# Computing in Quotient Groups

*William M. Kantor*[1]
University of Oregon

*Eugene M. Luks*[2]
University of Oregon

## Abstract

We present polynomial-time algorithms for computation in quotient groups $G/K$ of a permutation group $G$. In effect, these solve, for quotient groups, the problems that are known to be in polynomial-time for permutation groups. Since it is not computationally feasible to represent $G/K$ itself as a permutation group, the methodology for the quotient-group versions of such problems frequently differ markedly from the procedures that have been observed for the $K = 1$ subcases. Whereas the algorithms for permutation groups may have rested on elementary notions, procedures underlying the extension to quotient groups often utilize deep knowledge of the structure of the group.

In some instances, we present algorithms for problems that were not previously known to be in polynomial time, even for permutation groups themselves ($K = 1$). These problems apparently required access to quotients.

## 1. Introduction

Since the order of a permutation group $G$ on $n$ letters can be exponential in $n$, it is customary, in both theory and practice (see, e.g., [Ca], [FHL], [Si]), to specify $G$ by a small set of generating permutations (less than $n$ are needed and typically many fewer suffice). Despite the succinctness of such representations, a substantial polynomial-time machinery has developed for computing with permutation groups. A major stimulus for this activity was the application to the graph isomorphism problem (ISO), for early work ([Ba1], [FHL], [Lu1], [Mi1], [Mi2], [BL]) used groups to put significant instances of ISO into polynomial time. Ensuing studies resulted in algorithms for deciphering the basic building blocks of the group ([BKL], [Lu2], [Ne], [KT], [Ka1], [Ka2], [Ka3], [BLS1]), making available constructive versions of standard theoretical tools.

[1]Research partially supported by NSF Grant DMS 87-01784 and NSA grant MDA 904-88-H-2040.

[2]Research partially supported by NSF Grant DCR-8609491 and ONR Grant N00014-86-0419.



One essential ingredient has, to a great extent, been lacking. The facility to deal with quotient groups (equivalently, homomorphic images of groups) is a central methodology of group theory, but there has not seemed to be an effective computational analogue. In practice, group theory systems do offer permutation representations of quotients [Ca]. But, from the standpoint of worst-case complexity, this reduction back to permutation groups cannot work. The reason is that quotients of given permutation groups need not have faithful (1-1) permutation representations on a polynomial-size set. For example, in illustrating the computational blowup, Neumann [Ne] gives an example of a 2-group acting on $n$ letters, a quotient of which has no faithful representation on less than $2^{n/4}$ letters.

The above difficulties notwithstanding, we introduce methods for dealing with quotient group problems that close the apparent complexity gap. In fact, we are motivated to conjecture a **Quotient Group Thesis:**

> *If a problem for quotient groups $G/K$ of permutation groups has a polynomial-time solution when $K = 1$ then it has a polynomial-time solution in general.*

Corroborating testimony for the Quotient Group Thesis is our extension of the polynomial-time library for permutation groups (as we see it) to quotients of permutation groups. We employ a variety of techniques in this extension, including two very useful tools (the Sylow and Frattini methods in §5) for lifting problems on $G/K$ to problems on a "reasonable" $G$. In the process, we enhance the algorithmic infrastructure even for the case $K = 1$: some issues seem to have required access to quotients.

For several problems, the procedures for handling $G/K$ are easy consequences of those for the special case $K = 1$. But in other, critical instances this is far from the case. As confirmation, witness the difference in the nature of the underlying theoretical tools. For example, demonstrating, from first principles, that the center of a permutation group is computable in polynomial-time involves only elementary properties of groups and no other knowledge of the group structure ([Lu2], [CFL]); it is, in fact, interpretable as the subgroup fixing a set of points (in an augmentation of the set) [Lu2] and so computable

by the most basic algorithm in [FHL]. Such a concrete interpretation is not available for quotient groups. To show that the *center of a quotient group* can be computed efficiently, we make essential use of the Sylow structure made available via procedures in [Ka2], [Ka3], procedures that are dependent upon the ($\sim$ 15000 page) *classification of finite simple groups*. Nevertheless, we emphasize that our new methods do not require a deep knowledge of this classification or of other algebraic theory of great depth. The procedures that we cite (e.g., from [Lu2], [Ka2], [Ka3]) have elementary specifications; so they are quite accessible to non-specialists.

Another aspect of the procedures seems worth highlighting. From one point of view, we are effectively manipulating induced permutation representations in which the new permutation domains are themselves too large to enumerate. One example occurs in the consideration of the transitive action of a permutation group $G$ on its, possibly exponential, collection $\mathcal{P}$ of Sylow $p$-subgroups of $G$ (where $g \in G$ maps $P \mapsto P^g = g^{-1}Pg$); given two "points" $P_1, P_2 \in \mathcal{P}$, we need to find some (and sometimes all) $g \in G$ such that $P_1^g = P_2$. Another example is our computation of *cores* of subgroups $H$ of $G$, for the core is interpretable as the kernel of the permutation representation of $G$ on the set of (right) cosets of $H$ in $G$.

Section 2 has an extended glossary of group-theoretic terminology, most of which is standard; the reader should refer back to this as needed. In Section 3, we recall a few fundamental algorithmic results for permutation groups. We present, in Section 4, the backbone of the polynomial-time library for computing with permutation groups as well as with quotients of permutation groups; see that section also for a pointer to the proofs in Sections 5-12. Some intriguing open questions are indicated in Section 13.

In the Appendix, we discuss some problems that are routinely solved in practical computation but are of uncertain complexity. A polynomial-time solution to any of these would also resolve ISO. The "library" in Section 4 also serves as an update on solutions to special cases of the problems that are highlighted in the Appendix.

We emphasize that the issue herein is polynomial-time computation. With that in mind, we freely trade efficiency for exposition. In particular, we make no attempt either to optimize worst-case time-bounds or to describe efficient implementations. Of course, these are well-motivated, related issues, and each is the object of a growing literature.

A more complete collection of algorithms and proofs will appear in [KL].

## 2. Definitions and notation

We recall some group-theoretical terminology.

Throughout, let $G$ be a finite group. We write $H \leq G$ to indicate that $H$ is a subgroup of $G$, and $H \trianglelefteq G$ to indicate that $H$ is a normal subgroup; then $H < G$ and $H \lhd G$ indicate that the inclusions are strict. A group $G$ is *simple* if there is no $H$ such that $1 \lhd H \lhd G$. We say that $H$ is *subnormal* in $G$ if there is a chain of subgroups of the form $H = L_0 \trianglelefteq L_1 \trianglelefteq \cdots \trianglelefteq L_m = G$. If $S \subseteq G$ then $\langle S \rangle$ is the subgroup of $G$ generated by $S$. For $s, t \in G$, we write $s^t$ for the *conjugate of $s$ by $t$*, namely, $s^{-1}ts$; and extend the notation to subsets $S, T \subseteq G$ via $S^T = \{s^t \mid s \in S, t \in T\}$. The *normalizer* and *centralizer* of $S$ in $G$ are $\mathrm{N}_G(S) = \{g \in G \mid S^g = S\}$ and $\mathrm{C}_G(S) = \{g \in G \mid s^g = s, \forall s \in S\}$, respectively; subsets of $\mathrm{N}_G(S)$ or $\mathrm{C}_G(S)$ are said to *normalize* or *centralize* $S$, respectively. The *center* of $G$ is $Z(G) = \mathrm{C}_G(G)$. The *normal closure* $S$ in $G$ is the smallest normal subgroup of $G$ containing $S$, namely $\langle S^G \rangle$. If $H \leq G$ then the *core of $H$ in $G$*, $\mathrm{Core}_G(H)$, is the largest normal subgroup of $G$ contained in $H$, namely $\bigcap \{H^g \mid g \in G\}$.

We refer to [Ha] for a discussion of Sylow's Theorem: (i) If $p$ is a prime then a *Sylow $p$-subgroup* of $G$ is a $p$-subgroup whose order is the $p$-part of $|G|$; (ii) any $p$-subgroup of $G$ is contained in a Sylow $p$-subgroup; (iii) any two Sylow $p$-subgroups $P_1$, $P_2$ are conjugate in $G$: $P_1^g = P_2$ for some $g \in G$.

For $s, t \in G$, we write $[s,t]$ for the *commutator* $s^{-1}t^{-1}st$, and for $S, T \leq G$, we set $[S,T] = \langle [s,t] \mid s \in S, t \in T \rangle$. The *derived subgroup* of $G$ is $G' = [G,G]$. The *derived series* of $G$ is the series $G \geq G' \geq (G')' \geq \cdots$; $G$ is *solvable* if this series terminates with the group 1. The *lower central series* of $G$ is defined recursively by: $L_1(G) = G$ and $L_{i+1}(G) = [G, L_i(G)]$; $G$ is *nilpotent* if this series terminates with the group 1. The *upper central series* of $G$ is defined recursively by: $Z_0(G) = G$ and $Z_{i+1}(G)/Z_i(G) = Z(G/Z_i(G))$; $G$ is nilpotent iff this series terminates with the group $G$.

We refer to [Ha, Ch. 8] for an amplification of the following facts about composition series and chief series. A *composition series* of $G$ is a maximal chain $1 = H_0 \lhd H_1 \lhd \cdots \lhd H_m = G$ of subgroups; then each quotient group $H_i/H_{i-1}$ is a simple group, and is called a *composition factor* of $G$; the isomorphism types in the multiset $\{H_i/H_{i-1} \mid 1 \leq i \leq m\}$ are uniquely determined by $G$. A *chief series* of $G$ is a maximal chain $1 = K_0 \lhd K_1 \lhd \cdots \lhd K_r = G$ of normal (in $G$) subgroups; the isomorphism types in the multiset $\{K_i/K_{i-1} \mid 1 \leq i \leq r\}$ are uniquely determined by $G$ (even as groups with operators $G$).

If $\Sigma$ is any collection of simple groups, let $O_\Sigma(G)$ denote the largest normal subgroup of $G$ each of whose composition factors is isomorphic to a member of $\Sigma$, and let $O^\Sigma(G)$ denote the smallest normal subgroup of $G$ such that each composition factor of $G/O^\Sigma(G)$ is isomorphic to a member of $\Sigma$. If $\Sigma$ consists of all the groups

of prime order, then $O_\Sigma(G)$ is the maximal solvable normal subgroup of $G$, while $O^\Sigma(G)$ is the last term in the derived series of $G$. If $\Sigma$ consists of a single group of prime order $p$, the group $O_p(G)$ is the largest normal $p$-subgroup of $G$; the subgroup $\langle O_p(G) \mid p \,||G|\rangle$ is called the *Fitting* subgroup of $G$ and is the largest normal nilpotent subgroup of $G$.

The automorphism group of a group $G$ is denoted $\mathrm{Aut}(G)$. Then $H$ is a *characteristic* subgroup of $G$ if it is mapped to itself by all elements of $\mathrm{Aut}(G)$. Examples include the groups in the derived series, the upper and lower central series, $O_\Sigma(G)$ and $O^\Sigma(G)$, for any $\Sigma$.

We denote by $\mathrm{Sym}(\Omega)$ the group of all permutations of an $n$-element set $\Omega$, or by $\mathrm{Sym}(n)$ if the set does require explication. For $\Delta \subseteq \Omega, g \in \mathrm{Sym}(\Omega)$, we denote by $\Delta^g$ the image of of $\Delta$ under $g$. The group of $n \times n$ nonsingular matrices over a $q$-element field is denoted by $\mathrm{GL}(n,q)$.

For any fixed integer $d$, let $\Gamma_d$ designate the class of groups all of whose noncyclic composition factors are isomorphic to a subgroup of $\mathrm{Sym}(d)$; in particular, $\Gamma_d$ *contains all solvable groups.* A most significant effect of this restriction on a class of groups is that the primitive permutation groups (see [Wi]) in the class have polynomially-bounded order [BCP]. (Primitive groups arise naturally as the base cases in certain divide-and-conquer procedures; see, e.g., [Lu1]). There are fairly elementary procedures for testing membership in $\Gamma_d$ (see [Lu1, §4]). For our purposes, it is essential only that $d$ be fixed; the specific value of $d$ would play a role in more precise timing arguments [Ba2], [BL], [BKL]). The class $\Gamma_d$ arose originally in the context of testing graph isomorphism ([Lu1], [Ba2], [Mi1], [Mi2], [BL], [FSS]).

## 3. Algorithmic preliminaries

Unless indicated otherwise, subgroups of $\mathrm{Sym}(n) = \mathrm{Sym}(\Omega)$ are input via generators. Output of groups is always via generators. All procedures identifying elements or subgroups are constructive – i.e., computed via *straight-line programs*, in which each element is a product or inverse of previously constructed or input elements. Throughout this paper, all algorithms have polynomial-time worst-case complexity. Checking running time to this extent is straightforward. Keep in mind, for this, that any strictly decreasing sequence of subgroups of $\mathrm{Sym}(n)$ has polynomial length (the bound $\log n! = \mathrm{O}(n \log n)$ is an immediate consequence of Lagrange's Theorem [Ha]; for the sharper bound $3n - 2$, see [Ba4]).

In this section, we recall a few fundamental problems for which polynomial-time algorithms are known. For these, there is no reasonable corresponding problem for quotient groups as the underlying set is too involved in the actual statement of the problem (though this point is arguable for **3.1**).

*Given a group $G \leq \mathrm{Sym}(\Omega)$, each of the following problems is solvable in polynomial time.*

**3.1.** Given $h \in \mathrm{Sym}(\Omega)$, test whether $h \in G$. [FHL]

As a consequence, one can test whether a group $H$ is a subgroup of $G$ (applying membership tests to the generators). The basic methodology for **3.1** and **3.2** is due to Sims [Si].

**3.2.** Given $\Delta \subset \Omega$, find the *pointwise stabilizer of* $\Delta$ *in* $G$, i.e., $\{g \in G \mid \delta^g = \delta, \forall \delta \in \Delta\}$. [FHL]

**3.3.** Suppose that $G \in \Gamma_d$ (see §2). Given $\Delta \subset \Omega$, find the *set stabilizer of* $\Delta$ *in* $G$, i.e., $\{g \in G \mid \Delta^g = \Delta\}$. [Lu1]

## 4. A polynomial-time library

Let $K \trianglelefteq G \leq \mathrm{Sym}(n)$. In this section we list a number of problems for computing in $\mathbf{G} = G/K$. For the case $K = 1$, we believe that these present an overview of the polynomial-time toolkit. Of course, it is not feasible to list every polynomial-time result, but, to our knowledge, problems known to be in polynomial-time are fairly direct consequences of this list.

We always assume that generators are given for $G$ and $K$. Each element of $\mathbf{G}$ is specified by a single coset representative: elements of $\mathbf{G}$ are cosets of the form $Kg$ with $g \in G$. All subgroups of $\mathbf{G}$ are written using boldface type; they are specified by generators.

The following problems are listed **P1-P16**; in referring to an algorithm, it is convenient to use the label, **Pm**, of the corresponding problem. The various problems have been divided into three broad categories: **TOOLS**, **BUILDING BLOCKS**, and **CHARACTERISTIC SUBGROUPS**. The ordering of the list is not intended to reflect the order in which solutions have been obtained in the literature or are obtained in this paper.

*Given $\mathbf{G} = G/K$ for $K \trianglelefteq G \leq \mathrm{Sym}(\Omega)$, each of the following problems is solvable in polynomial-time.*

### TOOLS

**P1.** Find $|\mathbf{G}|$, the order of $\mathbf{G}$.

**P2.** (i) Find a generator-relator presentation for $\mathbf{G}$.
(ii) Given $\mathbf{G} = \langle M \rangle$, and a map $\pi: M \to H$, where $H$ is any group in which we are able to determine, in polynomial time, products and inverses of designated elements; decide whether or not $\pi$ is extendible to a homomorphism $\mathbf{G} \to H$.

**P3.** (i) Given $S \subseteq \mathbf{G}$, find the normal closure $\langle S^{\mathbf{G}} \rangle$.
(ii) Given $\mathbf{H} \leq \mathbf{G}$, test whether $\mathbf{H}$ is subnormal in $\mathbf{G}$; and, if so, find a sequence $\mathbf{H} = \mathbf{L}_0 \trianglelefteq \mathbf{L}_1 \trianglelefteq \cdots \trianglelefteq \mathbf{L}_m = \mathbf{G}$.

**P4.** Given $\mathbf{A}, \mathbf{B} \leq \mathbf{G}$, find $\mathbf{A} \cap \mathbf{B}$ in each of the following situations:

(i) $\mathbf{A}$ normalizes $\mathbf{B}$, or

(ii) more generally, $\mathbf{B}$ is subnormal in $\langle \mathbf{A}, \mathbf{B} \rangle$, or

(iii) $\mathbf{A} \in \Gamma_d$ (cf. §2).

**P5.** (i) Given $\mathbf{H} \leq \mathbf{G}$, find $\mathrm{Core}_{\mathbf{G}}(\mathbf{H})$.

(ii) More generally, for arbitrary $\mathbf{G} = G/K, \mathbf{H} = H/K$, find $\mathrm{Core}_{\mathbf{G}}(\mathbf{G} \cap \mathbf{H})$.

**P6.** Given $\mathbf{A}, \mathbf{B} \leq \mathbf{G}$ such that $\mathbf{A}$ normalizes $\mathbf{B}$.

(i) Find $\mathrm{C}_{\mathbf{A}}(\mathbf{B})$.

(ii) Given $g \in \mathbf{G}$ normalizing $\mathbf{A}$ and $\mathbf{B}$, determine whether there is some $a \in \mathbf{A}$ such that $b^a = b^g$ for all $b \in \mathbf{B}$; and if so, find such an $a \in \mathbf{A}$.

**P7.** Given $\mathbf{A} \leq \mathbf{G}$ with $\mathbf{A} \in \Gamma_d$.

(i) Given $\mathbf{B} \leq \mathbf{G}$, find $\mathrm{C}_{\mathbf{A}}(\mathbf{B})$.

(ii) Given $b_1, b_2 \in \mathbf{G}$, determine whether there is some $a \in \mathbf{A}$ such that $b_1^a = b_2$; and if so, find such an $a \in \mathbf{A}$.

**P8.** Suppose $\mathbf{G}$ is nilpotent.

(i) Given $\mathbf{B} \leq \mathbf{G}$, find $\mathrm{N}_{\mathbf{G}}(\mathbf{B})$.

(ii) Given $\mathbf{B}_1, \mathbf{B}_2 \leq \mathbf{G}$, determine whether or not $\mathbf{B}_1$ and $\mathbf{B}_2$ are conjugate in $\mathbf{G}$; and if so, find $g \in \mathbf{G}$ such that $\mathbf{B}_1^g = \mathbf{B}_2$.

**P9.** Compute the kernel of the homomorphism $\pi$ in each of the following situations:

(i) $\pi: \mathbf{G} \to H/L$, where $L \trianglelefteq H \leq \mathrm{Sym}(\Delta)$.

(ii) $\pi$ is an action of $\mathbf{G}$ on a permutation group $H \leq \mathrm{Sym}(\Delta)$, i.e., $\pi: \mathbf{G} \to \mathrm{Aut}(H)$ ($\mathbf{G}$ need not act on $\Delta$). More generally, let $H$ be a quotient of subgroups of $\mathrm{Sym}(\Delta)$.

(iii) $\pi$ is a linear representation of $\mathbf{G}$ over a finite field, i.e., $\pi: \mathbf{G} \to \mathrm{GL}(m, q)$.

## BUILDING BLOCKS

**P10.** (i) Test whether $\mathbf{G}$ is simple;

(ii) if it is not, find a proper normal subgroup $\mathbf{N}$, i.e., $\mathbf{1} < \mathbf{N} \trianglelefteq \mathbf{G}$.

(iii) Find a composition series $\mathbf{1} = \mathbf{H}_0 \trianglelefteq \mathbf{H}_1 \trianglelefteq \cdots \trianglelefteq \mathbf{H}_m = \mathbf{G}$ for $\mathbf{G}$, and find a faithful permutation representation for each of the composition factors $\mathbf{H}_i/\mathbf{H}_{i-1}$; specifically, a homomorphism $\pi_i: \mathbf{H}_i \to \mathrm{Sym}(\Delta_i)$ with $\mathrm{kernel}(\pi_i) = \mathbf{H}_{i-1}$ and $|\Delta_i| \leq |\Omega|$.

**P11.** (i) If $\mathbf{H} \trianglelefteq \mathbf{G}$, test whether $\mathbf{H}$ is a minimal normal subgroup;

(ii) if it is not, find $\mathbf{N} \trianglelefteq \mathbf{G}$ such that $\mathbf{1} < \mathbf{N} < \mathbf{H}$.

(iii) Find a chief series for $\mathbf{G}$.

**P12.** If $\mathbf{G}$ is simple, identify the isomorphism type of $\mathbf{G}$; that is, find the name of this simple group.

**P13.** (i) If $p$ is a prime, find a Sylow $p$-subgroup of $\mathbf{G}$ containing a given $p$-subgroup $\mathbf{P}$ of $\mathbf{G}$ ($\mathbf{P}$ could be $\mathbf{1}$).

(ii) Given Sylow $p$-subgroups $\mathbf{P}_1, \mathbf{P}_2$ of $\mathbf{G}$, find $g \in \mathbf{G}$ such that $\mathbf{P}_1^g = \mathbf{P}_2$.

(iii) Given a Sylow $p$-subgroup $\mathbf{P}$ of $\mathbf{L}$ where $\mathbf{L} \trianglelefteq \mathbf{G}$, find $\mathrm{N}_{\mathbf{G}}(\mathbf{P})$.

## CHARACTERISTIC SUBGROUPS

**P14.** Find the following subgroups of $\mathbf{G}$:

(i) the derived series (and hence, test whether or not $\mathbf{G}$ is solvable);

(ii) the lower central series (and hence, test whether or not $\mathbf{G}$ is nilpotent); and

(iii) the upper central series (in particular, find $Z(\mathbf{G})$).

**P15.** (i) Find the subgroup generated by all minimal normal subgroups of $\mathbf{G}$ (the *socle* of $\mathbf{G}$).

(ii) Find the intersection of all maximal normal subgroups of $\mathbf{G}$.

**P16.** For any collection $\Sigma$ of simple groups,

(i) find $O_\Sigma(\mathbf{G})$, and

(ii) find $O^\Sigma(\mathbf{G})$.

We discuss sources for the above and also indicate the situations where the extension from the case $K = 1$ is immediate.

**P1**: Finding $|G|$ is inherent in Sims's basic procedure ([Si]; see [FHL]) and the extension to $\mathbf{G}$ is trivial: $|G/K| = |G|/|K|$.

For $K = 1$, algorithms for **P2**(i) are standard (e.g., [Le]); an asymptotically fast implementation is given in [BLS2]. Extensions to general $K$ and procedures for **P2**(ii) follow easily from the nature of these presentations; see §12 for comments. Typical situations that we have in mind for $H$ in **P2**(ii): $H$ is input via a Cayley table; $H = \mathrm{Sym}(\Delta)$ for some listed set $\Delta$; $H = \mathrm{GL}(m, q)$ (in which case timings must be polynomial in $m$ and $\log q$); $H = \mathrm{Aut}(A)$ for some $A \leq \mathrm{Sym}(\Delta)$, for some listed set $\Delta$ ($\pi(M)$ being specified on generators of $A$, in which case we might also need to verify that $\pi(M) \subseteq \mathrm{Aut}(A)$). More generally, we may suppose $H$ is a *black box group* in the sense of [BS].

When $K = 1$, **P3**(i) is contained in [FHL]; the analogue for quotient groups is immediate since $\langle (H/K)^{G/K} \rangle = \langle H^G \rangle / K$. **P3**(ii) is an easy consequence of the observation that $\mathbf{H}$ is subnormal in $\mathbf{G}$ iff $\mathbf{H}$ is subnormal in $\langle \mathbf{H}^{\mathbf{G}} \rangle$.

When $K = 1$, **P4**(i) is an easy application of results in [FHL] (see [Ho], or [CFL] where it is directly reduced to **3.2**); the general case is an immediate consequence. In view of **P3**(ii), **P4**(ii) follows at once. For $K = 1$, **P4**(iii) is in [Lu1, §4.2]. The general case is solved herein, see §7. All parts of **P4** should be compared with the general problem INTERSECTION (see the Appendix), which is at least as hard as GRAPH-ISOMORPHISM (ISO) [Lu1].

Problem **P5** was previously open even for $K = 1$ (see [Ba3]). It is solved in §6.

For $K = 1$, problem **P6**(i) is solved in [Lu2], while **P7**(i) is a consequence of **P4**(iii) and the computability of $C_{\mathrm{Sym}(\Omega)}(B)$ (see, e.g., [CFL]). The general cases of **P6**(i), **P7**(i) are amongst the principal applications of the methods of this paper, see §§6,7. These problems should be compared with the general problem CENTRALIZER (see the Appendix), which again is at least as difficult as ISO. Methods for obtaining **P6**(ii), **P7**(ii) from **P6**(i), **P7**(i), respectively, are analogues of the familiar reduction of ISO to finding automorphism groups of graphs, see §11.

**P8**(i) and **P8**(ii) are dealt with in §§10,11; these are new results even for $K = 1$. General NORMALIZER (see the Appendix) clearly is at least as difficult as CENTRALIZER (cf. **P6**(i)).

In problem **P9**, we would typically expect the homomorphisms to be specified by images of generators (see **P2**(ii)). In this set of problems, it seems as if only the case $L = 1$ in (i) is immediate, for this case reduces to an application of **3.2**. **P9**(ii) and **P9**(iii) use the building blocks of the groups. These results will appear in [KL]. Note that **P9**(ii) is a generalization of **P6**(i).

**P10** is treated in [Lu2]. Although only the case $K = 1$ is dealt with explicitly, the general case is implicit in [Lu2, §4].

It is easy to reduce **P11**(i) to the case that **H** is a direct product of isomorphic simple groups, all conjugate in **G** (for, taking the smallest group $\mathbf{L} \neq \mathbf{1}$ in a composition series of **H**, we may assume that $\mathbf{H} = \langle \mathbf{L}^{\mathbf{G}} \rangle$). If then **H** is nonabelian, it is minimal normal in **G**. The interesting case then is when **H** is abelian. This has been resolved by Rónyai [Ró, §5.3] as an application of an elegant study of the "Building Blocks" in associative algebras. **P11**(ii) and **P11**(iii) follow easily.

**P12** appears in [Ka1]. The "name" refers to a standard naming of the finite simple groups (examples: "$Z_{97}$", "$A_{17}$", "$PSL(4, 19)$").

For $K = 1$, **P13**(i),(ii),(iii) are resolved in [Ka2], [Ka2], [Ka3], respectively. The general case is in §8.

For $K = 1$, **P14**(i) and **P14**(ii) are standard observations, both problems reducing to finding normal closures of sets of commutators (e.g., see [FHL] for **P14**(i)); the general case is an immediate consequence. On the other hand, **P14**(iii) is new even when $K = 1$ (cf. §6).

An algorithm for **P15**(i) is discussed in §9. Computation of the "abelian part" of the socle requires an application of Rónyai's work [Ró]. **P15**(ii) is implicit in [BLS1].

We assume in **P16** that $\Sigma$ is specified by a, possibly parametrized, list of names of groups. We outline methods for these problems in §9. **P16**(ii) is actually implicit in [BLS1], given the additional capability in **P12**. For $K = 1$, the special case $O_p(\mathbf{G})$ has been computed in [Ka2] and [Ne].

## 5. Two paradigms

We isolate two useful computational ideas. In each case we will not present an actual algorithm, but rather the outline of one. It is important to note that, while the methods themselves are based on elementary group theoretic facts, their implementation requires the Sylow machinery in the instances $K = 1$ of **P13**, which, at present, depend heavily on the classification of finite simple groups. Fortunately, properties of simple groups are not visibly involved in the specifications of this machinery, nor in its uses. For easy but striking examples of the use of these methods, see §§6,7.

**Sylow Method.**

*Problem*

*Input:* $G \leq \mathrm{Sym}(n)$, given via generators; $H \trianglelefteq G$ with $H$ specified only by a membership test.

*Find:* Generators for $H$.

*Method:* Reduce to the case in which G is a $p$-group as follows. For each prime $p$, $p \mid |G|$, find a Sylow $p$–subgroup $P$ of $G$ (using **P13**(i)), and then find generators for $P \cap H$.

Output $H$ as $\langle P \cap H \mid \text{one } P \text{ per prime } p \mid |G| \rangle$.

*Correctness*: Since $H \trianglelefteq G$, $P \cap H$ is a Sylow $p$-subgroup of $H$. □

Before turning to the next, equally elementary method, we recall the following standard fact concerning finite groups [Go, p.12]:

**Frattini argument.** *Let $P \leq K \trianglelefteq G$ with $P$ a Sylow subgroup of $K$. Then $G = K\mathrm{N}_G(P)$.*

This inspires the

**Frattini Method.**

*Problem*

*Input:* $K \trianglelefteq G \leq \mathrm{Sym}(n)$, given via generators, such that $G/K$ has some given isomorphism-invariant group-theoretic property; a subgroup $H$ of $G$ containing $K$, specified only by a membership test.

*Find:* Generators for $H$.

*Method*: Reduce to the case in which $K$ is nilpotent, as follows. If $K$ is not nilpotent, there is, for some prime $p$, a Sylow $p$-subgroup $P$ of $K$ (computed by **P13**(i)) that is not in normal $K$. Use **P13**(iii) to find $N_G(P)$ and $N_K(P)$. Recursively solve the problem for the triple $N_G(P), N_K(P), H \cap N_G(P)$, producing generators A for $H \cap N_G(P)$. Output $A$ along with generators for $K$.

*Correctness*: By the Frattini argument, $G = KN_G(P)$, so that $N_G(P)/N_K(P) \cong G/K$ has the given group-theoretic property. Moreover, $H = KN_H(P) = K(H \cap N_G(P)) = K\langle A\rangle$. For the timing, we observe that $N_G(P) < G$. □

The group-theoretic properties of $G/K$ we have in mind here include solvability or membership in $\Gamma_d$ (cf. §2). In view of the above reduction, the full group $G$ may be assumed to have the respective property.

## 6. Cores and centers

This section contains a simple but noteworthy use of the Sylow Method. We resolve **P5** and then apply the result to **P6**(i) and **P14**(iii).

Note that a polynomial-time computation of $\mathrm{Core}_G(H)$ would be an immediate consequence of a polynomial-time procedure for intersecting permutation groups. However, the latter seems out of reach (see the Appendix). Hence, a less direct approach will be required.

Recall first that intersections with $p$-groups are feasible (case $K = 1$ of **P4**(iii) [Lu1]). We also need to recall that it is easy to implement a normal-closure routine for $N = \langle S^G\rangle$ so as to return: (1) a set $T$ generating $N$ consisting entirely of conjugates (in $G$) of elements of $S$; and (2) for each $t \in T$, some $g \in G$ for which $t \in S^g$. For example, given $G = \langle M\rangle$, form $T$ as follows: after initializing $T := S$, repeatedly check (using **3.1**) whether there is some $t \in T, a \in M$ with $t^a \notin \langle T\rangle$, and, if so, add $t^a$ to $T$.

*Algorithm for* **P5**(ii). We may assume that $K = 1$. Note, for membership-testing, that, if $g \in G$, then $g \in \mathrm{Core}_G(G \cap H)$ iff $\langle g^G\rangle \leq H$. Use of the Sylow Method (§5) shows that it suffices, for each prime $p$, to take a Sylow $p$-subgroup $P$ of $G$ and find $P \cap \mathrm{Core}_G(G \cap H)$. For this:

**while** $(\langle P^G\rangle \not\leq H)$ **do**
**begin**
find $g \in G$ such that $P^g \not\leq H$
(* via the above normal-closure routine *);
$P := P \cap H^{g^{-1}}$
**end**;
**output** $P$.

*Correctness:* One needs only to observe that the value of $P \cap \mathrm{Core}_G(G \cap H)$ is a loop invariant and that, upon exit from the loop, $P \leq \mathrm{Core}_G(G \cap H)$. □

*Algorithm for* **P6**(i). Let $\mathbf{A} = A/K$, $\mathbf{B} = B/K$. View $G \times G$ as a permutation group on a $2n$-element set, the disjoint union of 2 copies of $\Omega$, in the natural manner (the first coordinate permuting only the first copy, the second coordinate permuting only the second). Note that $1 \times K \trianglelefteq G \times G$ and $1 \times B$ is normalized by $A \times A$.

Let $D$ be the diagonal group $\{(a, a) \mid a \in A\}$, and set $R := D(1 \times B)$ and $S := D(1 \times K)$.

Use **P5**(i) to find $\mathrm{Core}_R(S)$. Let $C$ be the group obtained by restricting $\mathrm{Core}_R(S)$ to the first $n$ points.

Output $C/K$ (i.e., output the set of $Kc$, $c$ ranging over the generators of $C$).

*Correctness:* We must show that $\mathrm{C}_{\mathbf{A}}(\mathbf{B}) = C/K$. Note that $C$ contains $K$ since $K \times K$ is a normal subgroup of $R$ contained in $S$. Observe, too, that $(a, x) \in S$ iff $a \in A$ and $a^{-1}x \in K$. Now, if $(a, x) \in S$, then

$(a, x) \in \mathrm{Core}_R(S)$
iff $(a, x)^R \subseteq S$
iff $(a, x)^{(1,b)} \in S$, $\forall b \in B$ (since $S^D = S$)
iff $a^{-1}x^b = a^{-1}a^b(a^{-1}x)^b \in K$, $\forall b \in B$
iff $a^b \equiv a \pmod K$, $\forall b \in B$. □

*Algorithm for* **P14**(iii). Successively use **P6**(i) and the definition of the upper central series.

The preceding method for finding $Z(\mathbf{G})$ is in stark contrast to the known algorithms for finding $Z(G)$. For example, in [CFL] this is found first by finding $\mathrm{C}_{\mathrm{Sym}(n)}(G)$; but no analogue of this is available for $\mathbf{G}$. Another method [Lu2] finds $Z(G)$ by first constructing a faithful permutation representation of $G/Z(G)$ on a set of size $O(n^2)$; evidently, such an approach cannot be iterated. Instead, we have made full use of (the case $K = 1$ of) the Sylow machinery which, in turn, depends on the classification of finite simple groups.

## 7. Computations with solvable groups

Despite the title, the algorithms in this section deal with a more general, though less standard, class of groups: we assume throughout that $\mathbf{G} \in \Gamma_d$, for some fixed $d$ (§2).

We present algorithms for **P4**(iii) and **P7**(i). The Frattini Method (§5) plays a critical role.

*Algorithm for* **P4**(iii). Let $\mathbf{A} = A/K$, $\mathbf{B} = B/K$. We seek $H = A \cap B$. If $K \in \Gamma_d$ then $A \in \Gamma_d$ so that this intersection can be found via the case $K = 1$ of **P4**(iii) [Lu1, §4.2]. Otherwise, $K$ is certainly not nilpotent, so that we can find, for some prime $p$, a Sylow $p$-subgroup $P$ of $K$ that is not normal in $K$ (using the case $K = 1$ of **P13**(i) [Ka2]). Use the case $K = 1$ of **P13**(iii) [Ka3] to find $N_A(P)$, $N_B(P)$, $N_K(P)$. Recursively, compute $N_A(P) \cap N_B(P)$. Output $H := [N_A(P) \cap N_B(P)]K$.

*Correctness:* Since $A$ normalizes $B$, $N_A(P)/N_K(P)$ normalizes $N_B(P)/N_K(P)$; also, $A = KN_A(P)$ by the

Frattini argument (§5), so that $N_A(P)/N_K(P) \cong A/K \in \Gamma_d$. Hence, the recursive call is valid. By the Frattini argument, $H = N_H(P)K = (N_A(P) \cap N_B(P))K$. For the timing, observe that $N_A(P) < A$. □

In the following, we denote, for any $r \in \mathrm{Sym}(\Omega)$, $\Delta_r = \{(\omega, \omega^r) \mid \omega \in \Omega\} \subset \Omega \times \Omega$ (the "graph" of $r$), so that $\Delta_r = \Delta_s$ iff $r = s$. Considering the natural action of $\mathrm{Sym}(\Omega) \times \mathrm{Sym}(\Omega)$ on $\Omega \times \Omega$ (i.e., via $(\alpha, \beta)^{(g,h)} = (\alpha^g, \beta^h)$), we note that $\Delta_r^{(g,h)} = \Delta_{g^{-1}rh}$.

*Algorithm for* **P7**(i). We may assume that $\mathbf{B} = \langle Kb \rangle$ is cyclic. The Frattini Method reduces the problem to the case $K$ nilpotent. In particular, we may assume $A \in \Gamma_d$.

Let $\mathbf{A} = A/K$. Letting $D$ be the diagonal subgroup $\{(a,a) \mid a \in A\}$ (so $D \cong A$), set $L := D(1 \times K) < \mathrm{Sym}(\Omega) \times \mathrm{Sym}(\Omega)$. Then $L \in \Gamma_d$.

Use **3.3** to find the set stabilizer $S$ of $\Delta_b$ in $L$.

Let $H$ be the first-coordinate projection of $S$ (generated by the first-coordinate projections of the generators of $S$). Output $H$.

*Correctness:* For $a \in A$ and $k \in K$, $\Delta_b^{(a,ak)} = \Delta_{a^{-1}bak}$. Thus $(a, ak)$ stabilizes $\Delta_b$ iff $b^a = bk^{-1}$. Hence, for $a \in A$, there is some $(a, ak) \in L$ stabilizing $\Delta_b$ iff $Ka \in C_{\mathbf{A}}(\mathbf{B})$. □

## 8. Sylow subgroups

We indicate algorithms for **P13** that are easy extensions of the case $K = 1$ [Ka2], [Ka3].

*Algorithm for* **P13**(i). Let $\mathbf{P} = P/K$. Use the case $K = 1$ of **P13**(i) to find a Sylow $p$-subgroup $Q$ of $P$ and to find a Sylow $p$-subgroup $R$ of $G$ containing $Q$. Then $RK/K$ is a Sylow $p$-subgroup of $G/K$ containing $P/K$.

*Algorithm for* **P13**(ii). Let the given Sylow $p$-subgroups be $\mathbf{P}_i = P_i/K$, $i = 1, 2$. Use the cases $K = 1$ of **P13**(i) and **P13**(ii) to find Sylow $p$-subgroups $R_1$ and $R_2$ of $G$ lying in $P_1$ and $P_2$, respectively, and to find $g \in G$ such that $R_1^g = R_2$. Then $\mathbf{P}_1^g = \mathbf{P}_2$.

*Algorithm for* **P13**(iii). Let $\mathbf{L} = L/K$ and $\mathbf{P} = P/K$.

Use the case $K = 1$ of **P13**(i) to find a Sylow $p$-subgroup $R$ of $P$. (Then $R$ is also Sylow in $L$, and $P = RK$.)

Use the case $K = 1$ of **P13**(iii) to find $N_G(R)$.

Then $N_{\mathbf{G}}(\mathbf{P}) = N_G(R)K/K$. (For, by the Frattini argument (cf. §5) applied to the triple $R \leq P \trianglelefteq N_G(P)$, we have $N_G(P) = PN_{N_G(P)}(R) = RKN_{N_G(P)}(R) \leq KN_G(R) \leq KN_G(P)$. But $N_{\mathbf{G}}(\mathbf{P}) = N_G(P)/K$.)

One consequence of **P13**(i) is an

*Algorithm for finding* $O_p(\mathbf{G})$. This uses the fact that $O_p(\mathbf{G}) = \mathrm{Core}_{\mathbf{G}}(\mathbf{P})$ for any Sylow $p$-subgroup $\mathbf{P}$ of $\mathbf{G}$. Note that this special case of **P5**(i) can also be computed by successively intersecting conjugates of $\mathbf{P}$ using **P4**(iii).

**Remarks.** (i) There are more elementary (and more practical) methods for computing $O_p(G)$ (case $K = 1$); in particular, these do not depend upon the classification of finite simple groups ([Ne], also [KL]).

(ii) As in [Ka2], all of these results can be extended to Hall subgroups either of $\mathbf{G}$ or of its normal subgroups.

## 9. Socles and other normal subgroups

Next we turn to **P15** and **P16**.

$\mathrm{Soc}(H)$, the socle of $H$, is the direct product of simple subgroups, and hence can be written

$$\mathrm{Soc}(H) = \mathrm{Soc}(H)' \times \prod_{p \mid |H|} \mathrm{Soc}_p(H)$$

where $\mathrm{Soc}(H)'$ is generated by the nonabelian minimal normal subgroups of $H$, while $\mathrm{Soc}_p(H)$ is generated by the minimal normal $p$-subgroups of $H$ and is elementary abelian.

A technique for computing the "nonabelian part" of the socle, $\mathrm{Soc}(H)'$, is indicated in [BKL, §5]; it is stated only for $K = 1$, but given **P6**(i), the technique extends to the general case. We now indicate an

*Algorithm for finding* $\mathrm{Soc}_p(\mathbf{G})$, where $p$ is any prime divisor of $|\mathbf{G}|$.

Find $O_p(\mathbf{G})$ (see the end of §8).

Use **P14**(iii) to find $Z(O_p(\mathbf{G}))$.

We may assume that $Z(O_p(\mathbf{G})) \neq 1$.

Find the elementary abelian $p$-group $V$ generated by all elements of order $p$ in $Z(O_p(\mathbf{G}))$. (Namely, for each generator $d$ of $Z(O_p(\mathbf{G}))$, take an element $d'$ in $\langle d \rangle$ of order $p$; then $V$ is generated by these elements $d'$.) Clearly, $\mathrm{Soc}_p(\mathbf{G}) \leq V$. Since $V$ is a vector space over $GF(p)$, we can use [Ró] as follows.

Each generator of $\mathbf{G}$ induces (by conjugation) a linear transformation of $V$, whose matrix with respect to a basis of $V$ can be found using **3.1** and linear algebra. Let $A$ be the algebra generated by these linear transformations of $V$. Then the minimal normal subgroups of $\mathbf{G}$ lying in $V$ are precisely the $A$-irreducible subspaces, so that $\mathrm{Soc}_p(\mathbf{G})$ is the span of these. This space is found using [Ró].

*Algorithm for* **P16**(i). Use **P15**(i) to find $\mathrm{Soc}(\mathbf{G})$.

Use **P12** to test whether any member of $\Sigma$ is isomorphic to a minimal subnormal subgroup of $\mathbf{G}$ (i.e., a simple factor of $\mathrm{Soc}(\mathbf{G})$). If not, output 1.

We may assume that some member of $\Sigma$ is isomorphic to a minimal subnormal subgroup $\mathbf{J}$ of $\mathbf{G}$. Use **P3**(i) to find $\mathbf{L} = \langle \mathbf{J}^{\mathbf{G}} \rangle$.

Recursively find $\mathbf{S}/\mathbf{L} = O_\Sigma(\mathbf{G}/\mathbf{L})$.

Output $\mathbf{S}$. Correctness is immediate.

*Algorithm for* **P16**(ii). Find $\mathbf{G}/\mathbf{G}'$ and $|\mathbf{G}/\mathbf{G}'|$ using **P14**(i) and **P1**.

If $\Sigma$ contains a cyclic group whose order, $p$, divides that of the abelian group $\mathbf{G}/\mathbf{G}'$, find a maximal normal subgroup $\mathbf{M}$ of $\mathbf{G}$ such that $|\mathbf{G}/\mathbf{M}| = p$.

If $\Sigma$ contains no such cyclic group, use the algorithm for **P15**(ii) [BLS1], which lists all maximal normal subgroups $\mathbf{M}$ of $\mathbf{G}$ such that $\mathbf{G}/\mathbf{M}$ is simple and nonabelian, and then use **P12** to test whether or not any such $\mathbf{G}/\mathbf{M}$ is isomorphic to a member of $\Sigma$.

If a maximal normal subgroup $\mathbf{M}$ of $\mathbf{G}$ has been found such that $\mathbf{G}/\mathbf{M}$ is isomorphic to a member of $\Sigma$, then output $O^\Sigma(\mathbf{G}) = O^\Sigma(\mathbf{M})$. Else output $O^\Sigma(\mathbf{G}) = \mathbf{G}$.

*Correctness*: If $O^\Sigma(\mathbf{G}) < \mathbf{G}$ then, by definition, $\mathbf{G}$ has a homomorphic image $\mathbf{G}/\mathbf{M}$ isomorphic to a member of $\Sigma$. Then all composition factors of $\mathbf{G}/O^\Sigma(\mathbf{M})$ are isomorphic to members of $\Sigma$, so that $O^\Sigma(\mathbf{M}) \geq O^\Sigma(\mathbf{G})$. Then $\mathbf{M} \geq O^\Sigma(\mathbf{G})$, and all composition factors of $\mathbf{M}/O^\Sigma(\mathbf{G})$ are isomorphic to members of $\Sigma$, so that we also have $O^\Sigma(\mathbf{M}) \leq O^\Sigma(\mathbf{G})$. □

## 10. Normalizers in nilpotent groups

We turn to a special case of the problem NORMALIZER (see the Appendix).

*Algorithm for* **P8**(i). *Case* 1. $|\mathbf{B}|$ *is prime.* Let $\mathbf{L} \trianglelefteq \mathbf{G}$ with $|\mathbf{L}|$ prime (i.e., $\mathbf{L}$ is a subgroup of order $p$ in $Z(\mathbf{G})$, found using **P14**(iii)). Let $\overline{\ }$ denote the natural homomorphism $\mathbf{G} \to \mathbf{G}/\mathbf{L}$.

Recursively find a group $\mathbf{H}$ such that $\mathbf{L} \leq \mathbf{H} \leq \mathbf{G}$ and $\overline{\mathbf{H}} = \mathrm{N}_{\overline{\mathbf{G}}}(\overline{\mathbf{B}})$. (Then $\mathrm{N}_{\mathbf{H}}(\mathbf{B}) = \mathrm{N}_{\mathbf{G}}(\mathbf{B})$: if $b \in \mathbf{G}$ normalizes $\mathbf{B}$ then $\overline{b}$ normalizes $\overline{\mathbf{B}}$, so that $\overline{b} = \mathbf{L}b \in \mathbf{H}/\mathbf{L}$ and hence $b \in \mathbf{H}$. Note that $\mathbf{H}$ acts on the abelian group $\mathbf{BL}$, which equals $\mathbf{L}$ or $\mathbf{B} \times \mathbf{L}$ and has order a prime or the product of two primes, so that $|\mathbf{BL}| \leq n^2$.)

Output $\mathrm{N}_{\mathbf{H}}(\mathbf{B})$ (as the stabilizer of $\mathbf{B}$ in the action of $\mathbf{H}$ on a small set: the set of subgroups of $\mathbf{BL}$ of order $|\mathbf{B}|$, where this set has size 1 or $1 + |\mathbf{B}|$).

*Case* 2. *Arbitrary* $|\mathbf{B}|$. Let $\mathbf{G} = \mathbf{G}_0 > \mathbf{G}_1 > \ldots$ be a normal series of $\mathbf{G}$ each of whose quotients is cyclic of prime order (this is just a chief series of our nilpotent group).

Find $i$ with $\mathbf{B} \not\leq \mathbf{G}_{i+1}, \mathbf{B} \leq \mathbf{G}_i$. Find $\mathbf{J} := \mathbf{B} \cap \mathbf{G}_{i+1}$.

Recursively find $\mathbf{H} = \mathrm{N}_{\mathbf{G}}(\mathbf{J})$. (Then $\mathbf{H} \geq \mathrm{N}_{\mathbf{G}}(\mathbf{B})$, so that $\mathrm{N}_{\mathbf{H}}(\mathbf{B}) = \mathrm{N}_{\mathbf{G}}(\mathbf{B})$; and $\mathbf{G}_i = \mathbf{B}\mathbf{G}_{i+1}$, so that $|\mathbf{B}/\mathbf{J}| = |\mathbf{G}_i/\mathbf{G}_{i+1}|$ is prime.)

Now use Case 1 to find $\mathrm{N}_{\mathbf{H}/\mathbf{J}}(\mathbf{B}/\mathbf{J}) = \mathrm{N}_{\mathbf{H}}(\mathbf{B})/\mathbf{J}$.

## 11. Conjugacy

Centralizer problems, such as **P6**(i) or **P7**(i), and normalizer problems, such as **P8**(i), can be thought of as finding stabilizers of "points" in some other action of $\mathbf{G}$. There is a corresponding question of determining all the elements of a group that carry a "point" to another "point". The relation between the problems is much like that between the problems of finding automorphism groups of objects (such as graphs) and testing isomorphism. Indeed, the reduction of testing graph isomorphism to finding automorphism groups (see, e.g., [Lu1]) has analogues here. We will indicate reductions of **P6**(ii), **P7**(ii), **P8**(ii), to **P6**(i), **P7**(i), **P8**(i), respectively. We remark, however, that an alternative to these reductions is a reformulation, and generalization, of the actual algorithms for **P6**(i), **P7**(i), **P8**(i), to produce algorithms to find the full collection of $a \in \mathbf{G}$ performing the conjugations in **P6**(ii), **P7**(ii), **P8**(ii), respectively; this collection is either $\emptyset$ or a coset of a subgroup of $\mathbf{G}$. This approach is similar to that for **3.3** given in [Lu1], where, by replacing the input $G$ by a coset $Gh$, one, in effect, finds the elements of $G$ that map $\Delta$ to $\Delta^{h^{-1}}$.

*Algorithm for* **P6**(i). Using **P6**(ii), find $\mathbf{L} := \mathrm{C}_{\langle g\rangle\mathbf{A}}(\mathbf{B})$ ($\langle g\rangle\mathbf{A}$ is a group since $g$ normalizes $\mathbf{A}$). Test whether $g \in \mathbf{LA}$. If, it is, find a factorization $g = la$, $l \in \mathbf{L}$, $a \in \mathbf{A}$ (this is straightforward [BLS1, §7]), and output $a$.

*Correctness:* If $a \in \mathbf{A}$ then $b^g = b^a, \forall b \in \mathbf{B}$, iff $ga^{-1} \in \mathbf{L}$. □

Recall that $\mathrm{Sym}(\Omega) \times \mathrm{Sym}(\Omega)$ acts naturally on the disjoint union $\Omega_1 \dot\cup \Omega_2$ of two copies $\Omega_1$, $\Omega_2$ of $\Omega$. Define $t \in \mathrm{Sym}(\Omega_1 \dot\cup \Omega_2)$ by $\omega_1^t = \omega_2$ and $\omega_2^t = \omega_1$, for all $\omega \in \Omega$.

*Algorithm for* **P7**(ii). Suppose $b_1 = Ks_1$ and $b_2 = Ks_2$. Form the following subgroups of $\mathrm{Sym}(\Omega) \times \mathrm{Sym}(\Omega)$: $\widehat{B} := \langle (s_1, s_2)\rangle$, $\widehat{K} := K \times K$, $\widehat{A} := \langle t\rangle(A \times A)$ (so $\widehat{A}$ is the *wreath product* $A \wr Z_2$ acting naturally on $\Omega_1 \dot\cup \Omega_2$ [Ha, p. 81]).

Use **P7**(i) to find $\widehat{H}/\widehat{K} := \mathrm{C}_{\widehat{A}/\widehat{K}}(\widehat{B}/\widehat{K})$. If some generator $h$ of $\widehat{H}$ switches $\Omega_1$ and $\Omega_2$, then $th$ fixes $\Omega_1$ and hence induces a permutation $r$ on $\Omega$, where $r \in A$; in this case, output $a := Kr$. Else output "no such $a$ exists".

*Correctness:* Note that $\widehat{A}$ and $\widehat{B}$ normalize $\widehat{K}$, and $\widehat{A}/\widehat{K} \cong \mathbf{A} \wr Z_2 \in \Gamma_d$; hence the use of **P7**(i) is valid. An element $t(r_1, r_2) \in t(A \times A)$ centralizes $(s_1, s_2) (\mathrm{mod}\, \widehat{K})$ iff $b_1^{Kr_1} = b_2$ and $b_2^{Kr_2} = b_1$, so an output of the form $Kr$ satisfies the requirement for $a$. On the other hand, if $Kr \in \mathbf{A}$ satisfies $b_1^{Kr} = b_2$ then $t(r, r^{-1})$ is in $\widehat{H}$ and it switches $\Omega_1$ and $\Omega_2$; and hence some generator of $\widehat{H}$ must switch $\Omega_1$ and $\Omega_2$. □

A reduction of **P8**(ii) to **P8**(i) can be constructed along the lines of that from **P7**(ii) to **P7**(i). The only tricky part is to maintain nilpotence in the wreath product construction: in general, nilpotence of $N$ does not imply that of $N \wr Z_2$. To avoid this problem, first reduce to the $p$-group case ($\mathbf{G}$, $\mathbf{B}_1$, $\mathbf{B}_2$ are direct products

of $p$-groups); then, to maintain $p$-groups, utilize wreath products with $Z_p$ in the construction.

## 12. Presentations

We comment briefly on **P2**. This material is essentially folklore.

Given an algorithm for the $K = 1$ case of **P2**(i), there is an obvious approach to the general case. For, suppose $G \cong \langle X \mid R \rangle$. To obtain a presentation for **G**: express generators of $K$ in terms of the image $\hat{X}$ of $X$ in $G$ and pull these expressions back to words in $X$; augment $R$ by the words so obtained. But, in order to make use of this approach, it is necessary that elements of $G$ be expressible as short words in $\hat{X}$. In fact, known algorithms for finding presentations of $G$ admit this facility ([BLS1], [Le]), $\hat{X}$ appearing as a "strong generating set" of $G$.

An algorithm for **P2**(ii) is a consequence of the straight-line (§3) construction of a presentation for **P2**(i). Duplicate the straight-line construction of $\hat{X}$ from the generators $M$ of $G$ in a straight-line program program starting with $\pi(M)$. This produces a map $\pi' : \hat{X} \to H$. Next, one verifies that the relations, which are words in $X$, are satisfied in the corresponding set $\pi'(\hat{X})$. This guarantees that the map $\hat{X} \to \pi'(\hat{X})$ extends to a homomorphism $\pi' : \mathbf{G} \to H$. Finally, it is necessary to verify that $\pi'$ agrees with the original input on $M$; namely, express each $a \in M$ as a word in $\hat{X}$ and check that the corresponding word in $\pi'(\hat{X})$ evaluates to $\pi(a)$.

## 13. Some open questions

We indicate some favorites from the questions inspired by these investigations.

1. SIZE OF REPRESENTATION DOMAIN

*Input:* $N \trianglelefteq G \leq \mathrm{Sym}(n)$, integer $m$.

*Question:* Is $G/N$ isomorphic to a subgroup of $\mathrm{Sym}(m)$?

The problem is in NP. This is easy to see if $m$ is entered in unary. A more general verification uses **P5**(i). We suspect that an efficient deterministic algorithm would require new mathematical tools. But what about special classes of groups? The problem is easily in P if $G/K$ is abelian. What about nilpotent groups?

2. EXTENDIBILITY OF HOMOMORPHISM

*Input:* $M \subseteq G \leq \mathrm{Sym}(n)$; a map $\pi : M \to H$ for some group $H$ (cf. **P2**(ii)).

*Question:* Is $\pi$ extendible to a homomorphism $\pi : G \to H$?

The problem is clearly in NP (using **P2**(ii)). Again, one should probably start with special cases. What about $G$ abelian, where $H$ is, say, a permutation group? Even the special case when $G$ is cyclic leads to the interesting question: given $h \in H$ and an integer $m$ (in unary); is $h = k^m$ for some $k \in H$?

3. INNER AUTOMORPHISM

*Input:* $G \leq \mathrm{Sym}(n)$; $\pi \in \mathrm{Aut}(G)$.

*Question:* Is there a $g \in G$ such that $\pi(a) = a^g$, for all $a \in G$.

By **P2**(ii), $\pi$ may be specified on generators. Here, too, the problem is in NP. Note that this is a generalization of **P7**(ii).

Other problems that arise naturally have to do with extension of the techniques herein when they are presently restricted to special classes of groups. Extend the $\Gamma_d$ hypothesis in **3.3**, **P4**(iii), **P7**; find normalizers in, say, solvable groups (cf. **P8**). These questions are strongly motivated by GRAPH-ISOMORPHISM (see the Appendix).

Finally, it still seems worth seeking *elementary* solutions to some of the elementary-sounding problems. Examples: (1) Should it really be necessary to invoke the classification of finite simple groups just to find elements of prime order $p$ in a permutation group $G$ where $p \,|\, |G|$? (2) Problems **P5**, **P6** may have a more direct approach. Note that there is an elementary algorithm for finding the cores of set-stabilizers [Lu2, §3].

## Appendix. Hard problems?

We highlight a set of problems, which are related to some of those discussed herein, but which seem unlikely to have polynomial-time solutions. This is suggested by the fact that these are at least as hard as GRAPH-ISOMORPHISM (ISO), the problem of testing whether two graphs are isomorphic. In practice, ISO is not a hard problem (e.g., see [McK]). Indeed, on average over all graphs, and even over regular graphs, isomorphism is known to be testable in linear time [BK], [Ku]. Furthermore, there is strong evidence that ISO is not NP-complete, else the polynomial-time hierarchy would collapse to $\Sigma_2^p = \Pi_2^p = \mathrm{AM}$ ([GMW]). Nevertheless, ISO has stubbornly resisted attempts to place it in polynomial-time. (At present the best algorithm for general graphs has worst-case complexity $\exp(c\sqrt{n \log n})$ [BKL].)

Consider now the following problems for permutation groups.

I. SET-STABILIZER (STAB)

*Input:* $G \leq \mathrm{Sym}(\Omega)$; $\Delta \subseteq \Omega$.

*Find:* $\mathrm{Stab}_G(\Delta) = \{g \in G \mid \Delta^g = \Delta\}$.

II. INTERSECTION (INTER)

*Input:* $G, H \leq \mathrm{Sym}(\Omega)$.

*Find:* $G \cap H$.

**III.** CENTRALIZER (CENT)

*Input:* $G, H \leq \mathrm{Sym}(\Omega)$.

*Find:* $\mathrm{C}_G(H)$.

**IV.** LARGEST NORMALIZED SUBGROUP (LNS)

*Input:* $G, H \leq \mathrm{Sym}(\Omega)$.

*Find:* $\mathrm{LNS}(H;G) = \bigcap\{H^g \mid g \in G\}$, the largest subgroup of $H$ that is normalized by $G$.

**V.** RELATIVE-CENTRALIZER (REL_CENT)

*Input:* $K \trianglelefteq G \leq \mathrm{Sym}(\Omega), B \subseteq \mathrm{Sym}(\Omega)$.

*Find:* $\mathrm{C}_G(B,K) = \{g \in G \mid (Kb)^g = Kb, \forall b \in B\}$.

**Remarks.**

(i) Note in REL_CENT that we do not assume that $B$ normalizes $K$.

(ii) **IV** should be compared with **P5**(ii), finding the largest subgroup of $H$ that is *in* $G$ and is normalized by $G$.

It is well known that ISO $\propto$ STAB (we use "$\propto$" to denote polynomial-time-Turing-reduction); see, e.g., [Lu1]. But, in fact, STAB is equivalent, with respect to polynomial-time reduction, to each of problems **II-V**.

Suppose we are given an instance $G, \Omega, \Delta$ of STAB. Let $G$ act in the diagonal on the disjoint union $\widehat{\Omega} = \Omega_1 \dot{\cup} \Omega_2$ of two copies of $\Omega$ (i.e., $(\omega_i)^g = \omega_i^g$, $\forall \omega \in \Omega$, $i = 1,2$, $\forall g \in G$). Let $a$ be the involution in $\mathrm{Sym}(\widehat{\Omega})$ specified by: $\omega_i^a =$ **if** $\omega \in \Delta$ **then** $\omega_{3-i}$ **else** $\omega_i$ for $i = 1,2$; and set $H := \langle a \rangle$, $B := \{a\}$ and $K := G$. Observe that $\mathrm{Stab}_G(\Delta) = G \cap G^a = \mathrm{C}_G(H) = \mathrm{C}_G(a) = \mathrm{LNS}(H;G) = \mathrm{C}_G(B,K)$, thus reducing **I** respectively to **II, III, IV, V**.

*Reductions in the other direction:*

INTERS $\propto$ STAB: Let $G \times H$ act on $\Omega \times \Omega$ in the natural way, and set $\Delta := \{(\omega,\omega) \mid \omega \in \Omega\}$. Then $\mathrm{Stab}_{G\times H}(\Delta) = \{(g,g) | g \in G \cap H\}$.

CENT $\propto$ STAB: $g \in G$ commutes with $h$ iff $g$, acting diagonally on $\Omega \times \Omega$, stabilizes $\{(\omega, \omega^h) \mid \omega \in \Omega\}$.

LNS $\propto$ INTERS: $L := H$; while $L$ is not normalized by $G$, intersect $L$ with its conjugates by the generators of $G$.

REL_CENT $\propto$ STAB: This reduction is implicit in the algorithm for **P5**(i) (§7).

**Variations.**

(i) Problems **II-V** also can be stated for quotient groups. In keeping with the Quotient Group Thesis, it is worth noting that each quotient-group problem remains polynomial-time equivalent to STAB. This is obvious for **II**, **IV**, **V**, where the quotient-group statement is interpretable as an instance of the same problem. **III** gets absorbed into **V**.

(ii) The indicated reduction shows that **V** remains hard even for $G/K \in \Gamma_d$, in fact, for $G = K$.. It would seem this hypothesis puts the problem tantalizingly close to **P7**(i). But note Remark (i). In the algorithm for **P7**(i), the hypothesis that $b$ normalizes $K$ is only needed in the Frattini reduction to the case $A \in \Gamma_d$. In fact, the algorithm shows that REL_CENT is in P if $G \in \Gamma_d$.

(iii) **III** remains "hard" even if $H$ is subnormal in $\langle G, H \rangle$ (compare with **P4**(ii)). For, in the earlier reduction, for each $\omega \in \Omega$, let $b_\omega$ be the involution of $\widehat{\Omega}$ that switches $\omega_1 \in \Omega_1$ with its counterpart $\omega_2 \in \Omega_2$, leaving everything else fixed; let $B = \langle b_\omega \mid \omega \in \Omega \rangle$. Then $G$ normalizes $B$ and $H \trianglelefteq B \trianglelefteq GB$. Find $C := \mathrm{C}_{GB}(H)$; it is an easy matter to construct the projection $\pi: GB \to G$. Then $\pi(C) = \mathrm{Stab}_G(\Delta)$.

We mention also one other "hard" problem.

**VI.** NORMALIZER (NORM)

*Input:* $G, H \leq \mathrm{Sym}(\Omega)$.

*Find:* $\mathrm{N}_G(H)$.

STAB $\propto$ NORM; in fact, in the above reductions from STAB, $\mathrm{Stab}_G(\Delta) = \mathrm{N}_G(H)$. We conjecture that there is a reverse reduction.

Another analogy with ISO puts some perspective on the "hardness" of these problems. In 1980, the second author had observed that **II** is polynomial-time-Turing-equivalent to the decision problem of testing nonemptiness of coset intersection (COS_INTERS): *Is* $Ga \cap Hb$ *nonempty?* In fact, this equivalence is analogous to that between finding graph automorphism-groups and ISO. The analogy has been reinforced by Babai and Moran, who show that the NP-completeness of COS_INTERS would imply the same collapse $\Sigma_2^p = \Pi_2^p = \mathrm{AM}$ ([BM]).

We remark, finally, that problems such as **I-VI** are not considered difficult in practical computation, and systems such as CAYLEY [Ca] allow quite efficient implementations. This should be no surprise, considering the ease with which ISO is handled in practice.

## References


[Ba1] L. Babai, *Monte Carlo algorithms for graph isomorphism testing,* Tech. Rep. 79-10, Dép. Math. et Stat., Univ. de Montréal 1979.

[Ba2] L. Babai, *Moderately exponential bound for graph isomorphism,* Proc. Conf. FCT 1981, Szeged, Springer Lect. Notes in Comp. Sci. 117 (1981), 34–50.

[Ba3] L. Babai, *A Las Vegas-NC algorithm for isomorphism of graphs with bounded multiplicity of eigenvalues,* Proc. 27th IEEE FOCS, 1986, 303–312.

[Ba4] L. Babai, *On the length of subgroup chains in the symmetric group,* Comm. in Alg. 14 (1986), 1729–1736.

[BCP] L. Babai, P. Cameron and P.J. Pálfy, *On the order of primitive groups with restricted nonabelian composition factors,* J. Algebra 79 (1982), 161–168.

[BK] L. Babai and L. Kučera, *Canonical labelling of graphs in linear average time,* Proc. 20th IEEE FOCS, 1979, 49–46.

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

[BL] L. Babai and E.M. Luks, *Canonical labeling of graphs,* Proc. 15th ACM STOC, 1983, 171-183.

[BLS1] L. Babai, E.M. Luks and Á. Seress, *Permutation groups in NC,* Proc. 19th ACM STOC, 1987, 409-420.

[BLS2] L. Babai, E.M. Luks and Á. Seress, *Fast management of permutation groups,* Proc. 29th IEEE FOCS, 1988, 272–282.

[BM] L. Babai and S. Moran, *Arthur-Merlin games: A randomized proof system and a hierarchy of complexity classes.* J. Comp. Sys. Sci.36 (1988), 254–276.

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

[Ca] J.J. Cannon, *An introduction to the group theory language Cayley,* in Computational Group Theory (ed. M. D. Atkinson), Academic Press 1984, 145–183.

[CFL] G. Cooperman, L. Finkelstein and E. Luks, *Reduction of group constructions to point stabilizers,* Proc 1989 ACM-SIGSAM Inter. Symp. on Symbolic and Algebraic Comp., (1989), 351-356.

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

[FSS] M. Fürer, W. Schnyder and E. Specker, *Normal forms for trivalent graphs and graphs of bounded valence,* Proc. 15th ACM STOC, 1983, 161–170.

[GMW] O. Goldreich, S. Micali and A. Wigderson, *Proofs that yield nothing but their validity and a methodology of cryptographic protocol design,* Proc. 27th IEEE FOCS, 1986, 174–187.

[Go] D. Gorenstein, Finite Groups. Harper and Row, New York 1968.

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

[Ho] C.M. Hoffmann, Group Theoretic Algorithms and Graph Isomorphism, Lect. Notes in Comp. Sci. 136, Springer 1982.

[Ka1] W.M. Kantor, *Polynomial-time algorithms for finding elements of prime order and Sylow subgroups,* J. Algorithms 6 (1985) 478–514.

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

[Ka3] W.M. Kantor, *Finding Sylow normalizers in polynomial time* (to appear in J. Algorithms).

[KL] W.M. Kantor and E.M. Luks, *Algorithms for quotients of permutation groups,* in preparation.

[Ku] L. Kučera, *Canonical labeling of regular graphs in linear average time,* Proc. 28th IEEE FOCS 1987, 271–279.

[KT] W.M. Kantor and D.E. Taylor, *Polynomial-time versions of Sylow's theorem,* J. Algorithms 9 (1988) 1–17.

[Le] J.S. Leon, *On an algorithm for finding a base and strong generating set for a group given by generating permutations,* Math. Comp. 35 (1980), 941–974.

[Lu1] E.M. Luks, *Isomorphism of graphs of bounded valence can be tested in polynomial time,* J. Comp. Syst. Sci. 25 (1982), 42–65.

[Lu2] E.M. Luks, *Computing the composition factors of a permutation group in polynomial time,* Combinatorica 7 (1987), 87–99.

[McK] B, McKay, *nauty User's Guide (version 1.2),* Tech. Rep. TR-CS-87-03, Dept. Comp. Sci., Austral. Nat. Univ. 1987.

[Mi1] G.L. Miller, *Isomorphism of k-contractible graphs, a generalization of bounded valence and bounded genus,* Inform. and Control 56, 1983, 1–20.

[Mi2] G.L. Miller, *Isomorphism of graphs which are pairwise k-separable,* Inform. and Control 56, 1983, 21–33.

[Ne] P.M. Neumann, *Some algorithms for computing with finite permutation groups,* Proc. of Groups-St. Andrews 1985 (Eds. E.F. Robertson and C.M. Campbell), London Math. Soc. Lect. Note 121, Cambridge U. Press 1987, 59–92.

[Ró] L. Rónyai, *Computing the structure of finite algebras,* to appear in J. Symbolic Computation (1989).

[Si] C.C. Sims, *Some group-theoretic algorithms,* Springer Lect. Notes in Math. 697 (1978), 108–124.

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