How hard is the word problem?

Sarah Rees, University of Newcastle, UK,

Introduction

Suppose (throughout this article) that

$$G = \langle X \mid R \rangle$$

is a finitely presented group, that is a group with finite generating set X, in which all equations are consequences of those in the finite set R. The *word problem* for G over X is defined to be the set $W(G, X)$ of all products of elements of $X \cup X^{-1}$ (known as *words* over X) which represent the identity.

The word problem is said to be *soluble* if the set $W(G, X)$ can be recognised by a halting Turing machine, that is, if there is a terminating algorithm which, given a word w over X, returns the answer 'yes' if w represents the identity element and 'no' otherwise. Solubility of the word problem for G does not depend on the choice of generating set X; a simple transformation turns a solution for $W(G, X)$ into one for $W(G, Y)$, for any finite set Y.

What does it mean to say that w represents the identity? It means that repeated application of the equations in R together with the rules of free cancellation transforms w into the empty word ϵ. Applying an equation $u = v$ from R means replacing a subword equal to either u or v by the other, and applying the rules of free cancellation to w means either deleting or inserting a subword of the form xx^{-1} or $x^{-1}x$, for $x \in X$. In general, there is no clear reason why every route to reduce any particular word w to the trivial word should not pass through a much longer word; if there is no upper bound on the length of intermediate words which must be encountered on any reduction of a word to the identity, then the design of an algorithm to do this is clearly difficult, if not impossible. In fact, it is a well known result of Novikov [16] and Boone [2, 3] that finitely presented groups exist with insoluble word problem.

When the word problem of a group is soluble, it is natural to ask how hard the problem is. That is the question discussed in this article. There are some answers but still much that is unknown. There are many different ways in which one might assess the difficulty of the word problem, indeed of any algorithm. It is standard to use bounds on time, or on space, or at least to use measures which give information about those bounds. It is also common to analyse the type of computational machine which is needed to perform the calculation (which could range from the most general Turing machine to a simple finite state automaton). We aim to introduce and compare briefly some of these different measures of complexity.

The literature on the word problem is rich, and the bibliography of this article is certainly not exhaustive. A very useful recent survey of geometrical aspects is provided by [5], which also has an extensive bibliography.

Max Dehn's work

The word problem was first studied by Max Dehn, together with the conjugacy and isomorphism problems [7, 8]. Dehn's motivation for the solution of the word problem was geometric; he wanted to solve the problem for the fundamental group of a surface, in which case a word represents the identity precisely when the corresponding loop contracts. His elegant, straightforward algorithm solved the problem for the fundamental group of a hyperbolic surface in linear time.

Any such group has a *Dehn presentation* over any generating set X, with the following properties. First, for each of the equations $u = v$ in the finite defining set R, u, the *left hand side*, is longer than v, the *right hand side*. Second, whenever w is a non-trivial word over X which represents the identity, the left hand side of at least one equation in R is a subword of w.

Dehn's algorithm uses a Dehn presentation. If a non-trivial input word w contains no left hand side as a subword then the answer 'no' can be returned. Otherwise w contains a left hand side u as a subword, substitution of u by the corresponding right hand side v gives a shorter representative of the same group element, and iteration solves the problem. The algorithm can operate in time which is linear in the length of the input word w.

A geometric view

In general, the word problem may be best understood in the way that Dehn did, geometrically.

The Cayley graph $\Gamma(G, X)$ is defined to have its vertices in correspondence with the elements of G, and, for each $x \in X$, and each $g \in G$, a directed edge labelled x connecting g to gx. Any word w labels a unique path γ_w from the identity vertex to the vertex corresponding to the group element represented by w; this path is a loop precisely if w represents the identity. If $u = v$ is an equation in a finite defining set R, then uv^{-1} labels a loop from any vertex to itself; we shall call such loops *relation cells*.

We can easily give a geometric interpretation of the basic reduction moves which we described in the introduction. Applying a relation $u = v$ to w corresponds in the Cayley graph to a modification of the path γ_w by replacing a section of it which goes part of the way around a relation cell (along u) by a section going around the remaining part (along v) instead. Applying the rules of free cancellation corresponds to the addition or deletion of a spur which simply goes along an edge, and then back again along the same edge. So when w does indeed represent the identity, then solving the word problem basically involves contracting the loop of the Cayley graph by pulling it across relation cells, one at a time. The subgraph of the Cayley graph consisting of γ_w and all the cells which form a part of its contraction is called a *van Kampen diagram* for w.

We define the *area* of a van Kampen diagram for w to be the number of relation cells it contains and its *diameter* to be the maximum of the minimal distances from each interior vertex to the path γ_w. When $w \to w_1 \to \ldots \to w_n = \epsilon$ is an associated chain of reductions of w to the empty word, for which the reduction of w_i to w_{i+1} involves just a single relation cell, we define the filling length of the chain of reductions to be the maximum of the lengths of all the w_i's. Now we define the *Dehn function* $\delta(n)$, the *isodiametric function* $d(n)$ and the *filling length* $\phi(n)$ to be the maximum over all such w of length n of the minimum values of these measures over all van Kampen diagrams for w. Although each of these measures is defined relative to a particular presentation, it is not hard to show that if the presentations are changed then each of the measures varies only within an equivalence class of the relation $\simeq$ which relates f and g when a constant c exists for which both $f(n) \leq cg(cn + c) + cn + c$ and $g(n) \leq cf(cn + c) + cn + c$.

The Dehn function is commonly used as a measure of the complexity of the word problem maybe because it suggests a solution; if w of length n represents the identity then it is equal in the free group to a product of conjugates of at most $\delta(n)$ conjugates of *relators* uv^{-1} (for $u = v \in R$), and hence an exhaustive search, taking time bounded by an exponential of $\delta(n)$, solves the problem. It is proved in [18] that the above representation of w also provides a solution to the word problem for G (and also for any finitely generated subgroup) on a non-deterministic Turing machine in time exactly $\delta(n)$. Furthermore, it is proved in [1] that any finitely generated group whose word problem can be solved on a non-deterministic Turing machine in polynomial time embeds as a finitely generated subgroup of a finitely presented group with polynomial Dehn function.

A wide variety of Dehn functions can be found. It is proved [18, 4] that the set of exponents α for which n^α is a Dehn function forms a dense subset of the set $\{1\} \cup [2, \infty)$, and that each of the functions $\epsilon_i(n)$ in the sequence defined by $\epsilon_i(n) = 2^{\epsilon_{i-1}(n)}$, $\epsilon_0(n) = n$ is a Dehn function [5, 9].

It is fairly easy to see that the Dehn function provides (up to $\simeq$ equivalence) an upper bound on the isodiametric function. It is also proved (by both Gersten and Cohen) that $A^{B^{d(n)}}$ is an upper bound on the Dehn function, and conjectured that a single exponential bound exists. Other connections are explored in [9, 10, 12] and elsewhere.

More on Dehn's algorithm

When Dehn's algorithm applies, it provides a solution to the word problem in linear time. In fact it applies to a wider class of groups than those originally studied by Dehn, precisely to any group with a linear Dehn function. Groups of this type, extensively studied by Gromov and many others, are known as word hyperbolic groups. The fundamental groups of compact hyperbolic manifolds give examples, but there are many others besides. The Cayley graph of such a group displays properties normally associated with negative curvature, and it is this which is exploited by Dehn's algorithm.

The Chomsky hierarchy of models of computation

Recall that the word problem for G over X is soluble if $W(G, X)$ is recognised by a halting Turing machine. Turing machines form the most powerful model of computation, one of four in the Chomsky hierarchy.

A *finite state automaton* is probably best understood as a finite directed graph whose edges are labelled by elements of a finite alphabet. A string is accepted by such a machine if it labels a path from the *start* vertex (state) to one of the subset of *accept* vertices (states). It is straightforward to prove that a group has word problem recognised by a finite state automaton if and only if it is finite; the Cayley graph provides the appropriate automaton.

A *pushdown automaton* is basically a finite state automaton to which a stack has been added as memory. A beautiful result due to Muller and

Schupp [15] proves that the groups with word problem recognised by push-down automata are precisely those which are virtually free.

A *linear bounded automaton* is a Turing machine for which the length of each tape used is bounded by a linear function of the input length. Groups with linear filling function are easily seen to have word problem recognised by machines of this type, and many examples of this type can be found.

Generalised Dehn algorithms

A recent preprint of Cannon, Goodman and Shapiro [6] modifies Dehn's algorithm by allowing reduction rules which include additional symbols not in the generating set X; it still works in linear time. This seemingly minor modification has a dramatic effect. In addition to solving the word problem for some more 'negatively curved' groups, the fundamental groups of geometrically finite (not necessarily compact) hyperbolic manifolds, it actually provides a linear time solution to the word problem for nilpotent groups. This may seem surprising since the Dehn function for nilpotent groups of class c is polynomial of degree at most (and sometimes exactly) $c+1$ [11].

A *real-time* Turing machine is defined (see [17]) to be a Turing machine which processes its input in real-time. A single tape contains the input and is scanned once only. After reading an input symbol, the machine may perform one operation on each of its (finitely many) work tapes. It must halt and give its decision when it reaches the end of the input. The family of real-time languages is a natural subset of those recognised in linear time.

Any Dehn algorithm and many generalised ones can run in real-time [13, 14]. In particular this is true of the word problem for nilpotent groups.

References

[1] J.-C.Birget, A.Yu Ol'shanskii, E.Rips, M.V.Sapir, Isoperimetric functions of groups and computational complexity of the word problem, to appear in Ann. of Maths.

[2] William W.Boone, Certain simple, unsolvable problems of group theory I-VI, Indag. Math. 16 (1954), 231–237, 492–497; 17 (1955), 252–256; 19 (1957) 22–27, 227–232.

[3] William W.Boone, The word problem, Proc. Nat. Acad. Sci. USA 44 (1958) 1061–1065.

[4] N.Brady, Martin R.Bridson, There is only one gap in the isoperimetric spectrum, Geom. Func. Anal. 10 (2000) 1053–1070

[5] Martin R.Bridson, The geometry of the word problem, to appear in Invitation to geometry and topology, OUP, 2002.

[6] J.Cannon, O.Goodman, M.Shapiro, Dehn's algorithm for non-hyperbolic groups, preprint.

[7] Max Dehn, Über die Topologie des dreidimensionalen Raumes, Math. Ann. 69 (1910), 137–168

[8] Max Dehn, Über unendliche diskontinuierliche Gruppen, Math. Ann. 71 (1912), 116–144.

[9] S.M.Gersten, Dehn functions and l_1-norms of finite presentations, MSRI Publ. 23, Springer, New York, 1992, 195–224.

[10] S.M.Gersten, Isoperimetric and isodiametric functions of finite presentations, LMS lecture notes 181, CUP, Cambridge, 1993, 79–96.

[11] S.M.Gersten, D.F.Holt, T.R.Riley, Isoperimetric inequalities for nilpotent groups, preprint.

[12] S.M.Gersten, T.Riley, Filling length in finitely presentable groups, to appear in Geo. Ded.

[13] Derek F.Holt, Word-hyperbolic groups have real-time word problem, Internat. J. Algebra Comput. 10 (2002), 221–227.

[14] Derek F.Holt, Sarah Rees, Solving the word problem in real time, J. London Math. Soc 63 (2001), 623–639.

[15] David Muller, Paul E.Schupp, Groups, the theory of ends, and context-free languages, J. Comput. System Sci. 26 (1983) 295–310

[16] P.S.Novikov, On algorithmic unsolvability of the problem of the identity (Russian) Doklady Akad. Nauk SSSR 85 (1952) 709–712

[17] Arnold L.Rosenberg, Real-time definable languages, J. Assoc. Comput. Mach. 14 (1967) 645–662.

[18] M.V.Sapir, J.-C.Birget, E.Rips, Isoperimetric and isodiametric functions of groups, to appear in Ann. of Maths.