LMS/EPSRC Course
Computational Group Theory
St Andrews 2013

Permutation Groups 2: Stabilizer Chains

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

Blocks (AKA imprimitivity)

G acting (transitively) on Ω, a ***block system*** is a G-invariant partition $\mathscr{B}$ of Ω, i.e. $\Omega = \cup_{B \in \mathscr{B}} B$ but $B_i \cap B_j = \emptyset$. Thus G also acts on $\mathscr{B}$.

Basic facts:

- All blocks have the same size.
- Trivial block systems: $\{\Omega\}$ and singleton sets. If only these two: ***primitive*** (otherwise ***imprimitive).***
- b blocks of size a ($n=a \cdot b$) iff $G \leq S_a \wr S_b$.
- Block systems are in bijection with subgroups $\mathrm{Stab}_G(\omega) \leq S \leq G$, S is stabilizer of block with ω.
- If $N \lhd G$ the orbits of N form a block system.

Orbit with Normal Subgroup

If we know $N \lhd G$, we can reduce:

- Determine the orbit Δ of ω under N.
- Determine the orbit of the set Δ under G.

  The image of a single point determines whether an image is new, if so whole block image is new.

  Cost is that of two smaller orbit algorithms: $|\Delta|+|\Delta^G|$ instead of $|\Omega|=|\Delta|\cdot|\Delta^G|$.
- For Stabilizer, take $\mathrm{Stab}_N(\omega)$ and correct $g \in \mathrm{Stab}_G(\Delta)$ with $n \in N$: $\omega^g = \omega^n$, $\omega^{g/n} = \omega$.

More in Bettina's lecture.

Some Fundamental Tasks

For groups of permutations of degree up to a few 10^6, order easily 10^9 (so the orbit approach is infeasible), we want to solve:

ORDER: find the order of a group. (Implies element membership test.)

HOMOMORPHISM: decompose element as generator product. (Rewriting problem.)

We want to identify the group STRUCTURE, possibly find isomorphisms.

Also centralizers, normalizers if index is huge.

Use Subgroups

The principal idea now is to use subgroups/cosets to factor the problem: As $|G|=|U|\cdot[G:U]$ this logarithmizes the problem.

Suitable subgroups: Point stabilizers $U=\mathrm{Stab}_G(\omega)$, index at most $|\Omega|$.

We can iterate this process for U.

Caveat: This works for any group with a natural action (matrix, automorphism, etc.) but often the problem is that $[G:\mathrm{Stab}_G(\omega)]$ is not small.

Case in point: $\mathrm{GL}_n(q)$, orbit length q^n.

Stabilizer Chains

Let $G \leq S_\Omega$. A list of points B=($\beta_{1,\ldots,}\beta_m$), $\beta_i \in \Omega$ is called a ***base***, if the identity is the only element $g \in G$ such that $\beta_i^g = \beta_i$ for all i.

The associated ***Stabilizer Chain*** is the sequence

$$G = G^{(0)} > G^{(1)} > \ldots > G^{(m)} = \langle 1 \rangle$$

defined by $G^{(0)} := G$, $G^{(i)} := \mathrm{Stab}_{G^{(i-1)}}(\beta_i)$. (Base guarantees that $\mathrm{G}^{(m)} = \langle 1 \rangle$.)

Note that every $g \in G$ is defined uniquely by base images $\beta_1^g, \ldots, \beta_m^g$. (If g,h have same images, then g/h fixes base.)

Base Length

The base length m often is short ($m \leq \log_2(|G|)$). In practice often $m < 10$.

We say that G is ***short-base*** if $\log|G| \leq \log^c |\Omega|$

Bounds on base length have been studied in theory. If there is no short base the groups must be essentially A_n and relatives.

Same concept also possible for other kinds of groups and mixed actions, but then no good orbit length/base length estimates.

Data structure

We will store for a stabilizer chain:

- The base points ($\beta_{1,\ldots,}\beta_m$).
- Generators for all stabilizers $G^{(i)}$. (Union of all generators is ***strong generating set***, as it permits reconstruction of the $G^{(i)}$.) Data structure thus is often called <u>**B**</u>ase and <u>**S**</u>trong <u>**G**</u>enerating <u>**S**</u>et.
- The orbit of β_i under $G^{(i-1)}$ and an associated transversal for $G^{(i)}$ in $G^{(i-1)}$ (possibly as *Schreier tree*).

Storage cost thus is $\mathcal{O}(m \cdot|\Omega|)$

Consequences

- Group order: $G = [G^{(0)}:G^{(1)}]\cdot[G^{(1)}:G^{(1)}]\cdot\ldots$ $[G^{(m-1)}:G^{(m)}]$ and thus $G=\prod_i |\beta_i^{G(i-1)}|$.

- Membership test in G for $x \in S_\Omega$:

1. Is $\omega = \beta_1^x \in \beta_1^G$? If not, terminate.
2. If so, find transversal element $t \in G^{(0)}$ such that $\beta_1^t=\beta_1^x$.
3. Recursively test membership of x/t (stabilizing β_1) in $G^{(1)}$.
   (Or test x/y=( ) in last step.)

More Consequences

Bijection $g \in G \Leftrightarrow$ base image $(\beta_1{}^g, \beta_2{}^g, \ldots)$.

- Enumerate G, equal distribution random elements.
- Write $g \in G$ as product in transversal elts.
- Write $g \in G$ as product in strong generators.
- Write $g \in G$ as product in generators of G. (Caveat: Long words)
- Chosen base: Find stabilizers, transporter elements, for point tuples.

Schreier-Sims algorithm

SIMS' (1970) primary idea is to use a membership test in a partial stabilizer chain to reduce on the number of Schreier generators.

Basic structure is a partial stabilizer, i.e. a subgroup $U \leq G^{(i-1)}$ given by generators and a base-point orbit β^U with transversal elements (products of the generators of U).

The basic operation now is to pass an element $x \in G^{(i-1)}$ to this structure and to consider the base point image $\omega = \beta^x$.

Base point image $\omega = \beta^x$

- If $\omega \in \beta^U$, transversal element $t \in U$ such that $\beta^t = \omega$. Pass $y = x/t$ to the next lower partial stabilizer $\leq G^{(i)}$.
- If $\omega \notin \beta^U$, add x to the generating set for U and extend the orbit of β. All new Schreier generators $y \in \mathrm{Stab}_U(\beta)$ are passed to next partial stab. $\leq G^{(i)}$.
- If no lower stabilizer was known, test whether the y was the identity. If so just return. (Successful membership test.)
- Otherwise start new stabilizer for generator y and the next base point. (Pick a point moved by y).

Homomorphisms

Embed permutation group G into direct product $D=G\times H$. A homomorphism $\varphi:G\rightarrow H$ can be represented as $U\leq D$ via

$$U=\{(g,h) \in G\times H \mid g^{\varphi}=h\}$$

Build a stabilizer chain for U using only the G-part.

Then decomposing $g \in G$ using this chain produces an H-part that is $g^{(\varphi-1)}$.
Use this to evaluate arbitrary homomorphisms.

Kernels, Relators

Vice versa, let $\varphi{:}H\rightarrow G$ and $U=\{(g,h)\in G\times H \mid h^{\varphi}=g\}$.

Form a stabilizer chain from generators of U, using the G-part.
The elements sifting through this chain (trivial g-part) are generators for ker φ.

If H is a free group, this yields a presentation for G.

Quandry

Deterministic Algorithm. Polynomial (in the degree $n=|\Omega|$) runtime, but larger exponent (n^3 if Schreier tree used).

The cause is the processing of all (mostly redundant) Schreier generators.

In practice not feasible if n is big ($>$1000). For short base ($\log|G|\leq\log^c$ n) we would like nearly linear time $\mathcal{O}(n \log^c n)$, best possible

Wrong Results are Cheap

Use only *some* generators (random subset, better: random subproducts). Wrong data structure. But:

▸Error results in chain that claims to be too small - can detect if group order is known.

▸Error analysis: A random element of *G* fails sifting in wrong chain with probability 1/2 - guarantee arbitrary small error probability.

▸But we can verify that a chain is correct:

➡Combinatorial Verification (Sims, see SERESS' book)

➡Presentation from stabilizer chain. Verify that group fulfills it. If too small, some relators fail to be. (Todd-Coxeter-Schreier-Sims; or Recognition see lecture 3.)

Other Actions

Every finite group is a permutation group in suitable actions. (E.g. matrices on vectors.) Same methods apply there.

It is possible to use different actions (e.g. matrix group on subspaces and on vectors)

But: Orbit lengths can be unavoidably huge if there are no subgroups of small index.

Approach can be useful for well-behaved groups. Not a panacea, but part of matrix group recognition.

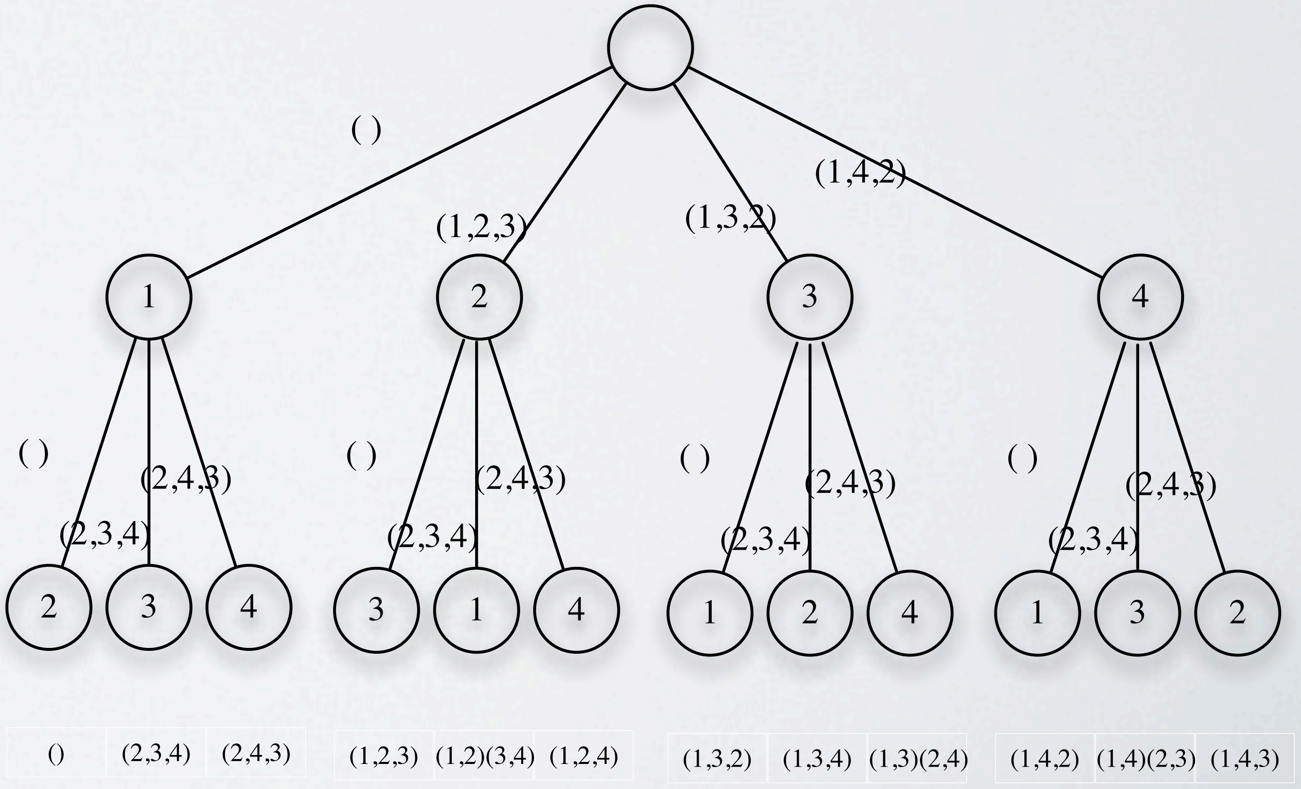
Backtrack

A stabilizer chain lets us consider the elements of *G* as leafs on a tree, branches corresponding to base point images.

Traverse the tree (depth first) by enumerating all possible base images. Find group elements with particular desired property.

Exponential run time but good in practice.

E.g.: Centralizer, Normalizer, Set Stab., Intersection, Conjugating elts., ...



Search Tree Pruning

It is crucial to reduce the search space down from $|G|$ to a manageable size. Tools:

Algebraic structure: Solution set forms a subgroup (if stabilizer) or double coset (if transporter). E.g., all elements mapping ω to δ lie in $\mathrm{Stab}_G(\omega)\cdot g\cdot \mathrm{Stab}_G(\delta)$ where $\omega^g=\delta$.

The closure properties of the structure mean that the existence of some elements implies existence of others.

For simplicity, assume that we are aiming to find $S= \mathrm{Stab}_G(\omega)$.

Double Coset Pruning

Assume we have found (or were given) some elements of S, generating subgroup U. (Hard part is to prove there are no further ones.)

If $g \in G$, then either all or no elements of UgU will be in S. Sufficient to test one.

Criterion: Only test g if it is minimal in UgU. (lexicographically as lists of base images.)

Minimal in UgU is hard. Instead use minimal in Ug and in gU (necessary, not sufficient). Restrict choice of possible base images.

Problem-specific Pruning

The real power of backtrack comes with pruning methods that are specific to the problem to be solved. For example:

- An element centralizing a permutation must map cycles to cycles of the same length. Images of the first cycle point thus are limited. Once the image ω^g of a first cycle point is chosen, the images of **all** other points in the cycle are given.

- An element normalizing a subgroup *U* must preserve the orbits of *U.* When also fixing the point ω, one must preserve the orbits of $\mathrm{Stab}_U(\omega)$ (these are called orbitals).

Base Change

For efficiency , it is helpful to use a base that causes problem-specific prunings to apply early.

E.g. when centralizing an element, choose the first base point in a cycle of longest length (as the choice of one point image determines all others).

There used to be algorithms that performed a ***base change***, i.e. computed a new stabilizer chain from an old one but with different base.

Modern, randomized, Schreier-Sims algorithms are so fast that is is usually easiest to just compute a new chain for the desired base.

Partition backtrack

Partition Backtrack (McKay, Leon, Theissen,...) is a convenient way to process the different kinds of pruning.

The algorithm maintains a partition (list of points) of Ω, indicating possible images of the base points. Tree root=(Ω), leaves=1point cells.

Selection of base images and pruning conditions are partition refinements, done by intersecting with particular partitions, such as (img, rest) or orbits of a subgroup.

TO BE
CONTINUED....