

COMPUTATIONS CONCERNING THE CLASSIFICATION OF EXTREMELY PRIMITIVE GROUPS

TIMOTHY C. BURNES AND ADAM R. THOMAS

ABSTRACT. In this note we provide a record of the MAGMA and GAP computations used in the paper [3] on the classification of extremely primitive groups. Throughout, we freely adopt the notation and terminology in [3].

1. CODE AND EXAMPLES

Let $G \leq \text{Sym}(\Omega)$ be an almost simple primitive permutation group with socle G_0 and point stabilizer $H = G_\sigma$. Recall that G is *extremely primitive* if H acts primitively on each of its orbits in $\Omega \setminus \{\alpha\}$. Also recall that the *base size* of G , denoted by $b(G, H)$ is the minimal size of a subset B of Ω with the property that the pointwise stabilizer of B in G is trivial. As noted in [3, Lemma 2.5], if $b(G, H) = 2$ then G is not extremely primitive.

In this section we present the MAGMA [1] and GAP [4] code we used to establish the main computational results in [3]. We also include several concrete examples.

1.1. Extremely primitive. The MAGMA function `ExtremelyPrimitive` takes a permutation group G as input. Then for each representative H of a conjugacy class of core-free maximal subgroups of G , the function attempts to find a random conjugate H^g such that $H \cap H^g < H$ is non-maximal (the parameter n in the second line is an upper bound on the number of random searches it will perform; typically we set $n = 100$). The function outputs a list of triples $[a, b, c]$, where b is the order of the a -th conjugacy class representative H and c is either 0 or 1. More precisely, $c = 1$ if and only if the random search succeeded in finding a conjugate H^g such that $H \cap H^g < H$ is non-maximal. In particular, if $c = 1$ then we can conclude that the action of G on G/H is not extremely primitive.

```
ExtremelyPrimitive:= function(G)
  n:=100;
  M:=MaximalSubgroups(G);
  A:=[];
  for a in [1..#M] do
    H:=M[a] 'subgroup;
    b:=#H;
    c:=0;
    if #Core(G,H) eq 1 then
      for j in [1..n] do
        g:=Random(G);
        if (g in H eq false) and (IsMaximal(H,H meet H^g) eq false) then
          c:=1;
          break;
        end if;
      end for;
      Append(~A, [a, b, c]);
    end if;
  end for;
```

Date: October 18, 2020.

```

return A;
end function;

```

For the almost simple groups we are interested in, we will typically construct $\text{Aut}(G_0)$ as a permutation group by using the command `AutomorphismGroupSimpleGroup` and we can then identify G as a subgroup of $\text{Aut}(G_0)$.

Example 1.1. The following shows that $G_2(3)$ has no extremely primitive actions:

```

A:=AutomorphismGroupSimpleGroup("G2",3);
G:=Socle(A);
ExtremelyPrimitive(G);

```

which returns

```

[
  [ 1, 1092, 1 ],
  [ 2, 1512, 1 ],
  [ 3, 576, 1 ],
  [ 4, 1344, 1 ],
  [ 5, 11232, 1 ],
  [ 6, 11232, 1 ],
  [ 7, 11664, 1 ],
  [ 8, 11664, 1 ],
  [ 9, 12096, 1 ],
  [ 10, 12096, 1 ]
]

```

and the result follows.

Example 1.2. If we run the same code for $G = G_2(4)$, then we get

```

[
  [ 1, 1092, 1 ],
  [ 2, 3600, 1 ],
  [ 3, 12096, 1 ],
  [ 4, 124800, 1 ],
  [ 5, 120960, 1 ],
  [ 6, 604800, 0 ],
  [ 7, 184320, 1 ],
  [ 8, 184320, 1 ]
]

```

and so the 6th subgroup H in this list requires further attention (here $H = J_2$). To resolve this case, we use `CosetAction` to explicitly construct G as a permutation group on the set of cosets G/H and we then inspect each nontrivial H -orbit in turn:

```

A:=AutomorphismGroupSimpleGroup("G2",4);
G:=Socle(A);
M:=MaximalSubgroups(G);
H:=M[6] 'subgroup;
f,L,K:=CosetAction(G,H);
H:=Stabilizer(L,1);
o:=Orbits(H);
for i in [2..#o] do
  [i,#o[i]];
  IsPrimitive(H,o[i]);
end for

```

```
" ";
end for;
```

This returns

```
[ 2, 100 ]
true
```

```
[ 3, 315 ]
true
```

and we conclude that the action of G on G/H is extremely primitive (in addition, we see that the orbits of H have size 1, 100 and 315). In the same way, we deduce that $G_2(4).2$ acts extremely primitively on the set of cosets of $J_2.2$. These two groups feature in the statement of [3, Theorem 1].

1.2. Base-two groups. The `BaseTwo` function below takes G as input as before. Then for each representative H of a class of core-free maximal subgroups the function seeks by random search to find an element $g \in G$ with $H \cap H^g = 1$. As before, the parameter n in the second line is an upper bound on the number of random searches the function performs (we usually set $n = 100$). Since $b(G, H) \geq 3$ if $|H|^2 > |G|$, the function only considers subgroups with $|H|^2 \leq |G|$. The output is a list of triples $[a, b, c]$, one for each class of maximal subgroups H , where a is a counter, b is the order of H and $c = 1$ if and only if the random search finds an element $g \in G$ with $H \cap H^g = 1$, otherwise $c = 0$. In particular, if $c = 1$ then $b(G, H) = 2$.

```
BaseTwo:= function(G)
  n:=100;
  M:=MaximalSubgroups(G);
  A:=[];
  for a in [1..#M] do
    H:=M[a]`subgroup;
    b:=#H;
    c:=0;
    if (#Core(G,H) eq 1) and (#H^2 le #G) then
      for j in [1..n] do
        g:=Random(G);
        if #(H meet H^g) eq 1 then
          c:=1;
          break;
        end if;
      end for;
      Append(~A, [a, b, c]);
    end if;
  end for;
  return A;
end function;
```

Example 1.3. Suppose $G = G_2(3)$. Then the following code gives the output below:

```
A:=AutomorphismGroupSimpleGroup("G2", 3);
G:=Socle(A);
BaseTwo(G);
```

```
[
  [ 1, 1092, 1 ],
```

```

    [ 2, 1512, 1 ],
    [ 3, 576, 1 ],
    [ 4, 1344, 0 ]
]

```

So for the first three subgroups in this list we conclude that $b(G, H) = 2$. However, the final subgroup H with $|H| = 1344$ requires further attention. Here we explicitly construct G as a permutation group on G/H and we determine the lengths of the nontrivial H -orbits:

```

A:=AutomorphismGroupSimpleGroup("G2",3);
G:=Socle(A);
M:=MaximalSubgroups(G);
H:=M[4] ' subgroup;
f,L,K:=CosetAction(G,H);
H:=Stabilizer(L,1);
o:=Orbits(H);
#H in [#o[i] : i in [1..#o]];

```

This returns `false`, so H does not have a regular orbit on G/H and thus $b(G, H) \geq 3$. By random search, it is easy to find elements $g_1, g_2 \in G$ such that $H \cap H^{g_1} \cap H^{g_2} = 1$, which implies that $b(G, H) = 3$ in this case. Alternatively, `#Stabilizer(H, [2,3])` returns 1 and thus $b(G, H) \leq 3$.

1.3. Variations. Notice that the above functions `ExtremelyPrimitive` and `BaseTwo` both utilise the `MaximalSubgroups` command to calculate a complete set of representatives of the conjugacy classes of maximal subgroups of G . This is somewhat restrictive. Indeed, there are situations when we want to work with a specific maximal subgroup of G , but `MaximalSubgroups` is not available. In these cases, we can often construct H directly and then implement variations of the previous routines.

Example 1.4. Suppose $G = F_4(2)$ and $H = 3.U_3(2)^2.3.2$. Here the maximal subgroups of G are not available in MAGMA, but we can construct H by observing that $H = N_G(C_G(x))$ for an element $x \in G$ of order 3 with $|C_G(x)| = 9|U_3(2)|^2 = 46656$. Indeed, we can use the following code to show that $b(G, H) = 2$:

```

A:=AutomorphismGroupSimpleGroup("F4",2);
G:=Socle(A);
repeat
  x:=Random(G);
until (Order(x) mod 3 eq 0) and
  (#Centralizer(G,x^(Order(x) div 3)) eq 46656);
H:=Normalizer(G,Centralizer(G,x^(Order(x) div 3)));
repeat
  g:=Random(G);
until #(H meet H^g) eq 1;

```

In the same way, we deduce that $b(G, H) = 2$ for the action of $G = F_4(2).2$ on the set of cosets of $H = 3.U_3(2)^2.3.2.2$.

Example 1.5. Suppose $G = F_4(2)$ and $H = L_3(2)^2.2$. Here we use the fact that $L_3(2)$ is generated by elements a and b with $a^2 = b^3 = (ab)^7 = 1$. Indeed, by random search we can find elements $a, b \in G$ satisfying these relations such that $L = \langle a, b \rangle \cong L_3(2)$ and $H = N_G(L)$. We can then check that $b(G, H) = 2$ in the usual manner.

```

A:=AutomorphismGroupSimpleGroup("F4",2);
G:=Socle(A);
repeat

```

```

a:=Random(G);
b:=Random(G);
until (Order(a) mod 2 eq 0) and (Order(b) mod 3 eq 0)
  and (Order(a^(Order(a) div 2)*b^(Order(b) div 3)) eq 7)
  and (#Normalizer(G,sub<G|a^(Order(a) div 2),b^(Order(b) div 3)>)
  eq 2*#PSL(3,2)^2);
H:=Normalizer(G,sub<G|a^(Order(a) div 2),b^(Order(b) div 3)>);
repeat
  g:=Random(G);
until #(H meet H^g) eq 1;

```

1.4. **Character tables.** Suppose the character tables of G and H are available in the GAP Character Table Library [2]. If the fusion map from H -classes to G -classes is also stored, then we can use GAP to compute $|x^G \cap H|$ precisely for all $x \in G$. We can then proceed in one of two ways:

- (a) We can use the Orbit Counting Lemma to compute the rank of G and then seek to rule out extreme primitivity via [3, Lemma 2.2] by inspecting the indices of a set of representatives of the conjugacy classes of maximal subgroups of H (which we may be able to compute using MAGMA).
- (b) In a different direction, we can use the information from the character tables to compute

$$\mathcal{Q}(G, H) = \sum_{i=1}^k \frac{|x_i^G \cap H|^2}{|x_i^G|} \quad (1.1)$$

precisely, where x_1, \dots, x_k represent the distinct G -classes of elements of prime order. Then [3, Lemma 2.6] implies that $b(G, H) = 2$ if $\mathcal{Q}(G, H) < 1$.

Example 1.6. Suppose $G = {}^2E_6(2)$ and $H = \Omega_{10}^-(2)$. We claim that G is not extremely primitive. To see this, we first observe that the character tables of G and H , together with the corresponding fusion map, are available in [2]:

```

LoadPackage( "CTblLib" );
t:=CharacterTable("2E6(2)");;
t1:=CharacterTable("O10-(2)");;
fus:=FusionConjugacyClasses(t1,t);;

```

Next we compute the rank of G by implementing the Orbit Counting Lemma:

```

S:=SizesConjugacyClasses(t);;
S1:=SizesConjugacyClasses(t1);;
a:=Size(t)/Size(t1);;
w:=0;;
for i in [1..Size(fus)] do
  b:=S[fus[i]];
  z:=0;;
  for j in [1..Size(fus)] do
    if fus[j] = fus[i] then
      z:=z+S1[j];;
    fi;
  od;
  w:=w+S1[i]*z*a/b;;
od;
w/Size(t1);

```

This returns 13, which is the rank of G . We now switch to MAGMA in order to compute the indices of a set of representatives of the classes of maximal subgroups of H .

```
H:=POmegaMinus(10,2);
M:=MaximalSubgroups(H);
A:=[ #H/#M[i]'subgroup: i in [1..#M]];
a,b:=Maximum(A);
a;
```

It follows that $|H : M| \leq 53616640$ for every maximal subgroup M of H and we conclude that G is not extremely primitive since $1 + 12.53616640 < |G : H|$.

As the next example demonstrates, this approach can still be effective even if the fusion map is not available in [2].

Example 1.7. Suppose $G = {}^2E_6(2).2$ and $H = O_{10}^-(2)$. Here the character tables are available, but the fusion map is not stored. However, we can use the GAP command `PossibleClassFusions` to determine the set of possible fusion maps. We find that there are two such maps and by calculating as above one can check that the rank of G is independent of the choice of fusion map; it is 12 in both cases.

```
t:=CharacterTable("2E6(2).2");;
t1:=CharacterTable("O10-(2).2");;
S:=SizesConjugacyClasses(t);;
S1:=SizesConjugacyClasses(t1);;
a:=Size(t)/Size(t1);;
F:=PossibleClassFusions(t1,t);;
for k in [1..Size(F)] do
  fus:=F[k];;
  w:=0;;
  for i in [1..Size(fus)] do
    b:=S[fus[i]];
    z:=0;;
    for j in [1..Size(fus)] do
      if fus[j] = fus[i] then
        z:=z+S1[j];;
      fi;
    od;
    w:=w+S1[i]*z*a/b;;
  od;
  Print("\n", [k,w/Size(t1)]);
od;
```

We can now proceed as before to show that G is not extremely primitive, noting that $|H : M| \leq 263208960$ for every maximal subgroup M of H .

Example 1.8. Suppose $G = {}^2E_6(2)$ and $H = \Omega_7(3)$. We claim that $b(G, H) = 2$. To see this, we access the character tables of G and H in [2], and we find that there are 12 possible fusion maps. For each choice of fusion map we compute $\mathcal{Q}(G, H)$ (see (1.1)) and we find that this is independent of the choice of fusion map. In particular,

$$\mathcal{Q}(G, H) = \frac{1946811337}{5927075840} < 1$$

and thus $b(G, H) = 2$ as claimed.

```
t:=CharacterTable("2E6(2)");;
t1:=CharacterTable("O7(3)");;
```

```

S:=SizesConjugacyClasses(t);;
S1:=SizesConjugacyClasses(t1);;
O:=OrdersClassRepresentatives(t);;
F:=PossibleClassFusions(t1,t);;
for k in [1..Size(F)] do
  fus:=F[k];;
  w:=0;;
  for i in [1..Size(O)] do
    if IsPrime(O[i]) then
      a:=S[i];;
      b:=0;;
      for j in [1..Size(fus)] do
        if fus[j] = i then
          b:=b+S1[j];;
        fi;
      od;
      w:=w+(b^2)/a;;
    fi;
  od;
  Print("\n", [k,w]);
od;

```

1.5. Lie type computations. In this section we explain how we can work with MAGMA's inbuilt functions for computing with groups of Lie type to construct certain subgroups of exceptional groups. We can then obtain information on their conjugacy classes and the action of class representatives on suitable modules (typically the adjoint module), which allows us to estimate $\mathcal{Q}(G, H)$.

To begin with, let us first present some general code for obtaining the sort of information we require on conjugacy classes. Let G be a subgroup of $GL(V)$, where V is a finite dimensional vector space over a finite field of characteristic p . Recall that if $x \in G$ has prime order r , then x is *semisimple* if $r \neq p$ and *unipotent* if $r = p$. We will use the MAGMA function `Classes` to determine the sizes of the conjugacy classes of G and a set of class representatives. We then use the functions `SInfo` and `UInfo` described below to obtain the desired information on class representatives. Note that both functions take as input the group G (given as a subgroup of $GL(V)$) and the output from `Classes(G)`.

Each row of output of `SInfo` is of the form

$$[r, d], [c_1, \dots, c_n], c$$

where $r \neq p$ is a prime, d is a non-negative integer and the c_i are the sizes of the G -classes x_i^G of semisimple elements with $|x_i| = r$ and $\dim C_V(x_i) = d$ for each i . In addition, $c = \sum_i c_i$. There is one row of output for each pair $[r, d]$.

```

SInfo := function(G,cl)
A:=[];
for i in [1..#cl] do
  if IsPrime(cl[i][1]) and IsSemisimple(cl[i][3]) then
    Append(~A, [cl[i][1], Dimension(Eigenspace(cl[i][3],1)), cl[i][2]]);
  end if;
end for;
B:={ [A[i][1], A[i][2]] : i in [1..#A] };
B:=Sort([i : i in B]);
for i in [1..#B] do
  C:=[];
  D:=[];

```

```

for j in [1..#A] do
  if [A[j][1],A[j][2]] eq B[i] then
    Append(~C,A[j]);
    Append(~D,A[j][3]);
  end if;
end for;
printf "%o, %o, %o\n", B[i], D, &+D;
end for;

return " ";
end function;

```

For unipotent elements $x \in G$, we use `JordanForm` to find the Jordan form of x on V and then the following function `Jform` converts the output of `JordanForm` so that the Jordan block structure is transparent.

```

function Jform(j)
  newjform:=[];
  newjform[1]:=[j[1][2],0];
  cur:=j[1];
  curnum:=1;
  for x in j do
    if x eq cur then
      newjform[curnum][2]+:=1;
    else
      cur:=x;
      curnum+=1;
      newjform[curnum]:=[x[2],1];
    end if;
  end for;
  return Reverse(newjform);
end function;

```

The function `UInfo` outputs rows of the form

$$\begin{array}{c} [a_1, b_1], \dots, [a_k, b_k] \\ [c_1, \dots, c_n] \\ c \end{array}$$

where the a_i, b_i are positive integers such that $\sum_i a_i b_i = \dim V$ (so $(a_1^{b_1}, \dots, a_k^{b_k})$ is a partition of $\dim V$) and the c_i are the sizes of the G -classes x_i^G of unipotent elements of order p with Jordan form $(J_{a_1}^{b_1}, \dots, J_{a_k}^{b_k})$. Here J_i denotes a standard unipotent Jordan block of size i and b_i is the multiplicity of J_{a_i} in the Jordan form. In addition, $c = \sum_i c_i$. There is one row of output for each partition $(a_1^{b_1}, \dots, a_k^{b_k})$.

```

UInfo := function(G,c1)
  A:=[];
  T:=[];
  for i in [1..#c1] do
    if IsPrime(c1[i][1]) and IsUnipotent(c1[i][3]) then
      j1,j2,j3:=JordanForm(c1[i][3]);
      J:=[r: r in Jform(j3)];
      Append(~A,J);
      Append(~T,c1[i][2]);
    end if;
  end for;
end function;

```

```

end for;
B:={ A[i] : i in [1..#A] };
B:=Sort([i : i in B]);
for i in [1..#B] do
  C:=[];
  D:=[];
  for j in [1..#A] do
    if A[j] eq B[i] then
      Append(~C,A[j]);
      Append(~D,T[j]);
    end if;
  end for;
  for j in [1..#B[i]] do printf "%o", B[i][j]; end for; " ";
  D; &+D; " ";
end for;

return " ";
end function;

```

Example 1.9. Suppose $G = \text{Sp}_8(2)$, with V the natural module. Then the output from the functions `SInfo` and `UInfo` is as follows:

```

G:=Sp(8,2);
cl:=Classes(G);

SInfo(G,cl);

[ 3, 0 ], [ 609280 ], 609280
[ 3, 2 ], [ 12185600 ], 12185600
[ 3, 4 ], [ 3655680 ], 3655680
[ 3, 6 ], [ 10880 ], 10880
[ 5, 0 ], [ 157925376 ], 157925376
[ 5, 4 ], [ 13160448 ], 13160448
[ 7, 2 ], [ 1128038400 ], 1128038400
[ 17, 0 ], [ 2786918400, 2786918400 ], 5573836800

UInfo(G,cl);

[ 2, 1 ][ 1, 6 ]
[ 255 ]
255

[ 2, 2 ][ 1, 4 ]
[ 5355, 16065 ]
21420

[ 2, 3 ][ 1, 2 ]
[ 321300 ]
321300

[ 2, 4 ]
[ 64260, 963900 ]
1028160

```

So for example, $\mathrm{Sp}_8(2)$ has two classes of unipotent elements with Jordan form (J_2^4) on the natural module, with respective sizes 64260 and 963900.

Now let us turn to the construction of certain maximal rank subgroups of exceptional groups of Lie type. For more details on these subgroups and the theory underpinning the constructions below, we refer the reader to [6]. Here we briefly recall the general set up.

Let G_0 be a simple exceptional group of Lie type over \mathbb{F}_q and let \bar{G} be a simple adjoint algebraic group over $\bar{\mathbb{F}}_q$ such that $\bar{G}_\sigma = \mathrm{Inndiag}(G_0)$ for an appropriate Steinberg endomorphism σ of \bar{G} . We are primarily interested in the case where $\bar{H} = N_{\bar{G}}(\bar{T})$ is the normalizer of a σ -stable maximal torus \bar{T} of \bar{G} , so $\bar{H} = \bar{T}.W(\bar{G})$ and $W(\bar{G})$ is the Weyl group of \bar{G} . As explained in [6, Section 1], σ acts on $W(\bar{G})$ and there is a bijection from the set of σ -classes of $W(\bar{G})$ to the set of conjugacy classes of maximal tori in \bar{G}_σ (recall that the σ -class of $w \in W(\bar{G})$ is the subset $\{x^{-1}wx^\sigma : x \in W(\bar{G})\}$). In the cases we are interested in, the σ -classes are precisely the conjugacy classes of $W(\bar{G})$.

Suppose we want to construct the normalizer of a maximal torus of \bar{G}_σ corresponding to the class of $w \in W(\bar{G})$. Then it is convenient to work with the subgroup $\bar{T}_{\sigma w} < \bar{G}_{\sigma w}$, where by a slight abuse of notation we use σw to represent the composition of σ and the inner automorphism of \bar{G} induced by a lift of w . Then the corresponding subgroup in $\bar{G}_{\sigma w}$ is $\bar{T}_{\sigma w}.C_{W(\bar{G})}(\sigma w)$. In all but one case (see Example 1.11), the map σw acts on \bar{T} via $s \mapsto s^{-q}$ for $s \in \bar{T}$. We will then construct $\bar{T}_{\sigma w}$ as a subgroup of \bar{G}_{σ^2} , which is a group defined over \mathbb{F}_{q^2} . For $\bar{G} = E_6$, we may also want to consider the normalizer of a maximal torus in the group $\bar{G}_{\sigma w}.2$, where we have extended by a graph automorphism. To construct these subgroups we will work in $\mathrm{Inndiag}(E_7(q^2))$ since it has a Levi subgroup containing $\mathrm{Inndiag}(E_6(q^2)).2$.

To make these constructions, we will use the Lie type machinery in MAGMA. First we use the function `GroupOfLieType` to construct the appropriate group $G = \bar{G}_{\sigma^2}$ over \mathbb{F}_{q^2} and then `AdjointRepresentation` yields a map from G to $\mathrm{GL}(V)$, where V is the adjoint module for G (possibly defined over an extension of \mathbb{F}_{q^2}).

The following function `MaxRank1` takes as input a Lie type `Gp` for the ambient simple algebraic group \bar{G} , such as "E7", and the field size q . The output is a subgroup

$$H = (Z_{q+1})^r.W(\bar{G}) < \bar{G}_{\sigma^2},$$

where r is the rank of \bar{G} . If the optional setting `graph` is set to `true` then $\bar{G} = E_6$ and we construct $H = (Z_{q+1})^r.W(\bar{G}).2 < \mathrm{Inndiag}(E_7(q^2))$, which is the corresponding maximal subgroup of $\bar{G}_{\sigma^2}.2$. Here H is returned as a subgroup of $\mathrm{GL}(V)$, where V is the adjoint module for \bar{G}_{σ^2} (unless `graph` is `true`, in which case V is the adjoint module for $E_7(q^2)$).

```
function MaxRank1(Gp,q:graph:=false)
K:=GF(q^2);
G0:=GroupOfLieType(Gp,K:Isogeny=="Ad");
t:=PrimitiveElement(K);
if graph then
  G:=GroupOfLieType("E7",K:Isogeny=="Ad");
else
  G:=G0;
end if;
Ad:=AdjointRepresentation(G : NoWarning);
W:=WeylGroup(G);
if graph then
  E6Rts:=[r : r in [1..#Roots(W)] | Roots(W)[r][7] eq 0 ];
  Wgens:=Generators(Stabiliser(W,SequenceToSet(E6Rts)));
  Tgens:=[TorusTerm(G,j,t^(q-1)) : j in [1..Rank(G0)]];
else
  Wgens:=Generators(W);
```

```

Tgens:=[j^(q-1) : j in Generators(StandardMaximalTorus(G))[1..Rank(GO)]];
end if;
S:=sub<Codomain(Ad)|Ad(Tgens), Ad([elt<G|i> : i in Wgens])>;
return S;
end function;

```

Example 1.10. The following code shows how we can compute the number of involutions in the maximal subgroup $3^7.W(E_7) = 3^7.(2 \times \text{Sp}_6(2)) < E_7(2)$.

```

H:=MaxRank1("E7",2);
c1:=Classes(H);
&+[c1[i][2] : i in [1..#c1] | c1[i][1] eq 2];

```

728703

The next two examples describe some additional special cases.

Example 1.11. Here $G_0 = E_6(2)$ and H_0 is the maximal rank subgroup $7^3:3^{1+2}.\text{SL}_2(3)$. We need to construct both H_0 and $H_0.2 = N_{\text{Aut}(G_0)}(H_0)$. To do this, we use the function `MaxRank2` below, which constructs H_0 and $H_0.2$ as subgroups of $E_7(8)$. Here the input is either `true` or `false`; if the former, then $H_0.2$ is constructed, otherwise H_0 is returned. In both cases, the group is constructed as a subgroup of $\text{GL}(V)$, where V is the adjoint module for $E_7(8)$.

In terms of the general set up discussed above, H_0 corresponds to the σ -class of the Weyl group element

$$w = (1, 3, 71)(2, 116, 50)(4, 26, 95) \dots \in W(E_7),$$

where we are viewing $w \in W(E_6)$ as an element of $W(E_7)$. With respect to the standard Lie notation and a set of simple roots $\{\alpha_1, \dots, \alpha_7\}$ for E_7 (labelled in the usual way), one can then check that the toral elements

$$h_{\alpha_1}(c_1^{-2})h_{\alpha_3}(c_1), h_{\alpha_2}(c_2^{-2})h_{\alpha_4}(c_2), h_{\alpha_5}(c_3^{-2})h_{\alpha_6}(c_3)$$

with $c_i \in \mathbb{F}_8^\times$ generate $\bar{T}_{\sigma w} \cong 7^3$, where $\beta = -\alpha_1 - 2\alpha_2 - 2\alpha_3 - 3\alpha_4 - 2\alpha_5 - \alpha_6$.

```

function MaxRank2(graph)
K:=GF(8);
t:=PrimitiveElement(K);
G:=GroupOfLieType("E7",K);
W:=WeylGroup(G);
Ad:=AdjointRepresentation(G);
Tgens:=[ TorusTerm(G,1,t^-2)*TorusTerm(G,3,t),
TorusTerm(G,2,t^-2)*TorusTerm(G,116,t),
TorusTerm(G,5,t^-2)*TorusTerm(G,6,t) ];
WE6:=sub<W|[W.i : i in [1..6]]>;
E6rts:=[r : r in [1..#Roots(W)] | Roots(W)[r][7] eq 0 ];
WE6g:=Stabiliser(W,SequenceToSet(E6rts));
w:=[x : x in WE6 | 4^x eq 26 and 2^x eq 116 and Order(x) eq 3][1];
if graph then
Wgens:=[elt<G|i> : i in Generators(Centraliser(WE6g,w)) ];
else
Wgens:=[elt<G|i> : i in Generators(Centraliser(WE6,w)) ];
end if;
S := sub< Codomain(Ad) | [Ad(i) : i in Tgens cat Wgens] >;
return S;
end function;

```

Example 1.12. Suppose $G = {}^2E_6(2).3$ and $H = 3.U_3(2)^3.3^2.Sym_3$ is a maximal rank subgroup. Here H is the fixed points of a σ -stable maximal rank subgroup $A_2^3.Sym_3 < E_6$, so a slightly different approach is required for this construction. The code below constructs H as a subgroup of $E_6(4).3$.

With respect to a set of simple roots $\{\alpha_1, \dots, \alpha_6\}$ for E_6 (labelled in the usual way), we choose the subsystem subgroup $\bar{H}^0 = A_2^3$ corresponding to $\{\alpha_1, \alpha_3, \alpha_5, \alpha_6, \alpha_2, -\alpha_0\}$, where α_0 is the highest root. In the code below, the elements $\mathbf{x1}$ and $\mathbf{x2}$ generate a subgroup $3.U_3(2)$ contained in the first factor A_2 , corresponding to $\{\alpha_1, \alpha_3\}$. The elements $\mathbf{w1}$ and $\mathbf{w2}$ generate a subgroup of $W(\bar{G})$ isomorphic to Sym_3 , which normalizes \bar{H}^0 by transitively permuting the factors. It follows that the subgroup generated by $\mathbf{x1}, \mathbf{x2}, \mathbf{w1}, \mathbf{w2}$ is isomorphic to $3.U_3(2)^3.Sym_3$. The remaining elements $\mathbf{d1}, \mathbf{d2}$ are diagonal automorphisms of $3.U_3(2)^3$, which are contained in the maximal split-torus of $\text{Inndiag}(E_6(4))$.

```
K:=GF(4);
t:=PrimitiveElement(K);
G:=GroupOfLieType("E6",K:Isogeny:="Ad");
Ad:=AdjointRepresentation(G : NoWarning);
W:=WeylGroup(G);
x1:=elt<G|<1,1>,<3,1>,<7,t>,<1+36,t>,<3+36,t^2>,<7+36,t^2>>;
x2:=elt<G|<1,1>,<3,t>,<7,1>,<1+36,t^2>,<3+36,1>,<7+36,1>>;
w1:=[w: w in W | 1^w eq 6 and 6^w eq 72 and 72^w eq 1 and 2^w eq 3][1];
w2:=[w: w in W | 1^w eq 5 and 5^w eq 1 and 3^w eq 6 and 2^w eq 72][1];
Tgens:=Generators(StandardMaximalTorus(G));
d1:=Tgens[1]*Tgens[2]*Tgens[3]^2*Tgens[4]*Tgens[5];
d2:=Tgens[1]*Tgens[2]*Tgens[3]*Tgens[4]*Tgens[5]*Tgens[6]^2;
H:=sub<Codomain(Ad)|Ad([x1,x2,d1,d2]),[Ad(elt<G|i>) : i in [w1,w2]]>;
c1:=Classes(H);
&+[c1[i][2] : i in [1..#c1] | c1[i][1] eq 3];
```

492074

The above code shows that $i_3(H) = 492074$, which is the key information we need in the proof of [3, Lemma 4.12].

2. RESULTS

Here we use the computational methods discussed in the previous section to verify the relevant results stated in [3].

2.1. Small groups.

Theorem 2.1. *Let G be an almost simple primitive group with socle G_0 and point stabilizer H , where G_0 is one of*

$${}^2B_2(8), {}^2B_2(32), {}^2F_4(2)', {}^3D_4(2), G_2(2)', G_2(3), G_2(4), G_2(5).$$

Then G is extremely primitive if and only if $(G, H) = (G_2(4), J_2)$ or $(G_2(4).2, J_2.2)$.

Proof. In each case, we first construct $\text{Aut}(G_0)$ as a permutation group via the command `AutomorphismGroupSimpleGroup` and then we set $G = \text{Aut}(G_0)$ or G_0 (note that in each case, there are at most two possibilities for G). Next we check for extreme primitivity by applying the function `ExtremelyPrimitive` from Section 1.1. For $G_0 = G_2(2)' \cong U_3(3)$ we construct $\text{Aut}(G_0)$ using `PGammaU(3,3)`. This just leaves the case $G = G_2(4).\alpha$ and $H = J_2.\alpha$ with $\alpha \in \{1, 2\}$ and we show that G is extremely primitive by proceeding as in Example 1.2. \square

G_0	H_0	Constructions	Ref in [3]
$E_7(2)$	$3^7.W(E_7)$	$H_0 < E_7(4)$	Lemma 4.7
$E_6(2)$	$7^3:3^{1+2}.SL_2(3)$	$H_0.2 < E_7(8)$	Lemma 4.11
${}^2E_6(2)$	$3^5.W(E_6)$	$H_0.3 < E_6(4), H_0.Sym_3 < E_7(4)$	Lemma 4.11
${}^2E_6(3)$	$4^6.W(E_6)$	$H_0 < E_6(9), H_0.2 < E_7(9)$	Lemma 4.11
${}^2E_6(4)$	$5^6.W(E_6)$	$H_0 < E_6(16), H_0.2 < E_7(16)$	Lemma 4.11
$F_4(4)$	$5^4.W(F_4)$	$H_0 < F_4(16)$	Lemma 4.15

TABLE 1. Normalizers of maximal tori

2.2. Maximal rank subgroups.

Proposition 2.2. *If (G_0, H_0) is one of the cases in Table 1, then $b(G, H) = 2$.*

Proof. As explained in [3], in each case we need to construct H as a subgroup of $GL(V)$ for some module V for G and then get information on the sizes of the conjugacy classes of H and the action of class representatives on V . For the constructions, we use the function `MaxRank1` from Section 1.5, except for the subgroup in the second row of Table 1, where we proceed as in Example 1.11. The relevant information on conjugacy classes and class representatives is then obtained by applying the functions `SInfo` and `UInfo`.

If $x \in H$ is semisimple and V is the adjoint module for G_0 (defined over a field extension of \mathbb{F}_q) then $\dim C_V(x) = \dim C_{\bar{G}}(x)$ and we can use the output from `SInfo` to identify $\dim C_{\bar{G}}(x)$, which often allows us to determine the structure of $C_{\bar{G}}(x)^0$. In turn, this information translates into a lower bound on $|x^G|$ and an upper bound on $|x^G \cap H|$, which yields an upper bound on the contribution to $\mathcal{Q}(G, H)$ from semisimple elements. Similarly, the contribution from unipotent elements can be estimated by considering the output from `UInfo`, noting that in almost all cases the Jordan form of x on the adjoint module V uniquely determines the \bar{G} -class of x (see [5]). In the cases where $G_0 = E_6^\varepsilon(q)$ and V is the adjoint module for a group of type E_7 , we can proceed in a similar fashion, using readily available information on the fusion of $E_6.2$ -classes in E_7 .

In this way, we can estimate $\mathcal{Q}(G, H)$ and we verify the bound $\mathcal{Q}(G, H) < 1$, which gives $b(G, H) = 2$ via [3, Lemma 2.6].

In some of these cases, we only need partial information in order to derive an effective upper bound on $\mathcal{Q}(G, H)$. For example, if $G_0 = E_6(2)$ and $H_0 = 7^3:3^{1+2}.SL_2(3)$ then the following code shows that $i_2(H) \leq 847$ and then by arguing as in the proof of [3, Lemma 4.11] we deduce that $\mathcal{Q}(G, H) < 1$:

```
H:=MaxRank2(true);
c1:=Classes(H);
&+[c1[i][2] : i in [1..#c1] | c1[i][1] eq 2];
```

847

In the remaining cases, we refer the reader to the proof of the relevant lemma in [3] (see the reference in Table 1) for further details. \square

Lemma 2.3. *If $G = {}^2E_6(2).3$ and $H = 3.U_3(2)^3.3^2.Sym_3$, then $b(G, H) = 2$.*

Proof. As explained in Example 1.12, we calculate that $i_3(H) = 492074$ and then by arguing as in the proof of [3, Lemma 4.12] we conclude that $\mathcal{Q}(G, H) < 1$ and thus $b(G, H) = 2$. \square

Lemma 2.4. *If $G_0 = E_6^\varepsilon(2)$ and $H_0 = \Omega_{10}^\varepsilon(2)$ then G is not extremely primitive.*

Proof. For $\varepsilon = -$ we refer the reader to Examples 1.6 and 1.7. Now assume $\varepsilon = +$, so $G = E_6(2).2$ and $H = O_{10}^+(2)$. This case requires a slightly modified approach because the character table of G is not available in [2]. First we use [2] to access the character tables of G_0 and H_0 and we find that there are 96 possible fusion maps from H_0 -classes to G_0 -classes. Then by proceeding as in Example 1.7, we deduce that G_0 has rank 35 (this calculation is independent of the choice of fusion map). Therefore, the rank of G is at most 35 and we can complete the argument in the usual manner, noting that $|H : M| \leq 16189440$ for every maximal subgroup M of H . \square

Lemma 2.5. *If $G_0 = F_4(2)$ and*

$$H_0 \in \{L_3(2)^2.2, 3.U_3(2)^2.3.2, Sp_4(2) \wr Sym_2, Sp_4(4).2\}$$

then $b(G, H) = 2$.

Proof. For the first two cases, we refer the reader to Examples 1.4 and 1.5. For the remaining two cases we have $G = \text{Aut}(G_0)$ and they are handled in a similar fashion. First we construct G and G_0 as permutation groups and then we construct a maximal subgroup $L = Sp_8(2)$ of G_0 . To do this, we use the fact that $Sp_8(2) = \langle a, b \rangle$, where the generators a and b satisfy the relations $a^2 = b^5 = (ab)^{17} = (ab^2)^{21} = 1$ (see [8], for example).

```
G:=AutomorphismGroupSimpleGroup("F4",2);
G0:=Socle(G);
repeat
  a:=Random(G0);
  b:=Random(G0);
until (Order(a) mod 2 eq 0) and (Order(b) mod 5 eq 0)
  and (Order(a^(Order(a) div 2)*b^(Order(b) div 5)) eq 17)
  and (Order(a^(Order(a) div 2)*b^(2*Order(b) div 5)) eq 21)
  and (#Normalizer(G0,sub<G|a^(Order(a) div 2),b^(Order(b) div 5)>))
  eq #PSp(8,2));
L:=Normalizer(G0,sub<G|a^(Order(a) div 2),b^(Order(b) div 5)>);
```

Next we take a set of representatives of the maximal subgroups of L and we identify $K_1 = Sp_4(2) \wr Sym_2$ and $K_2 = Sp_4(4).2$ by their orders. We then construct the relevant maximal subgroups H of G by taking the normalizers of K_1 and K_2 in G . Finally, it is easy to find an element $g \in G$ by random search such that $H \cap H^g = 1$.

```
M:=MaximalSubgroups(L);
exists(a){i : i in [1..#M] | #M[i]'subgroup eq #Sp(4,2)^2*2};
exists(b){i : i in [1..#M] | #M[i]'subgroup eq #Sp(4,4)*2};
for i in [a,b] do
  H:=Normalizer(G,M[i]'subgroup);
  repeat
    g:=Random(G);
  until #(H meet H^g) eq 1;
end for;
```

\square

Lemma 2.6. *If $G = F_4(2)$ and $H = {}^3D_4(2).3$, then G is not extremely primitive.*

Proof. This is essentially identical to the case $(G, H) = ({}^2E_6(2).2, O_{10}^-(2))$, which we discussed in Example 1.7. Indeed, we use [2] to compute the rank of G and then we turn to MAGMA in order to determine the indices of the maximal subgroups of H . In particular, there are 4 possible fusion maps and we deduce that G has rank 7.

The indices of the core-free maximal subgroups of H are as follows:

4064256, 978432, 179712, 163072, 89856, 69888, 17472, 2457, 819

and we have $|G : H| = 5222400$. Let n_i denote the i -th index in this list. Then to complete the proof via [3, Lemma 2.2], we need to show that there is no tuple $[a_1, \dots, a_9]$ of non-negative integers such that $\sum_i a_i = 6$ and $|G : H| = 1 + \sum_i a_i n_i$. We can use MAGMA to check this:

```
H:=AutomorphismGroupSimpleGroup("3D4",2);
M:=MaximalSubgroups(H);
A:={#H div #M[i]'subgroup: i in [1..#M] | #Core(H,M[i]'subgroup) eq 1};
#RestrictedPartitions(5222400-1,6,A);
```

This returns 0 and we conclude that G is not extremely primitive. \square

Lemma 2.7. *If $G_0 = {}^3D_4(q)$ and $H_0 = (q^2 - q + 1)^2:\text{SL}_2(3)$ with $q \in \{3, 4\}$, then $b(G, H) = 2$.*

Proof. In both cases, $H = N_G(P)$, where P is a Sylow $(q^2 - q + 1)$ -subgroup of G_0 , and it is easy to check that $b(G, H) = 2$ (it is sufficient to check this for $G = \text{Aut}(G_0)$). For example, here is the MAGMA code for the case $q = 4$:

```
G:=AutomorphismGroupSimpleGroup("3D4",4);
G0:=Socle(G);
P:=SylowSubgroup(G0,13);
H:=Normalizer(G,P);
repeat
  g:=Random(G);
until #(H meet H^g) eq 1;
```

\square

Lemma 2.8. *If $G_0 = {}^3D_4(q)$ and $H_0 = (q^2 + q + 1)^2:\text{SL}_2(3)$ with $q \in \{3, 4, 5\}$, then $b(G, H) = 2$.*

Proof. This is entirely similar to the previous lemma, noting that $H = N_G(P)$ for a Sylow r -subgroup P of G_0 with $r = 13, 7$ and 31 when $q = 3, 4$ and 5 , respectively. \square

2.3. Miscellaneous cases.

Lemma 2.9. *If $G = G_2(7)$ and $H = 2^3.\text{L}_3(2)$, then $b(G, H) = 2$.*

Proof. Here we use MAGMA to construct G as a permutation group and then we find $H = N_G(L)$ by identifying an appropriate elementary abelian subgroup L of order 8 in a Sylow 2-subgroup of G . It is then routine to find an element $g \in G$ by random search such that $H \cap H^g = 1$.

```
G:=AutomorphismGroupSimpleGroup("G2",7);
P:=SylowSubgroup(G,2);
S:=Subgroups(P:OrderEqual:=8);
exists(a){i : i in [1..#S] | #Normalizer(G,S[i]'subgroup) eq 8*#SL(3,2)};
H:=Normalizer(G,S[a]'subgroup);
repeat
  g:=Random(G);
until #(H meet H^g) eq 1;
```

\square

Lemma 2.10. *If $G_0 = F_4(2)$ and $H_0 = {}^2F_4(2)$, then G is not extremely primitive.*

Proof. Here $H_0 = C_{G_0}(x)$, where x is an involutory graph automorphism of G_0 . This observation allows us to construct H_0 and we then use random search to find an element $g \in G_0$ such that $H_0 \cap H_0^g = 1$. Therefore, $b(G_0, H_0) = 2$ and G_0 is not extremely primitive. Note that $|H_0| = 2^{12}(q^3 + 1)(2^4 - 1)(2^6 + 1)$.

```
G:=AutomorphismGroupSimpleGroup("F4",2);
g:=Socle(G);
n:=2^12*65*15*9;
repeat
  x:=Random(G);
until Order(x) mod 2 eq 0 and #Centralizer(g,x^(Order(x) div 2)) eq n;
h:=Centralizer(g,x^(Order(x) div 2));
repeat
  x:=Random(g);
until #(h meet h^x) eq 1;
```

To complete the proof, we may assume $G = G_0.2$ and $H = H_0.2$. Since $H_0 \cap H_0^g = 1$ it follows that $|H \cap H^g| \leq 2$ and thus $H \cap H^g$ is not a maximal subgroup of H . Therefore, G is not extremely primitive. \square

Lemma 2.11. *If $(G_0, \text{soc}(H))$ is one of the following*

$$(E_6(4), \text{Fi}_{22}), (E_6(4), \Omega_7(3)), (F_4(3), {}^3D_4(2)), (G_2(11), J_1), (G_2(7), U_3(3)),$$

then $b(G, H) = 2$.

Proof. First assume $G_0 = E_6(4)$ and $\text{soc}(H) = \text{Fi}_{22}$. Let V be the adjoint module for G_0 , so V is an irreducible 78-dimensional module for G_0 over \mathbb{F}_4 and $\text{soc}(H)$ acts irreducibly on V . From the 2-modular character table of $L = \text{Fi}_{22}.2$ (this is available in [2], for example), we see that L has a unique 78-dimensional irreducible module over \mathbb{F}_4 and so we can identify this module with V . This allows us to use MAGMA to construct V as an irreducible module for L over \mathbb{F}_4 and we can then determine the action on V of a set of conjugacy class representatives of L . To do this, we work with a permutation representation of L of degree 3510, which is available in [8], and we use the functions `SInfo` and `UInfo` defined in Section 1.5:

```
I:=IrreducibleModulesBurnside(L,GF(4):DimLim:=500);
for i in [1..#I] do
  [i,Dimension(I[i])];
end for;
f:=Representation(I[2]);
cl:=Classes(L);
CL:=[];
for i in [1..#cl] do
  Append(~CL,<cl[i][1],cl[i][2],f(cl[i][3])>);
end for;

SInfo(L,CL);
UInfo(L,CL);
```

If $x \in L$ has prime order $r \geq 3$ then the output from `SInfo(L,CL)` allows us to read off $\dim C_V(x) = \dim C_{\bar{G}}(x)$, where $\bar{G} = E_6$ is the ambient simple algebraic group. If $r = 3$ then $\dim C_V(x)$ uniquely determines $C_{\bar{G}}(x)^0$ and for $r \geq 5$ we deduce that $\dim C_{\bar{G}}(x) \leq 18$. As explained in the proof of [3, Lemma 7.11], this allows us to estimate the contribution to $\mathcal{Q}(G, H)$ from semisimple elements. Similarly, the output from `UInfo(L,CL)` gives the

Jordan form of each involution $x \in H$. This uniquely determines the G -class of x and we can then compute $|x^G \cap L|$. We get the results presented in the table below. Here, we label the G -classes of involutions in G_0 as in [7, Table 22.2.3] and we write τ and τ' for representatives of the two G -classes of involutory graph automorphisms of G_0 , where $C_{G_0}(\tau) = F_4(4)$.

Jordan form of x	G -class of x	$ x^G \cap L $
(J_2^{22}, J_1^{34})	A_1	3510
(J_2^{32}, J_1^{14})	A_1^2	1216215
(J_2^{38}, J_1^2)	A_1^3	36486450
(J_2^{26}, J_1^{26})	τ	61776
(J_2^{36}, J_1^6)	τ'	19459440 + 22239360

This allows us to calculate the contribution to $\mathcal{Q}(G, H)$ from unipotent elements and by combining this with the previous estimate for semisimple elements, we conclude that $\mathcal{Q}(G, H) < 1$ and thus $b(G, H) = 2$. We refer the reader to the proof of [3, Lemma 7.11] for the details.

The remaining cases can be handled in an identical fashion. Indeed, in each case $\text{soc}(H)$ acts irreducibly on the adjoint module V for G_0 and by suitably modifying the previous code, we can use MAGMA to calculate the action of the relevant conjugacy class representatives on V for each $x \in H$ of prime order. Then as explained in [3, Section 7], this information allows us to show that $\mathcal{Q}(G, H) < 1$, which gives $b(G, H) = 2$. \square

Lemma 2.12. *If $G_0 = {}^2E_6(2)$ and $\text{soc}(H) = \text{Fi}_{22}$, then G is not extremely primitive.*

Proof. Here $(G, H) = ({}^2E_6(2), \text{Fi}_{22})$ or $({}^2E_6(2).2, \text{Fi}_{22}.2)$ and we proceed as in Example 1.6, noting that the character tables of G and H , together with the associated fusion map, are stored in [2]. In both cases we calculate that G has rank 8 and we note that $|G : H| = 1185415168$.

To compute the indices of the maximal subgroups of H , it is convenient to use the permutation representation of H of degree 3510, which is available in [8]:

```
M:=MaximalSubgroups(H);
[#H div #M[i]'subgroup: i in [1..#M] | #Core(H,M[i]'subgroup) eq 1];
```

For $H = \text{Fi}_{22}.2$, we deduce that the indices of the maximal core-free subgroups of H are as follows:

3510, 61776, 142155, 694980, 1216215, 1647360, 3612614,
3648645, 5125120, 12812800, 15206400, 17791488

and since $1 + 7.17791488 < |G : H|$, we conclude that G is not extremely primitive.

Similarly, if $H = \text{Fi}_{22}$, then

3510, 14080, 61776, 142155, 694980, 1216215, 1647360,
3592512, 3648645, 12812800, 17791488, 679311360

are the indices of the maximal subgroups of H . Since

$$1 + 6.17791488 + 679311360 < |G : H|$$

it follows that if G is extremely primitive, then H must have at least two orbits of length 679311360. But $1 + 2.679311360 > |G : H|$, so this is not possible and thus G is not extremely primitive. \square

Lemma 2.13. *If $G_0 = F_4(2)$ and $\text{soc}(H) = L_4(3)$, then G is not extremely primitive.*

Proof. Here $(G, H) = (F_4(2), L_4(3).2)$ or $(F_4(2).2, L_4(3).2^2)$ and we can proceed as in the previous lemma, accessing the character table of H in GAP via

`CharacterTable("L4(3).2_2")` and `CharacterTable("L4(3).2^2")`,

respectively.

For $(G, H) = (F_4(2), L_4(3).2)$ we calculate that G has rank 94 and using MAGMA we find that $|H : M| \leq 10530$ for every maximal subgroup M of H . But $1 + 93 \cdot 10530 < |G : H|$ and thus G is not extremely primitive. Similarly, if $(G, H) = (F_4(2).2, L_4(3).2^2)$ then G has rank 66, $|H : M| \leq 21060$ and once again we deduce that G is not extremely primitive since $1 + 65 \cdot 21060 < |G : H|$. \square

Lemma 2.14. *If $G_0 = {}^2E_6(2)$ and $\text{soc}(H) = \Omega_7(3)$, then $b(G, H) = 2$.*

Proof. We proceed as in Example 1.8, using a combination of [2] and MAGMA to compute $\mathcal{Q}(G, H)$. More precisely, if $G = {}^2E_6(2)$ and $H = \Omega_7(3)$, then the computation in Example 1.8 gives

$$\mathcal{Q}(G, H) = \frac{1946811337}{5927075840}.$$

Similarly, if $G = {}^2E_6(2).2$ and $H = \text{SO}_7(3)$, then

$$\mathcal{Q}(G, H) = \frac{5140272007}{13039566848}.$$

In both cases, $\mathcal{Q}(G, H) < 1$ and thus $b(G, H) = 2$ by [3, Lemma 2.6]. \square

REFERENCES

- [1] W. Bosma, J. Cannon and C. Playoust, *The MAGMA algebra system I: The user language*, J. Symbolic Comput. **24** (1997), 235–265.
- [2] T. Breuer, *The GAP Character Table Library, Version 1.2.1*, GAP package, <http://www.math.rwth-aachen.de/~Thomas.Breuer/ctbllib>, 2012.
- [3] T.C. Burness and A.R. Thomas, *The classification of extremely primitive groups*, submitted (arxiv:2005.11553)
- [4] The GAP Group, *GAP – Groups, Algorithms, and Programming*, Version 4.11.0, 2020, (<http://www.gap-system.org>).
- [5] R. Lawther, *Jordan block sizes of unipotent elements in exceptional algebraic groups*, Comm. Algebra **23** (1995), 4125–4156.
- [6] M.W. Liebeck, J. Saxl and G.M. Seitz, *Subgroups of maximal rank in finite exceptional groups of Lie type*, Proc. London Math. Soc. **65** (1992), 297–325.
- [7] M.W. Liebeck and G.M. Seitz, *Unipotent and nilpotent classes in simple algebraic groups and Lie algebras*, Mathematical Surveys and Monographs, vol. 180, Amer. Math. Soc., 2012.
- [8] R.A. Wilson et al., *A World-Wide-Web Atlas of finite group representations*, <http://brauer.maths.qmul.ac.uk/Atlas/v3/>

T.C. BURNES, SCHOOL OF MATHEMATICS, UNIVERSITY OF BRISTOL, BRISTOL BS8 1UG, UK
Email address: t.burness@bristol.ac.uk

A.R. THOMAS, MATHEMATICS INSTITUTE, ZEEMAN BUILDING, UNIVERSITY OF WARWICK, COVENTRY CV4 7AL, UK
Email address: adam.r.thomas@warwick.ac.uk