Finitely Generated Semiautomatic Groups[1]

Sanjay Jain
Department of Computer Science, National University of Singapore, 13 Computing Drive, COM1, Singapore 117417, Republic of Singapore
sanjay@comp.nus.edu.sg

Bakhadyr Khoussainov
Department of Computer Science, University of Auckland, Private Bag 92019, Auckland, New Zealand
bmk@cs.auckland.ac.nz

Frank Stephan
Department of Mathematics and Department of Computer Science, National University of Singapore, 10 Lower Kent Ridge Road, S17, Singapore 119076, Republic of Singapore
fstephan@comp.nus.edu.sg

Abstract. The present work shows that Cayley automatic groups are semiautomatic and exhibits some further constructions of semiautomatic groups; as a consequence, various undecidability results for Cayley automatic groups hold also for finitely generated semiautomatic groups. However, in spite of their generality, the word problem of finitely generated semiautomatic groups is still solvable in quadratic time. Furthermore, the present work establishes that every finitely generated group of nilpotency class three is semiautomatic. It remains open whether the finitely generated semiautomatic groups are a strict superclass of the Cayley automatic groups.

Keywords: nilpotent groups, automatic groups, automatic structures, semiautomatic structures, Cayley automatic groups, semigroups, finitely generated groups.

Dedication

S. Barry Cooper was a leading researcher in Recursion Theory and, in his later years, was a driving force to extend this field to an area represented in the conferences "Computability in Europe" and "Theory and Applications of Models of Computation" [7]; this larger area of Computability in general should comprise all research which targets at computation and information from a theoretical point of view. While Cooper remained active in recursion theory itself and also worked with one of the authors of this paper [1] about topics from the difference hierarchy, he also reached out to neighbouring fields [4, 5], in particular when organising the two conference series. The present work is a representative item of his vision of computability in a more general sense than just recursion theory. This work deals with computation from the viewpoint of non-uniform usage of finite automata when describing functions like group operations and futhermore uses tools from the classical theory of computation such as coding of NP-complete problems or, more generally, Diophantine sets in order to obtain the result that certain groups cannot be semiautomatic in the sense described. Each of the authors had multiple interactions with Cooper and are in particular grateful for the support which they got when organising TAMC 2015 in Singapore; in his last year of life, Cooper came to Singapore to attend the conference and to present a special session talk. The authors would like to dedicate this article to the memory of S. Barry Cooper (9/10/1943–26/10/2015).

[1]F. Stephan (PI) and S. Jain (Co-PI) are supported in part by Singapore Ministry of Education Academic Research Fund Tier 1 grants R146-000-181-112 and R252-000-534-112 as well as Tier 2 grants MOE2013-T2-1-062 / R146-000-184-112 and MOE2016-T2-1-019 / R146-000-234-112. Additionally, S. Jain was supported in part by NUS grant C252-000-087-001. B. Khoussainov is supported in part by above mentioned grant MOE2016-T2-1-019 / R146-000-234-112 (as a collaborator) and by the Marsden Fund grant of the Royal Society of New Zealand. A conference version of this paper was presented at CiE 2016 [10]. Furthermore, F. Stephan acknowledges the funding of JSPS and NUS (R146-000-192-133 and R146-000-192-733) for the joint IMS-JSPS Workshop on Mathematical Logic and the Foundations of Mathematics in 2016 where he presented a survey on the joint results of this and other papers in the area of semiautomatic groups and semigroups.

1. Introduction

Hodgson [8, 9] as well as Khoussainov and Nerode [13] initiated the study of automatic structures, including that of groups. In their approach, such a group is given by a regular set A as the domain (denoting the representatives of the group) such that both, the group operation $\circ$ and the equality $=$, are automatic; that is, there is an automaton which reads the convoluted tuples (x, y, z) or (x, y) and decides whether such a tuple satisfies $x \circ y = z$ or $x = y$, respectively. Here, for $x = x_1 x_2 \ldots x_m$ and $y = y_1 y_2 \ldots y_n$ with $x_i, y_i \in \Sigma$, the convolution of the pair (x, y) is the string

$$\binom{x_1}{y_1}\binom{x_2}{y_2}\cdots\binom{x_{\max\{m,m\}}}{y_{\max\{m,n\}}}$$

over the new alphabet $(\Sigma \cup \{\#\})^2$, where x_i (respectively y_i) is taken to be $\#$ in case of $i > m$ (respectively, $i > n$). The convolution over triples or tuples in general is defined similarly. The advantage of this setting is that every function and relation definable in the language of group theory using parameters from the group is again automatic. Furthermore, automata providing the mappings can be found algorithmically. This also leads to the conclusion that for every fixed automatic group, the first-order theory is decidable [13]. Furthermore, automatic functions are precisely those which can be computed in linear time by a position-faithful one-tape Turing machine [3], thus the automatic functions coincide with the smallest reasonable time complexity class for functions.

Epstein, Cannon, Holt, Levy, Paterson and Thurston [6] argued that in the above formalisation, automaticity is, at least from the viewpoint of finitely generated groups, too restrictive. They furthermore wanted that the representatives of the group elements are given as words over the generators, leading to more meaningful representatives than arbitrary strings. Their concept of automatic groups led, for finitely generated groups, to a larger class of groups, though, by definition, it of course does not include groups which require infinitely many generators; groups with infinitely many generators, to some extent, were covered in the notion of automaticity by Hodgson, Khoussainov and Nerode. Nies and Thomas [16, 17] provide results which contrast and compare these two notions of automaticity and give an overview on results for groups which are automatic in the sense of Hodgson, Khoussainov and Nerode.

Kharlampovich, Khoussainov and Miasnikov [12] generalised the notion further to Cayley automatic groups. Here a finitely generated group $(A, \circ)$ is Cayley automatic iff the domain A is a regular set, for every group element there is a unique representative in A and, for every $a \in A$, the mapping $x \mapsto x \circ a$ is automatic. Note that the above requires multiplication by constants to be automatic only from one side; when multiplication by a constant from both sides are automatic, then the group is called Cayley biautomatic.

Finitely generated Cayley automatic groups have word problem decidable in quadratic time, carrying over the corresponding result from the two previous versions of automaticity. As opposed to the case of automatic groups (in the original sense of Hodgson), Miasnikov and Šunić [15] showed that several natural problems like the conjugacy problem can be undecidable for some Cayley automatic groups.

Jain, Khoussainov, Stephan, Teng and Zou [11] investigated the general approach where, in a structure for some relations and functions, it is only required that the versions of the functions or relations with all but one variable fixed to constants is automatic. Here the convention is to put the automatic domains, functions and relations before a semicolon and the semiautomatic relations after the semicolon. For a group, the semiautomatic group $(A, \circ; =)$ would be a structure where the domain A is regular, the group operation (with both inputs) is automatic and for each fixed element $a \in A$ the set $\{b \in A : b = a\}$ is regular — note that group elements might have several representatives in semiautomatic groups.

In the present work, for any group, ε represents the neutral element. One of the basic results obtained is that the notion $(A, \circ; =)$ collapses to an automatic group (in the sense of Hodgson, Khoussainov and Nerode), as

$$a = b \Leftrightarrow \exists c\, [a \circ c = \varepsilon \text{ and } b \circ c = \varepsilon].$$

For semiautomatic groups, the two interesting group structures are $(A, =; \circ)$ and $(A; \circ, =)$. In the first one, the equality is automatic, while in the second one, both the group operation and the equality are only semiautomatic. If a group is finitely generated, then the definition of being Cayley biautomatic is the same as having a presentation of the form $(A, =; \circ)$.

Finitely generated semiautomatic groups share with the other types of automatic groups one important property: The word problem can be decided in quadratic time and the algorithm is the same as known for the Cayley automatic groups [12]. Thus finitely generated groups with an undecidable or very complex word problem are not semiautomatic.

One of the problems left open in the present work is the following: Is every finitely generated semiautomatic group Cayley automatic? More generally, does every semiautomatic group admit a presentation where the multiplication with a fixed group element from one side only is semiautomatic and the equality is automatic?

2. Basic Facts and Examples

Finitely generated groups are groups for which there is a set $a_1, \ldots, a_n$ of generators such that every element of the group can be expressed as a finite word over the a_i and a_i^{-1} where a word like $a_1 a_1 a_2 a_1^{-1}$ then stands for $a_1 \circ a_1 \circ a_2 \circ a_1^{-1}$. A group $(A, \circ)$ has nilpotency class 3 if for all elements x_1, y_1, y_2, y_3 in the group it holds that the element x_4 of the sequence given inductively for $k = 1, 2, 3$ by

$$x_{k+1} = x_k \circ y_k \circ x_k^{-1} \circ y_k^{-1}$$

is the neutral element ε of A. Thus there are some rules to move one element over another element by generating some spin-off element, say

$$a_i \circ a_{i'} = a_{i'} \circ a_i \circ b_j$$

and the elements of the form $x \circ y \circ x^{-1} \circ y^{-1}$ generate a commutative subgroup B of the group. Furthermore, B is a normal subgroup of A, that is, for all $x \in B$ and $y \in A$, the element $y \circ x \circ y^{-1}$ is also in B. The factor group $A/B = \{x \circ B : x \in A\}$ with $x = y$ if $x \circ B = y \circ B$ as sets is also commutative and finitely generated; here $(x \circ B) \circ (y \circ B) = (x \circ (y \circ B \circ y^{-1})) \circ (y \circ B) = (x \circ y) \circ B \circ B = (x \circ y) \circ B$ and these equalities use that $B = y \circ B \circ y^{-1}$ for all $y \in A$. Thus A/B is isomorphic to a finite product of the form

$$\mathbb{Z}^r \times \{0, 1, \ldots, p_{r+1} - 1\} \times \ldots \times \{0, 1, \ldots, p_n - 1\}$$

for some $n, r \leqslant n$ and $p_{r+1}, \ldots, p_n \geqslant 2$. This is used in the construction of Theorem 8. Note that if $a_i^{p_n} = \varepsilon$ in the factor group, it means that $a_i^{p_n} \in B$ in the original group B and the automatic mappings for multiplication with generators have to take care of this fact.

In the case of a finitely generated group, it can be shown that, for every fixed $a \in A$, the mappings $x \mapsto x \circ a$ and $x \mapsto a \circ x$ are automatic by proving this fact only for the generators and their inverses. This fact is used at several places in the paper.

Furthermore, for a semiautomatic group with generators $a_1, a_2, \ldots, a_n$, it can be checked in quadratic time whether a word w over these generators is equal to the neutral element ε. The idea is to start with a representative of ε and then, from the front to the end of the word, multiply the representative with the corresponding generator or its inverse depending on the symbol of the word read. When the whole word is processed, a finite automaton is used for checking whether the group element is equal to ε. That the algorithm is in quadratic time stems from the fact that each of the automatic functions involved makes the word only longer by a constant amount of symbols and that the running of the automatic functions is in linear time. Thus, the maximum length is bounded by a constant multiplied with the length of the word involved. This gives the overall quadratic bound of the decision algorithm. The details are more or less the same as in the case of Cayley automatic groups as well as Thurston automatic groups and follow the known proof [6, 12].

The constructions in Theorem 8 and elsewhere use that there is a copy of the integers $(\mathbb{Z}, +, =, <)$ which is automatic including equality and order; this is well-known since the beginning of automatic structures [8, 13]. The automatic representation follows the well-known algorithm for verifying an addition in the examples

```
  3 8 3 8 8 2 8 3            1 2 3 4
+   1 2 3 4 5 6 7          +     5 6
= 3 9 6 2 2 8 5 0          = 1 2 9 0
```

where the algorithm goes from the end to the front, adding the two top digits and taking the carry digit into account. For automatic structures where the strings start in the front at the same position — instead of ending at the same position as the numbers in the example — one just codes the numbers in the reverse order, so 1234 would be stored as 4321 and 56 as 65 so that at the start the first two digits 4,6 have to be added, and then the next digits 3,5 plus carry and so on.

In summary, the natural numbers can be coded as binary or decimal numbers in reverse order so that the symbol a_n of the input always refers to the digit of 2^n or 10^n, respectively. For uniqueness, leading zeroes are not allowed in the representation. Then the usual binary or decimal addition of such numbers invokes an addition and the following automatic order is the order on the decimals: $a_0a_1 \dots a_n < b_0b_1 \dots b_m$ iff $n < m$ or $n = m$ and the largest k with $a_k \neq b_k$ has that $a_k < b_k$ in the order of the digits. Having addition and order on the binary or decimal natural numbers, the representation can be extended to that of $(\mathbb{Z}, +, =, <)$ by representing each integer z as a pair of two natural numbers (x, y) with $z = x - y$. Then $(x_1, y_1) = (x_2, y_2)$ iff $x_1 + y_2 = x_2 + y_1$ what can be checked in this representation. Multiplication of an integer z with a fixed convoluted vector like $(1, 3, 2)$ can be implemented as a constant repeated adding $(z, 0, 0) + (0, z, 0) + (0, z, 0) + (0, z, 0) + (0, 0, z) + (0, 0, z)$ for proof purposes; the actual automatic structure can apply slightly more efficient algorithms. Similarly, a convoluted tuple $(z_1, \dots, z_k)$ can be mapped to $v_1 \cdot z_1 + v_2 \cdot z_2 + \dots + v_k \cdot z_k$ for fixed vectors $v_1, \dots, v_k$ in $\mathbb{Z}^h$ for fixed h. The finitely generated automatic groups (in sense of Khoussainov and Nerode) are fully characterised as those which are given by finite index extension of an Abelian finitely generated group [17, 18]; thus all such groups are also automatic in the sense of Thurston. Note for this characterisation, that if a finitely generated group is a finite index extension of some subgroup, then the subgroup which it extends is also finitely generated [19, page 55]. An example for a nilpotent Cayley biautomatic group, that is, a finitely generated semiautomatic group of the form $(A, =; \circ)$, is the group of all unitriangular $n \times n$ matrices over $\mathbb{Z}$.

Example 1. *Consider the group of all matrices of the form*

$$\{\begin{pmatrix} 1 & a & d & f \\ 0 & 1 & b & e \\ 0 & 0 & 1 & c \\ 0 & 0 & 0 & 1 \end{pmatrix} : a, b, c, d, e, f \in \mathbb{Z}\}$$

which is generated by the three generators of the form

$$\begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

and their inverses

$$\begin{pmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

This group is Cayley biautomatic and has nilpotency class 3*; the reason is that the commutators* $x^{-1} \circ y^{-1} \circ x \circ y$ *of two upper unitriangular matrices* x, y *have* 0 *in the semidiagonal next to the main diagonal and the commutator of three elements have* 0 *in the first two semidiagonals next to the main diagonal and that the commutators of four elements are always the identity matrix. It can be represented by convoluted tuples of the form* (a, b, c, d, e, f) *where* a, b, c, d, e, f *are integers in any given semiautomatic representation* $(\mathbb{Z}, +, =; \cdot)$ *of the ring of integers.*

Note that not every subgroup of the above has a regular domain. For example the subgroup generated by

$$\begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

and their inverses consists of group elements of the form

$$\begin{pmatrix} 1 & a & d & f \\ 0 & 1 & b & e \\ 0 & 0 & 1 & c \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

where $a = c$, $a \cdot b = d + e$ and some constraint on f holds (on when f has to be odd and when it has to be even). The equation $a \cdot b = d + e$ implies that this group is not a regular subset of the representation of the unitriangular matrices from Example 1. These dependencies make it difficult to exploit on one hand the Cayley biautomaticity of the unitriangular matrices in their natural representation as a group and on the other hand to embed a given nilpotent group into this group in a way that its domain is a regular subset of the group in the matrix representation. It is conjectured that this is impossible and that there is no way to overcome this; that is, that there are finitely generated groups of nilpotency class 3 which are not Cayley automatic. However, one can generalise the just defined matrix group to have a more general example.

Example 2. *For a given number n of generators with $n \geqslant 3$, let the group G_n be represented by vectors of 4×4 matrices over the integers of the form*

$$\begin{pmatrix} 1 & m_i & m_{i,j} & m_{i,j,k} \\ 0 & 1 & m_j & m_{j,k} \\ 0 & 0 & 1 & m_k \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

where each coordinate is given by a triple of indices (i, j, k) with $1 \leqslant i < j < k \leqslant n$ of coordinates and the group operation is the component wise matrix multiplication; in a vector of matrices representing a group element, if two different matrices contain, perhaps at different positions, entries with the same indices, then the corresponding numbers have to be the same. For example, all group members in G_4 are of the form

$$\left(\begin{pmatrix} 1 & m_1 & m_{1,2} & m_{1,2,3} \\ 0 & 1 & m_2 & m_{2,3} \\ 0 & 0 & 1 & m_3 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & m_1 & m_{1,2} & m_{1,2,4} \\ 0 & 1 & m_2 & m_{2,4} \\ 0 & 0 & 1 & m_4 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & m_1 & m_{1,3} & m_{1,3,4} \\ 0 & 1 & m_3 & m_{3,4} \\ 0 & 0 & 1 & m_4 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & m_2 & m_{2,3} & m_{2,3,4} \\ 0 & 1 & m_3 & m_{3,4} \\ 0 & 0 & 1 & m_4 \\ 0 & 0 & 0 & 1 \end{pmatrix} \right)$$

and they satisfy that all numbers called m_2 in these matrices have the same value and also all numbers called $m_{3,4}$ have the same value. The generators are all the vectors a_h where $m_h = 1$ and all other entries (m_i with $i \neq h$, $m_{i,j}$ and $m_{i,j,k}$) are 0. For example, in G_4, $a_1 \circ a_1 \circ a_2$ is

$$\left(\begin{pmatrix} 1 & 2 & 2 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 2 & 2 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 2 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \right).$$

This group has nilpotency class 3. The group G_n is a semiautomatic group where equality is automatic and the group operation is semiautomatic; as a representation one takes a convolution of all the m_i, $m_{i,j}$, $m_{i,j,k}$ which, in turn, are *all represented in a fixed representation of the automatic group of integers. The group G_n is not the free group of nilpotency class 3: a_1 and $a_1^{-1} \circ a_2^{-1} \circ a_1 \circ a_2$ commute although they would not do it in the free group of nilpotency class 3.*

3. Constructions of Semiautomatic Groups

Recall that a Cayley automatic group is a finitely generated group $(A, \circ)$ where the domain A is regular, every group element has a unique representative in A and for every $a \in A$, the mapping $x \mapsto x \circ a$ is automatic. This notion is equivalent to allowing multiple representatives in A for the group elements, but additionally requiring that equality is automatic.

A semiautomatic finitely generated group where the equality is automatic is called Cayley biautomatic, Example 2 is an example of a Cayley biautomatic group. Miasnikov and Šunić [15] showed that there are Cayley automatic groups which are not Cayley biautomatic. They also showed that there are Cayley automatic groups for which the conjugacy problem is undecidable and that the isomorphism problem is also undecidable for the class of Cayley automatic groups.

Berdinsky and Khoussainov [2] have shown that every Baumslag Solitar group is Cayley automatic and Jain, Khoussainov, Stephan, Teng and Zou [11] announced that every Baumslag Solitar group is semiautomatic. This and other results follow now from a general transfer theorem which shows that every Cayley automatic group is semiautomatic.

Theorem 3. *If $(A, =; \{x \mapsto x \circ a : a \in A\})$ is a Cayley automatic group, then the group has a semiautomatic presentation $(B, x \mapsto x^{-1}; \circ, =)$; in this presentation the domain is regular and the inversion is an automatic function, whereas the equality and the group operation are semiautomatic.*

Proof. Given the Cayley automatic group A as in the statement of the theorem, let $B = \{(v, w) : v, w \in A\}$ be the set of convoluted pairs (v, w), where the pair (v, w) stands for the element $v^{-1} \circ w$ of A. Now $(v, w) \circ (\varepsilon, u) = v^{-1} \circ w \circ \varepsilon^{-1} \circ u = v^{-1} \circ w \circ u = (v, w \circ u)$, $(u, \varepsilon) \circ (v, w) = u^{-1} \circ \varepsilon \circ v^{-1} \circ w = (v \circ u)^{-1} \circ w = (v \circ u, w)$ and $(v, w) \circ (w, v) = v^{-1} \circ w \circ w^{-1} \circ v = \varepsilon$. As every fixed element $a \in A$ can be represented by either (ε, a) or (a^{-1}, ε), multiplication with a fixed group element from either side is automatic. Furthermore, the mapping $(v, w) \mapsto (w, v)$ is automatic and computes the inverse. The set of representations of a fixed element $a \in A$ is the set $\{(v, w) : (a, \varepsilon) \circ (v, w) = (\varepsilon, \varepsilon)\} = \{(v, w) : v \circ a = w\}$, where the latter set is easily seen to be regular. □

The above result shows that the undecidability results for Cayley automatic groups by Miasnikov, Šunić and Ventura [15, 20] generalise to finitely generated semiautomatic groups.

Corollary 4. *There is a semiautomatic group $(A; \circ, =)$ for which the conjugacy problem is undecidable. Furthermore, the isomorphism problem for semiautomatic groups is undecidable.*

The next result shows that all semiautomatic groups can have an automatic inversion.

Proposition 5. *Every semiautomatic group $(A; \circ, =)$ has a further semiautomatic presentation $(B, x \mapsto x^{-1}; \circ, =)$.*

Proof. The proposition is proven by introducing two new symbols, $+$ and $-$, such that $B = \{+, -\} \cdot A$ consists of all $+x$ representing $x \in A$ and $-x$ representing x^{-1} for $x \in A$. The inverse is now computed by the function interchanging $+$ and $-$. For fixed $a \in A$, $x \mapsto x \circ a$ becomes $+x \mapsto +(x \circ a)$ and $-x \mapsto -(a^{-1} \circ x)$, $x \mapsto a \circ x$ is implemented similarly and $\{+x : x = a\} \cup \{-x : x = a^{-1}\}$ is the regular set of representatives of a in B. □

Theorem 6. *Assume that $(A, \circ, =)$ is an automatic group (in the sense of Hodgson, Khoussainov and Nerode), $(B; \circ, =)$ is a semiautomatic group and $\{\phi_b : b \in B\}$ is a family of homomorphisms from A to A such that for each*

$b, b' \in B$, $\phi_{b \circ b'}(a) = \phi_b(\phi_{b'}(a))$ and each ϕ_b is an automatic mapping, then the semidirect product $A \rtimes_\phi B$ is also a semiautomatic group (with both equality and group operation being semiautomatic). Here the group operation in $A \rtimes_\phi B$ is defined by $(a,b) \circ (a',b') = (a \circ \phi_b(a'), b \circ b')$.

Proof. Consider the representation set $C = \{(a,b,\tilde{a}) : a, \tilde{a} \in A \text{ and } b \in B\}$, where $(a,b,\tilde{a}) \in C$ stands for $(a,\varepsilon) \circ (\varepsilon,b) \circ (\tilde{a},\varepsilon)$ in the group $A \rtimes_\phi B$. Now, for a fixed $a' \in A$, $b' \in B$ and arbitrary $(a,b,\tilde{a}) \in C$, the multiplications are defined as follows:

$$(a,b,\tilde{a}) \circ (a',\varepsilon) \mapsto (a,b,\tilde{a} \circ a');$$

$$(a',\varepsilon) \circ (a,b,\tilde{a}) \mapsto (a' \circ a, b, \tilde{a});$$

$$(a,b,\tilde{a}) \circ (\varepsilon,b') \mapsto (a, b \circ b', \phi_{b'^{-1}}(\tilde{a}));$$

$$(\varepsilon,b') \circ (a,b,\tilde{a}) \mapsto (\phi_{b'}(a), b' \circ b, \tilde{a}).$$

The last mapping is derived from the following equations $(\varepsilon,b') \circ (a,\varepsilon) \circ (\varepsilon,b) \circ (\tilde{a},\varepsilon) = (\varepsilon,b') \circ (a,\varepsilon) \circ (\varepsilon,b'^{-1}) \circ (\varepsilon,b') \circ (\varepsilon,b) \circ (\tilde{a},\varepsilon) = (\varepsilon,b') \circ (\varepsilon,b'^{-1}) \circ (\phi_{b'}(a),\varepsilon) \circ (\varepsilon,b' \circ b) \circ (\tilde{a},\varepsilon) = (\phi_{b'}(a),\varepsilon) \circ (\varepsilon,b' \circ b) \circ (\tilde{a},\varepsilon)$. Note that multiplying with $(a',b',\tilde{a}')$ in C can be defined using the above as $(a',b',\tilde{a}') = (a',\varepsilon) \circ (\varepsilon,b') \circ (\tilde{a}',\varepsilon)$. Now, all the four mappings above are automatic as they only use homomorphisms from B, which are automatic, and multiplication with fixed group elements in the basic groups A and B, which are automatic. It follows that $\circ$ is semiautomatic in C.

For equality, note that $(a,b,\tilde{a}) = (a',b',\tilde{a}')$ iff $b = b'$ (in group B) and $a \circ \phi_b(\tilde{a}) = a' \circ \phi_{b'}(\tilde{a}')$ (in group A). Thus, for a fixed $(a',b',\tilde{a}') \in C$ and any $(a,b,\tilde{a}) \in C$, $(a,b,\tilde{a}) = (a',b',\tilde{a}')$ iff $b = b'$ (in group B) and $a \circ \phi_{b'}(\tilde{a}) = a' \circ \phi_{b'}(\tilde{a}')$. As $\phi_{b'}$, $\circ$ restricted to A and equality in A are automatic, it follows that equality is semiautomatic in C. □

It can also be shown that the free product of finitely many semiautomatic groups is semiautomatic. The construction is very much inline with the one of Kharlampovich, Khoussainov and Miasnikov [12] for Cayley automatic groups with some adjustments for semiautomaticity.

Theorem 7. *The free product of finitely many semiautomatic groups is semiautomatic.*

Proof. Let $(A_1;\circ,=),\ldots,(A_n;\circ,=)$ be semiautomatic groups which all share the empty word ε as neutral element and use disjoint alphabets to represent the other elements. Note that, for each fixed $a \in A_k$, there is an automatic mapping $x \mapsto a \circ x$ (for $x \in A_k$) such that the length of $a \circ x$ is at most a constant more than the length of x. Let $\#$ be a symbol not appearing in the alphabets of $A_1,\ldots,A_k$. Now each member of the free product B of $A_1,\ldots,A_k$ is a word of the form $\#^+u_1\#^+u_2\#^+\ldots\#^+u_m\#^+$ with $u_1,\ldots,u_m$ representing elements different from ε and no two subsequent u_h, u_{h+1} are from the same group A_k. Any word from $\#^+$ denotes the neutral element of the group. It is sufficient to show that the multiplication with any fixed element from $A_1 \cup A_2 \cup \ldots \cup A_n - \{\varepsilon\}$ is automatic, multiplying with ε can be realised by the identity function. Consider $a \in A_k - \{\varepsilon\}$.

Now $x \mapsto a \circ x$ is given as follows: x is mapped to $\#a\#$ in the case that $x \in \#^+$; x is mapped to $\#a\#x$ in the case that the first component u_1 from x is not from A_k; x is mapped to the word x', where u_1 is replaced by $\#^{|u_1|}$ in the case that $u_1 = a^{-1}$; otherwise x is mapped to the word x', where u_1 is replaced by a word from $(a \circ u_1)\#^*$. To ensure automaticity of the mapping, in the last two cases above, enough $\#$'s are filled in to make sure that length of x and x' do not differ by more than a constant (independent of x).

The mapping $x \mapsto x \circ a$ is given as follows: x is mapped to $\#a\#$ in the case that $x \in \#^+$; x is mapped to $x\#a\#$ in the case that the last component u_m of x is not from A_k; the last part of the form $u_m\#^+$ is erased from x by the mapping in the case that $u_m = a^{-1}$ and the last part $u_m\#^+$ is replaced by $(u_m \circ a)\#$ in the case that $u_m \in A_k - \{a^{-1}\}$.

Furthermore, for comparing whether x of the form $\#^+u_1\#^+\ldots\#^+u_n\#^+$ represents a fixed element $\#a_1\#a_2\#\ldots\#a_m\#$, consider the automaton consisting of m distinct subautomatons: the h-th subautomaton checks whether the component u_h of x is from the same A_k as a_h and has the same value; the automaton accepts iff all these checks succeed and the number of components n of x is exactly m. □

4. Nilpotent Groups

Kharlampovich, Khoussainov and Miasnikov [12] showed that finitely generated groups of nilpotency class 1 or 2 are Cayley automatic. The next theorem uses semiautomatic groups in place of Cayley automatic groups and pushes the above result one step further. As it is open whether all the finitely generated groups of nilpotency class 3 are Cayley automatic, this result provides possible candidates for separating the two notions within the finitely generated groups.

Theorem 8. *Every finitely generated group of nilpotency class 3 can be represented as a semiautomatic group* $(A; \circ, =)$.

Proof. Let $a_1, \ldots, a_n$ be the finitely many generators in the original nilpotent group.

Consider the factor group of the given group over the quotient group generated by all elements of the form $x \circ y \circ x^{-1} \circ y^{-1}$. This group is Abelian and is isomorphic to

$$\mathbb{Z}^r \times \{0, 1, \ldots, p_{r+1} - 1\} \times \ldots \times \{0, 1, \ldots, p_n - 1\}$$

for some $r \leqslant n$ and natural numbers $p_{r+1}, \ldots, p_n \geqslant 2$.

Let $b_1, \ldots, b_{n'}$ be all the group elements of the form $a_i^{-1} \circ a_{i'}^{-1} \circ a_i \circ a_{i'}$ and let $c_1, \ldots, c_{n''}$ be all the group elements of the form $a_i^{-1} \circ b_j^{-1} \circ a_i \circ b_j$ or $b_j^{-1} \circ a_i^{-1} \circ b_j \circ a_i$. As the group is of nilpotency class three and as the b_i are commutators, the $c_1, \ldots, c_{n''}$ are "commutators of second degree" and commute with all group elements. Furthermore, for each i, j there is a k with

$$a_i \circ b_j = b_j \circ a_i \circ c_k, \; a_i \circ b_j^{-1} = b_j^{-1} \circ a_i \circ c_k^{-1}$$

and that for each i, i' there are j, k with

$$a_i \circ a_{i'} = a_{i'} \circ a_i \circ b_j, \; a_i \circ a_{i'}^{-1} = a_{i'}^{-1} \circ a_i \circ b_j^{-1} \circ c_k.$$

Similar rules allow to move a_i^{-1} over $a_{i'}, b_j$. Note that, as the group has nilpotency class 3, the elements c_k commute with all group elements. Thus, Furthermore, if $a_i \circ b_j = b_j \circ a_i \circ c_k$, then $b_j \circ a_i = a_i \circ b_j \circ c_k^{-1}$ and $a_i^{-1} \circ b_j = b_j \circ a_i^{-1} \circ c_k^{-1}$. Note that the group elements $b_j, b_{j'}$ also commute with each other, as when, for example, $b_{j'} = a_i^{-1} \circ a_{i'}^{-1} \circ a_i \circ a_{i'}$ then $b_j \circ b_{j'} = b_j \circ a_i^{-1} \circ a_{i'}^{-1} \circ a_i \circ a_{i'} = a_i^{-1} \circ a_{i'}^{-1} \circ a_i \circ a_{i'} \circ b_j = b_{j'} \circ b_j$. The reason is that the $c_k, c_{k'}$ produced by moving $a_i^{-1}, a_{i'}^{-1}$, respectively, over b_j, are cancelled out when moving $a_i, a_{i'}$ over b_j. Now, each group element is given by a convoluted tuple of integers

$$(m_1, \ldots, m_n, m'_1, \ldots, m'_{n'}, m''_1, \ldots, m''_{n''})$$

where, for $i = r+1, \ldots, n$, $m_i \in \{0, 1, \ldots, p_i - 1\}$. The above member of A represents

$$a_1^{m_1} \circ \ldots \circ a_n^{m_n} \circ b_1^{m'_1} \circ \ldots \circ b_{n'}^{m'_{n'}} \circ c_1^{m''_1} \circ \ldots \circ c_{n''}^{m''_{n''}}.$$

Note that several tuples of this type may represent the same group element due to products of some b_j and c_k being ε.

In the representation set A, the integers m_i and m'_j in the above are represented in binary, with the reverse ordering of the bits to allow automatic addition of components. Furthermore, each m''_k is represented as a convoluted tuple $(h_0, h_1, \ldots, h_n)$ of integers (in binary using reverse ordering of the bits) satisfying

$$m''_k = h_0 + h_1 \cdot m_1 + \ldots + h_n \cdot m_n,$$

The number of integers used in the overall representation described above is $n+n'+(n+1)\cdot n''$, which is a constant independent of the group element; therefore convolution can indeed be used to represent the group element.

Now it will be shown that multiplication with a fixed a_i is automatic and that equality is semiautomatic.

First, for automaticity of the multiplication with a fixed element, note that it is sufficient to show that multiplication with a fixed generator from $a_1, a_1^{-1}, \ldots, a_n, a_n^{-1}$ is automatic, as every other group element is a fixed product of these. This is shown in several steps, the number of steps is constant and each step is automatic. For showing that the mapping $x \mapsto a_i \circ x$ is automatic, it is now described how, $a_i \circ a_{i'}^{m_{i'}} a_{i'+1}^{m_{i'+1}} \ldots a_n^{m_n} b_1^{m'_1} \ldots b_{n'}^{m'_{n'}} c_1^{m''_1} \ldots c_{n''}^{m''_{n''}}$ is updated to $a_{i'}^{m_{i'}} \circ a_i \circ a_{i'+1}^{m_{i'+1}} \ldots a_n^{m_n} b_1^{s'_1} \ldots b_{n'}^{s'_{n'}} c_1^{s''_1} \ldots c_{n''}^{s''_{n''}}$, where $i' < i$ and $m''_k = (h_0^k, h_1^k, \ldots, h_n^k)$. Repeatedly using this mechanism to shift a_i over $a_1^{m_1} a_2^{m_2} \ldots a_{i-1}^{m_{i-1}}$ and then showing how to handle the increase of m_i by 1, gives the multiplication by a_i for any member of the group as represented in A. Now suppose $1 \leqslant i' < i$, $1 \leqslant q \leqslant n$, and $m_{i'} > 0$. There are $j, k_1, \ldots, k_n$ such that $a_i a_{i'} = a_{i'} a_i b_j$ and $b_j a_q = a_q b_j c_{k_q}$. Then, the following operations are done to update m'_j and various m''_k to obtain the corresponding s'_j and s''_j (values not updated are unchanged).

(a) $m_{i'}$ is added to m'_j (to handle the increase in the b_j).

(b) $m_{i'}(m_{i'}-1)/2$ is added to $m''_{k_{i'}}$ (to handle the increase in $c_{k_{i'}}$ due to moving of b_j generated in (a) over $a_{i'}^{m_{i'}}$). If $m_{i'}$ is odd, then the above addition can be achieved by adding $(m_{i'}-1)/2$ to $h_{i'}^{k_{i'}}$. If $m_{i'}$ is even, $m_{i'}(m_{i'}-1)/2 = m_{i'}(m_{i'}-2)/2 + m_{i'}/2$. Thus, the above addition can be achieved by adding $m_{i'}/2$ to $h_0^{k_{i'}}$ and adding $(m_{i'}-2)/2$ to $h_{i'}^{k_{i'}}$.

(c) $m_{i'} * m_q$ is added to m''_{k_q}, for $q = i'+1, \ldots, n$ (to handle the increase in c_{k_q} due to moving of $b_j^{m_{i'}}$ over $a_q^{m_q}$). This can be done by adding $m_{i'}$ to $h_q^{k_q}$.

Note that $a_i \circ a_{i'}^{-1} = a_{i'}^{-1} \circ a_i \circ b_j^{-1} \circ c_{k'}$ for some k' which permits to handle the case when $m_{i'} < 0$ in a similar manner. For the multiplication

$$a_i \circ a_i^{m_i} a_{i+1}^{m_{i+1}} \ldots a_n^{m_n} b_1^{m'_1} \ldots b_{n'}^{m'_{n'}} c_1^{m''_1} \ldots c_{n''}^{m''_{n''}},$$

one updates m_i to m_i+1 and, as a chain reaction, for $k = 1, \ldots, n''$, update h_0^k to $h_0^k - h_i^k$, for the tuple $(h_0^k, h_1^k, \ldots, h_n^k)$ representing m''_k (so that the new value of m_i is used rather than the older value in the computation of m''_k).

Similarly it can be shown that also the mappings $x \mapsto a_i^{-1} \circ x$, $x \mapsto x \circ a_i$ and $x \mapsto x \circ a_i^{-1}$ are automatic.

The above handles all multiplications by a_i on the left except for the case of $i > r$ and $m_i + 1 = p_i$. To handle this, an additional multiplication by $a_i^{-p_i}$ can be done using the above mechanism to bring the power of a_i to 0.

Now, for showing semiautomaticity of equality, for any fixed element

$$a_1^{m_1} \circ \ldots \circ a_n^{m_n} \circ b_1^{m'_1} \circ \ldots \circ b_{n'}^{m'_{n'}} \circ c_1^{m''_1} \circ \ldots \circ c_{n''}^{m''_{n''}} \in A$$

it is shown that the set of its representatives in A is regular. Note that in the vector of the exponents, for each further representative of the group element, the first n coordinates must also be $m_1, m_2, \ldots, m_n$, which can be easily checked. In the case that these n coordinates are equal, one can tailormake an automaton to check for equality. The automaton can, for each k and for the coordinates $(h_0, h_1, \ldots, h_n)$ representing m''_k, use the formula

$$h_0 + h_1 \cdot m_1 + \ldots + h_n \cdot m_n$$

to get the explicit value corresponding to m''_k in the other representative, in binary notation, as multiplication of integers by constants can be done automatically. However, the m'-coordinates and m''-coordinates can be different for the two representatives. The difference of these coordinates must, however, give a vector representing ε. Thus, it suffices to give a test for neutrality in the m'-coordinates and m''-coordinates in order to be able to decide equality to the fixed given element. Call a set $\{v_1, \ldots, v_r\}$ of vectors representing these coordinates to be independent over $\mathbb{Z}$ iff no v_h can be obtained from a linear combination of the others using coefficients from $\mathbb{Z}$. If one of the sets $\{v_1, v_2, \ldots, v_r\}$, $\{-v_1, v_2, \ldots, v_r\}$, $\{v_1 - v_2, v_2, \ldots, v_r\}$, $\{v_2 - v_1, v_2, \ldots, v_r\}$ is independent, then all of them are.

So Euclid's algorithm can be run on the vectors until all but one vector have a 0 in the first coordinate; then one can run the algorithm until, among all those vectors with a 0 in the first coordinate, all but at most one have a 0 in the second coordinate and so on. This implies that the number of independent vectors is not larger than the number of coordinates. Thus there are fixed vectors $\{v_1,\ldots,v_\ell\}$ such that two vectors

$$(m_1,\ldots,m_n,m'_1,\ldots,m'_{n'},m''_1,\ldots,m''_{n''}) \text{ and}$$
$$(m_1,\ldots,m_n,\tilde{m}'_1,\ldots,\tilde{m}'_{n'},\tilde{m}''_1,\ldots,\tilde{m}''_{n''})$$

represent the same element iff the difference

$$(0,\ldots,0,m'_1-\tilde{m}'_1,\ldots,m'_{n'}-\tilde{m}'_{n'},m''_1-\tilde{m}''_1,\ldots,m''_{n''}-\tilde{m}''_{n''})$$

is of the form $r_1\cdot v_1+r_2\cdot v_2+\ldots r_\ell\cdot v_\ell$ for some $r_1,\ldots,r_\ell\in\mathbb{Z}$. This is an existentially quantified formula, where the multiplication with fixed vectors (represented as convoluted tuples) can be done by an automatic function and their adding and comparing with the target as well. Thus the predicate whether the two vectors from above representing the two group elements are the same is automatic. Thus for each group element

$$a_1^{m_1}\circ\ldots\circ a_n^{m_n}\circ b_1^{m'_1}\circ\ldots\circ b_{n'}^{m'_{n'}}\circ c_1^{m''_1}\circ\ldots\circ c_{n''}^{m''_{n''}}$$

there is a finite automaton which decides whether another group element is equal to it. So the group $(A;\circ,=)$ is semiautomatic. □

For the representation used in the above theorem, by using the natural subgroup B of all elements in A generated by the b_j and c_k, the following Theorem 9 below can be shown. The key idea is to represent each group element in the form $b\circ a\circ\tilde{b}$ where $b,\tilde{b}$ are in B and a is either $a_1^{m_1}\circ\ldots\circ a_n^{m_n}$ or $a_n^{m_n}\circ\ldots\circ a_1^{m_1}$; these two orderings are used in order to facilitate inversion. Item **(b)** in the theorem below is proven by coding a computationally difficult problem into the theory of the structure of the group and then conclude that this problem would be solvable in the case that the given structure is semiautomatic.

Theorem 9. *In the following, $(A,\circ)$ denotes a finitely generated group of nilpotency class 3, B denotes the commutator subgroup generated by all elements of the form $x\circ y\circ x^{-1}\circ y^{-1}$ with $x,y\in A$ and $\bullet$ denotes the restriction of $\circ$ to the domain $(A\times B)\cup(B\times A)$.*

(a) *For every A as above, the structure $(A,B,x\mapsto x^{-1},\bullet;\circ,=)$ is semiautomatic.*
(b) *For some A as above, the structure $(A,B,\bullet,=;\circ)$ is not semiautomatic.*

Proof. (a): The notation from the proof of Theorem 8 is carried over for this proof. The group elements are represented as products $b\circ a\circ\tilde{b}$ where (i) $b,\tilde{b}$ are products of b_j's and c_k's represented in the same format as they are represented in Theorem 8 and (ii) a is a member of $(a_1^*a_2^*\ldots a_n^*\cup a_n^*\ldots a_2^*a_1^*)$ represented as a convoluted tuple $(m_0,m_1,\ldots,m_n)$, where $m_0\in\{-1,1\}$; the tuple $(m_0,m_1,\ldots,m_n)$ represents $a_1^{m_1}\circ\ldots\circ a_n^{m_n}$, if $m_0=1$, and $a_n^{m_n}\circ\ldots\circ a_1^{m_1}$, if $m_0=-1$.

The mappings $b\mapsto b^{-1}$, $a\mapsto a^{-1}$ and $\tilde{b}\mapsto\tilde{b}^{-1}$ are realised by negating all entries in the corresponding tuples; for mapping $(b\circ a\circ\tilde{b})$ to $(b\circ a\circ\tilde{b})^{-1}$, one has to exchange the entries of b and $\tilde{b}$ as well, as $(b\circ a\circ\tilde{b})^{-1}=\tilde{b}^{-1}\circ a^{-1}\circ b^{-1}$. Thus the mapping $x\mapsto x^{-1}$ (in the chosen representation) is automatic.

Note that, in the representation for $\hat{b}\in B$, all the m-coordinates are 0. Thus for the component m''_k in the representation of $\hat{b}$, the integers $h_1,h_2,\ldots$ (as in Theorem 8) can be ignored. Hence, the multiplication $\hat{b}\bullet(b\circ a\circ\tilde{b})$, can be done by adding m'_j coordinate of the representation of $\hat{b}$ to the corresponding m'_j coordinate of b and the h_0-component of the m''_k coordinate of $\hat{b}$ to the corresponding h_0-component of the m''_k coordinates of b. Similarly, when computing $(b\circ a\circ\tilde{b})\bullet\hat{b}$, the coordinates of $\hat{b}$ are added as above to those of $\tilde{b}$.

Note that, in the representation chosen for this proof, $\hat{b} \in B$ is actually a product of the form: $b' \circ \varepsilon \circ \tilde{b}'$, where $b', \tilde{b}'$ are represented as in Theorem 8. The coordinates of b' and $\tilde{b}'$ as above can be contracted to the coordinates of one member of B by simply adding, component-wise, prior to carrying out the multiplication $\bullet$ as described above. These arguments explain why $\bullet$ is an automatic function.

Multiplication of an element x with generators a_i from either side as done in Theorem 8, can easily be adjusted to the representation chosen here.

Now, assume a fixed group element $x = a_1^{m_1} a_2^{m_2} \ldots a_n^{m_n} \circ b'$, where $b' \in B$, is given. Let $a = a_1^{m_1} a_2^{m_2} \ldots a_n^{m_n}$. Note that for all representatives $y = b \circ a' \circ \tilde{b}$ of x in the representation chosen, the coordinates $m_1, m_2, \ldots, m_n$ corresponding to a' must match to that of a as above. Furthermore, there is a fixed element $\hat{b} \in B$ such that

$$a_n^{m_n} \circ \ldots \circ a_1^{m_1} = a_1^{m_1} \circ \ldots \circ a_n^{m_n} \circ \hat{b}.$$

Now, for any given representative $y = b \circ a' \circ \tilde{b}$ with the coordinates for a' being $m_0 = -1, m_1, \ldots, m_n$, the coordinate m_0 can be converted to $+1$, by replacing $\tilde{b}$ by $\hat{b} \circ \tilde{b}$. Furthermore, using the arguments given in the proof of Theorem 8, there is a fixed automatic homomorphism $\phi_a : B \to B$ such that $b \circ a = a \circ \phi_a(b)$. The products $\phi_a(b) \circ \tilde{b}$ or $\phi_a(b) \circ \hat{b} \circ \tilde{b}$ can be carried out by componentwise addition of the vectors involved. Once this is done, the algorithm from Theorem 8 can be used to compare y in the resulting representation with that of x. Thus equality is semiautomatic in the representation chosen.

For the proof of **(b)**, one starts with the free group $\hat{A}$ of nilpotency class 3 generated by $\hat{a}_1, \ldots, \hat{a}_7$. Let $\hat{B}$ denote the subgroup generated by $\{x^{-1} \circ y^{-1} \circ x \circ y : x, y \in \hat{A}\}$ and $\hat{C}$ denote the subgroup generated by $\{x^{-1} \circ y^{-1} \circ x \circ y : x \in \hat{B}, y \in \hat{A}\}$. For a suitable subgroup $\tilde{C}$ of $\hat{C}$ defined below, one chooses A by defining

$$A = \{x \circ \tilde{C} : x \in \hat{A}\}.$$

Furthermore, let B denote $\{x \circ \tilde{C} : x \in \hat{B}\}$ and C denote $\{x \circ \tilde{C} : x \in \hat{C}\}$. The choice of $\tilde{C}$ will be crucial. Note that one can make two c, c' from $\hat{C}$ to be equal in C by putting $c^{-1} \circ c'$ into $\tilde{C}$. Similarly, one can make c to be ε by putting c into $\tilde{C}$. Furthermore, one can make $c = c' \circ c''$ by putting $c^{-1} \circ c' \circ c''$ in $\tilde{C}$. The $\tilde{C}$ will be defined implicitly based on the above methods below.

The goal is to code a hard problem into the theory of the automatic structure $(A, B, \bullet, =; \circ)$. So one considers the structure which can check membership in A, membership in B, multiplication of two elements with one in B and one in A and equality; furthermore, multiplcation with fixed members from A from either side can also be used.

Now the following will be shown: For an NP-hard problem, one can choose A (by chosing $\tilde{C}$ appropriately) such that there is a automatic relation R in $B \times B \times B$ and a polynomial time function f mapping parameters (α, β, γ) to members of $B \times B \times B$ such that $f(\alpha, \beta, \gamma) \in R$ iff a corresponding instance of a fixed NP-complete problem can be solved. Since automatic relations can be solved in linear time [3], this would result in a proof that P = NP. Admittedly, the hypothesis P = NP is not yet refuted; however, a more complicated coding could do the same for any given Diophantine set and would result in a situation where automaticity permits to decide a nonrecursive Diophantine set, a contradiction. As the coding for this is much more involved, only the coding of the NP-hard problem is supplied in order to convince the reader that not every structure of the form $(A, B, \bullet, =; \circ)$ is semiautomatic. So the goal is to solve the following NP-hard set in the integers:

$$S = \{(\alpha, \beta, \gamma) : \exists \mu, \nu \in \mathbb{Z}\, [\mu^2 \leqslant \gamma^2 \text{ and } \mu^2 + \nu \cdot \beta = \alpha]\}.$$

Here the complexity of (α, β, γ) is measured in the number of bits needed for their binary represenation and the NP-hardness of this set was shown by Manders and Adleman [14]. Now a formula representing an automatic relation R will be defined along with special elements $b_1, c_1, c_2 \in B$ defined later such that $(c_1^{\alpha}, b_1^{\beta}, c_2^{\gamma^2}) \in R$ iff $(\alpha, \beta, \gamma) \in S$.

Note that due to $\bullet$ being automatic, one can compute for fixed $b \in B$ a power b^{δ} by using that the functions $b^{\delta} \mapsto b^{2\delta} = b^{\delta} \bullet b^{\delta}$ and $b^{\delta} \mapsto b^{2\delta+1} = b^{\delta} \bullet b^{\delta} \bullet b$ are both automatic and can be computed in linear time. Furthermore, the representatives of $b^{2\delta}$ and $b^{2\delta+1}$ are both only a constant longer than that of b^{δ}; this implies that the overall function $\delta \mapsto b^\delta$ can be computed in time polynomial in the number of digits of δ, as one starts with $u_0 = \varepsilon$ and one does, inductively for each digit $d = d_1, \ldots, d_\ell$ of δ, the update $u = u \bullet u \bullet b^d$, with $b^d = \varepsilon$ for $d = 0$ and $b^d = b$ for $d = 1$. Note that γ^2 can be computed beforehand in polynomial time from γ.

Recall that $a_1, \ldots, a_7$ are the generators of A. R will be defined by an existentially quantified formula which asks for the existence of an $x, x_1, x_2 \in A$ and $y, y_1, y_2 \in B$ satisfying the conditions laid out below. Note that

$$x \in a_1^{m_1} \circ a_2^{m_2} \circ \ldots \circ a_6^{m_6} \circ a_7^{m_7} \circ B$$

and similarly for x_1, x_2. Furthermore, for studying how the generator a_7 commutes with the other generators, let $b_1, \ldots, b_6$ denote the members of B satisfying

$$a_7 \circ a_1 = a_1 \circ a_7 \circ b_1, \ldots, a_7 \circ a_6 = a_6 \circ a_7 \circ b_6.$$

$\tilde{C}$ is chosen such that the following conditions are true for the degree of commutativity between $a_1, \ldots, a_6$ and $b_1, \ldots, b_6$:

$$\begin{aligned}
&b_1 \circ a_1 = a_1 \circ b_1 \circ c_1,\ b_2 \circ a_2 = a_2 \circ b_2 \circ c_1 \circ c_2,\\
&b_3 \circ a_3 = a_3 \circ b_3 \circ c_2,\ b_4 \circ a_4 = a_4 \circ b_4 \circ c_2,\\
&b_5 \circ a_5 = a_5 \circ b_5 \circ c_2,\ b_6 \circ a_6 = a_6 \circ b_6 \circ c_2.
\end{aligned}$$

Here by definition $c_1 = b_1^{-1} \circ a_1^{-1} \circ b_1 \circ a_1$ and $c_2 = b_3^{-1} \circ a_3^{-1} \circ b_3 \circ a_3$ and the other four equations redefine the corresponding commutators for a_i, b_i. Furthermore, one defines for all different $i, i', i'' \in \{1, \ldots, 7\}$ that

$$a_i^{-1} \circ (a_{i'}^{-1} \circ a_{i''}^{-1} \circ a_{i'} \circ a_{i''})^{-1} \circ a_i \circ (a_{i'}^{-1} \circ a_{i''}^{-1} \circ a_{i'} \circ a_{i''}) = \varepsilon$$

and this implies that $b_j \circ a_i = a_i \circ b_j$ for $i, j \in \{1, \ldots, 6\}$ with $i \neq j$. Now there is a $c \in C$ such that

$$a_7 \circ x = x \circ a_7 \circ b_1^{m_1} \circ b_2^{m_2} \circ \ldots \circ b_6^{m_6} \circ c.$$

It is required that y, y_1, y_2 depend on x, x_1, x_2 and each other as follows:

$$\begin{aligned}
&a_7 \circ x = (x \circ a_7) \bullet y,\ a_7 \circ x_1 = (x_1 \circ a_7) \bullet y_1,\\
&a_7 \circ x_2 = (x_2 \circ a_7) \bullet y_2,\ y = y_1 \bullet y_2.
\end{aligned}$$

The first of the following conditions is due to the choice of $m_1, \ldots, m_6$ and the second and third conditions will be imposed implicitly:

$$\begin{aligned}
&y \in a_1 \circ b_1^{m_1} \circ \ldots \circ b_6^{m_6} \circ C,\\
&y_1 \in a_1 \circ b_1^{m_1} \circ C,\\
&y_2 \in a_2 \circ b_2^{m_2} \circ \ldots \circ b_6^{m_6} \circ C
\end{aligned}$$

For the second and third conditions, one ensures that b_1 does not occur in y_2 and that $b_2, \ldots, b_6$ do not occur in y_1 by requiring the following commutativity conditions on y_1, y_2:

$$\begin{aligned}
&a_1 \circ y_2 = y_2 \circ a_1, a_2 \circ y_1 = y_1 \circ a_2, a_3 \circ y_1 = y_1 \circ a_3,\\
&a_4 \circ y_1 = y_1 \circ a_4, a_5 \circ y_1 = y_1 \circ a_5, a_6 \circ y_1 = y_1 \circ a_6.
\end{aligned}$$

In addition one requires that either a_7 does not occur in x, x_1, x_2 or a_7 commutes with $b_1, \ldots, b_6$. This is obtained by postulating, for $i = 1, \ldots, 6$ and $\tilde{x} = x, x_1, x_2$ and for all $\tilde{y} \in B$,

$$\text{if } a_i \circ \tilde{x} = (\tilde{x} \circ a_i) \bullet \tilde{y} \text{ then } a_7 \circ \tilde{y} = \tilde{y} \circ a_7.$$

Note that if a_7 occurs in $\tilde{x}$ then moving a_i over a_7 gives a member of $b_i^{-1} \circ C$ while all other commutators of a_i and $a_{i'}$ with $i' \in \{1, \ldots, 6\} - \{i\}$ do commute with a_7. If now $\tilde{y}$ is the member of B generated by moving a_i over x and $a_7 \circ \tilde{y} = \tilde{y} \circ a_7$ then either a_7 does not occur in $\tilde{x}$ and thus b_i does not occur in $\tilde{y}$ or a_7 and b_i commute. The above conditions together give that

$$b_1^\beta \bullet y_2 \bullet x = x \bullet b_1^\beta \bullet y_2 \bullet c_1^{\beta m_1 + m_2^2} \bullet c_2^{m_2^2 + m_3^2 + \ldots + m_6^2}$$

and thus one imposes the additional constraint that

$$b_1^\beta \bullet y_2 \bullet x = x \bullet b_1^\beta \bullet y_2 \bullet c_1^\alpha \bullet c_2^{\gamma^2}.$$

These conditions together enforce that $\beta m_1 + m_2^2 = \alpha$ and $m_2^2 + m_3^2 + \ldots + m_6^2 = \gamma^2$. Using the theorem that every natural number is the sum of four integer squares, one has that the solvability of the conditions is equivalent to the existence of integers m_1, m_2 with $\beta m_1 + m_2^2 = \alpha$ and $m_2^2 \leqslant \gamma^2$. Thus one has the following statement:

$(\alpha, \beta, \gamma) \in S$ iff $(c_1^\alpha, b_1^\beta, c_2^{\gamma^2}) \in R$ iff there are $x, x_1, x_2 \in A$ and $y, y_1, y_2 \in B$ such that

$$b_1^\beta \bullet y_2 \bullet x = x \bullet b_1^\beta \bullet y_2 \bullet c_1^\alpha \bullet c_2^{\gamma^2}$$

and the equations governing the specific form of the variables x, x_1, x_2, y, y_1, y_2 are satisfied, namely

$$a_7 \circ x = (x \circ a_7) \bullet y, \ a_7 \circ x_1 = (x_1 \circ a_7) \bullet y_1, \ a_7 \circ x_2 = (x_2 \circ a_7) \bullet y_2,$$
$$y = y_1 \bullet y_2,$$
$$a_1 \circ y_2 = y_2 \circ a_1, \quad a_2 \circ y_1 = y_1 \circ a_2, \quad a_3 \circ y_1 = y_1 \circ a_3,$$
$$a_4 \circ y_1 = y_1 \circ a_4, \quad a_5 \circ y_1 = y_1 \circ a_5, \quad a_6 \circ y_1 = y_1 \circ a_6$$

and, for $i = 1, \ldots, 6$ and $\tilde{x} = x, x_1, x_2$ and for all $\tilde{y} \in B$,

$$\text{if } a_i \circ \tilde{x} = (\tilde{x} \circ a_i) \bullet \tilde{y} \text{ then } a_7 \circ \tilde{y} = \tilde{y} \circ a_7.$$

This comprehensive formula defines that R is an automatic relation in the case that $(A, B, \bullet, =; \circ)$ is semiautomatic. If R would be automatic, P would be equal to NP. This shows that the assumption of R being automatic is unlikely. A more complicated construction could also code up an unsolvable Diophantine set which then would be solved if the corresponding structure is semiautomatic; the group and the formula required would, however, be much more complicated. Therefore the proof is here given by coding an NP-complete problem. □

5. Conclusion

The present work established that every Cayley automatic group is semiautomatic, thus permitting to prove that the semiautomatic groups have an undecidable isomorphism problem and an undecidable conjugacy problem. Prior work showed that finitely generated groups of nilpotency class 2 are on one hand Cayley biautomatic [12] and on the other hand not automatic [18]; however, the situation was left open for nilpotent groups of higher classes. The present paper shows that finitely generated groups of nilpotency class three are always semiautomatic. As one could not establish that they are always Cayley automatic, these groups form a natural candidate to separate these two notions. In general, the following questions are open at the point of writing of this paper:

- Is every finitely generated semiautomatic group Cayley automatic?
- More generally, does every semiautomatic group $(G; \circ, =)$ have a presentation in which the equality is automatic and, for each constant a, also the mapping $x \mapsto x \circ a$ is automatic?
- Are all finitely generated nilpotent groups semiautomatic?
- Are all finitely generated nilpotent groups Cayley automatic?
- Are all finitely generated nilpotent groups Cayley biautomatic?

The results in this paper give some progress towards these questions, but leave all of them open.

References

[1] Bahareh Afshari, George Barmpalias, S. Barry Cooper and Frank Stephan. Post's Programme for the Ershov Hierarchy. *Journal of Logic and Computation*, 17:1025–1040, 2007.

[2] Dimitry Berdinsky and Bakhadyr Khoussainov. On automatic transitive graphs. *Developments in Language Theory - Eighteenth International Conference*, DLT 2014, Ekaterinburg, Russia, August 26-29, 2014. Proceedings. *Springer LNCS* 8633:1–12, 2014.

[3] John Case, Sanjay Jain, Samuel Seah and Frank Stephan. Automatic functions, linear time and learning. *Logical Methods in Computer Science*, 9(3), 2013.

[4] S. Barry Cooper. Mathematics, metaphysics and the multiverse. *Computation, Physics and Beyond - International Workshop on Theoretical Computer Science*, WTCS 2012, Dedicated to Cristian S. Calude on the Occasion of His Sixtieth Birthday, Auckland, New Zealand, 21–24 February 2012, Revised Selected and Invited Papers. *Springer LNCS* 7160:252–267, 2012.

[5] S. Barry Cooper. The machine as data: a computational view of emergence and definability. *Synthese* 192(7):1955–1988, 2015.

[6] David B.A. Epstein, James W. Cannon, Derek F. Holt, Silvio V.F. Levy, Micheal S. Paterson and William P. Thurston. *Word Processing in Groups.* Jones and Bartlett Publishers, Boston, 1992.

[7] T.V. Gopal, Manindra Agrawal, Angsheng Li and S. Barry Cooper. A Roadmap to TAMC. *Theory and Applications of Models of Computation - Eleventh Annual Conference*, TAMC 2014, Chennai, India, 11–13 April 2014. Proceedings. *Springer LNCS* 8402:1–6, 2014.

[8] Bernard R. Hodgson. *Théories décidables par automate fini*. Ph.D. thesis, Département de mathématiques et de statistique, Université de Montréal, 1976.

[9] Bernard R. Hodgson. Décidabilité par automate fini. *Annales des sciences mathématiques du Québec*, 7(1):39–57, 1983.

[10] Sanjay Jain, Bakhadyr Khoussainov and Frank Stephan. Finitely generated semiautomatic groups. *Pursuit of the Universal, Twelfth Conference on Computability in Europe*, CiE 2016, Paris, France, 27 June - 1 July 2016, Proceedings. *Springer LNCS* 9709:282–291, 2016.

[11] Sanjay Jain, Bakhadyr Khoussainov, Frank Stephan, Dan Teng and Siyuan Zou. Semiautomatic structures. *Theory of Computing Systems*, 61(4):1254–1287, 2017.

[12] Olga Kharlampovich, Bakhadyr Khoussainov and Alexei Miasnikov. From automatic structures to automatic groups. *Groups, Geometry and Dynamical Systems*, 8(1):157–198, 2014.

[13] Bakhadyr Khoussainov and Anil Nerode. Automatic presentations of structures. *Logic and Computational Complexity, International Workshop*, LCC 1994, Indianapolis, Indiana, USA, October 13–16, 1994, Proceedings. *Springer LNCS*, 960:367–392, 1995.

[14] Kenneth L. Manders and Leonard Adleman. NP-complete decision problems for binary quadratics. *Journal of Computer and System Sciences* 16:168–184, 1978.

[15] Alexei Miasnikov and Zoran Šunić. Cayley graph automatic groups are not necessarily Cayley graph biautomatic. In, Dediu, A. H. and Martín-Vide, C. (eds.), *Language and Automata Theory and Applications - Sixth International Conference*, LATA 2012, A Coruña, Spain, March 5-9, 2012. Proceedings. *Springer LNCS*, 7183:401-407, 2012.

[16] André Nies. Describing Groups. *The Bulletin of Symbolic Logic*, 13(3):305-339, 2007.

[17] André Nies and Richard Thomas. FA-presentable groups and rings. *Journal of Algebra*, 320:569-585, 2008.

[18] Graham Oliver and Richard M. Thomas. Automatic presentations for finitely generated groups. *Twentysecond Annual Symposium on Theoretical Aspects of Computer Science* (STACS 2005), Stuttgart, Germany, Proceedings. *Springer LNCS*, 3404:693–704, 2005.

[19] John S. Rose. *A course on Group Theory.* Dover Publications, 2012.

[20] Zoran Šunić and Enric Ventura. The conjugacy problem in automaton groups is not solvable. *Journal of Algebra*, 364:148–154, 2012.