Computational Approaches to Finding Irreducible Representations

Joseph Thomas
Research Advisor: Klaus Lux

May 16, 2008

Introduction

Among the various branches of algebra, linear algebra has the distinctions of being well understood and applicable to a wide variety of scientific and mathematical problems. In particular, it can be used to convert mathematical problems into a format where computers can easily be applied to perform calculations that would be impractical for humans to attempt by hand. This project deals with one such application of linear algebra, in a branch of group theory called representation theory.

Given a group G and a field $\mathbb{F}$, a representation ϕ is a homomorphism from G to $GL_n(\mathbb{F})$, the invertible $n \times n$ matrices with entries in $\mathbb{F}$. A representation φ is *irreducible* if the only subspaces invariant under the matrices in the image $\varphi(G)$ are $\{0\}$ and $\mathbb{F}^n$. In much the same way that one can factor an integer into primes, a representation can be factored into irreducible representations. This factorization is unique, up to a change of basis, and can reveal important information about the group. However, as with factoring primes, expressing a representation as a product of irreducible representations requires considerable computation.

Fortunately, there is an algorithm called the *Meat Axe* that can be easily implemented on a computer to speed up the process of finding invariant subspaces and reducing representations. To understand the algorithm, we will require some ideas and notation from linear algebra. However, before delving into this material we will work through a basic example of reducing a representation so that the reader can gain an impression of how we want to use the ideas we are about to present.

Consider S_3, the group of permutations on three elements. For convenience, we will choose our field to be $\mathbb{Z}_3$, and recall that our vector space is spanned by the standard basis vectors $\{e_1, e_2, e_3\}$. Since S_3 is generated by the permutations (123) and (12), it is enough to define a representation ρ for these two elements. For the same reason, we only need to find a subspace invariant under $\rho(123)$ and $\rho(12)$ to reduce the representation.

Since we are dealing with permutations, we will send each generator to a permutation matrix. Thus, we define

$$\rho(123) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \text{ and } \rho(12) = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

(Note that, throughout this report, we will multiply matrices by row vectors, rather than column vectors).

To find a subspace invariant under the matrices $G_1 = \rho(123)$ and $G_2 = \rho(12)$, we will use the following procedure: First, we create new matrices from G_1 and G_2 via addition and multiplication. When we find a matrix that has a nontrivial null space, we take a basis for that null space, and apply the matrices G_1 and G_2 to the vectors in the basis. We record the resulting vectors, apply G_1 and G_2 to them, and continue in this fashion until applying G_1 and G_2 ceases to produce vectors that are linearly independent from the ones we have already recorded. This process is sometimes called "spinning up" the vectors in the null space.

So, for example, we might try:

$$G_1 + G_2 = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$

The vector $v_1 = \begin{bmatrix} 0 & -1 & 1 \end{bmatrix}$ clearly spans the null space of $G_1 + G_2$. Now we reapply G_1 and G_2 to $\begin{bmatrix} 0 & -1 & 1 \end{bmatrix}$, to produce the vectors $v_2 = \begin{bmatrix} 1 & 0 & -1 \end{bmatrix}$ and $\begin{bmatrix} -1 & 0 & 1 \end{bmatrix}$ respectively. Thus, our list of linearly independent vectors consists of v_1 and v_2, and one can easily confirm that reapplying G_1 or G_2 to these cannot add any more linearly independent vectors to the list.

This list of vectors is a basis for a subspace invariant under G_1 and G_2. To use this subspace to reduce the representation, we multiply each vector by G_1, and express the results as a linear combination of the vectors v_1 and v_2. Thus we have $G_1(v_1) = \begin{bmatrix} 1 & 0 & -1 \end{bmatrix} = 0 \cdot v_1 + 1 \cdot v_2$ and $G_1(v_2) = \begin{bmatrix} -1 & 1 & 0 \end{bmatrix} = (-1) \cdot v_1 + (-1) \cdot v_2$. Similarly, we apply G_2 to each of the vectors in the list, so that $G_2(v_1) = \begin{bmatrix} -1 & 0 & 1 \end{bmatrix} = 0 \cdot v_1 + (-1) \cdot v_2$ and $G_2(v_2) = \begin{bmatrix} 0 & -1 & 1 \end{bmatrix} = (-1) \cdot v_1 + 0 \cdot v_2$. Using the coefficients from these linear combinations, we can define a new representation φ_1 that sends (123) to $\begin{bmatrix} 0 & 1 \\ (-1) & (-1) \end{bmatrix}$ and (12) to $\begin{bmatrix} 0 & (-1) \\ (-1) & 0 \end{bmatrix}$.

Now we extend our basis to a basis for the whole vector space by adding the vector $e_1 = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$. If we denote our invariant subspace by U, then we can consider the quotient space $\mathbb{Z}_3^3/U$ to find a second representation. We claim $\mathbb{F}^3/U$ is spanned by the coset $e_1 + U$. Since any vector in $\mathbb{Z}_3^3/U$ can be written as a linear combination of e_1, v_1 and v_2, and $v_1, v_2 \in U$, determining which coset a vector belongs to amounts to expressing the vector as a linear combination of the basis vectors and noting which scalar multiple of e_1 is present.

Now we will define applying a matrix M to a coset $v + U$ by $M(v + U) =$

$M(v) + M(U)$; in other words, we apply M to v as usual, then form a coset using $M(v)$ and the image of U under M. As before, we apply G_1 and G_2 to the basis vectors for the quotient space, so that $G_1(e_1 + U) = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} + G_1(U)$ and $G_2(e_1 + U) = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} + G_2(U)$. Since U is invariant under G_1 and G_2, its image under these matrices is just U again. Since $\begin{bmatrix} 0 & 1 & 0 \end{bmatrix} = e_1 - v_1 - v_2$, $\begin{bmatrix} 0 & 1 & 0 \end{bmatrix} + U = e_1 + U$. Thus, we can conclude $G_1(e_1 + U) = (1)(e_1 + U)$ and $G_2(e_1 + U) = (1)(e_1 + U)$. Then we can define a second representation φ_2 as the function that maps (123) and (12) to the matrix $\begin{bmatrix} 1 \end{bmatrix}$.

Thus, we have decomposed ρ into two simpler representations, φ_1 and φ_2, which we could test for irreducibility and try to reduce further (though φ_2 is obviously irreducible), until we have only irreducible representations. When dealing with much larger matrices, one can imagine having to repeat this process several times to find all the irreducibles.

Clearly, we have left out some important implementation details in this demonstration. For example, in order to make this algorithm effective, one must be able to efficiently determine whether a representation is irreducible. One also has to find an intelligent way of combining the generator matrices to produce a matrix with a small nontrivial null space. Hopefully, though, this sketch motivates the linear algebra that follows, since this is the theory we will require to implement the reducing procedure on a computer.

Preliminary Linear Algebra

Naturally, the problem of determining the existence of a subspace invariant under a set of matrices requires some basic ideas from linear algebra. We will make use of definitions from Shaw [3]. We expect that the reader is familiar with the definition of a subspace. For the sake of clarity, we will assume that a *proper* subspace W of a vector space V is not the zero subspace, unless we specify otherwise. We denote a subspace relationship by "$\leq$" and a proper

subspace relationship by "$<$".

Definition. *Suppose that X and Y are vector spaces over a field $\mathbb{F}$. A map $M : X \to Y$ is said to be a linear map if for all $x_1, x_2 \in X$ and $\lambda \in \mathbb{F}$, $M(x_1 + x_2) = M(x_1) + M(x_2)$ and $M(\lambda x_1) = \lambda M x_1$. The kernel of a M is defined to be $\{x \in X | Mx = 0\}$ and abbreviated ker(M).*

By defining addition and scalar multiplication operations on linear maps, we can treat sets of linear maps as vector spaces. Thus, we have the following definition:

Definition. *Let V and W be vector spaces over a field $\mathbb{F}$. Let $L(V, W)$ be the set of all linear maps of the form $A : V \to W$. For $v \in V$, $A, B \in L(V, W)$ and $\lambda \in \mathbb{F}$, we define addition by $(A + B)v = Av + Bv$, and scalar multiplication by $(\lambda A)v = \lambda(Av)$. The zero vector in $L(V, W)$ is thus the linear mapping which sends all vectors in V to the zero vector in W.*

We are particularly interested in linear maps of the form $A : X \to \mathbb{F}$; that is, the maps which send vectors to scalars in the base field.

Definition. *Given a vector space V over a field $\mathbb{F}$, the dual space of V is the set $L(V, \mathbb{F})$, and is denoted by V^*.*

We will usually denote applying the function v' to a vector v by $v'v$. When we wish to be particularly clear, we will write $\langle v', v \rangle$.

The following definition lets us consider the interactions between dual spaces and linear maps.

Definition. *Let V and W be vector spaces, and A be a linear map from V to W. Then the transpose of A, A^T, is a map from W^* to V^*, given by $w' \mapsto w' \circ A$.*

Observe that when $V = W$, composing an element $w' \in W^*$ with A yields another element in W^*. In this case, we will allow ourselves to denote $w' \circ A$ by Aw', just as we would when A operates on elements of V.

Readers exposed to introductory linear algebra are probably familiar with the transpose, particularly as it applies to matrices. Consequently, we will present the properties of the transpose without proof.

- $0^T = 0$ and when $V = W$, $I_W^T = I_V$
- $(A + B)^T = A^T + B^T$
- $(\lambda A)^T = \lambda A^T$
- $(AB)^T = B^T A^T$
- $(A^T)^T = A$
- $(A^T)^{-1} = (A^{-1})^T$ (provided A is invertible)

Since we will be working with vector spaces of finite dimension, we examine some properties of dual spaces in this case. Let V be a finite dimensional vector space, with an ordered basis $\{e_1, e_2, \ldots, e_n\}$. Then we can define a unique ordered basis $\{e^1, e^2, \ldots, e^n\}$ (the index, by convention, is superscripted), by specifying that for each i in $1 \ldots n$, $\langle e^i, e_i \rangle = 1$ and $\langle e^i, e_j \rangle = 0$ for $j \neq i$.

To see that $\{e^1, e^2, \ldots, e^n\}$ spans V^*, take an element $v' \in V^*$, and then let $\lambda_i = v' e_i$ for each i in $1 \ldots n$. For any vector v in V, we can express v as a linear combination of the basis vectors $\rho_1 e_1 + \rho_2 e_2 + \ldots + \rho_n e_n$. Thus:

$$\begin{aligned} \langle v', v \rangle &= \langle v', \rho_1 e_1 + \rho_2 e_2 + \ldots + \rho_n e_n \rangle \\ &= v'(\rho_1 e_1) + v'(\rho_2 e_2) + \ldots + v'(\rho_n e_n) \\ &= \rho_1 \lambda_1 + \rho_2 \lambda_2 + \ldots + \rho_n \lambda_n \end{aligned}$$

But now observe that since $\rho_i = e^i v$ for i in $1 \ldots n$, we have:

$$\langle v', v \rangle = \lambda_1 \langle e^1, v \rangle + \lambda_2 \langle e^2, v \rangle + \ldots + \lambda_n \langle e^n, v \rangle$$

And then since V^* is a vector space, we have:

$$\begin{aligned} \langle v', v\rangle &= \langle \lambda_1 e^1 + \lambda_2 e^2 + \ldots + \lambda_n e^n, v\rangle \\ v' &= \lambda_1 e^1 + \lambda_2 e^2 + \ldots + \lambda_n e^n \end{aligned}$$

So that v' is a linear combination of our proposed basis elements. Applying this same argument to each of the vectors in $\{e^1, e^2, \ldots, e^n\}$, we can see why these vectors must be linearly independent. It follows then, that a vector space and its dual have the same dimension, and also that for every nonzero vector $v \in V$, there exists a $v' \in V$ so that $\langle v', v\rangle \neq 0$.

Since V^* is a vector space, we can find its dual space, V^{**}. When V is finite-dimensional, one can show V^{**} is isomorphic to V by defining a function $\phi : V \to V^{**}$, given by $\langle \phi(v), v'\rangle = \langle v', v\rangle$. To see ϕ is injective, suppose that ϕ sends a vector $v \in V$ to $0 \in V^*$. Then for all $v' \in V^*$, $\phi(v)v' = 0$. But we also defined $\phi(v)v' = v'v$, and we know that if v is a nonzero vector, there exists a $v' \in V^*$ so that $v'v \neq 0$. Thus, the fact that $\phi(v)v' = v'v = 0$ for all $v' \in V^*$ implies v must be the zero vector in V. Thus, $\ker(\phi) = \{0\}$, which implies ϕ is injective. Since $\dim V = \dim V^*$ and $\dim V^* = \dim V^{**}$, we have $\dim V = \dim V^{**}$. Then since ϕ is injective and $\dim V = \dim V^{**}$, we can conclude ϕ is surjective and thus an isomorphism. Since we will be dealing with only finite-dimensional vector spaces, from here on we allow ourselves to think of V as the dual space of V^*.

In the course of later proofs, we will study a particular type of subspace of the dual space, called the *annihilator*.

Definition. *Let V be a vector space, and W be a subspace of V. Then the annihilator of W, W°, is the set $\{v' \in V^* | \forall w \in W : v'w = 0\}$.*

Among linear maps of the form $A : V \to V$, when a subspace W of V has the property that $A(W) \subseteq W$, we say that W is *invariant* under A. Combining this definition with what we know about the dual space of V, we have the following result:

Theorem. *Let V be a vector space of finite dimension n over a field $\mathbb{F}$, and let A be a linear map from V to V. Then there exists a proper subspace of V invariant under A if and only if there exists a proper subspace of V^* invariant under A^T.*

Proof: Let W be a proper subspace of V invariant under A. Choose a basis $\{w_1, w_2, \ldots, w_m\}$ for W, where m satisfies $m < n$. We can extend this basis to a basis $B = \{w_1, w_2, \ldots, w_m, w_{m+1}, \ldots, w_n\}$ for V. Now, taking $\{e_1, e_2, \ldots, e_n\}$ as the standard basis of V, define a linear map $T : V \to V$ given by $e_i \mapsto w_i$. Since the vectors in B are linearly independent, it follows that $\text{rank}(T) = n$, so that T is invertible.

Now let $\bar{W}$ be the subspace of V spanned by $\{e_1, e_2, \ldots, e_m\}$, and consider the linear map $T^{-1} \circ A \circ T$. If $\bar{w} \in \bar{W}$, then $T(\bar{w}) \in W$, so that $A \circ T(\bar{w}) \in W$ (since W is invariant under A). Then $T^{-1} \circ A \circ T(\bar{w}) \in \bar{W}$, since by the definition of T, T^{-1} must map linear combinations of vectors in W back to linear combinations of vectors in the span of $\{e_1, \ldots, e_m\}$. Thus, $\bar{W}$ is invariant under the mapping $T^{-1}AT$.

By the definition of $\bar{W}$, $\bar{W}^\circ$ is a proper subspace of V^* and is spanned by $\{e^{m+1}, e^{m+2}, \ldots, e^n\}$. By definition, if $\bar{w}^\circ \in \bar{W}^\circ$, then $T^{-1}AT\bar{w}^\circ = \bar{w}^\circ \circ T^{-1} \circ A \circ T$. Now let $v \in \bar{W}$. Since $\bar{W}$ is invariant under $T^{-1}AT$, we know that there exists a $v' \in \bar{W}$ such that $T^{-1} \circ A \circ T(v) = v'$. Consequently, $\bar{w}^\circ \circ T^{-1} \circ A \circ T(v) = \bar{w}^\circ(T^{-1}AT(v)) = \bar{w}^\circ(v')$, and since $\bar{w}^\circ \in \bar{W}^\circ$, $\bar{w}^\circ(v') = 0$. Since we chose $\bar{w}^\circ$ generally, we know that for all $\bar{w}^\circ \in \bar{W}^\circ$, $T^{-1}AT\bar{w}^\circ$ has the property that for all $v \in \bar{W}$, $\bar{w}^\circ(T^{-1}AT(v)) = 0$, and thus $T^{-1}AT\bar{w}^\circ \in \bar{W}^\circ$. Hence, $\bar{W}^\circ$ is invariant under $T^{-1}AT$.

Now suppose that l is an element in $\bar{W}^\circ$, so that for some $l' \in \bar{W}^\circ$, $T^{-1}AT(l) = l'$. Then $A(Tl) = (Tl')$ and it follows that $T(\bar{W}^\circ)$, a proper subspace of V^*, is invariant under A^T.

Since we have demonstrated that V^{**} is isomorphic to V, this argument completes both directions of the proof. ■

Norton's Irreducibility Criterion

Clearly, the ability to determine when a representation is irreducible is essential to any search for irreducible representations. For this purpose, we have Norton's Irreducibility Criterion, which can be stated formally as follows:

Theorem. *Let V be an n dimensional vector space over a field $\mathbb{F}$, and S be a set of linear maps from V to V containing the identity map and closed under addition and composition. Let $B \in S$. If $ker(B) \neq \{0\}$, then V has no proper subspaces invariant under the elements of S if and only if:*

1. *For all $v \in \ker(B)\backslash\{0\}$, the image of v under the maps in S is V.*

2. *For some $x \in \ker_{V^*}(B)$, the image of x under the maps in S is V^*.*

Before attempting the proof, we prove a helpful lemma:

Lemma. *Let V be a finite dimensional vector space over a field $\mathbb{F}$, and let W be a subspace of V invariant under a linear map $\phi : V \to V$. Then $W \cap ker(\phi) = \{0\}$ implies $ker(\phi^T)$ is a subspace of W°.*

Proof: From the definition of the kernel, we know $\ker(\phi^T) = \{x \in V^* | x \circ \phi = 0\}$ where 0 denotes the function in V^* that sends all elements in V to the additive identity in $\mathbb{F}$. Now consider $\mathrm{im}(\phi)$. If x is an element in $\ker(\phi^T)$, then it follows from the definitions of our sets that for all $\phi y \in \mathrm{im}(\phi)$, $x(\phi y) = (x\phi)y = 0$. Then $x \in (\mathrm{im}(\phi))^\circ$, so that $\ker \phi^T \subseteq (\mathrm{im}(\phi))^\circ$. Similarly, if $x' \in V^*$ has the property that $x'(\phi y) = 0$ for all $y \in V$, we can conclude $(x' \circ \phi)y = 0$ for all $y \in V$, so that in V^*, $x' \circ \phi = 0$, and hence $x' \in \ker(\phi^T)$. Thus, we have $(\mathrm{im}(\phi))^\circ \subseteq \ker(\phi^T)$, and consequently $(\mathrm{im}(\phi))^\circ = \ker(\phi^T)$.

Now observe that since $W \cap \ker \phi = \{0\}$, ϕ is injective on the domain W. Since W is invariant under ϕ and W is finite dimensional, it follows that ϕ is surjective as well. Consequently, if we consider ϕ with respect to the whole domain V, we can conclude that $W \leq \mathrm{im}(\phi)$. Thus, if $x' \in V^*$ is an element such that $x'v = 0$ for all $v \in \mathrm{im}(\phi)$, the subset relationship allows

us to conclude that for all $w \in W$, $x'w = 0$. Hence $(\mathrm{im}(\phi))^\circ \leq W^\circ$. Since $(\mathrm{im}(\phi))^\circ = \ker(\phi^T)$, substitution yields $\ker(\phi^T) \leq W^\circ$. ■

Proof of Norton's Irreducibility Criterion:
⇒: Suppose V has no proper subspaces invariant under the linear maps in S. Let v be a nonzero vector in $\ker \phi$, and consider $S \cdot v$. Since S is closed under the addition and composition of its linear maps, we can be sure $S \cdot v$ is a subspace of V. Since the identity map is in S, we can be sure that $S \cdot v \neq \{0\}$. Thus, we must have $S \cdot v = V$, and the first of our two specified properties must hold.

In the preceding discussion of linear algebra, we demonstrated that for any linear map A, V has a proper subspace invariant under A if and only if V^* has a proper subspace invariant under A. Given this theorem, the fact that V has no proper subspaces invariant under the maps in S implies V^* has no proper subspaces invariant under the maps in S. Thus, to show $x \cdot S = V^*$ for some $x \in \ker_{V^*}(B)$, we only need to show there exists a nonzero element in $\ker_{V^*}(B)$; after that, the fact that applying the maps in S to the element produces a nontrivial subspace will allow us to conclude $x \cdot S = V^*$.

By definition, $\ker_{V^*}(B) = \{w \in V^* | \forall v \in V : w \circ B(v) = 0\}$. Thus, $\ker_{V^*}(B) = \mathrm{im}(B)^\circ$. Recall from the rank-nullity theorem that $\dim(\mathrm{im}(B)) + \dim(\ker(B)) = n$. We also know that $\dim(\mathrm{im}(B)) + \dim(\mathrm{im}(B)^\circ) = n$. Cancellation allows us to conclude that $\dim(\ker(B)) = \dim(\mathrm{im}(B)^\circ)$, and thus $\dim(\ker(B)) = \dim(\ker_{V^*}(B))$. The fact that $\dim(\ker(B)) > 0$ implies $\dim(\ker_{V^*}(B)) > 0$, and thus we can conclude that $\ker_{V^*}(B)$ contains a nonzero element.

⇐: Suppose that our two proposed conditions hold, and that W is a proper subspace of V, possibly $\{0\}$, invariant under the linear maps in S. By condition (1), $\ker(B) \cap W = \{0\}$, for if there were a nonzero vector v in the intersection, then $S \cdot v = V$, so that $S \cdot W \leq W$ and $S \cdot W = V$, thereby

implying $W = V$, a contradiction to our assumption that W is a proper subspace of V.

Since $\ker(B) \cap W = \{0\}$, our lemma implies $\ker_{V^*}(B)$ is a subspace of W°, so that we have $\ker_{V^*}(B) \leq W^\circ \leq V^*$. Now apply condition (2), and let $x \in \ker_{V^*}(B)$ be an element such that $x \cdot S = V^*$. By our established subspace relationships, $x \in W^\circ$. Now suppose $v' \in V^*$, so that v' can be expressed as the composition $x \circ A$ where $A \in S$. Then for all $w \in W$, $v'w = x \circ A(w) = x(A(w))$. Since W is invariant under the maps in S, $A(w) = w'$ for some $w' \in W$, and then since $x \in W^\circ$ we have $v'w = x(w') = 0$. This implies $v' \in W^\circ$, so that $V^* \leq W^\circ$. Consequently, we have $V^* = W^\circ$.

But recall that V^* has the property that for all $v \in V, v \neq 0$, there exists a $v^* \in V*$ such that $v^*v \neq 0$. The fact that $V^* = W^\circ$ implies that $v^*w = 0$ for all $v^* \in V^*$ and $w \in W$. Thus, it must be the case that W contains no nonzero vectors, which implies $W = \{0\}$. ■

In practice, Norton's irreducibility criterion gives us the following test for determining when a representation is irreducible, which we quote from R.A. Parker's paper about the Meat Axe algorithm [1]:

Norton's Irreducibility Criterion:

Let $A_1, A_2, \ldots, A_n$ be square matrices of the same size. Let B be an element of the algebra generated (under multiplication and addition) by the A_i. Then at least one of the following occurs:

- B is nonsingular;
- At least one non-zero null vector of B lies in a proper subspace invariant under $A_1, A_2, \ldots, A_n$;
- Every non-zero null vector of B^T lies in a proper subspace invariant under $A_1^T, A_2^T, \ldots, A_n^T$;
- There is no proper subspace invariant under $A_1, A_2, \ldots, A_n$.

Thus, we have an effective way to check if a representation is irreducible.

Word Generators

From the example given at the beginning of this paper, the reader will recall that in order to reduce a representation with a set of generator matrices $A_1, A_2, \ldots A_n$, we need to add and multiply the generator matrices to produce a matrix with a nontrivial null space. We call any matrix that results from adding and multiplying the matrices $A_1, A_2, \ldots A_n$ a *word*, and the algorithm we use to create different combinations of $A_1, A_2, \ldots A_n$ a *word generator*. Since we will be dealing with large matrices, we would would like the word generator we use to have a high probability of creating matrices with nontrivial null spaces. Optimally, we would also like our word generator to find matrices with small null spaces, so that we do not have to perform as many computations on the basis vectors.

Unfortunately, a word generator that tries to find singular matrices directly is constrained by the size of the finite field where the matrix entries reside. To realize this, however, we require a little more theoretical equipment.

Definition. *Let D be a representation from a group G to $GL_n(K)$. Then D is absolutely irreducible if D is irreducible and for all $A \in M_{n\times n}(K)$, the following implication holds:*

$$A \cdot D(g) = D(g) \cdot A \text{ for all } g \in G \implies A \text{ is a scalar matrix.}$$

Definition. *Let R be a ring, and G a group. Then the group ring is a vector space over R, where the basis vectors are the elements of G.*

Notice that we do not define any way for the elements of the ring to interact with the elements of the group—we can think an element of the group ring as a way of storing elements of the ring together with elements of the group.

When we have a representation $D : G \to GL_n(K)$, we can use D to define a new homomorphism F from the group ring KG to $M_{n\times n}(K)$. Since any

element in KG can be written as a unique linear combination of the basis vectors (the group elements), we can define F as the function that sends a linear combination of group elements to the same linear combination of invertible matrices that result when each group element is evaluated by D. Since F sends individual group elements to the same matrices that D does, we say that D can be extended to become a map from KG to $M_{n\times n}(K)$.

Then we have the following lemma from Schur:

Lemma. *If $D : G \to GL_n(K)$ is an absolutely irreducible representation, then D is a surjective map from KG to $M_{n\times n}(K)$.*

Thus, when we have an absolutely irreducible representation, we can express all of the singular matrices as words made up of our generator matrices. From a computational perspective, we are mainly interested in using finite fields as the base fields for our matrices; consequently, the "best case" scenario where a representation is absolutely irreducible allows us to estimate our chances of finding a singular word as a function of the field size.

Suppose that for a finite field K and a group G, we have an absolutely irreducible representation $D : G \to GL_n(K)$. Then since D can be extended to a surjective map from KG to $M_{n\times n}(K)$, we would like to know what proportion of the matrices in $M_{n\times n}(K)$ are singular. Since it is easier for us to reason about the nonsingular matrices, though, we will instead try to find $|GL_n(K)|/|M_{n\times n}(K)|$, and then solve for $1 - |GL_n(K)|/|M_{n\times n}(K)|$.

Let $q = |K|$, so that since we have q choices for each position in an $n \times n$ matrix, $|M_{n\times n}(K)| = q^{n^2}$. In order for a matrix in $M_{n\times n}(K)$ to have a trivial null space, its columns must be linearly independent. Since the first column has n entries, with q possibilities for each entry, we have $q^n - 1$ choices for the first column (any vector but the zero vector is an option). In order for the first and second column to be linearly independent, the second column must not be a multiple of the first. There are q such multiples, so we have $q^n - q$ choices for the second column. Likewise for the first, second, and third columns to be independent, the third must not be a sum of multiples of the

first two, and since there are q^2 such sums, there are $q^n - q^2$ choices for the third column. In general, then, we know that $|GL_n(K)| = \prod_{i=1}^{n} (q^n - q^{i-1})$. Consequently:

$$\frac{|GL_n(K)|}{|M_{n\times n}(K)|} = \frac{\prod_{i=1}^{n} (q^n - q^{i-1})}{q^{n\cdot n}}$$

Now observe that for each i in our product, we can factor $q^n - q^{i-1}$ into $q^{i-1}(q^{n-i+1} - 1)$. Thus, we can write:

$$\frac{|GL_n(K)|}{|M_{n\times n}(K)|} = \frac{q^{\frac{(n-1)n}{2}} \prod_{i=1}^{n} (q^{n-i+1} - 1)}{q^{n\cdot n}}$$

Now consider the product $\prod_{i=1}^{n} (q^{n-i+1} - 1)$. Converted to a sum, the term with the highest power of q is the product of the q^{n-i+1} terms from each $(q^{n-i+1} - 1)$. Since the exponents in this product are $n, n-1, \ldots, 1$, their sum is $n(n+1)/2$. So the summand with the highest power of q is $q^{(n^2+n)/2}$. Similarly, since each term in the product $\prod_{i=1}^{n} (q^{n-i+1} - 1)$ contains a different power of q, we can be sure that the summand containing the second highest power of q is $-1 \cdot \prod_{i=1}^{n-1} q^{n-i+1}$. (In other words, for each term $(q^{n-i+1} - 1)$ in the product, we choose between multiplying by q^{n-i+1} or -1. To find the summand with the second highest power of q, we choose every power of q except the linear one.) So, the summand with the second highest power of q is $-q^{((n^2+n)/2)-1}$. We will call the rest of the sum Q, and note that every term in Q has a power of q smaller than $q^{((n^2+n)/2)-1}$.

Using this information, we find:

$$\begin{aligned}
\frac{|GL_n(K)|}{|M_{n\times n}(K)|} &= \frac{q^{(n^2-n)/2}(q^{(n^2+n)/2} - q^{((n^2+n)/2)-1} + Q)}{q^{n^2}} \\
&= \frac{q^{n^2} - q^{n^2-1} + Q}{q^{n^2}} \\
&= 1 - \frac{q^{n^2-1} - Q}{q^{n^2}}
\end{aligned}$$

So consider $\frac{q^{n^2-1}-Q}{q^{n^2}}$. Every term in Q has a power of q smaller than q^{n^2-1}, so $(q^{n^2-1}-Q) \in O(q^{n^2-1})$. To determine how $\frac{q^{n^2-1}-Q}{q^{n^2}}$ behaves as $q \to \infty$, it is enough to consider the behavior of $\frac{q^{n^2-1}}{q^{n^2}}$. Since $\frac{q^{n^2-1}}{q^{n^2}} = \frac{1}{q}$, and we know $\lim_{q\to\infty} \frac{1}{q} = 0$ it follows $\lim_{q\to\infty} \frac{q^{n^2-1}-Q}{q^{n^2}} = 0$. Returning to the expression for $|GL_n(K)|/|M_{n\times n}(K)|$, we know:

$$\lim_{|K|\to\infty} \frac{|GL_n(K)|}{|M_{n\times n}(K)|} = 1 - \lim_{q\to\infty} \frac{q^{n^2-1}-Q}{q^{n^2}} = 1$$

This equality suggests that as the size of the base field increases, singular matrices comprise a smaller proportion of the total possible matrices. Consequently, a word generator that simply forms words and tests their nullity becomes less effective as the size of the field increases.

Fortunately, this does not mean we are computationally constrained to considering matrices with entries in small fields. If we can find a matrix B with an eigenvalue λ, then we know that the matrix $B-\lambda\cdot I$ is singular. Thus, the problem of finding singular words can be seen as a problem of finding words that have eigenvalues. A theorem by D.F. Holt and S. Rees states that as $|K| \to \infty$, there is a fairly sizable lower bound on the percentage of matrices in $M_{n\times n}(K)$ with an eigenvalue. Consequently, we can be fairly certain that a word generator that finds matrices with eigenvalues will be effective even for finite fields with many elements.

Using Word Generators to Analyze Representations

By placing further restrictions on the words we consider, we can use a word generator to gain further information about the structure of a representation. To do this, we must observe that a word describes a combination of generator matrices (and their inverses) under the operations of addition, multiplication, and scalar multiplication. Representations of the same group will each have

matrices identified with the group's generators, so we can evaluate the same word in different representations. This is similar to the way that we can examine the same polynomial over different fields.

The particular words we will consider are called *peakwords*. Given a group representation and its irreducible factors, a peakword has the property (among others) that it is singular when evaluated in one of the representation's irreducible factors, and nonsingular when evaluated in the others.

Studying the peakwords allows us to identify all of the invariant subspaces in the original representation. Since we can factor a representation whenever we have an invariant subspace, knowing all of the invariant subspaces allows us to determine all of the different ways of splitting the representation. Thus, in much the same way that we can determine all of the divisors of a given integer, the peakwords allow us to find all of the representations that are constituents of the original representation.

To use the peakwords in this fashion, one must perform two tasks. First, one must find a peakword for each irreducible representation. Next, one must evaluate those peakwords over the original representation, and analyze the null spaces of the resulting matrices to find the "simplest" invariant subspaces. Larger invariant subspaces can then be constructed from these smaller subspaces, until one has found all of the invariant subspaces.

Finding Peakwords

Finding a peakword for each irreducible representation is similar to the process used to find the word used to split the representation. However, because of the additional criteria imposed upon the peakwords, one must be sure that the algorithm used to find the peakwords is fairly efficient. In the course of implementing a "peakword-finder" in GAP, we refined our algorithm several times.

One computationally expensive approach to finding peakwords is to study

the characteristic polynomials of the words. We have already seen how, by finding the eigenvalues of a matrix, we can construct matrices that have non-trivial null spaces. In this case, we took linear factors from the characteristic polynomial, and interpreted them them as ways of combining matrices—that is, we treated addition in the base field as matrix addition, multiplication as matrix multiplication, and any constant coefficients as the identity matrix scaled by a constant from the base field.

As a consequence of the Cayley-Hamilton theorem, it is possible to select irreducible factors of higher order from the characteristic polynomial, and then evaluate the original matrix over these factors to obtain other singular matrices. Thus, one way to find peakwords is to evaluate the same word on each of the irreducibles, calculate the characteristic polynomial for each resulting matrix, and then factor each of those polynomials into irreducibles. We can then collect the irreducibles that appear exactly once among the factorizations. From these factors, we can then select ones that have the particular degree that we desire. The peakwords can then be taken to be the original word evaluated on the irreducible factor.

For large fields, this may be a satisfactory approach. However, calculating characteristic polynomials and then factoring them is time consuming. In later versions of our script, we attempted to optimize the search for peakwords by considering only the linear factors, with the additional restrictions that when we evaluated the word on the linear factor, the resulting matrix and its square both have null spaces of a particular dimension that is derived from the irreducible representation in which the word is singular.

An additional observation that proved important in optimizing the peakword finding script was that the same word can sometimes produce several peakwords, as a consequence of having distinct linear factors in different modules. As a result, it was important to iterate over the words and filter out every peakword that appeared, instead of iterating over the irreducible factors of the representation and finding a peakword for each. This complicates

the manipulation of data-structures necessary to implement the algorithm, but it minimizes the number of matrix operations we have to perform to analyze each word, and these are the most expensive part of the algorithm.

Determining Invariant Subspaces

Once we have found a peakword for each irreducible factor, we can use the peakwords to determine the simplest invariant subspaces, and use these subspaces to build all other invariant subspaces. To find the basic subspaces, we evaluate each peakword in the original representation, and record the null space of the resulting matrix. Taking bases for these null spaces, we can spin up each basis vector and record the *cyclic* subspace that results. The method by which these subspaces are constructed ensure each is invariant under the generators. For each cyclic subspace S, we then determine the other cyclic subspaces S contains, and store the results in a table, so that the entry in the i-th row and j-th column tells whether the subspace S_j is contained in S_i.

Next, for vectors v and w from the same basis, we consider the sum of their cyclic subspaces, $V+W$, and check which of the cyclic subspaces $V+W$ contains, a process that can be sped up by observing that $V+W$ contains all of the subspaces contained by V or W. These results are stored in a second set of tables.

Because the cyclic subspaces are the simplest invariant subspaces, we can describe any larger invariant subspace in terms of the cyclic subspaces it contains. Given a list of containments, we can check the first table to determine which other cyclic subspaces the list must contain. Then we can check the list against the tables of subspace sums, to see which other cyclic subspaces must be added to the list to obtain a closed subspace. Described another way, the tables allow us to determine which lists of cyclic subspaces describe valid invariant subspaces.

The advantage to constructing tables of subspace inclusions is that once the tables are constructed we do not have to perform any more subspace calculations find all of the invariant subspaces. In fact, because the tables described above contain only Boolean data, determining the necessary subspace inclusions for an initial list of cyclic subspaces amounts to performing a series of bitwise operations. As a result, one can easily iterate over all of the possible combinations of the cyclic subspaces and use the tables to complete each list of cyclic subspaces to a closed invariant subspace. In this way, one can determine all of the invariant subspaces, each expressed in terms of the cyclic subspaces it contains.

Current Work

Our planned GAP implementation of the procedures described above is divided between two scripts, one that calculates peakwords, and a second that constructs tables of subspace inclusions. The peakword-finding script is complete, although we may try to further optimize our implementation in the future. A copy of the most recent version of this script has been included at the end of this report. Currently, we are writing the script that builds the tables of inclusions.

Works Cited

[1] Parker, R.A. "The Computer Calculation of Modular Characters (The Meat-Axe)." Computational Group Theory. ed. Michael D. Atkinson. Orlando: Academic Press Inc, 1984.

[2] Serre, J.-P. Linear Representations of Finite Groups. New York: Springer-Verlag, 1977.

[3] Shaw, Ronald. Linear Algebra and Group Representations. New York: Academic Press Inc., 1982.