On computing efficient presentations for simple groups

George Havas
Centre for Discrete Mathematics and Computing
School of Information Technology and Electrical Engineering
The University of Queensland, Queensland 4072, Australia.
Email: havas@itee.uq.edu.au

joint work with
Colin Campbell and Edmund Robertson, The University of St Andrews
Colin Ramsay, The University of Queensland

This research was partially supported by the Australian Research Council and was also partially supported by the Engineering and Physical Sciences Research Council grant EP/C523229/01, “Multidisciplinary Critical Mass in Computational Algebra and Applications”.

Matrix Group Recognition, ICMS, 29 July 2009

Introduction

There is much interest in finding short presentations for the finite simple groups.

These are useful for verifying that a matrix representation is correct.

Indeed it has been suggested that all these groups are efficient in a technical sense.

In previous papers we produced nice efficient presentations for all except one of the simple groups with order less than one million.

Here we show that some larger simple groups are efficient.

Apart from some linear groups these results are all new.

We also show that some of their covering groups are efficient.

We make substantial use of systems for computational group theory and, in particular, of computer implementations of coset enumeration to find and verify our presentations.

In the consideration of presentations there are various notions of short.

These include length (addressed recently by Guralnick, Kantor, Kassabov and Lubotzky) and a technical notion: efficiency.

Epstein studied geometric properties of groups in 1961.

He used homological arguments to show that there is a lower bound on the minimal number of relations required to present a group.

If $\{X \mid R\}$ is a presentation of a finite group G, then $|R| - |X|$ is at least the smallest number $d(M)$ of generators of the Schur multiplier M of G; this presentation is efficient if $|R| - |X| = d(M)$.

There is much interest in finding short presentations for simple groups and for their covering groups.

Mind you, it is difficult to find best presentations.

We do not have a general answer even for cyclic groups (because the optimal addition chain problem is NP-complete).

We focus on the issue of efficiency.

Throughout this talk, simple group will refer to nonabelian simple group, since efficiency questions for cyclic groups have easy positive answers.

There is much recent work on short presentations for simple groups, including

Bray, Conder, Leedham-Green and O'Brien (preprint)
CHHR (2004, 2007)
GKKL (2007, 2008, 2009, . . .)
Korchagina and Lubotzky (2006)
John S. Wilson (2006). Indeed Wilson says

It seems reasonable to conjecture that the covering group of every finite simple group has a presentation with two generators and two relators.

If true, this implies that all finite simple groups are efficient.

We use Atlas notation for the names of simple groups and we denote the covering group of the simple group G by $\widehat{G}$.

In CHRR04 and CHRR07 we give **nice** efficient presentations for all except one, $S_4(4)$, of the simple groups with order less than one million.

We also leave the efficiency question for one other covering group, $\widehat{U_3}(5)$, unresolved

For larger simple groups the situation is rather different to that of smaller orders.

For smaller orders most simple groups and their covering groups were already known to be efficient when we produced our presentations.

We resolved six out of the eight previously unknown cases.

Now, for orders from 1 to 5 million, apart from $L_2(p)$ (p prime) and $L_2(13^2)$, it was unknown whether the other seven simple groups are efficient.

We prove that they all are by giving efficient presentations for each of them.

We also show that some larger groups are efficient.

In some cases we prove that the covering groups are efficient, but we leave the efficiency question for covering groups of 4 simple groups with orders between 1 and 5 million unresolved.

This work was motivated in part by a request from Bill Kantor for an efficient presentation for A_{10}. In GKKL (2009) it is shown that:

Theorem A. (A.1) All nonabelian finite simple groups of Lie type, with the possible exception of the Ree groups ${}^2G_2(q)$, have presentations with 2 generators and at most 80 relations.

(Further, any pair of generators can be used for such presentations.)

They state that a similar result holds for all finite simple groups, except perhaps the Ree groups and that the bounds of Theorem A are not optimal.

(A.2) All symmetric and alternating groups have presentations with 2 generators and 8 relations.

Another theorem improves on this result for the symmetric and alternating groups as follows:

Theorem C. For each $n \geq 5$, A_n and S_n have presentations with 3 generators, 7 relations and bit-length O$(\log n)$, using a bounded number of exponents.

They continue by stating that, if $\boldsymbol{a}$ and $\boldsymbol{b}$ are any generators of $\boldsymbol{G = A_n}$ or $\boldsymbol{S_n}$, then there is a presentation of $\boldsymbol{G}$ using $\mathbf{2}$ generators that map onto $\boldsymbol{a}$ and $\boldsymbol{b}$, with $\mathbf{9}$ relations.

Aspects of this work rely on using explicit efficient presentations for various "small" groups as building blocks.

For example, efficient presentations for $\boldsymbol{A_{10}}$ are used to construct $\mathbf{2}$-generator, $\mathbf{6}$-relator presentations for $\boldsymbol{A_{47}}$.

More generally, efficient presentations (CHRR04, CHRR07) for $\boldsymbol{\widehat{A}_n}$ $\boldsymbol{(n \leq 9)}$ and for $\boldsymbol{A_{10}}$ can be used in their presentations to save several relations.

Methodology

We use three different methods to find efficient presentations.

Method 1 involves a study of short presentations for perfect groups.

The size of the groups considered here means that this method has not produced any efficient presentation for the groups within the presentation length range that we have searched.

Method 2 uses a consideration of presentations based on different generating pairs for the groups.

In principle we could look at all (essentially different) generating sets.

The size of the groups considered here means that instead of all generating sets we choose well or just randomly.

Method 3 involves a study of one-relator quotients of free products $C_m * C_n$ for coprime m and n.

(By a one-relator quotient of a particular group we mean a presentation obtained by adding one extra relator to a presentation for the specified group.)

Each of our methods relies on investigations of search spaces of varying kinds.

With increasing group size and presentation length the search spaces grow enormously.

Under these circumstances we do not attempt exhaustive searches, but rather are satisfied when we have found at least one presentation for each simple group.

We find that variants of methods 2 and 3 suffice to handle all simple groups with order from 1 to 5 million.

We use GAP, Magma and stand-alone programs to do our searches.

We check that our presentations are correct by coset enumeration; we generally use the ACE enumerator either as available in GAP or Magma, or as a stand-alone program for some more difficult cases.

As far as reliability of results is concerned we assert that all of our presentations are correct.

Each new presentation has been verified by both GAP and Magma programs to present the specified group.

Here we intentionally do not use ACE for the GAP check but rather use GAP's internal coset enumerator (which was entirely independently written), providing a strong level of confidence in the results.

We use two results from CHRR04 and an extension which enable us to amalgamate relators in presentations to give presentations for associated groups with fewer relators.

The proofs of these results are constructive, which allows us to build efficient presentations using them.

Theorem 1. *Let G be a finite simple group. Suppose that G, or some stem extension of G, can be presented as*

$$\{\ a, b \mid a^p = b^q = w(a, b) = 1\ \}.$$

Then the covering group of G, all stem extensions of G, and G itself, are efficient.

Corollary 2. *Let G be a finite simple group. Suppose that G, or some stem extension of G, can be presented as*

$$\{\ a, b \mid u(a, b)^p = v(a, b)^q = w(a, b) = 1\ \}.$$

Suppose also that $u(a, b)$ and $v(a, b)$ generate the free group on a and b. Then the covering group of G, all stem extensions of G, and G itself, are efficient.

A natural extension of Theorem 1 gives methods for amalgamating relations given a presentation for (a stem extension of) a simple group with more relations, such as

$$P = \{a, b \mid a^p = b^q = w_1(a, b) = \ldots = w_n(a, b) = 1\}.$$

We point out that our primary focus is on presentations which are efficient in terms of deficiency.

This does not always coincide with best presentations for other purposes.

For deficiency-zero groups, the deficiency-one presentation

$$\{a, b \mid a^p = b^q = w(a, b) = 1\}$$

is likely to be much more useful for practical computation than the efficient presentation produced by Theorem 1,

$$\{a, b \mid a^p b^{-q} = \widetilde{w}(a, b) = 1\}.$$

For example, these deficiency-one presentations are better for coset enumeration than the corresponding efficient presentations.

A similar situation applies to presentations obtained from the extension of Theorem 1.

Orders 1 to 5 million

Sunday (1972) gives efficient presentations for $L_2(p)$ for all prime $p \geq 5$.

Based on this, CR (1980) give efficient presentations for $\widehat{L}_2(p)$.

We give efficient presentations for all other simple groups with order between 10^6 and 5×10^6.

We remark that our methods do give shorter presentations for individual PSLs than Sunday's generic presentation.

We adopt the convention of using upper-case letters to denote inverses in presentations so that, for example, $A = a^{-1}$.

We give presentations by listing sets of relators (often only implicitly specifying the generators).

$S_6(2)$, order 1451520, Mult = 2, not (2,3)-gen

We investigated 3395 random generating pairs for $S_6(2)$, taking about 50 CPU hours.

We obtained initial presentations with between 5 and 19 relators.

We reduced 3 of these presentations to 3-relator presentations for preimages of $S_6(2)$ and 76 to 4-relator presentations.

Most of these were proper preimages.

However one of the 3-relator presentations does present the group, namely:

$$b^{10}, BaBABAbbABB, aBababaababABB.$$

In addition we are able to use the extension of Theorem 1 to provide other efficient presentations for $S_6(2)$ among the 4-relator presentations.

As two examples, by amalgamating $(AB)^2$ and B^7 we can obtain:

$$abab^8,$$

$$BBaBBaBAABaBBaBBa, AbaBBaBAbbaaBAbbAAbABBaaBBaabb$$

and by amalgamating b^4 and a^7 we can obtain:

$$b^4a^7, AABaaaBAAABaabbAbbA, ABBAABBAAAbAAbbaabA.$$

A_{10}, order 1814400, Mult = 2, (2,3)-gen

We investigated 7035 random generating pairs for A_{10}, taking about 100 CPU days.

We obtained initial presentations with between 5 and 35 relators.

We reduced 6 of these presentations to 3-relator presentations for preimages of A_{10} and 121 to 4-relator presentations, including the following 5 which present A_{10}:

$$BAABAAAbAbAAA, aaBaaaaaBBBaaa, AAbaBabaaabaBabAAB$$

$$babaBAAAbababba, AAAABaaaabababab, AbaabbaBAABBaaBaBab$$

$$ABBABBAbbabb, bbAABAAAAbaabbbb, AABBAABAABBBBabbb$$

$$bAAAAbbAb, BABAbABAABaabAbaa, babAAbABaaBAAbAABa$$

$$aBaaBAAbbbABa, abaabaaabaBABa, baabABaaaBAAbaaa$$

$L_3(7)$, order 1876896, Mult = 3, (2,3)-gen

$L_3(7)$ is (2,3)-generated so it looks attractive to try for a one-relator quotient of $C_2 * C_3$, but we have not succeeded with that approach.

However:

$$ababAABabab, AAAAAbaabaabaab, aBBBabAAABAAbbbab$$

$L_2(128)$, order 2097024, Mult = 1, (2,3)-gen

$L_2(128)$ is (2,3)-generated and does arise as a one-relator quotient of $C_2 * C_3$.

```
census15/25.ph2.d2:681 Rel: (wZw)2, (Wz)3, w2z2w2zwzwzwz3wzwzwzw2z2;
INDEX = 2097024 (m=283082760 t=285928760)

census15/29.ph2.d2:553 Rel: (wZw)2, (Wz)3, wz2wz2w4zw2z4w2zw4z2wz2;
INDEX = 2097024 (m=78097739 t=80953562)
```

Use our amalgamation thereom to build an efficient presentation.

$L_2(169)$, order 2413320, Mult = 2, (2,3)-gen

An efficient presentation for $L_2(169)$ has been previously reported by CR (1988), as a one-relator quotient of $C_2 * C_3$. We now find many such presentations:

```
census15/23.ph2.d2:39 Rel: (wZw)2, (Wz)3, w5z3w2z8w2z3;
INDEX = 2413320 (m=8872300 t=9029601)
census15/23.ph2.d2:316 Rel: (wZw)2, (Wz)3, w5zwz2w10z2wz;
INDEX = 2413320 (m=2413320 t=3397688)
census15/27.ph2.d2:794 Rel: (wZw)2, (Wz)3, wzwzwz2wzw2z2w2z2w2zwz2wzwz;
INDEX = 2413320 (m=253717890 t=254196866)
census15/27.ph2.d2:939 Rel: (wZw)2, (Wz)3, wz4w3z2w2z4w2z2w3z4;
INDEX = 2413320 (m=49335198 t=50267677)
census15/27.ph2.d2:948 Rel: (wZw)2, (Wz)3, wzw2zw6z6w6zw2z;
INDEX = 2413320 (m=44896560 t=46570975)
census15/29.ph2.d2:118 Rel: (wZw)2, (Wz)3, w2zw3z8w3z8w3z;
INDEX = 2413320 (m=7709166 t=8770251)
```

```
census15/29.ph2.d2:665 Rel: (wZw)2, (Wz)3, wz2wzw5zwz2w2z2wzw5zwz2;
INDEX = 2413320 (m=8136328 t=8449808)
census15/31a.ph2.d2:231 Rel: (wZw)2, (Wz)3, wzw6zwz12wzw6z;
INDEX = 2413320 (m=108420272 t=112330062)
census15/31d.ph2.d2:13 Rel: (wZw)2, (Wz)3, wz2w2zw6zw2z2w2zw6zw2z2;
INDEX = 2413320 (m=18110523 t=18260725)
census15/31j.ph2.d2:211 Rel: (wZw)2, (Wz)3, wzwzwzwz2w2zwzwz2wzwzw2z2wzwzwz
INDEX = 2413320 (m=95208904 t=96639511)
census15/33a.ph2.d2:25 Rel: (wZw)2, (Wz)3, wz2w8z12w8z2;
INDEX = 2413320 (m=31366083 t=32010424)
census15/33f.ph2.d2:26 Rel: (wZw)2, (Wz)3, w2zwzw5zw3zw5zw3zw5zwz;
INDEX = 2413320 (m=23528008 t=23731554)
census15/33j.ph2.d2:146 Rel: (wZw)2, (Wz)3, w2zwzw3zwz2wzwzw3zwzwz2wzw3zwz;
INDEX = 2413320 (m=168593304 t=168977340)
census15/35.ph2.d2:124 Rel: (wZw)2, (Wz)3, w5z5w6z8w6z5; CR (1988)
INDEX = 2413320 (m=3258522 t=4567814)
```

$U_4(3)$, order 3265920, Mult $= 3 \times 12$, not (2,3)-gen

```
aaaaabbbbbbb, baBaBABAbabbbA, aaaabbbABABBBAb, aaBABAABAABaaBabaBB;
INDEX = 3265920 (m=3265920 t=4729489)

(Ba)^5, BAABBBaaBBBAAB, ABaBBABAbAbaBBB, babABaBaaBBAbAAb;
Subgroup Generators: Ba; INDEX = 653184 (m=37030816 t=38273947)

ABBaaaBBAA, (BAB)^5, aBBABBABABBaBBaB, bbABaBaBAABababaaBA;
Subgroup Generators: BAB; INDEX = 653184 (m=2353607 t =2831103)
Subgroup Generators: ; INDEX = 3265920 (m=11347637 t=13723387)

AbbaBAbbabb, baBBBAAAbaBabba, ABaBBBAAbabbbAb, (AAb)^5;
Subgroup Generators: AAb; INDEX = 653184 (m=1082866 t=1379934)
Subgroup Generators: ; INDEX = 3265920 (m=4806024 t=6387193)

(B)^7, abbaaBAABBBAABB, BAAbbaaBaabbaaBBaa, AbbAbbaBBAbbAbaBBBA;
INDEX = 3265920 (m=41262105 t=42726475)
```

```
baBAABaabABBAA, bABABAAAbABBBaa, ABabABAAAbbAbaBAA, (BBAB)^5;
Subgroup Generators: BBAB; INDEX = 653184 (m=20321596 t=20499523)

(b)^4, (Ba)^5, ABABABaaBABABA, ABBabAAAbbAABAb;
Subgroup Generators: Ba; INDEX = 653184 (m=46494484 t=46940902)
```

$G_2(3)$, order 4245696, Mult = 3, (2,3)-gen

$G_2(3)$ is (2,3)-generated so it looks attractive to try for a one-relator quotient of $C_2 * C_3$, but we do not find one directly. (Is there one?)

However looking at presentations on (2,3)-generating sets readily gives the quite nice presentation:

```
(a)^2, (B)^3, (aB)^13, bababaBabaBaBabaBaBaBabababaBababaBabaBaBaBaba;
INDEX = 4245696 (m=4245696 t=5090793)
```

Now using the extension to Theorem 1 (amalgamate a^2 and b^3) we get:

```
aabbb, (aB)^13, babababaBabaBaBabaBaBaBaBababababaBababaBabaBaBaBaba;
Subgroup Generators: aB; INDEX = 326592 (m=2804474 t=2986351)
```

This presentation is also satisfied by the online Atlas standard generators (and is much shorter than the one there).

Our theorem tells you what happens when you amalgamate certain types of powers.

But we can try amalgamating other things.

Amalgamate $(aB)^{13}$ and the long relator to get:

```
a2, b3, aBaBaBaBaBaBababaBaBabaBaBaBaBabababaBababaBabaBaBab;
gen: ab; INDEX = 326592 (m=24259894 t=24371633)
gen: ab; mend:1; INDEX = 326592 (m=15462319 t=15539007)
```

```
gen: b; mend:1; INDEX = 4245696 (m=80886845 t=80974845)
```

It follows that we can find efficient presentations for $\widehat{G}_2(3)$.

This presentation is **not** satisfied by the online Atlas standard generators (for which no presentation is given).

$S_4(5)$, order 4680000, Mult = 2, (2,3)-gen

$S_4(5)$ is (2,3)-generated and we find presentations as one-relator quotients of $C_2 * C_3$ directly.

```
census15/21.d2:81 Rel: (wZw)2, (Wz)3, w3z2w5z4w5z2;  # G(3,2,5,4)
INDEX = 4680000 (m=4680000 t=4704212)
census15/29.ph2.d2:599 Rel: (wZw)2, (Wz)3, w2z3w3z2wzw2z3w2zwz2w3z3;
INDEX = 4680000 (m=4680000 t=4962393)
census15/29.ph2.d2:977 Rel: (wZw)2, (Wz)3, w3zwzw2zwz2wzw4zwz2wzw2zwz;
```

```
INDEX = 4680000 (m=4680000 t=5342417)
census15/35.ph2.d2:454 Rel: (wZw)2, (Wz)3, w3z3w3z4w4z4w4z4w3z3;
INDEX = 4680000 (m=4680000 t=4766748)
census4/27c.E.d2:3650 Rel: vUvUvU, vUvvUv, uuuuvuvvuvuuvuuvuvuuvuvuuvv;
INDEX = 4680000 (m=4680000 t=6201255)
```

It follows that we can find “the” best presentation and efficient presentations for its cover.

Interestingly enough, it also arises as a one-relator quotient of $C_2 * C_{13}$:

$$xyxy^3xy^2xy^{-3}xy^6$$

Linear groups

Previous work has shown that 2-dimensional linear groups are relatively easy to show efficient.

On the one hand we have the infinite class $L_2(p)$, p prime.

In addition we can find efficient presentations for the following groups $L_2(q)$ and their covers.

```
L2(3^5)  order   7 174 332
L2(17^2) order  12 068 640
L2(7^3)  order  20 176 632
L2(19^2) order  23 522 760
L2(23^2) order  74 017 680
```

```
L2(3^5)   order 7174332                      2^2 3^5 11^2 61
  census15/25.ph2.d2:106 Rel: (wZw)2, (Wz)3, wz2w2zw14zw2z2;
  INDEX = 7174332 (m=72600049 t=73819479)
  census15/27.ph2.d2:544 Rel: (wZw)2, (Wz)3, wzwz2wz3w3zw2zw3z3wz2wz;
  INDEX = 7174332 (m=14758861 t=16910148)
  census15/33i.ph2.d2:29 Rel: (wZw)2, (Wz)3, wz4wz2wz2wzw2zw2zw2zwz2wz2wz4;
  INDEX = 7174332 (m=76727537 t=77204593)
```

```
  census15/33i.ph2.d2:97 Rel: (wZw)2, (Wz)3, w5zwz2wzwz2wzw6zwz2wzwz2wz;
  INDEX = 7174332 (m=147759715 t=153131529)
  census15/35.ph2.d2:390 Rel: (wZw)2, (Wz)3, wz2wz4w2z2w4zw2zw4z2w2z4wz2;
  INDEX = 7174332 (m=20473674 t=26780217)

L2(17^2) order  12068640                  2^5 3^2 5 17^2 29
census15/25.ph2.d2:560 Rel: (wZw)2, (Wz)3, wzw2zwzwz2w6z2wzwzw2z;
  INDEX = 12068640 (m=21266263 t=24887892)
census15/27.ph2.d2:923 Rel: (wZw)2, (Wz)3, wzwz4w5z4w5z4wz;
  INDEX = 12068640 (m= 30017724 t=31397328)
census15/31c.ph2.d2:40 Rel: (wZw)2, (Wz)3, w3zwz4w3z3w4z3w3z4wz;
  INDEX = 12068640 (m=120074333 t=125448989)
census15/31c.ph2.d2:66 Rel: (wZw)2, (Wz)3, w4z3wz5wzw5zwz5wz3;
  INDEX = 12068640 (m=280550187 t=292430044)

L2(7^3)  order  20176632                  2^3 3^3 7^3 19 43
census15/27.ph2.d2:567 Rel: (wZw)2, (Wz)3, wz2wzw2z2w2z2w2z2w2z2w2zwz2;
  INDEX = 20176632 (m=20176632 t=27577156)
```

```
census15/31j.ph2.d2:145 Rel: (wZw)2, (Wz)3, wzwzwzwz4wzwzw2zwzwz4wzwzwz;
  INDEX = 20176632 (m=61265317 t=66077373)
census15/33a.ph2.d2:302 Rel: (wZw)2, (Wz)3, w4z4w2z2w4z5w4z2w2z4;
  INDEX = 20176632 (m=53383115 t=64531504)
census4/27c.E.d2:1134 Rel: vUvUvU, vUvvUv, uuvuuvuvvuuvvuuvvuuvvuuvvuv;
  INDEX = 20176632 (m=20176632 t=27544653)


L2(19^2) order  23522760                    2^3 3^2 5 19^2 182
census15/23.ph2.d2:398 Rel: (wZw)2, (Wz)3, w2zwz2w5z3w5z2wz;
  INDEX = 23522760 (m=307943913 t=309894043)
census15/25.ph2.d2:447 Rel: (wZw)2, (Wz)3, w2z2w2z2wzwz5wzwz2w2z2;
  INDEX = 23522760 (m=23522760 t=29390662)
census15/27.ph2.d2:171 Rel: (wZw)2, (Wz)3, w2z3w2z2w11z2w2z3;
  INDEX = 23522760 (m=251881533 t=253137098)
census15/29.ph2.d2:124 Rel: (wZw)2, (Wz)3, w2z8w2z2w3z2w2z8;
  INDEX = 23522760 (m=78665641 t=81390063)
census15/29.ph2.d2:619 Rel: (wZw)2, (Wz)3, w3z3wzw4zwz4wzw4zwz3;
  INDEX = 23522760 (m=34893548 t=57037888)
```

```
census15/33c.ph2.d2:13 Rel: (wZw)2, (Wz)3, w5z2wzw2z5w6z5w2zwz2;
  INDEX = 23522760 (m=161703657 t=174365181)
census15/33h.ph2.d2:32 Rel: (wZw)2, (Wz)3, w4z2wzw2z2wzw2z5w2zwz2w2zwz2;
  INDEX = 23522760 (m=166227051 t=187655294)
census15/35.ph2.d2:887 Rel: (wZw)2, (Wz)3, wzw2zwzwz4wzwzw4zwzwz4wzwzw2z;
  INDEX = 23522760 (m=37169063 t=53385139)
census15/35.ph2.d2:923 Rel: (wZw)2, (Wz)3, w2z3w4z2w2z4w3z4w2z2w4z3;
  INDEX = 23522760 (m=148086119 t=180629558)

L2(23^2) order  74017680                  2^4 3 5 11 23^2 53
census15/25.ph2.d2:542 Rel: (wZw)2, (Wz)3, wzw2zwzw2z2w4z2w2zwzw2z;
  INDEX = 74017680 (m=145105980 t=172170274)
census15/27.ph2.d2:125 Rel: (wZw)2, (Wz)3, wz2wz6w8z6wz2;
  INDEX = 74017680 (m=113525791 t=139448007)
census15/33j.ph2.d2:67 Rel: (wZw)2, (Wz)3, w2zwz3wzwzw3zwz3wzw3zwzwz3wz;
  INDEX = 74017680 (m=184168005 t=222297555)
census15/35.ph2.d2:602 Rel: (wZw)2, (Wz)3, w2zwz2w2z2w3z2w2z3w2z2w3z2w2z2wz;
  INDEX = 74017680 (m=239540807 t=258330801)
```

Two linear groups left with order $< 10^7$.

$L_4(3)$, order 6065280, Mult = 2

```
BABBABaBaBAB, AbabAbAAbbaBabb, ABAAABaBaaaBaBAAABAB;
Subgroup Generators: ab; INDEX = 466560 (m=50260509 t=53389909)
mend:1; INDEX = 466560 (m=46633606 t=49053020)
gen:; INDEX = 6065280 (m=684365079 t=718575037)

BAAAAABBBaB, (B)^13, ABaaaaaaBABBab;
Subgroup Generators: b; INDEX = 466560 (m=57948651 t=60171938)
```

$L_5(2)$, order 9999360, Mult = 1

$L_5(2)$ is (2,3)-generated, so . . .

It is the last linear group with order below 10 million, but ???

The other groups with order between 5 and 10 million

$U_3(8)$, order 5515776, Mult = 3, (2,3)-gen

```
(ababaa)^2, BAbaBAAABabABAA, bABBAABAABBAbbaab;
Subgroup Generators: a; INDEX = 262656 (m=18703597 t=19079761)
Subgroup Generators: ; INDEX = 5515776 (m=188313099 t=191400773)

ABABaaBBBA, BABABABabbbaBA, bAbbAbAbABABAbbbbabbbabA;
INDEX = 5515776 (m=20111706 t=27583250)
```

$U_3(7)$, order 5663616, Mult = 1, (2,3)-gen

$\boldsymbol{U_3(7)}$ has trivial multiplier, so is harder.

We are one off efficient.

Questions

Who cares? Why bother?

Which efficient presentations do you want?

On which generating sets?

If a group is efficient on one generating set is it necessarily efficient on "other" generating sets?

Can we find another infinite family of efficient groups?