

From automatic structures to automatic groups

Olga Kharlampovich, Bakhadyr Khoussainov and Alexei Miasnikov*

Abstract. In this paper we introduce the concept of a Cayley graph automatic group (CGA group or graph automatic group, for short) which generalizes the standard notion of an automatic group. Like the usual automatic groups graph automatic ones enjoy many nice properties: these groups are invariant under the change of generators, they are closed under direct and free products, certain types of amalgamated products, and finite extensions. Furthermore, the word problem in graph automatic groups is decidable in quadratic time. However, the class of graph automatic groups is much wider than the class of automatic groups. For example, we prove that all finitely generated 2-nilpotent groups and Baumslag–Solitar groups $BS(1, n)$ are graph automatic, as well as many other metabelian groups.

Mathematics Subject Classification (2010). 20F10, 20F65.

Keywords. Automatic groups, Cayley graphs, word problem, nilpotent and metabelian groups.

Contents

1	Introduction	158
2	Finite automata	161
3	Automatic structures	163
4	Cayley graphs	166
5	Complete first-order theories of Cayley graphs	168
6	Cayley graph automatic groups: definitions and examples	170
7	Automatic versus graph automatic	173
8	The word and conjugacy problems	174
9	Universal Cayley graphs	177
10	Cayley graph automatic groups: constructions	178
11	Subgroups	185
12	Nilpotent Cayley graph automatic groups	185
13	Solvable graph automatic groups	191
14	Finitely generated FA presentable groups	193
	References	196

*The work of all three authors was partially supported by Marsden Fund of The Royal Society of New Zealand. The work of the third author was partially supported by the NSF grant DMS-0914773.

1. Introduction

Automata theory has unified many branches of algebra, logic, and computer science. These include group theory (e.g., automatic groups [16], branch and self-similar groups [2], [38]), the theory of automatic structures [8], [30], [29], [46]), finite model theory, algorithms and decidability, decision problems in logic [12], [43], model checking and verification [49]. In this paper we use finite automata in representation of infinite mathematical structures, emphasizing automata representations of groups via their Cayley graphs.

The idea of using automata to investigate algorithmic, algebraic and logical aspects of mathematical structures goes back to the work of Büchi and Rabin [12], [43]. They established an intimate relationship between automata and the monadic second order (MSO) logic, where, to put it loosely, automata recognizability is equivalent to definability in the MSO logic. Through this relationship Büchi proved that the MSO theory of one successor function on the set \mathbb{N} is decidable [12]. Rabin used automata to prove that the MSO theory of two successors is decidable [43]. The latter implies decidability of the first-order theories of many structures, for example: linear orders, Boolean algebras, Presburger arithmetic, and term algebras [43].

In 1995 Khossainov and Nerode, motivated by investigations in computable model theory and the theory of feasible structures, used finite automata for representation of structures [30], thus initiating the whole development of the theory of automatic structures (e.g. see [30], [8], [46], [45]). Here a structure is called *automatic* if it is isomorphic to a structure whose domain and the basic operations and relations are recognized by finite automata. Automaticity implies the following three fundamental properties of structures:

- (1) The first order theory of every automatic structure is uniformly decidable [30] [8];
- (2) The class of automatic structures is closed under definability (with parameters) in the first order logic and in certain extensions of it [8] [32] [35];
- (3) There is an automatic structure (a *universal automatic structure*) in which all other automatic structures are first-order interpretable [8].

There are many natural examples of automatic structures: some fragments of the arithmetic, such as $(\mathbb{N}; +)$, state spaces of computer programs, the linear order of the rational numbers, the configuration spaces of Turing machines. However, not that many groups are automatic in this sense. In particular (see Section 14), a finitely generated group is automatic (as a structure) if and only if it is virtually abelian.

In modern group theory there are already several ways to represent groups by finite automata. One of these is to consider finite automata with letter-by-letter outputs, known as Mealy automata. Every such automaton determines finitely many length preserving functions on the set of strings X^* over the alphabet X of the automaton. If these functions are permutations then they generate a group, called an *automata group*. Automata groups enjoy some nice algorithmic properties, for instance, decidability

of the word problem. These groups are also a source of interesting examples. For instance, the famous Grigorchuk group is an automata group. We refer to the book [3] for detail.

Another way to use finite automata in group representations comes from algorithmic and geometric group theory and topology. Ideas of Thurston, Cannon, Gilman, Epstein and Holt brought to the subject a new class of groups, termed *automatic groups*, and revolutionized computing with infinite groups (see the book [16] for details). The initial motivation for introducing automatic groups was two-fold: to understand the fundamental groups of compact 3-manifolds and to approach their natural geometric structures via the geometry and complexity of the optimal normal forms; and to make them tractable for computing.

Roughly, a group G generated by a finite set X (with $X^{-1} = X$) is automatic if there exists a finite automata recognizable (i.e., rational or regular) subset L of X^* such that the natural mapping $u \rightarrow \bar{u}$ from L into G is bijective, and the right-multiplication by each of the generators from X can be performed by a finite automata.

This type of automaticity implies some principal “tameness” properties enjoyed by every automatic group G :

- (A) G is finitely presented.
- (B) The Dehn function in G is at most quadratic.
- (C) There is a constant k such that the words from L (the normal forms) of elements in G which are at most distance 1 apart in the Cayley graph $\Gamma(G, X)$ of G are k -fellow travelers in $\Gamma(G, X)$.

Most importantly, as was designed at the outset, the word problem in automatic groups is easily computable (the algorithmic complexity of the conjugacy problem is unknown):

- (D) The word problem in a given automatic group is decidable in quadratic time.
- (E) For any word $w \in X^*$ one can find in quadratic time its representative in L .

Examples of automatic groups include hyperbolic groups, braid groups, mapping class groups, Coxeter groups, Artin groups of large type, and many other groups. In addition, the class of automatic groups is closed under direct sums, finite extensions, finite index subgroups, free products, and some particular amalgamated free products. Yet many classes of groups that possess nice representations and algorithmic properties fail to be automatic. Most strikingly, a finitely generated nilpotent group is automatic if and only if it is virtually abelian. To this end we quote Farb [17]: “The fact that nilpotent groups are not automatic is a bit surprising and annoying, considering the fact that nilpotent groups are quite common and have an easily solved word problem.” The book [16] and the survey by Gersten [18] also raise a similar concern, though they do not indicate what could be possible generalizations. In the view of the initial goals, nowadays we know precisely from Epstein–Thurston classification what are compact geometrisable 3-manifolds whose fundamental groups are automatic [16]. The upshot of this classification, is that the fundamental group of a

compact geometrisable 3-manifold M is automatic if and only if none of the factors in the prime decomposition of M is a closed manifold modeled on Nil or Sol. Thus, it turned out that the class of automatic groups is nice, but not sufficiently wide. In the geometric framework the quest for a suitable generalization comes inspired by the following “geometric” characterization of automatic groups as those that have a regular set of normal forms $L \subseteq X^*$ satisfying (C). Two main ideas are to replace the regular language L with some more general language, and keep the fellow traveller property, perhaps, in a more general form. Groups satisfying (C) with the formal language requirement of rationality weakened or eliminated entirely are called *combable*. In general combable groups are less amenable to computation than automatic groups. We refer to the work of Bridson [10] for an account of the relation between combable and automatic groups. On the other hand, a more relaxed fellow traveller property, called *asynchronous fellow travelers*, was introduced at the very beginning, see the book [16]. But Epstein and Holt [16] showed that the fundamental group of a closed Nil manifold is not even asynchronously automatic. Finally, a geometric generalization of automaticity, that covers the fundamental groups of all compact 3-manifolds which satisfy the geometrization conjecture, was given by Bridson and Gilman in [11]. However, as far as we know, these geometrically natural generalizations lose the nice algorithmic properties mentioned above.

In fact, from the algorithmic standpoint, the properties (A) and (B) can be viewed as unnecessary restrictions, depriving automaticity for a wide variety of otherwise algorithmically nice groups such as nilpotent or metabelian groups.

In this paper we propose a natural generalization of automatic groups and introduce the class of Cayley graph automatic groups. A finitely generated group G is called *Cayley graph automatic*, or *graph automatic* or *CGA* for short, if it satisfies the definition of an automatic group as above, provided the condition that the alphabet X is a set of generators of G , is removed. (Notice that a similar idea of representing group elements over an alphabet which is not a generating set for the group was used in [20] to extend the range of groups with a Dehn algorithm.) Equivalently, a finitely generated group is *graph automatic* if its Cayley graph is an automatic structure in the sense of Khossainov and Nerode. The former definition immediately implies that the standard automatic groups are graph automatic. However, there are many examples of graph automatic groups that are not automatic. These include Heisenberg groups, Baumslag–Solitar groups $BS(1, n)$, arbitrary finitely generated groups of nilpotency class two, and some nilpotent groups of higher class, such as unitriangular groups $UT(n, \mathbb{Z})$, as well as many metabelian groups and solvable groups of higher class, like $T(n, \mathbb{Z})$. Moreover, we do not have the restrictions (A) and (B) anymore. As in the case of automatic groups the class of graph automatic groups is closed under free products, direct sums, finite extensions, wreath products, and certain types of amalgamated products, etc. This shows that the class of graph automatic groups, indeed, addresses some concerns mentioned above, but whether the class is good enough remains to be seen. Firstly, we do not know if every finitely generated nilpotent group is graph automatic or not, in particular, the question if a finitely generated

free nilpotent group of class 3 is graph automatic is still open. Likewise, we do not know any geometric condition that would give a characterization of graph automatic groups similar to property (C). On a positive side though there is a crucial algorithmic result stating that again the word problem for graph automatic groups is decidable in quadratic time, so property (D) is preserved. Moreover, there is a new “logical” condition that gives a powerful test to check if a given group is graph automatic. Namely, the group G is graph automatic if and only if its Cayley graph $\Gamma(G, X)$ is first-order interpretable in an automatic structure or, equivalently, is interpretable in a fixed *universal* automatic structure (see above and also Section 9 for definitions and examples). In addition, there is a natural notion of a *graph biautomatic* group, which generalizes the standard class of biautomatic groups, with similar algorithmic properties. For instance, the conjugacy problem in graph biautomatic groups is decidable. In this case the proofs are simpler and more straightforward than in the classical one. It seems it might be a chance to address the old problem whether automaticity implies biautomaticity in this new setting, which might shed some light on the old problem itself, but presently this is a pure speculation.

We would like to mention that quite often the same algorithmic problems on automatic structures (say, groups) are approached differently in constructive model theory and geometric group theory. In this paper we also aim to present the main ideas and techniques in a unified form which makes them more available for use in both areas.

2. Finite automata

We start with some notation and basic definitions from finite automata theory. Let Σ be a finite alphabet. The set of all finite strings over Σ is denoted by Σ^* , λ is the empty string. Usually, we use variables u, v, w for strings. The length of a string u is denoted by $|u|$. For a set X , $P(X)$ is the set of all subsets of X . The cardinality of X is denoted by $|X|$.

Definition 2.1. A *nondeterministic finite automaton* (NFA for short) over Σ is a tuple (S, I, T, F) , where S is the set of *states*, $I \subseteq S$ is the set of initial states, T is the *transition function* $T: S \times \Sigma \rightarrow P(S)$, and $F \subseteq S$ is the set of *accepting states*.

We use the letter \mathcal{M} , possibly with indices, to denote NFA. One can visualize an NFA \mathcal{M} as a labelled graph, called the *transition diagram* of the automaton, such that the states of \mathcal{M} represent vertices of the graph, and there is a directed edge from a state s to a state q labelled by σ if and only if $q \in T(s, \sigma)$. These edges are called σ -*transitions*.

An NFA automaton \mathcal{M} is called *deterministic* if the set $T(s, \sigma)$ is a singleton for each $s \in S$, $\sigma \in \Sigma$, i.e., for each state s and each $\sigma \in \Sigma$ there only one σ -transition in \mathcal{M} at s .

Let \mathcal{M} be an NFA. A *run* of \mathcal{M} on the string $w = \sigma_1\sigma_2\dots\sigma_n$ is a sequence of states $s_1, s_2, \dots, s_n, s_{n+1}$ such that $s_1 \in I$ and $s_{i+1} \in T(s_i, \sigma_i)$ for all $i = 1, \dots, n$. The automaton might have more than one run on the string w . These runs of the automaton can be viewed as paths in the transition diagram labelled by w .

Definition 2.2. A given automaton \mathcal{M} accepts a string $w = \sigma_1\sigma_2\dots\sigma_n$ if \mathcal{M} has a run $s_1, s_2, \dots, s_n, s_{n+1}$ on w such that $s_{n+1} \in F$. The set of all strings $L(\mathcal{M})$ accepted by \mathcal{M} is called the *language accepted by \mathcal{M}* . A language $L \subseteq \Sigma^*$ is *FA recognizable* if there exists an NFA \mathcal{M} such that $L = L(\mathcal{M})$.

It is well known that the set of all NFA recognizable languages in Σ^* forms a Boolean algebra under the set-theoretic operations of union, intersection, and complementation; and every NFA recognizable language is also recognizable by a deterministic finite automata. For this reason, in general we do not distinguish between deterministic and non-deterministic automata, and refer to them as finite automata (FA for short). By the well-known Kleene theorem the class of FA recognizable languages is the same as the class of regular languages. Therefore, we often refer to FA recognizable languages also as to regular languages.

We now introduce automata recognizable relations over the set Σ^* . To explain, for a new symbol $\diamond \notin \Sigma$ put $\Sigma_\diamond = \Sigma \cup \{\diamond\}$. Then the *convolution* of a tuple $(w_1, \dots, w_n) \in \Sigma^{*n}$ is the string

$$\otimes(w_1, \dots, w_n)$$

of length $\max_i |w_i|$ over alphabet $(\Sigma_\diamond)^n$ defined as follows. The k 'th symbol of the string is $(\sigma_1, \dots, \sigma_n)$, where σ_i is the k 'th symbol of w_i if $k \leq |w_i|$ and \diamond otherwise. For instance, for $w_1 = aabaaab$, $w_2 = bbabbabbb$, and $w_3 = aab$, we have

$$\otimes(w_1, w_2, w_3) = \begin{pmatrix} a & a & b & a & a & a & b & \diamond & \diamond \\ b & b & a & b & b & a & b & b & b \\ a & a & b & \diamond & \diamond & \diamond & \diamond & \diamond & \diamond \end{pmatrix}.$$

Definition 2.3. The *convolution of a relation $R \subset \Sigma^{*n}$* is the relation

$$\otimes R = \{\otimes(w_1, \dots, w_n) \mid (w_1, \dots, w_n) \in R\}.$$

The convolution encodes relations in Σ^{*n} into usual languages but over the alphabet $(\Sigma_\diamond)^n$. This allows one to define *FA recognizable relations* in the standard way.

Definition 2.4. An n -ary relation $R \subset \Sigma^{*n}$ is *FA recognizable* if its convolution $\otimes R$ is *FA recognizable* in the alphabet $(\Sigma_\diamond)^n$.

Intuitively, a finite automaton recognizing an n -ary relation $R \subset \Sigma^{*n}$ can be viewed as a finite automaton with n heads. All heads read their own tapes and make

simultaneous transitions. Therefore, finite automata over the alphabet $(\Sigma_{\diamond})^n$ are also called *synchronous n -tape automata* on Σ . Sometimes, we refer to FA recognizable relations in Σ^{*n} as *regular relations*.

We will later on sometimes identify a tuple (w_1, \dots, w_n) with its convolution $\otimes(w_1, \dots, w_n)$, which is always clear from the context.

Example 2.5. The following linear orders on Σ^* are FA recognizable:

- The *lexicographic* order: $u \leq_{\text{lex}} v$ if and only if either $u = v$ or u is lexicographically less than v .
- The *prefix* order: $u \leq_{\text{pref}} v$ if and only if u is a prefix of v .
- The *length-lexicographic* order: $u \leq_{\text{lllex}} v$ if and only if either $|u| < |v|$ or $|u| = |v|$ and $u \leq_{\text{lex}} v$.

3. Automatic structures

In this section we introduce automatic structures, give several examples, and mention some known results that will be used in the sequent. By a *structure* \mathcal{A} we mean a tuple

$$(A; P_0^{n_0}, \dots, P_k^{n_k}, f_0^{m_0}, \dots, f_t^{m_t}),$$

where A is a set, called the domain of the structure \mathcal{A} ; each $P_i^{n_i}$ is a relation of arity n_i on A ; and each $f_j^{m_j}$ is a total operation of arity m_j on A . These relations and operations are often called *atomic*. The structure \mathcal{A} is *relational* if it contains no operations. Every structure \mathcal{A} can be transformed into a logically equivalent relational structure. This is done by replacing each atomic operation $f_j^{m_j} : A^{m_j} \rightarrow A$ with its graph:

$$\text{Graph}(f_j^{m_j}) = \{(a_1, \dots, a_{m_j}, a) \mid f_j^{m_j}(a_1, \dots, a_{m_j}) = a\}.$$

The sequence of symbols $P_0^{n_0}, \dots, P_k^{n_k}, f_0^{m_0}, \dots, f_t^{m_t}$ is called a *signature* of the structure. Below is one of the key definitions of this paper:

Definition 3.1. The structure $\mathcal{A} = (A; P_0^{n_0}, \dots, P_k^{n_k}, f_0^{m_0}, \dots, f_t^{m_t})$ is called *automatic* if the domain A , all the predicates $P_0^{n_0}, \dots, P_k^{n_k}$, and the graphs $\text{Graph}(f_0^{m_0}), \dots, \text{Graph}(f_t^{m_t})$ of all atomic operations in \mathcal{A} are FA recognizable.

Here are some examples of automatic structures:

Example 3.2. The structure $(1^*; S, \leq)$, where $S(1^n) = 1^{n+1}$ and $1^n \leq 1^m$ iff $n \leq m$ for $n, m \in \omega$, is automatic. In this example, the alphabet is unary.

Example 3.3. The structure $(\{0, 1\}^*; \leq_{\text{lex}}, \leq_{\text{pref}}, \leq_{\text{lllex}})$, where the orders are defined in Example 2.5, is automatic. Here, the alphabet is binary.

Example 3.4. The structure $(\text{Base}_k; \text{Add}_k)$, where $\text{Base}_k = \{0, 1, \dots, k-1\}^* \cdot \{1, \dots, k-1\}$, is automatic. In this example each word $w = x_0 \dots x_n \in \text{Base}_k$ is identified with the number

$$\text{val}_k(w) = \sum_{i=0}^n x_i k^i.$$

This gives a base- k -representation of natural numbers, where the least significant digit comes first. The predicate Add_k is the graph of the k -base addition of natural numbers, that is $\text{Add}_k = \{(u, v, w) \mid \text{val}_k(u) + \text{val}_k(v) = \text{val}_k(w)\}$. This structure is isomorphic to the natural numbers with addition $\mathcal{P} = \langle \mathbb{N}, + \rangle$, known as Presburger arithmetic.

Definition 3.5. A structure \mathcal{B} is *automata presentable* if there exists an automatic structure \mathcal{A} isomorphic to \mathcal{B} . In this case \mathcal{A} is called an *automatic presentation* of \mathcal{B} .

We would like to give several comments about this definition. The first is that an automatic presentation \mathcal{A} of a structure \mathcal{B} can be viewed as a finite sequence of automata representing the domain, the atomic relations, and operations of the structure. The sequence is finite. Hence, automatic presentations are just finite objects that describe the structure. The second is that if a structure \mathcal{B} has an automatic presentation, then \mathcal{B} has infinitely many automatic presentations. Finally, in order to show that \mathcal{B} has an automatic presentation, one needs to find an automatic structure \mathcal{A} isomorphic to \mathcal{B} . Thus, to show that \mathcal{B} does not possess an automatic presentation one needs to prove that for all automatic presentations \mathcal{A} all bijective mappings $f: \mathcal{B} \rightarrow \mathcal{A}$ fail to establish an isomorphism. In a logical formalism the definition of automaticity is a Σ_1^1 -definition in the language of arithmetic.

Every automatic structure has an automatic presentation over a binary alphabet. Indeed, if Σ is an alphabet of an automatic presentation of a structure, there exists an n such that $\Sigma \subseteq \{0, 1\}^n$. Hence, we can represent each symbol from Σ by a binary string of length n . This allows one to transform every FA-recognizable n -ary relation over Σ^* to an FA-recognizable relation over the set of all binary strings. Therefore, from now on, if need be, one can assume that we consider binary alphabets only. By this reason, the alphabets might not be mentioned explicitly.

Since we are mostly interested in the isomorphism types of the structures, we often abuse our definitions and refer to automata presentable structures as automatic structures. Below we give some examples of automatic (automata presentable) structures.

- The Boolean algebra \mathcal{B}_ω of finite and co-finite subsets of \mathbb{N} is automatic. To show this, we encode elements of \mathcal{B}_ω as finite binary strings in a natural way: if its i 's component is 0 then i is not in the set, otherwise it is, and the last digit shows that from this place on all the components are the same (0 or 1). For example, the string 01101101 represents the infinite set $\{1, 2, 4, 5, 7, 8, 9, 10, 11, \dots\}$ and the string 0110110 represents the finite set

$\{1, 2, 4, 5\}$. Under this encoding, \mathcal{B}_ω is automata presentable. The alphabet in this presentation of the Boolean algebra is binary.

- The additive group $(\mathbb{Z}, +)$ is automata presentable over a binary alphabet.
- Finitely generated abelian groups are automata presentable. This follows from the fact that such groups are finite direct sums of $(\mathbb{Z}, +)$ and finite abelian groups.
- Small ordinals of the form ω^n , where $n \in \mathbb{N}$, are automatic.

Below by \bar{x}_i we denote a tuple of variables, and by \bar{a}_i a tuple of constants from the domain A of a structure \mathcal{A} . For a first-order formula $\phi(\bar{x}_1, \dots, \bar{x}_k)$ in the language of \mathcal{A} , denote by $\phi(\mathcal{A})$ the set of all tuples $(\bar{a}_1, \dots, \bar{a}_k)$ in the structure \mathcal{A} that satisfy ϕ in \mathcal{A} . We now give the following definition that will be used in this paper quite often. For simplicity, we give it only for relational structures, in general, one can either replace a given structure by its relational analog, or extend the definition to accommodate operations (which is easy).

Definition 3.6. A relational structure $\mathcal{B} = (B; R_1, \dots, R_m)$ is *first-order interpretable* in a structure \mathcal{A} if there are formulas

$$\mathcal{D}(\bar{x}), \phi_1(\bar{x}_1, \dots, \bar{x}_{k_1}), \dots, \phi_m(\bar{x}_1, \dots, \bar{x}_{k_m})$$

of the first order logic in the language of \mathcal{A} such that:

- (1) all tuples $\bar{x}, \bar{x}_1, \dots, \bar{x}_{k_n}$ of variables have the same length, and
- (2) the structure $(D(\mathcal{A}); \phi_1(\mathcal{A}), \dots, \phi_m(\mathcal{A}))$ is isomorphic to \mathcal{B} .

The following is a foundational theorem in the study of automatic structures. The proof of the theorem follows from closure properties of finite automata under set-theoretic Boolean operations, the projection operation, and decidability of the emptiness problem for automata. Recall that the emptiness problem asks if there exists an algorithm to check if the language $L(\mathcal{M})$ of a given finite automaton \mathcal{M} is empty or not.

Theorem 3.7 (The Definability and Decidability Theorem, [8], [30], [32]). (1) *There is an algorithm that, given an automatic presentation of any structure \mathcal{A} and a first-order formula $\varphi(x_1, \dots, x_n)$, produces an automaton that recognizes the set $\phi(\mathcal{A})$.*

(2) *If a structure \mathcal{B} is first-order interpretable in an automatic structure \mathcal{A} then \mathcal{B} has an automatic presentation.*

(3) *The first-order theory of every automatic structure is decidable.* □

Note that there are several generalizations of this theorem for various extensions of the first order logic FO. For example, one can add to FO the following two quantifiers: \exists^ω (there exists infinitely many) and $\exists^{n,m}$ (there exists m many modulo n), denoting

the resulting logic as $\text{FO} + \exists^\omega + \exists^{n,m}$. The theorem above can be extended to this more powerful logic (one can use arbitrary formulas from $\text{FO} + \exists^\omega + \exists^{n,m}$ to interpret \mathcal{B} in \mathcal{A}) [45], see also [35] for more generalizations.

We will be using the theorem above without explicit references. In particular in the following case: Presburger arithmetic $\mathcal{P} = \langle \mathbb{N}, + \rangle$ is an automatic structure by Example 3.4, hence any structure definable in \mathcal{P} is automatic, this is how we prove some of our results. Furthermore, there are some *universal* automatic structures, i.e., automatic structures \mathcal{A} such that a structure \mathcal{B} has an automatic presentation if and only if it is first-order interpretable in \mathcal{A} . Consider the following two structures:

Example 3.8.

$$\mathbb{N}_2 = (\mathbb{N}; +, |_2),$$

where $+$ is the standard addition and $x|_2y \Leftrightarrow x = 2^k \& y = x \cdot z$ for some $k, z \in \mathbb{N}$.

Example 3.9.

$$\mathcal{M} = (\Sigma^*; R_a(x, y), x \preceq y, \text{el}(x, y))_{a \in \Sigma},$$

where Σ is a finite alphabet with $|\Sigma| \geq 2$, $R_a(x, y) \Leftrightarrow y = xa$, $x \preceq y \Leftrightarrow x$ is a prefix of y , $\text{el}(x, y) \Leftrightarrow |x| = |y|$. \square

The following theorem gives a pure model theoretic characterization of automatically presentable structures.

Theorem 3.10 ([8]). *The structures \mathbb{N}_2 and \mathcal{M} are universal automatic structures. In particular, \mathbb{N}_2 and \mathcal{M} are interpretable in each other.*

4. Cayley graphs

In the next section we will introduce automaticity into groups through their Cayley graphs. This section recalls the definition of Cayley graphs and some of their basic properties.

Let G be an infinite group generated by a finite set X . There exists the standard map from X^* onto G mapping the words v into the group elements \bar{v} . The *word problem* for G (with respect to X) is the following set:

$$W(G, X) = \{(u, v) \mid u, v \in X^* \& \bar{u} = \bar{v} \text{ in the group } G\}.$$

The word problem for G is *decidable* if there exists an algorithm that given two words $u, v \in X^*$ decides if $\bar{u} = \bar{v}$. It is not hard to see that decidability of $W(G, X)$ does not depend on a given finite set of generators for G .

The group G and the finite set X of generators determine the following graph, called a *Cayley graph of G* , and denoted by $\Gamma(G, X)$. The vertices of the graph are

the elements of the group. For each vertex g we put a directed edge from g to gx , where $x \in X$, and label the edge by x . Thus, $\Gamma(G, X)$ is a labelled directed graph.

We view a labelled directed graph $\Gamma = (V, E)$ with the labels of the graph from a finite set $\Sigma = \{\sigma_1, \dots, \sigma_n\}$ as the following structure:

$$(V, E_{\sigma_1}, \dots, E_{\sigma_n}),$$

where each E_{σ} , $\sigma \in \Sigma$, is a binary relation $\{(x, y) \mid (x, y) \in E \text{ and the label of } (x, y) \text{ is } \sigma\}$.

For the next lemma we need one definition from the theory of computable structures. We say that a Σ -labelled directed graph (V, E) is *computable* if its vertex set V can be enumerated $V = \{v_0, v_1, v_2, \dots\}$ in such a way that there exists an algorithm to decide for any $\sigma \in \Sigma$ and any two vertices $v_i, v_j \in V$ if there is an edge from v_i to v_j labelled by σ .

Here are some basic properties of the Cayley graph $\Gamma(G, X)$.

Lemma 4.1. *The Cayley graph $\Gamma(G, X)$ satisfies the following properties:*

- (1) *The graph is connected (strongly connected if $X^{-1} = X$), that is, between any two vertices of the graph there is a path (directed path) connecting one to another.*
- (2) *The out-degree and the in-degree of each node is $|X|$.*
- (3) *The graph is transitive, that is, for any two vertices g_1 and g_2 of the graph there exists an automorphism α of $\Gamma(G, X)$ such that $\alpha(g_1) = g_2$.*
- (4) *The group of (label respecting) automorphisms of $\Gamma(G, X)$ is isomorphic to G .*
- (5) *The graph $\Gamma(G, X)$ is computable if and only if the word problem in G is decidable.*

Proof. The first four statements of the lemma are standard. The last statement of the lemma needs an explanation. Assume that the word problem $W(G, X)$ in G is decidable. Then there exists an algorithm that, given any two words w and v over X , decides if $v = w$ in the group G . Now we construct the graph $\Gamma(G, X)$ as follows. The vertex set V of the graph consists of all words $v \in X^*$ such that any word w that is equal to v in G is length-lexicographically larger than or equal to v . Clearly, this set V of vertices is computable. Since the word problem is decidable in G , we can use the algorithm for the word problem to decide if there exists an edge from v_1 to v_2 labelled by $x \in X$. This shows that $\Gamma(G, X)$ is a computable graph. Assume that the Cayley graph $\Gamma(G, X)$ is computable. Then given any two words w_1 and w_2 in X^* one can effectively find two vertices v_1 and v_2 that represent w_1 and w_2 in the graph, respectively. Then $w_1 = w_2$ in the group G if and only if $v_1 = v_2$. Hence the word problem in G is decidable. \square

5. Complete first-order theories of Cayley graphs

In this section we collect some known results about first-order theories of Cayley graphs. This will have a direct relation with automaticity. Recall, that a complete theory T in the first order logic is called \aleph_1 -categorical if all models of T of cardinality \aleph_1 are isomorphic (see [24] for details). The lemma below shows that the elementary theory of every infinite Cayley graph is \aleph_1 -categorical. This result has two consequences. The first one is that decidability of the word problem for G is equivalent to decidability of the first order theory of the Cayley graph of G . The other one will be discussed in due course.

We start with a lemma that is true for all locally finite labelled directed graphs. In particular, the lemma can be applied to Cayley graphs. Recall that a (directed or not) graph is *locally finite* if every vertex of the graph has finitely many edges incident to it. Let Γ be a locally finite labelled and directed graph. For vertices x, y of the graph Γ denote by $d(x, y)$ the length of the shortest undirected path from x to y . Then the n -ball around a vertex x is the set

$$B_n(x) = \{y \mid d(x, y) \leq n\},$$

which may be viewed as a directed labelled subgraph of Γ .

Lemma 5.1. *Let Γ_1 and Γ_2 be locally finite labelled connected and directed graphs. Assume that a and b are vertices of Γ_1 and Γ_2 , respectively, such that for all $n \in \mathbb{N}$ there is an isomorphism from $B_n(a)$ to $B_n(b)$ sending a to b . Then there exists an isomorphism $\alpha: \Gamma_1 \rightarrow \Gamma_2$ such that $\alpha(a) = b$.*

Proof. Each of the n -balls $B_n(a)$ and $B_n(b)$ is a finite set. There are finitely many isomorphisms from $B_n(a)$ into $B_n(b)$ that send a to b . Denote this set by I_n . By assumption, $I_n \neq \emptyset$ for all $n \in \mathbb{N}$. The restriction of each isomorphism $\alpha \in I_n$ onto $B_{n-1}(a)$ gives an isomorphism $\alpha' \in I_{n-1}$. It is easy to see that the set $I = \cup_{n \in \mathbb{N}} I_n$ can be turned into a finitely branching infinite tree, where every automorphism $\alpha \in I_n$ is connected by an edge to the isomorphism $\alpha' \in I_{n-1}$. Now by König's lemma there is an infinite path P in the tree, which determines an isomorphism from Γ_1 to Γ_2 that sends a to b . \square

Recall that we view the Cayley graph $\Gamma(G, X)$ of a group G generated by a finite set $X = \{x_1, \dots, x_n\}$ as a structure

$$(V, E_{x_1}, \dots, E_{x_n}),$$

where each $E_x, x \in X$, is a binary relation on V such that uE_xv if and only if there is an edge from u to v in $\Gamma(G, X)$ with label x .

Lemma 5.2. *Let G be an infinite group generated by a finite set X . The elementary theory of the Cayley graph $\Gamma(G, X)$ is \aleph_1 -categorical.*

Proof. Fix an arbitrary element g of G and consider the n -ball $B_n(g)$ in the Cayley graph $\Gamma(G, X)$. Since $\Gamma(G, X)$ is transitive $B_n(g')$ is isomorphic to $B_n(g)$ for all $g' \in G$. The elementary theory of the graph $\Gamma(G, X)$ contains the following sentences:

- (1) The sentence $\Phi_{n,m}$ stating that there are m distinct elements x such that $B_n(x)$ is isomorphic to $B_n(g)$, where $m, n \in \mathbb{N}$. Since G is infinite the sentence $\Phi_{m,n}$ holds in $\Gamma(G, X)$ for arbitrary $m, n \in \mathbb{N}$.
- (2) The sentence Φ_n , $n \in \mathbb{N}$, stating that for all x there is an isomorphism from $B_n(x)$ onto $B_n(g)$ that sends x to g .

Denote by $T(G, X)$ the theory with axioms $\{\Phi_{n,m}, \Phi_n \mid m, n \in \mathbb{N}\}$. The theory $T(G, X)$ has the Cayley graph $\Gamma(G, X)$ as its model, since the graph $\Gamma(G, X)$ satisfies all the axioms of the theory. Hence $T(G, X)$ is consistent. Axioms $\Phi_{m,n}$ also imply that all models of $T(G, X)$ are infinite. Our goal is to show that any two models \mathcal{A} and \mathcal{B} of this theory are isomorphic in case \mathcal{A} and \mathcal{B} have cardinality \aleph_1 .

We note that both \mathcal{A} and \mathcal{B} are labelled directed locally finite infinite graphs. As undirected graphs they are unions of connected components. Note that each such component in the graphs \mathcal{A} and \mathcal{B} is countable since the graphs are locally finite. This implies that both \mathcal{A} and \mathcal{B} are disjoint unions of their countable connected components and the cardinality of the union is \aleph_1 .

Now, we show that any two components of \mathcal{A} and \mathcal{B} are isomorphic. Indeed, take two elements $a \in \mathcal{A}$ and $b \in \mathcal{B}$, respectively. By the axioms of $T(G, X)$, for each $n \in \mathbb{N}$ there is an isomorphism from $B_n(a)$ to $B_n(b)$ that maps a to b . Apply the lemma above to build an isomorphism from the component of a onto the component of b . This shows that all components of the graphs \mathcal{A} and \mathcal{B} are pairwise isomorphic. Therefore, we match the components of \mathcal{A} with components of \mathcal{B} , and build an isomorphism from \mathcal{A} to \mathcal{B} . Thus, $T(G, X)$ is an \aleph_1 -categorical theory. \square

It is worth to note that the lemma stays true if we remove the labels from the edges of the Cayley graph $\Gamma(G, X)$. Namely, let $\Gamma_u(G, X)$ be the directed graph obtained from $\Gamma(G, X)$ by removing the labels from all the edges. Then the theory of the unlabelled graph $\Gamma_u(G, X)$ is \aleph_1 -categorical. One can go even further and consider the undirected unlabelled version of $\Gamma(G, X)$ – the result still holds.

The lemma above allows us to address decidability of the word problem for the group G in terms of decidability of the theory $T(G, X)$.

Theorem 5.3. *The word problem $W(G, X)$ in G is decidable if and only if the theory $T(G, X)$ of the Cayley graph $\Gamma(G, X)$ is decidable.*

Proof. Assume that the word problem in G is decidable. Our goal is to show that the theory $T(G, X)$ is also decidable. It is clear that $T(G, X)$ is effectively axiomatizable by the sentences $\Phi_{n,m}$ and Φ_n as follows from the proof of the lemma above. It is known that every \aleph_1 -categorical theory T without finite models is complete, that is,

for any sentence ϕ either ϕ belongs to T or $\neg\phi$ belongs to T [24]. From the lemma above, we conclude that $T(G, X)$ is a complete first order theory. Since $T(G, X)$ is complete, for every ϕ either ϕ or $\neg\phi$ is deducible from the axioms $\Phi_{n,m}$ and Φ_n . This implies decidability of $T(G, X)$.

Assume that the theory $T(G, X)$ is decidable. Clearly, $\Gamma(G, X)$ is a model of T . Now we use the result of Harrington (and independently Khisamiev) that states the following. If T is \aleph_1 -categorical decidable theory then all of its countable models are computable [23] [27]. We conclude that $\Gamma(G, X)$ is also a computable model of $T(G, X)$ ¹. This proves the theorem. \square

We would like to make the following comments and give references on the results of this section. It is proved in the paper [1] that the theory of any given infinite Cayley graph is strongly minimal. This implies \aleph_1 -categoricity of the theory of each Cayley graph. However, no connection is made between decidability of the word problem in the group and decidability of the theory. Kuske and Lohrey establish this connection in [34]. They prove that the word problem in a finitely generated group is decidable if and only if the theory of the Cayley graph of the group is decidable. Moreover, it is shown that the theory of the graph is exponentially more difficult than the word problem and that this bound is sharp.

6. Cayley graph automatic groups: definitions and examples

In this section we introduce labelled automatic graphs and present several examples. Let $\Gamma = (V, E)$ be a labelled directed graph. The labels of the graph are from a finite set $\Sigma = \{\sigma_1, \dots, \sigma_n\}$.

Definition 6.1. We view the graph Γ as the following structure:

$$(V, E_{\sigma_1}, \dots, E_{\sigma_n}),$$

where $E_{\sigma} = \{(x, y) \mid (x, y) \in E \text{ and the label of } (x, y) \text{ is } \sigma\}$ for $\sigma \in \Sigma$. We say that the graph Γ is *automatic* if the structure $(V, E_{\sigma_1}, \dots, E_{\sigma_n})$ is automatic.

Here are examples of automatic graphs.

Example 6.2. Let T be a Turing machine. The configuration space of T is the graph $(\text{Conf}(T), E_T)$, where:

- (1) The set $\text{Conf}(T)$ is the set of all configurations of T , and
- (2) The set E_T of edges consists of all pairs (c_1, c_2) of configurations such that T has an instruction that transforms c_1 to c_2 .

¹In fact, one can effectively build the graph $\Gamma(G, X)$ without referencing Harrington and Khisamiev's theorems. The reader can construct $\Gamma(G, X)$ as an exercise.

The structure $(\text{Conf}(T), E_T)$ is clearly an automatic directed graph since the transitions $(c_1, c_2) \in E_T$ can be detected by finite automata.

The next example shows that the n -dimensional grid is also an automatic graph.

Example 6.3. Consider \mathbb{Z}^n as a labelled graph, where the labels are e_1, \dots, e_n . Identify each e_i with the vector $(0, \dots, 0, 1, 0, \dots, 0)$, whose all components are 0 except at position i . For any two vectors v and w in \mathbb{Z}^n , put an edge from v to w and label it with e_i if $v + e_i = w$. We represent each vector $v \in \mathbb{Z}^n$ as an n -tuple (x_1, \dots, x_n) of integers each written in a binary (or decimal) notation. Under this coding, the edge relation

$$E_i = \{(v, w) \mid v + e_i = w\}$$

is FA recognizable. Hence, the labelled graph \mathbb{Z}^n is automatic.

The next is a central definition of this paper that introduces automaticity for finitely generated groups.

Definition 6.4. Let G be a group generated by a finite set X of generators. We say that G is *Cayley graph automatic* if the graph $\Gamma(G, X)$ is an automatic graph.

We often refer to Cayley graph automatic groups as either *graph automatic groups* or *CGA groups*. The regular language L that encodes elements of G is termed the set of *normal forms* of G (relative to the given automatic presentation of G).

Here are several examples of CGA groups.

Example 6.5. Consider a finitely generated abelian group G . The group G can be written as $\mathbb{Z}^n \oplus A$, where A is a finite abelian group and $n \in \mathbb{N}$. The group G is generated by A and the vectors e_1, \dots, e_n in \mathbb{Z}^n . Using Example 6.3 and the fact that A is finite, it is easy to show that the group G is graph automatic.

Example 6.6. The Heisenberg group $\mathcal{H}_3(\mathbb{Z})$ is the subgroup of $\text{SL}(3, \mathbb{Z})$ consisting of all upper-triangular matrices:

$$X = \begin{pmatrix} 1 & a & b \\ 0 & 1 & c \\ 0 & 0 & 1 \end{pmatrix}.$$

The group has three generators:

$$A = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad B = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad C = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}.$$

We can represent the matrix X as the convoluted word $\otimes(a, b, c)$, where a, b and c are written in binary. The multiplication of X by each of these generators can easily be recognized by finite automaton. Indeed, the three automata that recognize the multiplication by A, B , and C accept all the strings of the form $\otimes(\otimes(a, b, c), \otimes(1 + a, b, c))$, $\otimes(\otimes(a, b, c), \otimes(a, 1 + b, c))$, and $\otimes(\otimes(a, b, c), \otimes(a, a + b, 1 + c))$, respectively. Thus, $\mathcal{H}_3(\mathbb{Z})$ is a graph automatic group.

Example 6.7. The example above can clearly be generalized to Heisenberg groups $\mathcal{H}_n(\mathbb{Z})$ consisting of all $n \times n$ matrices over \mathbb{Z} which have entries 1 at the diagonal and whose all other entries apart from first row or the last column are equal to 0.

The example below shows that the multiplication by a fixed matrix with integer entries is an FA recognizable event.

Example 6.8. Let A be any $n \times n$ matrix with integer coefficients. The matrix A naturally acts on the additive group \mathbb{Z}^n . This action is FA recognizable. This follows from the fact that both the multiplication of integers by a fixed integer and the addition on integers are FA recognizable.

Now we mention some properties of graph automatic groups that follow directly from the definitions. We start with the following easy lemma.

Lemma 6.9. *Let G be a graph automatic group over a generating set X . Then for a given word $y \in (X \cup X^{-1})^*$ there exists a finite automaton \mathcal{M}_y which accepts all the pairs $u, v \in \Gamma(G, X)$ with $v = uy$ and nothing else.*

Proof. Since $\Gamma(G, X)$ is automatic, for every $x \in X$ there exists an automaton \mathcal{M}_x such that for all $u, v \in \Gamma(G, X)$, the automaton \mathcal{M}_x detects if $v = u \cdot x$. Now one can use the automata $\mathcal{M}_x, x \in X$, to build a finite automaton \mathcal{M}_y that recognizes all $u, v \in \Gamma(G, X)$ such that $v = uy$. This can be done through Theorem 3.7. Indeed, there exists a formula $\phi(w, v)$ in the language of the Cayley graph $\Gamma(G, X)$ with free variables u, v such that $\phi(u, v)$ holds in $\Gamma(G, X)$ if and only if $v = uy$ in G . So the binary predicate defined by $\phi(u, v)$ is FA recognizable in $\Gamma(G, X)$. Hence we can build the desired automaton \mathcal{M}_y . \square

The theorem below shows that the definition of graph automaticity is independent on the generator sets. The proof is much simpler than the proof of the similar results for standard automatic groups [16].

Theorem 6.10. *If G is a graph automatic group with respect to a generating set X then G is Cayley graph automatic with respect to all finite generating sets Y of G .*

Proof. Consider a graph automatic graph $\Gamma(G, X)$. Let Y be any finite generating set for G . Each $y \in Y$ can be written as a product $x_1^{k_1} \dots x_n^{k_n}$ of elements of X . We

write this product as $w(y)$. By Lemma 6.9 the binary relation $\{(u, v) \in \Gamma(G, X)^2 \mid v = uy\}$ is FA recognizable in $\Gamma(G, X)$. This proves that $\Gamma(G, Y)$ is an automatic graph. Note that we did not need to change the automatic representation of the vertex set of the graph $\Gamma(G, X)$ in our proof. \square

7. Automatic versus graph automatic

In this section we recall the standard definition of automaticity introduced in [16], and compare it with our definition of graph automaticity.

Definition 7.1 ([16]). A group G with a finite generating set X is *automatic* if

- (1) There exists a regular subset $L \subseteq X^*$ such that the natural mapping $v \rightarrow \bar{v}$, $v \in L$, from L into G is onto.
- (2) The set $W_G = \{(u, v) \mid u, v \in L \text{ \& } \bar{u} = \bar{v} \text{ in } G\}$ is regular.
- (3) For each $x \in X$, there exists an automaton M_x that recognizes the relation:

$$\{(u, v) \mid u, v \in L \text{ and } \bar{u} = \overline{vx} \text{ in } G\}.$$

The automaton M for L , and automata M_x are called an *automatic structure* for the group G .

As mentioned in the introduction, automaticity of a group does not depend on a generating set, automatic groups have decidable word problem (in quadratic time), and they are finitely presented. The class of automatic groups is closed under finite free products, finite direct products, and finite extensions. Examples of automatic groups include free abelian groups \mathbb{Z}^n , hyperbolic groups, e.g. free groups, braid groups, and fundamental groups of many natural manifolds. Examples of non-automatic groups are $SL_n(\mathbb{Z})$, $UT_n(\mathbb{Z})$, $H_n(\mathbb{Z})$, $n \geq 3$, the wreath product $\mathbb{Z}_2 \text{ wr } \mathbb{Z}$, non-finitely presented groups, and Baumslag–Solitar groups.

There is a nice geometric property of automatic groups known as the *fellow traveller property* [16]. To explain, let G be an automatic group. Then in the notation of Definition 7.1 the following holds. There exists a constant K such that for any two words $w_1, w_2 \in L$ for which the corresponding paths end up in the Cayley graph of G at most distance 1 apart if two travelers start at the vertex 1 and travel at the same speed along the paths w_1 and w_2 , then at any given time t during the travel, the distance between $w_1(t)$ and $w_2(t)$ is uniformly bounded by K . In fact, the converse is also true, so a group G is automatic if and only if there exists a regular subset $L \subseteq X^*$ which maps onto G and satisfies the K -fellow traveller property for some constant K .

We now recast the definition of graph automaticity through the following lemma whose proof immediately follows from the definitions:

Lemma 7.2. *Let $\Gamma(G, X)$ be the Cayley graph of a group G generated by a finite set X viewed as the structure above:*

$$(V, E_{x_1}, \dots, E_{x_n}).$$

Then G is graph automatic if and only if the following conditions hold for some finite alphabet Σ :

- *There is an FA recognizable language $R \subseteq \Sigma^*$ and an onto mapping $v: R \rightarrow V$ for which the binary predicate $E(x, y) \subseteq R^2$ defined by $E(u, v) \leftrightarrow v(u) = v(v)$ is FA recognizable,*
- *All the predicates E_{x_1}, \dots, E_{x_n} are FA recognizable with respect to the mapping v , that is, for each $x \in X$ the set*

$$v^{-1}(E_x) = \{(u, v) \mid u, v \in R \text{ and } v(u)x = v(v)\}$$

is FA recognizable.

Thus, the definition of graph automaticity differs from the standard definition of automaticity in only one respect. Namely, it does not require that $X = \Sigma$. This immediately implies the following simple result showing that all automatic groups are graph automatic.

Proposition 7.3. *Every automatic group is graph automatic.*

However, the converse is not true. For instance, the Heisenberg group $\mathcal{H}_3(\mathbb{Z})$ is graph automatic (Example 6.6), but not automatic (see [16]). Later we will give more examples of such groups.

8. The word and conjugacy problems

Recall that the time complexity of the word problem in an automatic group is bounded by a quadratic polynomial. The theorem below shows that graph automatic groups enjoy the same property. They behave just like automatic groups in terms of complexity of the word problem.

We prove first the following result which is interesting in its own right.

Lemma 8.1. *Let Σ be a finite alphabet, $n \in \mathbb{N}$, $D \subseteq (\Sigma^*)^n$, and $f: D \rightarrow \Sigma^*$ be a function whose graph is FA recognizable. Then there exists a linear time algorithm that given $d \in D$ computes the value $f(d)$. Furthermore, there is a constant K such that $|f(d)| \leq |d| + K$ for any $d \in D$, where $|d| = \max\{|d_i| \mid i = 1, \dots, n\}$ for $d = (d_1, \dots, d_n) \in (\Sigma^*)^n$.*

Proof. To prove the lemma, let us denote by \mathcal{M} a finite automaton recognizing the graph of f . Consider the set P of all paths (runs) in \mathcal{M} labelled by words of the form (d_1, \dots, d_n, y) , where $d = (d_1, \dots, d_n) \in D$ and $|y| \leq \max |d|$, starting from the initial state q_0 . Let S be the set of all end states of paths from P . The membership problem to the set S can be decided in time $C \cdot \max\{|d_1|, \dots, |d_n|, |y|\}$, where C is a constant. There are two cases for S :

Case 1: Suppose that the set S contains an accepting state s . Hence there exists a path from the initial state to s such that the label of the path is of the form (d_1, \dots, d_n, y') with $|y'| \leq |d|$. One can find such a path in linear time in the size of the input d . Indeed, one can ignore the labels of the last coordinate of (d_1, \dots, d_n, y) , in which case the automaton \mathcal{M} becomes non-deterministic, and find a path in S with a label (d_1, \dots, d_n, \cdot) that ends up in an accepting state. Notice that it takes time at most $O(|d|)$, since we are looking for a shortest path (hence no self-intersections) with this property in a fixed non-deterministic automaton. Reading off the last coordinate along this path one can find y' such that (d_1, \dots, d_n, y') is accepted by \mathcal{M} . Notice that $f(d_1, \dots, d_n) = y'$ and $|y'| \leq |d|$.

Case 2. Suppose that the set S does not contain an accepting state. Then there exists a state $s \in S$ and a shortest path from s to an accepting state s' such that it is labelled by $(\diamond, \dots, \diamond, y'')$ with $|y''| \leq C'$, where C' is the number of states in M . Let y' be a string of length $|d| = |\otimes (d_1, \dots, d_n)|$ such that there is a path from q_0 to s labelled by (d_1, \dots, d_n, y') . Then $f(d) = y'y''$. Note that it takes linear time in the size of the input d to find s' and y' , and $|y'y''| \leq |d| + C'$. This proves the lemma. \square

Theorem 8.2. *The word problem in graph automatic groups is decidable in quadratic time.*

Proof. Let G be a group for which the Cayley graph $\Gamma(G, X)$ is automatic. We may assume that the regular subset $R \subseteq \Sigma^*$ (from the definition of graph automaticity given in Lemma 7.2) represents elements of G bijectively, so the function $v: R \rightarrow G$ is a bijection. Let $w = y_1 \dots y_n$ be a word in generators $y_i \in X \cup X^{-1}$. By \bar{w} we denote the element defined by w in G . We would like to find a representative $v^{-1}(\bar{w})$ of \bar{w} in L . To this end, denote by u_i the unique representative in L of the element $\bar{w}_i, i = 0, \dots, n$ (where $\bar{w}_0 = 1$), in particular, $u_n = v^{-1}(\bar{w})$. Notice, that for every $y \in X \cup X^{-1}$ the function $f_y: L \rightarrow L$ defined by $f_y(u) = v^{-1}(v(u)y)$ is FA recognizable. Hence by Lemma 8.1 for a given $u \in R$ one can compute $v = f_y(u)$ in time $C_y|u|$ for some constant C_y . Moreover, there is a constant K such that $|f_y(u)| \leq |u| + K$ for every $u \in L$. Since the set X is finite we may choose constants $C = \max\{C_y \mid y \in X \cup X^{-1}\}$ and $K = \max\{K_y \mid y \in X \cup X^{-1}\}$. The estimates above show that

$$|u_{i+1}| \leq |u_i| + K \leq \dots \leq |u_0| + (i + 1)K.$$

Hence one can compute u_n , by consecutively computing u_1, u_2, u_3, \dots , in time

$$\sum_{i=1}^n C(|u_0| + iK) = O(n^2),$$

as claimed. This proves the theorem. □

Below we introduce a notion of a *Cayley graph biautomatic group*. Let G be a group generated by a finite set X . Let $\Gamma(G, X)$ be the Cayley graph of G relative to X . Consider the *left Cayley graph* $\Gamma^l(G, X)$. It is a labelled directed graph with the vertex set G such that there is a directed edge (g, h) from g to h labelled by x if and only if $xg = h$. The graph $\Gamma^l(G, X)$ can be viewed as an algebraic structure $\Gamma^l(G, X) = (G; E_{x_1}^l, \dots, E_{x_n}^l)$, where a binary predicate $E_{x_i}^l$ defines the edges with the label x_i in $\Gamma^l(G, X)$.

Definition 8.3. A group G generated by a finite set X is *Cayley graph biautomatic* if the graphs $\Gamma(G, X)$ and $\Gamma^l(G, X)$ are automatic relative to one and the same regular set representing G . Equivalently, G is Cayley graph biautomatic if and only if the structure $(G; E_{\sigma_1}, \dots, E_{\sigma_n}, E_{\sigma_1}^l, \dots, E_{\sigma_n}^l)$ is automatic. Similar as above we often refer to these groups as *graph biautomatic groups*.

Recall that the standard biautomatic groups (in the sense of [16]) are defined in the following way. Let G be automatic group with respect to X . Let $L \subseteq X^*$ be a part of automatic structure for G . We say that G is biautomatic if L^{-1} is a part of automatic structure for G .

Proposition 8.4. *Every standard biautomatic group is Cayley graph biautomatic.*

Proof. Let G be a standard biautomatic group with a finite generating set X . Suppose $R \subseteq X^*$ is a regular set such that G is automatic relative to R and R^{-1} . It follows that the Cayley graph $\Gamma(G, X)$ is automatic, so all the binary relations E_{x_i} are FA recognizable. We need to show that the relations $E_{\sigma_i}^l$ are also FA recognizable. Since G is biautomatic the set of pairs $(u, v) \in R^2$ such that $u^{-1}x^{-1} = v^{-1}$ for a given $x \in X$ is FA recognizable, say by an automaton $M_{x^{-1}}$. Observe that $u^{-1}x^{-1} = v^{-1}$ if and only if $xu = v$. Rebuild the automaton $M_{x^{-1}}$ into an automaton M_x^l by interchanging the sets of initial and final states in M_x , then reversing each edge in M_x and changing each label x into x^{-1} . Clearly, $M_{x^{-1}}$ accepts a path with label $(u^{-1}, v^{-1}) \in R^2$ if and only if M_x^l accepts a path labelled (u, v) (in which case $v = xu$). Hence M_x^l recognizes E_x^l . This proves the proposition. □

Theorem 8.5. *The Conjugacy Problem in every graph biautomatic group G is decidable.*

Proof. Let G be a graph biautomatic group generated by a finite set X . Let $\Gamma(G, X)$ be the Cayley graph of G relative to X , with the regular set $R \subseteq \Sigma^*$ representing the domain. Cayley graphs $\Gamma(G, X)$ and $\Gamma^l(G, X)$ are automatic. Fix two words p and q in X^* . By Lemma 6.9, applied to the automatic graphs $\Gamma(G, X)$ and $\Gamma^l(G, X)$, one has that the sets of pairs $\{(u, up) \mid u \in R\}$ and $\{(u, qu) \mid u \in R\}$ are FA recognizable. Hence, the set

$$S_{p,q} = \{u \in R \mid up = qu \in G\}$$

is FA recognizable. Indeed, the formula

$$\Phi(u) = \exists z((up = z) \wedge (qu = z))$$

defines the set $S_{p,q}$ in the automatic structure $(G; E_{\sigma_1}, \dots, E_{\sigma_n}, E_{\sigma_1}^l, \dots, E_{\sigma_n}^l)$. It follows, that p and q are conjugate in G if and only if $S_{p,q} \neq \emptyset$, which is decidable. \square

Just as for automatic groups we do not, however, know if the Conjugacy Problem for graph automatic groups is decidable.

9. Universal Cayley graphs

In this section we prove that the Cayley graph of a free group with two natural extra predicates is universal. Recall that an automatic structure \mathcal{A} is *universal* if every other automatic structure \mathcal{B} can be interpreted in \mathcal{A} as defined in Definition 3.6.

Let F be a free group with basis $A = \{a_1, \dots, a_n\}$. We represent F by the set $F(A)$ of all reduced words in $A \cup A^{-1}$. Recall that a word is reduced if it contains no subwords of the form aa^{-1} , $a^{-1}a$ where $a \in A$. On the set $F(A)$ define the following two predicates \preceq and el :

$$x \preceq y \leftrightarrow x \text{ is a prefix of } y,$$

and

$$el(x, y) \leftrightarrow |x| = |y|.$$

Denote by $\Gamma_{\text{free}}(A)$ the Cayley graph $\Gamma(F, A)$ with two extra predicates \preceq and el , i.e.,

$$\Gamma_{\text{free}}(A) = (F(A); E_{a_1}, \dots, E_{a_n}, \preceq, el).$$

Now we prove the following theorem:

Theorem 9.1. *The automatic structure $\Gamma_{\text{free}}(A)$ is universal.*

Proof. It is easy to see that Γ_{free} is an automatic structure. Indeed, the set $F(A)$ and all the predicates in $\Gamma_{\text{free}}(A)$ are clearly FA recognizable.

Consider the structure

$$\mathcal{M} = (A^*; R_a(x, y), x \preceq y, el(x, y))_{a \in \Sigma},$$

defined in Example 3.9. By Theorem 3.10 \mathcal{M} is a universal automatic structure. Since interpretability is a transitive relation, it suffices to interpret the structure \mathcal{M} in $\Gamma_{\text{free}}(A)$ by first-order formulas. Notice, that the set A^* is a subset of $F(A)$, consisting of all words without “negative” letters a^{-1} when $a \in A$. Furthermore, all the predicates in \mathcal{M} are restrictions of the corresponding predicates from $\Gamma_{\text{free}}(A)$ onto A^* . Hence, it suffices to show that the subset A^* is definable in $\Gamma_{\text{free}}(A)$. Observe, first, that the formula

$$\Phi_{<}(u, v) = \exists z(z \preceq v \wedge z \neq v \wedge |z| = |u|)$$

defines the binary relation $|u| < |v|$ in $\Gamma_{\text{free}}(A)$. Now it is easy to see that the formula

$$\Phi(w) = \forall u \forall v (u \preceq w \wedge (\bigvee_{a \in A} E_a(u, v)) \rightarrow |u| < |v|)$$

defines A^* in $\Gamma_{\text{free}}(A)$. This proves the theorem. \square

10. Cayley graph automatic groups: constructions

Our goal is to show that graph automaticity is preserved under several natural group-theoretic constructions.

10.1. Finite extensions. Let G be a group and H be a normal subgroup of G . We say that G is a *finite extension of H* if the quotient group G/H is finite. It turns out that graph automaticity preserves finite extensions:

Theorem 10.1. *Finite extensions of Cayley graph automatic groups are again Cayley graph automatic.*

Proof. Let G be a finite extension of a graph automatic group H . Denote by

$$G/H = \{Hk_0, \dots, Hk_{r-1}\}$$

the set of all right cosets of G with respect to H . There exists a finite function g such that for all $0 \leq i, s \leq r-1$, we have an equality:

$$Hk_i \cdot Hk_s = Hk_{g(i,s)}. \quad (1)$$

Suppose H is generated by a finite set h_0, \dots, h_{n-1} , where $h_0 = 1$. Equality (1) implies that there are sequences $g_1(i, s), \dots, g_x(i, s)$ and $0 \leq u_1(i, s), \dots, u_x(i, s) \leq n - 1$ of integers such that

$$k_i k_s = h_{u_1(i,s)}^{g_1(i,s)} \dots h_{u_x(i,s)}^{g_x(i,s)} k_{g(i,s)}.$$

Similarly, there are sequences $f_1(i, j), \dots, f_m(i, j)$ and $0 \leq v_1(i, j), \dots, v_m(i, j) \leq n - 1$ of integers such that for $0 \leq i \leq r - 1$ and $0 \leq j \leq n - 1$ the following equalities hold:

$$k_i h_j = h_{v_1(i,j)}^{f_1(i,j)} \dots h_{v_m(i,j)}^{f_m(i,j)} k_i.$$

This implies that for all $s, i \leq r - 1, j \leq n - 1$, and $h \in H$ one has

$$\begin{aligned} h k_i h_j k_s &= h h_{v_1(i,j)}^{f_1(i,j)} \dots h_{v_m(i,j)}^{f_m(i,j)} k_i k_s \\ &= h h_{v_1(i,j)}^{f_1(i,j)} \dots h_{v_m(i,j)}^{f_m(i,j)} h_{u_1(i,s)}^{g_1(i,s)} \dots h_{u_x(i,s)}^{g_x(i,s)} k_{g(i,s)}. \end{aligned} \quad (2)$$

For $h \in H$ denote by \bar{h} the unique word form a regular set $R \subseteq \Sigma^*$ representing h under a fixed graph automatic presentation of H . Every element of G can be written uniquely as a product $h k_i$ for some $h \in H$. We encode the elements $h k_i$ of G as words $\bar{h} k_i$. Here we assume that the alphabet Σ does not contain symbols k_0, \dots, k_{r-1} . The set of such words is clearly FA recognizable. Notice that the elements k_s, h_j altogether generate G . The equalities (2) tell us that there are finite automata $M_{j,s}$ that for every h_j, k_s accept precisely all pairs of words of the form $(\bar{h} k_i, w)$, where $w = h k_i h_j k_s$ in G . Note that to build the automata $M_{j,s}$ one needs to use: the original automata that represent the group H , the sequences $g_1(i, s), \dots, g_x(i, s)$ and $u_1(i, s), \dots, u_x(i, s)$, the sequences $f_1(i, j), \dots, f_m(i, j)$ and $v_1(i, j), \dots, v_m(i, j)$, the function g , and the automata representing the multiplication by elements $h_v^{f(i,j)}$ and $h_u^{g(i,j)}$ in the group H . This shows that the group G is graph automatic. The theorem is proved. \square

A simple corollary of the proof is the following:

Corollary 10.2. *Finite extensions of graph biautomatic groups are again graph biautomatic.*

10.2. Semidirect products. Let A and B be finitely generated groups and $\tau : B \rightarrow \text{Aut}(A)$ a homomorphism. As usual, the *semidirect product* of A and B relative to τ , denoted $A \rtimes_\tau B$, is a group G generated by A and B such that: A is normal in G , every element $g \in G$ is uniquely presented as a product $g = ba$, where $a \in A, b \in B$, and multiplication is given by $(ba)(b_1 a_1) = b b_1 a^{b_1} a_1$, where $a^{b_1} = \tau(b_1)(a)$. We identify A and B with their images in G under monomorphisms: $a \rightarrow 1_B a$ and $b \rightarrow b 1_A$.

Recall that an automorphism $\alpha \in \text{Aut}(A)$ is automatic if its graph is an FA recognizable language.

Theorem 10.3. *Let A and B be graph automatic groups with finite sets of generators X and Y , and $\tau: B \rightarrow \text{Aut}(A)$ a homomorphism. Assume that the automorphism $\tau(y)$ is automatic for every $y \in Y$. Then the semidirect product $G = A \rtimes_{\tau} B$ is graph automatic.*

Proof. Let R and S be regular sets that give graph automatic representations of A and B . Since every element of G has a unique decomposition as ba , where $b \in B, a \in A$, this gives a bijective encoding of elements of G as words sr , where $s \in S, r \in R$. Observe, that G is generated by the set $X \cup Y$. For any $x \in X$ the relation $(sr)x = s_1r_1$ is obviously FA recognizable. Similarly, for each $y \in Y$, we have $(sr)y = syr^y$ and this relation is also FA recognizable, since the graph $\{(r, r^y) \mid r \in R\}$ is FA recognizable (because $\tau(y)$ is automatic). This proves the theorem. \square

The following result is an immediate corollary of the theorem.

Corollary 10.4. *The direct product of two graph automatic groups is graph automatic.* \square

Consider the group $G = (\mathbb{Z} \times \mathbb{Z}) \rtimes_A \mathbb{Z}$, where $A \in \text{SL}(2, \mathbb{Z})$. Here we mean that the action of a generator, say t , of \mathbb{Z} on $\mathbb{Z} \times \mathbb{Z}$ is given by the matrix A . Such groups play an important part as lattices in the Lie group $\text{Sol} = (\mathbb{R} \times \mathbb{R}) \rtimes \mathbb{R}$, where t acts on $\mathbb{R} \times \mathbb{R}$ by a diagonal matrix $\text{diag}(e^t, e^{-t})$. These groups are also interesting in our context because of the following observation. If A is conjugate in $\text{GL}(2, \mathbb{R})$ to a matrix $\text{diag}(\lambda, \lambda^{-1})$ for some $\lambda > 1$, then G has exponential Dehn function, hence G is not an automatic group [16], but it is graph automatic (see Proposition 10.5 below).

Proposition 10.5. *The group $G = (\mathbb{Z} \times \mathbb{Z}) \rtimes_A \mathbb{Z}$ is graph automatic for every $A \in \text{SL}(2, \mathbb{Z})$.*

Proof. We first note that every matrix $A \in \text{SL}(2, \mathbb{Z})$ gives rise to an FA recognizable automorphism of $\mathbb{Z} \times \mathbb{Z}$. See Example 6.8. Since the underlying groups $\mathbb{Z}, \mathbb{Z} \times \mathbb{Z}$ are graph automatic, by the theorem above the group G is graph automatic.

Alternatively, graph automaticity of G can also be shown via Theorem 3.7. Indeed, the Cayley graph Γ of G is first-order interpretable in $(\mathbb{Z}; +)$, which is automatic. To see this, represent elements of G as triples $(x, y, t) \in \mathbb{Z}^3$. This set is FA recognizable. Now observe that multiplication in G is given by

$$(x_1, y_1, t_1)(x_2, y_2, t_2) = ((x_1, y_1) + A(x_2, y_2))^{\top}, t_1 + t_2).$$

Therefore multiplication of (x_1, y_1, t_1) by a fixed generator of G is definable in $(\mathbb{Z}; +)$ as claimed. \square

10.3. Wreath products. For the next theorem we define the *restricted wreath product* of a group A by a group B . Let A_b be an isomorphic copy of A for each $b \in B$. Consider the direct sum of groups A_b denoted by K . Thus,

$$K = \bigoplus_{b \in B} A_b,$$

where elements of K are viewed as functions $f : B \rightarrow A$ such that $f(b) = 1_A$ for almost all $b \in B$. We write elements of K as (a_b) . Each element $c \in B$ induces an automorphism α_c of K as follows:

$$\alpha_c(a_b) = (a_{bc}).$$

The wreath product of A by B consists of all pairs of the form (b, k) , where $b \in B$ and $k \in K$, with multiplication defined by

$$(b, k) \cdot (b_1, k_1) = (bb_1, \alpha_{b_1}(k)k_1).$$

Thus, the wreath product of A by B is simply the semidirect product of $K = \bigoplus_{b \in B} A_b$ and B relative to the homomorphism $B \rightarrow \text{Aut}(K)$ given by $c \rightarrow \alpha_c$.

Theorem 10.6. *For every finite group G the wreath product of G by \mathbb{Z} is graph automatic.*

Proof. We give an explicit automatic presentation of the wreath product. The elements of the wreath product are of the form

$$(i, (\dots, g_{-n}, g_{-n+1}, \dots, g_{-1}, g_0, g_1, \dots, g_{m-1}, g_m, \dots)),$$

where $g_j \in G$ and $i \in \mathbb{Z}$. We refer to g_0 as the element of G at position 0. We can assume that g_k is the identity 1_G of the group G for all $k < -n$ or $k > m$, and $g_{-n} \neq 1_G$ and $g_m \neq 1_G$. We can represent the element above as the following string

$$\otimes(i, g_{-n} \dots g_{-1}(g_0, \star)g_1 \dots g_m),$$

where i is written in binary. The alphabet of these strings is clearly finite since G is a finite group. The symbol \star in this string represents elements of G at position 0. The generators of the wreath product are elements $(0, g)$ and $(1, g)$ represented by the strings $\oplus(0, (g, \star))$ and $\oplus(1, (1_G, \star))$, where $g \in G$. Multiplication by these generators works as follows:

$$\otimes(i, g_{-n} \dots (g_0, \star) \dots g_m) \cdot \oplus(0, (g, \star)) = \otimes(i, g_{-n} \dots (g_0 \cdot g, \star) \dots g_m)$$

and

$$\otimes(i, g_{-n} \dots (g_0, \star) \dots g_m) \cdot \oplus(1, (1_G, \star)) = \otimes(i + 1, g'_{-n+1} \dots (g'_0, \star) \dots g'_{m+1}),$$

where $g'_{j+1} = g_j$ for $j \in \{-n, \dots, m\}$. These operations can clearly be performed by finite automata. The theorem is proved. \square

The theorem above can be applied to construct many examples of graph automatic groups that are not finitely presented. Hence, these give us another class of graph automatic but not automatic groups.

Corollary 10.7. *There exist graph automatic not finitely presented (and hence not automatic) groups.*

Proof. The restricted wreath product of a non-trivial finite group G by \mathbb{Z} , by Theorem 10.6, is graph automatic. Now we use the following theorem by Baumslag [6]. For finitely presented groups A and B , the restricted wreath product of A by B is finitely presented if and only if either A is trivial or B is finite. Hence, for nontrivial finite group G , the restricted wreath product of G by \mathbb{Z} is not finitely presented but graph automatic. \square

10.4. Free products. In this section we prove that graph automaticity is preserved with respect to free products. The result follows from representation of elements of the free product by their normal forms.

Theorem 10.8. *If A and B are graph automatic groups then their free product $A \star B$ is again graph automatic.*

Proof. Since A and B are graph automatic we can assume that the elements of A and B are strings over disjoint alphabets Σ_1 and Σ_2 . Therefore, $A \cap B = \{\lambda\}$. A *normal form* is a sequence of the type

$$g = g_1 \square g_2 \square \dots \square g_n,$$

where $g_i \in A \cup B$, $g_i \neq \lambda$, $\square \notin \Sigma_1 \cup \Sigma_2$, and the adjacent elements g_i and g_{i+1} are not from the same group A or B , where $i \geq 0$. The set N of all normal forms is FA recognizable. Every element in $A * B$ is uniquely represented by some normal form g in N and every normal form $g \in N$ gives rise to a unique element in $A * B$. If a_1, \dots, a_n generate A and b_1, \dots, b_m generate B then these elements together generate the whole group $A \star B$. The multiplication by each of these generators can be performed by finite automata using the automata given for the underlying groups A and B . For instance, the automaton M that multiplies $g \in N$ by a generator $a \in A$ can be described as follows. Given $g, g' \in N$ the automaton M reads $\otimes(g, g')$. The aim is to detect if $ga = g'$ in G . Assume that

$$g' = g'_1 \square g'_2 \square \dots \square g'_n.$$

Notice that if $g_n \in B$ then g' must be of the form

$$g' = g_1 \square g_2 \square \dots \square g_n \square a.$$

And if $g_n \in A$ then g'_n must be of the form

$$g' = g_1 \square g_2 \square \dots \square g_{n-1} \square g'_n,$$

where $g_n a = g'_n$ in the group A . The last equality can be detected by a finite automaton using the automaton that recognizes the multiplication by a in the group A . \square

10.5. Amalgamated products. Let A and B be groups. Let ϕ be an isomorphism from a subgroup H_A of A into the subgroup H_B of B . By H we denote the isomorphism type of the group H_A . The *amalgamated product* of A and B by H , denoted by $A \star_H B$, is the factor group of $A \star B$ by the normal closure of the set $\{\phi(h)h^{-1} \mid h \in H_A\}$. The amalgamated product $A \star_H B$ is viewed as the result of identifying H_A and H_B in the free product $A \star B$. Below we show simple conditions guaranteeing graph automaticity of amalgamated products.

Theorem 10.9. *Let A, B be graph automatic groups and H a subgroup of A and B . Suppose that one of the following conditions holds:*

- (1) *The group H is graph biautomatic and A and B are finite extensions of H , or*
- (2) *H is a finite subgroup of both A and B .*

Then the amalgamated product $A \star_H B$ is graph automatic.

Proof. Assume that H is a graph biautomatic group, and A and B are finite extensions of H . By Corollary 10.2, both A and B can be assumed to be graph biautomatic groups. Moreover, we can assume that the set of elements of the subgroup H is a regular language.

As in the proof of Theorem 10.8, we consider normal forms. Note that elements of A and B are strings (under given automatic presentations) since A and B are graph automatic. We also assume that the alphabets Σ_1 and Σ_2 of graph automatic representations of A and B are disjoint.

We choose the set of representatives R_A and R_B of the cosets of H in A and in B , respectively. These two sets are finite by the assumption. Let a_1, \dots, a_m and b_1, \dots, b_s be the strings from R_A and R_B , respectively. Note that for every $g \in \{a_1, \dots, a_m, b_1, \dots, b_m\}$, by Lemma 6.9 and graph biautomaticity of A and B , the sets

$$\{(u, v) \mid u, v \in A \ \& \ ug = v\} \quad \text{and} \quad \{(u, v) \mid u, v \in A \ \& \ gu = v\}$$

are FA recognizable sets. Let \square be a symbol not in $\Sigma_1 \cup \Sigma_2$. Define an *A-normal form* as a sequence

$$g_1 \square g_2 \square \dots \square g_n \square z,$$

such that

- (1) each g_i belongs to either R_A or R_B , $g_i \neq \lambda$,
- (2) two consecutive g_i and g_{i+1} belong to distinct set of representatives, and
- (3) $z \in H_A$.

It is not hard to see that the set of all A -normal forms is a regular language that we denote by N . This set determines elements of the amalgamated group $A \star_H B$. Moreover, each element in the amalgamated group has a unique representation in A -normal form [9].

Let X and Y be finite set of generators for A and B , respectively. Our goal is now to show that the multiplication of elements in N by each of the generator elements from $X \cup Y$ is FA recognizable. We first consider the case when the generator is in X . Take $v \in N$ of the form $g_1 \square g_2 \square \dots \square g_n \square z$, and a generator $x \in X$.

Assume that $g_n \in R_B$. Using the fact that A is automatic, we can compute (the string representing) the element $z \cdot x$. Let w be the element $z \cdot x$. We now find an element $a \in R_A$ such that $w \in aH$. Hence, we can write the element w as $a \cdot h$ for some $h \in H$. Namely, $w = a(a^{-1}w)$. Thus, we have the equality

$$vx = g_1 \square g_2 \square \dots \square g_n \square a \square a^{-1}w.$$

From the above, since the underlying groups are biautomatic, we see that this is an FA recognizable event, that is, the set

$$\{(v, v') \mid v \text{ is of the form } g_1 \square g_2 \square \dots \square g_n \square z, \\ v' \in N, g_n \in B, x \in X, v' = vx\}$$

is FA recognizable.

Assume now that $g_n \in R_A$. Since A is biautomatic the set

$$\{(g_n, w) \mid w = g_n \cdot z \cdot x, g_n \in R_A, z \in H, x \in X\}$$

is FA recognizable. Now given $w = g_n \cdot z \cdot x$, we can represent it as the product ah for some $a \in R_A$ and $h \in H$. This is again an FA recognizable event. We conclude that the set multiplication by $x \in X$ of elements in N can be recognized by finite automata. The case when we multiply elements of N by the generators $y \in Y$ is treated similarly.

Now we prove the second part of the theorem. Since H is finite the set of all left co-sets with respect to H in both A and B is uniformly FA recognizable. In other words, the sets

$$\{(a_1, a_2) \mid a_1^{-1}a_2 \in H, a_1, a_2 \in A\} \quad \text{and} \quad \{(b_1, b_2) \mid b_1^{-1}b_2 \in H, b_1, b_2 \in B\}$$

are FA recognizable languages. Therefore we can select regular sets R_A and R_B of left-cost representatives of A and B , respectively. As above, one considers the set N of normal forms. For each $z \in H$ and $g \in X \cup Y$, for each of the groups A and B there exists a finite automaton that recognizes the language $\{(u, v) \mid uzg = v\}$. Therefore, since H is finite, we have that for each $g \in X \cup Y$ the set

$$\{(v, w) \mid v, w \in N, vg = w\}$$

is FA recognizable. This proves the second part of the theorem. □

As an application we give the following result.

Corollary 10.10. *The groups $SL_2(\mathbb{Z})$ and $GL_2(\mathbb{Z})$ are graph automatic.*

Proof. The group $SL_2(\mathbb{Z})$ is isomorphic to $\mathbb{Z}_4 \star_{\mathbb{Z}_2} \mathbb{Z}_6$. Similarly, the group $GL_2(\mathbb{Z})$ is isomorphic to $D_4 \star_{D_2} D_6$, where D_n is a dihedral group (see for instance [9]). By the theorem above the groups $SL_2(\mathbb{Z})$ and $GL_2(\mathbb{Z})$ both are graph automatic. \square

Of course, the groups $SL_2(\mathbb{Z})$ and $GL_2(\mathbb{Z})$ are already known to be automatic (see [16]).

11. Subgroups

In this section we describe a simple technique that is analogous, in some respect, to the technique of quasi-convex subgroups in standard automatic groups. Let G be a graph automatic group. Assume that R is a regular language representing the group G via a bijection $\nu: R \rightarrow G$. A finitely generated subgroup $H \leq G$ is called *regular* if the pre-image ν^{-1} is a regular subset of R . A similar definition describe regular subgroups of graph automatic monoids.

Proposition 11.1. *Let G be a graph automatic group or monoid and H a regular finitely generated subgroup of G . Then H is graph automatic.*

Proof. Suppose that R is a regular language representing the group G via a bijection $\nu: R \rightarrow G$. Let X_H be a finite generating set of H . Let X_G be a finite generating set of G containing X_H . Since G is graph automatic it is automatic relative to X_H . Hence the binary predicates E_x are FA presentable in R . Their restrictions to $\nu^{-1}(H)$ are FA presentable as well. Thus, H is graph automatic. \square

The following result, implied by the proposition above, turns out to be useful in applications.

Corollary 11.2. *Let $M_n(\mathbb{Z})$ be the multiplicative monoid of all $n \times n$ integer matrices. If H is a regular finitely generated subgroup of $M_n(\mathbb{Z})$ then H is graph automatic.*

Proof. It suffices to note that multiplication operation of matrices in $M_n(\mathbb{Z})$ by a fixed $n \times n$ -matrix is an automatic operation. \square

12. Nilpotent Cayley graph automatic groups

In this section we show that there are many interesting finitely generated nilpotent groups which are graph automatic. For this we need to introduce a particular technique

of polycyclic presentations that initially comes from Malcev’s work on nilpotent groups [37]. For the detailed exposition see the books [26], [25], [5].

Let G be a group, $a = (a_1, \dots, a_n)$ an n -tuple of elements in G , and $\alpha = (\alpha_1, \dots, \alpha_n)$ an n -tuple of integers. By a^α we denote the following product

$$a^\alpha = a_1^{\alpha_1} a_2^{\alpha_2} \dots a_n^{\alpha_n}.$$

Concatenation of two tuples a and b is denoted ab and a 1-tuple (x) is usually denoted by x .

Recall that every finitely generated abelian group A is a direct sum of cyclic groups:

$$A = \langle a_1 \rangle \times \dots \times \langle a_s \rangle \times \langle b_1 \rangle \times \dots \times \langle b_t \rangle$$

where $\langle a_i \rangle$ is infinite cyclic, and $\langle b_i \rangle$ is a finite cyclic group of order $\omega(b_i)$. Every element $g \in A$ can uniquely be represented in the form

$$g = a_1^{\alpha_1} \dots a_s^{\alpha_s} b_1^{\beta_1} \dots b_t^{\beta_t} \tag{3}$$

where $\alpha_i \in \mathbb{Z}$ and $\beta_j \in \{0, 1, \dots, \omega(b_j) - 1\}$. We call the tuple

$$a = (a_1, \dots, a_s, b_1, \dots, b_t)$$

a base of A and the tuple $\sigma(g) = (\alpha_1, \dots, \alpha_s, \beta_1, \dots, \beta_t)$ the coordinate of g in the base \bar{a} . In this notation we write the equality (3) as follows $g = a^{\sigma(g)}$.

One can generalize the notion of base to polycyclic groups. Recall that a group G is *polycyclic* if there is a sequence of elements $a_1, \dots, a_n \in G$ that generates G such that if G_i denotes the subgroup $\langle a_i, \dots, a_n \rangle$ then G_{i+1} is normal in G_i for every i . In this case

$$G = G_1 \geq G_2 \geq \dots \geq G_n \geq G_{n+1} = 1 \tag{4}$$

is termed a *polycyclic series* of G . The sequence $a = \langle a_1, \dots, a_n \rangle$ is called a base of G .

Let $a = (a_1, \dots, a_n)$ be a base of a polycyclic group G . Then the quotient G_i/G_{i+1} is a cyclic group generated by the coset $a_i G_{i+1}$. Denote by ω_i the order of the group G_i/G_{i+1} which is the order of the element $a_i G_{i+1}$ in G_i/G_{i+1} . Here $\omega_i = \infty$ if the order is infinite. We refer to the tuple $\omega(a) = (\omega_1, \dots, \omega_n)$ as the *order* of a . Now set $Z_{\omega_i} = \mathbb{Z}$ if $\omega_i = \infty$ and $Z_{\omega_i} = \{0, 1, 2, \dots, \omega_i - 1\}$ otherwise.

Lemma 12.1. *Let $a = (a_1, \dots, a_n)$ be a base of a polycyclic group G of order $\omega(a) = (\omega_1, \dots, \omega_n)$. Then for every $g \in G$ there is a unique decomposition of the following form:*

$$g = a_1^{\alpha_1} a_2^{\alpha_2} \dots a_n^{\alpha_n}, \quad \alpha_i \in Z_{\omega_i}. \tag{5}$$

Proof. Let $g \in G$. Since the quotient group G_1/G_2 is cyclic generated by $a_1 G_2$ one has $gG_2 = a_1^{\alpha_1} G_2$ for some unique $\alpha_1 \in Z_{\omega_1}$. The element $g' = a_1^{-\alpha_1} g$ belongs to

G_2 . Notice that (a_2, \dots, a_n) is a base of G_2 . Hence by induction on the length of the base there is a unique decomposition of g' of the type

$$g' = a_2^{\alpha_2} \dots a_n^{\alpha_n}, \quad \alpha_i \in Z_{\omega_i}.$$

Now $g = a_1^{\alpha_1} g'$ and the result follows. □

In the notation above for an element $g \in G$ the tuple $\sigma(g) = (\alpha_1, \dots, \alpha_n)$ from (5) is called the tuple of *coordinates* of g in the base a . Sometimes we write the equality (5) as $g = a^{\sigma(g)}$.

Finitely generated nilpotent groups are polycyclic so they have finite bases as above. Moreover, it is easy to see that an arbitrary finitely generated group G is nilpotent if and only if it has a finite base (a_1, \dots, a_n) such that the series (4) is central, i.e., $[G_i, G] \leq G_{i+1}$ for every $i = 1, \dots, n$ (here $G_{n+1} = 1$).

Now suppose G is an arbitrary finitely generated nilpotent group of nilpotency class m . The *lower central series* of G is defined inductively by

$$G_1 = G, \quad G_2 = [G_1, G], \quad \dots, \quad G_{i+1} = [G_i, G], \quad \dots$$

By assumption, we have $G_m \neq 1$ and $G_{m+1} = 1$. It follows that all the quotients G_i/G_{i+1} are finitely generated abelian groups. Let d_i be a tuple of elements from G_i such that its image in G_i/G_{i+1} under the standard epimorphism is a base of the abelian group G_i/G_{i+1} . Then the tuple $a = d_1 d_2 \dots d_m$ obtained by concatenation from the tuples d_1, \dots, d_m , is a base of G . We refer to a as a *lower central series* base of G .

Similarly, the *upper central series* of the group G is the sequence:

$$1 = Z_0(G) \leq Z_1(G) \leq Z_2(G) \leq \dots \leq Z_{i+1}(G) \leq \dots,$$

where $Z_{i+1}(G)$ is defined inductively as the set

$$Z_{i+1}(G) = \{x \in G \mid [x, y] \in Z_i(G) \text{ for all } y \in G\}.$$

In particular $Z_1(G)$ is the center of G . Thus, the group $Z_{i+1}(G)$ is the full preimage of the center of the group $G/Z_i(G)$ under the canonical epimorphism $G \rightarrow G/Z_i(G)$. If G is torsion-free then the quotients $Z_{i+1}(G)/Z_i(G)$ are free abelian groups of finite rank. In particular, one can choose a tuple d_i of elements from $Z_{i+1}(G)$ which form a standard basis of the free abelian group $Z_{i+1}(G)/Z_i(G)$, where $i = 1, \dots, m$. The tuple $a = d_m d_{m-1} \dots d_1$ obtained as concatenation of d_m, \dots, d_1 is called an *upper central base* of G . Notice that in this case $\omega_i = \infty$ for each $i = 1, \dots, n$. Such bases are called *Malcev's bases* of G .

Now we give the following important definition that singles out special type of polynomials needed to perform the multiplication operation in finitely generated nilpotent groups of nilpotency class 2.

Definition 12.2. We say that $p(x_1, \dots, x_n, y_1, \dots, y_n)$ is a *special quadratic polynomial* in variables x_1, \dots, x_n and y_1, \dots, y_m if

$$p(x_1, \dots, x_n, y_1, \dots, y_n) = \sum_{i,j} \alpha_{ij} x_i y_j + \sum_i \beta_i x_i + \sum_j \gamma_j y_j,$$

where $\alpha_{i,j}, \beta_i, \gamma_j$ are constants from \mathbb{Z} .

In tuple notation we write $p(x, y)$, where $x = (x_1, \dots, x_n)$ and $y = (y_1, \dots, y_n)$. If α and β are tuples of integers, then $f(\alpha, \beta)$ denotes the value of f obtained by substituting $x \rightarrow \alpha, y \rightarrow \beta$. Similarly, for a tuple of polynomials $f(x, y) = (f_1(x, y), \dots, f_k(x, y))$, we write $f(\alpha, \beta)$ to denote $(f_1(\alpha, \beta), \dots, f_k(\alpha, \beta))$. The following lemma indicates the use of special quadratic polynomials in calculating the group operation in a finitely generated group of nilpotency class at most 2.

Lemma 12.3. *Let G be a finitely generated 2-nilpotent group with a lower series base $a = (a_1, \dots, a_n)$. There exist a tuple of special polynomials $f(x, y) = (f_1(x, y), \dots, f_n(x, y))$ with $x = (x_1, \dots, x_n)$ and $y = (y_1, \dots, y_n)$ such that for any tuples of integers $\alpha, \beta \in \mathbb{Z}^n$ one has*

$$a^\alpha \cdot a^\beta = a^{f(\alpha, \beta)}.$$

Proof. Since G is a 2-nilpotent group then $G > [G, G] > 1$ is the lower central series of G . In particular, $[G, G]$ is a subgroup of the center $Z(G)$ of G . Let $\omega(a)$ be the order of the lower series base $a = (a_1, \dots, a_n)$. By definition of the base a , we have that a is a concatenation of two tuples $d_1 = (a_1, \dots, a_s)$ and $d_2 = (a_{s+1}, \dots, a_m)$ such that d_2 is a base of the abelian group $[G, G]$. For tuples $\alpha = (\alpha_1, \dots, \alpha_n), \beta = (\beta_1, \dots, \beta_n) \in \mathbb{Z}^n$ consider the following product

$$a_1^{\alpha_1} a_2^{\alpha_2} \dots a_n^{\alpha_n} \cdot a_1^{\beta_1} a_2^{\beta_2} \dots a_n^{\beta_n}. \tag{6}$$

Since G is 2-nilpotent all the commutators $[a_i^{\alpha_i}, a_1^{\beta_1}]$ are in the center of G , so using equalities $a_i^{\alpha_i} a_1^{\beta_1} = a_1^{\beta_1} a_i^{\alpha_i} [a_i^{\alpha_i}, a_1^{\beta_1}]$ one can rewrite the product (6) in the following form:

$$a_1^{\alpha_1 + \beta_1} a_2^{\alpha_2} \dots a_n^{\alpha_n} a_2^{\beta_2} \dots a_n^{\beta_n} \prod_{i=2}^n [a_i^{\alpha_i}, a_1^{\beta_1}]. \tag{7}$$

By induction on the length of the base, we can assume that there are special quadratic polynomials, say $g_2(\bar{x}, \bar{y}), \dots, g_n(\bar{x}, \bar{y})$, where $\bar{x} = (x_2, \dots, x_n), \bar{y} = (y_2, \dots, y_n)$, such that

$$a_2^{\alpha_2} \dots a_n^{\alpha_n} a_2^{\beta_2} \dots a_n^{\beta_n} = a_2^{g_2(\alpha, \beta)} \dots a_n^{g_n(\alpha, \beta)}.$$

Notice that for 2-nilpotent groups the equalities

$$[a_i^{\alpha_i}, a_1^{\beta_1}] = [a_i, a_1]^{\alpha_i \beta_1}$$

hold for every α_i, β_1 . So

$$\prod_{i=2}^n [a_i^{\alpha_i}, a_1^{\beta_1}] = \prod_{i=2}^n [a_i, a_1]^{\alpha_i \beta_1}.$$

Since $[a_i, a_1] \in [G, G]$ one has $[a_i, a_1] = a_{s+1}^{\delta_{s+1,i}} \dots a_n^{\delta_{ni}}$ for some $\delta_{ji} \in \mathbb{Z}$. Therefore

$$\begin{aligned} \prod_{i=2}^n [a_i^{\alpha_i}, a_1^{\beta_1}] &= \prod_{i=2}^n (a_{s+1}^{\delta_{s+1,i}} \dots a_n^{\delta_{ni}})^{\alpha_i \beta_1} \\ &= \prod_{i=2}^n a_{s+1}^{\delta_{s+1,i} \alpha_i \beta_1} \dots a_n^{\delta_{ni} \alpha_i \beta_1} \\ &= a_{s+1}^{\sum_{i=2}^n \delta_{s+1,i} \alpha_i \beta_1} \dots a_n^{\sum_{i=2}^n \delta_{ni} \alpha_i \beta_1}. \end{aligned}$$

Observe that $h_j(x_1, \dots, x_n, y_1) = \sum_{i=2}^n \delta_{ji} x_i y_1$ are special quadratic polynomials in x and y , so

$$\prod_{i=2}^n [a_i^{\alpha_i}, a_1^{\beta_1}] = a_{s+1}^{h_{s+1}(\alpha, \beta)} \dots a_n^{h_n(\alpha, \beta)}$$

Combining the latter one with the equality (7) one gets that the initial product (6) is equal to

$$a_1^{\alpha_1 + \beta_1} a_2^{g_2(\alpha, \beta)} \dots a_s^{g_s(\alpha, \beta)} a_{s+1}^{g_{s+1}(\alpha, \beta) + h_{s+1}(\alpha, \beta)} a_n^{g_n(\alpha, \beta) + h_n(\alpha, \beta)},$$

which proves the lemma. □

Theorem 12.4. *Every finitely generated group G of nilpotency class at most two is graph automatic.*

Proof. We prove the theorem by cases.

Case 1: If G is abelian then our Example 6.5 shows that G is graph automatic.

Case 2: Assume that G be a finitely generated torsion free 2-nilpotent group. Fix an arbitrary upper central Malcev's base a of G . We use notation from Lemma 12.3 throughout the proof. Every element $g \in G$ can be uniquely represented by its tuple of coordinates $\sigma(g)$ relative to the base a . The set of coordinates of elements of G

$$\sigma(G) = \{\sigma(g) \mid g \in G\} = Z_{\omega_1} \times \dots \times Z_{\omega_n} = \mathbb{Z}^n$$

is in bijective correspondence with G . This set is clearly definable by first-order formulas in the Presburger arithmetic $\mathcal{P} = \langle \mathbb{N}, + \rangle$. By Theorem 3.7, the set is FA recognizable.

Now we prove that the Cayley graph Γ of G relative to the generating set $\{a_1, \dots, a_n\}$ is interpretable in \mathcal{P} . It suffices to show that for a given generator a_i the set of pairs

$$\{(\sigma(g), \sigma(ga_i)) \mid g \in G\}$$

is first order interpretable in \mathcal{P} for each a_i . To this end let $g \in G$ and $\sigma(g) = (\alpha_1, \dots, \alpha_n)$. By Lemma 12.3 there exist a tuple of special polynomials $f(x, y)$ such that

$$ga_i = a^{f(\sigma(g), \varepsilon_i)}, \quad (8)$$

where $\varepsilon_i = (0, \dots, 0, 1, 0, \dots, 0)$ (all components are equal to 0, except for the i s, which is equal to 1). Therefore,

$$\{(\sigma(g), \sigma(ga_i)) \mid g \in G\} = \{(\sigma(g), f(\sigma(g), \varepsilon_i)) \mid g \in G\}.$$

Notice, that $f(\sigma(g), \varepsilon_i) = (f_1(\sigma(g), \varepsilon_i), \dots, f_n(\sigma(g), \varepsilon_i))$ and every $f_j(\sigma(g), \varepsilon_i)$ is a fixed linear function in $\sigma(g)$ since f , by Lemma 12.3, is a special polynomial. Each linear polynomial is first order definable in \mathcal{P} . Therefore the set

$$(\alpha, f_j(\alpha, \varepsilon_i)) \mid \alpha \in \mathbb{Z}^n\}$$

is first order definable in \mathcal{P} for every $j = 1, \dots, n$. Hence the set

$$\{(\alpha, f(\alpha, \varepsilon_i)) \mid \alpha \in \mathbb{Z}^n\}$$

is also first order definable in \mathcal{P} for every $i = 1, \dots, n$. All these are FA recognizable by Theorem 3.7. Thus G is graph automatic.

Case 3. Let G be an arbitrary finitely generated 2-nilpotent group. Then the set of all torsion elements in G forms a finite subgroup $\text{Tor}(G)$ of G . If k is the order of $\text{Tor}(G)$ then the subgroup G^k generated by $\{g^k \mid g \in G\}$ is a finitely generated torsion-free 2-nilpotent subgroup of G such that quotient G/G^k is finite. So G is a finite extension of G^k . By Case 2 the group G^k is graph automatic. By Theorem 10.1, the original group G is also graph automatic. This proves the theorem. \square

There are finitely generated nilpotent graph automatic groups which are not 2-nilpotent. For instance, the group $\mathcal{H}_n(\mathbb{Z})$, where $n > 3$, as proved in Example 6.7 are graph automatic. The following provide other examples of graph automatic nilpotent groups of class > 2 .

Example 12.5. The following groups are graph automatic:

- Let $\text{UT}(n, \mathbb{Z})$ be the group of upper triangular matrices over \mathbb{Z} (with 1 at the diagonal).
- The group $\text{UT}^m(n, \mathbb{Z})$ that consists of all matrices from $\text{UT}(n, \mathbb{Z})$ such that the first $m - 1$ diagonals above the main one have all entries equal to 0.

Proof. The proof follows from Proposition 11.2. \square

13. Solvable graph automatic groups

13.1. Baumslag–Solitar groups. The Baumslag–Solitar groups $BS(m, n)$, given by presentations

$$BS(m, n) = \langle a, b \mid a^{-1}b^m a = b^n \rangle,$$

play an important role in combinatorial and geometric group theory. It is well known that the groups $BS(m, n)$, for $m \neq n$, are not automatic (if $n = m$ they are automatic) [16]. It is also known that $BS(m, n)$ are all asynchronously automatic [16]. In this section we prove that the Baumslag–Solitar groups $BS(1, n)$ are graph automatic groups for all $n \in \mathbb{N}$.

Theorem 13.1. *The Baumslag–Solitar groups $BS(1, n)$ are graph automatic for $n \in \mathbb{N}$.*

Proof. To simplify our exposition we consider only the group $B(1, 2)$. We prove this theorem through the action of this group on the real line \mathbb{R} . We represent the elements a and b of the group $BS(1, 2)$ as linear functions $g_a: \mathbb{R} \rightarrow \mathbb{R}$ and $g_b: \mathbb{R} \rightarrow \mathbb{R}$ given by $g_a(x) = 2x$ and $g_b(x) = x + 1$. Let G be the group generated by g_a and g_b . The group multiplication on G is composition of functions. We show first that the group G is isomorphic to $BS(1, 2)$ via the isomorphism induced by the mapping $a \rightarrow g_a$ and $b \rightarrow g_b$.

Claim 13.2. *The elements g_a and g_b satisfy the identity $g_a^{-1}g_b g_a = g_b^2$.*

Indeed, given a real number $x \in \mathbb{R}$, we have the following equalities:

$$g_a^{-1}g_b g_a(x) = g_b g_a\left(\frac{1}{2}x\right) = g_a\left(\frac{1}{2}x + 1\right) = x + 2 = g_b^2(x).$$

For the next claim recall that $\mathbb{Z}[1/2]$ is the set of all dyadic numbers, that is, numbers of the form $i/2^j$, where $i, j \in \mathbb{Z}$.

Claim 13.3. *Each $g \in G$ is a linear function of the form $ax + b$, where $a = 2^n$ and $b \in \mathbb{Z}[1/2]$.*

The proof of the claim is by induction on the length of words over a, b representing elements of G . For g_a and g_b the claim is obvious. Suppose $g \in G$ is in the form $2^n x + m/2^k$. We need to show that the functions $g g_a$, $g g_a^{-1}$, $g g_b$, and $g g_b^{-1}$ are also of the same form. This can be shown through easy calculations. For instance,

$$g g_a(x) = g_a(g(x)) = g_a(2^n x + m/2^k) = 2^{n+1} x + m/2^{k-1},$$

and

$$g g_b(x) = g_b(g(x)) = g_b(2^n x + m/2^k) = 2^{n+1} x + (m + 2^k)/2^k.$$

The next claim shows that every function of the form $2^n x + m/2^k$ can be generated by the functions g_a and g_b . This reverses the claim above.

Claim 13.4. Assume that g is a function of the form $2^n x + m/2^k$. Then $g = g_a^n (g_a^k g_b^m g_a^{-k})$.

Indeed, first note the following equality:

$$g_a^k g_b^m g_a^{-k}(x) = g_b^m g_a^{-k}(2^k x) = g_a^{-k}(2^k x + m) = x + m/2^k.$$

Now it is easy to see that

$$g^n g_a^k g_b^m g_a^{-k}(x) = g_a^k g_b^m g_a^{-k}(2^n x) = 2^n x + m/2^k = g.$$

These claims show that the groups $\text{BS}(1, 2)$ and G are isomorphic. The isomorphism is induced by the mapping $a \rightarrow g_a$ and $b \rightarrow g_b$. So, we identify these two groups.

Now we give a representation of the Cayley graph for $\text{BS}(1, 2)$ with the generators a and b . Consider a function $g \in G$ of the form $2^n x + m/2^k$, where $k \geq 0$. We can always assume that m is odd if $k > 0$. Thus, we can represent the element g as the (convoluted) string $\otimes(n, m, k)$. We put the following conditions on these strings:

- (1) n and m are integers written in binary.
- (2) The integer k is written in unary.
- (3) If k is the empty string (thus k represents 0), then $m \in \mathbb{Z}$. Otherwise, m is odd.

We denote this set of strings by D . It is clear that D is finite automata recognizable set. It is also clear that the mapping $D \rightarrow G$ given by $(n, m, k) \rightarrow 2^n x + m/2^k$ is a bijection.

The multiplication by generators g_a and g_b of elements $g = 2^n x + m/2^k$ of G is now represented on D as follows:

$$(n, m, k) \rightarrow_a (n + 1, m, k - 1) \quad \text{and} \quad (n, m, k) \rightarrow_b (n, m + 2^k, k).$$

It is clear that the multiplication by a is recognized by finite automata. The multiplication by b is also finite automata recognizable because k is represented in unary. \square

13.2. Other metabelian groups. We have shown in Section 10.2 that the groups $G = (\mathbb{Z} \times \mathbb{Z}) \rtimes_A \mathbb{Z}$ are graph automatic, where $A \in \text{SL}(2, \mathbb{Z})$. This can clearly be generalized to higher dimensions:

Proposition 13.5. A group $G = \mathbb{Z}^n \rtimes_A \mathbb{Z}$, where $A \in \text{SL}(n, \mathbb{Z})$ is graph automatic.

Proof. The argument from Proposition 10.5 can be applied here as well. \square

13.3. Non-metabelian solvable groups. Let $T(n, \mathbb{Z})$ be the group of triangular matrices of size $n \times n$ over the integers \mathbb{Z} . So, matrices in $T(n, \mathbb{Z})$ have all zeros below the main diagonal.

Proposition 13.6. *The group $T(n, \mathbb{Z})$ is graph automatic.*

Proof. This follows from Proposition 11.2. □

Another interesting example comes from Theorem 10.6.

Example 13.7. Let K be a solvable finite group. Then by Theorem 10.6 the wreath product K by the group \mathbb{Z} is Cayley graph automatic. It is clear that G is solvable of the solvability class at least the class of K .

14. Finitely generated FA presentable groups

The Definition 3.1 suggests that automaticity into groups can also be introduced by requiring that the group operation is FA recognizable. In this section we do exactly this by considering groups (G, \cdot) in which the group operation \cdot is automatic. We recast the definition:

Definition 14.1. We say that a group G is *FA presentable* if the following conditions are satisfied:

- The domain of G is FA recognizable set.
- The graph of the group operation, that is, the set $\{(u, v, w) \mid u \cdot v = w\}$ is FA recognizable.

Note that the definition does not require that G is finitely generated. Examples of FA presentable groups are the following:

- The additive group of p -adic rational numbers: $\mathbb{Z}[1/p]$.
- Finitely generated abelian groups.
- The infinite direct sum $\bigoplus G$ of a finite group G .

For abelian (not necessarily finitely generated) FA-presentable groups see [39] [40] [48]. We also mention a recent result of Tsankov that the additive group of rational numbers is not FA presentable. The proof uses advanced techniques of additive combinatorics [48].

Our goal is to give a full characterization of finitely generated FA-presentable groups. Our proof follows Thomas and Oliver [42]. We start with the following definition.

Definition 14.2. A group is *virtually abelian* if it has a normal abelian subgroup of finite index.

An example of virtually abelian group is D_ω , the infinite dihedral group. One can view this group as the automorphism group of the graph that looks like a bi-infinite chain.

Lemma 14.3. *Finitely generated virtually abelian groups all are FA presentable.*

Proof. Let G be a finitely generated virtually abelian group. It is easy to see that there exists a finitely generated abelian torsion free normal subgroup A of G which has a finite index, say n , in G . We can assume that A is isomorphic to \mathbb{Z}^k . Let x_1, \dots, x_k be the generators of \mathbb{Z}^k . For simplicity of exposition we assume that $k = 2$ (a similar argument works in the general case).

Let t_1, \dots, t_n be all representatives of the quotient group G/\mathbb{Z}^k . Every element $g \in G$ can be written as $t_i x_1^{a_1} x_2^{a_2}$ where $a_1, a_2 \in \mathbb{Z}$ and $i \in \{1, \dots, n\}$. Since \mathbb{Z}^k is normal, we also have the following list of equalities for some fixed integers $c_{1,1,j}, c_{1,2,j}, c_{2,1,j}$ and $c_{2,2,j}$:

$$x_1 t_j = t_j x_1^{c_{1,1,j}} x_2^{c_{1,2,j}} \quad \text{and} \quad x_2 t_j = t_j x_2^{c_{2,1,j}} x_1^{c_{2,2,j}} \quad \text{where } j = 1, \dots, n.$$

In addition, there are integer constants c_i and c_j such that $t_i t_j = t_k x_1^{c_i} x_2^{c_j}$ for all $i, j = 1, \dots, n$. Taking all these into account we can now perform the group operation on G as follows:

$$\begin{aligned} t_i x_1^{a_1} x_2^{a_2} \cdot t_j x_1^{b_1} x_2^{b_2} &= t_i t_j x_1^{a_1 c_{1,1,j} + a_2 c_{2,1,j} + b_1} x_2^{a_1 c_{1,2,j} + a_2 c_{2,2,j} + b_2} \\ &= t_k x_1^{c_i + c_j} x_2^{c_i} x_1^{a_1 c_{1,1,j} + a_2 c_{2,1,j} + b_1} x_2^{a_1 c_{1,2,j} + a_2 c_{2,2,j} + b_2}. \end{aligned}$$

All these operations can now be performed by finite automata. This proves the lemma. □

The next lemma again suits a more general case of monoids. A *monoid* is a structure $(M; \cdot)$, where \cdot is an associative binary operation on M .

Lemma 14.4. *If $(M; \cdot)$ is an automatic monoid then for all $m_1, \dots, m_n \in M$ the following inequality holds true:*

$$|m_1 \dots m_n| \leq \max\{|m_i| \mid i = 1, \dots, n\} + C \cdot \log(n),$$

where C is a constant.

Proof. Let C be the number required by the Constant Growth Lemma (see Lemma 8.1). By the lemma we have

$$|m_1 \cdot m_2| \leq \max\{|m_1|, |m_2|\} + C \tag{\#}$$

for all $m_1, m_2 \in M$. So, for $n = 1, 2$ the lemma is obvious. For $n > 2$ we write $n = n_1 + n_2$ with $n_1 = n/2$. Consider the elements

$$x = m_1 \dots m_{n_1} \quad \text{and} \quad y = m_{n_1+1} \dots m_n.$$

From the induction assumption we have the following inequalities:

$$|x| \leq \max_{1 \leq i \leq n_1} |m_i| + C \cdot \log(n_1) \quad \text{and} \quad |y| \leq \max_{n_1 < i \leq n} |m_i| + C \cdot \log(n_1).$$

Therefore from the inductive assumption and (#) we have

$$|x \cdot y| \leq \max\{|x|, |y|\} + C \leq \max\{|m_i| \mid i = 1, \dots, n\} + C \cdot \log(n).$$

Thus, we have the desired inequality. □

Let $X = \{g_1, \dots, g_k\}$ be the set of generators of the group G . For each element $g \in G$, let $\delta(g)$ be the minimum n such that $g = a_1 \dots a_n$ in the group G , where each $a_i \in X$. Now we define the following:

$$G_n = \{g \in G \mid \delta(g) \leq n\} \quad \text{and} \quad \text{gr } \mathcal{G}(n) = |G_n|.$$

The function $\text{gr } G$ is called the *growth function* of the group G .

Lemma 14.5. *If G is FA-presentable then its growth function is bounded by a polynomial.*

Proof. By Lemma 14.4, for each $g \in G_n$ we have $\delta(g) \leq C \log(n)$, where C is a constant. Therefore there is a constant C_1 such that

$$\text{gr } G(n) \leq |\Sigma|^{C \log(n)} \leq 2^{C_1 \log(n)} \leq n^{C_1}.$$

This proves the lemma. □

Now we need two deep results from group theory. The first is the theorem of Gromov stating that finitely generated groups with polynomial growth are all virtually nilpotent [21]. The second comes from the theorems of Romanovski [44] and Noskov[41], stating that a virtually solvable group has a decidable first order theory if and only if it is virtually abelian. Virtually nilpotent groups are virtually solvable. Thus, we have proved the following:

Theorem 14.6. *A finitely generated group is FA-presentable if and only if it is virtually abelian.*

Since all virtually abelian groups are graph automatic we have the following result:

Corollary 14.7. *Every finitely generated FA-presentable group is graph automatic.* □

References

- [1] A. I. Abakumov, E. A. Palyutin, M. A. Taitslin, and Yu. E. Shishmarev, Categorical quasivarieties. *Algebra and Logika* **11** (1972), 3–38; English transl. *Algebra and Logic* **11** (1972), 1–20. [Zbl 0249.02024](#) [MR 0360407](#)
- [2] L. Bartholdi, R. I. Grigorchuk, and Z. Šunić, Branch groups. In *Handbook of algebra*, Vol. 3, North-Holland, Amsterdam 2003, 989–1112. [Zbl 1140.20306](#) [MR 2035113](#)
- [3] L. Bartholdi, A. G. Henriques, and V. V. Nekrashevych, Automata, groups, limit spaces, and tilings. *J. Algebra* **305** (2006), 629–663. [Zbl 1166.20303](#) [MR 2266846](#)
- [4] A. Blumensath, Automatic structures. Diploma thesis, RWTH Aachen, Aachen 1999. <http://www.mathematik.tu-darmstadt.de/blumensath/Publications/AutStr.pdf>
- [5] G. Baumslag, Lecture notes on nilpotent groups. CBMS Regional Conf. Ser. in Math. 2, Amer. Math. Soc., Providence, R.I., 1969. [Zbl 0241.20001](#) [MR 0283082](#)
- [6] G. Baumslag, Wreath products and finitely presented groups. *Math. Z.* **75** (1961), 22–28. [Zbl 0090.24402](#) [MR 0120269](#)
- [7] G. Baumslag and D. Solitar, Some two-generator one-relator non-Hopfian groups. *Bull. Amer. Math. Soc.* **68** (1962), 199–201. [Zbl 0108.02702](#) [MR 0142635](#)
- [8] A. Blumensath and E. Grädel, Automatic structures. In *15th Annual IEEE Symposium on Logic in Computer Science* (Santa Barbara, CA, 2000), IEEE Comput. Soc. Press, Los Alamitos, CA, 2000, 51–62. [MR 1946085](#)
- [9] O. Bogopolski, *Introduction to group theory*. EMS Textbk. Math., European Mathematical Society, Zürich 2008. [Zbl 1215.20001](#) [MR 2396717](#)
- [10] M. R. Bridson, Combinations of groups and the grammar of reparameterization. *Comment. Math. Helv.* **78** (2003), 752–771. [Zbl 1044.20018](#) [MR 2016694](#)
- [11] M. R. Bridson and R. H. Gilman, Formal language theory and the geometry of 3-manifolds. *Comment. Math. Helv.* **71** (1996), 525–555. [Zbl 0873.20026](#) [MR 1420509](#)
- [12] J. R. Büchi and L. H. Landweber, Definability in the monadic second-order theory of successor. *J. Symbolic Logic* **34** (1969), 166–170. [Zbl 0209.02203](#) [MR 0269492](#)
- [13] C. Choffrut, A short introduction to automatic group theory. In *Semigroups, algorithms, automata and languages* (Coimbra, 2001), World Sci. Publ., Singapore 2002, 133–154. [Zbl 1039.20013](#) [MR 2023786](#)
- [14] M. Dehn, Über unendliche diskontinuierliche Gruppen. *Math. Ann.* **71** (1911), 116–144. [JFM 42.0508.03](#)
- [15] M. Dehn, Transformation der Kurven auf zweiseitigen Flächen. *Math. Ann.* **72** (1912), 413–421. [JFM 43.0571.03](#)
- [16] D. B. A. Epstein, J. W. Cannon, D. F. Holt, S. V. F. Levy, M. S. Paterson, and W. P. Thurston, *Word processing in groups*. Jones and Bartlett Publishers, Boston 1992. [Zbl 0764.20017](#) [MR 1161694](#)
- [17] B. Farb, Automatic groups: a guided tour. *Enseign. Math.* (2) **38** (1992), 291–313. [Zbl 0811.20038](#) [MR 1189009](#)
- [18] S. M. Gersten, Introduction to hyperbolic and automatic groups. In *Summer School in Group Theory in Banff, 1996*, CRM Proc. Lecture Notes 17, Amer. Math. Soc., Providence, RI, 1999, 45–70. [Zbl 0926.20028](#) [MR 1653684](#)

- [19] R. H. Gilman, Formal languages and infinite groups. In *Geometric and computational perspectives on infinite groups* (Minneapolis, MN and New Brunswick, NJ, 1994), DIMACS Ser. Discrete Math. Theoret. Comput. Sci. 25, Amer. Math. Soc., Providence, RI, 1996, 27–51. [Zbl 0851.20030](#) [MR 1364178](#)
- [20] O. Goodman and M. Shapiro, On a generalization of Dehn’s algorithm. *Internat. J. Algebra Comput.* **18** (2008), 1137–1177. [Zbl 1227.20033](#) [MR 2468741](#)
- [21] M. Gromov, Groups of polynomial growth and expanding maps. *Inst. Hautes Études Sci. Publ. Math.* **53** (1981), 53–73. [Zbl 0474.20018](#) [MR 623534](#)
- [22] V. S. Harizanov, Pure computable model theory. In *Handbook of recursive mathematics*, Vol. 1, Stud. Logic Found. Math. 138, North-Holland, Amsterdam 1998, 3–114. [Zbl 0952.03037](#) [MR 1673621](#)
- [23] L. Harrington, Recursively presentable prime models. *J. Symbolic Logic* **39** (1974), 305–309. [Zbl 0332.02055](#) [MR 0351804](#)
- [24] W. Hodges, *Model theory*. Encyclopedia Math. Appl. 42, Cambridge University Press, Cambridge 1993. [Zbl 0789.03031](#) [MR 1221741](#)
- [25] D. Holt, B. Eick, and E. O’Brien, *Handbook of computational group theory*. Chapman & Hall/CRC Press, Boca Raton 2000. [Zbl 1091.20001](#) [MR 2129747](#)
- [26] M. I. Kargapolov and Yu. I. Merzlyakov, Fundamentals of the theory of groups. Graduate Texts in Math. 62, Springer-Verlag, New York 1979. [Zbl 0549.20001](#) [MR 0551207](#)
- [27] N. G. Khisamiev, Strongly constructive models of a decidable theory. On strongly constructive models of decidable theories. *Izv. Akad. Nauk Kazakh. SSR Ser. Fiz.-Mat.* **35** (1) (1974), 83–84. [MR 0354344](#)
- [28] B. Khoussainov and M. Minnes, Model-theoretic complexity of automatic structures. *Ann. Pure Appl. Logic* **161** (2009), 416–426. [Zbl 1221.03025](#) [MR 2552756](#)
- [29] B. Khoussainov and M. Minnes, Three lectures on automatic structures. In *Logic Colloquium 2007*, Lect. Notes Log. 35, Assoc. Symbol. Logic, La Jolla, CA, 2010, 132–176. [Zbl 1228.03017](#) [MR 2668230](#)
- [30] B. Khoussainov and A. Nerode, Automatic presentations of structures. In *Logic and computational complexity* (Indianapolis, IN, 1994), Lecture Notes in Comput. Sci. 960, Springer, Berlin 1995, 367–392. [MR 1449670](#)
- [31] B. Khoussainov, A. Nies, S. Rubin, and F. Stephan, Automatic structures: richness and limitations. *Log. Methods Comput. Sci.* **3** (2007), No. 2, Paper 2. <http://www.lmcs-online.org/ojs/viewarticle.php?id=210&layout=abstract&iid=2> [Zbl 1128.03028](#) [MR 2306568](#)
- [32] B. Khoussainov, S. Rubin, and F. Stephan, Automatic linear orders and trees. *ACM Trans. Comput. Log.* **6** (2005), 675–700. [MR 2176784](#)
- [33] D. Kuske, J. Liu, and M. Lohrey, The isomorphism problem on classes of automatic structures. In *25th Annual IEEE Symposium on Logic in Computer Science LICS 2010*, IEEE Computer Soc., Los Alamitos, CA, 2010, 160–169. [MR 2953905](#)
- [34] D. Kuske and M. Lohrey, Logical aspects of Cayley-graphs: the group case. *Ann. Pure Appl. Logic* **131** (2005), 263–286. [Zbl 1063.03005](#) [MR 2097229](#)
- [35] D. Kuske and M. Lohrey, First-order and counting theories of ω -automatic structures. *J. Symbolic Logic* **73** (2008), 129–150. [Zbl 1141.03015](#) [MR 2387935](#)

- [36] D. Kuske and M. Lohrey, Some natural decision problems in automatic graphs. *J. Symbolic Logic* **75** (2010), 678–710. [Zbl 1192.03010](#) [MR 2648160](#)
- [37] A. I. Mal'cev, On a class of homogeneous spaces. *Izvestiya Akad. Nauk. SSSR. Ser. Mat.* **13** (1949), 9–32; English transl. Amer. Math. Soc. Transl. 39, Amer. Math. Soc., Providence, R.I., 1951. [Zbl 0034.01701](#) [MR 0028842](#)
- [38] V. Nekrashevych, *Self-similar groups*. Math. Surveys Monogr. 117, Amer. Math. Soc., Providence, RI, 2005. [Zbl 1087.20032](#) [MR 2162164](#)
- [39] A. Nies, Describing groups. *Bull. Symbolic Logic* **13** (2007), 305–339. [Zbl 1167.20017](#) [MR 2359909](#)
- [40] A. Nies and P. Semukhin, Finite automata presentable abelian groups. *Ann. Pure Appl. Logic* **161** (2009), 458–467. [Zbl 1219.68116](#) [MR 2552759](#)
- [41] G. A. Noskov, On the elementary theory of a finitely generated almost solvable group. *Izv. Akad. Nauk SSSR Ser. Mat.* **47** (1983), 498–517; English transl. *Math. USSR-Izv.* **22** (1984), 465–482. [Zbl 0539.20018](#) [MR 703594](#)
- [42] G. P. Oliver and R. M. Thomas, Automatic presentations for finitely generated groups. In *STACS 2005*, Lecture Notes in Comput. Sci. 3404, Springer, Berlin 2005, 693–704. [Zbl 1118.20301](#) [MR 2151659](#)
- [43] M. O. Rabin, Decidability of second-order theories and automata on infinite trees. *Trans. Amer. Math. Soc.* **141** (1969), 1–35. [Zbl 0221.02031](#) [MR 0246760](#)
- [44] N. S. Romanovskii, On the elementary theory of an almost polycyclic group. *Mat. Sb. (N.S.)* **111(153)** (1980), 135–143; English transl. *Math. USSR-Sb.* 39 (1981), 125–132. [Zbl 0462.20029](#) [MR 560467](#)
- [45] S. Rubin, Automatic Structures. PhD thesis, University of Auckland, 2004.
- [46] S. Rubin, Automata presenting structures: a survey of the finite string case. *Bull. Symbolic Logic* **14** (2008), 169–209. [Zbl 1146.03028](#) [MR 2413002](#) (2009d:03094)
- [47] M. Sapir, Asymptotic invariants, complexity of groups and related problems. *Bull. Math. Sci.* **1** (2011), 277–364. [Zbl 06112155](#) [MR 2901003](#)
- [48] T. Tsankov, The additive group of the rationals does not have an automatic presentation. *J. Symbolic Logic* **76** (2011), 1341–1351. [Zbl 1247.03070](#) [MR 2895399](#)
- [49] M. Y. Vardi, Verification of concurrent programs: the automata-theoretic framework. *Ann. Pure Appl. Logic* **51** (1991), 79–98. [Zbl 0725.03013](#) [MR 1100497](#)

Received September 19, 2012; revised November 16, 2012

O. Kharlampovich, Department of Mathematics, McGill University, Montreal, Quebec, Canada

E-mail: olga@math.mcgill.ca

B. Khossainov, Computer Science Department, The University of Auckland, New Zealand

E-mail: bmk@cs.auckland.ac.nz

A. Miasnikov, Department of Mathematics, Stevens Institute of Technology, Hoboken, NJ, U.S.A.

E-mail: amiasnikov@gmail.com