Alex. Eng. J., Alex. Univ., Volume, 28No.2, pp. 491 - 501 (1989)

DEFINING RELATIONS OF FINITELY PRESENTED GROUPS

Fatma El-Said Mostafa* and Magdi Salah ElAzab**

* Engineering Mathematics and Physics Department
Faculty of Engineering, Alexandria University
Alexandria, Egypt

** Demonstrator in Faculty of Engineering
Mansoura University
Mansoura , Egypt

Abstract

Instead of letting the task of finding a set of relations with a good chance to be defining is left to intuition, Cannon developed a systematic method for constructing a set of defining relations that uses the Todd-Coxeter algorithm.

We describe essentially this method with some details, revailing ambiguities in his paper supporting it by illustrative examples and some comments.

Notations

G = <X,R> The group generated by the set of generators X and subjected to the set of relations R.

$W_h(g_i)$ Is a word in the generators g_i of the group G for each h G.

Introduction

Applications of computers to group theory began by a paper of Todd and Coxeter [11]. Since that time so many papers [1], [2], [3], [6], [7], [8], [10], [12] dealt with computations in group theory. In this paper a set of defining relations of a group of a given order and a given set of generators is to be determined.

Let a group G be given by a set of generators, the only prerequisite being that inverses and products of elements can be formed and elements can be compared. However, according to the order of the given group there are two algorithms, namely, one-stage algorithm and two-stage algorithm.

1. One-stage Algorithm

This algorithm is used if the order of the group G is small enough for keeping the coset table and also a list of all elements of G in store. With reasonable effort this list is computed from the generators of G and from it the coset table of G with respect to the unit subgroup H = <e> is constructed by just multiplying each element by each generator and looking up the result in the list of all elements. Next we define inductively from the coset table for each element $h \in G$ a representation by a word $W_h(g_i)$ in the generators of G. We begin the algorithm by assigning the empty word to the identity, i.e. we put $a_1 = e$ and $W(a_1) = 1$. Also we find for an element h its first occurance as a product h=kg of some element k to which a word $W_k(g_i)$ has already been assigned and define $W_h(g_i) = W_k(g_i)g$. At the same time the entries v = u g in the coset table that have been used for the definition of these words and the corresponding entires $u = vg^{-1}$ are marked (e.g. by replacing them by their negative value). Having defined the word $W_h(g_i)$ for all $h \in G$, we see that each entry in the coset table giving an information lg = t yields a relation $W_l(g_i)\, g\, W_t^{-1}(g_i) = 1$ in terms of the generators g_i which may or may not be trivial. We start the relation finding algorithm by looking up the first unmarked entry in the coset table. We mark it and if it yields a nontrivial relation, say $r_1(g_i) = 1$, we set up a relation table for r_1 and insert into it all possible marked entries from the coset table. If a row of the relation table closes, yielding a deduction a g_j = b, then the entry in the position T(a, g_j) must be b, if this entry is not already marked, we now mark it and the corresponding one $bg_j^{-1} = a$. When we have passed through all rows of the relation table of r_1, we search for further unmarked entries in the coset table. If there are any we treat the first we encounter as before, getting an additional relation and at each step we fill all marked entries into all relation tables that have been set up so far. The process comes to an end when all entries in the coset table have been marked and at that time also all relation tables will have closed.

Lemma 1.1

The relations $r_1(g_i) = 1, \ldots, r_m(g_i) = 1$ found to hold for the generators g_i in the process form a defining set of relations for G. The proof of this lemma is given in [9].

Let G be a group of order m , generated by the k generators g_1, .., g_k then the number of the defining set of relations for G can be derived as follows:

δ = no. of elements of the table-no. of marked entries

$= m * k - (m-1)$

$= m * (k - 1) + 1$

Consider the following illustrative example.

Example 1.2

Consider a group G of order 6 generated by the two generators A and B. We assume that the inverses and products of elements can be formed and elements can be compared. First we construct the coset table of G with respect to the unit subgroup. Of course the number of rows of this table is equal to 6.

Let $\hat{A}_1 = e$ and $W(\hat{A}_1) = 1$. Since T (1,1) = 0 we insert 2 = 1A , i.e. we have T (1,1) = 2 and T (2,3) = 1, and add the element A to the list of elements $\hat{A}$. Thus we have $\hat{A}_2 = A$. Repeating this step we put T (1,2) = 3 and T (3,4) = 1 and put $\hat{A}_3 = B$. Continuing we put T (1,3)=4 and T(4,1) = 1 and $\hat{A}_4 = A^{-1}$. At this stage we have the list of elements of G as

$$\hat{A} = \{ e , A , B , A^{-1} \}$$

When we test T (1,4) we find that it equals 0, but we can not insert a marked entry in that place, since we have the information $B^{-1} = \hat{A}_3$

= B so we insert the unmarked entry -3 in the position T (1,4) and put T (3,2) = -1. T(2,1) = 0 and the element $\hat{A}_2 A = A^2$ exists in the list $\hat{A}$ in the form A^{-1}, thus we put T (2,1) = -4 and T(4,3) = -2 Taking T(2,2) which is equal to zero but the element $\hat{A}_2 B$ = AB is not contained in the list $\hat{A}$ so we add the new element AB to the list $\hat{A}$. Going through all the elements of the coset, we have the following table, with marked and unmarked entries :

*	A	B	A^{-1}	B^{-1}
1	2	3	4	-3
2	-4	5	1	-5
3	6	-1	-5	1
4	1	-6	-2	-6
5	-3	-2	-6	2
6	-5	-4	3	-4

The marked entries of the coset table form the following list of all elements of G:

$$\hat{A} = \{ e , A, B, A^{-1} , AB , BA \} .$$

Now we start the relation finding algorithm by looking up the first unmarked entry in the coset table which is T(1,4). First we mark it, i.e. we put T(1,4) = 3 and T(3,2) = 1, forming the relation

$$Rel_1 = W(\hat{A}_1)\, B^{-1} W(\hat{A}_3)^{-1}$$

$$= 1 * B^{-1} * B^{-1} = B^{-2}.$$

The second unmarked entry is T (2,1) = -4, we mark it and get the second relation

$$Rel_2 = W(\hat{A})\ AW\ (\hat{A})^{-1} = A * A * A \cdot$$

Continuing, we get the following set of defining relations:

$$Rel_1 = W\ (\hat{A}_1)B^{-1}W\ (\hat{A}_3)^{-1} = B^{-1} * B^{-1}$$

$$Rel_2 = W(\hat{A}_2)A\ W(\hat{A}_4)^{-1} = A * A * A$$

$$Rel_3 = W(\hat{A}_2)\ B^{-1}\ W(\hat{A}_5)^{-1} = A * B^{-1} * (AB)^{-1}$$

$$Rel_4 = W(\hat{A}_3)B\ W(\hat{A}_1)^{-1} = B * B$$

$$Rel_5 = W(\hat{A}_3)\ A^{-1}\ W(\hat{A}_5)^{-1} = B * A^{-1} * (AB)^{-1}$$

$$Rel_6 = W(\hat{A}_4)B\ W(\hat{A}_6)^{-1} = A^{-1} * B * (BA)^{-1}$$

$$Rel_7 = W(\hat{A}_4)A^{-1}\ W(\hat{A}_2)^{-1} = A^{-1} * A^{-1} * A^{-1}$$

2. Two-Stage Algorithm

The application of the one-stage algorithm is limited by the order of the group G. However if the order of G is too big to be used in the one-stage algorithm, we follow another algorithm, called the two-stage algorithm, in order to get a presentation for the group G. The basic procedure in the two-stage algorithm is to try to find a set of words $h_1, h_2, ..., h_p$ in G, expressed in terms of the generators g_i, that generate a subgroup H, which is small enough for the one-stage algorithm. Going through the one-stage algorithm we obtained a presentation:

$$H = < h_1, \dots, h_p : S_1(h_j) = 1, \dots, S_q(h_j) = 1 > \qquad (1)$$

of H in terms of the h_j .

Since each h was supposed to be expressed in terms of the g_i then we have for each $h \in H$ a word $W_h(g_i)$ expressing h in terms of the g_i. Let us further suppose that we can still construct the coset table of G modulo H. Since the numbers in the coset table now represent cosets of H, then an equation tg = 1 read from an entry 1 in the t-th row and g-column of the coset table, yields $W_t(g_i)\, gW_1(g_i)^{-1} \in H$. However, we may evaluate $W_t(g_i)\, gW_1(g_i)^{-1}$ and obtain an element h of H which in turn is expressed as a word $W_h(g_i)$ by the information we had obtained earlier about H. Hence from tg = 1 we get the relation

$$r(g_i) = W_t(g_i) g W_1(g_i)^{-1} W_h(g_i)^{-1} = 1$$

which may or may not be trivial. With the modification that we use these more complicated relations instead of the easier ones in the one-stage algorithm we now proceed untill all entries in the coset table have been tested.

Let $r_1(g_i) = 1, \dots, r_m(g_i) = 1$ be the relations gathered in the process. Adding to these relations, the relations in (1) we obtain a set of relations in the form :

$$S_1(h_j)=1, \dots, S_q(h_j) = 1, r_1(g_i) = 1, \dots, r_m(g_i) = 1$$

$$h_1 = h_1(g_i), \dots, h_p = h_p(g_i) \qquad (2)$$

Cannon in [2] proved that these relations define G in terms of the set

$$\{ g_1, \ldots, g_k, \quad h_1, \ldots, h_p \}.$$

Few comments are appreciated:

1. The presentation obtained for G can be simplified by eliminating the redundant generators $h_1, \ldots, h_p$ by a sequence of Tietze transformations (see [5]) using $h_j = h_j(g_i)$. Doing this we arrive at the presentation

   $$G = < g_1, \ldots, g_k \mid r_\Omega (g_i) = 1, S_\mu (h_j(g_i)) = 1 >$$

2. For the two-stage algorithm, only the following had to be known
   (i) A generating set $(g_1, \ldots, g_k)$ of G ;
   (ii) Generators $h_1, \ldots, h_p$ of a subgroup H of G, expressed as words $h_j = h_j (g_i)$ in terms of the generators g_i of G;
   (iii) A presentation of H in terms of the h_j;
   (iv) A possibility to express a given element $h \in H$ in terms of the generators $g_i \in G$;
   (v) A coset table of G modulo H (for the generating set $\{g_1, \ldots, g_k\}$)

3. It is sufficient to deal with the deduced entries in the coset table to save the time of execution since the defined entries in the table yield redundant relators.
   Now, we introduce the following illustrative example.

Example:

Let the group G be generated by the two generators A,B and let the subgroup H of G be presented by

$$H = < X, Y | X^2 , Y^2 , (XY)^3 >$$

where X = AB and Y = BA. However, this presentation helps us to get all the elements of the subgroup. Hence we have the elements {1, X, Y, XY, YX, XYX, YXY} of H or in terms of the generators of G they are expressed as

$$< 1, AB, BA, AB^2A, BA^2B, AB^2A^2B >.$$

Further the coset table of G mod H is given in the form

*	A	B	A^{-1}	B^{-1}
1	2	3	3_2	2_1
2	4	1_1	1	3_3
3	1_2	2_3	4_4	1
4	3_4	4_5	2	4_5

The coset representatives are : $y_1 = 1$, $y_2 = A$, $y_3 = B$, $y_4 = A^2$

First , we form the permutations corresponding to the elements of H.

They are :

1	$\longrightarrow$	1
AB	$\longrightarrow$	$(2\ 4)$
BA	$\longrightarrow$	$(3\ 4)$
AB^2A	$\longrightarrow$	$(2\ 3\ 4)$
BA^2B	$\longrightarrow$	$(2\ 4\ 3)$
AB^2A^2B	$\longrightarrow$	$(2\ 3)$

Now, going through the deduced entries of the coset table only we have the relations

$$1A^{-1} = 3 \rightarrow A^{-1} = B \rightarrow A^{-1}B^{-1} \in H .$$

The corresponding permutation for the word $A^{-1}B^{-1}$, by the word tracing technique, is (3 4), thus equating it with the word $BA \in H$ we get the first relation

$$A^{-1}B^{-1} = BA \longrightarrow A^{-1}B^{-1}A^{-1}B^{-1} = 1 .$$

Continuing through the coset table we get the relations:

$$ABAB = 1$$
$$B^3 = 1$$
$$ABAB^{-2} = 1$$
$$A^4 = 1$$
$$A^3B^{-1}A^{-1}B^{-1} = 1$$

(the trivial relations are omitted).

Clearly, we can simplify the above presentation using any simplification technique (see [4]) to get the presentation of G in the form

$$G = < A, B \mid A^4 , B^3, (AB)^2 > .$$

References

[1] Cannon, J.J. ; Dimino, L.A., Havas, G. ; Watson, J.M. : Implementation and Analysis of the Todd-Coxeter Algorithm. Math. Comput. 27 (1973). 463-490.

[2] Cannon, J.J. : Construction of defining relators for finite groups. Discrete Math. 5(1973). 105-129.

[3] Dietze, A.; Schaps. M : Determining subgroups of a given finite index in a finitely presented group. Canad. J. Math. 26 (1974). 769 - 782.

[4] Havas, G.; Kenne, P.E.; Richardson. J.S., Robertson, E.F. : A Tietze transformation program. Computational group Theory, 1984, pp. 69-73.

[5] Johnson, D.L. : Presentation of groups. London Math. Soc. lecture note series, Vol. 22, Cambridge Univ. Press. Cambridge, 1976.

[6] Leech, J. : Coset enumeration on digital computers. Proc. Cambridge philos. Soc. 59 (1963). 257-267.

[7] Leech, J. : Coset Enumeration. PP. 21-35 in : Computational Problems in Abstract Algebra (Proc. Conf. Oxford, 1967). edited by J. Leech, Pergamon Oxford. 1970.

[8] Mendelsohn, N.S. : An algorithmic solution of a word problem in group theory. Canad. J. Math. 16 (1964) 509-516. Correction: Canada. J. Math. 17 (1965), 505.

[9] Neubuser. J. : An elementary introduction to coset table methods in computational group theory. Groupsst. Andrews 1981, pp. 1-45 London Math. Soc. lecture note series, 71, Cambridge Univ. Press, Cambridge New York, 1982.

[10] Sims, C.C. : Computational methods in the study of permutation groups. PP. 169-183 in: Computational Problems in Abstract Algebra (Proc. Conf. Oxford, 1967) edited by J. Leech, Pergamon, Oxford, 1970.

[11] Todd, J.A. ; Coxeter, H.S.M. : A practical method for enumerating cosets of a finite abstract group. Proc. Edinburgh Math. soc. (2) 5(1936), 26-34.

[12] Trotter, H.F. : A Machine program for coset enumeration Canad. Math. Bull. 7(1964), 357-368.