Theoretical Computer Science 5 (1977) 205–209.
© North-Holland Publishing Company

THE COMPLEXITY OF GROUP ALGEBRA COMPUTATIONS

M.D. ATKINSON

Department of Computing Mathematics, University College, Cardiff, United Kingdom

Communicated by M.S. Paterson
Received 6 July 1976
Revised 2 May 1977

Abstract. In this paper we study the computation of a set of bilinear forms associated with a finite group. In the case of a cyclic group these bilinear forms represent the well-known circular convolution of two sequences and can be evaluated efficiently using the fast finite Fourier transform. We shall use a similar technique in the case of a general finite group and will apply it to calculate group algebra products.

Let G denote a finite group of order n and let A denote an associative algebra defined over the complex field $\mathbf{C}$. The group algebra AG consists of the set of formal sums $\Sigma_{g\in G} a_g g$ where the coefficients a_g belong to A. and in which addition, scalar multiplication and multiplication of elements are defined in the following natural way:

$$\sum_{g\in G} a_g g + \sum_{g\in G} b_g g = \sum_{g\in G} (a_g + b_g)g, \tag{1}$$

$$\lambda \sum_{g\in G} a_g g = \sum_{g\in G} (\lambda a_g)g, \tag{2}$$

$$\sum_{g\in G} a_g g \sum_{h\in G} b_h h = \sum_{g\in G} c_g g, \quad \text{where } c_g = \sum_{x\in G} a_x b_{x^{-1}g}. \tag{3}$$

In these formulae $a_g, b_g \in A$ and $\lambda \in \mathbf{C}$. The formula for the sequence $\{c_g\}_{g\in G}$ in (3) is called the convolution of the sequences $\{a_g\}_{g\in G}$ and $\{b_g\}_{g\in G}$.

We are concerned with efficient computations within AG. Of the three operations above (1) and (2) can clearly be accomplished in time $O(n)$ whereas (3) apparently requires time $O(n^2)$. We therefore investigate faster ways of computing the convolution. Our techniques are likely to have most application in group algebra computations where a large number of group algebra elements are computed from an initial small number of elements. For example, searches for units and zero-divisors in AG may well have this form.

It is usual to employ two measures of the cost of computing a set of bilinear forms such as those defined by the c_g:

(i) the total number of arithmetic operations,

(ii) the total number of active multiplications (we recall that an active multiplication is one in which neither factor is a scalar constant).

It is clear that the sequence $\{c_g\}$ could be evaluated with precisely n^2 active multiplications (of the form $a_g b_h$ for all $g, h \in G$) and with $O(n^2)$ arithmetic operations. We shall show that the number of active multiplications can certainly be reduced to less than $n^{1.5}$ and that in some cases the total number of arithmetic operations can be reduced. In particular, our results are of interest from the point of view of bilinear complexity.

From now on K will denote a complete set of inequivalent irreducible matrix representations of G over $\mathbf{C}$. A typical member of K will be denoted by ρ, $\rho(g)$ will denote the image of some group element g under ρ, and f_ρ will denote the dimension of such a matrix.

For any sequence $\{a_g\}_{g\in G}$ of n elements of A we define its Fourier transform (with respect to G) to be the sequence $\{b_\rho\}_{\rho\in K}$ where

$$b_\rho = \sum_{g\in G} a_g \rho(g)$$

is a matrix of dimension f_ρ with entries in A. I am indebted to the referee for pointing out that this definition and the inversion formula of the following lemma are special cases of more general formulae valid in compact groups [4].

Lemma 1. *The sequence $\{a_g\}_{g\in G}$ can be recovered from the sequence $\{b_\rho\}_{\rho\in K}$ by the inversion formula*

$$a_g = \frac{1}{n} \sum_{\rho\in K} f_\rho \operatorname{tr}(b_\rho \rho(g^{-1})).$$

Proof. Consider the block diagonal matrix M whose blocks consist of $b_\rho\rho(g^{-1})$ repeated f_ρ times for each $\rho \in K$. A typical block $b_\rho\rho(g^{-1})$ is equal to $\Sigma_{x\in G} a_x \rho(xg^{-1})$ so M has the form $\Sigma_{x\in G} a_x R(xg^{-1})$ where $R(xg^{-1})$ is a block diagonal matrix in which a typical block $\rho(xg^{-1})$ occurs f_ρ times. Thus $R(xg^{-1})$ is a matrix which is equivalent to the image of xg^{-1} in the regular representation. Hence $\operatorname{tr}(R(xg^{-1})) = 0$ if $x \neq g$ and $\operatorname{tr}(R(xg^{-1})) = n$ if $n = g$. Therefore

$$a_g = \frac{1}{n} \sum_{x\in G} a_x \operatorname{tr}(R(xg^{-1})) = \frac{1}{n} \sum_{\rho\in K} f_\rho \operatorname{tr}(b_\rho \rho(g^{-1})). \quad \square$$

Theorem 2. *The convolution $\{c_g\}_{g\in G}$ of two sequences $\{a_g\}_{g\in G}$ and $\{b_g\}_{g\in G}$ can be computed in at most $\Sigma_{\rho\in K} f_\rho^3$ active multiplications.*

Proof. Consider the following program:

compute the transform $\{d_\rho\}_{\rho\in K}$ of $\{a_g\}_{g\in G}$;

compute the transform $\{e_\rho\}_{\rho\in K}$ of $\{b_g\}_{g\in G}$;
form the matrix products $d_\rho e_\rho$ for each $\rho \in K$;
invert the sequence $\{d_\rho e_\rho\}_{\rho\in K}$.

Active multiplications only occur when computing the products $d_\rho e_\rho$; hence at most $\Sigma_{\rho\in K} f_\rho^3$ active multiplications are performed in all. We complete the proof by showing that $\{c_g\}_{g\in G}$ is the sequence resulting from inverting $\{d_\rho e_\rho\}_{\rho\in K}$; equivalently, that $\{d_\rho e_\rho\}_{\rho\in K}$ is the sequence resulting from transforming $\{c_g\}_{g\in G}$. This follows because

$$d_\rho e_\rho = \sum_{g\in G} a_g \rho(g) \sum_{h\in G} b_h \rho(h) = \sum_{g,h\in G} a_g b_h \rho(gh)$$

$$= \sum_{g\in G} \left(\sum_{x\in G} a_x b_{x^{-1}g} \right) \rho(g) = \sum_{g\in G} c_g \rho(g). \quad \square$$

Corollary 3. *The convolution can be evaluated in at most $n^{1.5}$ active multiplications.*

Proof. The integers f_ρ satisfy $\Sigma_{\rho\in K} f_\rho^2 = n$ and from this it follows that $\Sigma_{\rho\in K} f_\rho^3 \leqslant n^{1.5}$. $\square$

Notice that the bound $n^{1.5}$ is fairly crude. Two examples should make this clear:
(i) for any abelian group $f_\rho = 1$ for all $\rho \in K$ and $|K| = n$; so $\Sigma f_\rho^3 = n$,
(ii) for the simple group of order 60 we have $\Sigma f_\rho^3 = 1 + 27 + 27 + 64 + 125 = 244$ whereas $60^{1.5} = 465$.

Notice also that the bound Σf_ρ^3 could be improved by using Strassen's [5] matrix multiplication method.

The technique of the theorem allows two group algebra elements to be multiplied in at most $n^{1.5}$ active multiplications. However, because of the computation of the transforms and the inverse the total number of arithmetic operations is still $O(n^2)$. Nonetheless we may justify the transform technique in two respects. Firstly, for certain algebras A (for example, matrix algebras of moderately large degree) active multiplications take substantially longer than other operations and may dominate the execution time. Secondly, it is likely that when the technique is used in practice there will be many group algebra computations to perform, the results of some operations being operands to further operations. In such a case great advantage is obtained by working throughout in transform space (where additions and multiplications can also be performed); the Fourier transform is applied to the initial group algebra elements and the inverse transform is applied at the end to get the results.

In spite of these remarks it would obviously be advantageous if the transform and its inverse could be computed more rapidly. This would make the technique more useful for short calculations where active multiplications did not dominate the execution time. We observe that in the case that G is a cyclic group of order n the transform is just the ordinary finite Fourier transform and for this it is well-known [2] that both it and its inverse can be evaluated in time $O(n(r_1 + \cdots + r_m))$ if

$n = r_1 \cdots r_m$. A similar result has been proved by Cairns [1] for any finite abelian group. We shall consider the case that G is a direct product of groups and prove the following result.

Theorem 4. *If $G = G_1 \times \cdots \times G_k$ is a direct product of the groups $G_1, \ldots, G_k$ then the Fourier transform and its inverse can be computed in $O(|G| \Sigma |G_i|)$ arithmetic operations.*

Proof. Suppose first that $G = H \times L$. We shall give a method in which the time t_G for computing the transform has the form

$$t_G \leqslant C(|G| \cdot |L| + |L| \cdot t_H),$$

and a method in which a similar inequality holds for the inverse transform (C here is some constant independent of G). The result then follows by induction on the number of direct factors.

The basic fact which we use is that if σ, τ are irreducible representations of H, L then the matrices $\sigma(h) \otimes \tau(l)$, $h \in H$, $l \in L$, form an irreducible representation of G, and, conversely, every irreducible representation of G has this form. We recall that the tensor product of the p-dimensional matrix $A = [a_{ij}]$ with the q-dimensional matrix B is the pq-dimensional matrix $A \otimes B$ which has the form

$$[a_{ij}B] = \begin{bmatrix} a_{11}B & \ldots & a_{1p}B \\ \vdots & & \vdots \\ a_{p1}B & \ldots & a_{pp}B \end{bmatrix}.$$

The relevant theoretical background may be found in [3].

Consider first the computation of the transform of a sequence $\{a_g\}_{g \in G}$. We have

$$b_{\sigma \otimes \tau} = \sum_{\substack{h \in H \\ l \in L}} a_{hl} \sigma(h) \otimes \tau(l) = \sum_{l \in L} \left(\sum_{h \in H} a_{hl} \sigma(h) \right) \otimes \tau(l),$$

where σ, τ run through the irreducible representations of H, L respectively. The computation of the sums $\Sigma_{h \in H} a_{hl} \sigma(h)$ for all σ and a fixed l is just a computation of the transform with respect to H of the sequence $\{a_{hl}\}_{h \in H}$; thus the computation of all sums $s(\sigma, l) = \Sigma_{h \in H} a_{hl} \sigma(h)$ requires the computation of $|L|$ transforms with respect to H. Then to compute all expressions

$$b_{\sigma \otimes \tau} = \sum_{l \in L} s(\sigma, l) \otimes \tau(l)$$

for each σ, τ requires a further time proportional to $|L| \Sigma_{\sigma, \tau} f_\sigma^2 f_\tau^2 = |L|^2 \cdot |H| = |G| \cdot |L|$. It now follows that

$$t_G \leqslant C(|G| \cdot |L| + |L| \cdot t_H). \tag{4}$$

Now we consider the computation of the inverse transform of the sequence

$\{b_{\sigma\otimes\tau}\}$ where σ, τ range over the irreducible representations of H, L. We need to compute expressions of the form

$$\sum_{\sigma,\tau} f_\sigma f_\tau \operatorname{tr}(b_{\sigma\otimes\tau}\sigma(h)\otimes\tau(l)) \tag{5}$$

for all $h \in H$, $l \in L$. To do this we partition $b_{\sigma\otimes\tau}$ into f_σ^2 square blocks each containing f_τ^2 elements:

$$b_{\sigma\otimes\tau} = [B_{ij}(\sigma, \tau)]$$

where $1 \leq i, j \leq f_\sigma$ and each $B_{ij}(\sigma, \tau)$ is a square matrix of dimension f_τ. We also write $\sigma(h)\otimes\tau(l)$ in a similar blocked form as $[\sigma_{ij}(h)\tau(l)]$ where, again, $1 \leq i, j \leq f_\sigma$ and $\tau(l)$ is of dimension f_τ. Then because

$$\operatorname{tr}(b_{\sigma\otimes\tau}\sigma(h)\otimes\tau(l)) = \sum_{i,j} \operatorname{tr}(B_{ij}(\sigma, \tau)\sigma_{ji}(h)\tau(l))$$

(where, here and subsequently, i, j run from 1 to f_σ), (5) becomes

$$\sum_\tau f_\tau \operatorname{tr}\left[\left(\sum_\sigma f_\sigma \sum_{i,j} B_{ij}(\sigma, \tau)\sigma_{ji}(h)\right)\tau(l)\right].$$

We shall write this as

$$\sum_\tau f_\tau \operatorname{tr}(X(\tau, h)\tau(l)) \tag{6}$$

where $X(\tau, h) = \sum_\sigma f_\sigma \sum_{i,j} B_{ij}(\sigma, \tau)\sigma_{ji}(h)$. For fixed τ and h the computation of $X(\tau, h)$ requires a time propertional to $\sum_\sigma f_\sigma^2 f_\tau^2 = |H| f_\tau^2$. So to compute all the $X(\tau, h)$ requires a time proportional to $|H| \cdot |H| \cdot \sum_\tau f_\tau^2 = |H| \cdot |H| \cdot |L| = |H| \cdot |G|$.

Then the computation of the expressions (5) for all τ and fixed h amounts to the computation of the inverse transform with respect to L of the sequence $\{X(\tau, h)\}_\tau$. So computation of all the expressions (5) can be done in a further time proportional to performing $|H|$ inversions with respect to L.

We therefore have a relation similar to (4) and this again yields the required result. □

References

[1] T.W. Cairns, On the fast Fourier transform on finite abeliar groups. *IEEE Trans. Computers* **20** (1971) 569–571.

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

[3] C.W. Curtis and I.R. Reiner, *Representation Theory of Finite Groups and Associative Algebras* (Interscience, New York, 1962).

[4] E. Hewitt and K.A. Ross, *Abstract Harmonic Analysis*, Vol 2 (Springer, Berlin, 1970).

[5] V. Strassen, Gaussian elimination is not optimal, *Numer. Math.* **13** (1969) 354–356.