Applications and Adaptations of the Low Index Subgroups Procedure

Marston Conder Peter Dobcsányi

Department of Mathematics

University of Auckland

Private Bag 92019 Auckland

New Zealand

Abstract

The low-index subgroups procedure is an algorithm for finding all subgroups of up to a given index in a finitely-presented group G, and hence to determine all transitive permutation representations of G of small degree. A number of significant applications of this algorithm are discussed, in particular to the construction of graphs and surfaces with large automorphism groups. Further, three useful adaptations of the procedure are described, along with parallelisation of the algorithm. In particular, one adaptation finds all *complements* of a given finite subgroup (in certain contexts), and another finds all *normal* subgroups of small index in the group G. Significant recent applications of these are also described in some detail.

1991 Mathematics Subject Classification: 20-04, 20F05

1 Introduction

Given a finitely-presented group $G = \langle X \mid R \rangle$, where X is a finite set of generators and R is a finite set of relators (each expressed as a word on $X \cup X^{-1}$), it is frequently desirable to

find all subgroups of G of index up to some specified integer N, or at least a representative of each conjugacy class of subgroups of index up to N.

A complete enumeration of such subgroups is algorithmically feasible for two reasons. First, there are finitely many subgroups of up to a given index in any finitely-generated group G, since every subgroup H of index n corresponds to a homomorphism $G \to S_n$ (equivalent to the representation of G on right cosets of H), and there are only finitely many such homomorphisms (since there are only finitely many possibilities for the image of each of the elements from a finite generating set). Second, by Schreier's theorem, every subgroup of finite index in a finitely-generated group is itself finitely-generated [20].

The first algorithmic methods to determine a representative from each conjugacy class of small-index subgroups in a finitely-presented group were developed by Dietze and Schaps [14] and Sims [23, Section 5.9] in the 1960s and early 1970s. Generally referred as *low-index subgroups algorithms*, these are powerful methods with several important uses and applications.

Obviously they provide information about the structure of the group G, "from the top down"; see [23, Section 5.6]. They can often be used to determine whether or not G is infinite, or at least to obtain a lower bound on the order of G (see the example in [1, p.203]). More generally, since every subgroup H of finite index n in G corresponds to a transitive permutation representation of G on n points, they can be used to determine all finite factor groups of G isomorphic to permutation groups of small degree.

The latter observation has been exploited by the first author in several areas where finitely-presented groups play an important role. For example, the automorphism group of every trivalent symmetric finite graph is a factor group of one of seven (known) finitely-presented groups, and the low-index subgroups procedure was used to find interesting cases of such factor groups and associated graphs in [10]. In turn the discovery of these graphs motivated the construction of an infinite family of 5-arc-transitive cubic graphs with full automorphism group S_n (for all but finitely many n) in [3]. Similarly, finite factor groups of triangle groups and other finitely-presented groups are pivotal in the study of "large" groups of symmetries of surfaces, or actions of given finite groups on surfaces of small

genus, and their determination can be very useful as illustrated in [4, 9, 11] for example.

Two different strategies are most commonly used in low-index subgroups algorithms. One strategy involves calculation of a partial coset-table for the identity subgroup, up to a certain number of cosets, and introduction of "forced coincidences" from which subgroup generators are systematically created; see Method A in [14] for example. The other strategy involves a back-track style filling of entries in the coset-table from the beginning; see the procedure LOW_INDEX in [23, Section 5.6] for example. While both strategies perform enumeration of subgroups by traversing a search tree, in the first strategy the branching step is realised through selections of different possible subgroup generators, while in the second the branching is induced by different choices for a coset number during filling of the coset-table.

In our implementation, we followed the first strategy, and used as starting point the algorithm which is outlined as Method A in [14]. The main reason for our choice is that this method produces a wide search tree which is better suited to parallelisation. In what follows, we refer to this procedure as *the low-index subgroups algorithm*. We describe the working and the main features of this algorithm in Section 2.

For a number of applications, it can be helpful to further specialise the low-index subgroups algorithm to better suit the questions which need to be answered. Three such adaptations will be described in Section 3.

The first adaptation involves pursuing only selected branches of the search tree: those which correspond to subgroups which avoid all conjugates of a given set of elements. The second adaptation (referred as *normal*) finds all normal subgroups of small index, and thereby all finite factor groups of small order. This method searches only for normal subgroups, by treatment of additional subgroup generators as additional relators; it runs appreciably faster than finding all classes of subgroups and eliminating those which are non-normal, thus enabling a search up to much greater index within given time constraints. The third adaptation (referred as *mixed*) also finds normal subgroups, but with cosets enumerated over an intermediate subgroup.

Finally, significant applications of these various adaptations are reported in Section 4.

2 The low-index subgroups algorithm

In this section we give a brief overview of the *standard* low-index subgroups algorithm. We refer the reader to [14, 23] for more details on the standard algorithm, and to [2, 22] for further background on coset enumeration.

To find a representative of each conjugacy class of subgroups of index up to given N in the finitely-presented group $G = \langle X \mid R \rangle$, the algorithm uses a back-track search through a tree, with nodes at level k corresponding to certain subgroups of G generated by k elements. The search begins (at level 0) with the identity subgroup, generated by the empty set $\emptyset$, and successively adjoins and removes elements to and from the generating set for the subgroup, on a last-in first-out basis.

At each stage of the search, *coset enumeration* is used to define sufficiently many right cosets of the current subgroup H, and to construct a (possibly partial) coset-table for H. This table indicates as far as possible the effect of right multiplication of each generator of G on those right cosets of H which have been defined. Rows of the coset-table are indexed by the cosets, and columns are indexed by elements of the generating set X and their inverses: if $X = \{x_1, x_2, \ldots, x_m\}$ then the entry in the row indexed by the ith coset Hw_i and column indexed by x_j (or x_j^{-1}) is the number representing the coset Hw_ix_j (or $Hw_ix_j^{-1}$ respectively), if this is known.

In our coset enumeration procedure, defining new cosets alternates with testing current definitions of coset numbers using the given relators and current subgroup generators. The latter operation is frequently referred to as *scanning*. If each entry in the coset-table is defined, and the definitions satisfy simultaneously all tests against the relators and subgroup generators, then the enumeration of the cosets is completed and we the coset-table is said to be *closed*. During scanning, it may turn out that two different coset numbers are found to be assigned to the same coset; this is called a *coincidence* (of coset numbers). When a coincidence is found, the larger coset number is replaced by the smaller one, and all possible consequent coincidences are dealt with similarly; see [22] for a detailed description of this procedure.

In our low-index subgroups algorithm, the definition of new cosets happens in *normal order*: as the entries of the coset-table are scanned row-by-row (from left to right), each time a new coset number is encountered, this number is the smallest possible, beginning with 1 for the trivial coset H itself. In other words, the first entry in the table other than 1 should be 2, and the next new entry after this should be 3, and so on. More formally, if coset i appears first in row r_i and column c_i, then if $1 < i < j$ then either $r_i < r_j$ or otherwise $r_i = r_j$ and $c_i < c_j$. For example, the following (partial) coset-table for a subgroup H of a 3-generator group G is normally ordered:

	x_1	x_2	x_3	x_1^{-1}	x_2^{-1}	x_3^{-1}
1	2	3	4	2	5	4
2	1	6	7	1	0	7
3	0	5	0	0	1	0
4	7	0	1	7	0	1
5	0	1	0	0	3	0
6	0	0	0	0	2	0
7	4	0	2	4	0	2

The significance of normal ordering is that to each subgroup H of finite index in G there exists exactly one coset-table in normal order. (For a proof see [14].) While there are other possible orderings (see [23, Section 5.6] for example), we chose this definition of normal ordering as the most appropiate for the algorithm being described.

Also in the arrangement of the columns, we follow the convention of listing first the generators and then their inverses, as used by MAGMA [1]. In particular, in the case of an involutory generator, where the column for the inverse is identical to the column for the generator, the inverse column is redundant and can be eliminated.

At any stage of the low-index subgroups algorithm, if the current subgroup has more than N right cosets, then at least $N+1$ cosets (and usually a few more than this number) are defined, and so by the pigeon-hole principle, at least two of these cosets must coincide

if we want the subgroup to have index at most N. For this reason, branches are created to new nodes at the next level of the search tree by identifying pairs of cosets: forcing $Hw_i = Hw_j$ is equivalent to adjoining $w_i w_j^{-1}$ to a set of generators for H. These branches are often denoted in the form $i = j$ (although in fact $i < j$ before coincidence is forced), and are taken in order of increasing j, illustrated for the above example at level 1 as follows:

Coincidence	Additional generator
$1 = 2$	x_1^{-1}
$1 = 3$	x_2^{-1}
$2 = 3$	$x_1 x_2^{-1}$
$1 = 4$	x_3^{-1}
$2 = 4$	$x_1 x_3^{-1}$
$3 = 4$	$x_2 x_3^{-1}$
$\vdots$	$\vdots$

To avoid generating the same subgroup more than once, the set of forced coincidences at any branch point depends not only on the size of the corresponding coset-table but also on the node's position in the search tree. Also for the same purpose, tests are built into the algorithm to reject sub-trees if their root node, during coset enumeration, produces a coincidence "forbidden" by the node's position.

When a coset-table gets *closed* during coset enumeration at a node, a new subgroup has been found, and its generators become output. Whether closed or not, if the set of possible forced coincidences corresponding to the node is not empty, then new branches will be created. The algorithm stops when the whole search tree has been traversed.

A complete description of this algorithm can be found in [14], together with a proof that this procedure is guaranteed to terminate and produce each subgroup of up to the given index exactly once.

If we seek (as is usually the case) only one representative from each conjugacy class of subgroups, then an additional test is needed to eliminate those sub-trees which will produce

only conjugates of subgroups found at an earlier stage. The conjugacy test determines for each defined coset Hw_j a (partial) coset-table for the conjugate subgroup $w_j^{-1}Hw_j$ by letting the jth row of the coset-table for H become the first row, then using the information from this to construct subsequent rows as far as possible, and redefining coset numbers to achieve normal ordering. After this transformation, the conjugacy test checks whether the coset-table for $w_j^{-1}Hw_j$ (so far as it can be determined) precedes the one for H in lexicographical order.

Analysing the search tree generated by the low-index subgroup algorithm, we make an important observation:

Proposition 2.1 *In the standard low-index subgroups algorithm (as decribed above), disjoint sub-trees of the search tree can be processed independently.*

This property has important consequences for special adaptations, and provides a basis for parallelisation of the algorithm.

3 Adaptations

There are numerous significant adaptations and applications of the low-index subgroups algorithm. As noted in [23], the standard algorithm may be used simply to find a proper subgroup (of finite index) and thus prove the group is non-trivial, which may not be so easy to achieve by other means. In this section we discuss adaptations and parallelisation of the standard algorithm.

3.1 Pursuing selected branches of the search tree

First we describe a general adaptation which has a number of applications. As noted earlier, branches of the search tree are abandoned in the basic procedure once they are found to be equivalent to earlier branches (in terms of the hierarchy induced by normal ordering). Similarly, we may choose in advance to ignore selected branches of the search tree if these are known to produce only subgroups in which we have no interest.

One of the most obvious instances of this occurs when we wish to find all subgroups which do not contain a particular element, or any of its conjugates. Adding this element as an additional generator if necessary, and reordering the generators to make it the first generator x_1 in a finite presentation for G, we simply ignore the first branch at level 1 of the search tree: the branch $1 = 2$ (equivalent to assuming the subgroup H contains x_1^{-1}).

This idea can be extended to ignoring several initial branches at the first level, to avoid subgroups containing any (conjugate of) one of a number of specified elements. This can often be a matter of rewriting the presentation of G in terms of a generating set $\{y_1, y_2, \ldots, y_m\}$ such that elements to be avoided are the first few in the ordered list $y_1^{-1}, y_2^{-1}, y_1 y_2^{-1}, \ldots$, corresponding to the first few branches at level 1 of the search tree.

One such application was used by the first author [5] to construct an infinite family of asymmetric combinatorially-regular maps of type $\{3, 7\}$. Briefly, a map is an embedding of a connected graph on a closed surface, dividing the surface into simply-connected faces, and a symmetry of the map is a permutation of its edges which preserves vertex-edge, edge-face and vertex-face incidence. The map is *combinatorially-regular* of type $\{p, q\}$ if every face has exactly p edges and every vertex is incident with exactly q edges. Maps of type $\{7, 3\}$ may be obtained from permutation representations of the $(2, 3, 7)$ triangle group $\Delta(2, 3, 7) = \langle\, x, y \mid x^2 = y^3 = (xy)^7 = 1 \,\rangle$ of finite degree in which x, y, xy and $[x, y] = x^{-1}y^{-1}xy$ have no fixed points, by making each 3-cycle of y a single vertex and joining two vertices if two points in the corresponding 3-cycles are interchanged by x. If each of x, y, xy and $[x, y]$ has no fixed points, then the resulting graph can be embedded into an orientable surface in such a way that the faces are all heptagons corresponding to the 7-cycles of xy, and its topological dual is a map of type $\{3, 7\}$.

Candidates for maps of this type with trivial symmetry group (for use as building blocks for the construction of an infinite family of examples in [5]) were found by determining subgroups of index 84 in the group $\Delta(2, 3, 7)$ which contain no conjugate of any of the elements x, y, xy or $[x, y]$. This turned out to be relatively easy, since if the generators are taken in the order given (namely x followed by y) then the first few fruitful branches of the search tree are as follows:

$1 = 2$ forcing $x \in H$

$1 = 3$ forcing $y \in H$

$2 = 3$ forcing $xy^{-1} \in H$ (and thus $(xy)^{-1} = y^{-1}x \in y^{-1}Hy$)

$5 = 7$ forcing $xyxy^{-1} \in H$ (and thus $[x, y] \in (xy)^{-1}Hxy$)

$15 = 19$ forcing $(xy)^2(xy^{-1})^2 \in H$

etc.

Accordingly the search could begin at the branch $15 = 19$, so that no branch from there on would produce a subgroup containing a conjugate of one of the unwanted elements.

Another obvious application is to the search for *torsion-free* subgroups of minimum index in a finitely-presented group, such as an infinite Coxeter group (or more generally a reflection group), in which it is known that every torsion element is conjugate to one of the involutory generators x_i or one of the products $x_i x_j$ of two of these.

This approach was taken by Everitt [17] in a classification of 3-manifolds tessellated by regular solids, or more specifically, spherical and hyperbolic 3-manifolds obtainable by identifying faces of regular solids of type $\{p, q, r\}$. In the context of this problem, a cell Δ is a fundamental region for a subgroup H which is complementary to the cell-stabilizer $G_\Delta = \langle a, b, c\rangle$ in the Coxeter group $G = [p, q, r] = \langle a, b, c, d \mid a^2 = b^2 = c^2 = d^2 = (ab)^p = (bc)^q = (cd)^r = (ac)^2 = (ad)^2 = (bd)^2 = 1\rangle$, and instances of such a subgroup H are required.

For example, in the case of type $\{5, 3, 6\}$ (corresponding to tesellations by hyperbolic dodecahedra), the cell-stabilizer $\langle a, b, c\rangle = [5, 3]$ is isomorphic to $A_5 \times C_2$, and so any complementary subgroup must have index 120, contain no word of odd length in the generators a, b, c, d, and contain no non-trivial element which is a power of any conjugate of the elements ab, ac or bc. It follows that subgroups complementary to $\langle a, b, c\rangle$ can be found by using the low-index subgroups algorithm to determine subgroups of index 120 in $G = [5, 3, 6]$, but starting the process at branch $2 = 5$ at level 1 (which is equivalent to making the element ad the first generator for the subgroup).

A similar approach was taken earlier in [12] to find torsion-free subgroups of minimum

index in particular groups which produce hyperbolic 3-manifolds and orbifolds of minimal volume, and the potential exists for further applications in geometric situations where such subgroups or complements of a given finitely-generated subgroup need to be determined.

The advantages of this sort of adaptation are a reduction in search time, enabling search to a higher index if necessary, and a reduction in unnecessary output (of unwanted subgroups).

3.2 Parallelisation

As noted in Section 2, distinct sub-trees of the search tree can be processed independently; that is, the computations involved in traversing any sub-tree of the search tree are entirely determined by information corresponding to the root node of the sub-tree (together with the initial parameters of the problem). This means that the branches of the search tree from a given level downward can be processed in any sequence without changing the result. In particular, there is nothing to prevent us from processing the branches in parallel and thereby gaining much improved computational performance.

Such a parallel implementation of the low-index algorithm has been developed by the second author and is described in detail in [15]. Called LOWX, this implementation is capable of running on many parallel hardware platforms, but its most important use to date has been on KALÁKA, a 170-node Linux cluster designed and built also by the second author [15]. Some spectacular results have been achieved by running LOWX on KALÁKA, and these are outlined in Section 4.

The source code of the most recent version of LOWX is always available from the second author's website [16]. It is released under the terms of the GNU General Public License; for details see the file "COPYING" in the source distribution.

3.3 Finding normal subgroups only

In this sub-section we describe a more specific (but perhaps more significant) adaptation than the first. The aim of this adaptation is to determine all *normal* subgroups of up to

a given index N in a finitely presented group $G = \langle X \mid R \rangle$, or equivalently, to produce all finite factor groups G/K of G of order at most N. This takes appreciably less time than finding all classes of subgroups of index up to N and eliminating those which are not normal, and in turn enables a search up to much higher index (within given computing resources). Some applications are outlined in Section 4.

The adaptation of the standard low-index subgroups procedure is easy: when a coincidence between cosets Ku and Kv of the current subgroup K is forced in the branching process, all conjugates of the element uv^{-1} must lie in K if K is to be normal; hence in the coincidence processing and subsequent coset enumeration phases the element uv^{-1} should be applied to all cosets currently active (and not just the trivial coset numbered "1"). In other words, the element uv^{-1} is treated as an additional *relator* rather than an additional subgroup generator.

The search still begins (at level 0) with the identity subgroup, generated by the empty set $\emptyset$, but then successively adjoins and removes elements to and from a set S of representatives of conjugacy classes of G which generate the normal subgroup K, again on a last-in first-out basis. This adaptation not only reduces the coset-table more than a forced coincidence in the standard procedure at each stage, but also speeds up the process, since it eliminates the need for the subgroup conjugacy test.

The reduction in computing time is spectacular. To illustrate this, consider the modular group $PSL(2, \mathbb{Z})$, the group of Möbius transformations $z \mapsto (az + b)/(cz + d)$ with $a, b, c, d \in \mathbb{Z}$ and $ad - bc = 1$. This group has a defining presentation $\langle x, y \mid x^2, y^3 \rangle$ in terms of the transformations $x : z \mapsto -1/z$ and $y : z \mapsto (z-1)/z$, and is thus isomorphic to a free product $C_2 * C_3$ of cyclic groups of orders 2 and 3. One way of finding all normal subgroups of index up to (say) 20 in this group is to apply the standard low-index subgroups algorithm and check each subgroup in the output for normality (using a conjugacy test), deleting all subgroups which are non-normal. On a 225Mhz Cyrix MMX processor, the standard algorithm took 112 seconds to find conjugacy class representatives of all subgroups of up to index 20. The normal subgroups adaptation described above, however, took only 0.05 seconds to find all normal subgroups up to the same index.

3.4 Mixed adaptation

In many instances, the need to determine all "small order" quotients G/K of a finitely-presented group G occurs in the context of finite groups of automorphisms of discrete structures in which the stabilizer H of a point is non-trivial but of fixed order.

For example, in the case of an arc-regular trivalent graph X, the stabilizer H of a vertex v in the automorphism group $Aut\,X$ will be cyclic of order 3 (cyclically permuting the 3 neighbours of v). The action of $Aut\,X$ on vertices of X is equivalent to its representation on right cosets of H, so the number of vertices of X is given by $|X| = |Aut\,X : H| = |Aut\,X|/3$.

Here, it is the size of the structure X rather than the order of the quotient G/K which is of interest. Indeed $|G/K| = |X||H|$, and correspondingly the order of the quotient G/K can be quite large while the size of the structure X is relatively small. This motivates the development of a *mixed* adaptation which finds normal subgroups of G by performing coset enumerations not over the normal subgroup K but over an intermediate subgroup HK where H is the stabilizer of a point of X.

For this to succeed, we assume that H is a fixed non-trivial subgroup of G, the image of which is core-free in any quotient G/K of interest in the particular context. The assumption that the image HK/K of H in G/K is core-free is sufficient to ensure that the action of G/K on the cosets of HK/K is faithful, since the kernel of this action must be trivial.

The mixed adaptation involves enumeration of cosets of HK, where H is fixed and K is variable. The search for candidates for the normal subgroup K can proceed as follows: whenever the algorithm forces two cosets HKx and HKy to merge, the deduction $xy^{-1} \in HK$ gives rise to multiple possibilities for the additional relator, namely all elements of the form wxy^{-1} where w is an element of the fixed subgroup H, and each one is checked in terms of producing a possible new branch of the search tree.

Although the potential number of branches at each node of the search tree is increased, the maximum index is reduced, and so this "mixed" adaptation can still run much faster than the normal subgroups adaptation for the higher index. For example, the mixed adaptation of the low-index subgroups procedure has been used to determine all finite

connected arc-transitive trivalent symmetric graphs on up to 768 vertices, as announced in the next Section.

4 Applications

In this section we outline three significant applications where adaptations of the low-index subgroups algorithm were run using the LOWX and KALÁKA systems described above. References are given in each subsection to relevant publications. Additionally, to facilitate further computational use and application, the authors have made available (as plain text files) online the raw computational results of all three applications, on the second author's website [16].

4.1 Normal subgroups of Hecke groups

For any integer $q \geq 2$, the *Hecke group* $H_q = C_2 * C_q$ is the free product of a cyclic group of order 2 and a cyclic group of order q, with presentation $\langle\, a, b \mid a^2 = b^q = 1 \,\rangle$. In the special case $q = 3$, we have the modular group $C_2 * C_3 \cong \mathrm{PSL}(2, \mathbb{Z})$, as described in 3.3 above. Using the LOWX program to perform the normal subgroups adaptation of the low-index subgroups procedure, we have determined all normal subgroups of index up to 1500 in the modular group, and all subgroups of index up to 500 in each of the Hecke groups H_q for $4 \leq q \leq 12$.

A summary list of the number of subgroups in each case is given below:

Theorem 4.1 Let H_q be the Hecke group $C_2 * C_q$ (where $q \geq 3$). Then:

(a) H_3 has exactly 95 normal subgroups of index ≤ 500 (and 309 of index ≤ 1500);

(b) H_4 has exactly 688 normal subgroups of index ≤ 500;

(c) H_5 has exactly 44 normal subgroups of index ≤ 500;

(d) H_6 has exactly 862 normal subgroups of index ≤ 500;

(e) H_7 has exactly 26 normal subgroups of index ≤ 500;

(f) H_8 has exactly 1187 normal subgroups of index ≤ 500;

(g) H_9 has exactly 184 normal subgroups of index ≤ 500;

(h) H_{10} has exactly 469 normal subgroups of index ≤ 500;

(i) H_{11} has exactly 6 normal subgroups of index ≤ 500;

(j) H_{12} has exactly 1798 normal subgroups of index ≤ 500.

More details and an abridged version of the results are provided in [6]. To give the reader an idea of computation times, we note that the normal subgroups of index up to 1500 in the modular group $C_2 * C_3$ were found in 129 hours 32 minutes.

4.2 Regular maps of small genus

Maps and combinatorially-regular maps were mentioned briefly in 3.1 above. In the orientable case, a map is called *regular* (see [13]) if its group of orientation-preserving symmetries has a single orbit on incident vertex-edge pairs (arcs). Further, if it has also orientation-reversing symmetries the map is said to be *reflexible*, and in this case its full symmetry group is transitive, indeed regular, on incident vertex-edge-face triples (blades). A similar definition applies in the case of non-orientable regular maps.

The full symmetry group of every reflexible regular map M is a factor group G/K of the generic infinite group $G = \langle\, u, v, t \mid (uv)^2 = t^2 = (ut)^2 = (tv)^2 = 1 \,\rangle$, in such a way that if p and q are the orders of Ku and Kv in G/K, then M has type $\{p, q\}$. Conversely, if G/K is any non-degenerate finite homomorphic image of this group G (with non-degenerate meaning the images of the elements uv, t, ut and tv all have order 2), and p and q are the orders of u and v in the image, then a reflexible regular map of type $\{p, q\}$ can be constructed with G/K as its full symmetry group. The map is *orientable* if and only if the images of u and v generate a proper subgroup (of index 2) not containing the image of t, that is, if and only if K contains no word in which t occurs an odd number of times, and the Euler characteristic of M is given by $\chi = V - E + F = |G : K| \, (\frac{1}{2p} - \frac{1}{4} + \frac{1}{2q})$.

For negative characteristic, values of the expression $\frac{1}{2p} - \frac{1}{4} + \frac{1}{2q}$ are bounded below by $-1/84$ (with the bound attained in cases where $\{p, q\} = \{3, 7\}$), and accordingly the above

Euler-Poincaré formula bounds the index $|G : K|$ in terms of the Euler characteristic and hence in terms of the genus. Indeed $|G : K| \leq 168(g-1)$ for orientable regular maps of genus $g > 1$, while $|G : K| \leq 84(g-2)$ for non-orientable regular maps of genus $g > 2$, the genus being related to the characteristic by $\chi = 2 - 2g$ or $\chi = 2 - g$ respectively. Hence the determination of reflexible regular maps of given genus may be reduced to a search for normal subgroups of specified index in the group $\langle\, u, v, t \mid (uv)^2 = t^2 = (ut)^2 = (tv)^2 = 1 \,\rangle$.

A similar approach may be taken for orientable regular maps which are not reflexible, usually known as *chiral* maps. In this case we seek finite factor groups of the group $G = \langle\, u, v \mid (uv)^2 = 1 \,\rangle$, which have no automorphism of order 2 inverting the images of the generators u and v, or equivalently, normal subgroups of finite index in G which are not normalised also by the automorphism of G taking $u \mapsto u^{-1}$ and $v \mapsto v^{-1}$. A test for this property (and a similar test for duality) can be easily incorporated into a post-processing phase, or indeed into the low-index subgroups procedure itself.

For this application we ran the LOWX program on the KALÁKA system, to determine all orientable regular maps of genus 2 to 15, reflexible and chiral, and all non-orientable regular maps of genus 3 to 30. Previous classifications went only as far as genus 7.

Theorem 4.2 There exist (up to isomorphism, duality and reflection) exactly:

(a) 220 reflexible orientable regular maps of genus 2 to 15 inclusive;

(b) 16 chiral orientable regular maps of genus 2 to 15 inclusive;

(a) 83 reflexible non-orientable regular maps of genus 3 to 30 inclusive.

Complete lists and more details are given in [7]. The following indicate the time taken for the longest branch of each computation:

Computation	Index	Time
Reflexible, type $\{3, 7\}$	2352	1 hour 57 minutes
Reflexible, type $\{3, -\}$	1344	5 hours 1 minute
Reflexible, type $\{4, -\}$	1120	14 hours 51 minutes
Reflexible, any type	560	11 hours 55 minutes

Chiral, type $\{3,7\}$	1176	47 minutes
Chiral, type $\{3,-\}$	672	2 hours 8 minutes
Chiral, type $\{4,-\}$	560	5 hours 37 minutes
Chiral, any type	280	5 hours 35 minutes.

4.3 Trivalent symmetric graphs of small order

Here we explain how the mixed adaptation of the low-index subgroups algorithm can be applied to determine all finite connected arc-transitive trivalent symmetric graphs on up to a given number of vertices.

An undirected simple graph Γ is said to be arc-transitive (or *symmetric*) if the group of all automorphisms of Γ has a single orbit on arcs (ordered edges) of Γ. In particular, the underlying graph of every regular map is symmetric. All symmetric graphs are necessarily regular, and they include null graphs (0-valent), ladder graphs (1-valent) and simple cycles (2-valent). The most interesting examples are k-valent where $k \geq 3$, and a partial census of 3-valent connected examples on up to 512 vertices was compiled by R.M. Foster several years ago, but with several gaps.

It follows from the work of Tutte and others that the automorphism group of every finite trivalent symmetric graph Γ is a factor group of one of seven finitely-presented groups G_1, G_2^1, G_2^2, G_3, G_4^1, G_4^2, G_5, depending on the type of Γ (the degree of arc-transitivity and the existence or otherwise of arc-reversing automorphisms of order 2). Conversely, every non-degenerate homomorphic image of one of these seven groups acts arc-transitively on a connected trivalent graph, the vertices of which may be identified with cosets of a certain particular subgroup (corresponding to the stabilizer of a vertex).

For example, if the full automorphism group of Γ acts regularly on 3-arcs (directed non-returning walks of length 3), then the automorphism group must be a non-degenerate homomorphic image of the group $G_3 = \langle h, a, p \mid h^3 = a^2 = p^2 = [h,p] = (hapa)^2 = (ap)^4 = 1 \rangle$, with the stabilizer of a vertex being an isomorphic image of the subgroup $H = \langle h, p, apa \rangle \cong S_3 \times C_2$ (of order 12). Accordingly, if Γ has n vertices then $\mathrm{Aut}\,\Gamma$ has

order $12n$, and hence in order to find all graphs of this type on up to 768 vertices, we seek all factor groups of G_3 modulo normal subgroups K of index at most 9216.

Similar observations can be made for each of the other six groups G_1, G_2^1, G_2^2, G_4^1, G_4^2, G_5, with the order of the corresponding subgroup H ranging from 3 up to 48. In each case, however, rather than determining the normal subgroups themselves (of quite large index in some cases), it is computationally more practical to use the mixed adaptation.

Recall that in the mixed adaptation, if K is the variable normal subgroup, then whenever the algorithm forces two cosets HKx and HKy to coincide, the deduction $xy^{-1} \in HK$ gives rise to multiple possibilities for the additional relator, namely all elements of the form wxy^{-1} where w is an element of the fixed subgroup H, and each one is checked as a possible new branch of the search tree.

In the context of these symmetric graphs, the subgroup HK is the pre-image in the given finitely-presented group of the stabilizer of a vertex, and hence the index of HK is exactly equal to the number of vertices of the corresponding graph. Moreover, the coset-table defines the natural permutation representation of the given group on the cosets of the subgroup HK, and as the image HK/K of H in G/K is core-free in each case of interest, this representation has kernel K and therefore corresponds precisely to the action of the group G/K on the vertices of the graph.

In particular, to find all arc-transitive actions of groups on trivalent symmetric graphs on up to 768 vertices (and hence the graphs themselves), we need only search for subgroups of the above form of index up to 768 in each of the seven groups (and determine in each case using MAGMA [1] the full automorphism group of the corresponding graph). Although the potential number of branches at each node of the search tree is increased (up to 48 times as many as previously), the mixed adaptation still runs much faster than the normal subgroups adaptation for the higher index.

The results of such computations, again using the LOWX program on the KALÁKA platform, have provided a complete list of all such graphs on up to 768 vertices.

Theorem 4.3 Up to isomorphism there are exactly:

(a) 135 finite 1-arc-regular trivalent graphs (of type G_1) on up to 768 vertices;

(b) 121 finite 2-arc-regular trivalent graphs of type G_4^2 on up to 768 vertices;

(c) 1 finite 2-arc-regular trivalent graph of type G_4^2 on up to 768 vertices;

(d) 31 finite 3-arc-regular trivalent graphs (of type G_3) on up to 768 vertices;

(e) 5 finite 4-arc-regular trivalent graphs of type G_4^1 on up to 768 vertices;

(f) no finite 4-arc-regular trivalent graphs of type G_4^2 on up to 768 vertices;

(g) 5 finite 5-arc-regular trivalent graphs (of type G_5) on up to 768 vertices.

The full list is given in [8]. This list fills the gaps in the Foster census, and as a bonus, the computations produced the smallest arc-transitive trivalent graph having no automorphism of order 2 which reverses an arc: a previously undiscovered graph on 448 vertices.

It is very difficult to give an accurate description of the computation times involved, however the following times taken by the longest branch of each computation provide reasonable estimates of these:

Type	Time required for longest branch at first level
G_1	13 hrs 09 mins
G_2^1	43 hrs 33 mins
G_2^2	3 hrs 48 mins
G_3	10 hrs 22 mins
G_4^1	4 hrs 38 mins
G_4^2	2 hrs 06 mins
G_5	3 hrs 48 mins.

5 Concluding remarks

The specific-branch and normal subgroup adaptations described in 3.1–3.3 are not only useful in a number of contexts, but have also produced some remarkable results, due mainly

to a drastic reduction in computing times. Similarly the type of mixed adaptation has the capacity to lower the maximum index to a large extent and further reduce computing times, again pushing out the boundaries of what is computationally feasible.

Finally we note that many other adaptations are possible, some of which were alluded to in previous sections and references [7, 8] given there.

For example, reflexibility of regular maps (see 4.2) corresponds to an outer automorphism of order 2 which inverts each of the generators u and v of the generic finitely-presented group $G = \langle\, u, v \mid (uv)^2 = 1 \,\rangle$, and a test for the presence of the image of such an automorphism can be incorporated into the low-index procedure analogously to the conjugacy test: simply switch the columns in the coset-table representing the effects of u and u^{-1}, and also those for v and v^{-1}, then redefine coset numbers to achieve normal ordering, and check whether the reflected table is the same as the original. Also topological duality (which interchanges vertices with faces) can be handled similarly, by switching the columns for u and v, and also those for u^{-1} and v^{-1}.

Analogously, given a finite factor group of one of the seven finitely-presented groups mentioned in 4.3, the type of the corresponding trivalent symmetric graph can often be determined by testing for the existence of additional automorphisms which have prescribed effects on the generators and their inverses (see [8]), and again these can be incorporated. The potential exists for numerous adaptations of this sort, especially in the context of geometric and combinatorial structures with large symmetry groups.

Acknowledgments

The authors are grateful to the N.Z. Marsden Fund and the University of Auckland Research Committee for their support, and to the referee and editor for several helpful comments.

References

[1] W. Bosma & J.J. Cannon, *Handbook of Magma Functions*, University of Sydney, 1996.

[2] J.J. Cannon, L.A. Dimino, G. Havas & J.M. Watson, Implementation and analysis of the Todd-Coxeter algorithm, *Mathematics of Computation* **27** (1973), 463–490.

[3] M.D.E. Conder, An infinite family of 5-arc-transitive cubic graphs, *Ars Combinatoria* **25A** (1988), 95–108.

[4] M.D.E. Conder, The symmetric genus of the Mathieu groups, *Bull. London Math. Soc.* **23** (1991), 445–453.

[5] M.D.E. Conder, Asymmetric combinatorially-regular maps, *J. Algebraic Combinatorics* **5** (1996), 323–328.

[6] M.D.E. Conder & P. Dobcsányi, Normal subgroups of low index in the modular group and other Hecke groups, *University of Auckland Mathematics Department Research Report Series*, No. 496, 23 pp., 2003.

[7] M.D.E. Conder & P. Dobcsányi, Determination of all regular maps of small genus, *J. Combinatorial Theory, Series B* **81** (2001), 224–242.

[8] M.D.E. Conder & P. Dobcsányi, Trivalent symmetric graphs on up to 768 vertices, *J. Combinatorial Mathematics & Combinatorial Computing*, **40** (2002), 41–63.

[9] M.D.E. Conder & B.J. Everitt, Regular maps on non-orientable surfaces, *Geometriae Dedicata* **56** (1995), 209–219.

[10] M.D.E. Conder & P.J. Lorimer, Automorphism groups of symmetric graphs of valency 3, *J. Combinatorial Theory Ser. B* **47** (1989), 60–72.

[11] M.D.E. Conder, C. Maclachlan, S. Todorovic-Vasiljevic & S.E Wilson, Bounds for the number of automorphisms of a compact non-orientable surface, *J. London Math. Soc.*, to appear 2003.

[12] M.D.E. Conder & G.J. Martin, Cusps, triangle groups and hyperbolic 3-folds, *J. Austral. Math. Soc. (Series A)* **55** (1993), 149–182.

[13] H.S.M. Coxeter & W.O.J. Moser, *Generators and Relations for Discrete Groups*, 4th ed. Springer-Verlag (Berlin), 1980.

[14] A. Dietze & M. Schaps, Determining subgroups of a given finite index in a finitely presented group, *Canadian J. Mathematics* **26** (1974), 769–782.

[15] P. Dobcsányi, *Adaptations, Parallelisation and Applications of the Low-index Subgroups Algorithm*, PhD thesis, University of Auckland, 142pp, 1999.

[16] P. Dobcsányi, Home Page, `http://www.math.auckland.ac.nz/~peter`.

[17] B.J. Everitt, *Images of Hyperbolic Reflection Groups*, PhD thesis, University of Auckland, 153pp, 1994.

[18] G. Havas, A Reidemeister-Schreier program, *Proc. Second International Conference on the Theory of Groups, Canberra 1973*, (Lect. Notes in Math. 372, Springer-Verlag, 1974), 347–356.

[19] D.F. Holt & W. Plesken, A cohomological criterion for a finitely presented group to be infinite, *J. London Math. Soc.* **45** (1992), 469–480.

[20] D.L. Johnson, *Presentations of Groups*, Cambridge University Press, 1990.

[21] P.J. Lorimer, Hyperbolic pyritohedra constructed from the Coxeter group $[4, 3, 5]$, *Computational Algebra and Number Theory (Sydney 1992)*, Kluwer, 1995, 303–321.

[22] J. Neubüser, An elementary introduction to coset-table methods in computational group theory, *Groups – St. Andrews 1981*, London Math. Soc. Lecture Note Series, vol. 71, 1982, pp. 1–45.

[23] C.C. Sims, *Computation with Finitely-Presented Groups* (Cambridge University Press, 1994).

Marston Conder: `conder@math.auckland.ac.nz`
`http://www.math.auckland.ac.nz/~conder`

Peter Dobcsányi: `peter@math.auckland.ac.nz`
`http://www.math.auckland.ac.nz/~peter`