ADDENDUM TO
AN ELEMENTARY INTRODUCTION TO COSET TABLE METHODS IN COMPUTATIONAL GROUP THEORY

COLIN M. CAMPBELL*, GEORGE HAVAS†[1] and EDMUND F. ROBERTSON*

*School of Mathematics and Statistics
University of St Andrews
North Haugh, St Andrews, Fife KY16 9SS, Scotland
Email: colinc@mcs.st-and.ac.uk, edmund@mcs.st-and.ac.uk
and
†ARC Centre for Complex Systems
School of Information Technology and Electrical Engineering
The University of Queensland, Queensland 4072, Australia
Email: havas@itee.uq.edu.au

Even after 25 years the article [30] by Joachim Neubüser remains the first source to which all three of us refer those who want to find out about the use of coset tables for studying groups. Our view is confirmed by the 14 Reference Citations from 1998 to 2005 which MathSciNet [1] reveals for this article. Here we loosely follow the structure of the original article and provide some updates on the area (oriented towards our own interests).

First we point out that two newer books [35, 22] include comprehensive details on coset enumeration and related topics in works which are much broader studies. They give excellent coverage of the areas addressed in this article and, further, provide much additional material. They also provide some alternative points of view and many references (as do the other materials cited here).

One of Neubüser's aims in writing his survey was to provide a unified view on coset table methods in computational group theory. He addressed the way coset table concepts were developed, implemented and used. In [22] Derek Holt follows the same kind of approach, including a long chapter "Coset Enumeration" and a shorter one "Presentations for Given Groups". Charles Sims in [35] focuses on finitely presented groups and he takes a perspective significantly based on some fundamental methods from theoretical computer science, namely automata theory and formal languages. He includes three chapters specifically relevant to coset table methods: "Coset enumeration"; "The Reidemeister-Schreier procedure"; and "Generalized automata"; with some extra implementation issues covered in an Appendix. He concludes his coset enumeration chapter with a section which points out that the Knuth-Bendix procedure can sometimes be used more effectively to enumerate cosets than Todd-Coxeter methods.

Among the available computer implementations of coset enumeration procedures are those in the computer algebra systems GAP [14] and MAGMA [5] and a stand-alone program, the ACE coset enumerator [18]. An implementation is also available via **quotpic** [23], a software package with a nice graphical interface. A particularly useful tool for small-scale experiments with coset enumerations is the Interactive

[1]The second author was partially supported by the Australian Research Council

Todd-Coxeter package, ITC [12]. Many aspects of implementation and performance issues are addressed in [16, 17], including some comparisons with Knuth-Bendix-based methods. Much work has been done on practical strategies for successful coset enumeration. Accessible introductions to readily-available strategies can be found in the documentation for the GAP package ACE [13] and in the MAGMA manual [29].

Neubüser describes several kinds of information available from coset tables. Such information can be readily extracted from the various implementations. A more recent program, PEACE (Proof Extraction After Coset Enumeration), gives users the opportunity to uncover proofs from the workings involved in coset enumerations. It is based on much earlier work of John Leech [24, 25]. Details appear in the Groups St Andrews 2005 proceedings [19] with a significant application in [21].

Reidemeister-Schreier-based methods for finding presentations of subgroups are described in [35, 22, 3]. The systems GAP, MAGMA and **quotpic** all incorporate efficient implementations. Applications of such methods continue to be widely used to address problems in finitely presented groups; see for example [11, 4]. They have also been extended to work in other structures, such as semigroups (see for example [33]) and Lie algebras. Initial implementations computed presentations on a set of Schreier generators for the subgroup and followed by simplification techniques. Subsequent, more complicated, algorithms utilise an augmented coset table which enables the construction of presentations on user-given sets of subgroup generators. Ideas which allow such a modified algorithm to be implemented more efficiently are described in [2] and such ideas are incorporated in GAP and MAGMA implementations.

Neubüser already gives some information about computing presentations for a concrete group and methods based on [7] are included in GAP and MAGMA. A newer method for finding short sets of defining relations is given by [15], which utilises ideas from double coset enumeration. Double coset enumeration is covered independently in [26]. Search-based methods for finding presentations with nice properties are used in [20, 6].

In his survey, Neubüser describes a method for computing all subgroups of low index by systematically forcing coincidences in larger coset tables. Now, more recent implementations of low index subgroup algorithms are available in GAP, MAGMA and **quotpic**. They use another method, due to Sims, which does a backtrack search through incomplete coset tables. Recent adaptations of the low index subgroup algorithm are described in [9].

The Schreier-Sims algorithm is now well covered in material on permutation groups, including chapters in [34, 22]. It is used extensively in GAP and MAGMA. Its application to matrix groups is outlined in a recent survey [32, §7.5].

Neubüser wrote that applications of coset table methods to group theoretical questions are too numerous to be listed in his article and are often hidden. This is even more valid now, 25 years later. Thus, most applications of GAP or MAGMA to finitely presented groups are likely to implicitly invoke coset enumeration and many other applications also do so. Recall there are no algorithms for answering quite simple questions about finitely presented groups, as Neubüser [31] reminds us: there are "proofs of the non-existence of algorithms that could **decide** if a finitely presented group is trivial, finite, abelian, etc.", referring to [3] "for a vivid description". Often an appropriate way to start addressing a problem about a finitely presented group is to find some kind of permutation representation for the group, which is just what coset enumeration attempts to do. Thus, ask GAP or MAGMA for the order of a finitely presented group; unless the group is obviously infinite they both embark upon a coset enumeration (attempting to find the index of a cyclic subgroup whose order they also try to determine).

One way for finding further information about applications is to try looking on MathSciNet. For example, a MathSciNet search "Anywhere" for "coset table OR coset enumeration OR Todd Coxeter" gave 172 matches in July 2006. Another way is to follow citations provided by papers in our admittedly limited bibliography.

Neubüser also wrote "we may also hope that we have not yet seen the last variation" on coset table methods. We finish by citing some other work which we have not mentioned above. This includes vector enumeration, Kan extensions, and parallel coset enumeration; see, for example, [27], [28], [8], and [10].

Additional note. Joachim Neubüser has informed us that on page 16 of the original article it says: "(i) A coset Ug is contained in the normalizer $N_G(U)$ iff $g^{-1}Ug = U$, i.e. iff $g^{-1}Ug \leq U$ and $gUg^{-1} \leq U$. These two conditions are satisfied iff $Ug^{-1}s_i = Ug^{-1}$ and $Ugs_i = Ug$, ..." He points out that the first condition is always enough, see page 114, exercise 17 (quoting a theorem of Takahasi) of Magnus, Karrass & Solitar (reference [41] of the original article).

References

[1] American Mathematical Society, *MathSciNet Mathematical Reviews on the Web*; `http://www.ams.org/mathscinet`

[2] D. G. Arrell and E. F. Roberston, A modified Todd-Coxeter algorithm, in *Computational Group Theory*, Academic Press (1984), 27–32.

[3] Gilbert Baumslag, *Topics in Combinatorial Group Theory*, Birkhäuser (1993).

[4] Jennifer Becker, Matthew Horak and Leonard VanWyk, Presentations of subgroups of Artin groups, *Missouri J. Math. Sci.* **10** (1998), 3–14.

[5] W. Bosma, J. Cannon and C. Playoust, The Magma algebra system I: the user language, *J. Symbolic Comput.* **24** (1997), 235–265.
See also `http://magma.maths.usyd.edu.au/magma/`

[6] Colin M. Campbell, George Havas, Alexander Hulpke and Edmund F. Robertson, Efficient simple groups, *Comm. Algebra* **31** (2003), 5191–5197.

[7] J. J. Cannon, Construction of defining relators for finite groups, *Discrete Math.* **5** (1973), 105–129.

[8] S. Carmody, M. Leeming and R. F. C. Walters, The Todd-Coxeter procedure and left Kan extensions, *J. Symbolic Comput.* **19** (1995), 459-488.

[9] Marston Conder and Peter Dobcsányi, Applications and adaptations of the low index subgroups procedure, *Math. Comp.* **74** (2005), 485–497.

[10] Gene Cooperman and Victor Grinberg, Scalable parallel coset enumeration: bulk definition and the memory wall, *J. Symbolic Comput.* **33** (2002), 563–585.

[11] F. Digne and Y. Gomi, Presentation of pure braid groups, *J. Knot Theory Ramifications* **10** (2001), 609–623.

[12] Volkmar Felsch, Ludger Hippe and Joachim Neubüser, *GAP package ITC Interactive Todd-Coxeter*, 2004; `http://www.gap-system.org/Packages/itc.html`

[13] Greg Gamble, Alexander Hulpke, George Havas and Colin Ramsay, *GAP package ACE; Advanced Coset Enumerator*, 2006; `http://www.gap-system.org/Packages/ace.html`

[14] The GAP Group, *GAP – Groups, Algorithms, and Programming, Version* 4.4, 2006; `http://www.gap-system.org/`

[15] Volker Gebhardt, Constructing a short defining set of relations for a finite group, *J. Algebra* **233** (2000), 526–542.

[16] George Havas and Colin Ramsay, Proving a group trivial made easy: a case study in coset enumeration, *Bull. Austral. Math. Soc.* **62** (2000), 105–118.

[17] George Havas and Colin Ramsay, Experiments in coset enumeration, in *Groups and Computation III*, Ohio State Univ. Math. Res. Inst. Publ. **8**, de Gruyter (2001), 183–192.

[18] George Havas and Colin Ramsay, *Coset enumeration: ACE version* 3.001, 2001; `http://www.itee.uq.edu.au/~havas/ace3001.tar.gz`

[19] George Havas and Colin Ramsay, On proofs in finitely presented groups, in *Groups St Andrews 2005*, London Math. Soc. Lecture Note Ser., Cambridge Univ. Press (to appear).

[20] George Havas and Edmund F. Robertson, Irreducible cyclic presentations of the trivial group, *Experiment. Math.* **12** (2003), 487–490.

[21] George Havas, Edmund F. Robertson and Dale C. Sutherland, The $F^{a,b,c}$ conjecture is true, II, *J. Algebra* **300** (2006), 57–72.

[22] Derek F. Holt, Bettina Eick and Eamonn A. O'Brien, *Handbook of Computational Group Theory*, CRC Press (2005).

[23] Derek F. Holt and Sarah Rees, A graphics system for displaying finite quotients of finitely presented groups, in *Groups and computation*, DIMACS Ser. Discrete Math. Theoret. Comput. Sci. **11** (1993), 113–126.

[24] John Leech, Computer proof of relations in groups, in *Topics in Group Theory and Computation*, Academic Press (1977), 38–61.

[25] John Leech, Coset enumeration, in *Computational Group Theory*, Academic Press (1984), 3–18.

[26] Stephen A. Linton, Double coset enumeration, in *Computational group theory, Part 2, J. Symbolic Comput.* **12** (1991), 415–426.

[27] S. A. Linton, On vector enumeration, in *Computational linear algebra in algebraic and related problems*, *Linear Algebra Appl.* **192** (1993), 235–248.

[28] S. A. Linton, Generalisations of the Todd-Coxeter algorithm, in *Computational algebra and number theory*, *Math. Appl.*, **325**, Kluwer Acad. Publ., (1995), 29–51.

[29] The Magma Development Team. *Interactive Coset Enumeration*, available via `http://magma.maths.usyd.edu.au/magma/htmlhelp/MAGMA.htm`

[30] J. Neubüser, An elementary introduction to coset table methods in computational group theory, in *Groups St Andrews 1981*, London Math. Soc. Lecture Note Ser. **71**, Cambridge Univ. Press (1982), 1–45.

[31] J. Neubüser, An invitation to computational group theory, in *Groups '93 Galway/St. Andrews, Vol. 2*, London Math. Soc. Lecture Note Ser. **212**, Cambridge Univ. Press (1995), 457–475.

[32] E. A. O'Brien, Towards effective algorithms for linear groups, in *Finite Geometries, Groups, and Computation*, Walter de Gruyter (2006), 163–190.

[33] N. Ruškuc, Presentations for subgroups of monoids, *J. Algebra* **220** (1999), 365–380.

[34] Ákos Seress, *Permutation group algorithms*, Cambridge Tracts in Mathematics **152**, Cambridge Univ. Press (2003).

[35] C. C. Sims, *Computation with finitely presented groups*, Cambridge Univ. Press, 1994.