Effective algorithms for groups of Lie type

Eamonn O'Brien

University of Auckland

February 2015

Overview of lecture

G “large" finite group described by generating set X.
e.g. $G = \langle X \rangle \leq \mathrm{GL}(\mathrm{d}, \mathrm{q})$ or $G = \langle X \rangle \leq \mathrm{Sym}(n)$.

Overview of lecture

G "large" finite group described by generating set X.
e.g. $G = \langle X \rangle \leq \mathrm{GL}(\mathrm{d}, \mathrm{q})$ or $G = \langle X \rangle \leq \mathrm{Sym}(n)$.

Can we answer the following?

- Conjugacy classes of elements or subgroups of G
- Sylow p-subgroups of G
- Maximal subgroups of G
- Automorphism group of G

Overview of lecture

G "large" finite group described by generating set X.
e.g. $G = \langle X \rangle \leq \mathrm{GL}(\mathrm{d}, \mathrm{q})$ or $G = \langle X \rangle \leq \mathrm{Sym}(n)$.

Can we answer the following?

- Conjugacy classes of elements or subgroups of G
- Sylow p-subgroups of G
- Maximal subgroups of G
- Automorphism group of G

Soluble Radical model of computation: uniform approach.

Overview of lecture

G "large" finite group described by generating set X.
e.g. $G = \langle X \rangle \leq \mathrm{GL}(\mathrm{d}, \mathrm{q})$ or $G = \langle X \rangle \leq \mathrm{Sym}(n)$.

Can we answer the following?

- Conjugacy classes of elements or subgroups of G
- Sylow p-subgroups of G
- Maximal subgroups of G
- Automorphism group of G

Soluble Radical model of computation: uniform approach.

- Explain the model.
- Discuss how to construct the model.

Characteristic structure

G has characteristic series $\mathcal{C}$ of subgroups:

$$1 \leq O_\infty(G) \leq S^*(G) \leq P(G) \leq G$$

Characteristic structure

G has characteristic series $\mathcal{C}$ of subgroups:

$$1 \leq O_\infty(G) \leq S^*(G) \leq P(G) \leq G$$

$O_\infty(G)$ = largest soluble normal subgroup of G, soluble radical

Characteristic structure

G has characteristic series $\mathcal{C}$ of subgroups:

$$1 \leq O_\infty(G) \leq S^*(G) \leq P(G) \leq G$$

$O_\infty(G)$ = largest soluble normal subgroup of G, soluble radical

$S^*(G)/O_\infty(G)$ = Socle $(G/O_\infty(G)) = T_1 \times \ldots \times T_k$ where T_i non-abelian simple

Characteristic structure

G has characteristic series $\mathcal{C}$ of subgroups:

$$1 \leq O_\infty(G) \leq S^*(G) \leq P(G) \leq G$$

$O_\infty(G)$ = largest soluble normal subgroup of G, soluble radical

$S^*(G)/O_\infty(G)$ = Socle $(G/O_\infty(G)) = T_1 \times \ldots \times T_k$ where T_i non-abelian simple

$\phi : G \longmapsto \mathrm{Sym}(k)$ is repn of G induced by conjugation on $\{T_1, \ldots, T_k\}$ and $P(G) = \ker \phi$

Characteristic structure

G has characteristic series $\mathcal{C}$ of subgroups:

$$1 \leq O_\infty(G) \leq S^*(G) \leq P(G) \leq G$$

$O_\infty(G)$ = largest soluble normal subgroup of G, soluble radical

$S^*(G)/O_\infty(G)$ = Socle $(G/O_\infty(G)) = T_1 \times \ldots \times T_k$ where T_i non-abelian simple

$\phi : G \longmapsto \mathrm{Sym}(k)$ is repn of G induced by conjugation on $\{T_1, \ldots, T_k\}$ and $P(G) = \ker \phi$

$P(G)/S^*(G) \leq \mathrm{Out}(T_1) \times \ldots \times \mathrm{Out}(T_k)$ and so is soluble

Characteristic structure

G has characteristic series $\mathcal{C}$ of subgroups:

$$1 \leq O_\infty(G) \leq S^*(G) \leq P(G) \leq G$$

$O_\infty(G)$ = largest soluble normal subgroup of G, soluble radical

$S^*(G)/O_\infty(G)$ = Socle $(G/O_\infty(G)) = T_1 \times \ldots \times T_k$ where T_i non-abelian simple

$\phi : G \longmapsto \mathrm{Sym}(k)$ is repn of G induced by conjugation on $\{T_1, \ldots, T_k\}$ and $P(G) = \ker \phi$

$P(G)/S^*(G) \leq \mathrm{Out}(T_1) \times \ldots \times \mathrm{Out}(T_k)$ and so is soluble

$G/P(G) \leq \mathrm{Sym}(k)$

Exploiting the characteristic series $\mathcal{C}$

Cannon, Holt et al. (1997–): use $\mathcal{C}$ in practical algorithms.

$$1 \leq L := O_\infty(G) \leq S^*(G) \leq P(G) \leq G$$

Exploiting the characteristic series $\mathcal{C}$

Cannon, Holt et al. (1997–): use $\mathcal{C}$ in practical algorithms.

$$1 \leq L := O_\infty(G) \leq S^*(G) \leq P(G) \leq G$$

Also compute series

$$1 = N_0 \lhd N_1 \lhd \cdots \lhd N_r = L \lhd G$$

where $N_i \trianglelefteq G$ and N_i/N_{i-1} is elementary abelian.

The Soluble Radical model

$$1 = N_0 \lhd N_1 \lhd \cdots \lhd N_r = L \leq S^*(G) \leq P(G) \leq G$$

where $N_i \trianglelefteq G$ and N_i/N_{i-1} is elementary abelian.

The Soluble Radical model

$$1 = N_0 \lhd N_1 \lhd \cdots \lhd N_r = L \leq S^*(G) \leq P(G) \leq G$$

where $N_i \trianglelefteq G$ and N_i/N_{i-1} is elementary abelian.

Given a **problem**:

The Soluble Radical model

$$1 = N_0 \lhd N_1 \lhd \cdots \lhd N_r = L \leq S^*(G) \leq P(G) \leq G$$

where $N_i \trianglelefteq G$ and N_i/N_{i-1} is elementary abelian.

Given a **problem**:

Solve problem first in $G/L = G/N_r$, and then, successively, solve it in G/N_i, for $i = r-1, \ldots, 0$.

The Soluble Radical model

$$1 = N_0 \lhd N_1 \lhd \cdots \lhd N_r = L \leq S^*(G) \leq P(G) \leq G$$

where $N_i \trianglelefteq G$ and N_i/N_{i-1} is elementary abelian.

Given a **problem**:

Solve problem first in $G/L = G/N_r$, and then, successively, solve it in G/N_i, for $i = r-1, \ldots, 0$.

$H := G/L$ has trivial Fitting subgroup.
So H has a socle S which is direct product of non-abelian simple groups T_i and these are permuted under conjugation by H.

The Soluble Radical model

$$1 = N_0 \lhd N_1 \lhd \cdots \lhd N_r = L \leq S^*(G) \leq P(G) \leq G$$

where $N_i \trianglelefteq G$ and N_i/N_{i-1} is elementary abelian.

Given a **problem**:

Solve problem first in $G/L = G/N_r$, and then, successively, solve it in G/N_i, for $i = r-1, \ldots, 0$.

$H := G/L$ has trivial Fitting subgroup.
So H has a socle S which is direct product of non-abelian simple groups T_i and these are permuted under conjugation by H.

Problem may have nice solution for H.

The Soluble Radical model

$$1 = N_0 \lhd N_1 \lhd \cdots \lhd N_r = L \leq S^*(G) \leq P(G) \leq G$$

where $N_i \trianglelefteq G$ and N_i/N_{i-1} is elementary abelian.

Given a **problem**:

Solve problem first in $G/L = G/N_r$, and then, successively, solve it in G/N_i, for $i = r-1, \ldots, 0$.

$H := G/L$ has trivial Fitting subgroup.
So H has a socle S which is direct product of non-abelian simple groups T_i and these are permuted under conjugation by H.

Problem may have nice solution for H.

In many cases, easy to reduce the computation for TF-group H to almost simple groups.

Examples of algorithms using Soluble Radical model

- Determine conjugacy classes of elements of G; (Cannon & Souvignier, 1997)
- Determine maximal subgroups of G; (Cannon & Holt, 2004) and (Eick & Hulpke, 2001)
- Determine the automorphism group of G; (Cannon & Holt, 2003)
- Determine conjugacy classes of subgroups of G; (Cannon, Cox & Holt, 2001)

How do we construct the characteristic chain?

Basic approach: Schreier-Sims techniques, developed first in permutation group context.

Sims (1970, 1971): base and strong generating set (BSGS). Determines chain of stabilisers.

How do we construct the characteristic chain?

Basic approach: Schreier-Sims techniques, developed first in permutation group context.

Sims (1970, 1971): base and strong generating set (BSGS). Determines chain of stabilisers.

$G \leq \mathrm{GL}(d, F)$ acts faithfully on $V = F^d$; $v \cdot g$, for $v \in V$

How do we construct the characteristic chain?

Basic approach: Schreier-Sims techniques, developed first in permutation group context.

Sims (1970, 1971): base and strong generating set (BSGS). Determines chain of stabilisers.

$G \leq \mathrm{GL}(d, F)$ acts faithfully on $V = F^d$; $v \cdot g$, for $v \in V$

Now compute BSGS for G, viewed as permutation group on the vectors with base points e.g. standard basis vectors for V.

How do we construct the characteristic chain?

Basic approach: Schreier-Sims techniques, developed first in permutation group context.

Sims (1970, 1971): base and strong generating set (BSGS). Determines chain of stabilisers.

$G \leq \mathrm{GL}(d, F)$ acts faithfully on $V = F^d$; $v \cdot g$, for $v \in V$

Now compute BSGS for G, viewed as permutation group on the vectors with base points e.g. standard basis vectors for V.

Central problem: good subgroup chain may not exist.

How do we construct the characteristic chain?

Basic approach: Schreier-Sims techniques, developed first in permutation group context.

Sims (1970, 1971): base and strong generating set (BSGS). Determines chain of stabilisers.

$G \leq \mathrm{GL}(d, F)$ acts faithfully on $V = F^d$; $v \cdot g$, for $v \in V$

Now compute BSGS for G, viewed as permutation group on the vectors with base points e.g. standard basis vectors for V.

Central problem: good subgroup chain may not exist.

Largest maximal subgroup of J_4 has index 173 067 389.

How do we construct the characteristic chain?

Basic approach: Schreier-Sims techniques, developed first in permutation group context.

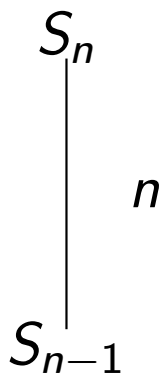
Sims (1970, 1971): base and strong generating set (BSGS). Determines chain of stabilisers.

$G \leq \mathrm{GL}(d, F)$ acts faithfully on $V = F^d$; $v \cdot g$, for $v \in V$

Now compute BSGS for G, viewed as permutation group on the vectors with base points e.g. standard basis vectors for V.

Central problem: good subgroup chain may not exist.

Largest maximal subgroup of J_4 has index 173 067 389.



How do we construct the characteristic chain?

Basic approach: Schreier-Sims techniques, developed first in permutation group context.

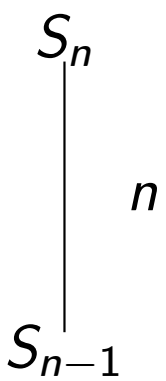
Sims (1970, 1971): base and strong generating set (BSGS). Determines chain of stabilisers.

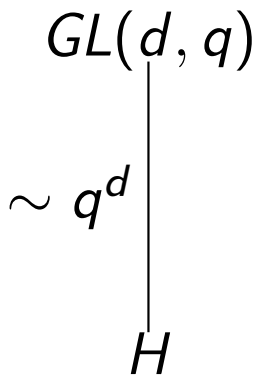
$G \leq \mathrm{GL}(d, F)$ acts faithfully on $V = F^d$; $v \cdot g$, for $v \in V$

Now compute BSGS for G, viewed as permutation group on the vectors with base points e.g. standard basis vectors for V.

Central problem: good subgroup chain may not exist.

Largest maximal subgroup of J_4 has index 173 067 389.





Geometry following Aschbacher

Aschbacher (1984)

G maximal subgroup of $\mathrm{GL}(d, q)$, let $V = GF(q)^d$ be underlying vector space

Geometry following Aschbacher

Aschbacher (1984)

G maximal subgroup of $\mathrm{GL}(d, q)$, let $V = GF(q)^d$ be underlying vector space

- G preserves some natural linear structure associated with the action of G on V, and has normal subgroup related to this structure,

Geometry following Aschbacher

Aschbacher (1984)

G maximal subgroup of $\mathrm{GL}(d, q)$, let $V = GF(q)^d$ be underlying vector space

- G preserves some natural linear structure associated with the action of G on V, and has normal subgroup related to this structure,

  OR

- G is almost simple modulo scalars: $T \leq G/Z \leq Aut(T)$ where T is simple. e.g. $G = \mathrm{SL}(d, q)$, invertible matrices of determinant 1.

Geometry following Aschbacher: general strategy

1. Determine (at least one of) its Aschbacher categories.
2. If $N \lhd G$ exists, recognise N and G/N recursively, ultimately obtaining a composition series for the group.

Geometry following Aschbacher: general strategy

1. Determine (at least one of) its Aschbacher categories.
2. If $N \lhd G$ exists, recognise N and G/N recursively, ultimately obtaining a composition series for the group.

7 categories giving normal subgroup

Prototype: G acts imprimitively on V

G preserves decomposition of V as direct sum

$$V_1 \oplus V_2 \oplus \cdots \oplus V_r$$

of $r > 1$ subspaces of dimension s, which are permuted transitively by G.

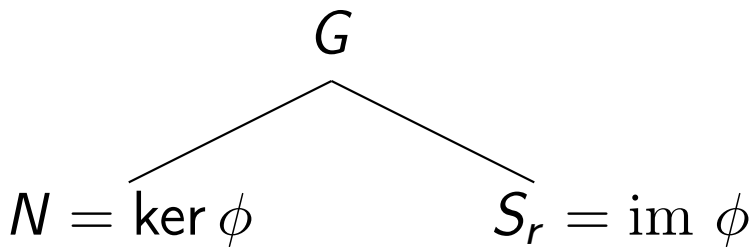
Prototype: G acts imprimitively on V

G preserves decomposition of V as direct sum

$$V_1 \oplus V_2 \oplus \cdots \oplus V_r$$

of $r > 1$ subspaces of dimension s, which are permuted transitively by G.

Then $\phi : G \to S_r$ where $r \leq d$ and $N = \ker \phi$.



Holt, Leedham-Green, O'B & Rees (1996)

Geometry following Aschbacher: general strategy

$G = \langle X \rangle \leq \mathrm{GL}(d, q)$.

1. Determine (at least one of) its Aschbacher categories.
2. If $N \lhd G$ exists, recognise N and G/N recursively, ultimately obtaining a composition series for the group.
3. Otherwise G is either classical group in natural representation or $T \leq G/Z \leq Aut(T)$ where T is simple.

Geometry following Aschbacher: general strategy

$G = \langle X \rangle \leq \mathrm{GL}(d, q)$.

1. Determine (at least one of) its Aschbacher categories.
2. If $N \lhd G$ exists, recognise N and G/N recursively, ultimately obtaining a composition series for the group.
3. Otherwise G is either classical group in natural representation or $T \leq G/Z \leq Aut(T)$ where T is simple.
   - "Reduce" from G to (quasi)simple group L.

Geometry following Aschbacher: general strategy

$G = \langle X \rangle \leq \mathrm{GL}(d, q)$.

1. Determine (at least one of) its Aschbacher categories.
2. If $N \lhd G$ exists, recognise N and G/N recursively, ultimately obtaining a composition series for the group.
3. Otherwise G is either classical group in natural representation or $T \leq G/Z \leq Aut(T)$ where T is simple.
   - "Reduce" from G to (quasi)simple group L.
   - Name L.

Geometry following Aschbacher: general strategy

$G = \langle X \rangle \leq \mathrm{GL}(d, q)$.

1. Determine (at least one of) its Aschbacher categories.
2. If $N \lhd G$ exists, recognise N and G/N recursively, ultimately obtaining a composition series for the group.
3. Otherwise G is either classical group in natural representation or $T \leq G/Z \leq Aut(T)$ where T is simple.
   - "Reduce" from G to (quasi)simple group L.
   - Name L.
   - Set up "constructive isomorphisms" between L and its *standard copy*.

Decide membership of category

Holt, Leedham-Green, Neumann, Praeger, Niemeyer, O'B, Rees, and others: algorithms to decide deciding membership in categories.

Decide membership of category

Holt, Leedham-Green, Neumann, Praeger, Niemeyer, O'B, Rees, and others: algorithms to decide deciding membership in categories.

After 20 years: membership of 5 of the geometric categories are decidable in polynomial time; other are decidable.

Base cases for Aschbacher

- Classical groups in natural repn.
- Other almost simple modulo scalars.

Base cases for Aschbacher

- Classical groups in natural repn.
- Other almost simple modulo scalars.

Liebeck (1985): almost all maximal non-classical subgroups of $\mathrm{GL}(d, q)$ have order at most q^{3d}: much smaller than $O(q^{d^2})$.

Constructive recognition

$C = \langle X \rangle \leq \mathrm{GL}(d, q)$ where C is (quasi)simple.

C is standard or "gold" copy.

C classical: natural copy fixing specific form.

C exceptional: specific faithful repn.

Constructive recognition

$C = \langle X \rangle \leq \mathrm{GL}(d, q)$ where C is (quasi)simple.

C is standard or "gold" copy.

C classical: natural copy fixing specific form.

C exceptional: specific faithful repn.

$G = \langle Y \rangle \cong C$.

Want to construct "effective" isomorphisms
$\phi : C \longmapsto G$ and $\tau : G \longmapsto C$.

Constructive recognition

$C = \langle X \rangle \leq \mathrm{GL}(d, q)$ where C is (quasi)simple.

C is standard or "gold" copy.

C classical: natural copy fixing specific form.

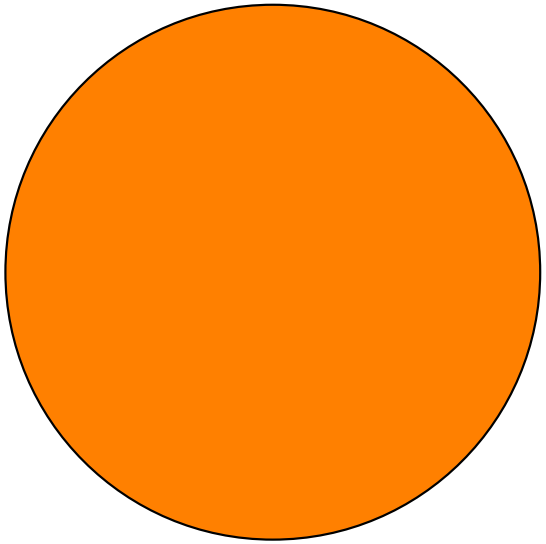
C exceptional: specific faithful repn.

$G = \langle Y \rangle \cong C$.

Want to construct "effective" isomorphisms
$\phi : C \longmapsto G$ and $\tau : G \longmapsto C$.

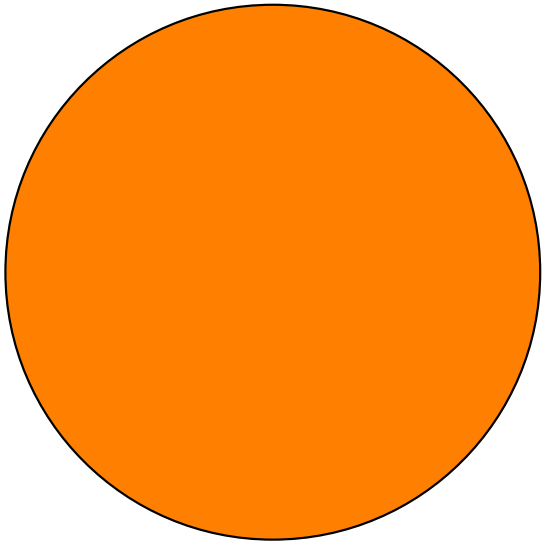
Key idea: use standard generators.

Using standard generators



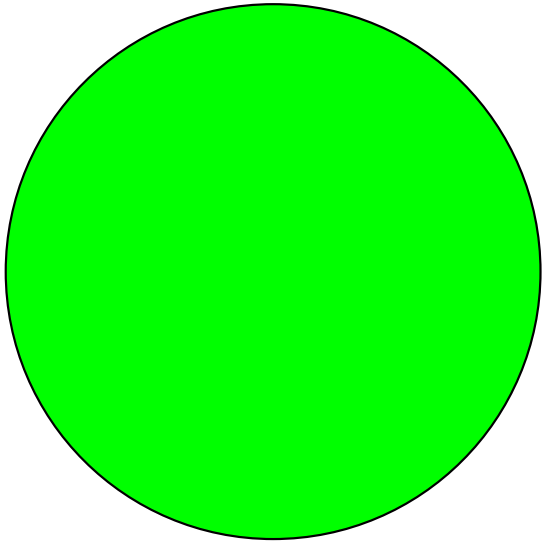
$C = \langle X \rangle$

Using standard generators

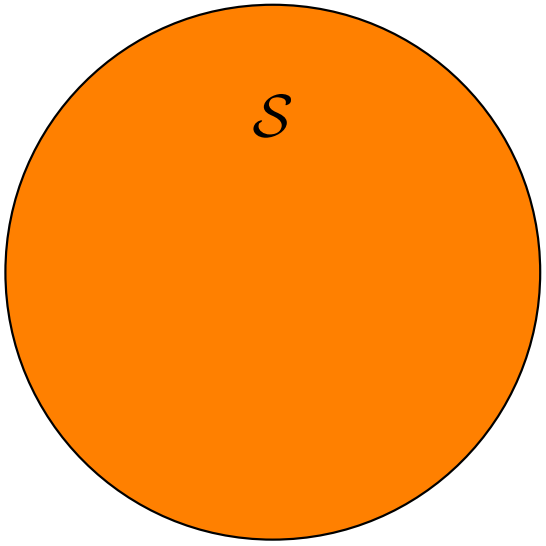


$C = \langle X \rangle$

$\langle Y \rangle = G$



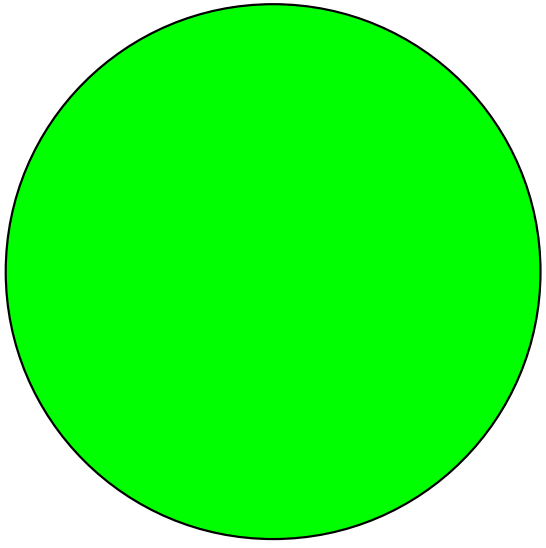
Using standard generators



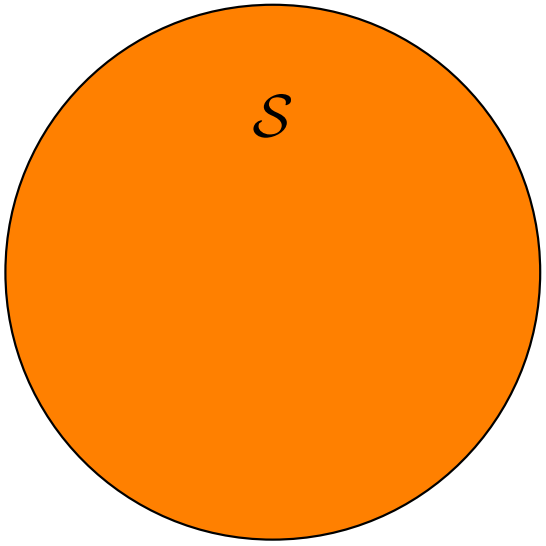
$C = \langle X \rangle$

Find $\mathcal{S} = w(X)$

$\langle Y \rangle = G$



Using standard generators

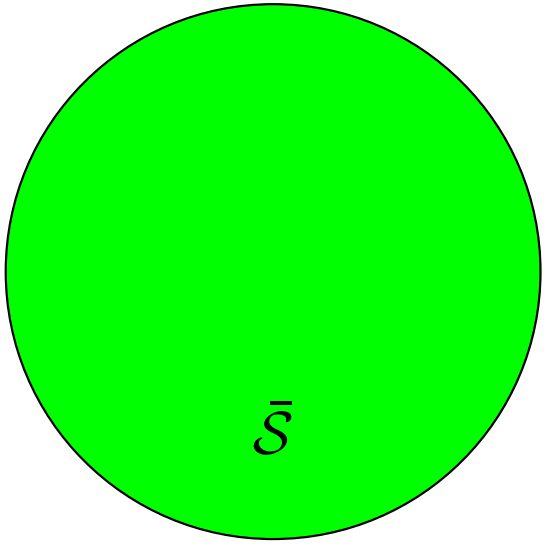


$C = \langle X \rangle$

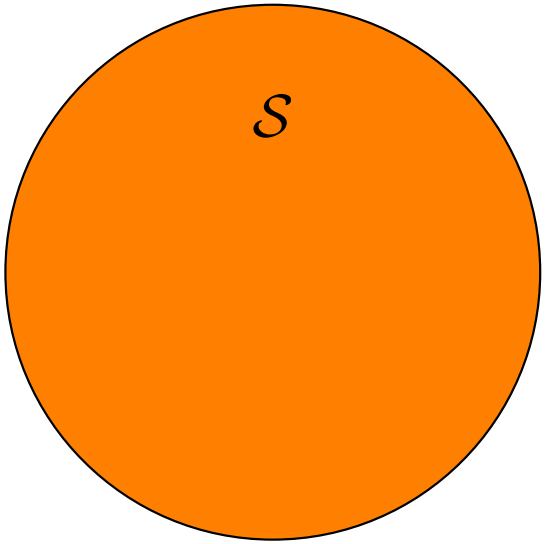
Find $\mathcal{S} = w(X)$

Find $\bar{\mathcal{S}} = w(Y)$

$\langle Y \rangle = G$



Using standard generators



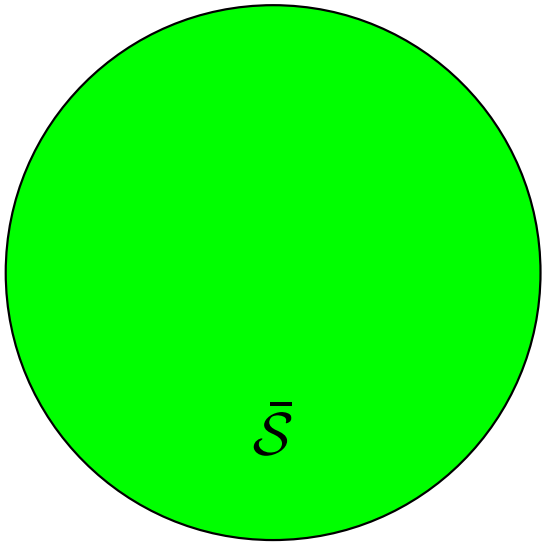
$C = \langle X \rangle$

Find $\mathcal{S} = w(X)$

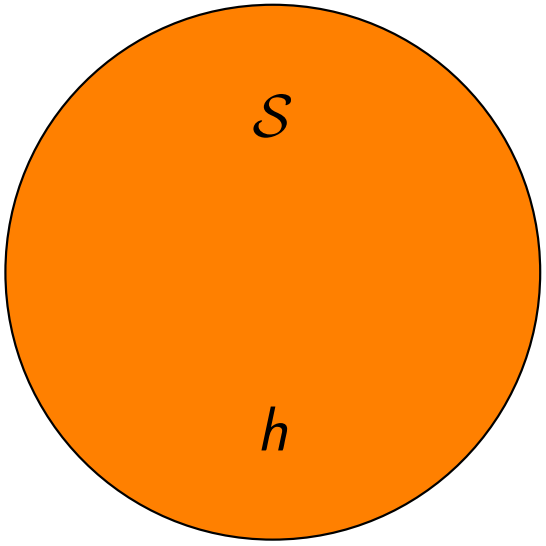
Find $\bar{\mathcal{S}} = w(Y)$

Define $\phi : C \mapsto G : \mathcal{S} \mapsto \bar{\mathcal{S}}$

$\langle Y \rangle = G$



Using standard generators



$C = \langle X \rangle$

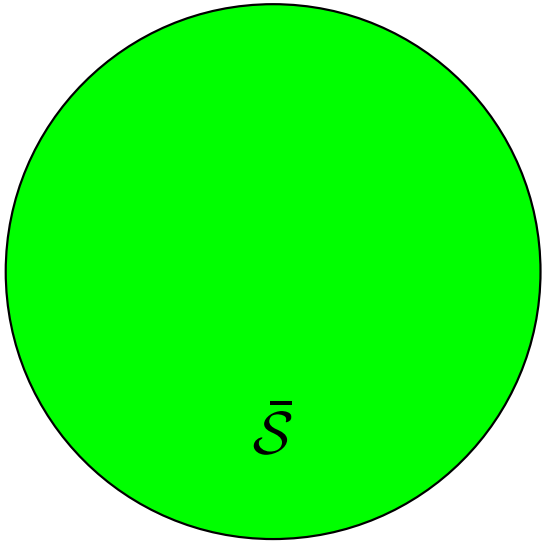
Find $\mathcal{S} = w(X)$

Find $\bar{\mathcal{S}} = w(Y)$

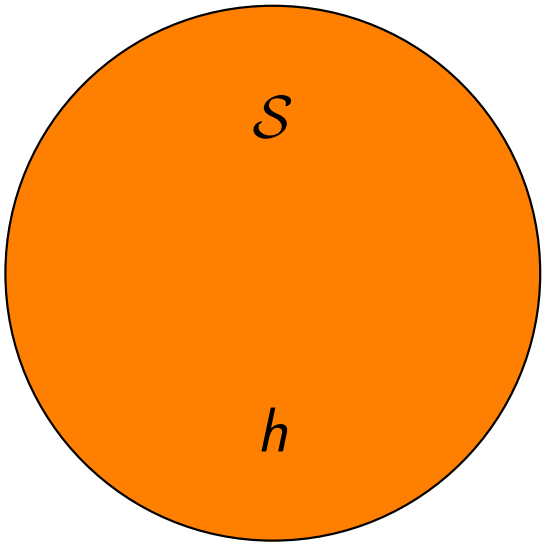
Define $\phi : C \mapsto G : \mathcal{S} \mapsto \bar{\mathcal{S}}$

$h = w(\mathcal{S})$

$\langle Y \rangle = G$



Using standard generators



$C = \langle X \rangle$

Find $\mathcal{S} = w(X)$

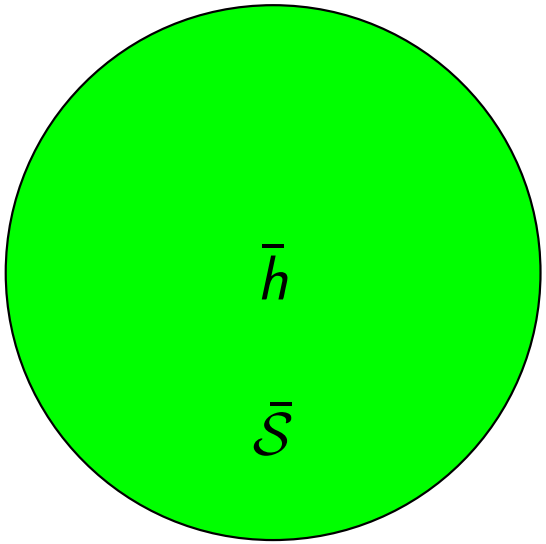
Find $\bar{\mathcal{S}} = w(Y)$

Define $\phi : C \mapsto G : \mathcal{S} \mapsto \bar{\mathcal{S}}$

$h = w(\mathcal{S})$

Thus $\bar{h} = w(\bar{\mathcal{S}})$

$\langle Y \rangle = G$



Application I: Maximal subgroups of classical groups

Kleidmann & Liebeck (1990): describe some maximal subgroups of classical groups where $d \geq 13$.

Bray, Holt & Roney-Dougal (2013): construct generating sets for geometric maximal subgroups, and all maximals for $d \leq 12$.

So obtain $M \leq C := \mathrm{SX}(d, q)$, classical group in natural representation.

Application I: Maximal subgroups of classical groups

Kleidmann & Liebeck (1990): describe some maximal subgroups of classical groups where $d \geq 13$.

Bray, Holt & Roney-Dougal (2013): construct generating sets for geometric maximal subgroups, and all maximals for $d \leq 12$.

So obtain $M \leq C := \mathrm{SX}(d, q)$, classical group in natural representation.

Use $\phi : C \longmapsto G$ to construct image of M in arbitrary representation G.

Application II: Conjugacy classes of elements

Wall (1963): description of conjugacy classes and centralisers of elements of classical groups.

Application II: Conjugacy classes of elements

Wall (1963): description of conjugacy classes and centralisers of elements of classical groups.

Liebeck and O'Brien, under development: algorithm which writes down classes in standard copy of group of Lie type.

Standard generators for $\mathrm{SL}(d, q)$

Leedham-Green & O'B (2009).
Natural module V for $C = \mathrm{SL}(d, q)$ with basis $\{e_1, \ldots, e_d\}$.
Define standard generators $\mathcal{S} = \{s, \delta, u, v\}$ for C:

Standard generators for $\mathrm{SL}(d, q)$

Leedham-Green & O'B (2009).
Natural module V for $C = \mathrm{SL}(d, q)$ with basis $\{e_1, \ldots, e_d\}$.
Define standard generators $\mathcal{S} = \{s, \delta, u, v\}$ for C:

s, δ, u lie in copy of $\mathrm{SL}(2, q)$ and act on $\langle e_1, e_2 \rangle$ as:

$$s = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \quad \delta = \begin{pmatrix} \omega & 0 \\ 0 & \omega^{-1} \end{pmatrix} \quad u = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$$

v maps

$$e_1 \mapsto e_d \mapsto -e_{d-1} \mapsto -e_{d-2} \mapsto -e_{d-3} \cdots \mapsto -e_1$$

Standard generators for $\mathrm{SL}(d, q)$

Leedham-Green & O'B (2009).
Natural module V for $C = \mathrm{SL}(d, q)$ with basis $\{e_1, \ldots, e_d\}$.
Define standard generators $\mathcal{S} = \{s, \delta, u, v\}$ for C:

s, δ, u lie in copy of $\mathrm{SL}(2, q)$ and act on $\langle e_1, e_2 \rangle$ as:

$$s = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \quad \delta = \begin{pmatrix} \omega & 0 \\ 0 & \omega^{-1} \end{pmatrix} \quad u = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$$

v maps

$$e_1 \mapsto e_d \mapsto -e_{d-1} \mapsto -e_{d-2} \mapsto -e_{d-3} \cdots \mapsto -e_1$$

Given $h \in C$, via echelonisation write $h = w(\mathcal{S})$.

Basic algorithm to construct standard generators

- Construct two subgroups H and K in G so $m \simeq d/2$ and

$$H = \begin{pmatrix} \mathrm{SL}_m & \\ & 1_{d-m} \end{pmatrix} \quad \text{and} \quad K = \begin{pmatrix} 1_m & \\ & \mathrm{SL}_{d-m} \end{pmatrix}$$

Basic algorithm to construct standard generators

- Construct two subgroups H and K in G so $m \simeq d/2$ and

$$H = \begin{pmatrix} \mathrm{SL}_m & \\ & 1_{d-m} \end{pmatrix} \quad \text{and} \quad K = \begin{pmatrix} 1_m & \\ & \mathrm{SL}_{d-m} \end{pmatrix}$$

- Recursively construct standard generators $\mathcal{S}_H$ and $\mathcal{S}_K$ for H and K

Basic algorithm to construct standard generators

- Construct two subgroups H and K in G so $m \simeq d/2$ and

$$H = \begin{pmatrix} \mathrm{SL}_m & \\ & 1_{d-m} \end{pmatrix} \quad \text{and} \quad K = \begin{pmatrix} 1_m & \\ & \mathrm{SL}_{d-m} \end{pmatrix}$$

- Recursively construct standard generators $\mathcal{S}_H$ and $\mathcal{S}_K$ for H and K
- all but cycle from standard generators for G contained in $\mathcal{S}_H$

Basic algorithm to construct standard generators

- Construct two subgroups H and K in G so $m \simeq d/2$ and

$$H = \begin{pmatrix} \mathrm{SL}_m & \\ & 1_{d-m} \end{pmatrix} \quad \text{and} \quad K = \begin{pmatrix} 1_m & \\ & \mathrm{SL}_{d-m} \end{pmatrix}$$

- Recursively construct standard generators $\mathcal{S}_H$ and $\mathcal{S}_K$ for H and K
- all but cycle from standard generators for G contained in $\mathcal{S}_H$
- cycle is constructed by glueing two cycles from $\mathcal{S}_H$ and $\mathcal{S}_K$.

Basic algorithm to construct standard generators

- Construct two subgroups H and K in G so $m \simeq d/2$ and

$$H = \begin{pmatrix} \mathrm{SL}_m & \\ & 1_{d-m} \end{pmatrix} \quad \text{and} \quad K = \begin{pmatrix} 1_m & \\ & \mathrm{SL}_{d-m} \end{pmatrix}$$

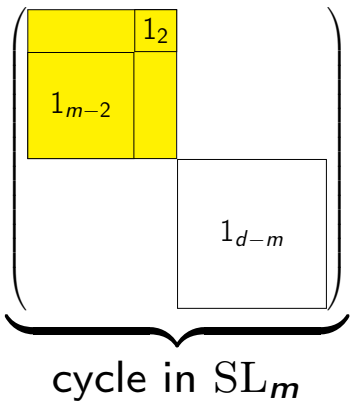
- Recursively construct standard generators $\mathcal{S}_H$ and $\mathcal{S}_K$ for H and K
- all but cycle from standard generators for G contained in $\mathcal{S}_H$
- cycle is constructed by glueing two cycles from $\mathcal{S}_H$ and $\mathcal{S}_K$. e.g. if $G = \mathrm{SL}(d, q)$ with even d and q, then

Basic algorithm to construct standard generators

- Construct two subgroups H and K in G so $m \simeq d/2$ and

$$H = \begin{pmatrix} \mathrm{SL}_m & \\ & 1_{d-m} \end{pmatrix} \quad \text{and} \quad K = \begin{pmatrix} 1_m & \\ & \mathrm{SL}_{d-m} \end{pmatrix}$$

- Recursively construct standard generators $\mathcal{S}_H$ and $\mathcal{S}_K$ for H and K
- all but cycle from standard generators for G contained in $\mathcal{S}_H$
- cycle is constructed by glueing two cycles from $\mathcal{S}_H$ and $\mathcal{S}_K$. e.g. if $G = \mathrm{SL}(d, q)$ with even d and q, then

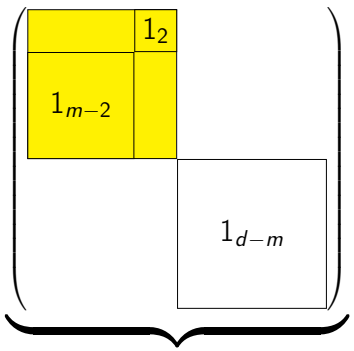


Basic algorithm to construct standard generators

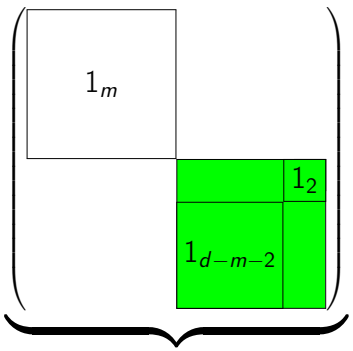
- Construct two subgroups H and K in G so $m \simeq d/2$ and

$$H = \begin{pmatrix} \mathrm{SL}_m & \\ & 1_{d-m} \end{pmatrix} \quad \text{and} \quad K = \begin{pmatrix} 1_m & \\ & \mathrm{SL}_{d-m} \end{pmatrix}$$

- Recursively construct standard generators $\mathcal{S}_H$ and $\mathcal{S}_K$ for H and K
- all but cycle from standard generators for G contained in $\mathcal{S}_H$
- cycle is constructed by glueing two cycles from $\mathcal{S}_H$ and $\mathcal{S}_K$. e.g. if $G = \mathrm{SL}(d, q)$ with even d and q, then



cycle in SL_m



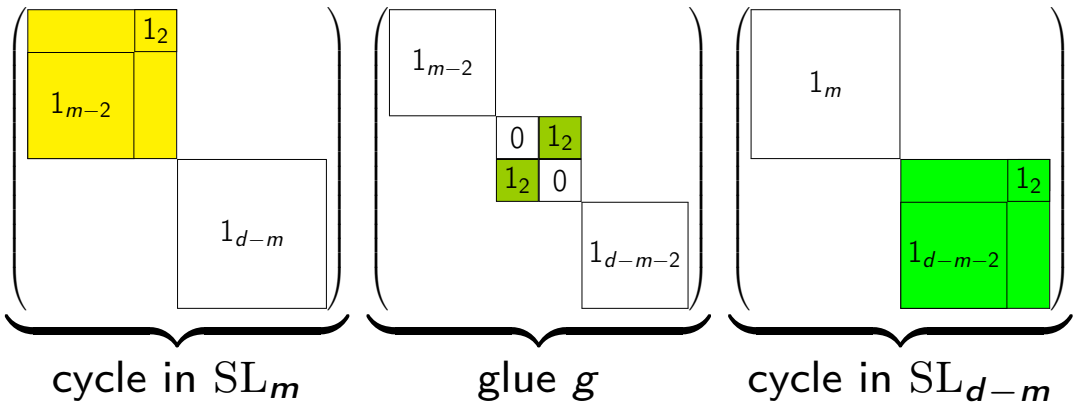
cycle in SL_{d-m}

Basic algorithm to construct standard generators

- Construct two subgroups H and K in G so $m \simeq d/2$ and

$$H = \begin{pmatrix} \mathrm{SL}_m & \\ & 1_{d-m} \end{pmatrix} \quad \text{and} \quad K = \begin{pmatrix} 1_m & \\ & \mathrm{SL}_{d-m} \end{pmatrix}$$

- Recursively construct standard generators $\mathcal{S}_H$ and $\mathcal{S}_K$ for H and K
- all but cycle from standard generators for G contained in $\mathcal{S}_H$
- cycle is constructed by glueing two cycles from $\mathcal{S}_H$ and $\mathcal{S}_K$. e.g. if $G = \mathrm{SL}(d, q)$ with even d and q, then

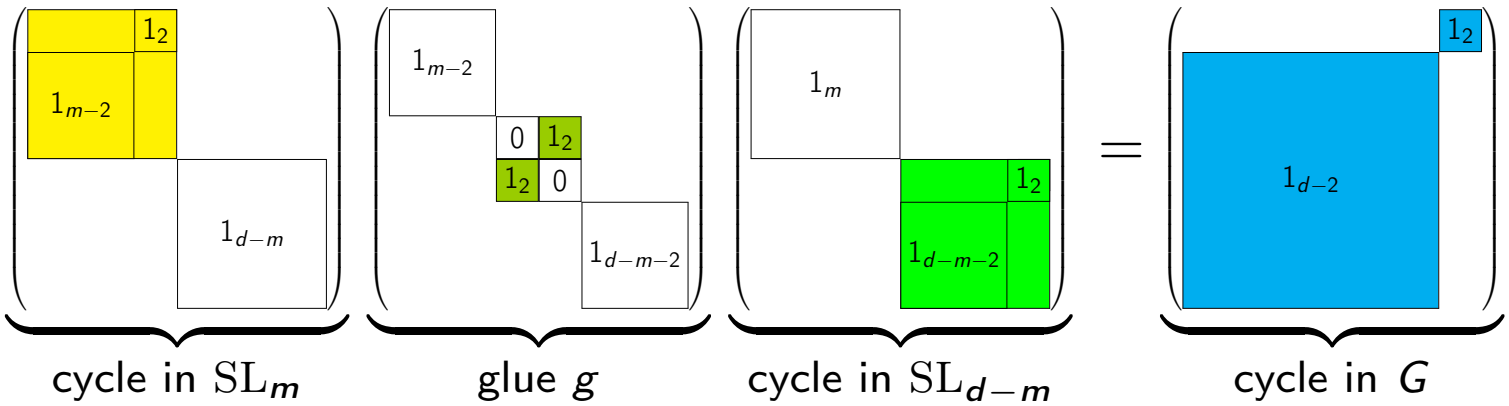


Basic algorithm to construct standard generators

- Construct two subgroups H and K in G so $m \simeq d/2$ and

$$H = \begin{pmatrix} \mathrm{SL}_m & \\ & 1_{d-m} \end{pmatrix} \quad \text{and} \quad K = \begin{pmatrix} 1_m & \\ & \mathrm{SL}_{d-m} \end{pmatrix}$$

- Recursively construct standard generators $\mathcal{S}_H$ and $\mathcal{S}_K$ for H and K
- all but cycle from standard generators for G contained in $\mathcal{S}_H$
- cycle is constructed by glueing two cycles from $\mathcal{S}_H$ and $\mathcal{S}_K$. e.g. if $G = \mathrm{SL}(d, q)$ with even d and q, then



Leedham-Green and O'B, 2009; Dietrich, L-G, Lübeck, O'B, 2013; D, L-G, O'B, 2015

Theorem

There is a Las Vegas algorithm that takes as input $G \cong \mathrm{SX}(d, q) = \langle X \rangle$, and returns standard generators $\mathcal{S}$ for G as words in X. The algorithm has complexity $O(d^4 \log q)$ measured in field operations.

Exceptional groups

Howlett, Rylands, Taylor (2001): minimal degree faithful matrix representation for each exceptional group. Designate as standard copy.

Exceptional groups

Howlett, Rylands, Taylor (2001): minimal degree faithful matrix representation for each exceptional group. Designate as standard copy.

Standard generators: those which satisfy reduced Curtis-Steinberg-Tits presentations.

Theorem (Liebeck & O'B; TAMS, 2014)

Can construct standard generators for representations of exceptional groups of rank at least 2 in polynomial time.

Exceptional groups

Howlett, Rylands, Taylor (2001): minimal degree faithful matrix representation for each exceptional group. Designate as standard copy.

Standard generators: those which satisfy reduced Curtis-Steinberg-Tits presentations.

Theorem (Liebeck & O'B; TAMS, 2014)

Can construct standard generators for representations of exceptional groups of rank at least 2 in polynomial time.

Bäärnhielm (2006, 2014): Algorithms to construct standard generators for matrix representations of Suzuki, large and small Ree groups.

Results

Key: *centralisers of involutions* and statistical group theory.

Results

Key: *centralisers of involutions* and statistical group theory.

Use centralisers of involutions to obtain smaller rank classical groups as subgroups of G.

Results

Key: *centralisers of involutions* and statistical group theory.

Use centralisers of involutions to obtain smaller rank classical groups as subgroups of G.

- Bray, 2001; Holmes et al. 2008; Parker & Wilson, 2009; Liebeck, 2015: can construct centraliser of involution in polynomial time.

Results

Key: *centralisers of involutions* and statistical group theory.

Use centralisers of involutions to obtain smaller rank classical groups as subgroups of G.

- Bray, 2001; Holmes et al. 2008; Parker & Wilson, 2009; Liebeck, 2015: can construct centraliser of involution in polynomial time.
- Dietrich et al. 2013; $G = \langle X \rangle \cong \mathrm{SX}(d, q)$. Polynomial time algorithm to construct involution as word in X.

Results

Key: *centralisers of involutions* and statistical group theory.

Use centralisers of involutions to obtain smaller rank classical groups as subgroups of G.

- Bray, 2001; Holmes et al. 2008; Parker & Wilson, 2009; Liebeck, 2015: can construct centraliser of involution in polynomial time.
- Dietrich et al. 2013; $G = \langle X \rangle \cong \mathrm{SX}(d, q)$. Polynomial time algorithm to construct involution as word in X.
- Praeger et al., 2015; In polynomial time, can construct $H \leq G \cong SX(d, q)$ where $H \cong SX(m, q)$ and $m \approx d/2$.

Constructive recognition for other families

- A_n: Bratus & Pak (2000), Holt; Beals et al. (2001-05); Jambor et al. (2013). Black-box.

Constructive recognition for other families

- A_n: Bratus & Pak (2000), Holt; Beals et al. (2001-05); Jambor et al. (2013). Black-box.
- Sporadics: standard generators and black box algorithms to construct these by Bray, Wilson; use reduction to 3 centralisers of involutions (Holmes *et al.*, 2008).

Writing elements as words in standard generators

Given $\bar{\mathcal{S}}$ and $g \in G$, write $g = w(\bar{\mathcal{S}})$.

Writing elements as words in standard generators

Given $\bar{\mathcal{S}}$ and $g \in G$, write $g = w(\bar{\mathcal{S}})$.

- Classical groups, absolutely irred reps in defining char: Costi (2009).

Writing elements as words in standard generators

Given $\bar{\mathcal{S}}$ and $g \in G$, write $g = w(\bar{\mathcal{S}})$.

- Classical groups, absolutely irred reps in defining char: Costi (2009).
- Classical groups, black box: Schneider (2014).

Writing elements as words in standard generators

Given $\bar{S}$ and $g \in G$, write $g = w(\bar{S})$.

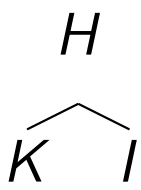
- Classical groups, absolutely irred reps in defining char: Costi (2009).
- Classical groups, black box: Schneider (2014).
- Exceptional groups.
  - Black: Kantor & Magaard (2012).
  - Absolutely irred reps in defining char: Cohen, Murray, Taylor (2004); Cohen & Taylor (2014).

The composition tree for G

Bäärnhielm, Leedham-Green & O'B
Neunhöffer & Seress

The composition tree for G

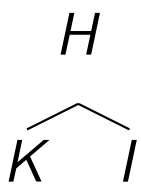
Bäärnhielm, Leedham-Green & O'B
Neunhöffer & Seress



- Node: section H of G.

The composition tree for G

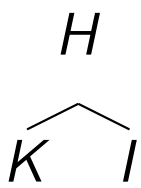
Bäärnhielm, Leedham-Green & O'B
Neunhöffer & Seress



- Node: section H of G.
- Image I: image under homomorphism or isomorphism.

The composition tree for G

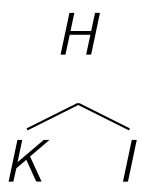
Bäärnhielm, Leedham-Green & O'B
Neunhöffer & Seress



- Node: section H of G.
- Image I: image under homomorphism or isomorphism.
- Kernel K.

The composition tree for G

Bäärnhielm, Leedham-Green & O'B
Neunhöffer & Seress



- Node: section H of G.
- Image I: image under homomorphism or isomorphism.
- Kernel K.
- Leaf is "composition factor" of G: simple modulo scalars. Cyclic not necessarily of prime order.

Verifying the outcome

Construction of tree relies on Monte Carlo algorithms.

Verifying the outcome

Construction of tree relies on Monte Carlo algorithms.

Obtain presentation for G on "nice generators" Y. If Y satisfies presentation, then we have verified tree.

Verifying the outcome

Construction of tree relies on Monte Carlo algorithms.

Obtain presentation for G on "nice generators" Y. If Y satisfies presentation, then we have verified tree.

To obtain presentation for node: need only presentation for associated kernel and image.

Verifying the outcome

Construction of tree relies on Monte Carlo algorithms.

Obtain presentation for G on "nice generators" Y. If Y satisfies presentation, then we have verified tree.

To obtain presentation for node: need only presentation for associated kernel and image.

So inductively need to know presentations **only for the leaves** – or composition factors.

Theorem (Guralnick, Kantor, Kassabov, Lubotzky, 2008)

Every non-abelian finite simple group of rank n over $\mathrm{GF}(q)$, with possible exception of Ree groups ${}^2G_2(q)$, has a presentation with a bounded number of generators and relations and total length $O(\log n + \log q)$.

Theorem (Guralnick, Kantor, Kassabov, Lubotzky, 2008)

Every non-abelian finite simple group of rank n over $\mathrm{GF}(q)$, with possible exception of Ree groups ${}^2G_2(q)$, has a presentation with a bounded number of generators and relations and total length $O(\log n + \log q)$.

Leedham-Green and O'B (2014): explicit short presentations for the classical groups on our standard generators.

Theorem (Guralnick, Kantor, Kassabov, Lubotzky, 2008)

Every non-abelian finite simple group of rank n over $\mathrm{GF}(q)$, with possible exception of Ree groups ${}^2G_2(q)$, has a presentation with a bounded number of generators and relations and total length $O(\log n + \log q)$.

Leedham-Green and O'B (2014): explicit short presentations for the classical groups on our standard generators.

Previous best: Babai *et al.* (1997) presentation of length $O(\log^2 |G|)$. Reduced Curtis-Steinberg-Tits presentations for groups of Lie rank at least 2.

Theorem (Guralnick, Kantor, Kassabov, Lubotzky, 2008)

Every non-abelian finite simple group of rank n over $\mathrm{GF}(q)$, with possible exception of Ree groups ${}^2G_2(q)$, has a presentation with a bounded number of generators and relations and total length $O(\log n + \log q)$.

Leedham-Green and O'B (2014): explicit short presentations for the classical groups on our standard generators.

Previous best: Babai *et al.* (1997) presentation of length $O(\log^2 |G|)$. Reduced Curtis-Steinberg-Tits presentations for groups of Lie rank at least 2.

Use these presentations for exceptional groups.

Output of CompositionTree

Given $G = \langle X \rangle \leq \mathrm{GL}(\mathrm{d}, \mathrm{q})$ as input.
Output:

- a composition series: $1 = G_0 \lhd G_1 \lhd G_2 \cdots \lhd G_m = G$.
- A representation $S_k = \langle X_k \rangle$ of G_k / G_{k-1}
- Effective maps $\tau_k : G_k \to S_k$, $\phi_k : S_k \to G_k$
  τ_k epimorphism with kernel G_{k-1}
- Map to write $g \in G$ as word in X.

Output of CompositionTree

Given $G = \langle X \rangle \leq \mathrm{GL}(\mathrm{d}, \mathrm{q})$ as input.
Output:

- a composition series: $1 = G_0 \lhd G_1 \lhd G_2 \cdots \lhd G_m = G$.
- A representation $S_k = \langle X_k \rangle$ of G_k / G_{k-1}
- Effective maps $\tau_k : G_k \to S_k$, $\phi_k : S_k \to G_k$
  τ_k epimorphism with kernel G_{k-1}
- Map to write $g \in G$ as word in X.

Construct presentation for group defined by tree and verify that G satisfies the relations.

From composition series to $\mathcal{C}$

Bäärnhielm, Holt, Leedham-Green & O'B (2014): refine composition series obtained from "geometric model" to obtain chief series reflecting characteristic structure.

Holt: developed Soluble Radical model algorithms using tree as infrastructure.

Publicly available in Magma; parts available in GAP.