Counting p-groups and Lie algebras using PORC polynomials

Bettina Eick and Michael Vaughan-Lee

November 23, 2018

Abstract

Counting problems whose solution is PORC were introduced in a famous paper by Higman (1960). We consider two specific counting problems with PORC solutions: the number of isomorphism types of d-generator class-2 Lie algebras over $\mathbb{F}_q$ (as a function in q) and the number of isomorphism types of d-generator p-class 2 p-groups (as a function in p). We prove lower bounds for the degrees of their PORC polynomials for all $d \in \mathbb{N}$ and we determine explicit PORC polynomials for $d \leq 7$.

1 Introduction

Let S be an infinite subset of the integers and let $f : S \to \mathbb{Q}$ be a function. We say that f is *PORC (polynomial on residue classes)* if there exists $m \in \mathbb{N}$ (called the modulus) and polynomials $g_0, \ldots, g_{m-1} \in \mathbb{Q}[x]$ such that

$$f(s) = g_i(s) \quad \text{for all} \quad s \in S \quad \text{with} \quad s \equiv i \bmod m.$$

Higman [7] initiated the investigation of counting problems whose solution is PORC. He proved, for example, that the number of isomorphism types of groups of order p^n, whose Frattini subgroup is elementary abelian and central, considered as a function of the prime p, for fixed n, is PORC. Moreover, he introduced his famous PORC conjecture suggesting that the number of isomorphism types of all groups of order p^n, considered as a function of the prime p, for fixed n, is PORC.

A central tool in Higman's approach is the translation of counting problems to applications of linear algebra. We consider two instances here. Let $d, k \in \mathbb{N}$ and let V denote a d-dimensional vector space over the finite field $\mathbb{F}_q$ with q elements. Write $L_{d,k}(q)$ for the number of orbits of $\mathrm{GL}(V)$ acting on the k-dimensional subspaces of $W = V \wedge V$ and $G_{d,k}(q)$ for the number of orbits of $\mathrm{GL}(V)$ acting on the k-dimensional subspaces of $W^+ = (V \wedge V) \oplus V$. Write Π for the set of all prime powers, π for the subset of all primes and π_o for the set of all odd primes.

1 Theorem:(Higman [7]) *Let $d, k \in \mathbb{N}$.*

(a) *$G_{d,k}(p)$ coincides with the number of isomorphism types of d-generator groups of order p^{d+k} and p-class 2 for all $p \in \pi_o$.*

(b) *$L_{d,k}(p)$ coincides with the number of isomorphism types of d-generator groups of order p^{d+k}, exponent p and class 2 for all $p \in \pi_o$.*

(c) *$L_{d,k}(q)$ coincides with the number of isomorphism types of d-generator Lie algebras of dimension $d+k$ and class 2 over $\mathbb{F}_q$ for all $q \in \Pi$.*

Let $d, k \in \mathbb{N}$. Eick & O'Brien [2] described an effective method to compute $G_{d,k}(q)$ or $L_{d,k}(q)$ for a fixed single $q \in \Pi$. Vaughan-Lee [10] introduced a method to determine a PORC polynomial on π_o describing $G_{d,k}(p)$ as a function in $p \in \pi_o$. This method translates readily to the computation of $L_{d,k}(p)$ for $p \in \pi_o$ and was used to determine PORC polynomials describing $G_{d,k}(p)$ for $d \leq 6$. Eick & Wesche [5] described the calculation of a PORC polynomial on Π for the action of $\mathrm{GL}(V)$ on the k-dimensional subspaces of $V \otimes V$ as a function in $q \in \Pi$. We recall the main principles of these methods in Section 2 for completeness.

We observe that a PORC function $f(s)$ on S with modulus m can be written as a polynomial in s whose coefficients are $\mathbb{Q}$-linear combinations of products of terms of the form $\gcd(s-i, p^k)$ where $0 \leq i < p^k$ and $p^k \mid m$, see Section 4. There is no unique way of writing these coefficients, since there are relations which hold between the gcds. Section 4 discusses how to compute a possibly short PORC polynomial for a PORC function.

We say that the PORC function f on S has degree l if f can be represented by a PORC polynomial in s which is, as a polynomial in s, of degree l and its leading coefficient is non-zero for all $s \in S$. Section 3 contains a proof for the following.

2 Theorem: *Let $d, k \in \mathbb{N}$.*

(a) *If $L_{d,k}(q) \neq 0$, then the degree of $L_{d,k}(q)$ is at least $kd(d-1)/2 - k^2 - d^2 + 1$.*

(b) *If $G_{d,k}(q) \neq 0$, then the degree of $G_{d,k}(q)$ is at least $kd(d+1)/2 - k^2 - d^2$.*

Finally, we combine the ideas of [10] and [5] with the methods described in Section 4 and use these to determine PORC polynomials on Π for $G_{d,k}(q)$ and $L_{d,k}(q)$ for $d \leq 7$, $k \in \mathbb{N}$ and all $q \in \Pi$, see Section 5 for details. The resulting polynomials are available in electronic form in [3] and [11]. Section 6 contains an abbreviated description of them.

The bounds of Theorem 2 are attained in many cases in the range $d \leq 7$. Exceptions are $L_{6,3}(q)$ and $G_{d,2}(q)$ for $d \geq 3$, where the actual degree exceeds the lower bound by 1.

2 The algorithm

Let V be a d-dimensional vector space over the field $\mathbb{F}_q$ with q elements. In this section we briefly describe the main features of the algorithm to compute a PORC polynomial for the number of orbits of $\mathrm{GL}(V)$ acting on the k-dimensional subspaces of $W = (V \wedge V)$ or $W^+ = W \oplus V$, respectively.

2.1 The type of a matrix

Green [6] introduced the notion of a *type* of a matrix. This plays a key role in our algorithms and we recall it here briefly.

Let $g \in \mathrm{GL}(V)$ have minimal polynomial $\mu(x)$. Let $p_1(x), \ldots, p_m(x)$ be the distinct irreducible factors of $\mu(x)$. Then for $1 \leq i \leq m$ there exists a sequence $s_i = (s_{i,1}, \ldots, s_{i,m_i})$ of natural numbers so that the rational canonical form of g is a block diagonal matrix whose diagonal blocks are the companion matrices of the polynomials $p_i(x)^{s_{i,j}}$. Let $n_i = \deg(p_i(x))$. The *type* of g is the sequence (which is assumed to be lexicographically sorted)

$$type(g) = ((n_1, s_1), \ldots, (n_m, s_m)).$$

If $t = ((n_1, s_1), \ldots, (n_m, s_m))$ is the type of a matrix in $\mathrm{GL}(V)$ and $\overline{s}_i = s_{i,1} + \ldots + s_{i,m_i}$, then $d = n_1\overline{s}_1 + \ldots + n_m\overline{s}_m$ holds. Hence for a given d there are only finitely many possible types and these can be listed easily (and are independent of the field-size).

3 Theorem: (Green [6], Eick & O'Brien [2]) *Let t be the type of a matrix in $\mathrm{GL}(V)$ for $V = \mathbb{F}_q^d$.*

(a) *The set $E_t = \{g \in \mathrm{GL}(V) \mid type(g) = t\}$ is the union of $k_t(q)$ different conjugacy classes which all have the same size $s_t(q)$. Both $k_t(q)$ and $s_t(q)$ can be described by polynomials in q.*

(b) *Let $\mathcal{U}_k$ denote the set of k-dimensional subspaces of V and let g be an element of type t in $\mathrm{GL}(V)$. The number of fixed points $\mathrm{Fix}_g(\mathcal{U}_k)$ depends on t only and can be described by a polynomial in q.*

Formulae for $k_t(q)$ and $s_t(q)$ as polynomials in q have been determined by Green [6]. An effective method to compute a polynomial in q for $\mathrm{Fix}_g(\mathcal{U}_k)$ using the type of g only was been described by Eick & O'Brien [2]. We write $\mathrm{Fix}_t(\mathcal{U}_k)$ for the number of fixed points of an element of type t.

2.2 Burnside's lemma

The number of orbits of a group acting on a finite set can be determined using Burnside's lemma. To evaluate this, let $\overline{g}$ denote the action of $g \in \mathrm{GL}(V)$ on W or W^+, respectively, and let $\mathcal{U}_k$ denote the set of k-dimensional subspaces of W or W^+, respectively. Then the number of orbits $\mathcal{O}$ of $\mathrm{GL}(V)$ acting on $\mathcal{U}_k$ is given by

$$\mathcal{O} = \frac{1}{|\mathrm{GL}(V)|} \sum_{g \in \mathrm{GL}(V)} \mathrm{Fix}_{\overline{g}}(\mathcal{U}_k).$$

Let $G = \mathrm{GL}(V)$ and let $\overline{G}$ denote the action of G on W or W^+, respectively. Let $\mathcal{R}$ denote a set of representatives for the conjugacy classes in G; for example, $\mathcal{R}$ can be chosen as the set of rational canonical forms of matrices in G. Let $\mathcal{T}$ be the set of types of matrices in G and $\overline{\mathcal{T}}$ the set of types of matrices in $\overline{G}$. For $t \in \mathcal{T}$ and $\overline{t} \in \overline{\mathcal{T}}$ let

$$A(t, \overline{t}) = |\{g \in \mathcal{R} \mid type(g) = t \text{ and } type(\overline{g}) = \overline{t}\}|.$$

Then Theorem 3(a) asserts that

$$\sum_{\bar{t}\in\overline{\mathcal{T}}} A(t,\bar{t}) = k_t(q).$$

Using $s_t(q)$ for the size of a conjugacy class of an element of type t in G, we obtain the following refined formula for $\mathcal{O}$

$$\mathcal{O} = \frac{1}{|G|}\sum_{t\in\mathcal{T}}\sum_{\bar{t}\in\overline{\mathcal{T}}} A(t,\bar{t}) \;\; s_t(q) \;\; \mathrm{Fix}_{\bar{t}}(\mathcal{U}_k).$$

As recalled above, polynomials in q for $s_t(q)$, for $\mathrm{Fix}_{\bar{t}}(\mathcal{U}_k)$ and for $|G|$ can be computed readily. It thus remains to compute a PORC polynomial for $A(t,\bar{t})$ for all $t\in\mathcal{T}$ and $\bar{t}\in\overline{\mathcal{T}}$ and this is the key problem of the algorithm. We consider this in more detail in Section 5.

3 The lower bounds

In this section we prove Theorem 2 based on the results and the notation of Section 2. Let $d,k\in\mathbb{N}$ and $V=\mathbb{F}_q^d$. Write m for the dimension of W or W^+, respectively, and $\mathcal{O}$ for the orbits of $G=\mathrm{GL}(V)$ on the set $\mathcal{U}_k$ of subspaces of dimension k in W or W^+, respectively. Then it follows that

$$|\mathcal{U}_k|/|G| \le |\mathcal{O}| \le |\mathcal{U}_k|.$$

Further,

$$|\mathcal{U}_k| = \prod_{i=1}^{k}\frac{q^{m-i+1}-1}{q^i-1} \quad \text{and} \quad |G| = \prod_{i=1}^{d}(q^d-q^i).$$

Hence if f is a PORC polynomial for $\mathcal{O}$, then

$$u = \sum_{i=1}^{k}(m-i+1-i) = mk-k^2$$

is an upper bound for the degree of f and a lower bound is

$$l = \sum_{i=1}^{k}(m-i+1-i) - d^2 = mk-k^2-d^2.$$

This proves the lower bound for $G_{d,k}(q)$ as in Theorem 2(b), since $G_{d,k}(q)$ coincides with $\mathcal{O}$ if we consider the action on W^+ and $m=d(d+1)/2$ in this case.
The function $L_{d,k}(q)$ coincides with $\mathcal{O}$ if we consider the action on W. In this case the diagonal matrices in G act as diagonal matrices on W and hence act trivially on $\mathcal{U}_k$. Thus in this case we obtain

$$(q-1)|\mathcal{U}_k|/|G| \le |\mathcal{O}|$$

and hence the lower bound improves to $mk-k^2-d^2+1$ with $m=d(d-1)/2$. This proves Theorem 2(a).
Alternatively, both lower bounds can also be obtained via the Burnside formula for $\mathcal{O}$ by noting that the type $t=(1,(1,\ldots,1))$ corresponding to the diagonal matrices in G yields summands whose degree coincides with the lower bound.

4 Simplifying PORC polynomials

Let S be an infinite subset of the integers and let $f : S \to \mathbb{Q}$ be a PORC function. As described in the introduction, this implies that there exists $m \in \mathbb{N}$ and polynomials $g_0, \ldots, g_{m-1} \in \mathbb{Q}[x]$ such that

$$f(s) = g_i(s) \quad \text{for all} \quad s \in S \quad \text{with} \quad s \equiv i \bmod m.$$

For $0 \leq i \leq m-1$ define $\chi_{(i,m)}(s) : S \to \mathbb{Q}$ via $\chi_{(i,m)}(s) = 1$ if $s \equiv i \bmod m$ and $\chi_{(i,m)}(s) = 0$ otherwise. Then a closed formula for the PORC function f is

$$f(s) = \sum_{i=0}^{m-1} \chi_{(i,m)}(s) g_i(s).$$

We investigate the *characteristic functions* $\chi_{i,m}(s)$ in more detail. If $\gcd(a,b) = 1$, then $\chi_{i,ab}(s) = \chi_{(i \bmod a),a}(s)\chi_{(i \bmod b),b}(s)$. Hence it is sufficient to consider the characteristic functions for prime powers $m = p^k$. For $0 \leq i < p^k$ and $\bar{i} = i \bmod p^{k-1}$ we note that

$$\chi_{i,p^k}(s) = \frac{\gcd(s-i, p^k) - \gcd(s-\bar{i}, p^{k-1})}{p^k - p^{k-1}}.$$

In summary, a PORC function f can be written as a polynomial whose coefficients are $\mathbb{Q}$-linear combinations of products of terms of the form $\gcd(s-i, p^k)$ where $0 \leq i < p^k$. However there is no unique way of writing these functions since there are relations which hold between the gcds. For example, if p is prime then

$$\sum_{i=0}^{p-1} \chi_{(i,p)} = 1,$$

and it follows from this relation that, for $0 \leq i < p$, $\chi_{(i,p)}(s)$ can be expressed as a $\mathbb{Q}$-linear combination of $1, \gcd(s,p), \gcd(s-1,p), \ldots, \gcd(s-(p-2),p)$. There are additional complications when the set S on which f is defined is the set of primes or the set of prime powers. Then, for example, the relation $\chi_{(0,2)}\chi_{(0,3)} = 0$ holds on S.

4.1 The case that S is the set of all integers

First we consider PORC functions which are defined on the whole of the integers $\mathbb{Z}$. Let p^k be a prime power, and let V_{p^k} be the set of functions $f : \mathbb{Z} \to \mathbb{Q}$ of the form

$$\sum_{i=0}^{p^k-1} \alpha_i \chi_{(i,p^k)}$$

with $\alpha_i \in \mathbb{Q}$. We view V_{p^k} as a vector space over $\mathbb{Q}$ — as such, it has dimension p^k and basis $\{\chi_{(i,p^k)} \,|\, 0 \leq i < p^k\}$. To simplify notation, for $0 \leq i < p^r$ and $r \geq 1$, we let $g_{(i,p^r)} : \mathbb{Z} \to \mathbb{Q}$ be defined by

$$g_{(i,p^r)}(s) = \gcd(s-i, p^r).$$

Note that $\chi_{(i,p^r)}$ and $g_{(i,p^r)}$ are elements in V_{p^k} for $0 \leq i < p^r$ and $r = 1, 2, \ldots, k$.

4 Theorem: *Let B_{p^k} be the subset of V_{p^k} consisting of the constant function* 1 *and the functions $g_{(i,p^r)}$ with $1 \le r \le k$ and $0 \le i < p^r - p^{r-1}$. Then B_{p^k} is a basis for V_{p^k}.*

Proof: The proof is by induction on k. The case $k = 1$ follows from the definition of $\chi_{(i,p)}$ $(0 \le i < p)$, and from the fact that

$$\sum_{i=0}^{p-1} g_{(i,p)} = 2p - 1.$$

So assume that the result is true for $k-1$, and consider the case k. Let W be the subspace of V_{p^k} spanned by B_{p^k}. By induction we may assume that $g_{(i,p^{k-1})} \in W$ for $0 \le i < p^{k-1}$, and so

$$\chi_{(i,p^k)} \in W \text{ for } 0 \le i < p^k - p^{k-1}.$$

Now let $p^k - p^{k-1} \le i < p^k$, and let $j = i \bmod p^{k-1}$. Then

$$\chi_{(i,p^k)} = \chi_{(j,p^{k-1})} - \sum_{r=0}^{p-2} \chi_{(j+rp^{k-1},p^k)} \in W$$

and this completes the proof. •

Using the basis B_{p^k} gives us a unique way of representing elements of V_{p^k}.
Let $m = q_1 q_2 \dots q_k$ be a product of prime powers (with q_i coprime to q_j for $i \ne j$), and let V_m be the set of functions $f : \mathbb{Z} \to \mathbb{Q}$ of the form

$$\sum_{i=0}^{m-1} \alpha_i \chi_{(i,m)}$$

with $\alpha_i \in \mathbb{Q}$. As a vector space over $\mathbb{Q}$, V_m has dimension m and we have the following corollary to Theorem 4.

5 Corollary: *V_m has basis B_m consisting of all products $f_1 f_2 \dots f_k$ with $f_i \in B_{q_i}$ for $i = 1, 2, \dots, k$.*

4.2 The case that S is the set of all prime powers

The situation is more complicated if our PORC functions are only defined over primes or prime powers (as is the case for our functions $L_{d,k}(q)$ and $G_{d,k}(q)$). Let $m > 1$ be a positive integer and let $0 \le i < m$. If i is coprime to m then Dirichlet's theorem on arithmetic progressions implies that there are infinitely many primes p equal to i mod m. So $\chi_{(i,m)}(p) = 1$ for infinitely many primes p. This shows that even when we consider the functions $\chi_{(i,m)}$ as being defined only on the set of prime powers, $\{\chi_{(i,m)} \,|\, 1 \le i < m, \gcd(i,m) = 1\}$ is linearly independent. But if i is not coprime to m then it happens frequently that $\chi_{(i,m)}(q) = 0$ for all prime powers q. In particular this is true if $\gcd(i,m)$ is divisible by two distinct primes. If $\gcd(i,m) = p^r$ for some prime p then $\chi_{(i,m)}(q)$ can only be non zero for $q = p^s$ with $s \ge r$, and it is straightforward to check whether or not

$\chi_{(i,m)}(p^s) = 0$ for all $s \geq r$. For $2 \leq m \leq 14$ the following functions take the value 0 on all prime powers:

$$\chi_{(0,6)},\ \chi_{(6,8)},\ \chi_{(6,9)},\ \chi_{(0,10)},\ \chi_{(0,12)},\ \chi_{(6,12)},\ \chi_{(10,12)},\ \chi_{(0,14)},\ \chi_{(6,14)},\ \chi_{(10,14)},\ \chi_{(12,14)}.$$

However, when we consider the functions $\chi_{(i,m)}$ as being defined only on prime powers, then

$$\{\chi_{(i,m)} \,|\, 0 \leq i < m,\ \chi_{(i,m)}(q) \neq 0 \text{ for some prime power } q\}$$

is linearly independent. To see this suppose that

$$\sum_{i=0}^{m-1} \alpha_i \chi_{(i,m)}(s) = 0$$

whenever s is a prime power, and suppose that $\chi_{(j,m)}(q) \neq 0$ for some prime power q. Then $\chi_{(i,m)}(q) = 0$ for $i \neq j$, and so

$$0 = \sum_{i=0}^{m-1} \alpha_i \chi_{(i,m)}(q) = \alpha_j.$$

As above, let Π be the set of prime powers, and let W_m be the set of functions $f : \Pi \to \mathbb{Q}$ of the form

$$\sum_{i=0}^{m-1} \alpha_i \chi_{(i,m)}$$

with $\alpha_i \in \mathbb{Q}$. Then as a vector space over $\mathbb{Q}$, W_m has dimension

$$|\{i \,|\, 0 \leq i < m,\ \chi_{(i,m)}(q) \neq 0 \text{ for some } q\}|.$$

We use these ideas to simplify our functions $L_{d,k}(q)$ and $G_{d,k}(q)$. A certain amount of simplification is built into the programs which compute the functions, but we apply further simplification to the initial output. For example, the first version of the functions $G_{(6,k)}(q)$ $(1 \leq k \leq 21)$ which were computed by our programs involved polynomials in q with coefficients which were $\mathbb{Q}$-linear combinations of 164 different products of elements $\gcd(q-i, p^k)$ with

$$p^k \in \{2, 4, 8, 16, 64, 3, 9, 5, 7, 11, 13, 17, 19\}.$$

These products included

$$1,\ \gcd(q,2).\gcd(q-58,64),\ \gcd(q,2)^2.\gcd(q-1,5),\ \gcd(q-1,17),\ \gcd(q,2).\gcd(q,7)$$

for example. (Products like $\gcd(q,2).\gcd(q-58,64)$ and $\gcd(q,2)^2.\gcd(q-1,5)$ can arise when two PORC functions are multiplied together.) The first step is to write these products as linear combinations of the basis elements B_m for V_m for various m. We write $\gcd(q,2).\gcd(q-58,64)$ as a linear combination of elements in B_{64}, and we write $\gcd(q,2)^2.\gcd(q-1,5)$ as a linear combination of elements in B_{10}, and so on. After these simplifications, all the coefficients were linear combinations of 74 elements of B_m for $m \in S$ where

$$S = \{2, 4, 8, 16, 3, 9, 5, 7, 11, 13, 17, 19, 6, 10, 12, 14, 15, 20\}.$$

(Note that if $M|m$ then B_M is a subset of B_m.) Next, for each $m \in S$ we found those values of i ($0 \leq i < m$) for which $\chi_{(i,m)}(q) = 0$ for all prime powers q. For each such i, m we expressed $\chi_{(i,m)}$ as a linear combination of the basis elements B_m, and used the relation $\chi_{(i,m)} = 0$ to eliminate an element of B_m from the functions $G_{(6,k)}(q)$. For example the relation $\chi_{(0,6)} = 0$ gives

$$\gcd(q, 2).\gcd(q, 3) = \gcd(q, 2) + \gcd(q, 3) - 1,$$

enabling us to eliminate $\gcd(q, 2).\gcd(q, 3)$ from the functions $G_{6,k}(q)$, and the relation $\chi_{(10,14)} = 0$ gives

$$\gcd(q, 2).\gcd(q - 3, 7) = \gcd(q, 2) + \gcd(q - 3, 7) - 1,$$

enabling us to eliminate $\gcd(q, 2).\gcd(q - 3, 7)$. After these eliminations, all the coefficients in the functions $g_{(6,k)}(q)$ were linear combinations of 64 elements of B_m for $m \in S$. Note that these simplifications are not purely cosmetic. As described in Section 2.2, the functions $L_{d,k}(q)$ and $G_{d,k}(q)$ are given as quotients $\frac{f}{|G|}$ for some PORC function f, and typically we need to simplify f before it can be cleanly divided by $|G|$.

5 The key problem of the algorithm

Section 2 gives an overview of an algorithm to compute PORC polynomials for $L_{d,k}(q)$ and $G_{d,k}(q)$ for given d and k. This uses Burnside's Lemma and the action of $G = \mathrm{GL}(V)$ for $V = \mathbb{F}_q^d$ on the set $\mathcal{U}_k$ of k-dimensional subspaces of W or W^+, respectively. The remaining key problem is the determination of $A(t, \bar{t})$ for a type t of an element of G and the possible types $\bar{t}$ of its induced action on W or W^+, respectively.

We discuss this remaining key problem here in more detail with a view towards determining $L_{d,k}(q)$ and $G_{d,k}(q)$ for small d. We observe that our approach is successful for $d \leq 7$, while the case $d = 8$ is out of reach at the current time. The following table contains the numbers $Nr(d)$ of types of elements in G for $d \leq 10$.

d	1	2	3	4	5	6	7	8	9	10
$Nr(d)$	1	4	8	22	42	103	199	441	859	1784

5.1 General ideas towards computing $A(t, \bar{t})$

Vaughan-Lee [10] describes a general approach towards computing $A(t, \bar{t})$ for a fixed type t and all possibilities for types $\bar{t}$. This approach is based on applications of inclusion-exclusion principles and these can be time-consuming. His implementation [11] allows us to cut down the runtime used for inclusion-exclusion calculations via ad-hoc arguments. These have to be programmed separately for each considered type t.

The alternative implementation by Eick & Wesche [4] is generic and does not need any human interaction, but it is less efficient than Vaughan-Lee's implementation. This generic implementation allows us to compute $L_{d,k}(q)$ and $G_{d,k}(q)$ for $d \leq 5$. In the cases $d = 6$ and $d = 7$ there are a few types t for which the calculation of $A(t, \bar{t})$ is not feasible using the generic implementation. (For $d = 7$ there are 11 of these types.)
We combined the two implementations to determine $L_{d,k}(q)$ and $G_{d,k}(q)$ for $d \leq 7$. In practice, the generic method determines $A(t, \bar{t})$ for as many types t as possible, while the more efficient ad-hoc method deals with the remaining difficult types.

5.2 An example of a difficult type

In this section we discuss the computation of $A(t, \bar{t})$ for the type $t = ((1, (1)), \ldots, (1, (1)))$. This is the most difficult type for both methods. We consider the action on W only; the same principles work for the action on W^+.
The rational canonical forms of the elements in G of type t are the diagonal matrices with pairwise distinct diagonal entries $a_1, \ldots, a_d$. Such a diagonal matrix A acts on W as a diagonal matrix B with diagonal entries $a_i a_j$ for $1 \leq i < j \leq d$. The type of B is determined by which coincidences $a_i a_j = a_k a_l$ hold. Since the eigenvalues of A are all distinct the equation $a_i a_j = a_k a_l$ is only possible if i, j, k, l are all distinct, and so the set R of possible equations between the eigenvalues of B has size $\frac{d(d-1)(d-2)(d-3)}{8}$.
For each subset $S \subseteq R$ let M_S be the set of diagonal matrices A over $\mathbb{F}_q$ with pairwise distinct diagonal entries $a_1, \ldots, a_d$ which satisfy the relations in S and satisfy none of the relations in $R \setminus S$. Let $g_S = |M_S|$. Higman's theory [7] implies that g_S is a PORC function of q, and an algorithm to compute g_S is described in [9], see also [5] and [10]. If $g_S \neq 0$ then all the matrices B arising from matrices $A \in M_S$ have the same type, and this type is easy to compute. Our aim is to determine g_S for all subsets $S \subseteq R$ for which $g_S \neq 0$.
One possible approach is as follows. For each subset $S \subseteq R$ let f_S be the PORC function giving the number of choices of distinct elements $a_1, \ldots, a_d \in \mathbb{F}_q$ satisfying the equations in S (and possibly also some other equations in $R \setminus S$). A method to compute f_S for given S is described in [9], see also [5] and [10]. For example, when $d = 8$ this takes somewhere between 0.05 and 0.2 seconds for each subset. For each subset $S \subseteq R$ we can then compute the PORC formula for g_S by iteratively replacing f_S by $f_S - f_T$ for each pair of subsets $S, T \subseteq R$ such that S is a proper subset of T. (This has to be done in the correct order. First we take $T = R$, then we take those T with $|T| = |R| - 1$, and next we take those T with $|T| = |R| - 2$, and so on. We can, of course, skip T whenever $f_T = 0$.) At the end of this process f_S has been replaced by g_S for all subsets S. This outline yields a practical algorithm for $d \leq 5$.

For $d \geq 6$ the algorithm sketched above is not practical since the set R of possible equations is too large. When $d = 7$, for example, $|R| = 105$ so that R has 2^{105} subsets S. Permuting the eigenvalues of A gives an action of $\mathrm{Sym}(d)$ on R, and it is only necessary to compute g_S for one element out of each orbit. But $\frac{2^{105}}{7!} \sim 8 \times 10^{27}$, so this reduction has very little impact for $d = 7$. However it turns out that $g_S = 0$ for most subsets S. More precisely, for $d = 7$ there are only 426 $\mathrm{Sym}(7)$-orbits with $g_S \neq 0$. For many subsets S it is possible to predict that $g_S = 0$. We exhibit two examples.

(a) If S contains both $a_1a_2 = a_3a_4$ and $a_1a_2 = a_3a_5$, then $f_S = g_S = 0$, since the two equations imply that $a_4 = a_5$ and this is impossible for the type t.

(b) If S contains $a_1a_2 = a_3a_4$ and $a_1a_2 = a_5a_6$ then $g_S = 0$ unless S also contains $a_3a_4 = a_5a_6$. In this case $a_3a_4 = a_5a_6$ is a consequence of the first two equations.

It is easy to decide whether a relation $a_i = a_j$ is a consequence of the relations in S. Just compute the PORC formula for the number of choices of $a_1, a_2, \ldots, a_d$ satisfying the relations in S, and also compute the PORC formula for the number of choices satisfying the relations in $S \cup \{a_i = a_j\}$. If the two PORC formulae are the same then $a_i = a_j$ is a consequence of the relations in S. Similarly it is easy to see whether any relations in $R \backslash S$ are consequences of the relations in S.

Our algorithm only generates $\mathrm{Sym}(d)$-orbits of subsets S of R such that the relations in S do not have any consequences $a_i = a_j$, and such that no relation in $R \backslash S$ is a consequence of the relations in S. If $d = 7$, then this reduces the set of all orbits of subsets to only 483 orbits. It now remains to compute f_S and then g_S for these orbits.

If $d = 8$, then there are 9288 orbits of subsets $S \subseteq R$ where we are unable to prove that $g_S \neq 0$. It took several hours of CPU-time to find representatives for these orbits. We estimate that it would take about a month of CPU-time to compute the functions g_S for these representatives. We have not attempted to complete this calculation. Even if we did, it would only enable us to compute $A(t, \bar{t})$ for matrices of type $t = (1, (1)), \ldots, (1, (1)))$, and it would take several more months of CPU-time to compute $A(t, \bar{t})$ for all possible types t. Further, the action on W^+ would be much more time-consuming to process than the action on W.

The timings mentioned above are for programs running in MAGMA V2.19-10 on a desktop computer with an Intel Core I7-4770 CPU.

6 Explicit polynomials for small d

In this section we exhibit the PORC polynomials (or their leading terms) for $L_{d,k}(q)$ and $G_{d,k}(q)$ for $d \leq 7$. The full PORC polynomials are available in electronic form in the package [3] based on GAP [8] and code [11] based on Magma [1].

6.1 Polynomials for L

Let $l = d(d-1)/2$ and note that $L_{d,l}(q) = 1$ and $L_{d,k}(q) = L_{d,l-k}(q)$. Thus it is sufficient to list $L_{d,k}(q)$ for $1 \leq k \leq l/2$ and the case $d \in \{1, 2\}$ is trivial. The next table lists PORC polynomials (or their leading terms) for $L_{d,k}(q)$ for $3 \leq d \leq 7$ and $1 \leq k \leq l/2$. The PORC polynomials are valid for all prime powers q.

$$L_{3,1}(q) = 1$$

$$L_{4,1}(q) = 2$$

$$L_{4,2}(q) = 4$$
$$L_{4,3}(q) = 6$$
$$L_{5,1}(q) = 2$$
$$L_{5,2}(q) = 6$$
$$L_{5,3}(q) = 22$$
$$L_{5,4}(q) = 57$$
$$L_{5,5}(q) = 3q + 2(q,2) + (q,3) + 2(q-1,3) + (q-1,4) + 63$$
$$L_{6,1}(q) = 3$$
$$L_{6,2}(q) = 14$$
$$L_{6,3}(q) = 3q^2 + 10q - 5(q,2) + 3(q-1,3) + 2(q-1,4) + 117$$
$$L_{6,4}(q) = q^9 + q^8 + 3q^7 + 6q^6 + 13q^5 + (26-(q,2))q^4 + (59-6(q,2))q^3 + \dots$$
$$L_{6,5}(q) = q^{15} + q^{14} + 3q^{13} + 5q^{12} + 10q^{11} + 16q^{10} + 29q^9 + 45q^8 + \dots$$
$$L_{6,6}(q) = q^{19} + q^{18} + 3q^{17} + 5q^{16} + 10q^{15} + 16q^{14} + 29q^{13} + 43q^{12} + \dots$$
$$L_{6,7}(q) = q^{21} + q^{20} + 3q^{19} + 5q^{18} + 10q^{17} + 16q^{16} + 29q^{15} + 44q^{14} + \dots$$
$$L_{7,1}(q) = 3$$
$$L_{7,2}(q) = 20$$
$$L_{7,3}(q) = q^6 + q^5 + 5q^4 + 10q^3 + (38-2(q,2))q^2 + \dots$$
$$L_{7,4}(q) = q^{20} + q^{19} + 3q^{18} + 5q^{17} + 10q^{16} + 15q^{15} + 27q^{14} + 40q^{13} + \dots$$
$$L_{7,5}(q) = q^{32} + q^{31} + 3q^{30} + 5q^{29} + 10q^{28} + 16q^{27} + 28q^{26} + 43q^{25} + \dots$$
$$L_{7,6}(q) = q^{42} + q^{41} + 3q^{40} + 5q^{39} + 10q^{38} + 16q^{37} + 29q^{36} + 44q^{35} + \dots$$
$$L_{7,7}(q) = q^{50} + q^{49} + 3q^{48} + 5q^{47} + 10q^{46} + 16q^{45} + 29q^{44} + 45q^{43} + \dots$$
$$L_{7,8}(q) = q^{56} + q^{55} + 3q^{54} + 5q^{53} + 10q^{52} + 16q^{51} + 29q^{50} + 45q^{49} + \dots$$
$$L_{7,9}(q) = q^{60} + q^{59} + 3q^{58} + 5q^{57} + 10q^{56} + 16q^{55} + 29q^{54} + 45q^{53} + \dots$$
$$L_{7,10}(q) = q^{62} + q^{61} + 3q^{60} + 5q^{59} + 10q^{58} + 16q^{57} + 29q^{56} + 45q^{55} + \dots$$

6.2 Polynomials for G

Let $l = d(d+1)/2$ and note that $G_{d,l}(q) = 1$ and $G_{d,k}(q) = G_{d,l-k}(q)$. Thus it is sufficient to list $G_{d,k}(q)$ for $1 \le k \le l/2$ and the case $d = 1$ is trivial. The next table lists PORC polynomials (or their leading terms) for $G_{d,k}(q)$ for $3 \le d \le 7$ and $1 \le k \le l/2$. The PORC polynomials are valid for all prime powers q.

$$G_{2,1}(q) = 3$$
$$G_{3,1}(q) = 4$$
$$G_{3,2}(q) = q + (15 - (q,2))$$
$$G_{3,3}(q) = 3q + (30 - 3(q,2))$$
$$G_{4,1}(q) = 6$$

$$G_{4,2}(q) = 4q + (50 - 2(q,2))$$
$$G_{4,3}(q) = q^5 + 2q^4 + 7q^3 + (26 - (q,2))q^2 + (98 - 10(q,2) + (q-1,3))q + \dots$$
$$G_{4,4}(q) = q^8 + 2q^7 + 5q^6 + 10q^5 + 24q^4 + (56 - 3(q,2))q^3 + \dots$$
$$G_{4,5}(q) = q^9 + 2q^8 + 5q^7 + 10q^6 + 21q^5 + (45 - (q,2))q^4 + (102 - 8(q,2))q^3 + \dots$$
$$G_{5,1}(q) = 7$$
$$G_{5,2}(q) = q^2 + 15q + \frac{1}{2}(267 - 16(q,2) - (q,3))$$
$$G_{5,3}(q) = q^{11} + 2q^{10} + 5q^9 + 10q^8 + 20q^7 + 38q^6 + (76 - 2(q,2))q^5 + \dots$$
$$G_{5,4}(q) = q^{19} + 2q^{18} + 5q^{17} + 10q^{16} + 20q^{15} + 35q^{14} + 61q^{13} + 99q^{12} + \dots$$
$$G_{5,5}(q) = q^{25} + 2q^{24} + 5q^{23} + 10q^{22} + 20q^{21} + 36q^{20} + 63q^{19} + 104q^{18} + \dots$$
$$G_{5,6}(q) = q^{29} + 2q^{28} + 5q^{27} + 10q^{26} + 20q^{25} + 36q^{24} + 64q^{23} + 106q^{22} + \dots$$
$$G_{5,7}(q) = q^{31} + 2q^{30} + 5q^{29} + 10q^{28} + 20q^{27} + 36q^{26} + 64q^{25} + 107q^{24} + \dots$$
$$G_{6,1}(q) = 9$$
$$G_{6,2}(q) = q^3 + 7q^2 + (54 - (q,2))q + \frac{1}{2}(682 - 50(q,2) - (q,3) + 4(q-1,3))$$
$$G_{6,3}(q) = q^{18} + 2q^{17} + 5q^{16} + 10q^{15} + 19q^{14} + 34q^{13} + 60q^{12} + 100q^{11} + 166q^{10} + \dots$$
$$G_{6,4}(q) = q^{32} + 2q^{31} + 5q^{30} + 10q^{29} + 20q^{28} + 35q^{27} + 62q^{26} + 101q^{25} + 164q^{24} + \dots$$
$$G_{6,5}(q) = q^{44} + 2q^{43} + 5q^{42} + 10q^{41} + 20q^{40} + 36q^{39} + 64q^{38} + 106q^{37} + 174q^{36} + \dots$$
$$G_{6,6}(q) = q^{54} + 2q^{53} + 5q^{52} + 10q^{51} + 20q^{50} + 36q^{49} + 65q^{48} + 108q^{47} + 179q^{46} + \dots$$
$$G_{6,7}(q) = q^{62} + 2q^{61} + 5q^{60} + 10q^{59} + 20q^{58} + 36q^{57} + 65q^{56} + 109q^{55} + 181q^{54} + \dots$$
$$G_{6,8}(q) = q^{68} + 2q^{67} + 5q^{66} + 10q^{65} + 20q^{64} + 36q^{63} + 65q^{62} + 109q^{61} + 182q^{60} + \dots$$
$$G_{6,9}(q) = q^{72} + 2q^{71} + 5q^{70} + 10q^{69} + 20q^{68} + 36q^{67} + 65q^{66} + 109q^{65} + 182q^{64} + \dots$$
$$G_{6,10}(q) = q^{74} + 2q^{73} + 5q^{72} + 10q^{71} + 20q^{70} + 36q^{69} + 65q^{68} + 109q^{67} + 182q^{66} + \dots$$
$$G_{7,1}(q) = 10$$
$$G_{7,2}(q) = q^4 + 7q^3 + (31 - (q,2))q^2 + \frac{1}{2}(327 - 18(q,2) + (q,3) + 2(q-1,3))q + \dots$$
$$G_{7,3}(q) = q^{26} + 2q^{25} + 5q^{24} + 10q^{23} + 19q^{22} + 33q^{21} + 58q^{20} + 95q^{19} + 155q^{18} + \dots$$
$$G_{7,4}(q) = q^{47} + 2q^{46} + 5q^{45} + 10q^{44} + 20q^{43} + 35q^{42} + 62q^{41} + 102q^{40} + 166q^{39} + \dots$$
$$G_{7,5}(q) = q^{66} + 2q^{65} + 5q^{64} + 10q^{63} + 20q^{62} + 36q^{61} + 64q^{60} + 107q^{59} + 176q^{58} + \dots$$
$$G_{7,6}(q) = q^{83} + 2q^{82} + 5q^{81} + 10q^{80} + 20q^{79} + 36q^{78} + 65q^{77} + 109q^{76} + 181q^{75} + \dots$$
$$G_{7,7}(q) = q^{98} + 2q^{97} + 5q^{96} + 10q^{95} + 20q^{94} + 36q^{93} + 65q^{92} + 110q^{91} + 183q^{90} + \dots$$
$$G_{7,8}(q) = q^{111} + 2q^{110} + 5q^{109} + 10q^{108} + 20q^{107} + 36q^{106} + 65q^{105} + 110q^{104} + \dots$$
$$G_{7,9}(q) = q^{122} + 2q^{121} + 5q^{120} + 10q^{119} + 20q^{118} + 36q^{117} + 65q^{116} + 110q^{115} + \dots$$
$$G_{7,10}(q) = q^{131} + 2q^{130} + 5q^{129} + 10q^{128} + 20q^{127} + 36q^{126} + 65q^{125} + 110q^{124} + \dots$$
$$G_{7,11}(q) = q^{138} + 2q^{137} + 5q^{136} + 10q^{135} + 20q^{134} + 36q^{133} + 65q^{132} + 110q^{131} + \dots$$
$$G_{7,12}(q) = q^{143} + 2q^{142} + 5q^{141} + 10q^{140} + 20q^{139} + 36q^{138} + 65q^{137} + 110q^{136} + \dots$$
$$G_{7,13}(q) = q^{146} + 2q^{145} + 5q^{144} + 10q^{143} + 20q^{142} + 36q^{141} + 65q^{140} + 110q^{139} + \dots$$

$$G_{7,14}(q) = q^{147} + 2q^{146} + 5q^{145} + 10q^{144} + 20q^{143} + 36q^{142} + 65q^{141} + 110q^{140} + \dots$$

We observe that the coefficients of the leading terms of $L_{d,k}(q)$ and $G_{d,k}(q)$ for fixed d and large enough k exhibit a uniform pattern. This observation as well as upper bounds for the degree of the PORC polynomials still have to be investigated.

References

[1] W. Bosma, J. Cannon, and C. Playoust. The MAGMA algebra system I: The user language. *J. Symb. Comput.*, 24:235 – 265, 1997.

[2] B. Eick and E. A. O'Brien. Enumerating p-groups. *J. Austral. Math. Soc.*, 67:191 – 205, 1999.

[3] B. Eick, M. Vaughan-Lee, and M. Wesche. *PORC - Computing with PORC polynomials*, 2018. `http://www.icm.tu-bs.de/~beick/soft/`.

[4] B. Eick and M. Wesche. *Classtwoalg - Enumeration of class two algebras*, 2018. A GAP package available from `http://www.icm.tu-bs.de/~morwesch/research/research.html`.

[5] B. Eick and M. Wesche. Enumeration of nilpotent associative algebras of class 2 over arbitrary finite fields. *J. Algebra*, 503:573–589, 2018.

[6] J. A. Green. The characters of the finite general linear groups. *Trans. Amer. Math. Soc.*, 80:402–447, 1955.

[7] G. Higman. Enumerating p-groups. II: Problems whose solution is porc. *Proc. London Math. Soc.*, 10:566 – 582, 1960.

[8] The GAP Group. GAP – *Groups, Algorithms and Programming, Version 4.4*. Available from `http://www.gap-system.org`, 2005.

[9] M. Vaughan-Lee. Choosing elements from finite fields. *ArXiv*, 2012.

[10] M. Vaughan-Lee. On Graham Higman's famous PORC paper. *Int. J. Group Theory*, 1(4):65–79, 2012.

[11] M. Vaughan-Lee. Magma code. `http://users.ox.ac.uk/~vlee/PORC/pclasstwogroups`, 2018.