Permutation Groups in NC

László Babai
Eugene M. Luks
Ákos Seress

CIS-TR-87-02
March 2, 1987

Department of Computer and Information Science
University of Oregon

To appear in Proc. 19th ACM STOC, 1987

Permutation Groups in NC

László Babai *
Eötvös University, Budapest,
and University of Chicago

Eugene M. Luks †
University of Oregon

Ákos Seress *
Mathematical Institute of the
Hungarian Academy of Sciences

Abstract

We show that the basic problems of permutation group manipulation admit efficient parallel solutions. Given a permutation group G by a list of generators, we find a set of NC-efficient strong generators in NC. Using this, we show, that the following problems are in NC: membership in G; determining the order of G; finding the center of G; finding a composition series of G along with permutation representations of each composition factor. Moreover, given G, we are able to find the pointwise stabilizer of a set in NC. One consequence is that isomorphism of graphs with bounded multiplicity of eigenvalues is in NC.

The analysis of the algorithms depends, in several ways, on consequences of the classification of finite simple groups.

1. Introduction

We resolve the central problem in parallel management of permutation groups. The key to this resolution is

Theorem 1.1. *Given a permutation group by a list of generators, one can find a set of NC-efficient strong generators (SGS) in NC.*

Using this, NC-algorithms for basic permutation group manipulation, and well beyond, are derived.

Theorem 1.2. *Given a permutation group G by a list of generators, the following problems are in NC:*
(a) *Test membership in G.*
(b) *Find order of G.*
(c) *Find the center of G.*
(d) *Find a composition series of G.*

We remark that an NC-solution of the membership problem has been suggested to be impossible, that is, the problem was conjectured to be LOGSPACE-complete for P [MC].

We also report the resolution of a problem that is inspired by graph isomorphism applications.

Theorem 1.3. *Given a permutation group G, one can find the pointwise stabilizer of a set in NC.*

Note that, in earlier work, pointwise set-stabilizers were not known even for then "manageable" groups, i.e. those for which membership-testing was available (cf. [LM], [Lu86]). In particular, this was the source of a Las Vegas - deterministic gap [Ba86] in isomorphism testing that we now close.

Theorem 1.4. *Isomorphism of graphs with bounded multiplicity of eigenvalues is in NC.*

For comparison, we briefly review earlier work. The results of Theorems 1.1–1.3 were known for solvable groups [LM] and more generally for groups with bounded nonabelian composition factors [Lu86]. A Las Vegas algorithm for a somewhat less restrictive class of groups for the pointwise set-stabilizer problem appears in [Ba86]. In the same paper, a Las Vegas version of Theorem 1.4 appears. Polynomial time sequential algorithms for each problem have been known for some time: [Si70], [FHL] for Theorems 1.1 and 1.3 as well as for Theorem 1.2 (a),(b); [Lu87] (cf. [BKL]) for Theorem 1.2(c),(d); [BGM] for Theorem 1.4. A more detailed account of the history of the problems can be found in [Lu86] which is a key reference for several parts of the new algorithm as well.

The principal novelty of this paper is our ability to handle *symmetric* and *alternating groups* (the "giants") (Section 5). Two other chief ingredients are: (1) the augmented structure forest and the descent through the associated semisimple structure (involving the "noncommutative linear algebra") [Lu86]; (2) the composition series algorithm [Lu87].

Most striking is the depth of group-theoretic machinery that is required for parallelizing even the rudimentary task of membership testing. Three levels of group

*Research partially supported by Hungarian Natl. Found. for Scient. Res. Grant 1812
†Research supported by NSF Grant DCR-8609491.

theory arise: elementary 19th century style combinatorial arguments; structure theory of primitive permutation groups; and applications of three deep results, currently derivable only by using the full force of the classification of finite simple groups.

Remarkably, the new methods have sequential implications as well; these will be exploited in another paper [BLS].

2. Definitions and preliminaries

We assume familiarity with the complexity class NC ([Pi],[Co]), informally, the class of problems solvable in polylog ($= \log^{const} n$) time using a polynomial number of processors. We refer to any standard text, e.g. [Ha], for basic facts about groups. For permutation group concepts we refer to [Wi] and [Cam]. We mention two sources of information on the classification of finite simple groups [Go], [Car], but no knowledge of these works will be required. Cameron [Cam] gives a nice survey of all the consequences of the simple groups classification relevant to our work.

2.1. Group theory

We write $H \leq G$ if H is a subgroup of G and $H \lhd G$ if H is a normal subgroup.

Lemma 2.1. [Ha, p.96] *Let $H \leq G$ and assume S is a set of generators of G and R is a (complete) set of (right) coset representatives of G mod H. Then the set*

$$\{\rho\sigma\rho_1^{-1} | \rho, \rho_1 \in R, \sigma \in S, \rho\sigma\rho_1^{-1} \in H\}$$

generates H.

The generators described here are called *Schreier generators* of H; their number is $|S||G:H|$.

For $H \leq G$ the *normal closure* $NCl_G(H)$ of H in G is the smallest normal subgroup of G containing H. A group $G \neq 1$ is called *simple* if it has no nontrivial normal subgroups. G is *semisimple* if it is the direct product of simple groups. A *composition series* of G is any series

$$1 = G_r \lhd \cdots \lhd G_1 \lhd G_0 = G$$

where the quotients G_{i-1}/G_i are simple; these quotients are the *composition factors*. One calls G *solvable* if all composition factors of G are cyclic. The following is folklore.

Proposition 2.2. *Let $1 = G_r \lhd \ldots \lhd G_1 \lhd G_0 = G$ be a composition series of G and $N_i = NCl_G(G_i)$. Then each quotient N_{i-1}/N_i is semisimple; in fact, it is isomorphic to the direct product of copies of G_{i-1}/G_i.* ♠

The *socle* of G is the subgroup generated by all minimal normal subgroups and is denoted $Soc(G)$. The socle is semisimple.

The *automorphism group* of G is denoted $Aut(G)$. Every element $g \in G$ induces an *inner automorphism* $x \mapsto g^{-1}xg$. The group of inner automorphisms, $Inn(G)$, is normal in $Aut(G)$. The factor group $Out(G) = Aut(G)/Inn(G)$ is the *outer automorphism group*. The following result is needed for some of our analyses; it is now a consequence of the classification of finite simple groups.

Theorem 2.3. (Schreier's Hypothesis). *The outer automorphism group of every finite simple group is solvable.*

2.2. Permutation groups

The group of all permutations of an n-element set A is denoted $Sym(A)$, or S_n if the specific set is inessential. Subgroups of S_n are the *permutation groups* of *degree* n. The *even* permutations of A form the *alternating group* $Alt(A)$ (or A_n). We shall refer to $Sym(A)$ and $Alt(A)$ as the *giants*. These two families of groups require special treatment in most algorithms (see Section 5).

The *support*, $supp(\pi)$, of $\pi \in Sym(A)$ consists of those elements of A actually displaced by π. The *degree* of π is $deg(\pi) = |supp(\pi)|$.

We say that G *acts on* A if a homomorphism $G \rightarrow Sym(A)$ is given. This action is *faithful* if its kernel is the identity. The *orbit* of $a \in A$ under G is the set of images $\{a^\gamma | \gamma \in G\}$. G is *transitive* on A if there is only one orbit. G is *t-transitive* if the action of G induced on the set of ordered t-tuples of distinct elements of A is transitive ($t \leq n$). The maximum such t is the *degree of transitivity* of G. The degree of transitivity of the giants is $\geq n-2$.

Theorem 2.4. *The degree of transitivity of any permutation group other than the giants is ≤ 5.*

This is another consequence of the classification of finite simple groups we require and is essentially due to Curtis, Kantor, and Seitz [CKS] (cf. [Cam]). A combination of Theorem 2.4 with elementary tricks for permutations facilitated the breakthrough in handling the giants that has led, in combination with [Lu86] and [Lu87], to the main results of this paper as well as to the start of the largely independent project of [BLS].

2.3. Orbits, orbitals, blocks

If G acts on A, the orbits of the induced (componentwise) G-action on $A \times A$ are called *orbitals* [Si76]. The *stabilizer* of $x \in A$ is the subgroup $G_x = \{\gamma \in G | x^\gamma = x\}$. If G is transitive, there is a bijection between the orbitals of G and the orbits of G_x. For an orbital Θ of G and $x \in A$,

the (out)neighbors of x in the (di)graph (A, Θ) form the orbit $\Theta(x) = \{y|(x,y) \in \Theta\}$ of the stabilizer G_x.

If G is transitive on A and $G_x = 1$ for some (thus, every) $x \in A$, then G is *regular*. If G is transitive and $D \subseteq A$, D is called a *block* (for G) if for all $\gamma \in G$, either $D^\gamma = D$ or $D^\gamma \cap D = \emptyset$, and G is called *primitive* if no nontrivial blocks exist. (Trivial blocks have 0, 1 or n elements.) If D is a block then the set of images of D is called a *block system* and an action of G is induced on the block system. The block system is *minimal*, if that action is primitive. We shall need the following elementary results on the structure of primitive groups. They all follow from the O'Nan-Scott Lemma [Sc] (cf. [Cam], [Lu82]).

Theorem 2.5. *Let $G \leq Sym(A)$ be primitive and suppose $Soc(G)$ is abelian. Then, $n = p^d$ for some prime p, A can be identified with the d-space over $GF(p)$ so that $G \leq AGL(d,p)$ (the group of affine transformations of A), and $Soc(G) \cong \mathbf{Z}_p^d$ is the group of translations of A.*

Theorem 2.6. *Let $G \leq Sym(A)$ be primitive. Then*

$$Soc(G) = T_1 \times ... \times T_d$$

where the T_i are isomorphic simple groups. If $Soc(G)$ is nonabelian then G contains a normal subgroup N such that
(a) *$Soc(G) \leq N \leq Aut(T_1) \times ... \times Aut(T_d)$;*
(b) *G/N is a subgroup of S_d;*
(c) *$n \geq 5^d$.*

Theorem 2.7. *Let $G \leq Sym(A)$ be primitive. If G has more than one minimal normal subgroup then G has precisely two minimal normal subgroups, each of order $|A|$.*

2.4. Groups of Cameron type

Important examples of primitive groups whose socles are products of alternating groups are obtained in the following way.

First we define a class of imprimitive groups. Let B be a set of k elements, and suppose $1 \leq s \leq k/2$. Let $C = rB = B_1 \dot\cup \ldots \dot\cup B_r$ denote the disjoint union of r copies of B. An s-transversal of C is a subset $X \leq C$ such that $|X \cap B_i| = s$ for $i = 1, ..., r$. Let A denote the set of s-transversals; and $n = |A| = \binom{k}{s}^r$. The *wreath product* $W(B,r) = Sym(B) \wr S_r \leq Sym(C)$ consists of all permutations of C that respect the partition $\{B_i\}$. Clearly,

$$Soc(W(B,r)) = Alt(B_1) \times ... \times Alt(B_r).$$

Let now $W(B,r) \geq G \geq Soc(W(B,r))$ and assume G acts transitively on the set of blocks $\{B_i\}$. Under these conditions, the action of G on A is *primitive* (and alternating type, since $Soc(G) = Soc(W(B,r))$.) We say that the primitive groups so obtained are of *Cameron type*.

Theorem 2.8. [Cam] *There exists a constant c such that every primitive group of degree n and order $> n^{c \log n}$ is of Cameron type.*

This is the third consequence of the simple groups classification (proved via a result of Kantor [Ka1]) required for the analysis of our algorithms. For large n, c approaches 1. We remark that the value of c does not play a role in the algorithms; its existence enters only in the analysis.

2.5. Cameron schemes

In Section 4 we shall analyze a combinatorial structure associated with the action of $W(B,r)$ on A. Let A, B, C be as above. For an s-transversal $X \in A$, let $X_i = X \cap B_i$. For $X, Y \in A$, let $d_i = |X_i \cap Y_i|$ and let $f_1 \leq f_2 \leq ... \leq f_r$ be the sorted sequence $\{d_i\}$. We call $(f_1, ..., f_r)$ the *intersection pattern* of X and Y. Let us partition $A \times A$ according to intersection patterns: $A \times A = R_0 \cup ... \cup R_N$. We call the system $\mathcal{C}(n,k,s,r) = (A; R_0, ..., R_N)$ the *Cameron scheme* with parameters (n,k,s,r). This is a particular *association scheme* [Bo], [De], [MS]; it includes the Hamming schemes ($s = 1$) and the Johnson schemes ($r = 1$) as particular cases. The scheme can be thought of as a coloring of the edges of the complete graph on n vertices (including self-loops); we refer to the R_i as *color classes*.

It is clear that each group of Cameron type acts on a Cameron scheme. In fact, the color classes are precisely the orbitals of the action of $W(B,r)$ on A. It may, however, happen, that the color-classes split under the action of a Cameron-type group $G \leq W(B,r)$. One of our key subroutines, NATURAL_ACTION, will recover the imprimitive action of G on $C = rB$ using the orbital structure of the primitive G-action on A, thereby reducing the Cameron-type groups to imprimitive groups with a unique maximal block system of $r \leq \log n / \log 5$ blocks, in which giants act on each block.

2.6. Strong generators

In algorithms, permutation groups will always be input and output via a set of generators.

A standard tool for permutation group computation is a *strong generating set* (SGS) [Si70]. As generalized in [Bab79] (see also [FHL]), an SGS for G presumes any tower of subgroups

$$G = G_0 \geq G_1 \geq \ldots \geq G_r = 1.$$

An SGS is then a union of sets C_i of coset representatives for $G_{i-1} \bmod G_i$. Hence, any $\alpha \in G$ has a unique factorization $\alpha = \rho_1 \rho_2 \cdots \rho_r$ with $\rho_i \in C_i$. We call an SGS

NC-efficient if it comes along with an NC-procedure to factor any $\alpha \in G$. It is useful to observe that an SGS for a factor group G/N, pulled back to G, appended to an SGS for N, gives an SGS for G.

3. Organization of the algorithm

The overall algorithm follows the lines of [Lu86, Sections 4 and 5].

A *structure forest* for a permutation group $G \leq Sym(A)$ is a forest on which G acts as automorphisms such that: the leaves form the permutation domain A; the roots correspond to the orbits; and denoting by $G(v)$ the permutation group induced by G_v on the children of v, each $G(v)$ is *primitive*. As noted in [LM], NC contains the problem of computing a structure forest.

We shall need further refinements of the structure of the groups $G(v)$. In an *extended structure forest* we allow smaller trees $T(v)$ be appended from each node v of the structure forest. We identify v with the root of $T(v)$ and think of $T(v)$ being placed entirely between the levels of v and its original children. The leaves of $T(v)$ must form a faithful permutation domain for $G(v)$, and the entire group G should act on the extended forest. $T(v)$ is required to be a structure forest in the above sense for the new action of $G(v)$.

The insertion of these "small" trees allows us to utilize the structure of $G(v)$ through a different permutation representation. We use $G1(w)$ to denote the permutation group induced by G_w on the set of the *immediate children* of the node w of the extended structure forest. ($G1(v) = G(v)$ if no $T(v)$ has been appended at v of the original forest.)

Delving further into the structure of the primitive groups $G1(w)$, we define the *augmented structure forest* for G to be an extended structure forest F together with an assignment of to each node $w \in F$ of a tower of normal subgroups of $G1(w)$

$$(1) \quad 1 = G1_{m(w)}(w) \lhd \cdots \lhd G1_1(w) \lhd G1_0(w) = G1(w)$$

with semisimple quotients $G1_{i-1}(w)/G1_i(w)$, and such that the induced action of G on $\{G1(w)\}_{w \in F}$ induces, in turn, isomorphisms between subgroups at corresponding places in the towers.

The "small" trees $T(v)$ will arise from new permutation representations of some of the primitive groups $G(v)$, called "large groups", and found via our routine NATURAL_ACTION (Section 4).

The main phases of the SGS algorithm are these. The input is a set of generators for $G \leq Sym(A)$.

Main procedure

1. Construct structure forest.
2. For a representative v of each G-orbit of nodes of the forest, use NATURAL_ACTION to decide if $G(v)$ is a "large group" and, if so, construct new action and corresponding structure tree $T(v)$.
3. Via the G-action, transfer each $T(v)$ to all nodes in the orbit v^G, thus obtaining an extended structure forest.
4. For a representative w of each G-orbit of nodes of the extended forest F, construct a semisimple tower (1) of normal subgroups of $G1(w)$.
5. Via the G-action, transfer each semisimple tower to all nodes in the orbit w^G, thus obtaining an augmented structure forest.
6. As in [Lu86], use the augmented structure forest to construct efficient strong generators for G. **end.**

Phase 6 uses Luks' "generalized (commutative and noncommutative) linear algebra" and follows the lines of [Lu86]. We shall not discuss the details of that procedure here but we give a detailed account of the extra tools required for the implementation of the procedures of [Lu86] in the absence of structural constraints on G.

In order to complete Phases 4 and 6, we need to be able to perform management of the primitive groups $G1(w)$. This comprises finding efficient strong generators, normal closure, kernel of action, and a composition series. For giants, all this will be accomplished in Section 5; for "small groups" in Section 6 (building on [Lu87]). The remaining "large groups" will not occur as $G1(w)$, thanks to the routine NATURAL_ACTION. Thus, the proof of Theorem 1.1 will be complete by the end of Section 6. Parts (a) and (b) of Theorem 1.2 are follow immediately. A solution to part (d) is explicitly given in the course of the main procedure. We comment on the solution of part (c) as well as on further consequences in Section 7.

4. Reducing large to giant

In this section we classify primitive groups as "large" and "small". Large groups are seen to have a specific structure and a "natural" (often imprimitive) action comprised of giants acting on each block with a small group permuting the blocks. Thereby algorithmic problems are reduced to problems for giants and and small groups.

This objective is achieved by the subroutine NATURAL_ACTION. The procedure involves a global variable n, the degree of the permutation group which is the input of the full algorithm. We shall always assume that n is sufficiently large.

Procedure NATURAL_ACTION

INPUT: a primitive group $G \leq Sym(A)$, where $m := |A| \leq n$.

Step 1. if $m \leq 4 \log n$, **then (output** "small group"; **halt).**

Step 2. if G is 6-transitive, **then (output** $D := B_1 := A$, $r := 1$; **output** "giant"; **halt).**

Step 3. Consider the orbitals (G-orbits on $A \times A$). Let Γ be the second smallest and Δ the largest orbital. (* the smallest orbital is the diagonal *)

for each $(x, y) \in \Gamma$ compute (in parallel) the sets

$$B(x,y) = \Delta(y) - \Delta(x),$$

$$C(x,y) = A - \bigcup_{z \in B(x,y)} \Delta(z);$$

$D := \{C(x,y) | (x,y) \in \Gamma\}$.

Step 4. Consider the (transitive) G-action on D. Select a system $\{B_1, ..., B_r\}$ of minimal (nonsingleton) blocks of imprimitivity (* $\bigcup_i B_i = D$ *).

if ($k := |B_i| > 4 \log n$ **and** $r \leq \log n / \log 5$ **and** the stabilizer of B_1 is 6-transitive on B_1)

then output ("large group, faithfully acting on D" and a structure tree for the G-action on D that represents the blocks B_i by nodes adjacent to the leaves)

else output "small group"; **end.**

We say *G fails the large groups test* if "small group" is output. Otherwise G is said to *pass* the large groups test. The following result justifies the term "small groups" and provides additional information about large groups.

Theorem 4.1.

(1) *If NATURAL_ACTION outputs "giant" then G is a giant.*

(2) *If NATURAL_ACTION outputs "large group" then G acts faithfully on D and the stabilizer of each block B_i restricted to B_i contains $Alt(B_i)$.*

(3) *If NATURAL_ACTION outputs "small group" then, for sufficiently large n, $|G| < \exp(7 \log^2 n \ \log\log n)$.*

Statement (1) is obviously correct. For (2) we need the following lemma, whose proof is implicit in [Lu82, Lemma 3.6].

Lemma 4.2. *For $p \neq r$ primes, the order of the Sylow r-subgroups of the affine linear group $AGL(d,p)$ is less than p^{2d}.* ♠

Corollary 4.3. *For $k \geq 4d \log p$, the order of A_k does not divide the order of $AGL(d,p)$.*

Proof. Let $r = 3$ if $p = 2$ and let $r = 2$ otherwise. The result follows from Lemma 4.2 (except for the two easy cases $p = 2$, $d \leq 2$). ♠

Proof of Theorem 4.1, part (2). We say that the group H is *involved* in the group K if $H \cong L/M$ for some $M \lhd L \leq K$. If a simple group H is involved in K then H is involved in a composition factor of K.

We may assume G is not a giant. Let K be the kernel of the G-action on D. The stabilizer of B_1 restricted to B_1 is 6-transitive, whence it contains A_k. As the G-action on the set of blocks is transitive, the same holds for each B_i. Also, it follows that A_k is involved in G/K.

If $Soc(G)$ were abelian, then, by Theorem 2.5, $m = p^d$ for some prime p and $G \leq AGL(d,p)$. But, $d \log p = \log m \leq \log n \leq k/4$ and therefore, by Corollary 4.3, the order of A_k could not divide $|G|$. Hence $Soc(G)$ is nonabelian and the results stated in Theorem 2.6 apply. We use the notation of Theorem 2.6 and refer to $N \lhd G$ established there.

First we show that A_k is not involved in $G/Soc(G)$. Indeed, otherwise A_k must be involved either in G/N or in $N/Soc(G)$. The first case is impossible because $G/N \leq S_d$ (Theorem 2.6(b)) and $d \leq \log m / \log 5 < k/8$ (Theorem 2.6(c)). In the second case, A_k is involved in $N/Soc(G) \leq Out(T)^d$, a solvable group by Schreier's Hypothesis (Theorem 2.3), again a contradiction.

It follows now that A_k is involved in $Soc(G)$ and $K \not\geq Soc(G)$. Now $Soc(G)$ must be the unique minimal normal subgroup for otherwise, by Theorem 2.7, we have the contradiction:

$$n^2 \geq m^2 = |Soc(G)| \geq |A_k| = k!/2 > 2^k \geq n^4.$$

It follows that K contains no minimal normal subgroup, whence $K = 1$. ♠

Proof of Theorem 4.1, part (3). Assume the order of $|G|$ exceeds the stated bound. By Theorem 2.8 it follows that G is of Cameron type and A can be identified with the set of points of a Cameron scheme $C(m,k,s,r)$. Of course, the parameters and the identification are not known *a priori*. Our task is to prove that NATURAL_ACTION will have recovered this structure by Step 3.

In addition to the material of Section 2.5, we introduce some more notation concerning this Cameron scheme. We use the letters r, k, B_i, $C = rB = B_1 \cup ... \cup B_r$ to mean what they do in Section 2.5. We shall prove that this concurs with the output of NATURAL_ACTION (with D corresponding to C, the only object where identical notation could lead to confusion). We call the action of G on C "natural".

Each $a \in A$ corresponds to an s-transversal $T(a) \subset rB$.

Let Σ_i be the color class corresponding to the intersection pattern $(s-i, s, ..., s)$ and Φ to $(0, 0, ..., 0)$.

Claim 1. Σ_i $(0 \leq i \leq s)$ and Φ are orbitals of G, i.e. they do not split.

Proof. For Φ this follows from the fact that $G \geq A_k^r$. For Σ_i we need in addition that the stabilizer of any $a \in A$ acts transitively on the set of blocks $\{B_i\}$. ♠

Claim 2. $rs \leq \log m$.

Proof. $m = \binom{k}{s}^r \geq (k/s)^{rs} \geq 2^{rs}$. ♠

Claim 3. If $k < 2rs^2$ then $|G|$ satisfies the bound stated in Part (3) of Theorem 4.1.

Proof. $|G| \leq (k!)^r r! < (2rs^2)^{2r^2s^2} r! <$
$(2 \log^2 n)^{2 \log^2 n} (\log n)! < \exp(7 \log^2 n \ \log\log n)$. ♠

Claim 4. If $k \geq 2r$ then $\Gamma = \Sigma_1$.

Proof. Fix $x \in A$ and consider an orbital Θ. We have to prove that $|\Sigma_1(x)| < |\Theta(x)|$ for any Θ other than Σ_1 and the diagonal Σ_0. Observe that for $i > 1$,

$$|\Sigma_i(x)| = r\binom{s}{s-i}\binom{k-s}{i} > rs(k-s) = |\Sigma_1(x)|.$$

Assume now that Θ is contained in the color class with intersection pattern $(i_1, i_2, ...)$ where $i_2 < s$; let $(x, y) \in \Theta$. Just counting the images of y under the stabilizer of x in A_k^r we obtain

$$|\Theta(x)| \geq \binom{s}{i_1}\binom{k-s}{s-i_1}\binom{s}{i_2}\binom{k-s}{s-i_2}$$

$$\geq s^2(k-s)^2 > rs(k-s),$$

since $k \geq 2r$. ♠

Claim 5. If $k \geq 2rs^2$ then $\Delta = \Phi$.

Proof. We have to prove that Φ is the largest color class in the Cameron scheme. (Note that G plays no role here.)

First observe that for $1 \leq i \leq s$, the inequality $k \geq 2rs^2$ implies

$$r\binom{k-s}{s-i}\binom{s}{i} < \binom{k-s}{s}.$$

Let now the color class Θ have intersection pattern $(0^{r_0}, ..., s^{r_s})$. (The exponents denote multiplicities.) Then

$$|\Theta(x)| = \binom{r}{r_0, r_1, ..., r_s}\prod_{i=0}^{s}\binom{k-s}{s-i}^{r_i}\binom{s}{i}^{r_i} <$$

$$\binom{r}{r_0, r_1, ..., r_s}\binom{k-s}{s}^r\frac{1}{r^{r-r_0}} < \binom{k-s}{s}^r = |\Phi(x)|. \quad ♠$$

Claim 6. If $k \geq 2rs^2$ then the G-action on D is similar to the natural G-action on C.

Proof. For $b \in rB = C$, let $U(b) = \{u \in A | b \in T(u)\}$. We claim that $D = \{U(b) | b \in rB\}$. By Claims 2 and 3, $\Gamma = \Sigma_1$ and $\Delta = \Phi$. Thus, for any $(x, y) \in \Gamma$, the set $T(x) - T(y)$ is a singleton $\{b(x, y)\}$. Now, a simple inspection of the Cameron scheme, using the fact that $k > 3s$, shows that $C(x, y) = U(b(x, y))$.

The result follows since G acts transitively on C. ♠

5. The giants

5.1. The legal moves

Recall that the "giants" are the symmetric and alternating groups in their natural action. By testing 6-transitivity, we can decide whether or not G is a giant. We describe a procedure for *constructing* NC-efficient strong generators of the giants from the given generators. Henceforth, we use the term "construction" to mean a sequence of the following *legal operations: multiplication, inversion,* and *taking powers of permutations.* These operations can be implemented in NC. The exponent in the last case can be any integer with a polynomial number of digits [MC].

A *permutation circuit* is an algebraic circuit with permutations as inputs and outputs and legal operations as gates. A construction will be in NC if (from the generators) an NC-procedure builds a polylog depth, polynomial size permutation circuit which in turn (again from the generators) computes the desired output.

The reason for the constraint on the set of legal operations is that the procedure will be applied to the case when the actual permutation group G is imprimitive and acts on a set B of blocks as a giant. In such a case, although we know *a priori* that some $\sigma \in G$ acts on B as a given 3-cycle, no such permutation will be guaranteed to belong to G unless it has been constructed, by way of legal operations, from the generators of G.

We note that a byproduct of the procedure yields a simple, elementary proof of the old result, known to Jordan (1895) [Jo], (and vastly surpassed by Theorem 2.4) that the only $c \log^2 n / \log\log n$ -fold transitive permutation groups are the giants [BS1]. It also yields an $\exp(\sqrt{n \log n}(1 + o(1)))$ upper bound on the diameter of any Cayley graph of the giants [BS2].

The crux of the matter is the following result.

Theorem 5.1. *Given generators of a giant, one can construct, in NC, a cycle of length 3 (using legal operations only).*

Once a cycle of length 3 has been found, an NC-efficient set of strong generators is easily constructed (Section 5.6). Sections 5.2–5.5 are devoted to the proof of Theorem 5.1.

5.2. Pruning the Schreier generators

We begin the procedure with a preprocessing phase: finding coset representatives for the first $t < \log^c n$ members of the stabilizer chain.

Given $G \leq Sym(A)$ and $x \in A$, finding (right) coset representatives for G mod G_x amounts to a transitive closure problem. Once the coset representatives are known, we construct Schreier generators for G_x (Lemma 2.1). This step increases the number of generators by a factor $\leq n$. In order to avoid a superpolynomial blow-up, we prune the generators of our new giant G_x, keeping just enough to make the subgroup they generate 6-transitive on $A - \{x\}$. (Transitive closure on the set of 6-tuples of distinct elements of $A - \{x\}$.) The possible loss of odd permutations causes no harm (and is easily corrected by picking a single odd permutation from the

Schreier generators). This way the number of generators will never exceed n^6 and we can repeat the process a polylog number of times.

Remark. We make no attempt to minimize the number of processors. For $n > 25$, testing 4-transitivity would suffice. A simple trick [BLS] reduces the task to testing 2-transitivity and even more can be saved using additional tricks from [BLS].

Given the coset representatives just constructed, one easily constructs a member of G having a prescribed restriction on a subset of size t:

Lemma 5.2. *Given a giant $G \leq Sym(A)$ and an injection $f : D \to A$ where $D \subset A$ and $|D| \leq t \leq \log^c n$, one can construct in NC an element $\tau \in G$ such that $\tau|_D = f$.*

Proof. Let $A = \{1, ..., n\}$. Let G_i be the pointwise stabilizer of $\{1, ..., i\}$. Let $\{\alpha(i,j) : i \leq j \leq n\}$ be the coset representatives of G_{i-1} mod G_i just constructed, where $\alpha(i,j)$ fixes $1, ..., i-1$ and moves i to j $(1 \leq i \leq t)$. For any distinct $a_1, ..., a_d \in A$, recursively define $\pi(a_1, ..., a_d) = \rho\alpha(d, a_d^\rho)^{-1}$, where $\rho = \pi(a_1, ..., a_{d-1})$. Then, for $i \leq d$ we have $a_i^{\pi(a_1,...,a_d)} = i$. Let now $D = \{l_1, ..., l_d\}$. Then $\tau = \pi(l_1, ..., l_d)\pi(f(l_1), ..., f(l_d))^{-1}$ works. ♠

5.2. A commutator lemma

For $\pi \in Sym(A)$, we call a subset B of $supp(\pi)$ *independent* with respect to π if $B \cap B^\pi = \emptyset$. The *commutator* of $\pi, \tau \in Sym(A)$ is $[\pi, \tau] = \pi\tau\pi^{-1}\tau^{-1}$. The following is easily verified.

Lemma 5.3. *Let $\pi, \tau \in Sym(A)$. Assume that B is an independent set w.r. to π and $\tau|_{B^\pi}$ is the identity. Then $[\pi, \tau]|_B = \tau^{-1}|_B$.* ♠

Corollary 5.4. *Let G and t be as in Lemma 5.2. Assume $\pi \in G$ of degree s is given and $d \leq \min\{s/3, t/2\}$. Then, for any $(d_1, ..., d_r)$ such that $d_1 + \cdots + d_r = d$, we can find, in NC, an element $\lambda \in G$ such that λ includes cycles of lengths $d_1, ..., d_r$, and $deg(\lambda) \leq 2s$.*

Proof. Let $\pi \in G$ have degree s. As $s \geq 3d$, obviously, a π -independent set B of size d can be found. Since $t \geq 2d$, we can, by Lemma 5.2, *construct* an element $\tau \in G$ that fixes B^π pointwise and acts on B as a permutation with cycle structure $(d_1, ..., d_r)$. Now, the commutator $\lambda = [\pi\tau]$ will have the prescribed cycle structure on B by Lemma 5.3. Moreover, $deg(\pi) = deg(\tau\pi^{-1}\tau^{-1}) = s$, therefore $deg(\lambda) \leq 2s$. ♠

5.3. Large powers

Let p_i denote the i^{th} prime number, $p(r) = p_1 ... p_r$ and $f(n) = \min\{r | p(r) > n^4\}$. Let $g(n) = \sum_{i=1}^{f(n)} p_i$. The following estimates follow from the Prime Number Theorem.

Proposition 5.5. $f(n) = O(\frac{\log n}{\log\log n})$ *and* $g(n) = O(\frac{\log^2 n}{\log\log n})$. ♠

Lemma 5.6. *Let $\pi \in S_n$, $k = deg(\pi)$. Suppose π contains cycles of each prime length p_i, $i \leq r = f(n)$. Let $m(i)$ be the product of the lengths of all cycles of π divided by the highest possible power of p_i. Then $2 \leq deg(\pi^{m(i)}) < k/4$ for some $i \leq r$.*

Proof. Let $K = supp(\pi)$. For each $x \in K$, consider the set $P(x)$ of those primes p_i dividing the length of the π-cycle through x. Clearly, the product of these primes is $\leq k$.

Let $n(i)$ denote the number of points x such that $p_i \in P(x)$. Let us estimate the weighted average W of the $n(i)$ with weights $\log p_i$. Recall that the sum of the weights is $\sum \log p_i > \log(n^4) = 4 \log n$, therefore

$$W < \sum_{x \in K} \sum_{p_i \in P(x)} \log p_i / (4 \log n)$$

$$\leq (k \log k)/(4 \log n) \leq k/4.$$

We infer that $n(i) < k/4$ for some $i \leq r$. Clearly, $\pi^{m(i)}$ is not the identity and it fixes all but $n(i)$ points. ♠

5.5. Reducing the degree

Lemma 5.7. *Given an element $\pi \in G$ of degree $s \geq 3g(n)$, one can construct, in NC, a nonidentity element of degree $\leq s/2$.*

Proof. For $c = 2$ and $n \geq n_0$ we have $t \geq 2g(n)$. Let us apply Corollary 5.4 with $r = f(n)$, $d_i = p_i$ and thus $d = g(n)$. We obtain $\lambda \in G$ of degree $\leq 2s$ such that λ includes cycles of each prime length p_i, $i \leq r$. By Lemma 5.6, some NC-computable power $\lambda^m \neq 1$ has degree $< 2s/4 = s/2$. ♠

Proof of Theorem 5.1. Repeating the procedure of Lemma 5.7, we shall in $O(\log n)$ rounds arrive at some $\pi \in G$ of degree $2 \leq deg(\pi) < 3g(n)$. Since $t \geq 3g(n)$, by Lemma 5.2 we can construct $\tau \in G$ such that $|supp(\pi) \cap supp(\tau)| = 1$. Thus the commutator $[\pi, \tau]$ is a 3-cycle. ♠

5.6. Strong generators for a giant

Let $A = \{1, \ldots, n\}$. Let A_i be the set of transpositions $\{(i\ j) : i < j \leq n\}$ $(1 \leq i \leq n-1)$. These sets together form the *standard set* of strong generators of $Sym(A)$. Let $B_i = \{\tau(n-1\ n) : \tau \in A_i\}$. The sets B_i combine to the *standard set* of strong generators of $Alt(A)$.

Theorem 5.8

(a) *If G is one of the giants acting on A, the corresponding standard set of strong generators can be constructed in NC.*

(b) *Each standard set of strong generators is NC-efficient for the respective giant.*

Proof. I. First we prove part (b) for $G = Sym(A)$. Let $\pi \in G$. We give an NC-procedure to factor π via standard strong generators (cf. Section 2.6). For $i \in A$, let $l(i)$ be the length of the π-cycle through i. Set

$$j(i) = \max\{j \mid 0 \leq j \leq l(i) - 1,\ i^{\pi^j} \geq i\},$$

and $k(i) = i^{\pi^{j(i)}}$.

Claim. $\pi = (1\ k(1))(2\ k(2)) \cdots (n-1\ k(n-1))$.

Clearly, it suffices to prove the Claim for the case when π is a single cycle, say, of length l. Let $i_0 = \min\{supp(\pi)\}$. Then $k(i) = i$ for $i < i_0$ and $j(i_0) = l - 1$. Now, $(i_0\ j(i_0))\pi$ is an $l-1$-cycle and the proof follows by induction.

Since $l(i)$, $j(i)$, and $k(i)$ are clearly NC-computable, the proof of Part (b) for $G = Sym(A)$ is complete. Part (b) for $G = Alt(A)$ follows immediately. As a matter of fact, let $\pi \in Alt(A)$, and let $\pi = \tau_1\tau_2 \cdots \tau_s$ be the representation of π given in the Claim, where the τ_i are transpositions, i.e. the terms with $i = k(i)$ have been omitted. Now s is even, and, setting $\rho = (n-1\ n)$,

$$\pi = (\tau_1\rho)(\rho\tau_2)(\tau_3\rho)(\rho\tau_4)\cdots(\tau_{s-1}\rho)(\rho\tau_s).$$

Each parenthesised term is a standard strong generator of $Alt(A)$: $\rho\tau = \tau'\rho$ where $\tau' = \rho\tau\rho^{-1}$.

II. Now we prove part (a). By Theorem 5.1, we have already constructed a 3-cycle. All 3-cycles are conjugates of a single one. The conjugating elements can be constructed by Lemma 5.2 with $t = 3$. Products of pairs of 3-cycles provide the additional standard strong generators of $Alt(A)$: $(a\ c\ b)(c\ b\ d) = (a\ b)(c\ d)$. This completes the proof for $Alt(A)$. Assume now that $G = Sym(A)$ and $\pi \in G$ is an odd generator. Using the strong generators just constructed for $Alt(A)$, we can construct $\sigma = (1\ 2)\pi$ and thus the transposition $(12) = \sigma\pi^{-1}$. The conjugates of (1 2) provide the standard strong generators for $Sym(A)$. ♠

5.7. Giant management

Let $G \leq Sym(A)$ be a giant, $|A| \geq 5$. For a set $S \subset G$, the normal closure is $Alt(A)$ if $S \subset Alt(A)$, and $Sym(A)$ otherwise. The kernel of any G-action is either 1 or $Alt(A)$ or $Sym(A)$ and these cases are easily distinguished. Likewise the unique composition series of G is easily constructed.

6. Managing small groups

Again, let n be a global parameter (the degree of the input group for the main procedure). The resource limitations in the definition of NC below refer to this global parameter.

Theorem 6.1 *Let $G \leq Sym(A)$, where $|A| = m \leq n^c$. Assume $|G| < \exp(\log^c n)$. Then the following problems are in NC:*

(a) *Finding efficient strong generators of G.*

(b) *Finding the normal closure of a subgroup of G.*

(c) *Finding the kernel of any G-action.*

(d) *Finding a composition series for G.*

(e) *Finding the pointwise set-stabilizer of $B \subset A$.*

The proof of (a) is based on a combination of the Schreier generator method and Sims' sifting. Sifting becomes feasible because the length of any subgroup chain in G is $\leq \log|G| < \log^c n$. We need the following routine.

Procedure SIFTSTEP

INPUT: $S \subseteq Sym(A), S \neq \emptyset$.
OPTIONAL INPUT: $x \in A$.

Step 1. if $\langle S\rangle = 1$ **then halt.**
Step 2. if no $x \in A$ has been input **then** let x be any element of the support of a member of S.
Step 3. for each $y \in x^S$, select $\tau = \tau(y) \in S$ such that $x^\tau = y$; $T := \{\tau(y) | y \in x^S\}$.
Step 4. output x, T and the set $S' = \{\sigma\tau^{-1} | \sigma \in S,\ \tau = \tau(x^\sigma)\}$; **end.**

We informally describe **Procedure** SCHREIER_SIFT. This procedure constructs strong generators with respect to the stabilizer chain

$$G = G_0 \geq G_1 \geq \ldots \geq G_r = 1,$$

where G_i fixes $\{a_1, \ldots, a_i\} \subset A$. The order of the elements of A as well as the value $r \leq \log|G|$ are determined by the procedure such that G_{i+1} always be a proper subgroup of G_i. At the end of the i^{th} Phase, we shall have a set S_i of generators for G_i. If $G_i = 1$, set $r = i$ and halt. Else, we begin Phase (i+1) by letting a_{i+1} be any element in the support of any member of S_i unless a_{i+1} has already been defined. We solve (in NC) the transitive closure problem that yields coset representatives of G_i mod G_{i+1}. Next we construct the corresponding set R_{i+1} of Schreier generators for G_{i+1}. We apply SIFTSTEP to $S = R_{i+1}$ with optional input a_{i+2}, then again to S' and a_{i+3}, etc. until we halt because $S^{(j)} = \{1\}$. Set $S_{i+1} = \bigcup T^{(j)}$ where $T^{(j)}$ is the set T output by the j^{th} round of SIFTSTEP. End Phase (i+1).

Observe that for each i, $|S_i| < nr$ and $r \leq log|G|$. Consequently, for small groups, this procedure remains within NC, proving part (a) of the Theorem. Part (b) is solved

by a simple modification along the lines of [FHL], again noting that every chain of subgroups has now polylog length. The kernel of action easily reduces to normal closure (cf. [LM]). Given these ingredients, Luks' algorithm for finding a composition series in a permutation group G [Lu87] can be implemented in NC, provided $|G| < \exp(\log^c n)$. Finally, part (e) is inherent in SCHREIER_SIFT if preference for the next input point is always given to unfixed elements of B. ♠

7. Corollaries

In addition to the results listed in Theorem 1.2, we mention several more.

Theorem 7.1 *Given $G \leq Sym(A)$, NC includes*
(1) *Finding the normal closure of a subset of G.*
(2) *Finding the kernel of an action of G.*
(3) *Finding the derived series of G.*
(4) *Finding the centralizer of $H \leq Sym(A)$ in G assuming that G normalizes H.*
(5) *Finding $G \cap H$ assuming that G normalizes H.*
(6) *Factoring $\alpha \in GH$ as $\gamma\delta$, with $\gamma \in G$ and $\delta \in H$, assuming that G normalizes H.*

The proofs of (1) and (2) follow the lines of [Lu86]. For (3), we note that the commutator subgroup is obtainable as the normal closure of the commutators of the generators; use this repeatedly, noting the following lemma.

Lemma 7.2 *The length of the derived series of any $G \leq S_n$ is $O(\log^2 n)$.*

Proof. Let m be the length of the largest orbit of G. We actually prove, by induction on the depth k of the structure forest of G, that the length $d(G)$ of the derived series of G is $O(\log^2 m)$. By [Ba86, Lemma 11.2], any chain of normal subgroups of a primitive group has length $O(\log^2 n)$. This settles the case $k = 1$. The induction step uses the observation $d(G) \leq d(N) + d(G/N)$ if $N \lhd G$. ♠

We say that $K \lhd G$ is an *NC-constructible kernel* if NC contains the problem of constructing a representation $G \to Sym(B)$ with kernel K; in particular, by Theorem 7.1(2), one can find such K. The centralizer in (4) is an NC-constructible kernel [Lu87]. Note that part (c) of Theorem 1.2 is a particular case of (4).

We give an indication of the solution to (5). Our basic procedures (see [Lu86]) construct a series of normal subgroups of G

$$1 = G_r \lhd \cdots \lhd G_1 \lhd G_0 = G,$$

with $r = O(\log^c n)$ and semisimple G_i/G_{i+1}. Available, as well, are faithful representations of the quotients G_i/G_{i+1}. This gives rise to the normal series

$$H = G_r H \lhd \cdots \lhd G_1 H \lhd G_0 H = GH.$$

Using the epimorphism $G_i/G_{i+1} \to G_iH/G_{i+1}H$ and exploiting the semisimplicity of G_i/G_{i+1}, one constructs a faithful action for each $G_iH/G_{i+1}H$. Then $G \cap G_{i+1}H$ is the kernel of this action restricted to $G \cap G_iH$; thus, $G \cap H$ is constructed in r rounds from $G \cap GH = G$.

The solution to (6) involves keeping track of factorizations in the construction of an SGS for GH from generators for G (factored as $\gamma = \gamma 1$) and generators for H (factored as $\delta = 1\delta$). Acceptable "G-factors" for products, inverses, powers are always obtainable as corresponding products, inverses, powers of G-factors. Similarly, the factorization of $\alpha \in GH$ parallels its membership test. ♠

Remarks. 1. The solutions to (5) and (6) resolve the first three open problems proposed in [Ba86].
2. A solution to (4) is also obtainable via (5) and an NC-construction of the centralizer of H in $Sym(A)$.

For application in Section 8, we show the constructibility of some other classes of characteristic subgroups. Let $\mathcal{T}$ be a class of isomorphism types of simple groups. For any group G, we define, $Res_{\mathcal{T}}(G)$ to be the (unique) minimal $R \lhd G$ such that G/R is a product of simple groups from the class $\mathcal{T}$. In particular, we are concerned with the subgroups $Res_{\mathcal{A}}(G)$, $Res_{\mathcal{N}}(G)$, $Res_p(G)$, $Res(G)$ corresponding to $\mathcal{T}$ being the class of abelian simple groups, nonabelian simple groups, simple p-groups, all simple groups, respectively. We call $Res(G)$ the *residual* of G.

Theorem 7.3 *Given $G \leq Sym(A)$, $Res_{\mathcal{A}}(G)$ is an NC-constructible kernel.*

Proof. For each prime $p \leq |A|$, $Res_p(G)$ is the (normal) subgroup generated by the derived group of G and the pth powers of the generators of G. Linear algebra in $G/Res_p(G)$ is used to establish $Res_p(G)$ as an NC-constructible kernel. The action of G on the disjoint union of these actions over all p has kernel $Res_{\mathcal{A}}(G)$. ♠

In anticipation of a later strengthening (Theorem 8.3), for now we list the following as a lemma.

Lemma 7.4 *Let $G \leq Sym(A)$ and suppose $|G| < \exp(\log^c n)$. Then $Res_{\mathcal{N}}(G)$, and therefore $Res(G) = Res_{\mathcal{A}}(G) \cap Res_{\mathcal{N}}(G)$, is an NC-constructible kernel.*

Proof. We determine

$$\mathcal{M}(G) = \{M \lhd G \mid G/M \text{ is nonabelian simple}\}.$$

Faithful representations of each G/M are constructible [Lu87] and then $Res_{\mathcal{N}}(G)$ is the kernel of the action on the disjoint union. Since $G/Res_{\mathcal{N}}(G) \cong \prod_{M \in \mathcal{M}(G)} G/M$, one knows that $|\mathcal{M}(G)| \leq |A|$.

Now let K be any maximal normal subgroup of G. We recursively determine $\mathcal{M}(K)$ (the depth of the recursion $G > N > \cdots$ will be polylog). Let $M \in \mathcal{M}(G)$. We

may assume that $M \neq K$. Then $N = M \cap K \in \mathcal{M}(K)$ and M/N is the centralizer of K/N in G/N. Thus, if $\alpha \in G - K$ centralizes K mod N, $M = NCL_G(\langle \alpha, N \rangle)$. So, knowing N, we only need a suitable α to locate M. Now, in general, centralizers in quotient groups seem difficult to obtain (even sequentially). However, in this case, we can reduce the problem to Theorem 7.1(4). Using representations of G/K and K/N we find $N < X < K < Y < G$ such that $|Y : K|$ and $|K : X|$ are polynomial size, and determine coset representatives for Y mod K and K mod X; for example, take for X, Y, appropriate terms in point-stabilizer chains in those representations (Theorem 6.1(e)). This enables us to represent Y on the cosets of X in Y and to find α centralizing the action of K. Of course, we do this for all $N \in \mathcal{M}(K)$ in parallel, rejecting those that do not yield a suitable M. ♠

8. Pointwise set-stabilizers

We outline the ingredients of Theorem 1.3. For brevity herein, we must rely heavily on the discussion in [Lu86, Section 6]. The following may be uncomfortably vague without that reference in hand.

Let H^O denote the group induced by $H \leq G$ in the orbit O. We may assume that each G^O is primitive and that each orbit (under current investigation) contains exactly one *target* point to be fixed; G is always interpreted as the group induced on these orbits. The objective is to locate a collection Y of orbits and a normal subgroup $N \triangleleft G$ such that

I. For $O \in Y$, $N^O = Soc(G^O)$.

II. The actions of N on the orbits in Y are reasonably "independent".

III. Fixing targets in Y makes "significant" progress.

Since N^O, for $O \in Y$, is necessarily transitive, one can use elements in N^O to modify (in parallel) each generator of G so that it fixes the target in O. Furthermore, by II, this can be done in parallel for many O. The method then depends upon a formalization of "significant" in III.

The inadequacy of the method in [Lu86] is that N was captured in a series of length $O(\log^c n)$, where c is a function of the class of composition factors of G. By offering an alternate approach to III, we are able to use a shorter series, reducing c to $O(1)$.

It is convenient to deal first with the case when the G^O are small groups, which, for the moment, we'll interpret as $\log |G^O| \leq \log^c n$ with c arbitrary but fixed. For $H \leq G$, let $\mathcal{R}(H)$ consist of those elements that project into $Res(H^O)$ (Section 6) for each O.

Lemma 8.1 *$\mathcal{R}(H)$ is an NC-constructible kernel.*

Proof. Construct, by Lemma 7.4, an action of each H^O with kernel $Res(H^O)$. Then H acts naturally on the disjoint union of the domains with kernel $\mathcal{R}(H)$. ♠

As noted in [Lu86], the ith term in tower

$$1 \triangleleft \cdots \triangleleft \mathcal{R}(\mathcal{R}(\mathcal{R}(G))) \triangleleft \mathcal{R}(\mathcal{R}(G)) \triangleleft \mathcal{R}(G) \triangleleft G$$

projects on the i residual of each G^O, hence the tower length, t, is at most $\log^c n$. One also knows that the socle of each (primitive) G^O occurs as a projection of a unique term in the tower. We choose N to be the term in the tower that projects onto a maximal number of socles; that is, going from N to $\mathcal{R}(N)$ the action on a maximal collection, X, of orbits first becomes trivial. Then X includes at least $1/t$ of the orbits. It suffices then to describe a procedure that will decrease G^O for all $O \in X$, so we'll assume that X contains all orbits. The subcollection $Y \subset X$ is chosen as in [Lu86], in particular, at each $O \in Y$ there is a subgroup $1 \neq S(O) \triangleleft G^O$ such that $N = \prod_{O \in Y} S(O)$ (in two of three 'socle-types', $S(O) = N^O$). Next, N is employed to cut G down to a subgroup H that fixes the target points in Y. As a consequence, $(N \cap H)^O < N^O$ for all $O \in X$. (For the groups with two minimal normal subgroups, this procedures requires $\log n$ rounds of N, Y selection and application). The following lemma enables us to measure progress and justify the significance of Y.

Lemma 8.2 $\log |H| \leq \log |G|(1 - \frac{1}{\log^c n})$

Idea of proof. One shows that if $\alpha \in G$ projects into N^O for $O \in Y$ then it does so for all $O \in X$. This is used to show $\log |G| \leq y \log^c n$, where $y = |Y|$. Since $\frac{|G|}{|H|} \geq \frac{|N|}{|N \cap H|} \geq 2^y$, the result follows. ♠

Thus, by the bound on G, $O(\log^{c+1} n)$ rounds suffice to fix targets in X.

We turn now to the general group case. Again, we may assume the orbit constituents are primitive. It suffices to consider only the orbits where giants or large groups occur (as declared by procedure NATURAL_ACTION), for having fixed the targets there, we dispatch the rest with the above small-groups technique.

We suppose that we have deciphered the natural actions. Now, it is possible that groups across various orbits are "linked" in a diagonal action (see [Lu86]). In such case, one can identify their natural actions (thus, identifying their C's, B_i's). Using the deciphered natural action, we can identify each target point x with its s-transversal in the corresponding C and we color the points of the transversal so as to recall their x-origin. In general, a point may lie in several transversals and we refine the colorings so that identical colors imply identical originating sets. The objective now is to find the subgroup that fixes colored sets in the natural actions.

In each C, we group the blocks B_i into classes according to their list of colors. The first task is to stabilize classes in the action on the blocks. Here, we make use of the observation that the induced action on the blocks

of C is a polynomial size group, for $k \geq 4 \log m$ forces $r! = O(m)$. But the class- (or color-) stabilization problem easily reduces to pointwise set stabilizer in this case by looking at actions on appropriate cosets within each orbit [LM], [Ba86]. Since these orbit actions are small, this is done by above methods.

The next step uses an $N \triangleleft G$, as in the small-groups case, to cut the group down. Suppose that not all color classes are singletons. Then using elements of the $Alt(B_i)$'s we can modify each generator of G so that it fixes colors. By the independence of the natural actions, this can be done in parallel across all such C. These modified generators, together with the easily-constructed color-preserving subgroup of the alternating-group-product action, generate the answer.

The above generator modification may not be possible when color classes are singletons. However, an element that is not so modifiable should not be in the answer anyway. Thus, in a first pass, we cut down to modifiable elements. Use the (singleton) color classes to rank the elements in each B_i. Thus the actions $B_i \xrightarrow{\alpha} B_j$ of $\alpha \in G$ can be identified as creating an *even* or an *odd* permutation of ranks. Assign to each B_i a new two-element set, with one point colored red and one blue. The action of G is extended to the union of these new pairs so that colors are preserved iff the map between the corresponding B_i's is even. The objective is to cut G down so that it preserves the red set. This is again a small group problem. Now, every element in G is modifiable as we proceed as above.

We conclude with an application promised in Section 7.

Theorem 8.3 *Given $G \leq Sym(A)$, NC contains the problem of finding $Res(G)$.*

Idea of proof. The proof of Lemma 7.4 required pointwise set stabilizer, which we now have. Also, to guarantee a small depth of recursion, we modify the proof so that it suffices to assume G/K is semisimple. ♠

Remark. As far as we know, the determination of $Res(G)$ was not previously observed to be in P.

9 Applications to graph isomorphism

We discuss only the proof of Theorem 1.4. Other graph-isomorphism applications ([Lu86], [Ba86]) will be developed in a final paper.

We rely on the discussion in [Ba86]. There were three points at which coin-tossing was invoked:

(1) To factor polynomials over tiny (input in unary) fields.

(2) To find a prime $p < 4n^2$ modulo which eigenspaces of a (0,1)-matrix remain non-isotropic.

(3) To find a pointwise set stabilizer in a group with polynomial size orbit constituents.

The first two are avoidable: the necessary factorization for (1) is in NC [vzGS]; for (2), instead of picking a random p, try them all in parallel. Finally, (3) has now been resolved even without the orbit restriction.

10. Open problems

1. Find the Sylow subgroups of G in NC. A polynomial-time solution to this problem is given by Kantor [Ka2], but the NC question is open even for solvable groups.

2. Find, in NC, set-stabilizers in 2-groups. By [Lu86], this would put trivalent graph isomorphism in NC.

3. Find the descending central series of a nilpotent group in NC. The problem is easily in P. Note that, unlike the derived series, this series does not necessarily have poly-log length. On the other hand, some central composition series, which is at least as long, is NC-constructible [LM].

4. Find the ascending central series of a nilpotent group in NC. We can show the problem is in P but our technique uses group *intersection* in the *sequential* construction of each term as the center of the group modulo the last term. Thus there are *two* obstructions to parallelizability.

5. If H is a subgroup of G of polynomial index, construct a complete set of coset representatives for G mod H. Again, the problem is in P. In fact, it is in Las Vegas–NC, for one can generate random elements of G using an SGS and an appropriate number of these will hit all cosets with high probability [Ba79]. Note that if the group induced by the action of G on the cosets of H itself has polynomial size, the "reachability-lemma" of Babai and Szemerédi [BSz] (see also [Ba86]) guarantees an NC-solution.

6. Now that pointwise set stabilizers are available, we can also, in NC, produce an SGS in Sims' sense [Si70], that is, allowing G_i (Section 2.6) to be the pointwise set stabilizer of the first i points. However, it remains open whether such an SGS is necessarily NC-efficient. To be precise, are the factorizations, $\alpha = \rho_1 \rho_2 \cdots \rho_r$ with $\rho_i \in C_i$, obtainable in NC?

7. The deterministic methods for the small-orbit-group case of pointwise set-stabilizer (Section 8) do not follow the basic approach of the Las Vegas solution in [Ba86]. That probabilistic method relied on an approach to finding large independent sets in a modular lattice. Indeed, for the latter problem, a Las Vegas – deterministic gap persists. We refer the reader to [Ba86, Section 9] for a description of this and related questions.

References

[Ba79] Babai, L., *Monte Carlo algorithms in graph isomorphism testing*, Tech. Rep. 79-10, Dép. Math. et Stat., Univ. de Montréal, 1979.

[Ba86] L. Babai, A Las Vegas–NC algorithm for isomorphism of graphs with bounded multiplicity of eigenvalues, Proc. 27th IEEE FOCS, 1986, 303–312.

[BGM] L. Babai, D.Yu. Grigoryev and D.M. Mount, *Isomorphism of graphs with bounded eigenvalue multiplicity,* Proc. 14th ACM STOC, 1982, 310–324.

[BKL] L. Babai, W.M. Kantor and E.M. Luks, *Computational and the classification of finite simple groups,* Proc. 24th FOCS, 1983, 162–171.

[BLS] L. Babai, E.M. Luks and Á. Seress, *Managing permutation groups in $O(n^4 \log^c n)$ time,* in preparation

[BS1] L. Babai and Á. Seress, *On the degree of transitivity of permutation groups: a short proof,* J. Combinatorial Theory-A, to appear

[BS2] L. Babai and Á. Seress, *On the diameter of Cayley graphs of the symmetric group,* manuscript

[BSz] L. Babai and E. Szemerédi, *On the complexity of matrix group problems,* Proc. 24th IEEE FOCS, 1984, 229-240.

[Bo] R.C. Bose, *Strongly regular graphs, partial geometries, and partially balanced designs,* Pacific J. Math. 13 (1963), 389-419.

[Cam] P.J. Cameron, *Finite permutation groups and finite simple groups,* Bull. London Math. Soc. 11 (1979), 161-169.

[Car] R. Carter, Simple groups of Lie type, Wiley, London 1972

[Ca] S.A. Cook, *The classification of problems which have fast parallel algorithms,* Proc. Conf. FCT'83, Lecture Notes in Comp. Sci. 158, Springer 1983, 78–93.

[CKS] C.W. Curtis, W.M. Kantor and G.L. Seitz, *The 2-transitive permutation representations of the finite Chevalley groups,* Trans. A.M.S. 218 (1976), 1-57.

[De] P. Delsarte (1973), *An algebraic approach to the association schemes of coding theory",* Philips Res. Rept. Suppl. 10 (1973).

[FHL] M.L. Furst, J. Hopcroft and E.M. Luks, *Polynomial time algorithms for permutation groups,* Proc. 21th IEEE FOCS, 1980, 36–41.

[vzGS] J. von zur Gathen and G. Seroussi, *Boolean circuits versus arithmetic circuits,* Proc. 6th Intl. Conf. Comp. Sci., Santiago, Chile, 1986, 171-184.

[Go] D. Gorenstein, *Finite simple groups and their classification,* Academic Press, 1986

[Ha] M. Hall, Jr., The Theory of Groups, Macmillan, New York 1959.

[HW] G.H. Hardy and E.M. Wright, An Introduction to the Theory of Numbers, 5th ed., Clarendon Press 1979.

[Jo] C. Jordan, *Nouvelles recherches sur la limite de transitivité des groupes qui ne contiennent pas le groupe alterné,* Journ. de Mathématiques (5) 1 (1895), 35-60.

[Ka1] W.M. Kantor, *Permutation representations of the finite classical groups of small degree or rank,* J. Algebra 60 (1979), 158–168.

[Ka2] W.M. Kantor, *Sylow's theorem in polynomial time,* J. Comp. Sys. Sci. 30 (1985), 359-394.

[Lu82] E.M. Luks, *Isomorphism of graphs of bounded valence can be tested in polynomial time,* J. Comp. Sys. Sci. 25 (1982), 42–65.

[Lu86] E.M. Luks, *Parallel algorithms for permutation groups and graph isomorphism,* Proc. 27th IEEE FOCS, 1986, 292–302.

[Lu87] E.M. Luks, *Computing the composition factors of a permutation group in polynomial time,* Combinatorica 7 (1987), 87–99.

[LM] E.M. Luks and P. McKenzie, *Fast parallel computation with permutation groups,* Proc. 26th IEEE FOCS, 1985, 505–514.

[MS] F.J. Macwilliams and N.J.A. Sloane (1978), The Theory of Error-correcting Codes, North–Holland, Amsterdam 1978.

[MC] P. McKenzie and S.A. Cook, *The parallel complexity of the abelian permutation group membership problem,* Proc. 24th IEEE FOCS, 1983, 154–161.

[Mu] K. Mulmuley, *A fast parallel algorithm to compute the rank of a matrix over an arbitrary field,* Combinatorica 7 (1987), 101–104.

[Pi] N. Pippenger, *On simultaneous resource bounds,* Proc. 20th IEEE FOCS, 1979, 307–311.

[Sc] L.L. Scott, *Representations in characteristic p,* in: Proc. Santa Cruz Conf. on Finite Groups, A.M.S. 1980, 319–322.

[Si67] C.C. Sims, *Graphs and finite permutation groups,* Math. Z. 95 (1967), 76–86.

[Si70] C.C. Sims, *Computational methods in the study of permutation groups,* in: Computational Problems in Abstract Algebra, J. Leech, ed., Pergamon Press 1970, 169–183.

[Wi] H. Wielandt, Finite Permutation Groups, Acad. Press, N.Y. 1964.