

May 11, 1989

Abstract

We examine the FFT of functions $u : \mathbf{R}^n \to \mathbf{C}$, where u is both periodic in all variables and in addition possess certain symmetries. It is shown that all such functions allow symmetric FFT's, where all the symmetries are used to obtain computational savings. These symmetric FFT's includes the previously known symmetric transforms that are currently used in fast Poisson solvers. In addition new symmetries appear. These can e.g. be used in the construction of fast Poisson solvers over certain 2–dimensional triangular domains. The triangular Poisson solvers have obvious applications in domain decomposition methods. The paper also contains a general description of nonsymmetric multidimensional FFT's. This formalism may be useful for choosing multidimensional FFT algorithms for implementation on parallel computers.

1 Motivation

Suppose we want to solve Poisson's equation $\nabla^2 v = u$ over a two dimensional triangular domain $\Omega \subset \mathbf{R}^2$, limited by three lines l, m and n. Let v satisfy the boundary conditions:

$$\begin{array}{rcl} v & = & 0 \quad \text{on } l \text{ and } m \\ \frac{\partial v}{\partial \eta} & = & 0 \quad \text{on } n \end{array} \tag{1}$$

and let u be real valued. This problem may sometimes be solved by the following procedure: Suppose the u can be extended to a function $\tilde{u}$ defined on $\mathbf{R}^2$ by imposing the following conditions:

$$\begin{array}{lrcl} (o) & \tilde{u}|_\Omega & = & u \\ (i) & u \circ \sigma_l & = & -u \\ (ii) & u \circ \sigma_m & = & -u \\ (iii) & u \circ \sigma_n & = & u \\ (iv) & u^* & = & u \end{array} \tag{2}$$

where σ_l, σ_m and σ_n are reflections of the plane $\mathbf{R}^2$ about the lines l, m and n; $\circ$ denotes function composition and * is complex conjugation.

The Laplacian operator commutes with any isometry (i.e. distance preserving transformation) of $\mathbf{R}^n$, e.g.

$$\nabla^2(\tilde{v} \circ \sigma_l) = (\nabla^2 \tilde{v}) \circ \sigma_l$$

So if $\nabla^2 \tilde{v} = \tilde{u}$, then the symmetries (i)-(iv) carries over to $\tilde{v}$, and $\tilde{v}$ must satisfy the boundary conditions 1. The solution to the boundary value problem is therefore given by $v = \tilde{v}|_\Omega$.

If $\tilde{u}$ is doubly periodic, i.e. if there exists two non-parallel translations τ_1 and τ_2 of $\mathbf{R}^2$ such that

$$\tilde{u} \circ \tau_1 = \tilde{u} \circ \tau_2 = \tilde{u} \tag{3}$$

then $\tilde{u}$ can be expanded in a Fourier series, and it is reasonable to belive that a numerical solution of the Poisson problem can be found by fast Fourier transforms (FFT's).

Several questions arise:

- For what type of geometries and boundary conditions does this approach succeed?
- The extended function possess in addition to the periodicity condition 3 the symmetries (i)-(iv). Does there exist 'symmetric' FFT algorithms where these additional symmetries are used to economize the computations?

Symmetric FFT's are previously studied for periodic functions in a single variable, see [11]. In this paper we define a general class of symmetric functions in arbitrary many variables. It is shown that there for all these functions exists symmetric FFT's. This general theory includes all the previously known 1–dimensional symmetric FFT's. In 2–D several new possibilities arise. These include symmetries derived from Poisson equations over certain triangular domains. Contrary to the more familiar 'rectangular' FFT's, these triangular transforms cannot be decomposed into 1–D symmetric transforms (without breaking the symmetry). The corresponding triangular fast Poisson solvers may be very useful in domain decomposition methods, and spectral domain decomposition methods for solving elliptic PDE's. In 3-D the new symmetric FFT's allows fast Poisson solvers over e.g. a tetrahedron. Some of the symmetric transforms are not naturally interpreted in terms of Poisson problems, but they may have other applications, e.g. in crystal lattice calculations. In a forthcomming paper we show that the symmetric FFT's are closely connected to generalizations of the cyclic reduction algorithm. For problems discretized by finite differences, cyclic reduction may be coupled with symmetric FFT's to obtain faster algorithms, similar to the approach in [10].

The paper is organized as follows: Section 2 describes the FFT algorithms in terms of Fourier analysis on finite commutative groups. This formalism is useful in itself, since it naturally describes different possible FFT algorithms. E.g. multidimensional FFT's can be computed in other ways than the familiar approach of doing a series of one dimensional transforms first in one direction, then in the next direction, etc. The alternative transforms may be attractive for implementation on parallel computers, since the data flow is different from the conventional multidimensional transforms. The number of arithmetical operations are also slightly less.

In section 3 we define a general class of symmetric functions. This is the basis for our symmetric transforms. Section 4 shows how the symmetric functions can be discretized and how symmetric transforms can be constructed in general. Section 5 classifies all the symmetries that arise in connection with 2–D Poisson problems. In section 6 we describe the implementation of a Poisson solver over an equilateral triangle. It is shown that the symmetric FFT's may either be regarded as an expansion of symmetric functions in general eigenfunctions of the Laplacian, or as an expansion of a general function defined on a domain in eigenfunctions of the Laplacian satisfying homogeneous boundary conditions at the boundary of the domain. The final section 7 contains some concluding remarks.

2 Fourier transforms on finite Abel groups

Let u be a vector of length N. The discrete Fourier transform (DFT) and the inverse discrete Fourier transform (IDFT) of u are usually defined as

$$\hat{u}(k) = \frac{1}{N} \sum_{j=0}^{N-1} u(j) e^{2\pi i j k/N}$$

and

$$\check{u}(k) = \sum_{k=0}^{N-1} u(j) e^{-2\pi i j k/N}$$

The fast Fourier transform (FFT) is a family of divide and conquer algorithms for computing $\hat{u}$ and $\check{u}$. The computational cost of these algorithms is $\mathcal{O}(n \log n)$ operations, compared to $\mathcal{O}(n^2)$ operations for a direct evaluation of the sum.

For the discussion of multidimensional symmetric FFTs, it is useful to phrase the theory of DFT and FFT in the language of Fourier transforms on finite Abel groups. Since this formalism is not found in the numerical literature, we are giving the necessary details below. The basic group theoretic concepts of this section are found in any textbook of abstract algebra, see e.g. [4]. For an introduction to Fourier analysis on groups, see [3] and the references therein.

A *group* is a set W with a multiplication function $\cdot : \mathcal{W} \times \mathcal{W} \mapsto \mathcal{W}$ satisfying the following conditions:

1. $\mathbf{p} \cdot (\mathbf{q} \cdot \mathbf{r}) = (\mathbf{p} \cdot \mathbf{q}) \cdot \mathbf{r}$ for all $\mathbf{p}, \mathbf{q}, \mathbf{c} \in \mathcal{W}$
2. There exists an identity e such that $\mathbf{e} \cdot \mathbf{p} = \mathbf{p}$ for all $\mathbf{p} \in \mathcal{W}$
3. For all p there exists an inverse $\mathbf{p}^{-1}$ such that $\mathbf{p}^{-1} \cdot \mathbf{p} = \mathbf{e}$

We usually write $\mathbf{pq}$ instead of $\mathbf{p} \cdot \mathbf{q}$.

A group is *Abelian* (or commutative) if $\mathbf{pq} = \mathbf{qp}$ for all elements p and q in the group. A finite Abel group can be defined as a direct product

$$\mathcal{G} = \mathcal{Z}_{m_1} \otimes \mathcal{Z}_{m_2} \otimes \cdots \otimes \mathcal{Z}_{m_k}$$

where $\mathcal{Z}_{m_i}$ are cyclic groups of m_i symbols, i.e. the integers $0, 1, \ldots, m_i - 1$ under addition modulo m_i. An element $\mathbf{q} \in \mathcal{G}$ can be represented by a k–dimensional integer vector with components $0 \leq q_i < m_i$. The product $\mathbf{pq}$ of two elements $\mathbf{p}, \mathbf{q} \in \mathcal{G}$ is defined as a vector $\mathbf{r} \in \mathcal{G}$ with components

$$r_i = (p_i + q_i) \bmod m_i$$

The identity of $\mathcal{G}$ is the element $\mathbf{e} = (0, 0, \ldots, 0)^T$.

A *subgroup* of $\mathcal{G}$, written $\mathcal{H} < \mathcal{G}$, is a subset $\mathcal{H} \subset \mathcal{G}$ such that $\mathbf{h}_i \mathbf{h}_j \in \mathcal{H}$ for all $\mathbf{h}_i, \mathbf{h}_j \in \mathcal{H}$. The subgroup *generated* by a set $\{\mathbf{h}_1, \ldots, \mathbf{h}_i\}$, written $\langle \mathbf{h}_1, \ldots, \mathbf{h}_i \rangle$, is the smallest subgroup of $\mathcal{G}$ containing $\{\mathbf{h}_1, \ldots, \mathbf{h}_i\}$. A subgroup $\mathcal{H}$ partitions $\mathcal{G}$ in a union of *right cosets* $\mathcal{H}\mathbf{g}$ defined by $\mathcal{H}\mathbf{g} = \{ \mathbf{hg} \mid \mathbf{h} \in \mathcal{H} \}$, where $\mathbf{g} \in \mathcal{G}$. Left cosets $\mathbf{g}\mathcal{H}$ are defined similarly. If $\mathcal{H}\mathbf{g} = \mathbf{g}\mathcal{H}$ for all $\mathbf{g} \in \mathcal{G}$, the subgroup is called *normal*, (so all subgroups of Abel groups are normal). The cosets of a normal subgroup form a group called the *quotient group*, $\mathcal{K} = \mathcal{G}/\mathcal{H}$, where the elements of $\mathcal{K}$ are the cosets of $\mathcal{H}$, and the product of two cosets is defined by $(\mathcal{H}\mathbf{p}) \cdot (\mathcal{H}\mathbf{q}) = \mathcal{H}(\mathbf{pq})$.

A *homomorphism* of $\mathcal{G}$ is a mapping Φ from $\mathcal{G}$ into another group $\mathcal{G}'$ such that $\Phi(\mathbf{qr}) = \Phi(\mathbf{q})\Phi(\mathbf{r})$. If Φ is bijective, it is called an *isomorphism*. An *automorphism* is an isomorphism from $\mathcal{G}$ onto itself. An automorphism of a finite Abel group $\mathcal{G}$ can be written as a linear transformation

$$\mathbf{q} = \alpha\mathbf{p}$$

where α is a $k \times k$ integer matrix with determinant ± 1, i.e. a *unimodular* integer matrix. A *translation* of the group $\mathcal{G}$ is a bijective mapping of $\mathcal{G}$ onto itself defined by

$$\mathbf{g} \mapsto \mathbf{tg} \text{ for all } \mathbf{g} \in \mathcal{G}$$

where $\mathbf{t} \in \mathcal{G}$. The composition of an automorphism and a translation is in this paper called an *isometry* of the group.

A function u on $\mathcal{G}$ is a mapping of $\mathcal{G}$ into the complex numbers: $\mathsf{u}: \mathcal{G} \rightarrow \mathbf{C}$. The set of all functions on $\mathcal{G}$ forms a Hilbert space with the inner product

$$(\mathsf{u}, \mathsf{v}) = \sum_{\mathbf{g} \in \mathcal{G}} \mathsf{u}(\mathbf{g})\mathsf{v}^*(\mathbf{g})$$

The inner product is invariant under isometries of $\mathcal{G}$:

$$\sum_{\mathbf{g} \in \mathcal{G}} \mathsf{u}(\mathbf{g})\mathsf{v}^*(\mathbf{g}) = \sum_{\mathbf{g} \in \mathcal{G}} \mathsf{u}(\alpha(\mathbf{tg}))\mathsf{v}^*(\alpha(\mathbf{tg}))$$

where α is an automorphism and $\mathbf{t} \in \mathcal{G}$.

An important set of functions on $\mathcal{G}$ are the *characters* of the group defined as homomorphisms of $\mathcal{G}$ into the multiplicative group of complex numbers with modulus 1, i.e.:

Definition 1 *A character on $\mathcal{G}$ is a function $\varepsilon(\mathbf{g})$ on $\mathcal{G}$ satisfying*

1. $|\varepsilon(\mathbf{g})| = 1$ *for all* $\mathbf{g} \in \mathcal{G}$.
2. $\varepsilon(\mathbf{pg}) = \varepsilon(\mathbf{p}) \cdot \varepsilon(\mathbf{g})$ *for all* $\mathbf{p}, \mathbf{g} \in \mathcal{G}$.

The set of all characters on $\mathcal{G}$, denoted $\hat{\mathcal{G}}$, is called the *dual group* of $\mathcal{G}$. There is a 1–1 correspondence between the elements of $\mathcal{G}$ and $\hat{\mathcal{G}}$, and the elements of $\hat{\mathcal{G}}$ can be indexed by the elements of $\mathcal{G}$ in the following manner:

Lemma 1 *The characters on $\mathcal{G}$ can be indexed such that*

1. $\hat{\mathcal{G}} = \{ \varepsilon_{\mathbf{g}} \mid \mathbf{g} \in \mathcal{G} \}$
2. $\varepsilon_{\mathbf{g}}(\mathbf{p}) = \varepsilon_{\mathbf{p}}(\mathbf{g})$ *for all* $\mathbf{p}, \mathbf{g} \in \mathcal{G}$

We henceforth assume that the characters are indexed consistently with this lemma.

Note that $\varepsilon_{\mathbf{p}}(\mathbf{g}) \cdot \varepsilon_{\mathbf{q}}(\mathbf{g}) = \varepsilon_{\mathbf{g}}(\mathbf{p}) \cdot \varepsilon_{\mathbf{g}}(\mathbf{q}) = \varepsilon_{\mathbf{g}}(\mathbf{pq}) = \varepsilon_{\mathbf{pq}}(\mathbf{g})$, hence $\varepsilon_{\mathbf{p}} \cdot \varepsilon_{\mathbf{q}} = \varepsilon_{\mathbf{pq}}$, and $\hat{\mathcal{G}}$ is a group isomorphic to $\mathcal{G}$ under this product.

The most common way of writing the characters is

$$\varepsilon_{\mathbf{p}}(\mathbf{q}) = e^{-2\pi i p_1 q_1/m_1} \cdot e^{-2\pi i p_2 q_2/m_2} \cdots e^{-2\pi i p_k q_k/m_k} = e^{\frac{-2\pi i}{m} \mathbf{p}^T D \mathbf{q}} \tag{4}$$

where D is the diagonal integer matrix $D = \mathrm{diag}\{m/m_1, m/m_2, \ldots, m/m_k\}$, and m is the least common multiple of $\{m_1, \ldots, m_k\}$. There are also other ways of indexing $\hat{\mathcal{G}}$ consistent with lemma 1, but they are all equivalent under an automorphism on $\hat{\mathcal{G}}$.

The most important property of the characters is that they form a perpendicular basis for the Hilbert space of functions on $\mathcal{G}$:

$$(\varepsilon_{\mathbf{p}}, \varepsilon_{\mathbf{q}}) = \begin{cases} \#(\mathcal{G}) & \text{if } \mathbf{p} = \mathbf{q} \\ 0 & \text{otherwise} \end{cases}$$

The discrete Fourier transform is an expansion of a function on $\mathcal{G}$ in this set of orthogonal functions:

Proposition 1 *A function* u *on* $\mathcal{G}$ *can be expanded in a Fourier series*

$$u(\mathbf{g}) = \sum_{\mathbf{p}\in\mathcal{G}} \hat{u}(\mathbf{p})\varepsilon_{\mathbf{p}}(\mathbf{g}) \tag{5}$$

with Fourier coefficients

$$\hat{u}(\mathbf{p}) = \frac{(u, \varepsilon_{\mathbf{p}})}{(\varepsilon_{\mathbf{p}}, \varepsilon_{\mathbf{p}})} = \frac{1}{\#(\mathcal{G})} \sum_{\mathbf{g}\in\mathcal{G}} u(\mathbf{g})\varepsilon^*_{\mathbf{p}}(\mathbf{g}) \tag{6}$$

Definition 2 *Equations 5 and 6 is our definition of the IDFT and DFT, respectively.*

To describe the FFT algorithms, we must understand the duality between subgroups of $\mathcal{G}$ and subgroups of $\hat{\mathcal{G}}$. Let $\mathcal{H} < \mathcal{G}$ and let $\mathcal{K} = \mathcal{G}/\mathcal{H}$. Let $\hat{\mathcal{K}}$ be the subset of $\hat{\mathcal{G}}$ defined by

$$\hat{\mathcal{K}} = \{ \varepsilon_{\mathbf{k}} \in \hat{\mathcal{G}} \mid \varepsilon_{\mathbf{k}}(\mathbf{h}) = 1 \text{ for all } \mathbf{h} \in \mathcal{H} \}.$$

$\hat{\mathcal{K}}$ is a subgroup of $\hat{\mathcal{G}}$. The characters in the set $\hat{\mathcal{K}}$ are constant on each coset of $\mathcal{H}$ and can be regarded as functions on $\mathcal{K}$. It can be shown that $\hat{\mathcal{K}}$ contains all the characters on $\mathcal{K}$, hence $\hat{\mathcal{K}}$ is the dual group of $\mathcal{K}$.

The restriction of a character in $\hat{\mathcal{G}}$ to $\mathcal{H}$ is a character on $\mathcal{H}$, but the characters on $\mathcal{H}$ are represented $\#(\mathcal{K})$ times in $\hat{\mathcal{G}}$. It is easy to check that every character belonging to the same coset of $\hat{\mathcal{K}}$ in $\hat{\mathcal{G}}$ acts identically on $\mathcal{H}$. We may therefore associate each coset of $\hat{\mathcal{K}}$ with a character on $\mathcal{H}$. To sum up, we have the following duality:

Lemma 2 *Let* $\mathcal{H} < \mathcal{G}$ *and* $\mathcal{K} = \mathcal{G}/\mathcal{H}$. *Then the dual groups of* $\mathcal{H}$ *and* $\mathcal{K}$ *are given by*

$$\hat{\mathcal{K}} = \{ \varepsilon_{\mathbf{k}} \in \hat{\mathcal{G}} \mid \varepsilon_{\mathbf{k}}(\mathbf{h}) = 1 \textit{ for all } \mathbf{h} \in \mathcal{H} \}$$

and

$$\hat{\mathcal{H}} = \hat{\mathcal{G}}/\hat{\mathcal{K}}$$

Example 1 *Let* $\mathcal{G} = \mathcal{Z}_6 = \{0, 1, \ldots, 5\}$ *and* $\mathcal{H} = \{0, 3\}$. *Then*

$$\mathcal{K} = \{\{0,3\},\{1,4\},\{2,5\}\}\ ,\quad \hat{\mathcal{G}} = \{\varepsilon_0, \ldots, \varepsilon_5\}\ ,\quad \hat{\mathcal{K}} = \{\varepsilon_0, \varepsilon_2, \varepsilon_4\}$$

and

$$\hat{\mathcal{H}} = \{\{\varepsilon_0, \varepsilon_2, \varepsilon_4\}, \{\varepsilon_1, \varepsilon_3, \varepsilon_5\}\}$$

Let us now split $\mathcal{G}$ in the cosets of $\mathcal{H}$ and $\hat{\mathcal{G}}$ in the cosets of $\hat{\mathcal{K}}$, i.e. let $\mathcal{Q} = \{\mathbf{q}_i\}$ and $\mathcal{R} = \{\mathbf{r}_i\}$ be subsets of $\mathcal{G}$ such that $\{\mathbf{q}_i\}$ is a collection of one element from each coset of $\mathcal{H}$ and $\{\varepsilon_{\mathbf{r}_i}\}$ is a collection of one element from each coset of $\hat{\mathcal{K}}$. Two elements $\mathbf{g}, \mathbf{p} \in \mathcal{G}$ may be written $\mathbf{g} = \mathbf{h}\mathbf{q}_i$ and $\mathbf{p} = \mathbf{k}\mathbf{r}_j$ where $\mathbf{h} \in \mathcal{H}$ and $\varepsilon_{\mathbf{k}} \in \hat{\mathcal{K}}$. By inserting this splitting in equation 6 and using $\varepsilon_{\mathbf{k}}(\mathbf{h}) = 1$ we obtain:

$$\#(\mathcal{G})\cdot\hat{u}(\mathbf{p}) = \#(\mathcal{G})\cdot\hat{u}(\mathbf{k}\mathbf{r}_j) = \sum_{\mathbf{g}\in\mathcal{G}} u(\mathbf{g})\varepsilon^*_{\mathbf{p}}(\mathbf{g}) = \sum_{\mathbf{q}_i} \hat{u}(\mathbf{q}_i, \mathbf{r}_j)\varepsilon^*_{\mathbf{r}_j}(\mathbf{q}_i)\varepsilon^*_{\mathbf{k}}(\mathbf{q}_i) \tag{7}$$

where

$$\hat{u}(\mathbf{q}_i, \mathbf{r}_j) = \sum_{\mathbf{h}\in\mathcal{H}} u(\mathbf{h}\mathbf{q}_i)\varepsilon^*_{\mathbf{r}_j}(\mathbf{h}). \tag{8}$$

Since $\varepsilon_{\mathbf{r}_j}$ is a character on $\mathcal{H}$, $\hat{u}(\mathbf{q}_i, \mathbf{r}_j)$ is the Fourier component $\mathbf{r}_j$ of the function $v = u|_{\mathcal{H}\mathbf{q}_i}$, defined by $v(\mathbf{h}) = u(\mathbf{h}\mathbf{q}_i)$. Equations 7–8 express the fact that the Fourier transform on a group $\mathcal{G}$ may be found from the Fourier transforms on each of the cosets of a subgroup $\mathcal{H} < \mathcal{G}$. E.g. if $\mathcal{K} = \mathcal{Z}_2$, the original problem splits in two similar problems of half the size. We have proven the following proposition:

Proposition 2 *Let $\mathcal{G}$ be a finite Abel group and $\mathcal{H}$ a subgroup of $\mathcal{G}$. The DFT of a function u defined on $\mathcal{G}$ can be found from the DFT's of $u|_{\mathcal{H}\mathbf{q}_i}$ for all cosets $\mathcal{H}\mathbf{q}_i$.*

The FFT algorithms are divide and conquer algorithms based on recursive applications of this splitting.

Note that the factors $\varepsilon_{\mathbf{k}}(\mathbf{q}_i)$ are the characters on the group $\mathcal{K}$. If $\mathcal{K} = \mathcal{Z}_2$ or if $\mathcal{K} = \mathcal{Z}_2 \otimes \mathcal{Z}_2$ these factors are ± 1, and multiplication by $\varepsilon^*_{\mathbf{k}}(\mathbf{q}_i)$ can be avoided.

The factors $\varepsilon_{\mathbf{r}_j}(\mathbf{q}_i)$ are called *twiddle factors*. Multiplication by a twiddle factor in the dual group corresponds to a translation of the original group, i.e. $\hat{w}(\mathbf{p}) = \varepsilon_{\mathbf{p}}(\mathbf{t})\cdot\hat{v}(\mathbf{p})$ is the DFT of the function $w(\mathbf{g}) = v(\mathbf{t}\mathbf{g})$. Obviously $\varepsilon_{\mathbf{r}_j}(\mathbf{e}) = 1$, so $\hat{u}(\mathbf{e}, \mathbf{r}_j)$ needs no 'twiddling'. In the case where $\mathcal{G} = \mathcal{H} \otimes \mathcal{K}$, we can choose both $\mathcal{Q}$ and $\mathcal{R}$ as subgroups of $\mathcal{G}$. Then $\varepsilon_{\mathbf{r}_j}(\mathbf{q}_i) \equiv 1$ and we can avoid 'twiddling' altogether. This is however generally not possible. In the one dimensional case, this 'twiddle–free' splitting is possible if and only if $\#(\mathcal{H})$ and $\#(\mathcal{K})$ have no common factors.

Example 2 (The Cooley–Tukey algorithm) *Let n be an even number, $\mathcal{G} = \mathcal{Z}_n$ and $\mathcal{H} = \langle(2)\rangle$. Then $\mathcal{K} = \mathcal{Z}_2$ and $\hat{\mathcal{K}} = \{\varepsilon_{\mathbf{k}_0}, \varepsilon_{\mathbf{k}_1}\}$, where $\mathbf{k}_0 = (0)$ and $\mathbf{k}_1 = (n/2)$. Choose $\mathcal{Q} = \{\mathbf{q}_1, \mathbf{q}_2\} = \{(0), (1)\}$ and $\mathcal{R} = \{\mathbf{r} = (r) \mid 0 \le r < n/2\}$. Equations 7–8 gives:*

$$\begin{aligned} \#(\mathcal{G})\cdot\hat{u}(\mathbf{r}) &= \hat{u}(\mathbf{q}_0, \mathbf{r}) + \hat{u}(\mathbf{q}_1, \mathbf{r})\varepsilon^*_{\mathbf{r}}(\mathbf{q}_1) \\ \#(\mathcal{G})\cdot\hat{u}(\mathbf{k}_1\mathbf{r}) &= \hat{u}(\mathbf{q}_0, \mathbf{r}) - \hat{u}(\mathbf{q}_1, \mathbf{r})\varepsilon^*_{\mathbf{r}}(\mathbf{q}_1) \end{aligned}$$

where

$$\begin{aligned} \hat{u}(\mathbf{q}_0, \mathbf{r}) &= \sum_{\mathbf{h}\in\mathcal{H}} u(\mathbf{h})\varepsilon^*_{\mathbf{r}}(\mathbf{h}) \\ \hat{u}(\mathbf{q}_1, \mathbf{r}) &= \sum_{\mathbf{h}\in\mathcal{H}} u(\mathbf{h}\mathbf{q}_1)\varepsilon^*_{\mathbf{r}}(\mathbf{h}) \end{aligned}$$

Hence the original transform is computed by two transforms of half the size. The celebrated Cooley–Tukey algorithm [2] is a recursive application of these equations. If $n = 2^k$ the total cost of computing a complex FFT by this algorithm is

$$5n \log_2 n - 6n$$

real floating point operations (multiplications and additions are counted separately).

The common way of computing 2–dimensional Fourier transforms is to first apply the Cooley–Tukey algorithm on each 'row' of the data, and afterwards on each 'column'. (Or in our language by doing the reductions first in a sequence of subgroups of the type $\mathcal{H}_i = \langle (h_i, 0), (0, 1) \rangle$, and afterwards in subgroups $\mathcal{H}_j = \langle (0, h_j) \rangle$). Proposition 2 shows however that other reductions are possible:

Example 3 (Radix 2⊗2 reduction)
Let m and n be even numbers, $\mathcal{G} = \mathcal{Z}_m \otimes \mathcal{Z}_n$ and $\mathcal{H} = \langle (2,0), (0,2) \rangle$. Then $\mathcal{K} = \mathcal{Z}_2 \otimes \mathcal{Z}_2$ and

$$\hat{\mathcal{K}} = \{\varepsilon_{\mathbf{k}_0}, \varepsilon_{\mathbf{k}_1}, \varepsilon_{\mathbf{k}_2}, \varepsilon_{\mathbf{k}_3}\}$$

where $\mathbf{k}_0 = (0,0)$, $\mathbf{k}_1 = (m/2, 0)$, $\mathbf{k}_2 = (0, n/2)$ and $\mathbf{k}_3 = (m/2, n/2)$. Choose

$$\mathcal{Q} = \{\mathbf{q}_0, \mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3\} = \{(0,0), (1,0), (0,1), (1,1)\}$$

and

$$\mathcal{R} = \{\, \mathbf{r} = (r_1, r_2) \mid 0 \le r_1 < m/2, \quad 0 \le r_2 < n/2 \,\}.$$

Equations 7–8 gives:

$$\begin{aligned}
\#(\mathcal{G}) \cdot \hat{\mathsf{u}}(\mathbf{r}) &= \mathsf{v}_0(\mathbf{r}) + \mathsf{v}_1(\mathbf{r}) \\
\#(\mathcal{G}) \cdot \hat{\mathsf{u}}(\mathbf{r}\mathbf{k}_1) &= \mathsf{w}_0(\mathbf{r}) + \mathsf{w}_1(\mathbf{r}) \\
\#(\mathcal{G}) \cdot \hat{\mathsf{u}}(\mathbf{r}\mathbf{k}_2) &= \mathsf{v}_0(\mathbf{r}) - \mathsf{v}_1(\mathbf{r}) \\
\#(\mathcal{G}) \cdot \hat{\mathsf{u}}(\mathbf{r}\mathbf{k}_3) &= \mathsf{w}_0(\mathbf{r}) - \mathsf{w}_1(\mathbf{r})
\end{aligned}$$

where

$$\begin{aligned}
\mathsf{v}_0(\mathbf{r}) &= \hat{\mathsf{u}}(\mathbf{q}_0, \mathbf{r}) + \hat{\mathsf{u}}(\mathbf{q}_1, \mathbf{r}) \varepsilon^*_{\mathbf{r}}(\mathbf{q}_1) \\
\mathsf{w}_0(\mathbf{r}) &= \hat{\mathsf{u}}(\mathbf{q}_0, \mathbf{r}) - \hat{\mathsf{u}}(\mathbf{q}_1, \mathbf{r}) \varepsilon^*_{\mathbf{r}}(\mathbf{q}_1) \\
\mathsf{v}_1(\mathbf{r}) &= \hat{\mathsf{u}}(\mathbf{q}_2, \mathbf{r}) \varepsilon^*_{\mathbf{r}}(\mathbf{q}_2) + \hat{\mathsf{u}}(\mathbf{q}_3, \mathbf{r}) \varepsilon^*_{\mathbf{r}}(\mathbf{q}_3) \\
\mathsf{w}_1(\mathbf{r}) &= \hat{\mathsf{u}}(\mathbf{q}_2, \mathbf{r}) \varepsilon^*_{\mathbf{r}}(\mathbf{q}_2) - \hat{\mathsf{u}}(\mathbf{q}_3, \mathbf{r}) \varepsilon^*_{\mathbf{r}}(\mathbf{q}_3)
\end{aligned}$$

and

$$\hat{\mathsf{u}}(\mathbf{q}_i, \mathbf{r}) = \sum_{\mathbf{h} \in \mathcal{H}} \mathsf{u}(\mathbf{h}\mathbf{q}_i) \varepsilon^*_{\mathbf{r}}(\mathbf{h})$$

The $m \times n$ problem is reduced to four similar problems of size $m/2 \times n/2$. If $m = n = 2^k$ the total cost of computing a complex $m \times n$ FFT by recursively applying this splitting is

$$\frac{17}{2} n^2 \log_2 n - 9n^2$$

real floating point operations. It is interesting to note that the standard method of doing first the Cooley–Tukey reductions in one direction and afterwards in the other direction costs

$$10n^2 \log_2 n - 12n^2$$

real floating point operations. I.e. for moderately large n the $2\otimes 2$ reduction requires 15% less flops, and it is cheaper already for $n = 8$. This saving is similar to the savings obtained by using radix 4 transforms rather than radix 2 transforms in the 1–D case, see [12] for details. In 3–D, the radix $2\otimes 2\otimes 2$ transform is 25% cheaper than the composition of radix 2 transforms. Not only is the operation count lower, but the number of memory references in the $2\otimes 2$ transform is only about the half, and in the $2\otimes 2\otimes 2$ transform a third. Our main reason for showing these multidimensional reductions here is however that it is necessary to do this kind of reductions to take benefit of all the symmetries of certain symmetric multidimensional functions.

3 The periodic symmetry groups

In this section we are formalizing the concept of symmetric functions. We define a general class of symmetries which is the basis for our symmetric FFT's.

Let u be a complex valued function

$$u : \mathbf{R}^n \to \mathbf{C}$$

Let ω and θ be isometries, i.e. distance preserving mappings:

$$\begin{aligned} \omega &: \mathbf{R}^n \to \mathbf{R}^n \\ \theta &: \mathbf{C} \to \mathbf{C} \end{aligned}$$

A *symmetry* of u is a pair (ω, θ) such that

$$u \circ \omega = \theta \circ u$$

The set of all symmetries of u forms a group $\mathcal{S}$ called the *symmetry group* of u, where the group product is defined by

$$(\omega_1, \theta_1) \cdot (\omega_2, \theta_2) = (\omega_1 \circ \omega_2, \theta_2 \circ \theta_1)$$

We will see that under quite general assumptions, a lot can be said about the structure of $\mathcal{S}$. Since we want u to be expandable in a Fourier series, we assume that u is periodic in n different directions. To exclude continuous symmetry groups from the discussion, we also assume that in each direction there is a shortest nonzero translation symmetry.

Assumption 1

1. *There exists a set of translations* $T = \{\tau_i\}_{i=1}^n$ *in* n *linearly independent directions such that*

   $$(\tau_i, \mathrm{e}) \in \mathcal{S} \text{ for all } i$$

   where e *is the identity isometry on* $\mathbf{C}$. T *is called the* periodic translations *of* u

2. *Given a translation τ such that $(\tau, \cdot) \in \mathcal{S}$, there exists a shortest nonzero translation τ' such that $\tau' || \tau$ and $(\tau', \cdot) \in \mathcal{S}$.*

To understand the structure of $\mathcal{S}$ it is useful first to study how $\mathcal{S}$ acts on $\mathbf{R}^n$. Let $\mathcal{W} = \{ \omega \mid (\omega, \cdot) \in \mathcal{S} \}$. $\mathcal{W}$ is a group of isometries acting on $\mathbf{R}^n$, containing minimal nonzero translations in n linearly independent directions. This is called a n–dimensional *crystallography* group. For a given n there are only a finite number of crystallography groups. For $n = 1, 2, 3$ and 4 there are exactly $2, 17, 230$ and 4783 different crystallography groups [5]. The number for $n = 5$ is unknown. The following proposition restricts the possible rotations and reflections in $\mathcal{W}$.

Proposition 3 (The crystallographic restriction)
Given a n–dimensional crystallography group $\mathcal{W}$. If $\rho \in \mathcal{W}$ is a rotation of $\mathbf{R}^n$ through a nonzero angle ϕ, then

$$\phi \in \{2\pi/6, 2\pi/4, 2\pi/3, 2\pi/2\}$$

Proof of the 2–dimensional case is given in [7]. The general case follows by a similar argument. Since the composition of two reflections about non–parallel planes is a rotation through two times the angle between the planes, we get:

Corollary 1 *If $\sigma_1, \sigma_2 \in \mathcal{W}$ are reflections about non–parallel $n-1$ dimensional hyperplanes, then the angle ϕ between the planes must satisfy:*

$$\phi \in \{2\pi/12, 2\pi/8, 2\pi/6, 2\pi/4\}$$

This corollary restricts the geometry of the possible domains where the Poisson equation can be solved by the techniques in this paper. We will now see how $\mathcal{S}$ acts on $\mathbf{C}$. Let $\Theta = \{ \theta \mid (\cdot, \theta) \in \mathcal{S} \}$.

Lemma 3 *If u is a non–constant function, then Θ is a finite group.*

Proof: If $(\mathrm{e}, \theta) \in \mathcal{S}$, where e is the identity isometry on $\mathbf{R}^n$, then either θ is the identity, or $\theta = \sigma_l$, where σ_l is reflection of the complex plane about a line l. All other isometries of $\mathcal{C}$ fixes at most one point, and this leads to a constant or ambiguous u. (E.g. if $u(r) = -u(r)$ then $u \equiv 0$). Now let $\mathcal{P}$ be the parallelepiped of $\mathcal{R}^n$ spanned by the periodic translations $\{\tau_i\}_{i=1}^n$. $\mathcal{W}$ is a discrete group, and there are therefore only a finite number of isometries in $\mathcal{W}$ sending points in $\mathcal{P}$ to other points in $\mathcal{P}$. By the same argument as above, each of these isometries can at most be associated with two different elements of Θ. Due to the periodicity of u, the remaining isometries in $\mathcal{W}$ cannot be associated with any new elements in Θ.

□

According to Leonardi's theorem [7], the only finite two–dimensional isometry groups are the cyclic groups $\mathcal{C}_n$ and the dihedral groups $\mathcal{D}_n$. $\mathcal{C}_n$, $n \in \{1, 2, 3, \ldots\}$ are the groups generated by a rotation $2\pi/n$ about a point p. $\mathcal{D}_n$, $n \in \{1, 2, 3, \ldots\}$ are the groups generated by a rotation $2\pi/n$ about a point p and a reflection about a line passing through the same point. After a linear transformation

$$u(z) \rightarrow (u(z) - p) \cdot e^{i\phi}$$

we may assume that the symmetry point is the origin, and the reflection axis is the real axis. Hence we may without loss of generality assume:

Assumption 2 *The group Θ is either generated by*

$$\Theta = \langle \theta_1 \rangle$$

or by

$$\Theta = \langle \theta_1, \theta_2 \rangle$$

where $\theta_1(z) = z \cdot e^{2\pi i/k}$ for some integer k, and $\theta_2(z) = z^*$.

Definition 3 *A symmetry group satisfying assumptions 1-2 is denoted a* periodic symmetry group.

By the *isometries* in a periodic symmetry group, $\mathcal{S} = (\mathcal{W}, \Theta)$,we henceforth mean the elements of $\mathcal{W}$.

4 Symmetric FFT's

In this section we will show that it is in principle possible to construct symmetric variants of the FFT algorithm for any function with a periodic symmetry group. The computational savings compared to the complex FFT of the same size is proportional to the number of symmetries in the group $\mathcal{S}/\mathcal{T}$, where $\mathcal{T}$ is the set of periodic translations in assumption 1. The symmetric algorithms can be summarized in the following steps:

1. Identify the symmetry group $\mathcal{S}$. Discretize the domain with a uniform grid such that every symmetry in $\mathcal{S}$ also carries over to the discretized function. This gives a function u on a finite Abel group $\mathcal{G}$.

2. Find a sequence of subgroups $\mathcal{H}^1 < \mathcal{H}^2 < \cdots < H^i = \mathcal{G}$ such that every symmetry of u also is a symmetry of $u|_{\mathcal{H}^j}$. Use all the symmetries of $\mathcal{S}$ in the divide phase of the FFT.

3. Lift every symmetry in $\mathcal{S}$ to a dual symmetry group $\hat{\mathcal{S}}$ on the Fourier transform $\hat{u}$. Use all the dual symmetries in the conquer phase of the algorithm.

Let us first define a uniform grid and show how to construct the groups $\mathcal{G}$ and the subgroups $\mathcal{H}^j$.

Definition 4 *A uniform grid $\mathcal{G}$, is the image of a point $x \in \mathbf{R}^n$ under the action of a discrete group of translations $\mathcal{T}$, i.e. $\mathcal{G} = \mathcal{T}(x)$.*

Lemma 4 *For every crystallography group $\mathcal{W}$, there exists a uniform grid $\mathcal{G}$ such that $\mathcal{W}\mathcal{G} = \mathcal{G}$.*

Proof: Let $\mathcal{T} < \mathcal{W}$ be the (normal) subgroup consisting of all the translations in $\mathcal{W}$. Let $\tilde{\mathcal{W}} = \mathcal{W}/\mathcal{T}$. Think of $\tilde{\mathcal{W}}$ as a group of isometries acting on $\mathcal{R}^n$, where points in $\mathcal{R}^n$ equivalent under translations in $\mathcal{T}$ are identified. The domain of $\tilde{\mathcal{W}}$ is a parallelepiped $\mathcal{P} \subset \mathbf{R}^n$ spanned by the shortest nonzero translations in n different directions. Since $\mathcal{W}$ is discrete and $\mathcal{P}$ has finite volume, $\tilde{\mathcal{W}}$ is a finite group.

Every $\tilde{w}_i \in \tilde{\mathcal{W}}$ can be written as a linear map

$$\tilde{w}_i(x) = A_i x + g_i$$

where A_i is an orthogonal matrix. Since $\tilde{\mathcal{W}}$ is finite, there exists for every $\tilde{w}_i \in \tilde{\mathcal{W}}$ a smallest positive integer k_i such that

$$\tilde{w}_i^{k_i}(x) = x + t_i$$

where $x \mapsto t_i$ is a translation in $\mathcal{T}$. (Hence $A_i^{k_i} = I$, and the eigenvalues of A_i are the k_i'th roots of 1).

Now let x be an arbitrary point in $\mathbf{R}^n$ and consider the 'centeroid' of the orbit of x under the action of $\tilde{w}_i$

$$c = k_i^{-1} \sum_{j=0}^{k_i-1} \tilde{w}_i^j(x)$$

A calculation gives

$$\begin{aligned} \tilde{w}_i(c) &= k_i^{-1}\Big(\sum_{j=0}^{k_i-1} A_i \cdot \tilde{w}_i^j(x) + g_i\Big) = k_i^{-1} \sum_{j=1}^{k_i} \tilde{w}_i^j(x) = k_i^{-1}\Big(\sum_{j=0}^{k_i-1} \tilde{w}_i^j(x) + t_i\Big) \\ &= c + k_i^{-1} t_i \end{aligned}$$

It can also be shown that $k_i^{-1} t_i$ is the orthogonal projection of g_i onto the eigenspace of A_i corresponding to the eigenvalue $\lambda = 1$.

By a similar calculation we see that if we let c be the centeroid of e.g. the point $x = 0$ under the action of $\tilde{\mathcal{W}}$

$$c = \#(\tilde{\mathcal{W}})^{-1} \sum_{\tilde{w} \in \tilde{\mathcal{W}}} \tilde{w}(0) = \#(\tilde{\mathcal{W}})^{-1} \sum_{i=1}^{\tilde{\mathcal{W}}} g_i$$

then for all $\tilde{w}_i \in \tilde{\mathcal{W}}$

$$\tilde{w}_i(c) = c + k_i^{-1} t_i = \tau_i'(c),$$

where $\tau_i'(x) = x + k_i^{-1} t_i$.

Let $\mathcal{T}'$ be the group of translations

$$\mathcal{T}' = \langle \tau_i' \rangle_{i=1}^{\#(\tilde{\mathcal{W}})}$$

We have shown that the grid $\mathcal{G} = \mathcal{T}'(c)$ is preserved under the action of $\tilde{\mathcal{W}}$. Since $\mathcal{T} < \mathcal{T}'$ we also find that $\mathcal{W}\mathcal{G} = \mathcal{G}$.

□

Proposition 4 *There exists a family of finer and finer grids $\{\mathcal{G}^k\}_{k=1}^{\infty}$ such that $\mathcal{W}\mathcal{G}^k = \mathcal{G}^k$ and $\mathcal{G}^l \subset \mathcal{G}^k$ if l divides k.*

Proof: Let $\mathcal{G}^1 = \mathcal{T}'(c)$ be the grid of lemma 4, where $\mathcal{T}' = \langle \tau_i' \rangle$ and $\tau_i'(x) = x + t_i'$. For every integer k let $\mathcal{G}^k = \mathcal{T}^k(c)$, where $\mathcal{T}^k = \langle \tau_i^{1/k} \rangle$ and $\tau_i^{1/k}(x) = x + k^{-1} t_i'$. If l divides k then $\mathcal{T}^l \subset \mathcal{T}^k$, so $\mathcal{G}^l \subset \mathcal{G}^k$. The fact that $\mathcal{W}\mathcal{G}^k = \mathcal{G}^k$ follows from lemma 4 and the linearity of the maps in $\mathcal{W}$:

$$w_j(\mathsf{c}+k^{-1}\mathsf{t}_i') = w_j(k^{-1}\mathsf{c}+(k-1)k^{-1}(\mathsf{c}+\mathsf{t}_i')) = k^{-1}w_j(\mathsf{c})+(k-1)k^{-1}w_j(\mathsf{c}+\mathsf{t}_i') =$$

$$\mathsf{c}+k^{-1}\mathsf{t}_j'+(k-1)k^{-1}\mathsf{t}_{ji}' = (\tau_j^{1/k}\cdot\tau_{ji}^{(k-1)/k})(\mathsf{c})$$

where $\mathsf{t}_{ji}' = w_j w_i(\mathsf{c}) - \mathsf{c}$.

□

The grids $\{\mathcal{G}^k\}_{k=1}^{\infty}$ are however not the only uniform grids that are invariant under $\mathcal{W}$. For example if $\mathcal{W}$ is a two–dimensional crystallography group generated by two translations and a rotation through an angle $2\pi/4$, then the construction above will place a gridpoint on the point of rotation. Another possibility is that the point of rotation is midway between four gridpoints. We conjecture that every uniform invariant grid is a subset of one of the grids in the family $\{\mathcal{G}^k\}_{k=1}^{\infty}$, although this is not proven.

Let $u : \mathbf{R}^n \rightarrow \mathbf{C}$ be a function with a periodic symmetry group $\mathcal{S} = (\mathcal{W}, \Theta)$. To discretize u we choose the invariant grid $\mathcal{G}^k$ for some integer $k = 2^i$ (or more generally we may choose an integer that is factorizable in the product of small primes). The restriction of the continuous function u to the grid, $\mathsf{u} = u|_{\mathcal{G}^k}$ can be regarded as a function on a finite Abel group $\mathcal{G} = \mathcal{Z}_{m_1}\otimes\mathcal{Z}_{m_2}\otimes\cdots\otimes\mathcal{Z}_{m_n}$, where the periods m_i are the number of gridpoints within the periodic translation of u in each direction. We let the identity node e of $\mathcal{G}$ correspond to the gridpoint c. Then the sequence of finer grids $\mathcal{G}^1, \mathcal{G}^2, \ldots, \mathcal{G}^{2^{i-1}}, \mathcal{G}^{2^i}$ corresponds to a family of subgroups $\{\mathcal{H}^j\}_{j=1}^{i}$, nested within each other as:

$$\mathcal{H}^1 < \mathcal{H}^2 < \cdots \mathcal{H}^{i-1} < \mathcal{H}^i = \mathcal{G}$$

where

$$w\mathcal{H}^j = \mathcal{H}^j \quad \text{for all } w \in \mathcal{W}$$

and

$$\mathcal{K}^j = \mathcal{H}^j/\mathcal{H}^{j-1} = \mathcal{Z}_2\otimes\mathcal{Z}_2\otimes\cdots\otimes\mathcal{Z}_2 \ .$$

To every symmetry $(w,\theta) \in \mathcal{S}$ of u, there must correspond a symmetry (w',θ) of the discretized u, where w' is an isometry on the group $\mathcal{G}$. We may write $w'(\mathsf{g}) = \alpha(\mathsf{gt})$, where $\mathsf{t} \in \mathcal{G}$ and α is an automorphism on $\mathcal{G}$. Since the periodic symmetries (τ, e) are already taken into account by regarding u as a function on a cyclic group, we obtain:

Lemma 5 *The discretized function* $\mathsf{u} = u|_{\mathcal{G}}$ *has a finite group of symmetries* $\mathcal{S}' = \mathcal{S}/\mathcal{T}$, *where* $\mathcal{T}$ *is the periodic translations of* u. *The restrictions* $\mathsf{u}|_{\mathcal{H}^j}$ *possess the same symmetries as* u.

The family of subgroups $\{\mathcal{H}^j\}_{j=1}^{i}$ is the basis for our symmetric FFT's. The algorithms consists of two phases: a divide phase, where u is split in the cosets of smaller and smaller subgroups $\mathcal{H}^j, \mathcal{H}^{j-1}, \ldots$ and a conquer phase where equations 7–8 are used to compute the FFT on a higher level.

In each step of the divide phase, each of the functions $\{\mathsf{u}|_{\mathcal{H}^j\mathbf{q}_l}\}$ are split in $\#(\mathcal{K}^j)$ smaller functions of the type $\mathsf{u}|_{\mathcal{H}^{j-1}\mathbf{q}_i}$. Although $\mathsf{u}|_{\mathcal{H}^j}$ retains all the symmetries of u, this is generally not the case for the restriction to other cosets $\mathcal{H}^j\mathbf{q}$:

Lemma 6 *Given a coset* $\mathcal{H}^j\mathbf{q}$ *and an isometry* $w \in \mathcal{W}$. *Then* w *is either an isometry on* $\mathcal{H}^j\mathbf{q}$ *or a bijective mapping of* $\mathcal{H}^j\mathbf{q}$ *onto another coset* $\mathcal{H}^j\mathbf{q}'$.

Proof: $w(\mathcal{H}^j\mathbf{q}) = (w\mathcal{H}^j)\cdot(w\mathbf{q}) = \mathcal{H}^j w\mathbf{g} = \mathcal{H}^j\mathbf{g}'$. If $\mathbf{g}' = \mathbf{g}\mathbf{h}'$ for some $\mathbf{h}' \in \mathcal{H}^j$, then w is an isometry on $\mathcal{H}^j\mathbf{g}$. Otherwise w maps $\mathcal{H}^j\mathbf{g}$ onto another coset $\mathcal{H}^j\mathbf{g}'$.

□

Proposition 5 *Let* $\mathsf{v} = \mathsf{u}|_{\mathcal{H}^j\mathbf{q}}$ *be defined by* $\mathsf{v}(\mathbf{h}) = \mathsf{u}(\mathbf{hq})$. *If* u *has a symmetry group* $\mathcal{S} = (\mathcal{W}, \Theta)$, *then* v *has a symmetry group* $\mathcal{S}' = (\mathcal{W}', \Theta')$, *where*

$$\mathcal{W}' = \{\, w' \mid w'(\mathbf{h}) = \mathbf{q}w(\mathbf{h}\mathbf{q}^{-1}) \;\; \textit{for all} \;\; w \in \mathcal{W} \;\; \textit{s.t.} \;\; w\mathcal{H}^j\mathbf{q} = \mathcal{H}^j\mathbf{q} \,\}$$

and

$$\Theta' = \{\, \theta' \mid (w, \theta') \in \mathcal{S} \;\; \textit{for some} \;\; w \in \mathcal{W} \;\; \textit{s.t.} \;\; w\mathcal{H}^j\mathbf{q} = \mathcal{H}^j\mathbf{q} \,\}.$$

$\mathcal{W}'$ *is (isomorphic to) a subgroup of* $\mathcal{W}$, *and* $\Theta' < \Theta$

Proof: Check that

$$(\mathsf{v} \circ w')(\mathbf{h}) = (\mathsf{u} \circ w)(\mathbf{hq}) = (\Theta \circ \mathsf{u})(\mathbf{hq}) = (\Theta \circ \mathsf{v})(\mathbf{h})$$

□

So $\mathsf{u}|_{\mathcal{H}^j\mathbf{q}_i}$ has in general fewer symmetries than u itself, but the 'lost' symmetries can be taken care of by processing only cosets that are unrelated by isometries in $\mathcal{W}$.

Given a function u with a symmetry group $\mathcal{S}$. The *fundamental domain* of $\mathcal{S}$ is a minimal set of points so that the function u can be reconstructed from its values in the fundamental domain, by using the symmetries in $\mathcal{S}$.

The divide phase of the symmetric FFT is summarized by:

1. At every step, split each function in $\{\mathsf{u}|_{\mathcal{H}^j\mathbf{q}}\}$ in $\#(\mathcal{K}^j)$ functions of the type $\mathsf{u}|_{\mathcal{H}^{j-1}\mathbf{q}_i}$.
2. For each coset $\mathcal{H}^j\mathbf{q}$, store and process only $\mathsf{u}|_{\mathcal{H}^j\mathbf{q}}$ in a fundamental domain of the symmetry group of the function.
3. Whenever several cosets $\mathcal{H}^j\mathbf{q}_{i_1}, \ldots, \mathcal{H}^j\mathbf{q}_{i_k}$ are related by isometries in $\mathcal{W}$, store and process u only in the fundamental domain of one of these cosets.

To take benefit of the symmetries in the conquer phase of the algorithm, we must show that the symmetries of u can be lifted to symmetries on $\hat{\mathsf{u}}$. First we need the following result:

Lemma 7 *To every automorphism* α *on* $\mathcal{G}$, *there corresponds an automorphism* $\hat{\alpha}$ *on* $\mathcal{G}$ *such that*

$$\varepsilon_{\mathbf{p}}(\alpha\mathbf{g}) = \varepsilon_{\hat{\alpha}\mathbf{p}}(\mathbf{g}) \;\; \textit{for all} \;\; \mathbf{p}, \mathbf{g} \in \mathcal{G}.$$

$\hat{\alpha}$ *is called the* dual *automorphism of* α.

Proof: Define the function $\varepsilon_{\mathbf{p}'}$ on $\mathcal{G}$ by $\varepsilon_{\mathbf{p}'}(\mathbf{g}) = \varepsilon_{\mathbf{p}}(\alpha\mathbf{g})$. From the definition of characters it immediately follows that $\varepsilon_{\mathbf{p}'}$ is a character on $\mathcal{G}$. Let $\hat{\alpha}$ be the map defined by $\hat{\alpha}(\mathbf{p}) = \mathbf{p}'$. It is also easily verified that $\hat{\alpha}$ defines an automorphism on $\mathcal{G}$.

□

The dual automorphism is calculated by:

Lemma 8 *Let $\varepsilon_{\mathbf{p}}(\mathbf{q})$ be defined as in equation 4. Then $\hat{\alpha} = D^{-1}\alpha^T D$.*

From assumption 2, the elements of Θ are either rotations of $\mathbf{C}$ about the origin or reflections about a line passing through the origin. These cases must be treated separately when the symmetries are lifted:

Proposition 6 *Assume that $\theta \mathsf{o} \mathsf{u} = \mathsf{u} \mathsf{o} w$, where $w(\mathbf{g}) = \alpha(\mathbf{gt})$. Let $\hat{\alpha}$ be the dual automorphism of α.*

1. *If $\theta(z) = e^{i\phi}z$ then $\hat{\mathsf{u}}(\mathbf{p}) = \theta \mathsf{o}(\varepsilon^*_{\hat{\alpha}\mathbf{p}}(\mathbf{t}) \cdot \hat{\mathsf{u}}(\hat{\alpha}\mathbf{p}))$*
2. *If $\theta(z) = e^{i\phi}z^*$ then $\hat{\mathsf{u}}(\mathbf{p}^{-1}) = \theta \mathsf{o}(\varepsilon^*_{\hat{\alpha}\mathbf{p}}(\mathbf{t}) \cdot \hat{\mathsf{u}}(\hat{\alpha}\mathbf{p}))$*

Proof:

1. We have $\theta \mathsf{o}(\mathsf{u}, \varepsilon_{\mathbf{p}}) = (\theta \mathsf{o} \mathsf{u}, \varepsilon_{\mathbf{p}})$, so

$$\begin{aligned} \hat{\mathsf{u}}(\mathbf{p}) &= \theta \mathsf{o}(\theta^{-1} \mathsf{o} \mathsf{u}, \varepsilon_p) = \theta \mathsf{o}(\mathsf{u} \mathsf{o} w^{-1}, \varepsilon_{\mathbf{p}}) \\ &= \theta \mathsf{o}(\mathsf{u}, \varepsilon_{\mathbf{p}} \mathsf{o} w) = \theta \mathsf{o}((\mathsf{u}, \varepsilon_{\hat{\alpha}\mathbf{p}}) \cdot \varepsilon^*_{\hat{\alpha}\mathbf{p}}(\mathbf{t})) = \theta \mathsf{o}(\varepsilon^*_{\hat{\alpha}\mathbf{p}}(\mathbf{t}) \cdot \hat{\mathsf{u}}(\hat{\alpha}\mathbf{p})) \end{aligned}$$

2. Here $\theta \mathsf{o}(\mathsf{u}, \varepsilon_{\mathbf{p}}) = (\theta \mathsf{o} \mathsf{u}, \varepsilon_{\mathbf{p}^{-1}})$, and the result follows by the same calculation.

□

We see that the dual symmetries does not always fit into the definition of symmetries given in section 3. The important fact for the symmetric FFT's is however that the dual symmetries are isometric transformations of the domain, coupled with a pointwise transformation of the function. The set of dual symmetries $\hat{\mathcal{S}}$ can be represented by triplets $(\tilde{\alpha}, \mathbf{t}, \theta)$ representing relations

$$\hat{\mathsf{u}}(\mathbf{p}) = \theta \mathsf{o}(\varepsilon^*_{\tilde{\alpha}\mathbf{p}}(\mathbf{t}) \cdot \hat{\mathsf{u}}(\tilde{\alpha}\mathbf{p}))$$

where

$$\tilde{\alpha}\mathbf{p} = \left\{ \begin{array}{ll} \hat{\alpha}\mathbf{p} & \text{if } \theta(z) = e^{i\phi}z \\ \hat{\alpha}\mathbf{p}^{-1} & \text{if } \theta(z) = e^{i\phi}z^* \end{array} \right\}$$

Since there is a 1–1 correspondence between $\mathcal{S}$ and $\hat{\mathcal{S}}$ we may define a group product on $\hat{\mathcal{S}}$ via the product on $\mathcal{S}$, so that the two groups are isomorphic. Their geometric properties, such as fundamental domains, may however be different.

In equations 7–8 $\hat{\mathsf{u}}$ is computed from the smaller parts

$$\hat{\mathsf{u}}(\mathbf{q}_i, \mathbf{r}) = \sum_{\mathbf{h} \in \mathcal{H}} \mathsf{u}(\mathbf{h}\mathbf{q}_i) \varepsilon^*_{\mathbf{r}}(\mathbf{h})$$

The symmetries of each of these parts appear most clearly after each term is multiplied by the corresponding twiddle factor $\varepsilon^*_{\mathbf{p}}(\mathbf{q}_i)$:

Proposition 7 *Let*

$$\hat{\mathsf{v}}_{\mathbf{q}_i}(\mathbf{r}) = \hat{\mathsf{u}}(\mathbf{q}_i, \mathbf{r}) \cdot \varepsilon^*_{\mathbf{p}}(\mathbf{q}_i) = \sum_{\mathbf{h} \in \mathcal{H}} \mathsf{u}(\mathbf{h}\mathbf{q}_i) \varepsilon^*_{\mathbf{r}}(\mathbf{h}\mathbf{q}_i)$$

Let $(w, \theta) \in \mathcal{S}$, where $w(\mathbf{g}) = \alpha(\mathbf{gt})$, and let $\hat{\alpha}$ be the dual isometry of α.

1. *If* $\theta(z) = e^{i\phi}z$ *then* $\hat{\mathrm{v}}_{\mathbf{q}_i}(\mathbf{p}) = \theta \mathrm{o}(\varepsilon^*_{\hat{\alpha}\mathbf{p}}(\mathbf{t}) \cdot \hat{\mathrm{v}}_{\mathbf{q}_j}(\hat{\alpha}\mathbf{p}))$
2. *If* $\theta(z) = e^{i\phi}z^*$ *then* $\hat{\mathrm{v}}_{\mathbf{q}_i}(\mathbf{p}^{-1}) = \theta \mathrm{o}(\varepsilon^*_{\hat{\alpha}\mathbf{p}}(\mathbf{t}) \cdot \hat{\mathrm{v}}_{\mathbf{q}_j}(\hat{\alpha}\mathbf{p}))$

where $\alpha(\mathbf{q}_j) = \mathbf{h}\mathbf{q}_i$ *for some* $\mathbf{h} \in \mathcal{H}$.

Proof: The proof of this lemma is similar to the proof of lemma 6. We omit the details.

□

The lemma states that a symmetry of $\hat{\mathrm{u}}$ is either a symmetry of the twiddled part $\hat{\mathrm{v}}_{\mathbf{q}_i}$, or a 1–1 mapping of $\hat{\mathrm{v}}_{\mathbf{q}_i}$ onto another part $\hat{\mathrm{v}}_{\mathbf{q}_j}$. Hence the parts that are 'missing' in the conquer phase (corresponding to the cosets that were rejected in the divide phase because of symmetries) may be recovered by using this lemma.

The conquer phase of the symmetric FFT is summarized by:

1. At each level compute $\hat{\mathrm{u}}$ only in a fundamental domain of the symmetry group of the function at that level.
2. Whenever a 'missing' part $\hat{\mathrm{v}}_{\mathbf{q}_j}$ is needed, or when an element of $\hat{\mathrm{v}}_{\mathbf{q}_i}$ outside its fundamental domain is needed, use proposition 7 to recover the elements.

The discussion in this section may be summarized in the following main theorem of this paper:

Theorem 1 *Given a function* $u : \mathbf{R}^n \to \mathbf{C}$ *with a periodic symmetry group* $\mathcal{S}$ *containing the periodic translations* $\mathcal{T}$. *Then there exists a discretization of* u:

$$\mathrm{u} = u|_{\mathcal{G}}$$

with a symmetry group

$$\mathcal{S}' = \mathcal{S}/\mathcal{T}$$

such that $\hat{\mathrm{u}}$ *can be computed by a symmetric FFT; where all the symmetries in* $\mathcal{S}'$ *are used to economize the computations.*

In the remaining parts of this paper, the general approach of this section is shown in more detail for two–dimensional symmetries.

5 The 2-D Poisson symmetries

In this section we are discussing the possible periodic symmetry groups that arise in connection with two–dimensional Poisson problems. The symmetries of interest in boundary value problems are:

$$\begin{array}{llcll} (i) & u & = & u^* & \text{(real valuedness)} \\ (ii) & u\mathrm{o}\sigma_l & = & -u & \text{(Dirichlet boundary)} \\ (iii) & u\mathrm{o}\sigma_m & = & u & \text{(Neumann boundary)} \\ (iv) & u\mathrm{o}\tau & = & u & \text{(periodic boundary)} \end{array} \tag{9}$$

where σ_l and σ_m are reflections of the plane $\mathbf{R}^2$ about lines l and m, and τ is a translation of $\mathbf{R}^2$. Symmetries, such as $\mathrm{u}\mathrm{o}\rho = e^{i\phi} \cdot \mathrm{u}$ where ρ is a rotation, are not of direct interest in

boundary value problems (as far as we are aware of). Other type boundary conditions, such as Robbins conditions cannot be treated directly by the methods in this paper, although they can sometimes be handled by using a combination of several different symmetric FFT's.

Before we discuss the 2-D case in detail, let us briefly review the 1–dimensional case. There are two different 1–dimensional crystallography groups. One of them is generated by a single translation. The other is generated by two reflections σ_a and σ_b about two different points a and b. This leaves us with four possible symmetry groups due to boundary conditions:

$$\begin{array}{llllllll} 1) & u\circ\tau & = & u & & & & \text{(periodic boundary)} \\ 2) & u\circ\sigma_a & = & -u & ; \quad u\circ\sigma_b & = & -u & \text{(Dirichlet–Dirichlet)} \\ 3) & u\circ\sigma_a & = & -u & ; \quad u\circ\sigma_b & = & u & \text{(Dirichlet–Neumann)} \\ 4) & u\circ\sigma_a & = & u & ; \quad u\circ\sigma_b x & = & u & \text{(Neumann–Neumann)} \end{array} \tag{10}$$

In addition there may in each case be a real valuedness symmetry:

$$u = u^*$$

This leaves us with eight possible symmetry groups. To derive a fast FFT, we must first discretize the domain with a uniform grid that is invariant under the action of the crystallography group. For the cases 2–4 this leaves us with four different grids; each of the points a and b must be placed either on a node of the grid or midway between two nodes. All these grids are subsets of one of the grids in the family $\{\mathcal{G}^k\}_{k=1}^{\infty}$ constructed in proposition 4, and symmetric FFT's may be constructed by the general approach of the previous section. Symmetric transforms for most of these 1–D symmetries are found elsewhere in the literature, see [11] and the references therein.

In the 2–dimensional case there are seventeen different crystallography groups. For a beautiful description of these, see [7]. These groups are known from the periodic drawings of M.C. Escher [8]. The groups of interest to us are those that can be generated by reflections and translations alone. This leaves us with the nine different groups shown in table 1. The remaining eight groups requires rotations or glide–reflections as generators. These eight represents 'strange' type boundary conditions, such as periodicity between adjacent sides, or a single side glued onto itself after a rotation through an angle of π.

The $\mathcal{W}_i^j$ notation used to name the groups in this paper is the so–called Fejes Tóth notation. The i index denotes the highest degree of rotational symmetry in the group, e.g. $\mathcal{W}_3^1$ contains a rotation through an angle of $2\pi/3$. The j index separates between different groups in the same family. Table 2 shows the equivalent short international name used by crystallographers.

The possible combinations of boundary conditions are derived as follows: A reflection induce either Dirichlet or Neumann conditions, we have either $u\circ\sigma = -u$ or $u\circ\sigma = +u$. If the product of two reflections is an odd root of the identity, the two sides must be of the same type. E.g. for $\mathcal{W}_3^1$ we find $(\sigma_{ab}\sigma_{ac})^3 = \mathbf{e}$. If these two sides were of different type, we would get $u \equiv -u \Rightarrow u \equiv 0$. Similar restrictions appear for $\mathcal{W}_6^1$. For the remaining isometry groups any combinaiton of Dirichlet and Neumann is possible on the reflection sides.

The groups $\mathcal{W}_1^1$, $\mathcal{W}_2^2$ and $\mathcal{W}_2^3$ are not of immediate interest, since they represent 'strange' type boundary conditions. However these groups may appear later as coset symmetries in the FFT reductions.

The symmetries derived from the groups $\mathcal{W}_1$, $\mathcal{W}_1^2$ and $\mathcal{W}_2^2$ can be factorized in the direct product of one dimensional symmetries, and the corresponding symmetric FFT's can

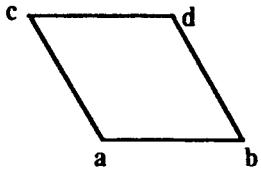
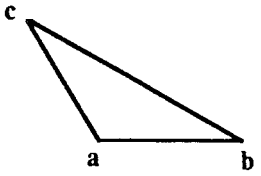
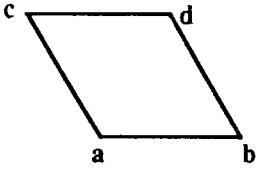
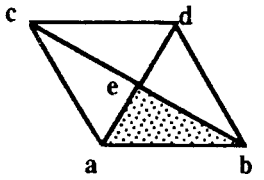
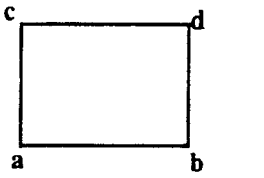
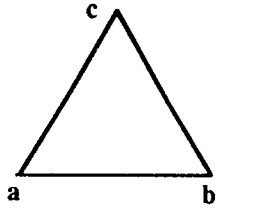
Group	*Domain*	*Generators*	*Boundary conditions:* Sides in counter-clockwise order, starting with ab.
$\mathcal{W}_1$	 parallelogram or rectangle	$\langle \tau_{ab}, \tau_{ac} \rangle$	Periodic in both directions
$\mathcal{W}_1^1$	 isoceles triangle	$\langle \tau_{ab}, \sigma_{bc} \rangle$	—
$\mathcal{W}_1^2$	 parallelogram or rectangle	$\langle \tau_{ac}, \sigma_{ac}, \sigma_{bd} \rangle$	PDPD or PDPN or PNPD or PNPN
$\mathcal{W}_2^1$	 triangle with 90° angle	$\langle \tau_{ab}, \tau_{ac}, \sigma_{ae}, \sigma_{be} \rangle$	—
$\mathcal{W}_2^2$	 rectangle	$\langle \sigma_{ab}, \sigma_{ac}, \sigma_{bd}, \sigma_{cd} \rangle$	Any combination of D and N
$\mathcal{W}_3^1$	 equilateral triangle	$\langle \sigma_{ab}, \sigma_{ac}, \sigma_{bc} \rangle$	DDD or NNN

Table 1: Translation–reflection groups for Poisson problems (cont. next page)

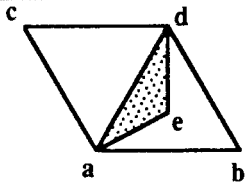
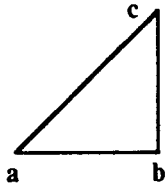
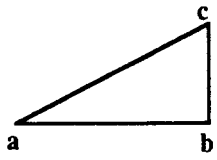
$\mathcal{W}_3^2$		30°–30°–120° triangle	$\langle \tau_{ab}, \tau_{ac}, \sigma_{ab}, \sigma_{ac} \rangle$	—
$\mathcal{W}_4^1$		45°–90°–45° triangle	$\langle \sigma_{ab}, \sigma_{ac}, \sigma_{bc} \rangle$	Any combination of D and N
$\mathcal{W}_6^1$		30°–90°–60° triangle	$\langle \sigma_{ab}, \sigma_{ac}, \sigma_{bc} \rangle$	DDD or DNN or NDD or NNN

Table 1 continued.

$\mathcal{W}_1 = p1$	$\mathcal{W}_2 = p2$	$\mathcal{W}_3 = p3$	$\mathcal{W}_4 = p4$	$\mathcal{W}_6 = p6$
$\mathcal{W}_1^1 = cm$	$\mathcal{W}_2^1 = cmm$	$\mathcal{W}_3^1 = p3m1$	$\mathcal{W}_4^1 = p41$	$\mathcal{W}_6^1 = p6m$
$\mathcal{W}_1^2 = pm$	$\mathcal{W}_2^2 = pmm$	$\mathcal{W}_3^2 = p31m$	$\mathcal{W}_4^2 = p4g$	
$\mathcal{W}_1^3 = pg$	$\mathcal{W}_2^3 = pmg$			
	$\mathcal{W}_2^4 = pgg$			

Table 2: Equivalence between Fejes Tóth and short international names

therefore be computed by doing symmetric 1–D transforms, first in one direction, then in the other direction. It is however possible that true 2–D transforms may be the fastest alternative also for these symmetries (see the comments in section 2). The remaining symmetries cannot be factorized in 1–D transforms, without breaking the symmetry and thereby loosing computational efficiency.

The subgroup relations between the seventeen crystallography groups are shown in figure 1. The diagram shows that every crystallography group is a subgroup of either $\mathcal{W}_4^1$ or $\mathcal{W}_6^1$. These two groups therefore contains information about all the remaining fifteen groups. For example a family of invariant grids for $\mathcal{W}_6^1$ are also invariant grids for all subgroups of $\mathcal{W}_6^1$. Figure 2 and 3 shows $\mathcal{W}_4^1$ and $\mathcal{W}_6^1$ with generators and invariant grids, constructed as in proposition 4. Both these groups can be generated by two translations and reflections passing through the node e.

The action of the reflections in $\mathcal{W}_4^1$ on the discrete group are given by the unimodular

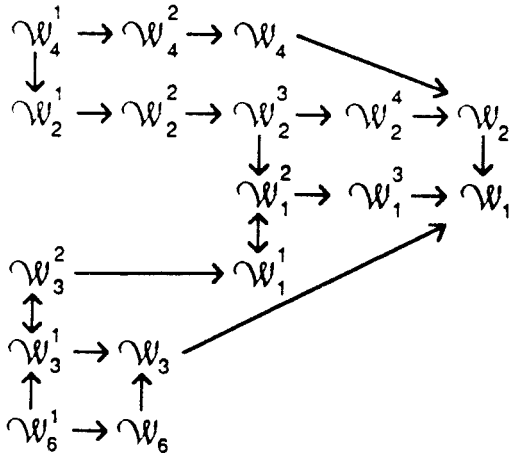


Figure 1: Subgroup relationship between 2–D crystallography groups

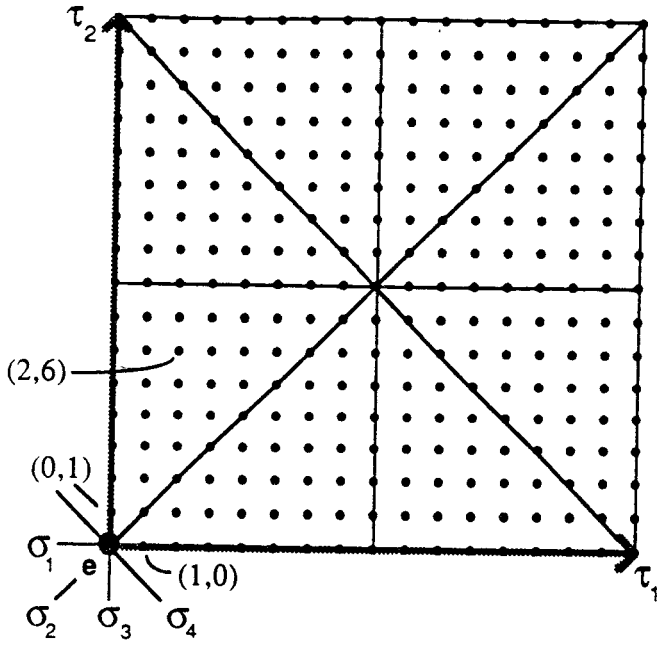


Figure 2: Group $\mathcal{W}_4^1$ with generators and $\mathcal{G}^{16}$ grid

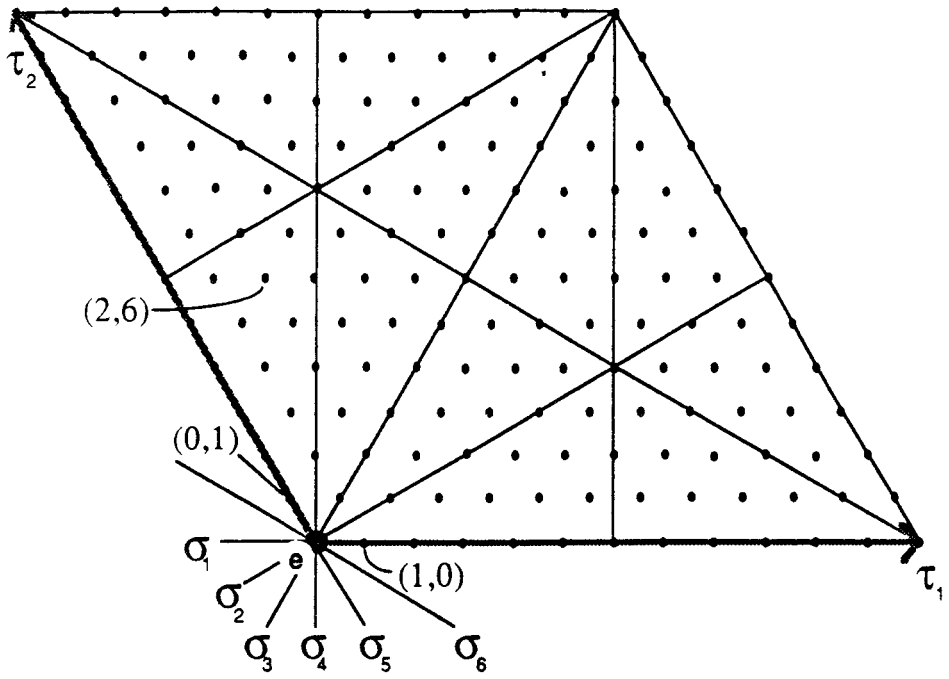


Figure 3: Group $\mathcal{W}_6^1$ with generators and $\mathcal{G}^{12}$ grid

matrices:

$$\sigma_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \sigma_2 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$
$$\sigma_3 = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} \quad \sigma_4 = \begin{pmatrix} 0 & -1 \\ -1 & 0 \end{pmatrix} \tag{11}$$

If the characters are defined by $\varepsilon_{\mathbf{p}}(\mathbf{g}) = \exp(-2\pi i/k\ \mathbf{p}^T D\mathbf{q})$ for

$$D = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

we see the dual automorphisms are given by $\hat{\sigma}_i = \sigma_i$.

For the group $\mathcal{W}_6^1$ the action of the the reflections on the group are given by:

$$\sigma_1 = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix} \quad \sigma_2 = \begin{pmatrix} 1 & 0 \\ 1 & -1 \end{pmatrix} \quad \sigma_3 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$
$$\sigma_4 = \begin{pmatrix} -1 & 1 \\ 0 & 1 \end{pmatrix} \quad \sigma_5 = \begin{pmatrix} -1 & 0 \\ -1 & 1 \end{pmatrix} \quad \sigma_6 = \begin{pmatrix} 0 & -1 \\ -1 & 0 \end{pmatrix} \tag{12}$$

These matrices are not orthogonal, but they are simultaneously orthogonalized by the similarity transform

$$M\sigma_i M^{-1}$$

where the matrix

$$M = \begin{pmatrix} 1 & -1/2 \\ 0 & \sqrt{3}/2 \end{pmatrix}$$

contains the directions of the translations in the group. If

$$\varepsilon_{\mathbf{p}}(\mathbf{g}) = \exp(-2\pi i/k\ \mathbf{p}^T D\mathbf{q}),$$

then the dual automorphisms $\hat{\sigma}_i$ are diagonalized by the matrix $M^{-T}D^T$. Hence $\mathcal{G}$ and the dual group $\hat{\mathcal{G}}$ must in general be displayed in different coordinate systems to show orthogonal symmetries. In the two dimensional case we may however escape from this by flipping a coordinate axis in the dual space, i.e. letting D be given by:

$$D = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

The dual automorphisms of $\sigma_1, \ldots, \sigma_6$ are then given by

$$\hat{\sigma_1} = \sigma_2\ ; \quad \hat{\sigma_2} = \sigma_1\ ; \quad \hat{\sigma_3} = \sigma_6\ ; \quad \hat{\sigma_4} = \sigma_5\ ; \quad \hat{\sigma_5} = \sigma_4\ ; \quad \hat{\sigma_6} = \sigma_3 \tag{13}$$

So $\mathcal{G}$ and $\hat{\mathcal{G}}$ can be displayed in the same diagram.

We are now prepared to find the dual symmetry groups for the symmetries in table 1. We assume the grids are generated as in proposition 4. Two different cases must be treated separately:

1. The groups $\mathcal{W}_1$, $\mathcal{W}_1^2$ and $\mathcal{W}_2^2$ with both Dirichlet and Neumann conditions on two parallel sides.
2. The remaining symmetries of table 1.

Case 2 is simplest. All the symmetry groups of case 2 can be generated by reflection symmetries about lines passing through the point $\mathbf{e}$. From proposition 6 we see that these reflection symmetries lifts as

$$\mathsf{u} \circ \sigma_i = \pm \mathsf{u} \Rightarrow \hat{\mathsf{u}} \circ \hat{\sigma_i} = \pm \hat{\mathsf{u}}$$

while the real value symmetry lifts as

$$\mathsf{u}(\mathbf{g}) = \mathsf{u}^*(\mathbf{g}) \Rightarrow \hat{\mathsf{u}}(\mathbf{p}) = \hat{\mathsf{u}}^*(\mathbf{p}^{-1})\ .$$

Let us first assume that $\mathcal{S}$ does not contain the real value symmetry. Then the symmetries in the $\mathcal{W}_4^1$ family lifts to the same symmetries on the dual space. For the $\mathcal{W}_6^1$ family, the reflections are permuted, and we may get a different crystallography group on the dual space. The reflections in $\mathcal{W}_3^1$ and $\mathcal{W}_3^2$ are mapped onto each other in the dual space.

The addition of the real value symmetry to $\mathcal{S}$, induces an inversion symmetry in $\hat{\mathcal{S}}$. The effect of this depends on whether or not an inversion symmetry was already present in $\hat{\mathcal{S}}$, and on the sign of this symmetry. If

$$\hat{\mathsf{u}}(\mathbf{p}) = \pm \hat{\mathsf{u}}(\mathbf{p}^{-1})$$

then

$$\mathsf{u}(\mathbf{g}) = \mathsf{u}^*(\mathbf{g}) \Rightarrow \hat{\mathsf{u}}(\mathbf{p}) = \pm \hat{\mathsf{u}}^*(\mathbf{p})$$

hence $\hat{\mathsf{u}}$ is either real or purely imaginary. If no reflection symmetry group was already present in $\hat{\mathcal{S}}$ then $\hat{\mathsf{u}}$ remains complex, but the isometry group of $\hat{\mathsf{u}}$ becomes larger.

In case 1 above, the Diriclet–Neumann direction induces a symmetry

$$\mathsf{u} \circ \tau = -\mathsf{u}$$

u		û	
Data	*Isometries*	*Data*	*Isometries*
C	$\mathcal{W}_1$	C	$\mathcal{W}_1$
C	$\mathcal{W}_1^1$	C	$\mathcal{W}_1^1$
C	$\mathcal{W}_2^1$	C	$\mathcal{W}_2^1$
C	$\mathcal{W}_2^1$	C	$\mathcal{W}_2^1$
C	$\mathcal{W}_2^2$	C	$\mathcal{W}_2^2$
C	$\mathcal{W}_3^1$	C	$\mathcal{W}_3^2$
C	$\mathcal{W}_3^2$	C	$\mathcal{W}_3^1$
C	$\mathcal{W}_4^1$	C	$\mathcal{W}_4^1$
C	$\mathcal{W}_6^1$	C	$\mathcal{W}_6^1$
R	$\mathcal{W}_1$	C	$\mathcal{W}_2$
R	$\mathcal{W}_1^1$	C	$\mathcal{W}_2^1$
R	$\mathcal{W}_1^2$	C	$\mathcal{W}_2^2$
R	$\mathcal{W}_2^1$	R or I	$\mathcal{W}_2^1$
R	$\mathcal{W}_2^2$	R or I	$\mathcal{W}_2^2$
R	$\mathcal{W}_3^1$	C	$\mathcal{W}_6^1$
R	$\mathcal{W}_3^2$	C	$\mathcal{W}_6^1$
R	$\mathcal{W}_4^1$	R or I	$\mathcal{W}_4^1$
R	$\mathcal{W}_6^1$	R or I	$\mathcal{W}_6^1$

$\mathcal{W}_2$ is the group generated by
two translations and inversion.

Table 3: Duality between isometries of u and û

where τ is a translation through twice the distance between the parallel sides. This symmetry lifts to

$$\hat{\mathsf{u}} = -\varepsilon_{\mathbf{p}}(\mathbf{t}) \cdot \hat{\mathsf{u}}$$

where $\mathbf{t} = \tau(\mathbf{e})$. Hence:

Proposition 8 *In case 1 above, the Fourier coefficients with an even numbered coordinate in the Dirichlet–Neumann direction(s) are zero.*

Proof: $\mathbf{t}^2 = \mathbf{e}$ hence $\varepsilon_{\mathbf{p}}(\mathbf{t})$ is $+1$ for the even numbered coordinates and -1 for the odd.

□

Table 3 summarize the resulting crystallography groups in the dual space.

FFT algorithms for all the 2-D Poisson symmetries can be constructed by finding a sequence of invariant subgroups $\mathcal{H}^1 < \mathcal{H}^2 < \cdots < H^i = \mathcal{G}$ and using the general approach of the previous section. The symmetries of u restricted to cosets, $\mathsf{v} = \mathsf{u}|_{\mathcal{H}\mathbf{q}_i}$ are found by using proposition 5. The isometries of v must form a subgroup of $\mathcal{W}$, and the possible crystallography groups for v can be read out of figure 1. Note that reflections mapping the coset onto itself does not necessarily pass through a gridpoint of the coset. Therefore we cannot always avoid isometries of the type $w(\mathbf{h}) = \sigma(\mathbf{ht})$ on v. In the dual space $\hat{\mathsf{v}}$ retains (a subset of) the symmetries of $\hat{\mathsf{u}}$ after a multiplication by a twiddle factor $\varepsilon_{\mathbf{p}}^*(\mathbf{q})$, where $\mathbf{q}$ is the 'offset' of the coset. Table 3 and figure 1 are keys to understanding the different possibilities.

6 A triangular Poisson solver

In this section we are briefly describing the implementation of a triangular Poisson solver based on symmetric FFT's. We want to solve $\nabla^2 v = u$ in an equilateral triangle with homogeneous Dirichlet boundary conditions on all sides, and where u is real valued.

Extend u to a periodic function on $\mathbf{R}^2$ by reflection symmetries about all the sides of the domain. The isometries of u forms the group $\mathcal{W}_3^1$. Let the domain be discretized with a uniform grid $\mathcal{G}^k$ as in figure 3. We let $k = 3 \cdot 2^i$ for some i (when 3 divides k, we get one gridpoint in each corner of the triangle). The symmetries of the discretized function u and its DFT $\hat{u}$ are generated by:

$\mathcal{S}$			$\hat{\mathcal{S}}$		
$u \circ \sigma_2$	$=$	$-u$	$\hat{u} \circ \sigma_1$	$=$	$-\hat{u}$
$u \circ \sigma_4$	$=$	$-u$	$\hat{u} \circ \sigma_5$	$=$	$-\hat{u}$
$u \circ \sigma_6$	$=$	$-u$	$\hat{u} \circ \sigma_3$	$=$	$-\hat{u}$
u^*	$=$	u	$\hat{u}(\mathbf{p}^{-1})$	$=$	$\hat{u}^*(\mathbf{p})$

The Fourier transform $\hat{u}$ is complex valued with isometry group $\mathcal{W}_6^1$.

Our symmetric FFT implementation is based on radix 2⊗2 reductions, except at the bottom level where radix 3⊗3 reductions are performed. Recall that in the divide phase of unsymmetric radix 2⊗2 transforms, each coset is split in four smaller copies. In this symmetric transform, the number of different 'daughter' cosets and their symmetries depends on the symmetries of the 'mother' coset. From proposition 5 we find the following symmetries in the divide phase:

Isometries of cosets level j		Isometries of cosets level $j-1$
$\mathcal{W}_3^1$	$\longrightarrow$	$\mathcal{W}_3^1$, $\mathcal{W}_1^1$
$\mathcal{W}_1^1$	$\longrightarrow$	$\mathcal{W}_1^1$, $\mathcal{W}_1^1$, $\mathcal{W}_1$
$\mathcal{W}_1$	$\longrightarrow$	$\mathcal{W}_1$, $\mathcal{W}_1$, $\mathcal{W}_1$, $\mathcal{W}_1$

Geometrically, the first of these splittings are described as follows: The triangular grid splits in four interlacing grids with twice the mesh size. Three of these combine in the fundamental domain of the group $\mathcal{W}_1^1$. The remaining constitute the fundamental domain of $\mathcal{W}_3^1$.

After two radix 2⊗2 reductions we are left with one $\mathcal{W}_3^1$ coset, two $\mathcal{W}_1^1$ cosets, and one $\mathcal{W}_1$ coset. The remaining twelve cosets are equivalent to one of these four under symmetries in $\mathcal{S}$.

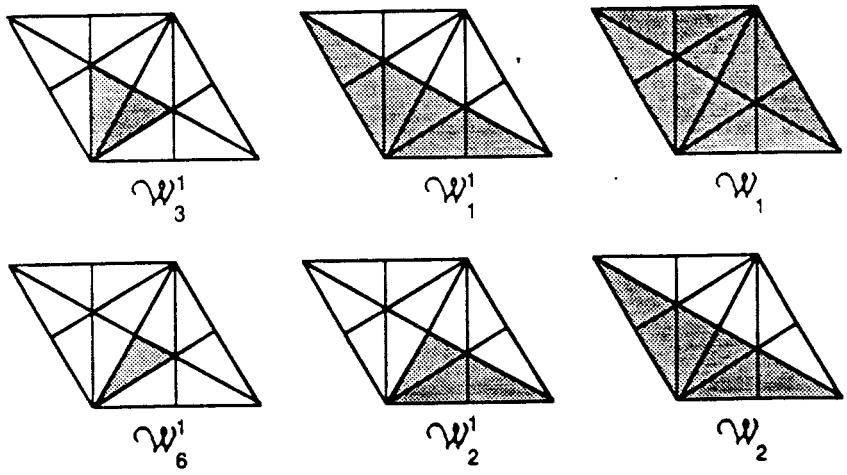


Figure 4: Fundamental domains for isometry groups in the symmetric FFT

In the conquer phase, the DFT of the cosets assemble as:

Isometries of parts level $j-1$		Isometries of parts level j
$\mathcal{W}_6^1$, $\mathcal{W}_2^1$	$\longrightarrow$	$\mathcal{W}_6^1$
$\mathcal{W}_2^1$, $\mathcal{W}_2^1$, $\mathcal{W}_2$	$\longrightarrow$	$\mathcal{W}_2^1$
$\mathcal{W}_2$, $\mathcal{W}_2$, $\mathcal{W}_2$, $\mathcal{W}_2$	$\longrightarrow$	$\mathcal{W}_2$

In total six different isometry groups are involved in the symmetric FFT. Fundamental domains for these are shown in figure 4. The fundamental domains are not shown at the boundary of the regions. Some care must be taken when defining the fundamental domain at the boundary (to avoid double points).

The implementation of the symmetric transform was aimed at running efficiently on a Cray type vector computer. The code is divided in two main parts; a symbolic part, where multiplication factors and permutation vectors are calculated, and a numerical part where a given input vector u is transformed to $\hat{u}$. If many vectors are to be transformed, the symbolic part needs to be run only once. Even if we change from the grid $\mathcal{G}^k$ to a corser grid $\mathcal{G}^l$, where l divides k, the same multiplication factors and permutation data can be reused. We are therefore mainly interested in organizing the computations so that the numerical part of the computations run as fast as possible.

In the numerical part the data is initially sorted so that the main conquer loops can follow the diagram of figure 5. Four complex numbers are read sequentially from memory. The first is (possibly) sign–changed, the other three are multiplied by complex factors. All four

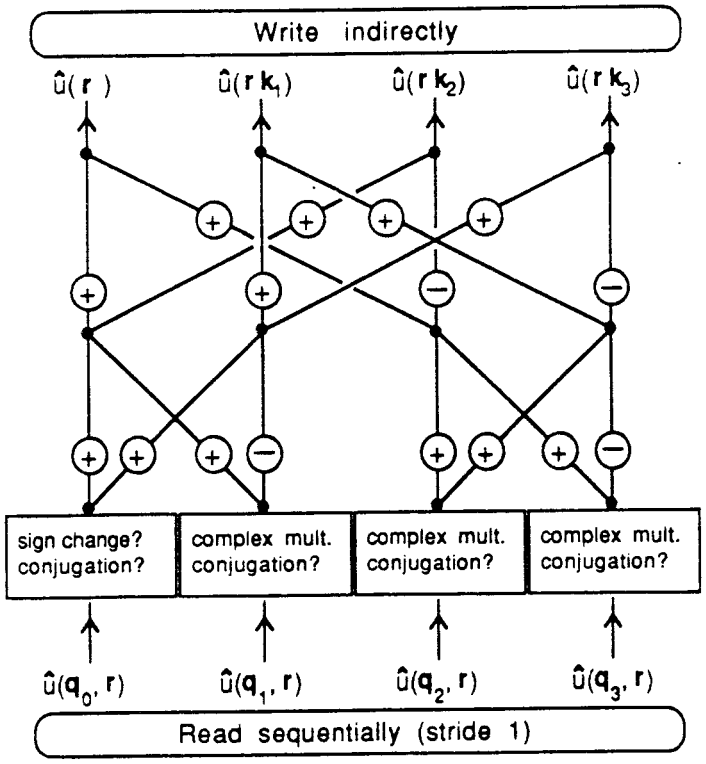


Figure 5: Data flow in main conquer loop

numbers are then (possibly) complex conjugated. Finally the four numbers flow through the 'butterfly diagram', and the results are written indirectly back to the memory. The complex coefficients, signchange and conjugation data, and the storage addresses are computed from equations 7–8 and propositions 6–7. These numbers depends on the computation level j, and the symmetries of the part to be assembled ($\mathcal{W}_6^1$, $\mathcal{W}_2^1$ or $\mathcal{W}_2$). It is therefore natural to process these different possibilities in separate loops or subroutines. A problem with this approach is that the vector lengths gets very short at the bottom levels. To avoid this, we instead implemented the conquer phase by an explicit loop unrolling, i.e. at each level j all the different parts $\hat{u}(\mathbf{q}_i, \mathbf{r})$ are sorted and stacked together in four long vectors. The conquer phase is at each level performed in a single long loop where the four vectors are read contiguously. The vector length then becomes almost the same, about $k^2/48$, at all levels. The disadvantage of this approach is that more memory space is needed to store the coefficients of the transform. Another technical problem is that some grid points lying on lines of symmetry are needed several different places in memory to fit into the scheme of figure 5. To avoid irregularities in the data flow, we need between two conquer loops an intermediate loop where these 'double points' are copied to a different memory location.

Figure 6 shows timings of our symmetric transform, SFFT, compared to other fast vectorized FFT's for unsymmetric 2-D problems. The symmetry group $\mathcal{S}$ contains twelve symmetries, so we hope that SFFT is twelve times faster than the unsymmetric codes. The figure shows that our code SFFT is more than twelve times faster than OMNILIB and BENCHLIB. For large k, SFFT is about nine times faster than VECTOR-FFT, the fastest available 2–D complex unsymetric FFT for Cray computers. The timing of 2) 4) and 5) are due to [6]. The actual times are shown in table 4. The timing of SFFT is given in table 5. The symbolic

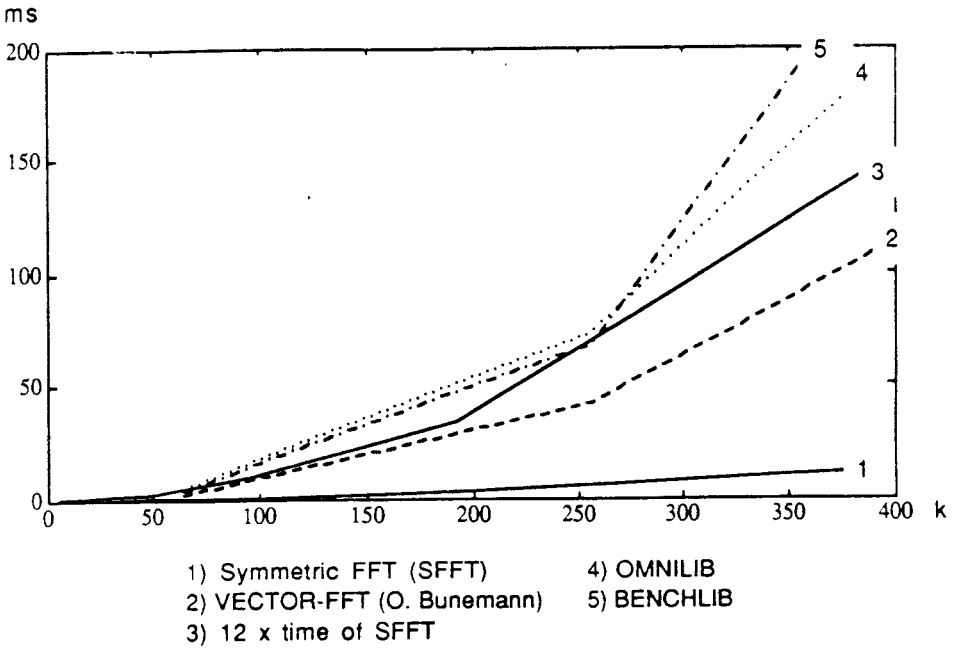


Figure 6: Timings in ms for 2-D FFTs on $k \times k$ grids; Cray-XMP

Routine	k=64	k=256	k=512
VECTOR-FFT	2.09	41.5	170
OMNILIB	4.76	73.1	299
BENCHLIB	4.29	68.0	387

Table 4: Timings in ms for vectorized FFT's on $k \times k$ grid; Cray-XMP

part of SFFT is very slow, but since this part is only to be run once, no work was spent on optimizing this part of the code.

In the remaining part of this section we will show how the symmetric FFT may be used to find the solution of the Poisson problem. We are giving two different methods for discretizing ∇^2 on the grid:

- Finite difference approximation.
- Spectral collocation approximation.

Let h denote the distance between neighbor nodes in the grid. A second order finite difference approximation to ∇^2 is given by the matrix $(-2/3h^2)A$, where A is defined by the matrix stencil

Grid size k	6	12	24	48	96	192	384
Time of transform	4.1E-2	5.9E-2	9.4E-2	21.E-2	81.E-2	2.8	11.8
Time of symbolic part	1.2	2.4	6.2	18.	63.	232.	889.

Table 5: Timings in ms for SFFT on $k \times k$ grid; Cray-XMP

$$A = \left[\begin{array}{ccc} & -1 & -1 \\ -1 & 6 & -1 \\ -1 & -1 & \end{array} \right]$$

We seek a solution of the equation

$$Av = -(3h^2/2)u \tag{14}$$

Or more precisely, since A is a singular matrix, we seek the vector

$$v = -(3h^2/2)A^+u,$$

where A^+ is the pseudoinverse of A. From the symmetry properties of u it can be shown that $u \in \mathcal{R}(A)$, so $Av = -(3h^2/2)u$.

Note that A commutes with the reflections $\sigma_1, \ldots, \sigma_6$, so since

$$u \circ \sigma_i = -u \quad \text{for } i \in \{2,4,6\}$$

we must also have

$$v \circ \sigma_i = -v \quad \text{for } i \in \{2,4,6\}$$

and $v = 0$ on the boundary of the equilateral triangle. A is a (2–D generalization of a) circulant matrix, and the characters ε_p are the eigenfunctions of A:

Proposition 9 *The characters ε_p are the eigenvectors of A:*

$$A\varepsilon_p = \lambda_p \cdot \varepsilon_p$$

Proof: Let $g_1 = (1,0)^T$ and $g_2 = (0,1)^T$. Then

$$\begin{aligned} A \cdot \varepsilon_p(g) &= 6\varepsilon_p(g) - \varepsilon_p(gg_1) - \varepsilon_p(gg_1^{-1}) \\ &\quad -\varepsilon_p(gg_2) - \varepsilon_p(gg_2^{-1}) - \varepsilon_p(gg_1g_2) - \varepsilon_p(gg_1^{-1}g_2^{-1}) \\ &= \lambda_p \cdot \varepsilon_p(g) \end{aligned}$$

where

$$\lambda_p = 6 - (\varepsilon_p(g_1) + \varepsilon_p^*(g_1)) - (\varepsilon_p(g_2) + \varepsilon_p^*(g_2)) - (\varepsilon_p(g_1g_2) + \varepsilon_p^*(g_1g_2))$$

□

Thus the DFT of u is just the expansion of u in the eigenvectors of the matrix A. We can therefore find v by:

1. Compute $\hat{u}$ by SFFT.
2. Let $\hat{v}(p) = \begin{cases} -(3h^2/2)\hat{u}(p)/\lambda_p & \text{if } \lambda_p \neq 0 \\ 0 \text{ otherwise} \end{cases}$
3. Compute v by ISFFT.

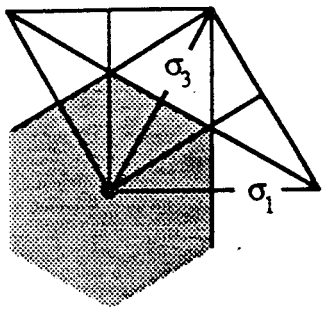


Figure 7: The domain of $\mathbf{p}$ nodes

ISFFT is the inverse transform of SFFT. Except from a scaling by the factor $\#(\mathcal{G})$, it is a symmetric FFT for a vector with symmetry group $\hat{\mathcal{S}}$. It may also be computed by reversing all the steps of the SFFT algorithm. In both cases ISFFT costs the same to compute as the SFFT, so the total cost of computing $\hat{\mathrm{v}}$ is approximately twice the cost of the SFFT algorithm.

The spectral collocation approximation originates in a continuous description of the DFT. Let $\mathbf{g}_1$ and $\mathbf{g}_2$ be the grid nodes $\mathbf{g}_1 = (1,0)^T$, $\mathbf{g}_2 = (0,1)^T$ on figure 3. If we let $\mathbf{g}_1$ and $\mathbf{g}_2$ be unit coordinate vectors for the plane, it is readily shown that the continuous Laplacian is given by:

$$\nabla^2 = \frac{4}{3h^2}\left(\frac{\partial^2}{\partial g_1^2} + \frac{\partial^2}{\partial g_2^2} + \frac{\partial^2}{\partial g_1 \partial g_2}\right)$$

Define the characters as in equation 4, where $\mathbf{g}$ is allowed to run over the entire plane. By direct calculation we find:

Lemma 9 *The characters are eigenfunctions of the continuous Laplacian:*

$$\nabla^2 \varepsilon_{\mathbf{p}} = \lambda_{\mathbf{p}} \cdot \varepsilon_{\mathbf{p}}$$

where

$$\lambda_{\mathbf{p}} = -(16\pi h/3k)^2(p_1^2 + p_2^2 - p_1 p_2)$$

The function $\mathrm{u} = \sum_{\mathbf{p}\in\mathcal{G}} \hat{\mathrm{u}}(\mathbf{p})\varepsilon_{\mathbf{p}}$ can be regarded as a collocation approximation of the continuous u in eigenfunctions of the Laplacian, where u interpolates u in the gridpoints. Some care must be taken when u is defined as a continuous function. If τ_1 and τ_2 are the translations of figure 3, then

$$\varepsilon_{\tau_1 \mathbf{p}}(\mathbf{g}) = \varepsilon_{\tau_2 \mathbf{p}}(\mathbf{g}) = \varepsilon_{\mathbf{p}}(\mathbf{g}) \quad \text{for all } \mathbf{g}\in\mathcal{G}$$

so an infinite family of different continuous functions are identical (or *aliased*) on the grid. To represent the lowest eigenmodes of the Laplacian, we must choose the functions $\varepsilon_{\mathbf{p}}$ with the lowest wavenumbers, i.e. we choose $\mathbf{p}$ from the hexagonal region in figure 7.

For a continuous description of *symmetric* FFT's we first note that characters related by automorphisms share the same eigenvalue:

Lemma 10 *Let $\hat{\alpha}$ be an automorphism on $\mathcal{G}$. Then*

$$\nabla^2 \varepsilon_{\hat{\alpha}\mathbf{p}} = \lambda_{\mathbf{p}} \cdot \varepsilon_{\hat{\alpha}\mathbf{p}}$$

where $\lambda_{\mathbf{p}}$ is the eigenvalue of $\varepsilon_{\mathbf{p}}$.

Proof: ∇^2 commutes with any isometry of the plane, so:

$$\nabla^2 \varepsilon_{\hat{\alpha}\mathbf{p}} = \nabla^2(\varepsilon_{\mathbf{p}} \circ \alpha) = (\nabla^2 \varepsilon_{\mathbf{p}}) \circ \alpha = \lambda_{\mathbf{p}} \cdot \varepsilon_{\mathbf{p}} \circ \alpha = \lambda_{\mathbf{p}} \cdot \varepsilon_{\hat{\alpha}\mathbf{p}}$$

□

Now define the linear combination of characters:

$$\begin{aligned} \tilde{\varepsilon}_{\mathbf{p}} &= \varepsilon_{\mathbf{p}} + \varepsilon_{\mathbf{p}} \circ \sigma_6 \circ \sigma_2 + \varepsilon_{\mathbf{p}} \circ \sigma_6 \circ \sigma_4 - \varepsilon_{\mathbf{p}} \circ \sigma_2 - \varepsilon_{\mathbf{p}} \circ \sigma_4 - \varepsilon_{\mathbf{p}} \circ \sigma_6 \\ &= \varepsilon_{\mathbf{p}} + \varepsilon_{\hat{\alpha}_1 \mathbf{p}} + \varepsilon_{\hat{\alpha}_2 \mathbf{p}} - \varepsilon_{\hat{\sigma}_2 \mathbf{p}} - \varepsilon_{\hat{\sigma}_4 \mathbf{p}} - \varepsilon_{\hat{\sigma}_6 \mathbf{p}} \end{aligned}$$

where $\hat{\alpha}_1 = \sigma_6 \hat{\circ} \sigma_2 = \sigma_1 \circ \sigma_3$, $\hat{\alpha}_2 = \sigma_6 \hat{\circ} \sigma_4 = \sigma_5 \circ \sigma_3$, $\hat{\sigma}_2 = \sigma_1$, $\hat{\sigma}_4 = \sigma_5$ and $\hat{\sigma}_6 = \sigma_3$.

Let Δ denote the fundamental domain of $\mathcal{W}_3^1$ shown in figure 4. It is readily checked that $\tilde{\varepsilon}_{\mathbf{p}}(\mathbf{g}) = 0$ for all points $\mathbf{g}$ on the boundary of Δ, so $\tilde{\varepsilon}_{\mathbf{p}}$ are eigenfunctions of ∇^2 satisfying homogeneous Dirichlet boundary conditions. From the symmetries of $\hat{\mathsf{u}}$, we see that

$$\mathsf{u}(\mathbf{g}) = \sum_{\mathbf{p} \in \mathcal{G}} \hat{\mathsf{u}}(\mathbf{p}) \varepsilon_{\mathbf{p}}(\mathbf{g}) = {\sum_{\mathbf{p} \in \hat{\Delta}}}' \hat{\mathsf{u}}(\mathbf{p}) \tilde{\varepsilon}_{\mathbf{p}}(\mathbf{g})$$

where $\hat{\Delta}$ is the kite–shaped sector of the hexagon in figure 7 limited by the reflection lines of σ_1 and σ_3. Some boundary points $\mathbf{p}$ are countend several times in the right hand sum. The prime $'$ indicates that the corresponding terms of the sum are given a weight inversely proportional to the multiplicity of the point. Hence:

Proposition 10 *The function* u *is the collocation approximation of* u *in the space of eigenfunctions of* ∇^2 *satisfying homogeneous Dirichlet boundary conditions on* Δ. *The approximant* u *interpolates* u *in the gridpoints.*

The spectral solution to the Poisson problem is the function v satisfying

$$\begin{aligned} \nabla^2 \mathsf{v} &= \mathsf{u} \text{ in } \Delta \\ \mathsf{v} &= 0 \text{ on } \partial(\Delta) \end{aligned}$$

We can compute v by:

1. Compute $\hat{\mathsf{u}}$ by SFFT.
2. Let $\hat{\mathsf{v}}(\mathbf{p}) = \hat{\mathsf{u}}(\mathbf{p})/\lambda_{\mathbf{p}}$
3. Compute v by ISFFT.

The only difference between the spectral collocation solution and the finite difference solution is the value of $\lambda_{\mathbf{p}}$. The advantage of the spectral solution is that for an infinitely smooth function u, the error decays exponentially in h, compared to $\mathcal{O}(h^2)$ for the finite difference solution [1]. The finite difference solution has the advantage that the solution process can be speeded up by combining symmetric FFT's with cyclic reduction techniques [9].

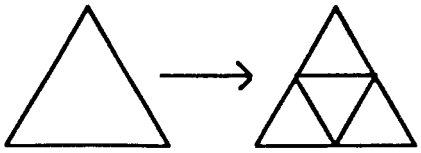


Figure 8: First step of symmetric decimation in space FFT for equilateral triangle

7 Concluding remarks

The algorithms in this paper are formulated as 'decimation in frequency' algorithms. For every symmetry group there also exists a symmetric dual transform called a 'decimation in space' algorithm. This is obtained by reversing the steps of the decimation in frequency algorithm for the dual symmetry group. E.g. for the equilateral triangle, the first step of the decimation in frequency algorithm decompose the grid in four interlacing grids of half the meshsize. The first step of the decimation in space algorithm, on the other hand, decompose the grid in four smaller copies as shown in figure 8, where three of the triangles combine in the fundamental domain of $\mathcal{W}_1^1$. If we are satisfied with obtaining the Fourier coefficients in scrambled order, these transforms may be slightly faster, since an initial ordering of the nodes can be avoided.

In the 1–D case two different approaches to symmetric FFT's are found in the literature:

1. Several symmetric transforms are combined to a complex unsymmetric transform. E.g. two real sequences can be combined into a complex sequence. The Fourier transform is finally decomposed into the transfoms of the original sequences.

2. A special symmetric transform is constructed for the specific symmetry group involved.

Our approach follows strategy 2. The first strategy has the advantage that the general complex transform can be used to handle different symmetries. Swarztrauber [10] notes that the latter approach is somewhat faster, since the combination and decomposition is avoided. It is an open question if the approach 1 can be used for all the symmetry groups of this paper, and in that case how to describe this approach in a general manner.

It was our experience, when coding the equilateral triangle, that these symmetric transforms are difficult to program and debug. However once one of the more complex 2-D symmetries are coded, it is fairly trivial to modify the code to handle the other 2-D symmetry groups. We even belive that a general code can be developed that can handle any periodic symmetry group in any number of dimensions. It should be possible to obtain an effective implementation on a vector computer, at least if one does not try to optimize the memory requirement, and if the time of the symbolic preprocessing is of no concern.

References

[1] Canuto C., Hussaini M.Y., Quarteroni A., Zang T.A.
Spectral Methods in Fluid Dynamics
Springer Verlag, New York, 1987.

[2] Cooley J.W., Tukey J.W.
An algorithm for the machine computation of complex Fourier series
Math. Comp. vol. 19,pp.297-301, 1965.

[3] Dym H., McKean H.P.
Fourier Series and Integrals
Academic Press, London 1972.

[4] Fraleigh J.B.
A First Course in Abstract Algebra
Addison–Wesley, London, 1982.

[5] Grunbaum G.C., Shepard B.
Tilings and Patterns
Freeman, New York 1986.

[6] Leary B.
Make your computations more efficient with FFTs
EMAIL distribution message, Stanford 1988.

[7] Martin G.E.
Transformation Geometry
Springer Verlag, New York 1980.

[8] MacGillavry P.
Symmetry aspects of M.C. Eschers periodic drawings
Oosthoek, Utrecht 1965.

[9] Munthe–Kaas H.
Cyclic Reduction and Matrices with a Group Structure
Dept. for Math. Sciences, Norwegian Inst. of Technology, Numerical Analysis tech. rep. 1989.

[10] Swarztrauber P.N.
The methods of cyclic reduction, Fourier analysis and the FACR algorithm for the discrete solution of Poisson's equation on a rectangle
Siam Rev., vol. 19, p.490, 1977.

[11] Swarztrauber P.N.
Symmetric FFT's
Math. Comp., vol. 47, no. 175, pp.323-346, 1986.

[12] Temperton, C.
Self-sorting mixed-radix fast Fourier transforms
J. Comput. Phys., vol. 52, pp.1-23, 1983.