A generalized FFT for Clifford algebras

Paul Leopardi [*]

Abstract

This paper describes a relationship between fast real matrix representations of real universal Clifford algebras and the generalized Fast Fourier Transform for supersolvable groups. Detailed constructions of algorithms for the forward and inverse representations for Clifford algebras are given, with proof that these need at most $\mathrm{O}(d \log d)$ operations. The algorithms have been implemented and tested in the GluCat `C++` library, and some timing results are included.

1 Introduction

Generalized Fast Fourier Transforms. After Cooley and Tukey re-discovered the fast Fourier transform (FFT) in 1963–1965 ([24], [33], [25]), various researchers found ways to generalize the discrete Fourier transform (DFT) from cyclic groups to abelian [11] and non-abelian groups, resulting in generalized Fourier transforms (GFTs). More recently, there have been a number of investigations into fast algorithms for the GFT on non-abelian groups, resulting in generalized FFTs (GFFTs) ([4], [10], [27]). For a summary of the state of the art, see Maslen and Rockmore [49]. See Maslen and Rockmore [46] for a more detailed survey, the book by Clausen and Baum [19], and later articles ([7], [47], [48], [20], [52], [21]).

One motivation for studying the GFT for finite groups is the need to efficiently perform multiplications in the group algebra. The GFT is an isomorphism from

[*] The author is a PhD student at the School of Mathematics, University of New South Wales, supported by a University Postgraduate Award. Key results of this work were presented at the Clifford minisymposium at ICIAM, Sydney, 7 July 2003. Thanks to Sören Krausshar, minisymposium co-organizer. Thanks also to the staff of the School of Mathematics at UNSW, and the reviewers. This paper is dedicated to the memory of Pertti Lounesto.

1991 *Mathematics Subject Classification :*

Key words and phrases : Generalized FFT, Clifford algebras, real matrix representations, linear complexity.

the group algebra to a subalgebra of a complex matrix algebra. Multiplication in this complex matrix algebra is often more efficient than multiplication in the group algebra. Conversely, there has been some investigation to see whether matrix multiplication can itself be made more efficient by use of a suitable group algebra [23]. For these and other applications, see also Clausen and Baum ([19] Chapters 10, 11), Rockmore [59], and the recent book by Chirikjian and Kyatkin [17].

Numerical analysis with Clifford algebras. At the same time, there has been interest in numerical computation with Clifford algebras. Computation can in many cases be done using a symbolic and coordinate free approach, as per the CLIFFORD package for Maple [1], but for eg. the numerical solution of differential and integral equations, numerical Clifford algebra tools are arguably more suitable. One of the first such tools was the standalone CliCal calculator for MS-DOS [42] [43]. The GABLE tutorial package [45] uses Matlab. More recently there have been a number of `C` and `C++` libraries including CLU [55], GaiGen [32], a prototype by Arvind Raja [58], and GluCat [41]. See the articles by Lounesto ([43], [2] pp. iv–xv) for earlier surveys.

One of the key tasks such packages must perform is multiplication in the Clifford algebra. As noted by Lounesto [43], multiplication in a d dimensional real universal Clifford algebra requires $\mathrm{O}(d^2)$ operations, but only $\mathrm{O}(d^{3/2})$ in a suitable isomorphic subalgebra of a matrix algebra.

What is the connection between the two? The situation for Clifford algebras then seems very much like that for group algebras. This raises the questions:

- How is a real matrix representation of a Clifford algebra related to a GFT for a finite group?
- How can this relationship be used to make numerical Clifford multiplication more efficient?

This paper. For a real universal Clifford algebra, we use the term *matrix representation* to mean an algebra homomorphism from the Clifford algebra to a matrix algebra. The term *fast real matrix representation* is used here in the same spirit as FFT and GFFT, ie. a fast algorithm for a real matrix representation.

The main results of this paper are detailed constructions for fast real matrix representations and fast inverse real matrix representations for real universal Clifford algebras, with proof that these algorithms need at most $\mathrm{O}(d \log d)$ operations. The algorithms have been implemented and tested in GluCat and some timing results are included here.

The recursive expressions needed for these algorithms have been known since at least 1993 [22] and possibly well before then [56], but they have apparently not yet been used for this purpose.

The algorithms described here are not to be confused with either the *discrete Clifford Fourier transform* of Felsberg, et al. [29] or the related transforms as described in [15] and [16]. Those transforms are based on abelian groups.

2 The GFT for finite groups

For the complex group algebra of a finite group, we use the term *matrix representation* in the sense of Curtis and Reiner ([26] pp. 45–47), Jacobson ([37] p 403) and Clausen and Baum ([19] pp. 30–33) to mean an algebra homomorphism from the complex group algebra to a complex matrix algebra:

Definition 2.1. *Let A be a finite dimensional algebra over a field K. A* matrix representation *of A of degree N is an algebra homomorphism*

$$T : A \to K(N),$$

where $K(N)$ is the algebra of $N \times N$ matrices over K.

With this definition in mind, we can now define the generalized Fourier transform of a finite group.

Definition 2.2. *([10], [27], [19] Section 2.3, pp. 36–40) A* generalized Fourier transform *(GFT) for a finite group $\mathbb{G}$ is an algebra injection D, which is a direct sum of a complete set of inequivalent irreducible complex matrix representations of the group algebra $\mathbb{C}\mathbb{G}$.*

$$D : \mathbb{C}\mathbb{G} \to \mathbb{C}(M),\ D = \bigoplus\nolimits_{k=1}^{n} D_k,$$
$$\text{where } D_k : \mathbb{C}\mathbb{G} \to \mathbb{C}(m_k),\ \text{and } \sum_{k=1}^{n} m_k = M.$$

This definition corresponds most closely to Clausen and Baum's Definition (2.1.3) ([19] p 39) together with Theorem (2.1.5) ([19] p 40). For an equivalent definition in terms of representations of finite groups and complex functions on finite groups, see Maslen and Rockmore ([47] pp. 172–173, [49] p 1153) or Chirikjian and Kyatkin, ([17] Section 8.1). In brief, the correspondence is as follows:

Maslen-Rockmore, Chirikjian-Kyatkin	↔	Clausen-Baum, this paper
Complex function of finite group	↔	Element of complex group algebra
Convolution product	↔	Group algebra product
Fourier transform at complex matrix representation of group	↔	Matrix representation of complex group algebra

We now define *generalized fast Fourier transforms.*

Definition 2.3. *As per Clausen and Baum [19], we call any fast algorithm for the GFT a* generalized fast Fourier transform *(GFFT).*

Here *fast* means faster than the naive sparse matrix–vector multiplication algorithm for the linear transformation D from $\mathbb{C}\mathbb{G}$ to $D(\mathbb{C}\mathbb{G})$ using the usual bases for $\mathbb{C}\mathbb{G}$ and $\mathbb{C}(m_k)$.

Linear complexity.

Definition 2.4. *([6] [19] (3.2) p 52)*

For $c \geqslant 2$, the c-linear complexity $L_c(X)$*, of a linear operator X counts non-zero additions $\mathbb{A}(X)$, and multiplications by all non-zero scalars up to absolute value c, except* 1 *and* -1*. Multiplication by a larger scalar is counted as a number of multiplications by scalars of size c or less.*

The ∞-linear complexity $L_\infty(X)$ *counts non-zero additions and non-zero multiplications by all scalars except* $0, 1$ *and* -1.

The GFFT for supersolvable groups. Fast algorithms for the GFT are known for some broad classes of finite groups. For the symmetric group S_n Maslen [48] gives an algorithm which requires $\mathrm{O}(n(n-1)n!)$ operations, and Maslen and Rockmore [49] gives a related fast algorithm for the wreath product $S_n[G]$. Maslen and Rockmore [47] gives a general approach which is applied to a number of classes of finite groups including Weyl groups and Chevalley groups.

For solvable groups, Beth [10] and Clausen and Baum ([19] p 102) show that the GFT, D has $L_\infty(D) = \mathrm{O}(|G|^{3/2})$. For supersolvable groups, including all p-groups, there is a faster algorithm. Baum [6] proves that the GFT, D for supersolvable groups has $L_\infty(D) = \mathrm{O}(|G| \log_2 |G|)$.

3 A model for the real universal Clifford algebras

We now review the well known relationships between models for the real Clifford algebras. GluCat models each real universal Clifford algebra as a vector space of maps from integer sets to real numbers, with a multiplication defined on signed integer sets.

The real universal Clifford algebra $\mathbb{R}_{p,q}$, can also be modelled as a quotient of the group algebra $\mathbb{R}\mathbb{G}_{p,q}$, where the group $\mathbb{G}_{p,q}$ is a 2-group, here called a *real frame group.*

Definition 3.1. *For finite $S \subset \mathbb{Z}\backslash\{0\}$, define the group $\mathbb{G}_S$ via the map $g : S \rightarrow \mathbb{G}_S$ and the power-commutator presentation:*

$$\begin{aligned}\mathbb{G}_S := \Big\langle \mu, g_k \mid k \in S,\ \mu^2 = 1,\ {g_k}^2 = \mu,\ \forall k < 0,\ {g_k}^2 = 1,\ \forall k > 0,\\ [\mu, g_k] = 1,\ \forall k \in S,\ [g_k, g_m] = \mu,\ \forall k \neq m \Big\rangle.\end{aligned}$$

Lemma 3.2. *For finite $S, T \subset \mathbb{Z} \setminus \{0\}$, $\mathbb{G}_S \simeq \mathbb{G}_T$ if and only if $|S_-| = |T_-|$ and $|S_+| = |T_+|$, where $S_- := \{x \in S \mid x < 0\}$ and $S_+ := \{x \in S \mid x > 0\}$.*

Proof. $\mathbb{G}_S$ and $\mathbb{G}_T$ are isomorphic if and only if they have exactly corresponding presentations as per Definition 3.1. ■

We now define $\mathbb{G}_{p,q}$ as a special case of Definition 3.1.

Definition 3.3. *With* $\varsigma(a,b) := \{a, a+1, \ldots, b\} \setminus \{0\}$*, define* $\mathbb{G}_{p,q}$ *as* $\mathbb{G}_{\varsigma(-q,p)}$.

Groups of this type have been extensively studied by Salingaros [60], Braden [13], Lam and Smith [40], Bergdolt [9] and others, but there is no generally accepted name for them.

Each member w of real frame group $\mathbb{G}_S$ can be expressed as the canonically ordered product

$$w = \mu^a \prod_{k \in S}^{p} {g_k}^{b_k}, \quad \text{where } a, b_k \in \mathbb{F}_2 := \{0, 1\}.$$

Each canonically ordered product corresponds to a signed index set, where the index sets are subsets of $\varsigma(-p,q)$. ([44] 21.3, p 282, [58] p 306)

$$(a, B) \cong \mu^a \prod_{k \in S}^{p} {g_k}^{\chi(B)_k} \text{ where } a \in \mathbb{F}_2$$

and $\chi(B)$ is the characteristic function of B.

The real frame group, $\mathbb{G}_{p,q}$, can therefore be represented by a multiplication defined on signed index sets. In other words, the multiplication is defined on $\mathbb{F}_2 \times \mathbb{P}\varsigma(-p,q)$, where $\mathbb{P}\varsigma(-q,p)$ is the power set of $\varsigma(-q,p)$, a set of index sets with cardinality 2^{p+q}. Thus $|\mathbb{G}_{p,q}| = 2^{p+q+1}$.

The framed model $\mathbb{R}^{\mathbb{P}\varsigma(-q,p)}$ of $\mathbb{R}_{p,q}$ is the vector space of maps from $\mathbb{P}\varsigma(-q,p)$ to $\mathbb{R}$, isomorphic to the vector space of 2^{p+q} tuples of real numbers indexed by subsets of $\varsigma(-q,p)$.

The real universal Clifford algebra $\mathbb{R}_{p,q}$ can also be obtained from $\mathbb{G}_{p,q}$, by taking the quotient of the real group algebra $\mathbb{R}\mathbb{G}_{p,q}$, by the two-sided ideal $\langle 1+\mu \rangle$ ([40] pp. 778–779). The ideal $\langle 1+\mu \rangle$ consists of all elements of the form $(1+\mu)a$ with $a \in \mathbb{R}_{p,q}$. We have $(1+\mu)a = a(1+\mu)$ since $1+\mu$ is in the centre of $\mathbb{R}_{p,q}$.

This construction by quotient is equivalent to identifying μ in the group with -1 in $\mathbb{R}$ and defining multiplication on $\mathbb{R}^{\mathbb{P}\varsigma(-q,p)}$ by using the group multiplication, linearity and the distributive rule. Thus $\mathbb{R}_{p,q}$ can be identified with $\mathbb{R}^{\mathbb{P}\varsigma(-q,p)}$, and has real dimension 2^{p+q}.

The basis elements of $\mathbb{R}_{p,q}$ are here denoted by $\mathbf{e}_T$ for $T \subseteq \varsigma(-q,p)$, and the canonical generators are $\mathbf{e}_{\{k\}}$ for $k \in \varsigma(-q,p)$.

4 Real matrix representations of Clifford algebras

For a real universal Clifford algebra, we use the term *matrix representation* in the sense of Definition 2.1 to mean an algebra homomorphism from the Clifford algebra to a real matrix algebra.

Definition 4.1. *A* real matrix representation *of a finite dimensional algebra A over $\mathbb{R}$ is an algebra homomorphism from A to a real matrix algebra.*

GluCat implements real matrix representations of Clifford algebras, based on the constructions in Porteous [56], which build on those in [3]. For the real universal Clifford algebra, $\mathbb{R}_{p,q}$, the matrix representation $P_{p,q}$ implemented in GluCat is a minimum degree faithful real matrix representation [36] [54].

Definition 4.2.

$$M(p,q) = \begin{cases} \lceil \frac{p+q}{2} \rceil + 1, & \textit{if } q-p \equiv 2,3,4 \pmod 8, \\ \lceil \frac{p+q}{2} \rceil, & \textit{otherwise.} \end{cases}$$

Theorem 4.3. *(Porteous [56] Prop. 10.46, p 192, Chapter 13)*
The degree N, of any faithful real matrix representation $R : \mathbb{R}_{p,q} \to \mathbb{R}(N)$, must have $N \geqslant 2^{M(p,q)}$ with $M(p,q)$ as per Definition 4.2. This bound is attained, that is, there is a faithful real matrix representation R, of $\mathbb{R}_{p,q}$, such that $R : \mathbb{R}_{p,q} \to \mathbb{R}(2^{M(p,q)})$.

Proof. The existence of a faithful real matrix representation of $\mathbb{R}_{p,q}$ of degree $2^{M(p,q)}$ is given by the construction in Definition 4.14 below.

That $2^{M(p,q)}$ is the minimum degree for a faithful real matrix representation of $\mathbb{R}_{p,q}$ is a consequence of the isomorphism theorems of Porteous ([56] Propositions 13.12, 13.17, 13.20, 13.22 and Corollaries 13.24 and 13.25) as illustrated by [56] Table 13.26, p 250 and [57] Table 15.27, p 133. These isomorphisms give the minimum degree for a faithful representation of $\mathbb{R}_{p,q}$ using matrices over one of the rings $\mathbb{R}, {}^2\mathbb{R}, \mathbb{C}, \mathbb{H}$ or ${}^2\mathbb{H}$. These are tabulated in Hile and Lounesto, [36], p 54, with $n = p+q$.

$$\mathbb{R}_{p,q} \simeq \begin{cases} \mathbb{R}(2^{n/2}), & \text{if } q-p \equiv 0,6 \pmod 8, \\ \mathbb{C}(2^{(n-1)/2}), & \text{if } q-p \equiv 1,5 \pmod 8, \\ \mathbb{H}(2^{(n-2)/2}), & \text{if } q-p \equiv 2,4 \pmod 8, \\ {}^2\mathbb{H}(2^{(n-3)/2}), & \text{if } q-p \equiv 3 \pmod 8, \\ {}^2\mathbb{R}(2^{(n-1)/2}), & \text{if } q-p \equiv 7 \pmod 8. \end{cases}$$

In turn, Porteous [56] Proposition 10.46 gives that the minimum degree for a faithful real matrix representation of one of these rings is: for ${}^2\mathbb{R}$, 2; for $\mathbb{C}$, 2; for $\mathbb{H}$, 4; and for ${}^2\mathbb{H}$, 8. ■

Injection of $\mathbb{R}_{p,q}$ into $\mathbb{R}_{m,m}$. The construction of a faithful real matrix representation of $\mathbb{R}_{p,q}$ can be broken down into two cases, 1) $p \neq q$ and 2) $p = q = m$. For $p \neq q$, the construction can be done in two steps. The first step is to construct an algebra injection from $\mathbb{R}_{p,q}$ to $\mathbb{R}_{m,m}$, where $m = M(p,q)$. The second step is the construction of a representation of $\mathbb{R}_{m,m}$. The first step is described here.

Definition 4.4. *For $p \neq q$ we define the algebra injection $\Upsilon_{p,q} : \mathbb{R}_{p,q} \to \mathbb{R}_{m,m}$, where $m = M(p,q)$ by*

$$\Upsilon_{p,q} := \begin{cases} \Upsilon_{p-4,q+4} \circ \alpha_{p,q}, & \textit{if} q-p \equiv 0,6 \pmod 8 \; \textit{and} q-p < -4, \\ \Upsilon_{p+4,q-4} \circ \beta_{p,q}, & \textit{if} q-p \equiv 0,6 \pmod 8 \; \textit{and} q-p > 3, \\ \gamma_{p,q}, & \textit{if} q-p = -2, \\ \Upsilon_{r(p,q),s(p,q)} \circ \iota_{p,q}, & \textit{otherwise,} \end{cases}$$

where $\alpha_{p,q}, \beta_{p,q}, \gamma_{p,q}$ and $\iota_{p,q}$ are algebra homomorphisms, defined on generators as follows.

$$\begin{aligned}
\alpha_{p,q} &: \mathbb{R}_{p,q} \to \mathbb{R}_{p-4,q+4}, \\
\alpha_{p,q}\, \mathbf{e}_{\{p-k\}} &:= \mathbf{e}_{\{-q-k-1\}}\, \mathbf{e}_{\{-q-4,-q-3,-q-2,-q-1\}}, \ \textit{for}\ k = 0,1,2,3, \\
\alpha_{p,q}\, \mathbf{e}_{\{j\}} &:= \mathbf{e}_{\{j\}}, \ \textit{otherwise},
\end{aligned}$$

$$\begin{aligned}
\beta_{p,q} &: \mathbb{R}_{p,q} \to \mathbb{R}_{p+4,q-4}, \\
\beta_{p,q}\, \mathbf{e}_{\{-q+k\}} &:= \mathbf{e}_{\{p+k+1\}}\, \mathbf{e}_{\{p+1,p+2,p+3,p+4\}}, \ \textit{for}\ k = 0,1,2,3, \\
\beta_{p,q}\, \mathbf{e}_{\{j\}} &:= \mathbf{e}_{\{j\}}, \ \textit{otherwise},
\end{aligned}$$

$$\begin{aligned}
\gamma_{p,q} &: \mathbb{R}_{p,q} \to \mathbb{R}_{q+1,p-1}, \\
\gamma_{p,q}\, \mathbf{e}_{\{k\}} &:= \mathbf{e}_{\{-k,q+1\}}, \ k \neq p, \\
\gamma_{p,q}\, \mathbf{e}_{\{p\}} &:= \mathbf{e}_{\{q+1\}},
\end{aligned}$$

$$\begin{aligned}
\iota_{p,q} &: \mathbb{R}_{p,q} \to \mathbb{R}_{r(p,q),s(p,q)}, \\
\iota_{p,q}\, \mathbf{e}_{\{k\}} &:= \mathbf{e}_{\{k\}}, \ \textit{where} \\
r(p,q) &:= \begin{cases} p+k, & \textit{if}\ q-p \equiv k \pmod 8,\ \textit{for}\ k=1,2,3, \\ p, & \textit{otherwise}, \end{cases} \\
s(p,q) &:= \begin{cases} q+1, & \textit{if}\ q-p \equiv 5,7 \pmod 8, \\ q+2, & \textit{if}\ q-p \equiv 4 \pmod 8, \\ q, & \textit{otherwise}. \end{cases}
\end{aligned}$$

Lemma 4.5.

$$\begin{aligned}
&s(p,q) - r(p,q) \equiv 0 \ \textit{or}\ 6 \pmod 8 \ \textit{and} \\
&\qquad M(p,q) = M(r(p,q), s(p,q)) = \frac{r(p,q)+s(p,q)}{2}.
\end{aligned}$$

Proof. Tabulate for each value of $q-p \pmod 8$. ■

Lemma 4.6. *$\iota_{p,q}$ is an algebra injection.*

Proof. The notation of the framed model makes this obvious. ■

Lemma 4.7. *$\alpha_{p,q}, \beta_{p,q}$ and $\gamma_{p,q}$ are algebra isomorphisms.*

When $\alpha_{p,q}, \beta_{p,q}$ and $\gamma_{p,q}$ are restricted to the signed basis elements of $\mathbb{R}_{p,q}$, each becomes a group isomorphism.

Proof. For $\alpha_{p,q}$ and $\beta_{p,q}$, this follows from Porteous Prop 13.23 ([56] p 248) and Lounesto 16.4 ([44] p 216).

For $\gamma_{p,q}$, this follows from Porteous Prop 13.20 ([56] p 248) and Lounesto 16.3 ([44] p 215). ■

Lemma 4.8. *$\Upsilon_{p,q}$ is an algebra injection, and for $q-p \equiv 0,6 \pmod 8$, $\Upsilon_{p,q}$ is an isomorphism.*

Proof. This follows from Lemmas 4.5, 4.6 and 4.7. ■

The Kronecker product. To complete the construction of the real matrix representation $\mathbb{R}_{p,q}$, and for what follows, we need the Kronecker matrix product.

Definition 4.9. *If $A \in \mathbb{R}(r)$ and $B \in \mathbb{R}(s)$, then*

$$(A \otimes B)_{j,k} := A_{j,k} B,$$

if $A \otimes B$ is treated as an $r \times r$ block matrix with $s \times s$ blocks.

A well known property of the Kronecker product is:

Lemma 4.10. *If $A, C \in \mathbb{R}(r)$ and $B, D \in \mathbb{R}(s)$, then $(A \otimes B)(C \otimes D) = AC \otimes BD$.*

An orthonormal anticommuting generating set for $\mathbb{R}(2^m)$.

Definition 4.11. *(Porteous [56], pp. 242–243)*

Given $\mathbb{A}$, a real associative algebra with unit, the finite set $S \subset \mathbb{A}$ is an orthonormal anticommuting set *for A if and only if*

- *S is linearly independent,*
- *each $x \in S$ has $x^2 = 0, 1$ or -1, and*
- *the elements of S anticommute in pairs.*

If, addition, $|S| = p + q$,

$$\begin{aligned}\Big|\{x \in S \mid x^2 = 1\}\Big| &= p \text{ and}\\ \Big|\{x \in S \mid x^2 = -1\}\Big| &= q,\end{aligned}$$

then S is called an orthonormal anticommuting set of type (p, q) for $\mathbb{A}$

Definition 4.12. *Here and in what follows, define:*

$$\begin{aligned} I_n &:= \text{ unit matrix of dimension } 2^n, \ I := I_1, \\ J &:= \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}, \ K := \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}. \end{aligned}$$

Lemma 4.13. *([56] Proposition 13.17, p 247)*

If S is an orthonormal anticommuting set of type $(m-1, m-1)$ for $\mathbb{R}(2^{m-1})$, which generates $\mathbb{R}(2^{m-1})$ as an algebra, then $\{-JK \otimes A \mid A \in S\} \cup \{J \otimes I_{m-1}, K \otimes I_{m-1}\}$ is an orthonormal anticommuting set of type (m, m) for $\mathbb{R}(2^m)$, which generates $\mathbb{R}(2^m)$ as an algebra.

Remarks. Braden ([13] Lemma 7, p 617) gives an equivalent construction using induced complex representations.

Definition of the real matrix representation of $\mathbb{R}_{p,q}$.

Definition 4.14. *We here construct the representation $P_{p,q} : \mathbb{R}_{p,q} \to \mathbb{R}(2^{M(p,q)})$. First, abbreviate $P_{m,m}$ as P_m.*

For $p \neq q$, define the real matrix representation of each generator $\mathbf{e}_{\{k\}} \in P_{p,q}$, by

$$P_{p,q}(\mathbf{e}_{\{k\}}) := P_m \circ \Upsilon_{p,q}(\mathbf{e}_{\{k\}}),$$

with $\Upsilon_{p,q}$ as per Definition 4.4.

For $m > 0$, use Lemma 4.13 to recursively define the real matrix representation of each generator of $\mathbb{R}_{m,m}$

$$\begin{aligned} &P_m\,\mathbf{e}_{\{-m\}} := J \otimes I_{m-1}, \quad P_m\,\mathbf{e}_{\{m\}} := K \otimes I_{m-1}, \\ &\textit{for} \quad -m < k < m, \quad P_m\,\mathbf{e}_{\{k\}} := -JK \otimes P_{m-1}\,\mathbf{e}_{\{k\}}. \end{aligned}$$

We can now make $P_{p,q} : \mathbb{R}_{p,q} \to \mathbb{R}(2^m)$, into an algebra homomorphism by defining

$$\begin{aligned} &P_{0,0}(x) := [x] \in \mathbb{R}(1), \quad P_{p,q}\,\mathbf{e}_T := \prod_{k \in T} P_{p,q}\,\mathbf{e}_{\{k\}}, \ \textit{and} \\ &P_{p,q}(x) := \sum_{T \subseteq \varsigma(-q,p)} x_T P_{p,q}\,\mathbf{e}_T, \quad \textit{for}\, x = \sum_{T \subseteq \varsigma(-q,p)} a_T\,\mathbf{e}_T. \end{aligned}$$

Lemma 4.15. *Each basis matrix $P_m(\mathbf{e}_T)$, is* monomial*, having one non-zero in each row and each column ([19] p 52), and each non-zero is -1 or 1.*

Proof. By induction. Note that I, J and K have this property. Now verify that the matrix product and the Kronecker product preserve this property, ie. if both operands have this property, so does the result. Finally, note that each basis matrix is the result of a sequence of matrix and Kronecker products starting with I, J and K. ■

Lemma 4.16. $P_{p,q} = P_m \circ \Upsilon_{p,q}$.

Proof. By definition, the left hand side and right hand side agree on generators of $\mathbb{R}_{p,q}$. Now note that $P_{p,q}$ and P_m are defined as algebra homomorphisms, and by Lemma 4.8, $\Upsilon_{p,q}$ is an algebra injection. ■

Theorem 4.17. *$P_{p,q}$ as per Definition 4.14 is a minimum degree faithful real matrix representation of $\mathbb{R}_{p,q}$.*

Proof. Since by Lemma 4.16, $P_{p,q} = P_m \circ \Upsilon_{p,q}$, and by Lemma $\Upsilon_{p,q}$ is an algebra injection, all that is left to verify is that P_m is an algebra isomorphism. This follows from Porteous Prop. 13.17 and Corollary 13.18 ([56] p 247). ■

Bound for 2-linear complexity of the real matrix representation.

Theorem 4.18. *$L_2(P_m)$ is bounded by $d^{3/2}$, where d is the dimension of $\mathbb{R}(2^m) \cong \mathbb{R}_{m,m}$.*

Proof. Since $P_m\,\mathbf{e}_T$ is of size $2^m \times 2^m$ and is monomial, it has 2^m non-zeros.

$\mathbb{R}(2^m)$ has 4^m basis elements.

$\mathbb{A}(P_m)$ is therefore bounded by

$$4^m \times 2^m = (4^m)^{3/2} = d^{3/2},$$

where d is the dimension of $\mathbb{R}(2^m) \cong \mathbb{R}_{m,m}$. There are no non-trivial multiplications.

■

5 Fast real matrix representations of Clifford algebras

Clifford algebras and supersolvable groups. Since $\mathbb{G}_{p,q}$ is a 2-group, it is supersolvable ([19] p 109). The real matrix representation of Clifford algebras is therefore related to the GFT for supersolvable groups:

$$\begin{tikzcd}
\mathbb{C}\mathbb{G}_{p,q} \arrow[r, "D"] \arrow[d, "\text{project}"'] & D\left(\mathbb{C}\mathbb{G}_{p,q}\right) \subseteq \mathbb{C}(N) \arrow[d, "\text{project}"] \\
\mathbb{R}\mathbb{G}_{p,q} \arrow[r, "D"] \arrow[d, "\text{quotient}"'] & D\left(\mathbb{R}\mathbb{G}_{p,q}\right) \subseteq \mathbb{C}(N) \arrow[d, "\text{quotient}"] \\
\mathbb{R}_{p,q} \arrow[r, "P_{p,q}"'] & P_{p,q}\left(\mathbb{R}_{p,q}\right) \subseteq \mathbb{R}(2^{M(p,q)})
\end{tikzcd}$$

The GFT for $\mathbb{G}_{p,q}$ maps from the complex group algebra $\mathbb{C}\mathbb{G}_{p,q}$ to a suitable complex matrix algebra.

$$D : \mathbb{C}\mathbb{G}_{p,q} \to \mathbb{C}(N)$$

As a real algebra, the group algebra $\mathbb{C}\mathbb{G}_{p,q}$ has dimension four times that of the real Clifford algebra $\mathbb{R}_{p,q}$. One factor of two comes from $|\mathbb{C}/\mathbb{R}|$, the other factor comes from $|\mathbb{G}_{p,q}| \,/\, |\mathbb{P}\varsigma(-q,p)|$.

A fast real matrix representation of the neutral Clifford algebra $\mathbb{R}_{m,m}$. The neutral frame group $\mathbb{G}_{m,m}$ is an extraspecial 2-group $\mathbb{G}_{m,m} \cong D_4^{(m)}$, where D_4 is the dihedral group of order 8, $G^{(m)} := G \circ G \circ \ldots \circ G$ (m times), and $\circ$ is the central product of groups. $|\mathbb{G}_{m,m}| = 2^{2m+1}$ [13] [40].

For $\mathbb{R}_{m,m}$ we would expect $L_\infty(P_m) = \mathrm{O}(m4^m)$ this way:

$$\begin{tikzcd}
\mathbb{R}_{m,m} \arrow[r, "P_m"] \arrow[d] & \mathbb{R}(2^m) \\
\mathbb{C}\mathbb{G}_{m,m} \arrow[r, "D"'] & D\left(\mathbb{C}\mathbb{G}_{m,m}\right) \arrow[u]
\end{tikzcd}$$

but there are also explicit fast algorithms for both the real matrix representation and its inverse, with $L_2(P_m) = \mathrm{O}(m4^m)$ and $L_2(P_m^{-1}) = \mathrm{O}(m4^m)$, which do not involve the group algebra $\mathbb{C}\mathbb{G}_{m,m}$.

$\mathbb{Z}_2$ grading. The fast algorithms for the representation of $\mathbb{R}_{p,q}$ take advantage of $\mathbb{Z}_2$-grading.

The algebras $\mathbb{R}_{p,q}$ are $\mathbb{Z}_2$-*graded* ([3], p 5, [39] Chapter 4, p 76, [5] 166). Each $x \in \mathbb{R}_{p,q}$ can be split into odd and even parts, $x = x^+ + x-$, with $\mathit{odd} \times \mathit{odd} = \mathit{even}$, etc. Scalars are even and the generators are odd.

We can express $P_{p,q}$ in terms of its actions on the even and odd parts of a multivector: $P_{p,q}(x) = P_{p,q}(x^+) + P_{p,q}(x^-)$, for $x \in \mathbb{R}_{p,q}$.

Lemma 5.1. *For $m > 0$, for all $a \in \mathbb{R}_{m-1}$, we have*

$$P_m(a^+) = I \otimes P_{m-1}(a^+), \; P_m(a^-) = -JK \otimes P_{m-1}(a^-).$$

Proof. We know a^+ is a sum of even terms. Since P_{m-1} is an isomorphism, we need only deal with the product of two generators. By Lemma 4.10,

$$\big(-JK \otimes P_{m-1}(\mathbf{e}_{\{j\}})\big)\,\big(-JK \otimes P_{m-1}(\mathbf{e}_{\{k\}})\big) = I \otimes P_{m-1}(\mathbf{e}_{\{j\}}\,\mathbf{e}_{\{k\}}).$$

The result for a^- follows immediately. ■

Recursive expressions for P_m.

Theorem 5.2. *(Cnops [22])*

For $m > 0$, for the real matrix representation P_m as per Definition 4.14, for $x \in \mathbb{R}_{m,m}$, with

$x = a + b\,\mathbf{e}_- + c\,\mathbf{e}_+ + d\,\mathbf{e}_-\,\mathbf{e}_+,\ \ \mathbf{e}_- := \mathbf{e}_{\{-m\}},\ \ \mathbf{e}_+ := \mathbf{e}_{\{m\}},\ a, b, c, d \in \mathbb{R}_{m-1,m-1},$ *we have*

$$\begin{aligned}
P_m(x^+) &= I \otimes A^+ - K \otimes B^- - J \otimes C^- + JK \otimes D^+,\\
P_m(x^-) &= -JK \otimes A^- + J \otimes B^+ + K \otimes C^+ - I \otimes D^-,\\
P_m(x) &= \begin{bmatrix} A - D & -B + C \\ \widehat{B} + \widehat{C} & \widehat{A} + \widehat{D} \end{bmatrix},
\end{aligned}$$

where

$$\begin{aligned}
A &:= P_{m-1}(a), \quad B := P_{m-1}(b), \quad C := P_{m-1}(c),\ D := P_{m-1}(d),\\
A^+ &:= P_{m-1}(a^+),\ A^- := P_{m-1}(a^-),\ \widehat{A} := P_{m-1}(\widehat{a}),\ \textit{etc.}
\end{aligned}$$

Proof. First, split $x \in \mathbb{R}_{m,m}$ into components with respect to $\mathbf{e}_-$ and $\mathbf{e}_+$ and then split each component into its even and odd parts.

$$\begin{aligned}
x &= a + b\,\mathbf{e}_- + c\,\mathbf{e}_+ + d\,\mathbf{e}_-\,\mathbf{e}_+,\\
x^+ &= a^+ + b^-\,\mathbf{e}_- + c^-\,\mathbf{e}_+ + d^+\,\mathbf{e}_-\,\mathbf{e}_+,\\
x^- &= a^- + b^+\,\mathbf{e}_- + c^+\,\mathbf{e}_+ + d^-\,\mathbf{e}_-\,\mathbf{e}_+.
\end{aligned}$$

We have, from Definition 4.14, and by Lemmas 4.10 and 5.1:

$$\begin{aligned}
P_m(x^+) &= P_{m-1}(a^+) + P_{m-1}(b^-)P_{m-1}(\mathbf{e}_-)\\
&\quad + P_{m-1}(c^-)P_{m-1}(\mathbf{e}_+) + P_{m-1}(d^+)P_{m-1}(\mathbf{e}_-\,\mathbf{e}_+)\\
&= I \otimes A^+ + (-JK \otimes B^-)(J \otimes I_{m-1})\\
&\quad + (-JK \otimes C^-)(K \otimes I_{m-1}) + (I \otimes D^+)(JK \otimes I_{m-1})\\
&= I \otimes A^+ - K \otimes B^- - J \otimes C^- + JK \otimes D^+.
\end{aligned}$$

Similarly,

$$\begin{aligned}
P_m(x^-) &= -JK \otimes A^- + J \otimes B^+ + K \otimes C^+ - I \otimes D^-, \text{ therefore}\\
P_m(x) &= P_m(x^+) + P_m(x^-)\\
&= I \otimes (A^+ - D^-) + K \otimes (C^+ - B^-)\\
&\quad + J \otimes (B^+ - C^-) + JK \otimes (D^+ - A^-)\\
&= \begin{bmatrix} A - D & -B + C \\ \widehat{B} + \widehat{C} & \widehat{A} + \widehat{D} \end{bmatrix}.
\end{aligned}$$

■

Remarks. This recursive expression for P_m is equivalent to that in Cnops [22]. Cnops credits Porteous [56], but the expression does not appear there.

GluCat actually uses another equivalent recursive expression, which has a similar proof:

Corollary 5.3. *If* $x := a + \mathbf{e}_- b + c\,\mathbf{e}_+ + \mathbf{e}_- d\,\mathbf{e}_+$, *then*

$$\begin{aligned} P_m(x^+) &= I \otimes A^+ + K \otimes B^- - J \otimes C^- + JK \otimes D^+ \\ P_m(x^-) &= -JK \otimes A^- + J \otimes B^+ + K \otimes C^+ + I \otimes D^-. \end{aligned}$$

This expression is less expensive for GluCat to evaluate because in GluCat it takes less operations to split x in this way. Also, x is split into its even and odd parts, only once, at the top level of recursion. Each lower level deals with either an even or an odd multivector, and unlike the Cnops expression, each level does not need a grade involution.

The linear complexity of $P_{p,q}$.

Theorem 5.4. *For* $m \geqslant 0$,

$$L_2(P_m) \leqslant m4^m = \frac{1}{2} d \log_2 d,$$

where $d = 4^m$ *is the dimension of* $\mathbb{R}_{m,m}$.

Proof. In the matrix expression for $P_m x$ from Theorem 5.2, if we count non-zero additions at each level of recursion, we obtain at most 4^m additions at each of m levels. So $\mathbb{A}(P_m) \leqslant m4^m$.

There are no non-trivial multiplications, so the result follows. ∎

Lemma 5.5. *For* $\Upsilon_{p,q}$ *as per Definition 4.4,* $L_2(\Upsilon_{p,q}) = 0$.

Proof. By 4.8 $\Upsilon_{p,q}$ is an algebra injection. By definition, $\Upsilon_{p,q}$ maps generators in $\mathbb{R}_{p,q}$ to signed basis elements in $\mathbb{R}_{m,m}$, and so is a one-one mapping between signed basis elements. Thus there are no non-zero additions and the only multiplications are by 1 and -1. ∎

Theorem 5.6. *For* $p, q \geqslant 0$, $L_2(P_{p,q}) \leqslant m4^m \leqslant 4d(\log_2 d + 3)$, *where* $m = M(p,q)$ *and* $d = 2^{p+q}$ *is the dimension of* $\mathbb{R}_{p,q}$.

Proof. By Lemmas 5.4 and 5.5, $L_2(P_{p,q}) \leqslant m4^m$, where $m = M(p,q)$.

Since $m = M(p,q) \leqslant (p+q+3)/2$, we have

$$L_2(P_{p,q}) \leqslant \frac{p+q+3}{2} 2^{p+q+3} = 4(p+q+3)2^{p+q} = 4d(\log_2 d + 3).$$

∎

To prove a similar bound for 2-linear complexity of the expressions for $P_m(x^+)$ and $P_m(x^-)$ from Corollary 5.3, we first need some technical lemmas.

Lemma 5.7. *If $x \in \mathbb{R}_{m,m}$ for $m > 0$ then $X^+ := P_m(x^+)$ and $X^- := P_m(x^-)$, have no non-zero entries in common:*

$$X^+_{j,k} X^-_{j,k} = 0 \textit{ for all } 1 \leqslant j,k \leqslant 2^m.$$

Proof. By induction. Examine the expressions from Corollary 5.3.

$$\begin{aligned} P_m(x^+) &= I \otimes A^+ + K \otimes B^- - J \otimes C^- + JK \otimes D^+ \\ &= \begin{bmatrix} (A-D)^+ & (B+C)^- \\ (B-C)^- & (A+D)^+ \end{bmatrix}, \\ P_m(x^-) &= -JK \otimes A^- + J \otimes B^+ + K \otimes C^+ + I \otimes D^- \\ &= \begin{bmatrix} (A-D)^- & (B+C)^+ \\ (B-C)^+ & (A+D)^- \end{bmatrix}. \end{aligned}$$

If the lemma is true for $m-1$, then each pair $(A-D)^+$, $(A-D)^-$ have no non-zero entries in common, and therefore X^+ and X^- have non non-zero entries in common. Now note that if x in $\mathbb{R}_{0,0}$ then $x^- = 0$.

Therefore if x in $\mathbb{R}_{1,1}$ then $(A-D)^- = (B-C)^- = (A+D)^- = (B+C)^- = 0$. ■

Remarks. This corresponds to the *checkerboard grading* of Lam ([39] p 81).

Corollary 5.8. *If $x \in \mathbb{R}_{m,m}$ for $m > 0$ then*

$$\operatorname{nnz}(P_m x^\pm) \leqslant \frac{1}{2} 4^m = 2^{2m-1}.$$

Theorem 5.9. *Define P_m^+ and P_m^- by $P_m^+(x) = P_m(x^+)$, $P_m^-(x) = P_m(x^-)$. Then, for $m > 0$,*

$$L_2(P_m^+ + P_m^-) \leqslant m4^m = \frac{1}{2} d \log_2 d,$$

where $d = 4^m$ is the dimension of $\mathbb{R}_{m,m}$.

Proof. For $P_1^\pm$, we have $\mathbb{A}(P_1^\pm) \leqslant 2$, since $A^- = B^- = C^- = D^- = 0$.

For $m > 1$ we examine the recursive expression for P_m^+ from Corollary 5.3.

$$P_m x^+ = \begin{bmatrix} A^+ - D^+ & B^- + C^- \\ B^- - C^- & A^+ + D^+ \end{bmatrix},$$

By Corollary 5.8, A^+ etc. have at most $\frac{1}{2}4^{m-1}$ non-zero entries.

$$\begin{aligned} \operatorname{nnz}(A^+) &\leqslant \frac{1}{2} 4^{m-1}, \text{ etc., so} \\ \mathbb{A}(P_m^+) &= 4 \times \frac{1}{2} 4^{m-1} + 2 \times \mathbb{A}(P_{m-1}^+) + 2 \times \mathbb{A}(P_{m-1}^-) \\ &\leqslant \frac{1}{2} m 4^m. \\ \text{Similarly, } \mathbb{A}(P_m^-) &\leqslant \frac{1}{2} m 4^m. \end{aligned}$$

Now, by Lemma 5.7, $P_m^+(x)$ and $P_m^-(x)$ have no non-zero entries in common, so $\mathbb{A}(P_m^+ + P_m^-) \leqslant m4^m$. There are no non-trivial multiplications, so the result follows. ■

6 Inverse real matrix representations of Clifford algebras

Since $P_{p,q}$ is an algebra injection, it has an inverse. It is convenient to extend the definition of this inverse function from $P_{p,q}(\mathbb{R}_{p,q})$ to the whole of $\mathbb{R}(2^m)$. GluCat uses the following definition.

Definition 6.1. *Define $Q_m := Q_{m,m} = P_m^{-1}$. For $p \neq q$, define $Q_{p,q}$ by*

$$Q_{p,q} : \mathbb{R}(2^m) \to \mathbb{R}_{p,q}, \text{ where } m = M(p,q) \text{ as per Definition 4.2,}$$
$$Q_{p,q} := \begin{cases} P_{p,q}^{-1}, & \text{if } q - p \equiv 0, 6 \pmod 8, \\ \pi_{p,q} \circ Q_{r(p,q),s(p,q)}, & \text{otherwise,} \end{cases}$$

where $\pi_{p,q}$ is an algebra projection defined by

$$\pi_{p,q} : \mathbb{R}_{r(p,q),s(p,q)} \to \mathbb{R}_{p,q}, \text{ with } r \text{ and } s \text{ as per Definition 4.4,}$$
$$\pi_{p,q}\left(\mathbf{e}_{\{k\}}\right) := \begin{cases} \mathbf{e}_{\{k\}}, & \text{if } -q < k < p \text{ and } k \neq 0, \\ 0, & \text{otherwise.} \end{cases}$$

One way to compute the inverse of the real matrix representation $P_{p,q}$ is to use the inner products described below.

The real framed inner product. Recall that if $x \in \mathbb{R}_{p,q}$, then x can be expressed as

$$x = \sum_{T \subseteq \varsigma(-q,p)} x_T \, \mathbf{e}_T$$

The basis $\{\, \mathbf{e}_T \mid T \subseteq \varsigma(-q,p)\}$ is orthonormal with respect to the real framed inner product

$$a \bullet b := \sum_{T \subseteq \varsigma(-q,p)} a_T b_T.$$

We have $\mathbf{e}_S \bullet \mathbf{e}_T = \delta_{S,T}$ and $a_T = a \bullet \mathbf{e}_T$.

The normalized Frobenius inner product. Since the real matrix representation $P_{p,q}$ is an isomorphism, it preserves the real framed inner product. That is, there is an inner product

$$\bullet : P_{p,q}(\mathbb{R}_{p,q}) \times P_{p,q}(\mathbb{R}_{p,q}) \to \mathbb{R}, \text{ with } P_{p,q}(\mathbb{R}_{p,q}) \subseteq \mathbb{R}(2^m), m = M(p,q),$$

such that, for $a, b \in \mathbb{R}_{p,q}$,

$$P_{p,q}(a) \bullet P_{p,q}(b) = a \bullet b, \text{ so } P_{p,q}(a) \bullet P_{p,q}\, \mathbf{e}_T = a \bullet \mathbf{e}_T = a_T.$$

We will call this the *normalized Frobenius* inner product.

Lemma 6.2. *The normalized Frobenius inner product*

$$\begin{aligned} A \bullet B &:= 2^{-m} \operatorname{tr} A^T B \\ &= 2^{-m} \sum_{j,k=1}^{2^m} A_{j,k} B_{j,k}, \text{ for } A, B \in \mathbb{R}(2^m), \end{aligned}$$

satisfies

$$P_{p,q}(x) \bullet P_{p,q}(x') = x \bullet x', \text{ for } x, x' \in \mathbb{R}_{p,q}.$$

Proof. For P_m, we prove this by induction on m. The lemma is trivially true for $m = 0$. For $m > 0$, we assume the lemma is true for $m - 1$. Using Theorem 5.2, for $x, x' \in \mathbb{R}_{m,m}$, with $x = a + b\,\mathbf{e}_- + c\,\mathbf{e}_+ + d\,\mathbf{e}_-\,\mathbf{e}_+$, $x' = a' + b'\,\mathbf{e}_- + c'\,\mathbf{e}_+ + d'\,\mathbf{e}_-\,\mathbf{e}_+$, $\mathbf{e}_- := \mathbf{e}_{\{-m\}}$, $\mathbf{e}_+ := \mathbf{e}_{\{m\}}$, $a, a', b, b'c, c', d, d' \in \mathbb{R}_{m-1,m-1}$, we have

$$\begin{aligned} P_m(x) \bullet P_m(x') &= \frac{1}{2}\Big((A - D) \bullet (A' - D') + (\widehat{A} + \widehat{D}) \bullet (\widehat{A}' + \widehat{D}')\Big) \\ &\quad + \frac{1}{2}\Big((C - B) \bullet (C' - B') + (\widehat{C} + \widehat{B}) \bullet (\widehat{C}' + \widehat{B}')\Big) \\ &= A \bullet A' + B \bullet B' + C \bullet C' + D \bullet D' \\ &= a \bullet a' + b \bullet b' + c \bullet c' + d \bullet d' \\ &= x \bullet x', \text{ where} \end{aligned}$$

$$\begin{aligned} &A := P_{m-1}(a),\ B := P_{m-1}(b),\ C := P_{m-1}(c),\ D := P_{m-1}(d), \\ &\widehat{A} := P_{m-1}(\widehat{a}), \text{ etc.} \end{aligned}$$

For general (p, q) we note that $\Upsilon_{p,q}$ also preserves the inner product $\bullet$. ■

A naive algorithm for the inverse real matrix representation. The following theorem shows that we can use the normalized Frobenius inner product to compute the inverse real matrix representation.

Theorem 6.3. *The inverse real matrix representation, $Q_{p,q}$ satisfies, for $X \in \mathbb{R}(2^m)$, $T \subseteq \varsigma(-q, p)$,*

$$(Q_{p,q}X)_T = X \bullet P_{p,q}(\mathbf{e}_T).$$

Proof. This follows from Lemma 6.2, since $P_{p,q}(x) \bullet P_{p,q}(\mathbf{e}_T) = x_T$. ■

The naive algorithm for $Q_{p,q}$ evaluates $X \bullet P_{p,q}(\mathbf{e}_T)$ for each $T \subseteq \varsigma(-q, p)$.

Bound for the 2-linear complexity of the inverse real matrix representation.

Theorem 6.4. $L_2(Q_m) \leqslant d^{3/2} + d$, *where* $d = 4^m$.

Proof. Since each $P_m\,\mathbf{e}_T$ is monomial, with $\operatorname{nnz}(P_m\,\mathbf{e}_T) = 2^m$, and there are 4^m subsets $T \subseteq \varsigma(-m, m)$, the number of non-zero additions $\mathbb{A}(Q_m)$ is bounded by $(2^m - 1)4^m \leqslant d^{3/2}$, where $d = 4^m$. Therefore $\mathbb{A}(Q_m) \leqslant 2^m \times 4^m$. The naive algorithm also needs at most 4^m divisions by 2^m. ■

7 Fast inverse real matrix representations of Clifford algebras

The Cnops recursive expression for Q_m.

Theorem 7.1. *(Cnops [22]) For $m > 0, X \in\subseteq \mathbb{R}(2^m)$ and $Q_{p,q}$ as per Definition 6.1,*

$$Q_m(X) = \frac{1}{2}\Big(\widehat{x_{22}} + x_{11} + (\widehat{x_{21}} - x_{12})\,\mathbf{e}_- \\ + (\widehat{x_{21}} + x_{12})\,\mathbf{e}_+ + (\widehat{x_{22}} - x_{11})\,\mathbf{e}_-\,\mathbf{e}_+ \Big),$$

where

$$\mathbf{e}_- := \mathbf{e}_{\{-m\}},\ \mathbf{e}_+ := \mathbf{e}_{\{m\}},$$
$$X = \begin{bmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{bmatrix},\ x_{11} := Q_{m-1}(X_{11}),\ etc.$$

Proof. From Theorem 5.2 above, for $m > 0$ and $x \in \mathbb{R}_{m,m}$, if $x = a + b\,\mathbf{e}_- + c\,\mathbf{e}_+ + d\,\mathbf{e}_-\,\mathbf{e}_+$, then

$$P_m x = \begin{bmatrix} A - D & -B + C \\ \widehat{B} + \widehat{C} & \widehat{A} + \widehat{D} \end{bmatrix},$$

where A, B, etc. are as per Theorem 5.2. Therefore, for

$$X_{11} := A - D, \qquad X_{12} := -B + C,$$
$$X_{21} := \widehat{B} + \widehat{C}, \qquad X_{22} := \widehat{A} + \widehat{D},$$

we have

$$\widehat{X_{22}} + X_{11} = 2A, \qquad \widehat{X_{21}} - X_{12} = 2B,$$
$$\widehat{X_{21}} + X_{12} = 2C, \qquad \widehat{X_{22}} - X_{11} = 2D.$$

■

Remarks. As per Theorem 5.2, this recursive expression for Q_m is equivalent to the expression in Cnops [22], and there is no similar expression in Porteous [56].

This recursive expression uses division by two at each level of recursion. A more efficient algorithm delays these divisions to the top level of recursion. See Theorem 7.8 below.

GluCat uses a different recursive expression which has slightly better floating point accuracy. To properly describe it, we first need to introduce a binary operation related to the Kronecker product, and prove a few technical lemmas.

The left Kronecker quotient. The *left Kronecker quotient* is a binary operation which is an inverse operation to the Kronecker matrix product.

Definition 7.2. *The left Kronecker quotient* $\obslash$ *is defined by*

$$\begin{aligned} &\obslash : \mathbb{R}(r) \times \mathbb{R}(rs) \to \mathbb{R}(s), \ \textit{for } A \in \mathbb{R}(r),\ \mathrm{nnz}(A) \neq 0,\ C \in \mathbb{R}(rs), \\ A \obslash C := \frac{1}{\mathrm{nnz}(A)} \sum_{A_{j,k} \neq 0} \frac{C_{j,k}}{A_{j,k}}, \end{aligned}$$

where C is treated as an $r \times r$ block matrix with $s \times s$ blocks, ie. as if $C \in \mathbb{R}(s)(r)$.

Theorem 7.3. *The left Kronecker quotient is an inverse operation to the Kronecker matrix product, when applied from the left. For $A \in \mathbb{R}(r)$, $\mathrm{nnz}(A) \neq 0$, $B \in \mathbb{R}(s)$, we have $A \obslash (A \otimes B) = B$.*

Proof.

$$A \obslash (A \otimes B) = \frac{1}{\mathrm{nnz}(A)} \sum_{A_{j,k} \neq 0} \frac{A_{j,k} B}{A_{j,k}} = \frac{1}{\mathrm{nnz}(A)} \sum_{A_{j,k} \neq 0} B = B.$$

■

Lemma 7.4. *For $A \in \mathbb{R}(2^n)$, $B \in \mathbb{R}(2^n)$, $C \in \mathbb{R}(2^n s)$, if $\mathrm{nnz}(A) = 2^n$ then*

$$A \obslash (B \otimes C) = (A' \bullet B)C, \ \textit{where } A'_{j,k} = \begin{cases} \frac{1}{A_{j,k}}, & \textit{if } A_{j,k} \neq 0, \\ 0 & \textit{otherwise.} \end{cases}$$

Proof.

$$A \obslash (B \otimes C) = \frac{1}{\mathrm{nnz}(A)} \sum_{A_{j,k} \neq 0} \frac{B_{j,k} C}{A_{j,k}} = \frac{1}{2^n} \sum_{j,k=1}^{2^n} A'_{j,k} B_{j,k} C = (A' \bullet B)C.$$

■

Lemma 7.5.

$$\begin{aligned} \textit{If } r > 0 \textit{ and } A \in \mathbb{R}(2^{r+s}) = \sum_{T \subseteq \varsigma(-r,r)} (P_r\, \mathbf{e}_T) \otimes A_T, \ \textit{where} \\ A_T \in \mathbb{R}(2^s), \ \textit{then } (P_r\, \mathbf{e}_T) \obslash A = A_T, \ \textit{for } T \subseteq \varsigma(-r,r). \end{aligned}$$

Proof. We have, using the definition of Lemma 7.4, $(P_r\, \mathbf{e}_T)' = P_r\, \mathbf{e}_T$, since each basis matrix consists of $0, -1, 1$ entries only. Then by the same lemma,

$$\begin{aligned} (P_r\, \mathbf{e}_T \obslash ((P_r\, \mathbf{e}_S) \otimes A_S) &= (P_r\, \mathbf{e}_T) \bullet (P_r\, \mathbf{e}_S) A_S \\ &= (\,\mathbf{e}_T \bullet \mathbf{e}_S) A_S, \text{ so} \\ (P_r\, \mathbf{e}_T) \obslash A &= A_T. \end{aligned}$$

■

Corollary 7.6. *If $m > 0$, $T, U, V, W \in \mathbb{R}(2^{m-1})$, and*

$$\begin{aligned} X \in \mathbb{R}(2^m) := I \otimes T + J \otimes U + K \otimes V + JK \otimes W, \ \textit{then} \\ I \obslash X = T,\ J \obslash X = U,\ K \obslash X = V,\ JK \obslash X = W. \end{aligned}$$

The GluCat recursive expression for Q_m.

Theorem 7.7. *For $m > 0, X \in \mathbb{R}(2^m)$ and Q_m as per Definition 6.1,*

$$Q_m(X) = t^+ - w^- + (u^+ - v^-)\,\mathbf{e}_- + (v^+ - u^-)\,\mathbf{e}_+ + (w^+ - t^-)\,\mathbf{e}_-\,\mathbf{e}_+,$$

where

$$\begin{aligned}\mathbf{e}_- &:= \mathbf{e}_{\{-m\}},\ \ \mathbf{e}_+ := \mathbf{e}_{\{m\}},\\ t &:= Q_{m-1}(I \obslash X),\ \ u := Q_{m-1}(J \obslash X),\\ v &:= Q_{m-1}(K \obslash X),\ \ w := Q_{m-1}(JK \obslash X).\end{aligned}$$

Proof. From Theorem 5.2 above, we have, for $m > 0$ and $x \in \mathbb{R}_m$, if $x = a + b\,\mathbf{e}_- + c\,\mathbf{e}_+ + d\,\mathbf{e}_-\,\mathbf{e}_+$, then

$$\begin{aligned}X := P_m(x) = I \otimes (A^+ - D^-) &+ K \otimes (C^+ - B^-)\\ &+ J \otimes (B^+ - C^-) + JK \otimes (D^+ - A^-),\end{aligned}$$

where A, B, etc. are as per Theorem 5.2. Using Corollary 7.6, we have

$$\begin{aligned}I \obslash X &= A^+ - D^-, & J \obslash X &= C^+ - B^-,\\ K \obslash X &= B^+ - C^-, & JK \obslash X &= D^+ - A^-,\end{aligned}$$

and so, for $t := Q_{m-1}(I \obslash X)$, u, v, w etc. as above, we have

$$\begin{aligned}t^+ &= a^+, & t^- &= -d^-,\\ u^+ &= c^+, & u^- &= -b^-,\\ v^+ &= b^+, & v^- &= -c^-,\\ w^+ &= d^+, & w^- &= -a^-.\end{aligned}$$

So now,

$$\begin{aligned}x &= a + b\,\mathbf{e}_- + c\,\mathbf{e}_+ + d\,\mathbf{e}_-\,\mathbf{e}_+\\ &= t^+ - w^- + (u^+ - v^-)\,\mathbf{e}_- + (v^+ - u^-)\,\mathbf{e}_+ + (w^+ - t^-)\,\mathbf{e}_-\,\mathbf{e}_+.\end{aligned}$$

■

The 2-linear complexity of $Q_{p,q}$.

Theorem 7.8. *For $m > 0$, $L_2(Q_m) \leqslant (m+1)4^m = \frac{1}{2}d\log_2 d + d$, where $d = 4^m$ is the dimension of $\mathbb{R}_{m,m}$.*

Proof. The GluCat recursive expression for Q_m uses $\obslash$ four times. Each time needs at most 4^{m-1} additions.

Q_m also uses Q_{m-1} four times. So,

$$\mathbb{A}(Q_m) \leqslant 4^m + 4\mathbb{A}(Q_{m-1}) \ \leqslant m4^m = \frac{1}{2}d\log_2 d.$$

For Q_m, each of the four uses of $\obslash$ needs 4^{m-1} divisions by 2. These divisions can all be delayed to the top level of recursion, so that instead of 4^m divisions by 2 at each of m levels, we can use 4^m divisions by 2^m at the top level only.

So $L_2(Q_m) \leqslant (m+1)4^m = \frac{1}{2}d\log_2 d + d$. ■

Lemma 7.9. *For $q-p \pmod 8 \neq 0, 6$, $L_2(\pi_{p,q}) = 0$, where $\pi_{p,q}$ is as per Definition 6.1.*

Proof. $\pi_{p,q}$ maps distinct generators to distinct generators or zero, and so maps distinct basis elements to either distinct signed basis elements or zero. So $\pi_{p,q}$ does not require any nontrivial additions or multiplications. ∎

Theorem 7.10. *For $p, q \geqslant 0$ and $p+q > 0$, $L_2(Q_{p,q}) \leqslant (m+1)4^m \leqslant 4d(\log_2 d + 4)$, where $m = M(p,q)$ and $d = 2^{p+q}$ is the dimension of $\mathbb{R}_{p,q}$.*

Proof. By Theorem 7.8 and Lemmas 5.5 and 7.9, $L_2(Q_{p,q}) \leqslant (m+1)4^m$, where $m = M(p,q)$. The result for d follows by the same argument as for Theorem 5.6. ∎

8 Lower bounds

The representation P_m has a corresponding representation matrix with respect to an ordering of the bases for $\mathbb{R}_{m,m}$ and $\mathbb{R}(2^m)$. If the basis of $\mathbb{R}_{m,m}$ is given a natural lexicographical ordering by index set, and $\mathbb{R}(2^m)$ is given a basis ordered by column, then by row, the corresponding representation matrix R_m, for P_m shows an interesting pattern. Figure 1 shows the pattern for R_1 and R_2.

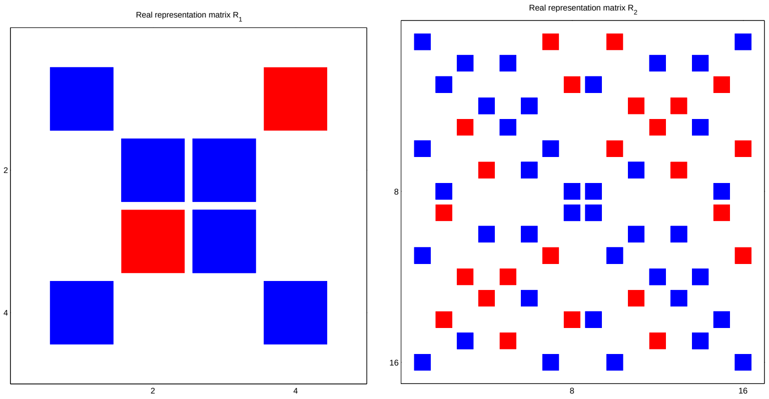

Figure 1: Representation matrices R_1, R_2 with red $= -1$, blue $= 1$, white $= 0$

We can use the properties of the representation matrix R_m and Morgenstern's Theorem [51] to obtain a lower bound on $L_2(P_m)$.

Lemma 8.1. *Let R_m be the representation matrix for P_m as described above. Then*

$$\det R_m = 2^{\frac{1}{2}m4^m}.$$

Proof. Let S be the ordering of subsets of $\varsigma(-m,m)$ used for R_m. Then

$$\begin{aligned}\left(R_m^T R_m\right)_{i,j} &= \sum_{a=1}^{4^m}\sum_{b=1}^{4^m} (P_m\,\mathbf{e}_{S_i})_{a,b} \left(P_m\,\mathbf{e}_{S_j}\right)_{a,b} \\ &= 2^m P_m\,\mathbf{e}_{S_i} \bullet P_m\,\mathbf{e}_{S_j} = 2^m\,\mathbf{e}_{S_i} \bullet \mathbf{e}_{S_j} = 2^m \delta_{i,j}.\end{aligned}$$

Therefore

$$\det R_m^2 = 2^{m4^m}, \text{ and } \det R_m = 2^{\frac{1}{2}m4^m}.$$

■

Theorem 8.2. *(Morgenstern [51], Clausen and Baum [19] Theorem 5.1, p 71)* $L_c(A) \geqslant \log_c |\det A|$ *for any invertible complex matrix* A *and* $2 \leqslant c < \infty$.

Corollary 8.3. *The* 2*-linear complexity* $L_2(P_m)$ *has a lower bound*

$$L_2(P_m) \geqslant \frac{1}{2} m4^m.$$

Corollary 8.4. *Together with Theorem 5.4, we therefore have*

$$\frac{1}{4} d \log_2 d \leqslant L_2(P_m) \leqslant \frac{1}{2} d \log_2 d,$$

so the recursive algorithm for P_m *given by Theorem 5.2 is optimal, possibly up to a factor of 2.*

9 GluCat timing results

GluCat [41] is a `C++` template library for Clifford algebras. The library is based on a prototype by Raja [58] and previous work by Lounesto [42] [43] and others. GluCat is the result of a coursework masters project at the University of New South Wales, supervised by Bill McLean.

On a 2 GHz Athlon 64 PC with 1 GB of PC3200 memory, a GluCat implementation of P_m and Q_m which uses the `C++` standard `hash_map` ([63] 17.6.1 p 497) was tested using 3 timing runs for $\mathbb{R}_{m,m}$ from $m = 1$ to 11. For $m = 4$ to 11, P_m took approximately $(1.88m + 6.8 \pm 2.1)4^m\,\mu s$ and for $m = 3$ to 11, its inverse took approximately $(5.4m + 10.7 \pm 8.9)4^m\,\mu s$.

Four different multiplication algorithms were also compared on three timing runs for $\mathbb{R}_{m,m}$ from $m = 1$ to 11, using the same architecture as for the test for P_m and its inverse.

- The Matrix multiplication starts in $\mathbb{R}(4^m)$ and stays in $\mathbb{R}(4^m)$.
- The Fast Framed–Matrix–Framed multiplication starts and ends in $\mathbb{R}_{m,m}$ and uses the matrix multiplication. For conversion to and from matrices, the fast real matrix representation algorithm is used.
- The Naive Framed–Matrix–Framed multiplication starts and ends in $\mathbb{R}_{m,m}$ and uses the matrix multiplication. For conversion to and from matrices, the naive real matrix representation algorithm is used.
- The Framed multiplication starts and ends in $\mathbb{R}_{m,m}$ and uses multiplication directly in $\mathbb{R}_{m,m}$.

Multiplication speed was timed by squaring an element of $\mathbb{R}_{m,m}$ for each m. Each coordinate of each element was the result of a randomization process and was extremely unlikely to be zero.

The mean time for each of the four algorithms is plotted in Figure 2. The graph shows the time for the Fast Framed–Matrix–Framed multiplication slowly approaching the $\mathrm{O}(d^{3/2})$ behaviour of the Matrix multiplication. The Framed and the Naive Framed–Matrix–Framed multiplications are so slow that timing was not attempted for $m > 8$. Note that the vertical (time) axis is logarithmic.

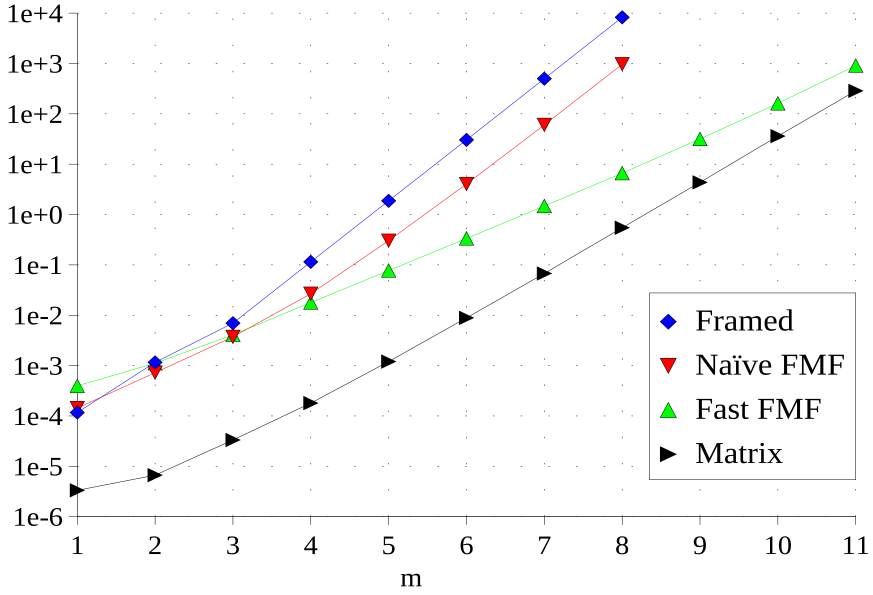

Figure 2: GluCat squaring times (in seconds) for $\mathbb{R}_{m,m}$ using `hash_map`

10 Suggestions for further research

Optimality Are the algorithms given here for $P_{p,q}$ and $Q_{p,q}$ optimal in terms of 2-linear complexity?

More realistic computational models. What is the computational complexity of the fast real matrix representation of Clifford algebras with more realistic computational models, including finite precision arithmetic?

Error analysis. Given a computational model, what are the forward and backward errors (of the fast real matrix representation of Clifford algebras and its inverse as compared to the naive algorithms? ([35] Chapters 1, 24)

What are the forward and backward errors of Clifford multiplication via matrix multiplication using either the fast real matrix representation or the naive algorithms? ([35] Chapters 1, 23)

Fast complex matrix representation. There is an analogous construction for the fast complex matrix representation of Clifford algebras. Does it have better theoretical properties? How does its performance compare in practice to the fast real matrix representation? In what circumstances does it result in faster multiplication than the real matrix representation? ([45] 2.5.1, pp. 8–9)

Generalization to other quotient algebras. A Clifford algebra can be constructed as a quotient of a group algebra by an ideal generated by an element in the centre of the group algebra. For which groups is there a similar construction such that a GFFT for the group algebra implies a fast real matrix representation of the quotient algebra? Is there a construction for the fast real matrix representation of the quotient algebra which does not involve use of the GFFT for the group algebra? ([61], [64])

Applications. What are the applications of the fast real matrix representation of quotient algebras, besides fast multiplication? Are there applications in compression, coding, signal processing and statistics as per the GFFT for group algebras? ([19] Chapters 10, 11, [59], [17])

References

[1] R. Ablamowicz, B. Fauser, *CLIFFORD - A Maple 8 Package for Clifford Algebra Computations*, (version 8, December 27, 2002),

`http://math.tntech.edu/rafal/cliff8/index.html`

[2] R. Ablamowicz, P. Lounesto, J. M. Parra, editors, *Clifford algebras with numeric and symbolic computations*, Birkhäuser, 1996.

[3] M. F. Atiyah, R. Bott, A. Shapiro, “Clifford modules”, *Topology* 3 (1964), pp. 3–38, Supplement 1.

[4] L. Babai, L. Rónyai, “Computing irreducible representations of finite groups”, *Math. Comp.* 55 (1990), 705–722.

[5] H. Bass, “Clifford algebras and spinor norms over a commutative ring”, *Amer. J. Math.* 96 (1974), 156–206.

[6] U. Baum, “Existence and efficient construction of fast Fourier transforms on supersolvable groups”, *Computational Complexity* 1 (1991), no. 3, 235–256

[7] U. Baum, M. Clausen, “Computing irreducible representations of supersolvable groups”, *Math. Comp.* 63 (1994), pp. 351–359.

[8] E. Bayro Corrochano, G. Sobczyk, *Geometric algebra with applications in science and engineering.* Birkhäuser, 2001.

[9] G. Bergdolt, “Orthonormal basis sets in Clifford algebras”, in [2], 1996.

[10] T. Beth, “On the computational complexity of the general discrete Fourier transform”, *Theoretical Computer Science*, 51 (1987), no. 3, 331–339.

[11] L. Bluestein, “A linear filtering approach to the computation of the discrete Fourier transform”, *IEEE Transactions on Audio and Electroacoustics*, Volume 18, Issue 4, Dec 1970, pp451–455.

[12] F. Brackx, R. Delanghe, H. Serras, editors, *Clifford algebras and their applications in mathematical physics*, Proceedings of the Third International Conference held in Deinze, 1993. Fundamental Theories of Physics, 55. Kluwer, 1993.

[13] H. W. Braden, “N-dimensional spinors: Their properties in terms of finite groups”, *Journal of Mathematical Physics*, 26 (4), April 1985.

R. Brauer, H. Weyl, “Spinors in n dimensions”, *Amer. J. Math.*, 57 (1935), pp. 425–449.

[14] P. Budinich, A. Trautman, “An introduction to the spinorial chessboard”, *J. Geom. Phys.* 4 (1987), no. 3, 361–390.

[15] T. Bülow, M. Felsberg, G. Sommer, “Non-commutative hypercomplex Fourier transforms of multidimensional signals”, pp187–207 of [62].

[16] V. M. Chernov, “Clifford algebras as projections of group algebras”, pp461–476 of [8].

[17] G. S. Chirikjian, A. B. Kyatkin, *Engineering applications of noncommutative harmonic analysis. With emphasis on rotation and motion groups.* CRC Press, 2001.

[18] M. Clausen, “Fast generalized Fourier transforms”, *Theoretical Computer Science*, 67 (1989) no. 1, pp. 55-63.

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

[20] M. Clausen, M. Müller, "A fast program generator of FFTs", Proceedings AAECC-13, Hoonolulu, *Lecture Notes in Computer Science*, 1719 (1999), pp. 29–42.

[21] M. Clausen, M. Müller, "Generating Fast Fourier Transforms of solvable groups", *J. Symbolic Computation*, (2001) 11, pp. 1–18.

[22] J. Cnops, "Spherical geometry and Möbius transformations", pp75–84 of [12], 1993.

[23] H. Cohn, C. Umans, "A group-theoretic approach to fast matrix multiplication", *Proceedings of the 44th Annual Symposium on Foundations of Computer Science*, 11-14 October 2003, Cambridge, MA, IEEE Computer Society, pp. 438–449.

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

[25] J. W. Cooley, "How the FFT gained acceptance", *Signal Processing Magazine*, IEEE , Volume: 9 , Issue: 1 , Jan. 1992, pp. 10–13.

[26] C. .W. Curtis, I. Reiner *Representation Theory of Finite Groups and Associative Algebras* , John Wiley, 1962.

[27] P. Diaconis, D. Rockmore, "Efficient computation of the Fourier transform on finite groups", *J. Amer. Math. Soc.*, 3 (1990), no. 2, pp. 297–332.

[28] C. Doran, D. Hestenes, F. Sommen, N. Van Acker, "Lie Groups as Spin Groups", *Journal of Mathematical Physics*, 34 (8) (1993) pp. 3642–3669.

[29] M. Felsberg, T. Bülow, G. Sommer, V. M. Chernov, "Fast algorithms of hypercomplex Fourier transforms", pp231–254 of [62].

[30] L. Finkelstein, W. Kantor, editors, *Proc. 1995 DIMACS Workshop in Groups and Computation.*

[31] P. Fleckenstein, *Geoma: C++ Template Classes for Geometric Algebras*, nklein software, 2000.

`http://www.nklein.com/products/geoma/`

[32] D. Fontijne, *Gaigen*, 2001.

`http://carol.wins.uva.nl/~fontijne/gaigen/`

[33] M. Heideman, D. Johnson, C. Burrus, C., "Gauss and the history of the fast fourier transform", *ASSP Magazine*, IEEE [see also IEEE Signal Processing Magazine] , Volume: 1 , Issue: 4 , Oct 1984, pp. 14–21.

[34] D. Hestenes, G. Sobczyk, *Clifford algebra to geometric calculus : a unified language for mathematics and physics*, D. Reidel, 1984.

[35] Nicholas Higham, *Accuracy and stability of numerical algorithms*, 2nd Edition, SIAM, 2002.

[36] G. N. Hile, P. Lounesto, "Matrix representations of Clifford algebras", *Linear Algebra Appl.* 128 (1990), pp. 51–63.

[37] N. Jacobson, *Basic Algebra I*, W. H. Freeman, 1974.

[38] G. James, M. Liebeck, *Representations and characters of groups*, Cambridge University Press, 1995 (first published 1993).

[39] T. Y. Lam, *The algebraic theory of quadratic forms*, W. A. Benjamin, Inc., 1973.

[40] T. Y. Lam, T. L. Smith, "On the Clifford-Littlewood-Eckmann groups: a new look at periodicity mod 8", *Rocky Mountains Journal of Mathematics*, vol 19, no 3, Summer 1989, pp. 749–785.

[41] P. Leopardi, *GluCat*, `http://glucat.sf.net`

[42] P. Lounesto, R. Mikkola, V. Vierros, *CLICAL User Manual: Complex Number, Vector Space and Clifford Algebra Calculator for MS-DOS Personal Computers*, Institute of Mathematics, Helsinki University of Technology, 1987.

[43] P. Lounesto, "Clifford algebra calculations with a microcomputer", pp39–55 of [50].

[44] P. Lounesto, *Clifford algebras and spinors*, 1st edition, Cambridge University Press, 1997.

[45] S. Mann, L. Dorst, T. Bouma, "The Making of a Geometric Algebra Package in Matlab," Univerrsity of Waterloo Research Report CS-99-27 1999.

[46] D. Maslen, D. Rockmore, "Generalized FFTs: A survey of some recent results", in [30].

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

[48] D. Maslen, "The efficient computation of Fourier transforms on the symmetric group", *Math. Comp.* 67 (1998), no. 223, 1121–1147.

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

[50] A. Micali, R. Boudet, J. Helmstetter, editors, *Clifford algebras and their applications in mathematical physics : proceedings of second workshop held at Montpellier, France, 1989*, Kluwer Academic Publishers, 1992.

[51] J. Morgenstern, "Note on a lower bound of the linear complexity of the fast Fourier transform", *Journal of the ACM*, 20 (1973), pp. 305–306.

[52] M. Müller, M. Clausen, "SUGAR - A computer system for SUpersolvable Groups and Algorithmic Representation theory" (abstract), *Minisymposium on Applications of Nonabelian Group Theory to Imaging and Coding Theory*, June 6, 2001.

[53] S. Okubo, "Real representations of finite Clifford algebras. I. Classification", *Journal of Mathematical Physics*, 32 (1991), no. 7, pp. 1657–1668.

[54] S. Okubo, "Real representations of finite Clifford algebras. II. Explicit construction and pseudo-octonion", *Journal of Mathematical Physics*, 32 (1991), no. 7, pp. 1669–1673.

[55] C. B. Perwass, *The CLU Project*,

`http://www.perwass.de/cbup/clu.html`

[56] I. Porteous, *Topological geometry*, Van Nostrand Reinhold, 1969, 2nd Edition, 1981.

[57] I. Porteous, *Clifford algebras and the classical groups*, Cambridge University Press, 1995.

[58] A. Raja, “Object-oriented implementations of Clifford algebras in C++: a prototype”, in [2].

[59] D. Rockmore, “Some applications of generalized FFTs”, in [30], pp. 329–369, 1997.

[60] N. Salingaros, “On the Classification of Clifford Algebras and Their Relation to Spinors in n Dimensions”, *Journal of Mathematical Physics*, 23 (1982), pp. 1–7.

[61] T. L. Smith, “Decomposition of generalized Clifford algebras”, *Quart. J. Math. Oxford*, 42 (1991), pp. 105-112.

[62] G. Sommer, editor, *Geometric Computing with Clifford Algebras*, Springer, 2001.

[63] B. Stroustrup, *The C++ programming language*, 3rd edition, Addison-Wesley, 1997.

[64] M. Vela, “Central simple $\mathbb{Z}/n\mathbb{Z}$-graded algebras”, *Communications in Algebra*, 30 (4), (2002) pp. 1995–2001.

School of Mathematics
UNSW Sydney
Australia 2052
paul.leopardi@unsw.edu.au