Finding Subgroups

Alexander Hulpke
Department of Mathematics
Colorado State University
Fort Collins, CO, 80523, USA
http://www.math.colostate.edu/~hulpke

A Popular Problem

Given a finite (permutation, matrix) group G, find subgroups: Frequent task (CANNON, 2011)

-Not all, but specific properties, one or all.

-Up to conjugacy U ~ $g^{-1}Ug$.

There is no universal algorithm (how to specify desired properties). Best strategies:

- Search in smaller groups (sub- or factor-)

- Use hooks in existing algorithms (or put in own hooks). Thus survey what exists.

Admission of Failure

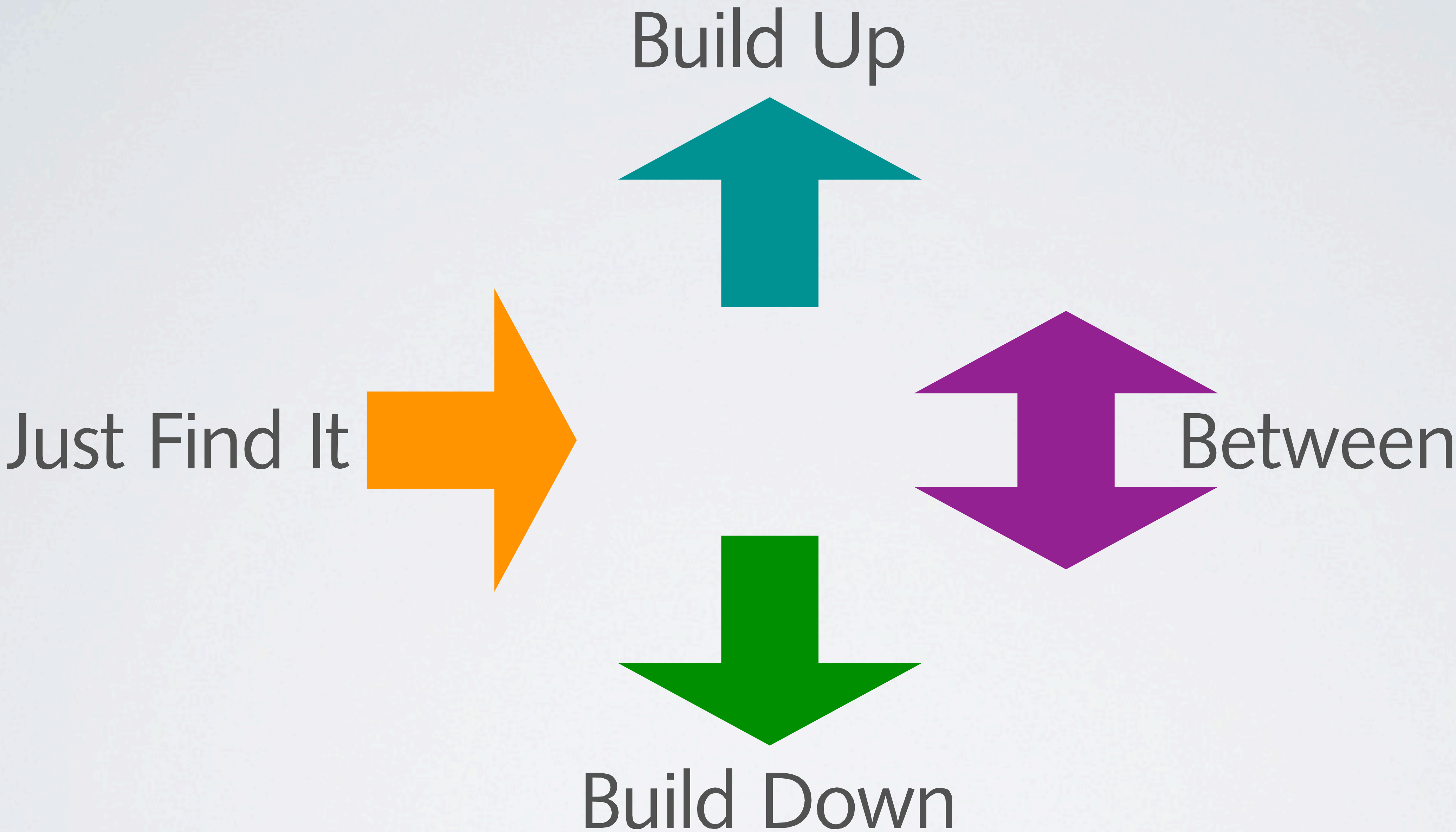
How to organize this?

One obviously would like an organization by task.

I only can give a classification by algorithm (showing you a nice wrench set instead of repairing your car).

Still, I hope this might help to give an idea where to put limitation hooks (and what kinds of hooks are suitable).

Possible Strategies



Direct Constructions

Direct methods for the construction of particular subgroups without the need to build part of the subgroup lattice.

Direct Constructions

Direct methods for the construction of particular subgroups without the need to build part of the subgroup lattice.

- Cyclic Subgroups
- Stabilizers
- Sylow Subgroups
- PCore
- Composition Series, Radical

Cyclic Subgroups

There are algorithms that compute conjugacy class representatives of elements.

Eliminate coprime powers g^x, $\gcd(|g|,x)=1$, to get cyclic subgroups up to conjugacy.

Cyclic Subgroups

There are algorithms that compute conjugacy class representatives of elements.

Eliminate coprime powers g $\gcd(|g|,x)=1$, to get cyclic subgroups up to conjugacy.

Cyclic Subgroups

There are algorithms that compute conjugacy class representatives of elements.

Eliminate coprime powers g^x, $\gcd(|g|,x)=1$, to get cyclic subgroups up to conjugacy.

- Cyclic Subgroups
- Stabilizers
- Sylow Subgroups
- PCore
- Composition Series, Radical

Orbit / Stabilizer

If G acts on Ω, the Orbit/Stabilizer algorithm finds the orbit of $\omega \in \Omega$ by repeated computation of images under generators.

Schreier generators for $\mathrm{Stab}_G(\omega)$.

- Cyclic Subgroups
- Stabilizers
- Sylow Subgroups
- PCore
- Composition Series, Radical

Orbit / Stabilizer

If G acts on Ω, the Orbit/ Stabilizer algorithm finds the orbit of $\omega \in \Omega$ by repeated computation of images under generators.

Schreier generators for $\mathrm{Stab}_G(\omega)$.

Problems:

- Cost (time and memory) is proportional to stabilizer index.
- Not every subgroup is a natural stabilizer.

- Cyclic Subgroups
- Stabilizers
- Sylow Subgroups
- PCore
- Composition Series, Radical

Backtrack

Backtrack search algorithms for particular actions of perm. groups:

- (Point, Tuple), Set stabilizer
- Element Centralizer
- Subgroup Normalizer
- Intersection

- Cyclic Subgroups
- Stabilizers
- Sylow Subgroups
- PCore
- Composition Series, Radical

Backtrack

Backtrack search algorithms for particular actions of perm. groups:

- (Point, Tuple), Set stabilizer
- Element Centralizer
- Subgroup Normalizer
- Intersection

- Cyclic Subgroups
- Stabilizers
- Sylow Subgroups
- PCore
- Composition Series, Radical

Problems:

- Inherently exponential time, hard to analyze.
- Very hard to extend.

Sylow Structure

Sylow subgroups have a nontrivial centre.

They thus lie in the centralizer of an element of order p. Compute these centralizers and then search in the centre factor.

PCore as intersection of conjugates.

- Cyclic Subgroups
- Stabilizers
- → Sylow Subgroups
- → PCore
- Composition Series, Radical

Composition Structure

Composition series by finding kernels of nontrivial homomorphisms. (Transitive, Imprimitive actions; them analysis of primitive case.)

(Solvable) Radical is the kernel of the action of G on nonabelian composition factors.

- Cyclic Subgroups
- Stabilizers
- Sylow Subgroups
- PCore
- Composition Series, Radical

Build Up

It seems plausible to try to build up subgroups from smaller ones, starting with cyclic subgroups.

Build Up

It seems plausible to try to build up subgroups from smaller ones, starting with cyclic subgroups.

- Find Generators
- Cyclic Extension

Indeed this is possible. A main difficulty is that it will generate lots of irrelevant and redundant groups and conjugacy tests for subgroups are expensive.

Find Generators

Any $U < G$ is generated by a set of elements $u_1,\ldots,u_k$. Each u_i lies in conj. class.

Thus classify such tuples (with particular properties, e.g. presentation) up to conjugacy ($C_G(u)$ orbits on x^G are given by double cosets $C_G(x)\backslash G/C_G(u)$).

GAP-Function: `IsomorphicSubgroups`, `TomDataSubgroupsAlmostSimple`

Hook: Internal function `MorClassLoop`

Find Generators

Any $U<G$ is generated by a set of elements $u_1,\ldots,u_k$. Each u_i lies in conj. class.

Thus classify such tuples (with particular properties, e.g. presentation) up to conjugacy ($C_G(u)$ orbits on x^G are given by double cosets $C_G(x)\backslash G/C_G(u)$).

GAP-Function: `IsomorphicSubgroups`, `TomDataSubgroupsAlmostSimple`

Hook: Internal function `MorClassLoop`

Cyclic Extension

More systematically, construct U as $U = \langle V, x \rangle$ with smaller V, $x \in N_G(V)$.

Seed with perfect subgroups. Find via generator images from catalog.

GAP-Function: `LatticeByCyclicExtension`

Hook: Function to discard V, flag to skip nontrivial perfect subgroups.

Problems: Many, expensive conjugacy tests. Store by list of cyclic subgroups.

Cyclic Extension

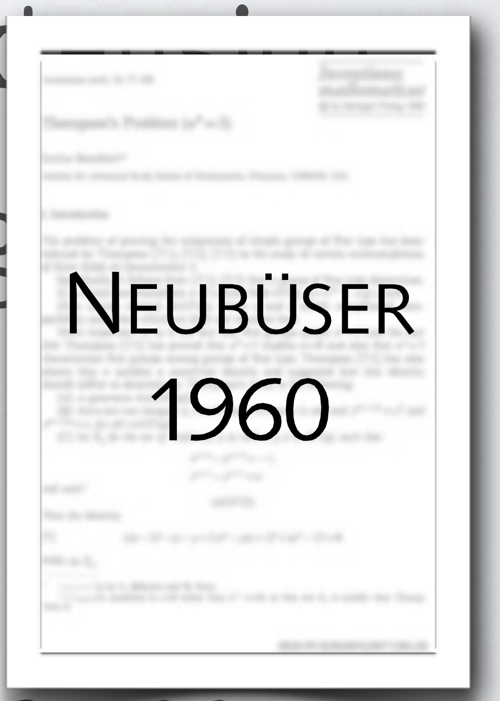
More systematically, construct U as $U = \langle V, x \rangle$ with smaller V, $x \in N_G(V)$.

Seed with perfect subgroups. Find via generator images from catalog.

GAP-Function: `LatticeByCyclicExt`

Hook: Function to discard V, flag nontrivial perfect subgroups.

Problems: Many, expensive conj tests. Store by list of cyclic subgroups.



Cyclic Extension

More systematically, construct U as $U = \langle V, x \rangle$ with smaller V, $x \in N_G(V)$.

Seed with perfect subgroups. Find via generator images from catalog.

GAP-Function: `LatticeByCyclicExtension`

Hook: Function to discard V, flag to skip nontrivial perfect subgroups.

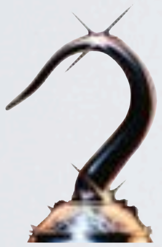
Problems: Many, expensive conjugacy tests. Store by list of cyclic subgroups.

Cyclic Extension

More systematically, construct U as $U = \langle V, x \rangle$ with smaller V, $x \in N_G(V)$.

Seed with perfect subgroups. Find via generator images from catalog.

GAP-Function: `LatticeByCyclicExtension`



Hook: Function to discard V, flag to skip nontrivial perfect subgroups.

Problems: Many, expensive conjugacy tests. Store by list of cyclic subgroups.

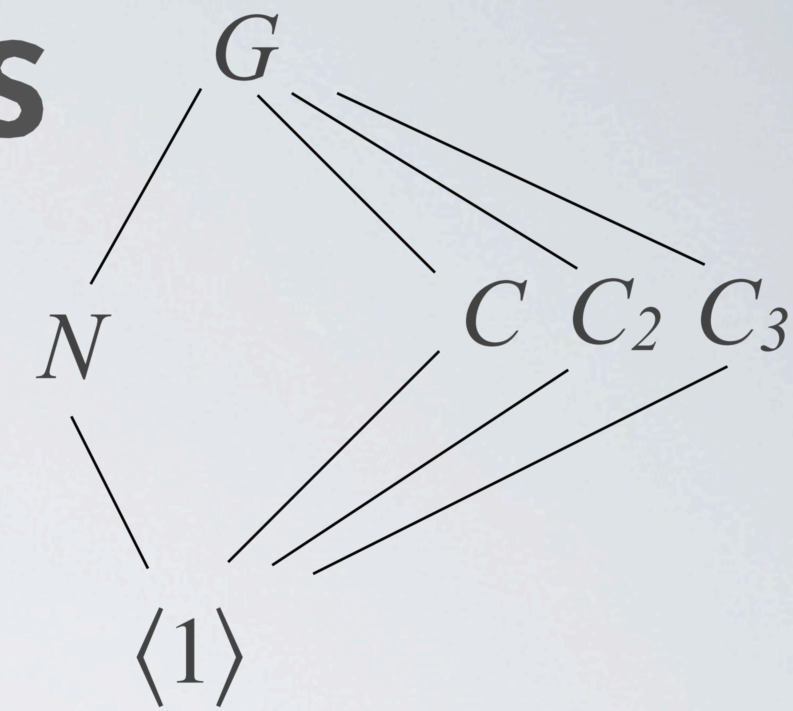
Complements

Intermezzo

classes. Sufficient to find representatives.

As $C \cong G/N$, consider as homomorphisms $G \to G$ with kernel N. Every $g \in G$ is mapped to $g \cdot n_g$ with the cofactor $n_g \in$ N. Find these n_g for generators of G/N.

Complements



For $N \lhd G$, a *complement* is a subgroup $C \leq G$ such that $N \cap C = \langle 1 \rangle$ and $NC = G$.

Complements fall in classes under N. In general there can be multiple (or no) classes. Sufficient to find representatives.

As $C \cong G/N$, consider as homomorphisms $G \to G$ with kernel N. Every $g \in G$ is mapped to $g \cdot n_g$ with the cofactor $n_g \in$ N. Find these n_g for generators of G/N.

TECHNICAL DETAILS

Linear Algebra for Cocycles

If N is elementary abelian then one can *collect* these equations into a G/N-part and an N-part. E.g. a relator $Na{\cdot}Nb{\cdot}Na$ for G/N becomes

$$a{\cdot}n_a{\cdot}b{\cdot}n_b{\cdot}a{\cdot}n_a = a{\cdot}b{\cdot}n_a{}^b{\cdot}n_b{\cdot}a{\cdot}n_a = \underline{a{\cdot}b{\cdot}a}{\cdot}n_a{}^{ba}{\cdot}n_b{}^a{\cdot}n_a$$

Thus in G, $(a{\cdot}b{\cdot}a)^{-1}\in N$ equals the (linearly written!) vector space element $n_a{\cdot}(M_{ba}+1)+ n_b{\cdot}M_a$ where M_x is the action induced by x on N. For given M_x, the n_y are variables.

These equations for all relators yield a linear inhomogeneous system of equations, solutions are cocycles Z^1. Conjugacy by N gives B^1.

TECHNICAL DETAILS

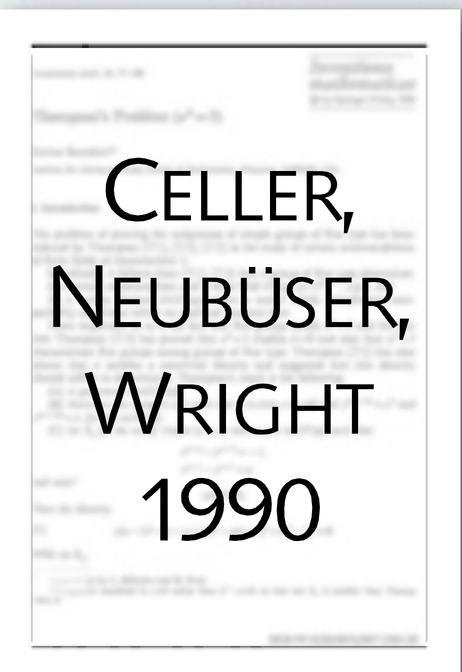
Linear Algebra for Cocycles

If N is elementary abelian then one can *collect* these equations into a G/N-part and an N-part. E.g. a relator $Na\cdot Nb\cdot Na$ for G/N becomes

$$a\cdot n_a\cdot b\cdot n_b\cdot a\cdot n_a = a\cdot b\cdot n_a^{b}\cdot n_b\cdot a\cdot n_a = \underline{a\cdot b\cdot a}\cdot n_a^{ba}\cdot n_b^{a}\cdot n_a$$

Thus in G, $(a\cdot b\cdot a)^{-1}\in N$ equals the (linearly written!) vector space element $n_a\cdot(M_{ba}+1$ $n_b\cdot M_a$ where M_x is the action induced by N. For given M_x, the n_y are variables.

These equations for all relators yield a lin inhomogeneous system of equations, so are cocycles Z^1. Conjugacy by N gives B^1.



TECHNICAL DETAILS

Linear Algebra for Cocycles

If N is elementary abelian then one can *collect* these equations into a G/N-part and an N-part. E.g. a relator $Na{\cdot}Nb{\cdot}Na$ for G/N becomes

$$a{\cdot}n_a\cdot b{\cdot}n_b{\cdot}a{\cdot}n_a = a{\cdot}b{\cdot}n_a^{\,b}{\cdot}n_b{\cdot}a{\cdot}n_a = \underline{a{\cdot}b\cdot a}{\cdot}n_a^{\,ba}{\cdot}n_b^{\,a}{\cdot}n_a$$

Thus in G, $(a{\cdot}b\cdot a)^{-1}\in N$ equals the (linearly written!) vector space element $n_a{\cdot}(M_{ba}+1)+n_b{\cdot}M_a$ where M_x is the action induced by x on N. For given M_x, the n_y are variables.

These equations for all relators yield a linear inhomogeneous system of equations, solutions are cocycles Z^1. Conjugacy by N gives B^1.

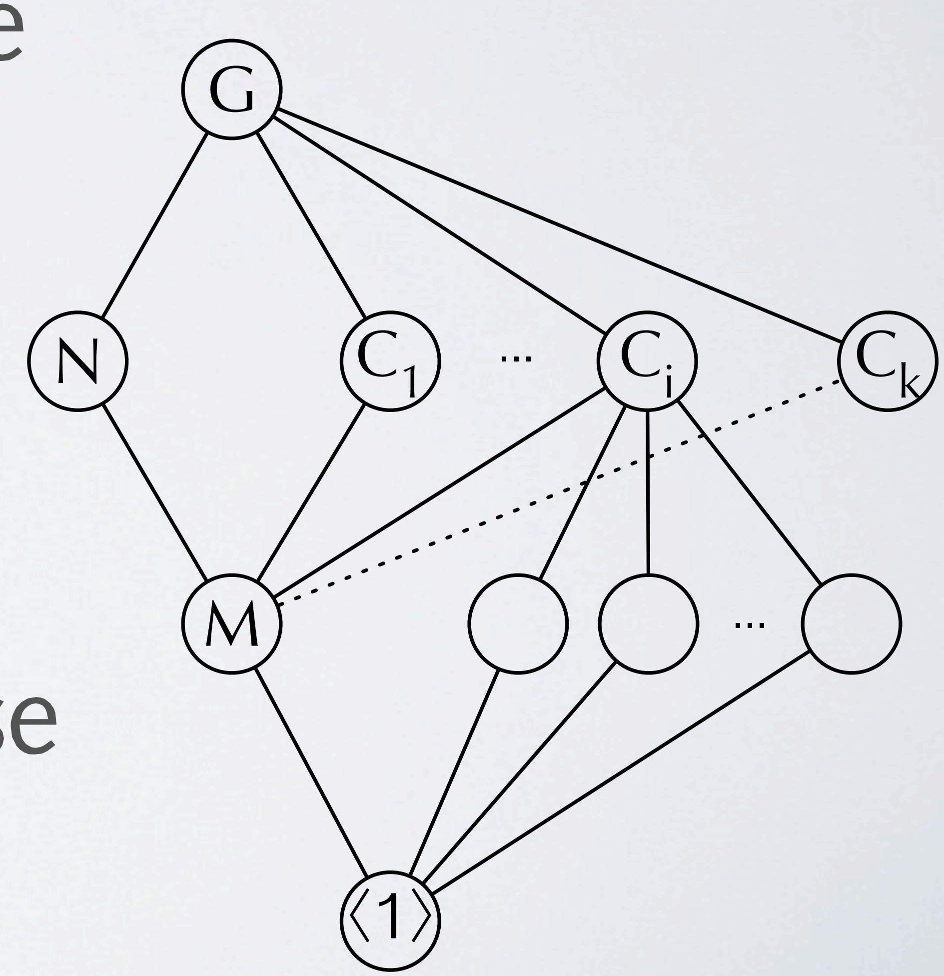
TECHNICAL DETAILS

Solvable Normal Subgroup

If N is solvable, take a G-normal series for N with elementary abelian factors.

In each step take $M \lhd G$ elementary abelian with $N \geq M$. Assume we found representatives C_i for the complements to N/M in G/M.

Then for each C_i find representatives for the complements to M in C_i and fuse these under action of $N_G(C_i)$.



TECHNICAL DETAILS

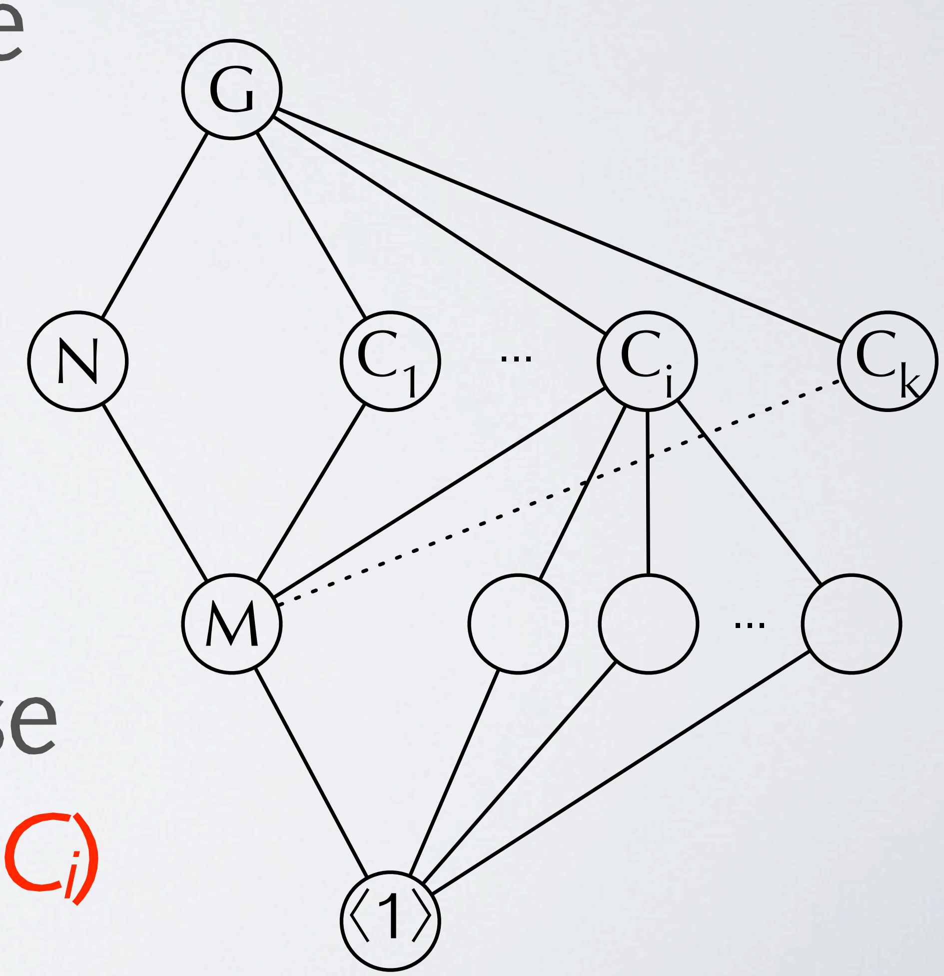
Solvable Normal Subgroup

If N is solvable, take a G-normal series for N with elementary abelian factors.

As $G/N = C$, elements of G induce no outer automorphisms of C_i

In each step take $M \lhd G$ elementary abelian with $N \geq M$. Assume we found representatives C_i for the complements to N/M in G/M.

Then for each C_i find representatives for the complements to M in C_i and fuse these under action of ~~$N_G(C_i)$~~. $C_G(C_i)$

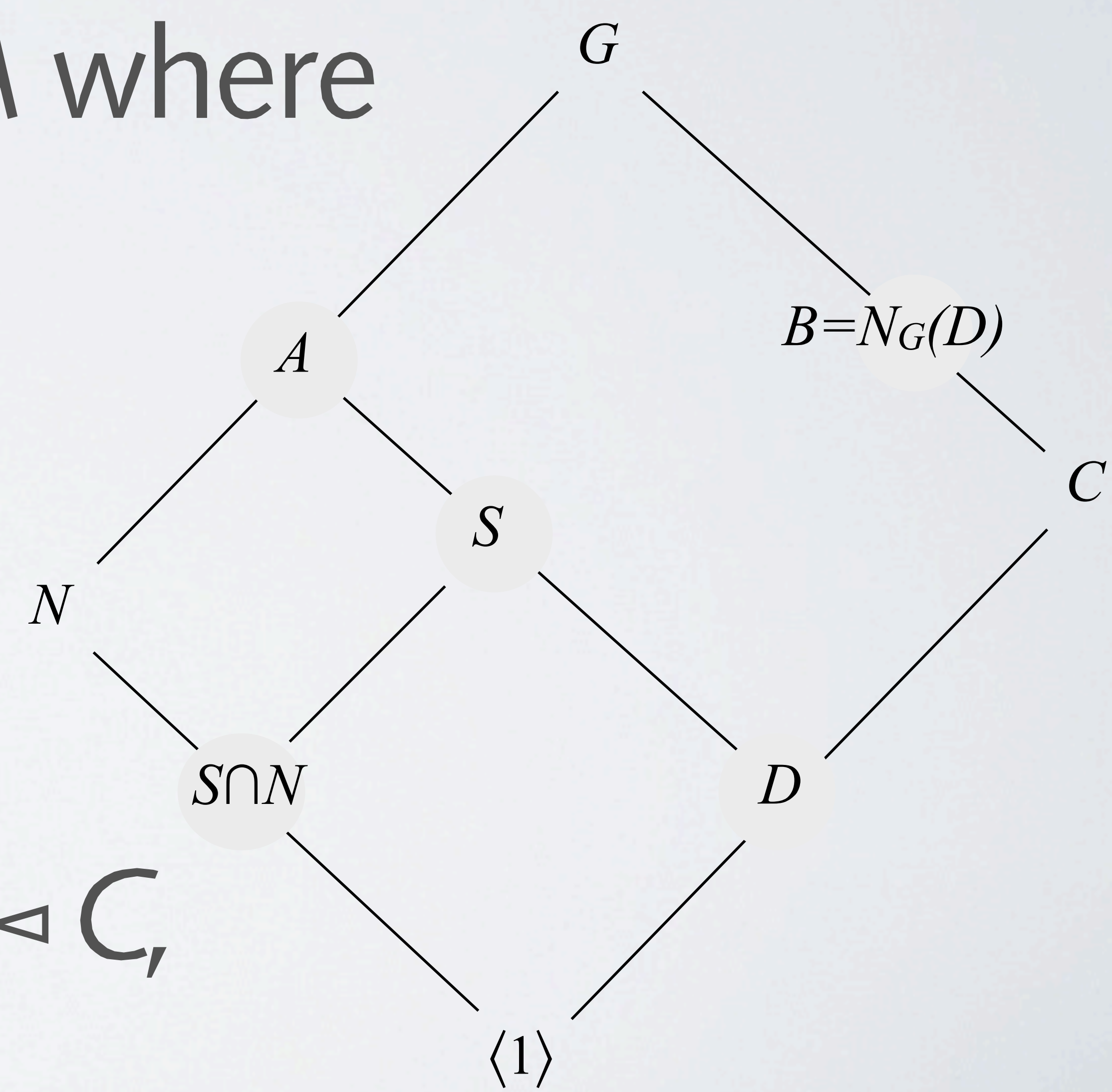


TECHNICAL DETAILS

Solvable factor

If G/N is solvable, there is $A \lhd G$ such that $A/N \lhd G/N$ is a normal p-subgroup.

Any complement C in G contains a complement $D \leq C$ in A where D is a p-subgroup.



Up to conjugacy $D \leq S$, where $S \leq A$ is a fixed p-Sylow subgroup.

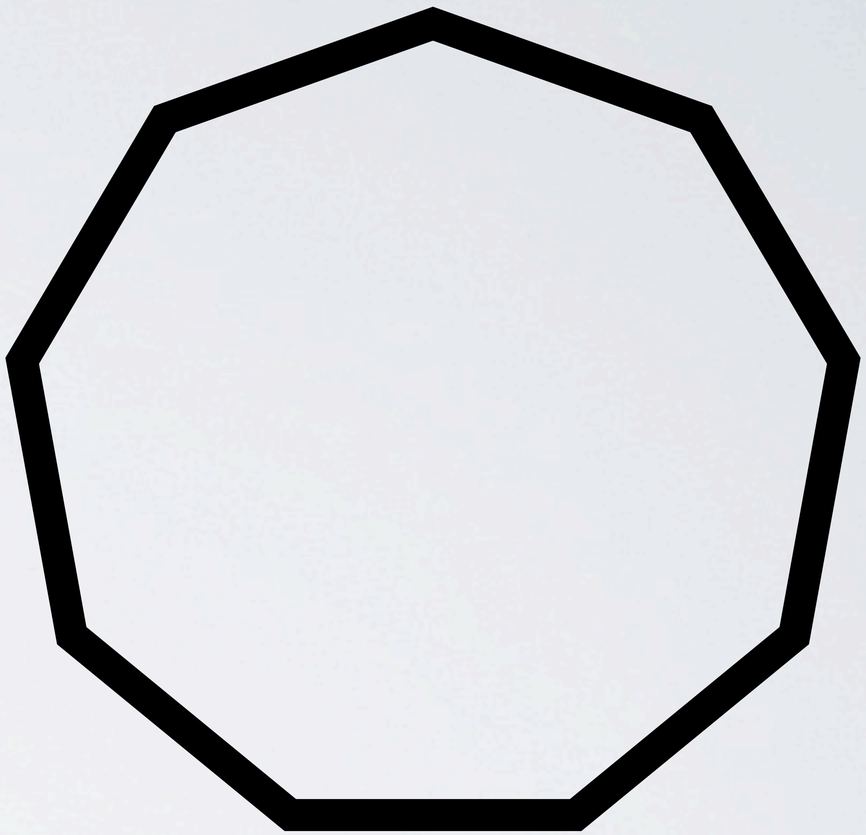
As $A \lhd G$ we have that $D \lhd C$, and thus $C \leq N_G(D) =: B$.

In Between

For a given $U<G$ the subgroups $U<V<G$ correspond to block systems for the action of G on cosets of U.

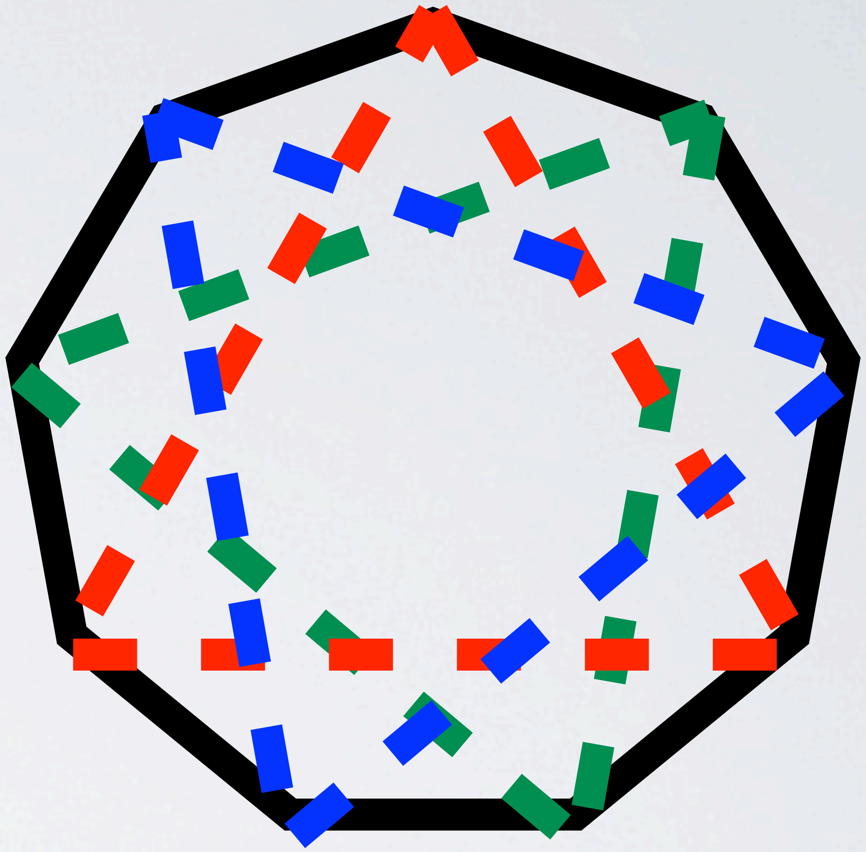
In Between

For a given $U<G$ the subgroups $U<V<G$ correspond to block systems for the action of G on cosets of U.



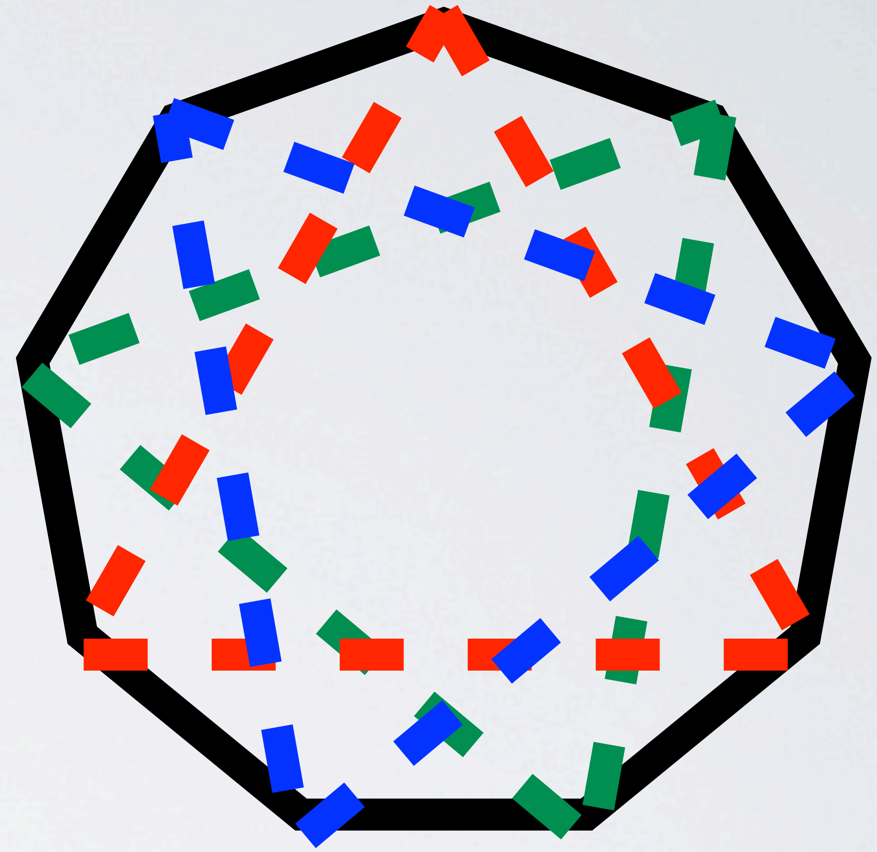
In Between

For a given $U<G$ the subgroups $U<V<G$ correspond to block systems for the action of G on cosets of U.



In Between

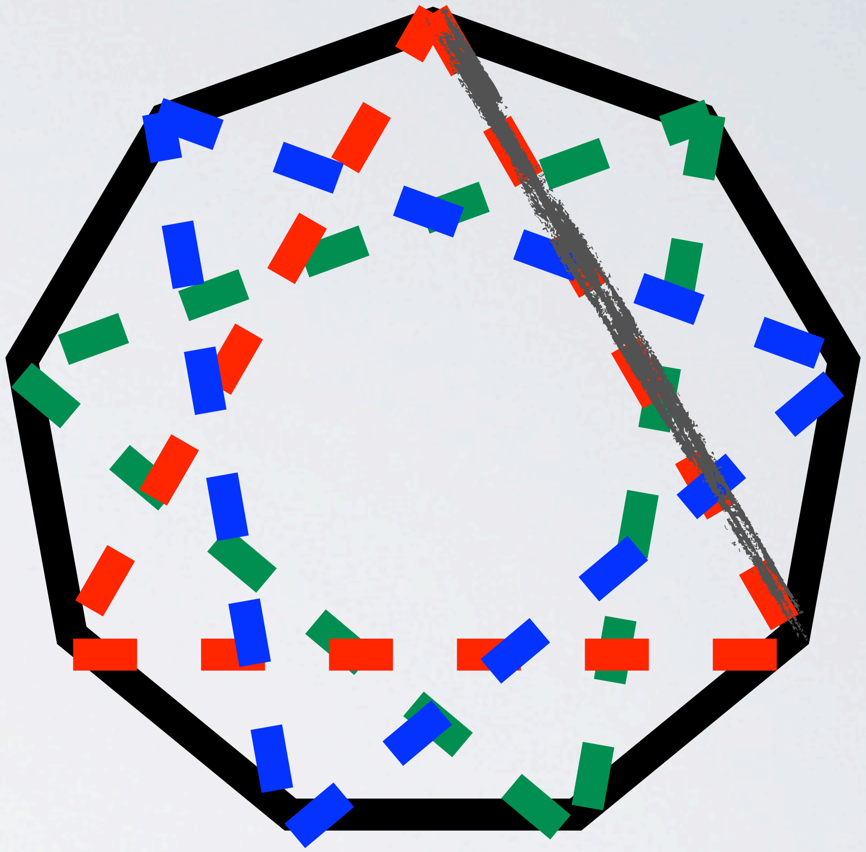
For a given $U<G$ the subgroups $U<V<G$ correspond to block systems for the action of G on cosets of U.



Finding blocks is easy: Take a potential block seed containing two points (orbit of point stabilizer). Form Images, combine to partition (fusing cells if neccessary).

In Between

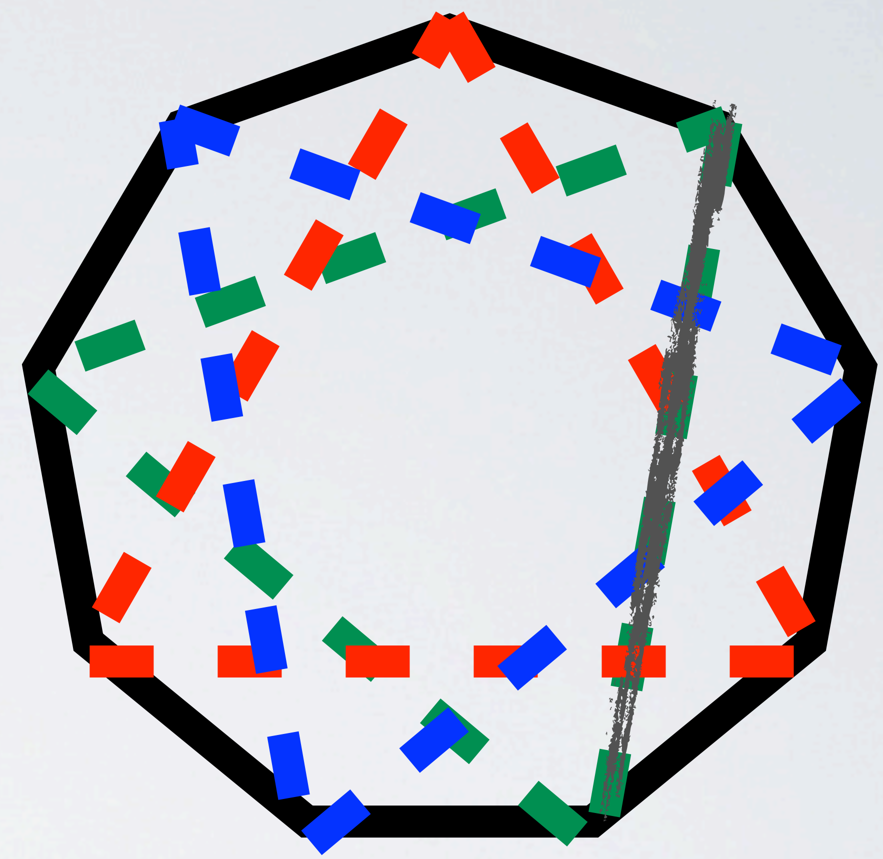
For a given $U<G$ the subgroups $U<V<G$ correspond to block systems for the action of G on cosets of U.



Finding blocks is easy: Take a potential block seed containing two points (orbit of point stabilizer). Form Images, combine to partition (fusing cells if neccessary).

In Between

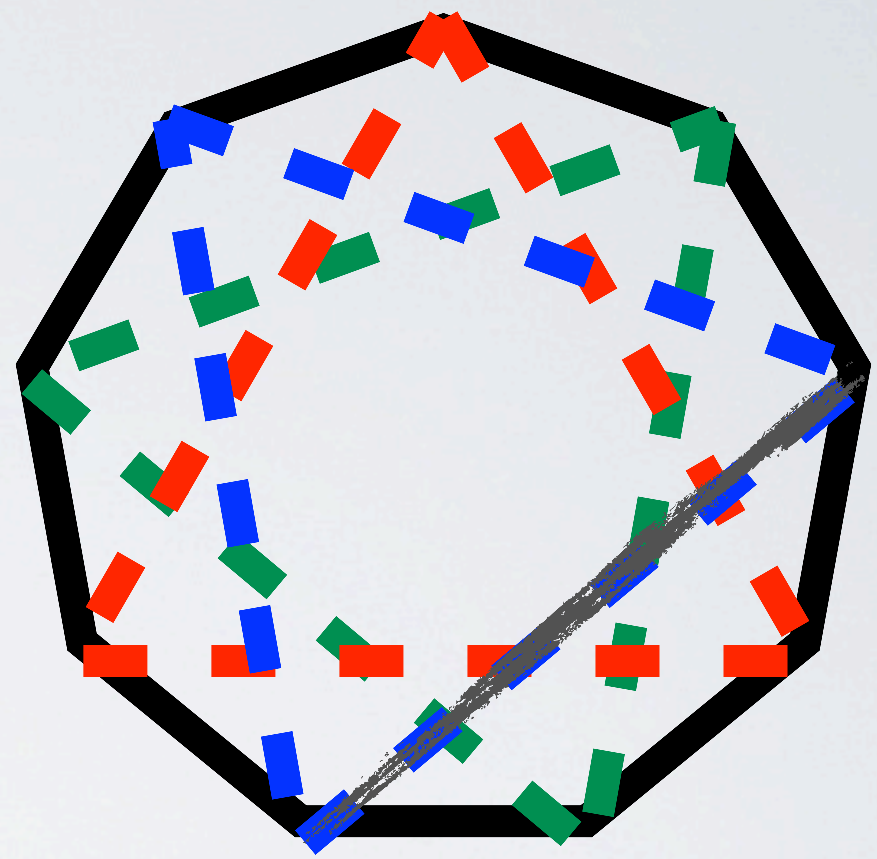
For a given $U<G$ the subgroups $U<V<G$ correspond to block systems for the action of G on cosets of U.



Finding blocks is easy: Take a potential block seed containing two points (orbit of point stabilizer). Form Images, combine to partition (fusing cells if neccessary).

In Between

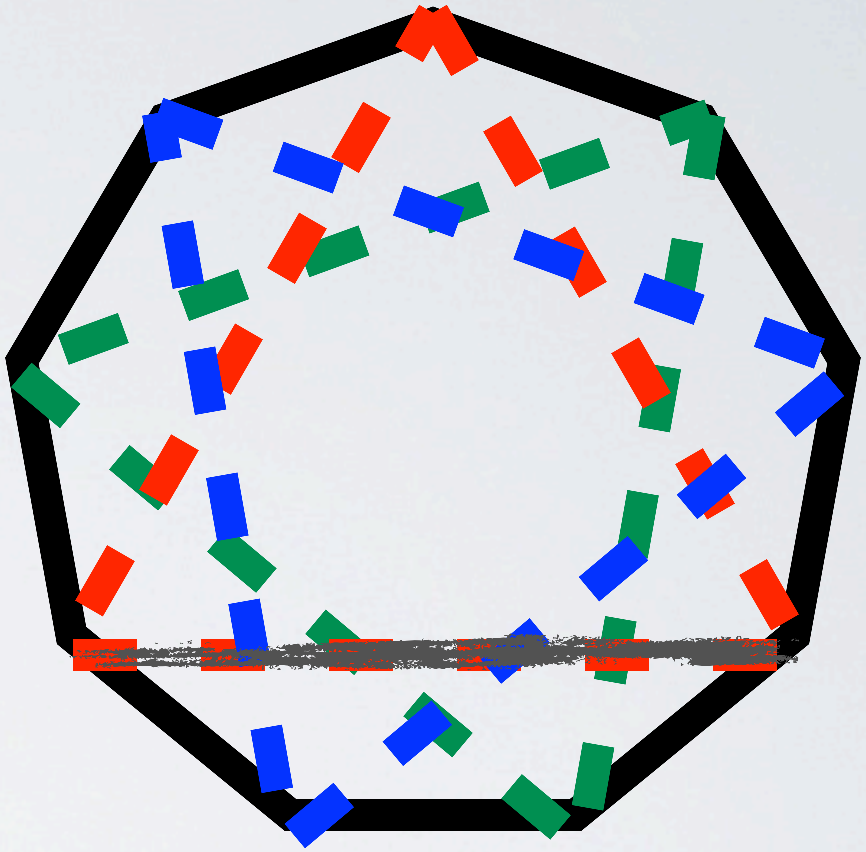
For a given $U<G$ the subgroups $U<V<G$ correspond to block systems for the action of G on cosets of U.



Finding blocks is easy: Take a potential block seed containing two points (orbit of point stabilizer). Form Images, combine to partition (fusing cells if neccessary).

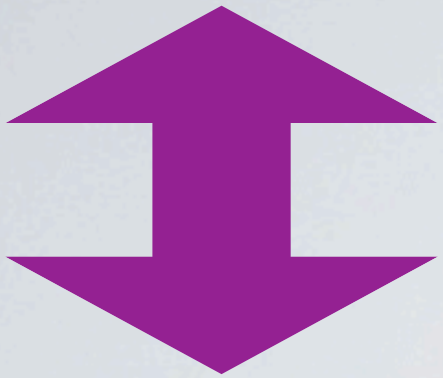
In Between

For a given $U<G$ the subgroups $U<V<G$ correspond to block systems for the action of G on cosets of U.

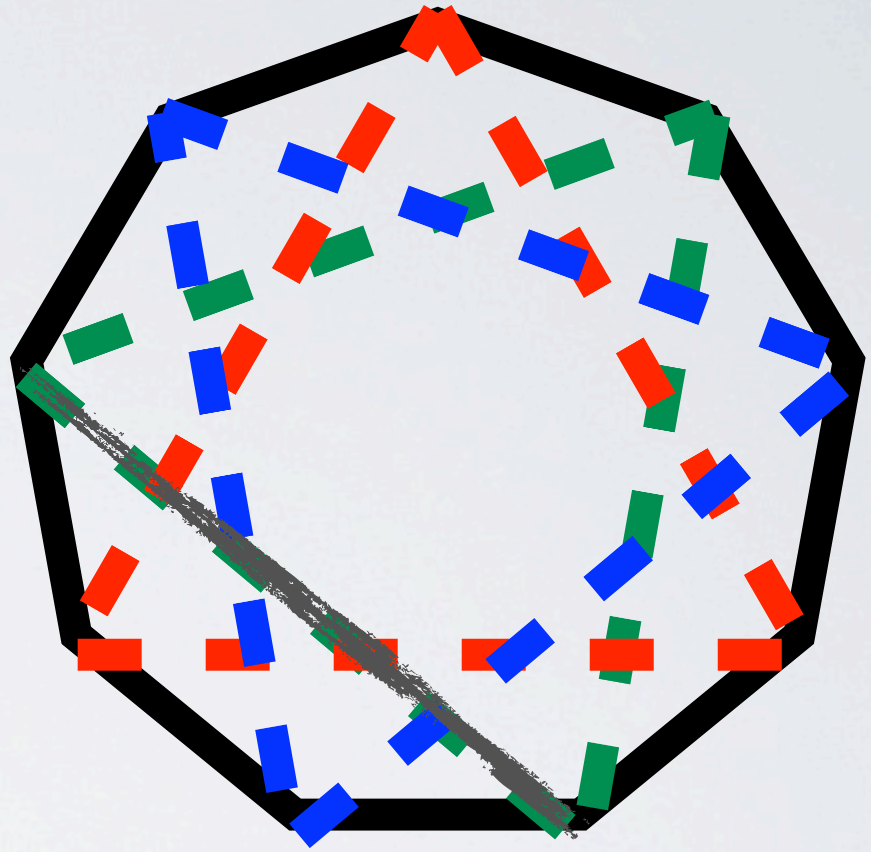


Finding blocks is easy: Take a potential block seed containing two points (orbit of point stabilizer). Form Images, combine to partition (fusing cells if neccessary).

In Between



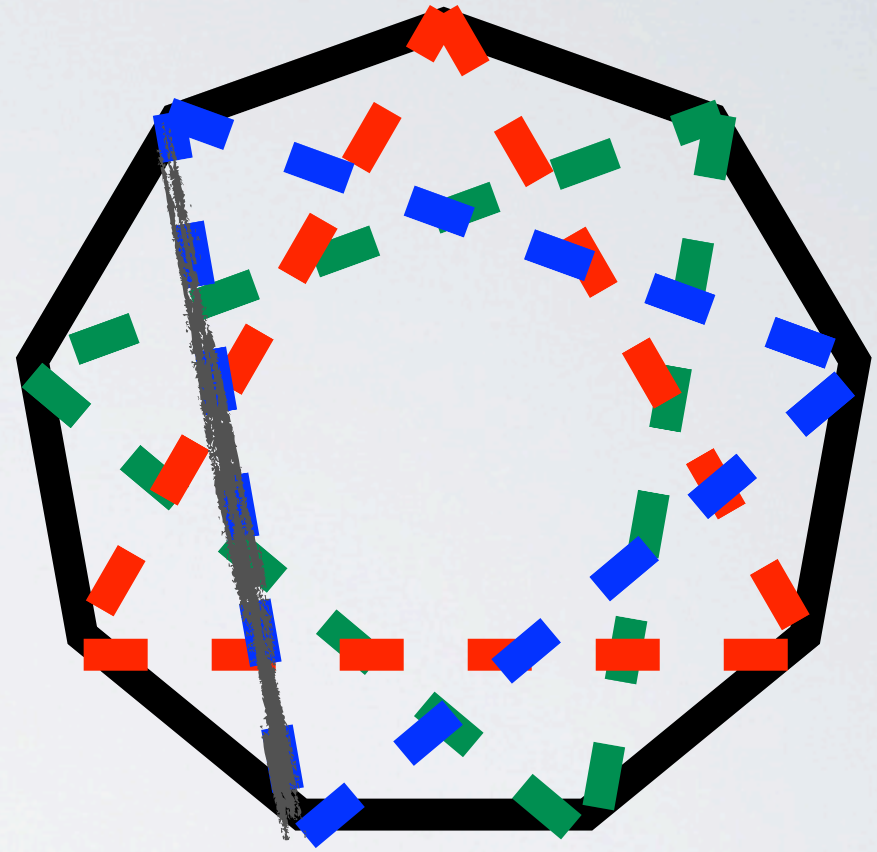
For a given $U<G$ the subgroups $U<V<G$ correspond to block systems for the action of G on cosets of U.



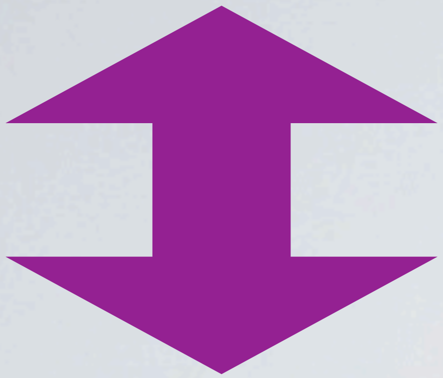
Finding blocks is easy: Take a potential block seed containing two points (orbit of point stabilizer). Form Images, combine to partition (fusing cells if neccessary).

In Between

For a given $U<G$ the subgroups $U<V<G$ correspond to block systems for the action of G on cosets of U.

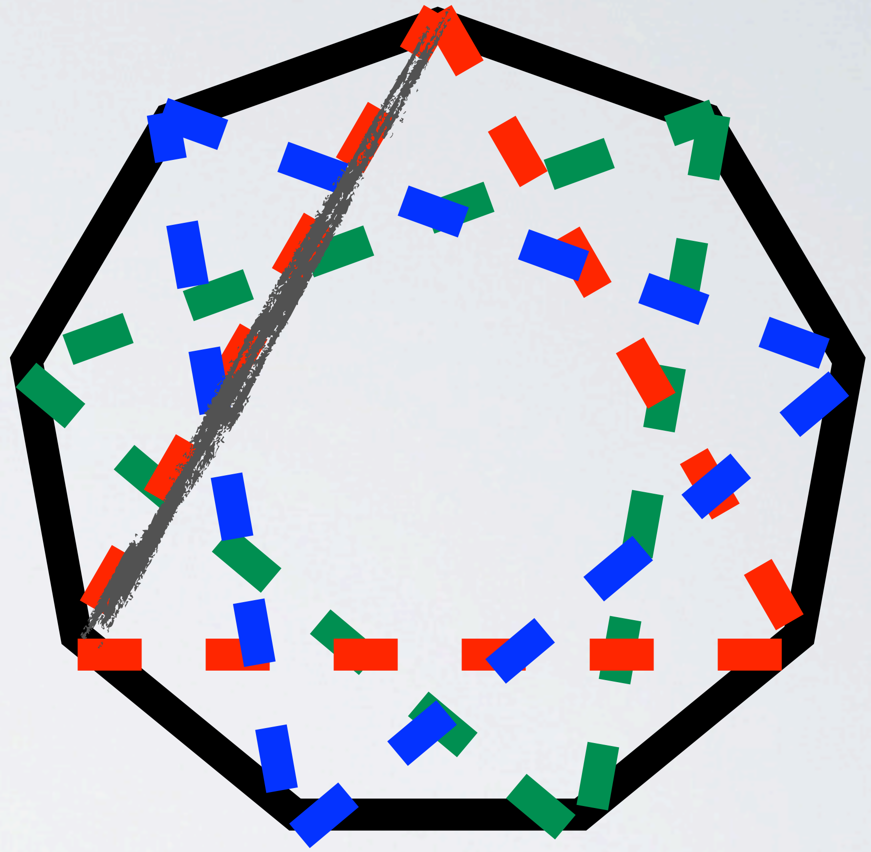


Finding blocks is easy: Take a potential block seed containing two points (orbit of point stabilizer). Form Images, combine to partition (fusing cells if neccessary).



In Between

For a given $U<G$ the subgroups $U<V<G$ correspond to block systems for the action of G on cosets of U.



Finding blocks is easy: Take a potential block seed containing two points (orbit of point stabilizer). Form Images, combine to partition (fusing cells if neccessary).

In Between

For a given $U<G$ the subgroups $U<V<G$ correspond to block systems for the action of G on cosets of U.

- Subgroups Between
- Maximal Subgroups
- Maximality Test (Future thesis topic ...)

Finding blocks is easy: Take a potential block seed containing two points (orbit of point stabilizer). Form Images, combine to partition (fusing cells if neccessary).

Subgroups Between

Take the permutation action on subgroups and construct all block systems systematically.

This yields all subgroups between two.

GAP-Function: `IntermediateSubgroups`, `AscendingChain`.

Caveat: Needs to write down permutations for action on cosets. Inefficient if there are many blocks.

Primitive Groups

If a permutation group G affords no nontrivial block systems, it is ***primitive***.

The O'NAN-SCOTT Theorem states that then $Soc(G)=T^n$, T simple and either

- $\mathrm{Soc}(G)=p^n$ and $G \leq \mathrm{AGL}_n(q)$. $\mathrm{Stab}_G(1)$ complements $\mathrm{Soc}(G)=p^n$. Or
- $Soc(G)=T^n$ nonabelian, $G \leq \mathrm{Aut}(T) \wr S_n$. $\mathrm{Stab}_G(1) \rhd \mathrm{Stab}_{Soc}(1)$ which combines diagonal/direct product of maximals.

Primitive Groups

If a permutation group G affords no nontrivial block systems, it is *primitive*.

The O'NAN-SCOTT Theorem states that then $Soc(G)=T^n$, T simple and either

- $Soc(G)=p^n$ and $G \leq AGL_n(q)$. $Stab_G(1)$ complements $Soc(G)=p^n$. Or
- $Soc(G)=T^n$ nonabelian, $G \leq Aut(T) \wr S_n$.

  Using Cohomology

  $Stab_G(1) \rhd Stab_{Soc}(1)$ which combines diagonal/direct product of maximals.

Primitive Groups

- $Soc(G)=T^n$ nonabelian, $G \leq \mathrm{Aut}(T) \wr S_n$. $\mathrm{Stab}_G(1) \triangleright \mathrm{Stab}_{Soc}(1)$ which combines diagonal/direct product of maximals.

Primitive Groups

$\boxed{\{(t,t,\ldots,t)\mid t\in T\}}$

- $Soc(G)=T^n$ nonabelian, $G\leq \mathrm{Aut}(T)\wr S_n$. $\mathrm{Stab}_G(1)\rhd \mathrm{Stab}_{Soc}(1)$ which combines diagonal/direct product of maximals.

Primitive Groups

$\boxed{\{(t,t,\dots,t)\mid t\in T\}}$

$\boxed{\begin{array}{c}\{(m_1,m_2,\dots,m_n)\mid m_i\in M\}\\ M \text{ maximal in } T.\end{array}$

- $Soc(G)=T^n$ nonabelian, $G\leq \mathrm{Aut}(T)\wr S_n$. $\mathrm{Stab}_G(1)\rhd \mathrm{Stab}_{Soc}(1)$ which combines diagonal/direct product of maximals.

Primitive Groups

$\{(t,t,\ldots,t)\mid t\in T\}$

$\{(m_1,m_2,\ldots,m_n)\mid m_i\in M\}$
M maximal in T.

$\{(m_1,m_1,\ldots,m_1,\ldots,m_k,m_k,\ldots,m_k)\mid m_i\in M\}$, $n=kl$

- $Soc(G)=T^n$ nonabelian, $G\leq \mathrm{Aut}(T)\wr S_n$. $\mathrm{Stab}_G(1)\rhd \mathrm{Stab}_{Soc}(1)$ which combines diagonal/direct product of maximals.

Find Maximal Subgroups

The possible socles arise from chief factors M/N of G.

Homomorphism $\varphi: G \to$ action on M/N.

In the Image φ construct maximal subgroups as normalizers of these diagonals and direct products.

Base case: M/N simple, maxes of simple.

For the abelian case need complements to abelian chief factors.

Find Maximal Subgroups

The possible socles arise from chief factors M/N of G.

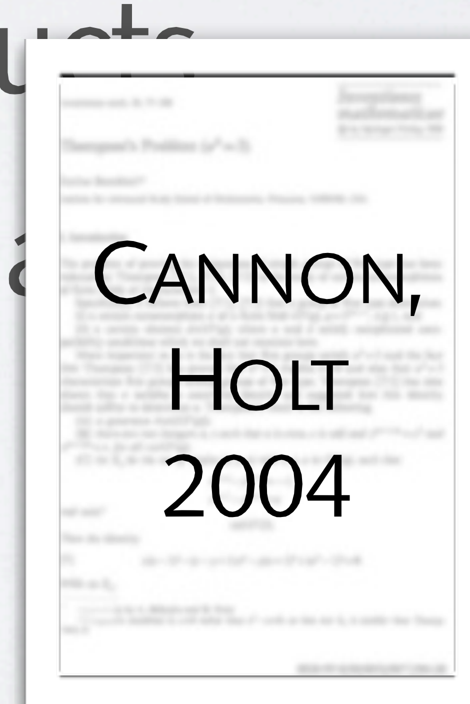
Homomorphism $\varphi:G\rightarrow$action on M/N.

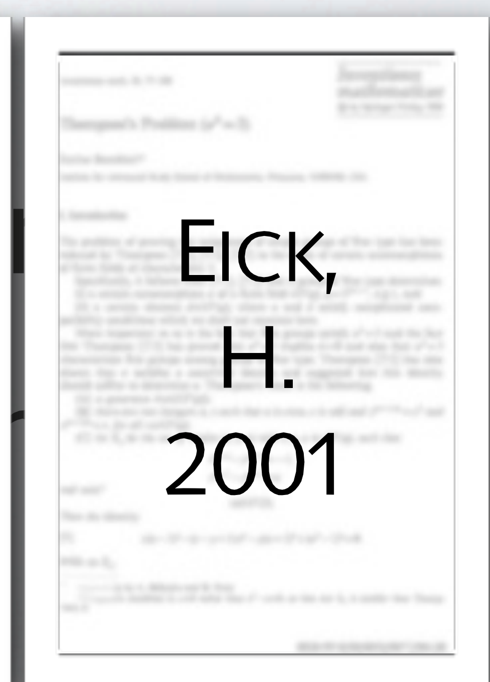
In the Image φ construct maximal subgroups as normalizers of these diagonals and direct produc

Base case: M/N simple, ma

For the abelian case need

to abelian chief factors.





Find Maximal Subgroups

The possible socles arise from chief factors M/N of G.

Homomorphism φ:$G \to$ action on M/N.

In the Image φ construct maximal subgroups as normalizers of these diagonals and direct products.

Base case: M/N simple, maxes of simple.

For the abelian case need complements to abelian chief factors.

Find Maximal Subgroups

The possible socles arise from chief factors M/N of G.

Homomorphism $\varphi: G \to$ action on M/N.

In the Image φ c…
subgroups as no…
diagonals and di…

Kleidmann, Liebeck;
Bray, Holt, Roney-Dougal

Base case: M/N simple, maxes of simple.

For the abelian case need complements to abelian chief factors.

Find Maximal Subgroups

The possible socles arise from chief factors M/N of G.

Homomorphism $\varphi: G \to$ action on M/N.

In the Image φ construct maximal subgroups as normalizers of these diagonals and direct products.

Base case: M/N simple, maxes of simple.

For the abelian case need complements to abelian chief factors.

All maximal subgroups

Remaining (awkward) case: Twisted wreath (Stabilizer in *Soc(G)* trivial). Difficult complement, only for $n>5$.

Table Lookup/Generator Search for maximals of simple.

Similar ideas for maximality test/ intermediate subgroups.

GAP-Function: (nonsolvable new, not all) `MaximalSubgroupClassReps`

Hook: Options for O'NAN-SCOTT classes.

All maximal subgroups

Remaining (awkward) case: Twisted wreath (Stabilizer in *Soc(G)* trivial). Difficult complement, only for $n>5$.

Table Lookup/Generator Search for maximals of simple.

Similar ideas for maximality test/ intermediate subgroups.

GAP-Function: (nonsolvable new, not all) `MaximalSubgroupClassReps`


Hook: Options for O'Nan-Scott classes.

Downwards

Today's standard tool is the (Solvable-Radical) Trivial-Fitting model:

- $R = Rad(G)$ is the largest solvable normal subgroup. (G/Rad is Trivial-Fitting)

- $S^*/Rad =$ Socle(G/Rad) is direct product of simple nonabelian.

- G/S^* acts on this socle as outer automorphisms and thus is small (often solvable).

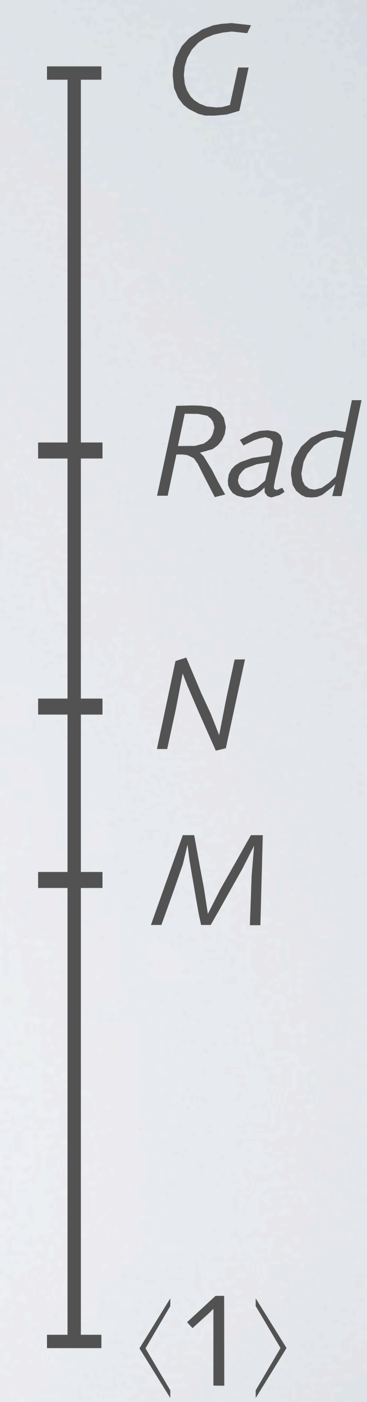
Lifting Subgroups

Construct subgroups inductively in steps over the elementary abelian layers of the radical:

$M, N \lhd G$, N/M elementary abelian.

Assume subgroups of G/N, determine those of G/M.

Induction start with G/Rad.



Lifting Subgroups

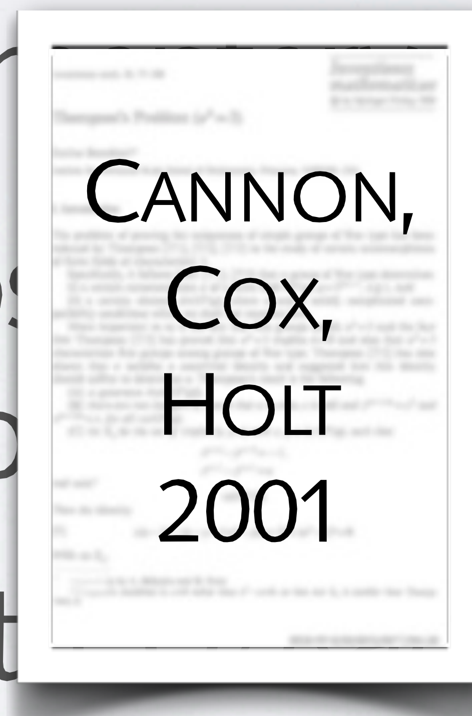
Construct subgroups inductively in steps over the elementary abelian layers of the radical:

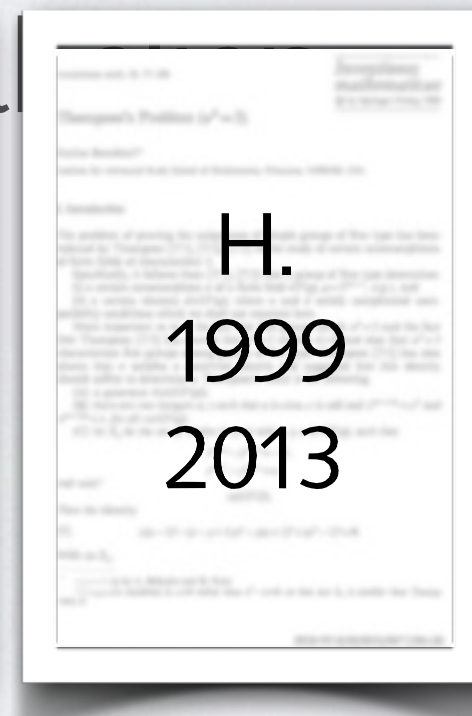
$M,N \lhd G$, N/M ele

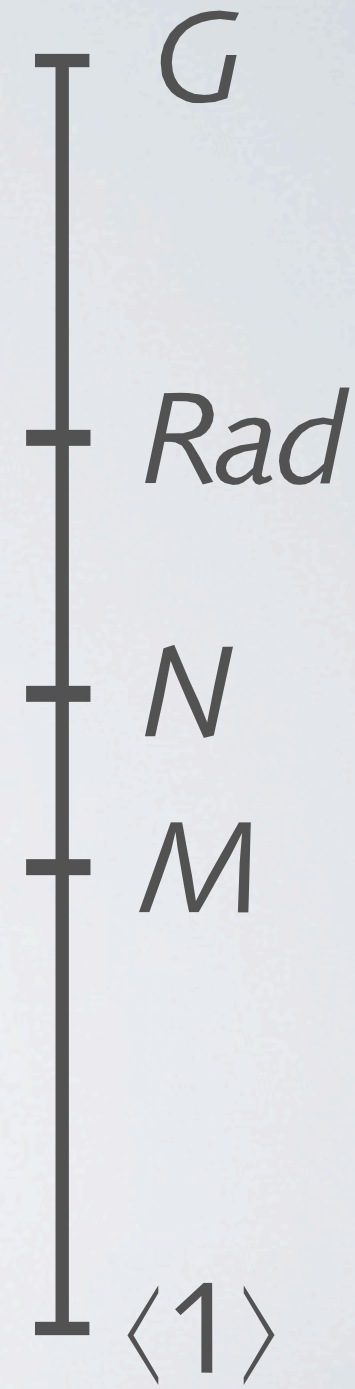
Assume subgroups

determine those o

Induction start wit







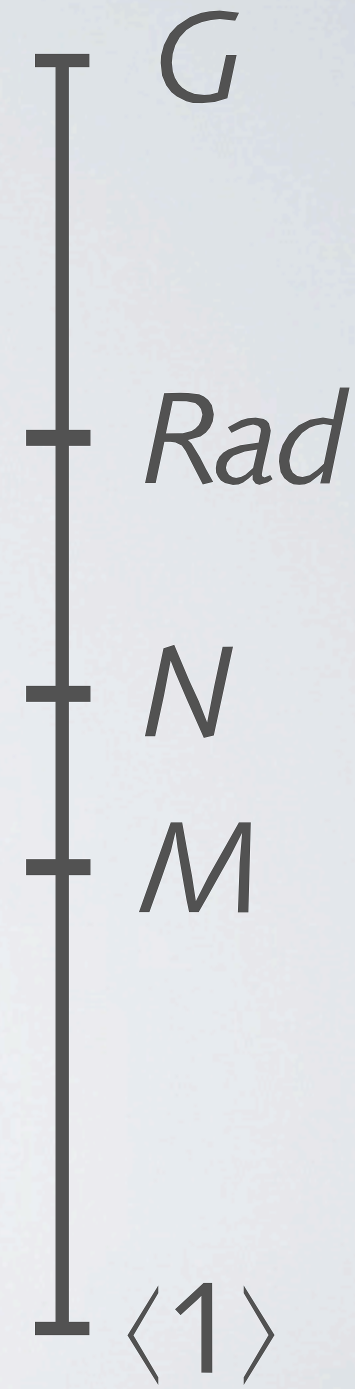
Lifting Subgroups

Construct subgroups inductively in steps over the elementary abelian layers of the radical:

$M,N \lhd G$, N/M elementary abelian.

Assume subgroups of G/N, determine those of G/M.

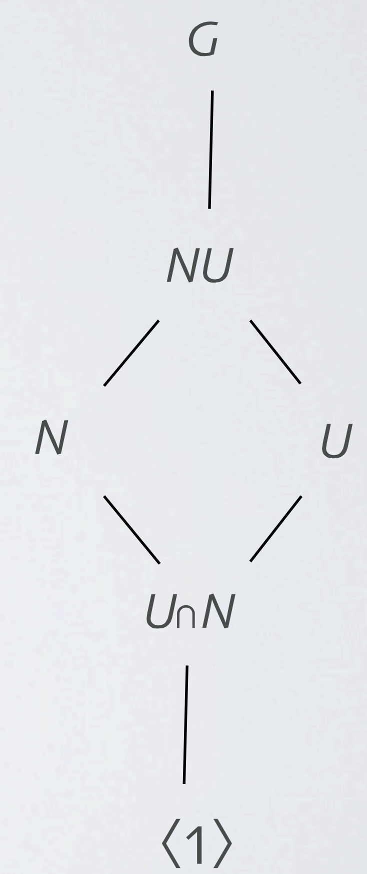
Induction start with G/Rad.



Inductive Step

WLOG (description in factor group, work with pre-images and representatives) $N \lhd G$ elementary abelian.

If $U \leq G$, then $U/U \cap N$ is complement to $N/U \cap N$ in $UN/U \cap N$.



As N is solvable this can be done using cohomology.

Further fusion under the normalizer of NU - action on cohomology.

Solvable/ Nonsolvable Case

If G is solvable this is all what is needed.

GAP-Function: `SubgroupsSolvableGroup`

Hook: Functions to limit cases in which complements are searched for.

In the general case one needs initially subgroups of G/Rad.

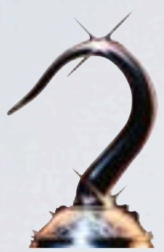
GAP-Function: `LatticeViaRadical`

Hook: Only subgroups within $Rad < H \lhd G$. (No filter function, as hard to specify.)

Solvable/ Nonsolvable Case

If G is solvable this is all what is needed.

GAP-Function: `SubgroupsSolvableGroup`



Hook: Functions to limit cases in which complements are searched for.

In the general case one needs initially subgroups of G/Rad.

GAP-Function: `LatticeViaRadical`

Hook: Only subgroups within $Rad<H\lhd G$.
(No filter function, as hard to specify.)

Solvable/ Nonsolvable Case

If G is solvable this is all what is needed.

GAP-Function: `SubgroupsSolvableGroup`

Hook: Functions to limit cases in which complements are searched for.

In the general case one needs initially subgroups of G/Rad.

GAP-Function: `LatticeViaRadical`

Hook: Only subgroups within $Rad < H \lhd G$. (No filter function, as hard to specify.)

Solvable/ Nonsolvable Case

If G is solvable this is all what is needed.

GAP-Function: `SubgroupsSolvableGroup`

Hook: Functions to limit cases in which complements are searched for.

In the general case one needs initially subgroups of G/Rad.

GAP-Function: `LatticeViaRadical`

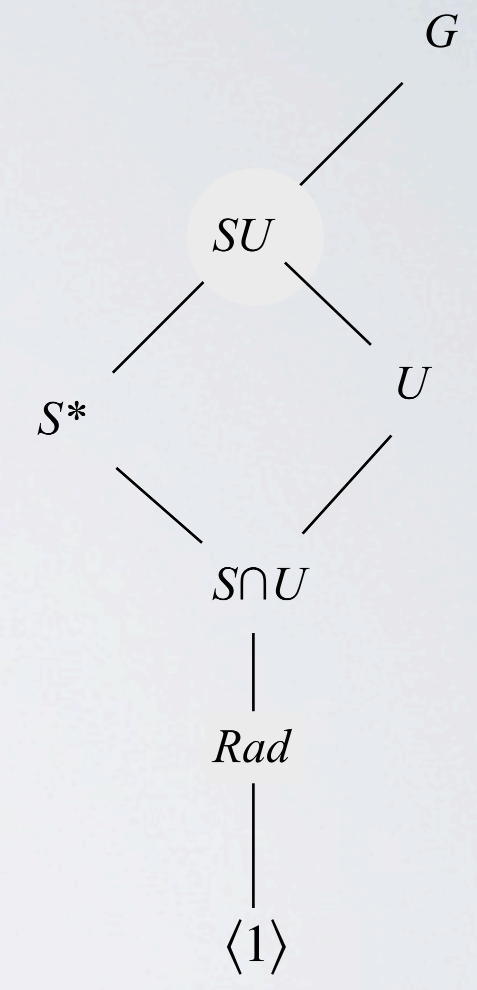
Hook: Only subgroups within $Rad < H \triangleleft G$.
(No filter function, as hard to specify.)

Radical Factor

For the initial step, *G/Rad*, use the structure with large normal subgroup:

S^*/Rad is direct product of simples. Combine subgroups from simple factors.

Any other subgroup U then intersects S^* in one of its subgroups, again complement.



Subgroups of Direct Product

S^*/Rad=Socle(G/Rad)$= \times T_i$ is a direct product of simple nonabelian groups.

- Any subgroup $U \leq S^*/Rad$ projects on each component T_i as a subgroup $V_i \leq T_i$.

- Conjugate V_i belong to conjugate U .

- For each factor T_i find list of subgroup classes with cyclic extension/tables.

- Possible combinations are Subdirect Products.

Subgroups of Direct Product

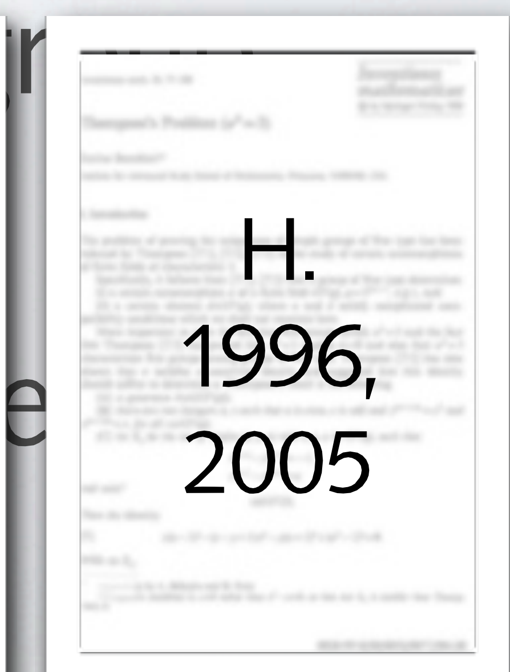
S^*/Rad=Socle(G/Rad)$= \times T_i$ is a direct product of simple nonabelian groups.

- Any subgroup $U \leq S^*/Rad$ projects on each component T_i as a subgroup $V_i \leq T_i$.

- Conjugate V_i belong to conjugate U .

- For each factor T_i find lis
classes with cyclic extensio

- Possible combinations ar
Products.



Subgroups of Direct Product

S^*/Rad=Socle(G/Rad)$= \times T_i$ is a direct product of simple nonabelian groups.

- Any subgroup $U \leq S^*/Rad$ projects on each component T_i as a subgroup $V_i \leq T_i$.

- Conjugate V_i belong to conjugate U .

- For each factor T_i find list of subgroup classes with cyclic extension/tables.

- Possible combinations are Subdirect Products.

TECHNICAL DETAILS

Subdirect Products$_U$

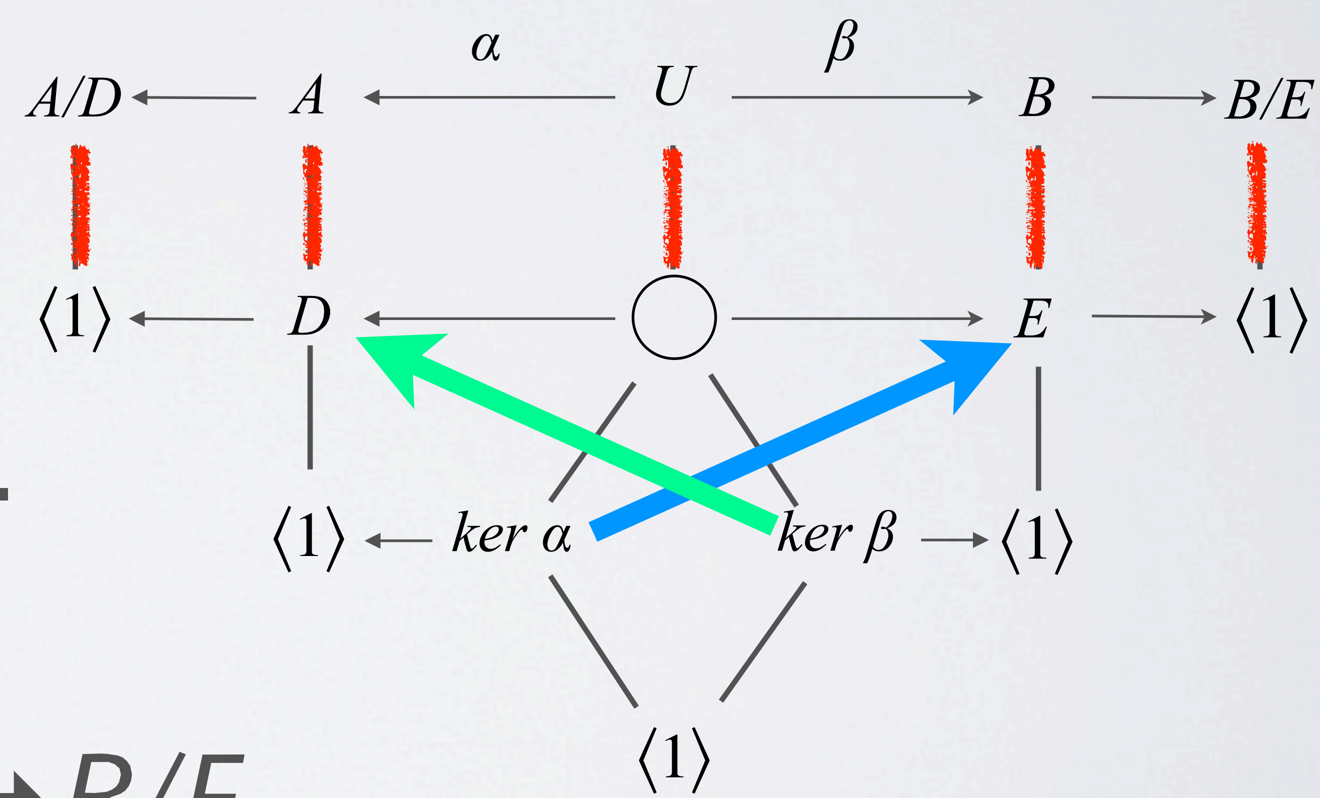
Single product. Let $U \le G \times H$ with $A \le G$, $B \le H$ chosen and direct factor projections $\alpha : U \twoheadrightarrow A$, $\beta : U \twoheadrightarrow B$. Then:

$D = (\ker \beta)^{\alpha} \lhd A$ and

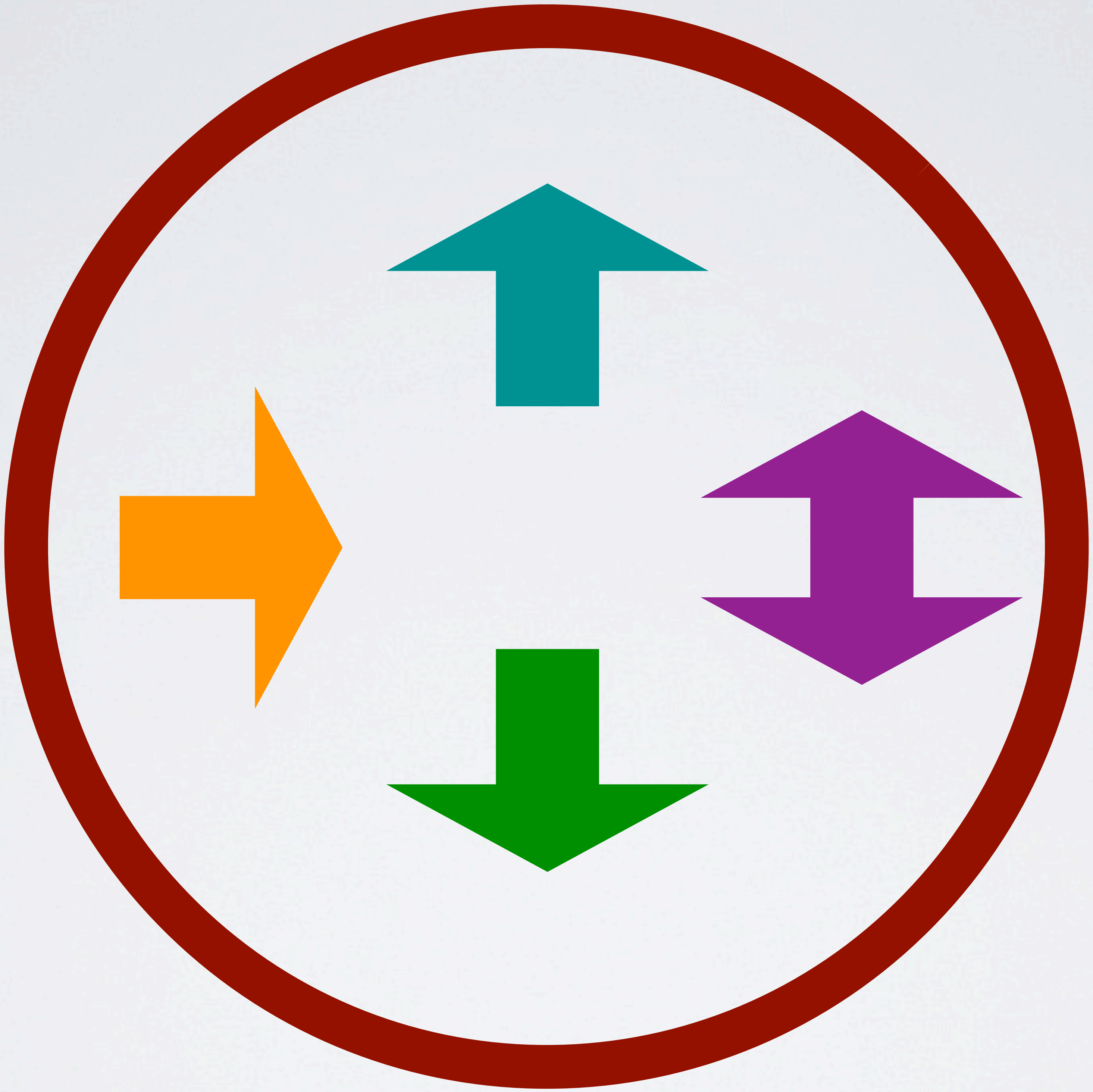
$E = (\ker \alpha)^{\beta} \lhd B$.

Factor groups A/D and B/E isomorphic.

Possible D,E and isomorphisms $A/D \to B/E$ parameterize U.



Where to go from here



How to use this

- Try to reduce the search to a subgroup (or automorphism group of the subgroup)?
- Determine how the subgroups you are searching for behave in relation to normal subgroups: Intersection, Span, Complements.

Inclusion Information

The only generic lattice structure algorithm I know determines inclusion information by testing inclusion of all conjugates in each subgroup.

Can one do better?

- Fixed Cosets: $U < V^g$ iff $VgU = Vg$

- Maximal subgroups of subgroups

- As part of construction?