CONTEXT-FREE PAIRS OF GROUPS. I - CONTEXT-FREE PAIRS AND GRAPHS

TULLIO CECCHERINI-SILBERSTEIN AND WOLFGANG WOESS

Abstract. Let G be a finitely generated group, A a finite set of generators and K a subgroup of G. We call the pair (G,K) context-free if the set of all words over A that reduce in G to an element of K is a context-free language. When K is trivial, G itself is called context-free; context-free groups have been classified more than 20 years ago in celebrated work of Muller and Schupp as the virtually free groups.

Here, we derive some basic properties of such group pairs. Context-freeness is independent of the choice of the generating set. It is preserved under finite index modifications of G and K, respectively. If G is virtually free and K is finitely generated then (G,K) is context-free. A basic tool is the following: (G,K) is context-free if and only if the Schreier graph of (G,K) with respect to A is a context-free graph.

1. Introduction

Let G be a finitely generated grou p and K a subgroup (not necessarily finitely generated). We can choose a finite set $A \subset G$ of generators such that every element of G is of the form $g = g_1 \cdots g_n$, where $n \geq 0$ and $g_1, \ldots, g_n \in A$. Thus, A generates G as a semigroup. We shall say that (G,K) is *context-free,* if – loosely spoken – the language of all words over A that represent an element of K is context-free.

The precise definition needs some preparation. Let $\mathbf{\Sigma}$ be a finite *alphabet* and $\psi : \mathbf{\Sigma} \to G$ be a (not necessarily injective) mapping such that $A = \psi(\mathbf{\Sigma})$ satisfies the above finite generation property for G. Then ψ has a unique extension, also denoted ψ, as a monoid homomorphism $\psi : \mathbf{\Sigma}^* \to G$. Recall that $\mathbf{\Sigma}^*$ consists of all *words* $w = a_1 \cdots a_n$, where $n \geq 0$ and $a_1, \ldots, a_n \in \mathbf{\Sigma}$ (repetitions allowed). The number n is the *length* $|w|$ of w. If $n = 0$ this means that $w = \epsilon$, the *empty word.* This is the neutral element of $\mathbf{\Sigma}^*$, and $\mathbf{\Sigma}^*$ is a free monoid with the binary operation of *concatenation* of words. The extension of ψ is of course given by

$$\psi(a_1 \cdots a_n) = \psi(a_1) \cdots \psi(a_n)\,,$$

where the product on the right hand side is taken in G. Given these ingredients, we shall say that $\psi : \mathbf{\Sigma} \to G$ is a *semigroup presentation* of G, referring to the fact that A generates G as a semigroup. A *language* over $\mathbf{\Sigma}$ is a non-empty subset of $\mathbf{\Sigma}^*$.

(1.1) Definition. The *word problem* of (G,K) with respect to ψ is the language

$$L(G,K,\psi) = \{w \in \mathbf{\Sigma}^* : \psi(w) \in K\}\,.$$

We say that the triple (G,K,ψ) is *context-free,* if $L(G,K,\psi)$ is a context-free language.

Date: August 12, 2009.

2000 *Mathematics Subject Classification.* 20F10, 68Q45, 05C25 .

Key words and phrases. Finitely generated pair of groups, word problem, context-free language, push-down automaton, context-free graph.

The first author was partially supported by a visiting professorship at TU Graz. The second author was partially supported by a visiting professorship at Università di Roma - La Sapienza.

A *context-free grammar* is a quadruple $\mathcal{C} = (\mathbf{V}, \mathbf{\Sigma}, \mathbf{P}, S)$, where $\mathbf{V}$ is a finite set of *variables,* disjoint from the finite alphabet $\mathbf{\Sigma}$ (the *terminal symbols*), the variable S is the *start symbol,* and $\mathbf{P} \subset \mathbf{V} \times (\mathbf{V} \cup \mathbf{\Sigma})^*$ is a finite set of *production rules.* We write $T \vdash u$ or $(T \vdash u) \in \mathbf{P}$ if $(T, u) \in \mathbf{P}$. For $v, w \in (\mathbf{V} \cup \mathbf{\Sigma})^*$, we write $v \Longrightarrow w$ if $v = v_1 T v_2$ and $w = v_1 u v_2$, where $u, v_1, v_2 \in (\mathbf{V} \cup \mathbf{\Sigma})^*$ and $T \vdash u$. This is a single derivation step, and it is called *rightmost,* if $v_2 \in \mathbf{\Sigma}^*$. A *derivation* is a sequence $v = w_0, w_1, \dots, w_k = w \in (\mathbf{V} \cup \mathbf{\Sigma})^*$ such that $w_{i-1} \Longrightarrow w_i$; we then write $v \overset{*}{\Longrightarrow} w$. A *rightmost derivation* is one where each step is rightmost. The succession of steps of any derivation $T \overset{*}{\Longrightarrow} w \in \mathbf{\Sigma}^*$ can be reordered so that it becomes a rightmost derivation. For $T \in \mathbf{V}$, we consider the language $L_T = \{w \in \mathbf{\Sigma}^* : T \overset{*}{\Longrightarrow} w\}$. The *language generated by* $\mathcal{C}$ is $L(\mathcal{C}) = L_S$.

A *context-free language* is a language generated by a context-free grammar. As a basic reference for Language and Automata Theory, we refer to the magnificent monograph of Harrison [5].

The above definition of a context-free pair, or rather triple, (G, K, ψ) makes sense when G is a finitely generated monoid and K is a sub-monoid, but here we are interested in groups. When in addition $K = \{1_G\}$, this leads to the notion of G being a *context-free group.* In two celebrated papers, Muller and Schupp [10], [11] have carried out a detailed study of context-free groups and more generally, context-free *graphs.* In particular, context-freeness of a group is independent of the particular choice of the generating set A of G. The main result of [10], in combination with a fundamental theorem of Dunwoody [4], is that a finitely generated group is context-free if and only if it is virtually free, that is, it contains a free subgroup with finite index. (In [10], it is assumed that $A = A^{-1}$ and that $\psi : \mathbf{\Sigma} \to A = \psi(\mathbf{\Sigma})$ is one-to-one, but the results carry over immediately to the more general setting where those two properties are not required.)

Previously, Anisimov [1] had shown that the groups whose word problem $L(G, \{1_G\}, \psi)$ is *regular* (see §2 for the definition) are precisely the finite groups.

The abovementioned context-free graphs are labelled, rooted graphs with finitely many isomorphism classes of *cones.* The latter are the connected components of the graph that remain after removing a ball around the root with arbitrary radius. See §4 for more precise details. As shown in [11], there is a natural correspondence between such graphs and *pushdown automata*, which are another tool for generating context-free languages; see §3.

Among subsequent work, we mention Pélecq [12] and Sénizergues [15], who studied actions on, resp. quotients of context-free graphs. Group-related examples occur also in Ceccherini-Silberstein and Woess [3].

More recently, Holt, Rees, Röver and Thomas [6] have introduced and studied *co-context-free* groups, which are such that the complement of $L(G, \{1_G\}, \psi)$ is context-free, see also Lehnert and Schweitzer [8]. This concept has an obvious extension to co-context-free pairs of groups, resp. graphs, on whose examination we do note (yet) embark.

In the present notes, we collect properties and examples of context-free pairs of groups (G, K).

- The language $L(G, K, \psi)$ is regular if and only if the index $[G : K]$ of K in G is finite (Proposition 2.5).
- The property that $L(G, K, \psi)$ is context-free does not depend on the specific choice of the semigroup presentation ψ, so that context-freeness is just a property of the pair (G, K), a consequence of Lemma 3.1.
- If (G, K) is context-free then $L(G, K, \psi)$ is a deterministic context-free language (see §3 for the definition) for any semigroup presentation $\psi : \mathbf{\Sigma} \to G$ (Corollary 4.10.a).
- If (G, H) is context-free and H is a finitely generated subgroup of G, then the pair $(H, K \cap H)$ is context-free (Lemma 3.1).
- If $[G : H] < \infty$ and K is a subgroup of H then (G, K) is context-free if and only if (H, K) is context-free (Proposition 3.3).
- If H, K are subgroups of G such that H/K is finite, then (G, H) is context-free if and only if (G, K) is context-free (Proposition 4.11).
- If (G, K) is context-free then $(G, g^{-1}Kg)$ is context-free for every $g \in G$ (Corollary 4.10.b).
- If G is virtually free and K is a finitely generated subgroup of G then (G, K) is context-free (Corollary 5.6).

Several of these properties rely on the following.

- A fully deterministic, symmetric labelled graph (see §2 for definitions) is context-free in the sense of Muller and Schupp if and only if the language of all words which are labels of a path that starts and ends at a given root vertex is context-free (Theorems 4.2 and 4.7).

The (harder) "if" part is not contained in previous work. It implies the following.

- The pair (G, K) is context-free if and only if for some ($\Longleftrightarrow$ any) symmetric semigroup presentation $\psi : \mathbf{\Sigma} \to G$, the *Schreier graph* of (G, K) with respect to ψ is a context-free graph. (See again §2 for precise definitions).

In a second paper [17], a slightly more general approach to context-freeness of graphs via cuts and tree-sets is given. It allows to show that certain structural properties ("irreducibility") are preserved under finite-index-modifications of the underlying pair of groups. This is then applied to random walks, leading in particular to results on the asymptotic behaviour of transition probabilities.

In concluding the Introduction, we remark that with the exception of some "elementary" cases, context-free pairs of groups are always pairs with more than one *end*. Ends of pairs of groups were studied, e.g., by Scott [14], Swarup [16] and Sageev [13]. In particular, the interplay between context-freeness of pairs and decomposition as amalgamated products or HNN-extensions needs still to be explored.

2. Schreier graphs, and the regular case

Let $\mathbf{\Sigma}$ be a finite alphabet. A *directed graph labelled by* $\mathbf{\Sigma}$ is a triple (X, E, ℓ), where X is the (finite or countable) set of *vertices,* $E \subset X \times \mathbf{\Sigma} \times X$ is the set of *oriented, labelled edges* and $\ell : E \ni (x, a, y) \mapsto a \in \mathbf{\Sigma}$ is the *labeling* map.

For an edge $e = (x, a, y) \in E$, its *initial vertex* is $e^- = x$ and its *terminal vertex* is $e^+ = y$, and we say that e is *outgoing* from x and *ingoing* into y. If $y = x$ then e is a *loop*, which is considered both as an outgoing and as an ingoing edge. We allow *multiple edges*, i.e., edges of the form $e_1 = (x, a_1, y)$ and $e_2 = (x, a_2, y)$ with $a_1 \neq a_2$, but here we exclude multiple edges where also the labels coincide. The graph is always assumed to be *locally finite*, that is, every vertex is an initial or terminal vertex of only finitely many edges. We also choose a fixed vertex $o \in X$, the *root* or *origin.* We shall often just speak of the graph X, keeping in mind the presence of E and ℓ.

We call X *fully labelled* if at every vertex, each $a \in \Sigma$ occurs as the label of at least one outgoing edge. We say that X is *deterministic* if at every vertex all outgoing edges have distinct labels, and *fully deterministic* if it is fully labelled and deterministic. Finally, we say that X is *symmetric* or *undirected* if there is a proper involution $a \mapsto a^{-1}$ of $\mathbf{\Sigma}$ (i.e., $(a^{-1})^{-1} = a$, excluding the possibility that $a^{-1} = a$) such that for each edge $e = (x, a, y) \in E$, also the *reversed edge* $e^{-1} = (y, a^{-1}, x)$ belongs to E.

A *path* in X is a sequence $\pi = e_1 e_2 \ldots e_n$ of edges such that $e_i^+ = e_{i+1}^-$ for $i = 1, \ldots, n-1$. The vertices $\pi^- = e_1^-$ and $\pi^+ = e_n^+$ are the *initial* and the *terminal* vertex of π. The number $|\pi| = n$ is the *length* of the path. The *label* of π is $\ell(\pi) = \ell(e_1)\ell(e_2)\cdots\ell(e_n) \in \mathbf{\Sigma}^*$. We also admit the *empty path* starting and ending at a vertex x, whose label is ϵ. Denote by $\Pi_{x,y} = \Pi_{x,y}(X)$ the set of all paths π in X with initial vertex $\pi^- = x$ and terminal vertex $\pi^+ = y$. The following needs no proof.

(2.1) Lemma/Definition. *Let (X, E, ℓ) be a labelled graph, $x \in X$ and $w \in \mathbf{\Sigma}^*$. We define $\Pi_x(w) = \{\pi : \pi^- = x\,,\ \ell(\pi) = w\}$, the set of all paths that start at x and have label w. The set of all terminal vertices of those paths is denoted $x^w = \{\pi^+ : \pi \in \Pi_x(w)\}$.*

Analogously, we define $\overline{\Pi}_x(w) = \{\pi : \pi^+ = x\,,\ \ell(\pi) = w\}$, the set of all paths that terminate at x and have label w, and write $x^{-w} = \{\pi^- : \pi \in \overline{\Pi}_x(w)\}$.

If X is fully labelled, then $\Pi_x(w)$ is always non-empty.

If X is deterministic, then $\Pi_x(w)$ has at most one element, and if that element exists, it is denoted $\pi_x(w)$, while x^w just denotes its endpoint.

If X is fully deterministic, then x^w is a unique vertex of X for every $x \in X$, $w \in \mathbf{\Sigma}^*$.

Finally, if X is symmetric (not necessarily deterministic), then $\overline{\Pi}_x(w) = \Pi_x(w^{-1})$, where for $w = a_1 \cdots a_n$, one defines $w^{-1} = a_n^{-1} \cdots a_1^{-1}$.

With a labelled, directed graph as above, we can associate various languages. We can, e.g., consider the language

$$L_{x,y} = L_{x,y}(X) = \{\ell(\pi) : \pi \in \Pi_{x,y}(X)\}\,, \quad \text{where } \ x, y \in X\,. \tag{2.2}$$

(2.3) Definition. Let G be a finitely generated group, K a subgroup and $\psi : \mathbf{\Sigma} \to G$ a semigroup presentation of G. The *Schreier graph* $X = X(G, K, \psi)$ has vertex set

$$X = K\backslash G = \{Kg : g \in G\}$$

(the set of all right K-cosets in G), and the set of labelled, directed edges

$$E = \{e = (x, a, y) : x = Kg\,,\ y = Kg\psi(a)\,,\ \text{where } g \in G\,,\ a \in \mathbf{\Sigma}\}\,.$$

X is a rooted graph with origin $o = K$, the right coset corresponding to the neutral element 1_G of the group G. The Schreier graph is fully deterministic. It is also *strongly connected:* for every pair $x, y \in X$, there is a path from x to y. (This follows from the fact that $\psi(\mathbf{\Sigma})$ generates G as a semigroup.) When $K = \{1_G\}$ then we write $X(G, \psi)$. This is the *Cayley graph* of G with respect to ψ, or more loosely speaking, with respect to the set $\psi(\mathbf{\Sigma})$ of generators.

Note that X can have the loop $e = (x, a, x) \in E$ with $x = Kg$. This holds if and only if $\psi(a) \in g^{-1}Kg$. It can also have the multiple edges $e_1 = (x, a_1, y)$ and $e_2 = (x, a_2, y)$ with $x = Kg$ and $a_1 \neq a_2$ if and only if $\psi(a_2)\psi(a_1)^{-1} \in g^{-1}Kg$. In particular, there might be multiple loops. The following is obvious.

(2.4) Lemma. *Let K be a subgroup of G and $\psi : \mathbf{\Sigma} \to G$ be a semigroup presentation of G. Then*

$$L(G, K, \psi) = L_{o,o}(X)$$

is the language of all labels of closed paths starting and ending at $o = K$ in the Schreier graph $X(G, K, \psi)$.

A context-free grammar $\mathcal{C} = (\mathbf{V}, \mathbf{\Sigma}, \mathbf{P}, S)$ and the language $L(\mathcal{C})$ are called *linear,* if every production rule in $\mathbf{P}$ is of the form $T \vdash v_1Uv_2$ or $T \vdash v$, where $v, v_1, v_2 \in \mathbf{\Sigma}^*$ and $T, U \in \mathbf{V}$. If furthermore in this situation one always has $v_2 = \epsilon$ (the empty word), then grammar and language are called *right linear* or *regular.*

A *finite automaton* $\mathcal{A}$ consits of a finite directed graph $X = (X, E, \ell)$ with label set $\mathbf{\Sigma}$ and labelling map ℓ, together with a root vertex o and a nonempty set $F \subset X$. The vertices of X are called the *states* of $\mathcal{A}$, the root o is the *initial* state, and the elements of F are the *final* states. The automaton is called *(fully) deterministic* provided the labelled graph X is (fully) deterministic. The language *accepted* by $\mathcal{A}$ is

$$L(\mathcal{A}) = \bigcup_{x \in F} L_{o,x}(X)\,.$$

If $\mathcal{A}$ is deterministic, then for each $w \in L(\mathcal{A})$ there is a unique path $\pi \in \bigcup_{x\in F} \pi_{o,x}(X)$ such that $\ell(\pi) = w$. A state $y \in X$ is called *useful* if there is some word $w \in L$ such that the vertex y lies on a path in $\bigcup_{x\in F} \pi_{o,x}(X)$ with label w. It is clear that we can remove all useless states and their ingoing and outgoing edges to obtain an automaton which accepts the same language and is *reduced:* it has only useful states. It is well known [5, Chapter 2] that a language $L \subseteq \mathbf{\Sigma}^*$ is regular if and only if L is accepted by some deterministic finite automaton.

The following generalizes Anisimov's [1] characteriziation of groups with regular word problem, and also simplifies its proof, as well as the simpler one of [10, Lemma 1].

(2.5) Proposition. *Let G be a finitely generated group, K a subgroup and $\psi : \Sigma \to G$ a semigroup presentation of G. Then (G, K) has regular word problem with respect to ψ if and only if K has finite index in G.*

Proof. Suppose first that the index of K in G is finite. Consider the finite automaton $\mathcal{A} = (X, o, \{o\})$ where X is the Schreier graph $X(G, K, \psi)$, and the initial and unique final state is $o = K$ (as a vertex of X). Then $L(G,K,\psi) = L(\mathcal{A})$: indeed, $w \in \mathbf{\Sigma}^*$ belongs to $L(G,K,\psi)$, i.e. $\psi(w) \in K$, if and only if $K = K\psi(w)$. This shows that $L(G,K,\psi)$ is regular.

Conversely, suppose that $L = L(G,K,\psi)$ is regular and accepted by the reduced, deterministic finite automaton $\mathcal{A} = (X,o,F)$. For $y \in X$ there is some word $w \in L$ such that the vertex y lies on the unique path from o to F with label w. We choose one such w and let w_y be the label of the final piece of the path, starting at y and ending at F. We set $g_y = \psi(w_y)^{-1} \in G$.

Let $g \in G$. There are $w, \overline{w} \in \mathbf{\Sigma}^*$ with $\psi(w) = g$ and $\psi(\overline{w}) = g^{-1}$. Thus, $w\overline{w} \in L = L(G,K,\psi)$, and there is a (unique) path π with label $w\overline{w}$ from o to some final state. Now consider the initial piece π_w of π, that is, the path starting at o whose label is our w that we started with. [Thus, we have proved that such a path π_w must exist in X !] Let y be the final state (vertex) of π_w. Then clearly $ww_y \in L(\mathcal{A})$, which means that $gg_y^{-1} = \psi(ww_y) \in K$. Since $\psi(\mathbf{\Sigma}^*) = G$, it follows that

$$G = \bigcup_{y \in X} Kg_y\,,$$

and K has finitely many cosets in G.

□

(2.6) Corollary. *Let G be finitely generated and K a subgroup. Then the property of the pair (G,K) to have a regular word problem is independent of the semigroup presentation of G.*

We shall see that the same also holds in the context-free case. Another corollary that we see from the proof of Proposition 2.5 is the following.

(2.7) Corollary. *Let G be finitely generated and K a subgroup with finite index. Then for any semigroup presentation $\psi : \mathbf{\Sigma} \to G$, any reduced deterministic automaton $\mathcal{A} = (X,o,F)$ that accepts $L(G,K,\psi)$ has a surjective homomorphism (as a labelled oriented graph with root o) onto the Schreier graph $X(G,K,\psi)$. Also, at each vertex of X there is an outgoing edge with label a for each $a \in \mathbf{\Sigma}$.*

Proof. Let $\mathcal{A} = (X,o,F)$ be deterministic and reduced, as in part 2 of the proof of Proposition 2.5.

Let $y \in X$, and recall the construction of the label w_y of a path from y to F, and $g_y = \psi(w_y)^{-1} \in G$. If v is another path from y to F, and $h = \psi(v)^{-1}$, then we can take $w \in L_{o,y}$ (which we know to be non-empty) and find that $ww_y, wv \in L(G,K,\psi)$, so that $\psi(w) \in Kg_y \cap Kh$. Thus $Kg_y = K\psi(w) = Kh$, and the map $\kappa : X \to K\backslash G,\ y \mapsto Kg_y$ is well defined. It has the property that when $w \in L_{o,y}$, then $K\psi(w) = Kg_y$. The map κ is clearly surjective, and $\kappa(o) = K$ by construction.

Now let $y \in X$ and $a \in \mathbf{\Sigma}$. Take $w \in L_{o,y}$ and consider the word wa. Again by part 2 of the proof of Proposition 2.5, there is a unique path π_{wa} in X starting at o with label wa. If y is its final vertex, then there is the edge $e = (y,a,z)$ in X. In this situation, $\kappa(z) = K\psi(wa) = Kg_y\psi(a) = \kappa(y)\psi(a)$. This means that in the Schreier graph, there is the edge with label a from $\kappa(y)$ to $\kappa(z)$. Therefore κ is a homomorphism of labelled graphs. □

The following simple example shows that, in general, the map κ constructed in the proof of the previous corollary is not injective.

Let $G = \mathbb{Z}_2 = \{1,\mathrm{t}\}$ be the group of order two and $K = \{1\}$ the trivial subgroup. Let $\mathbf{\Sigma} = \{a\}$ and consider the presentation $\psi \colon \mathbf{\Sigma} \to G$ such that $\psi(a) = t$. Then $L(G, K, \psi) = \{a^{2n} : n \geq 0\}$.

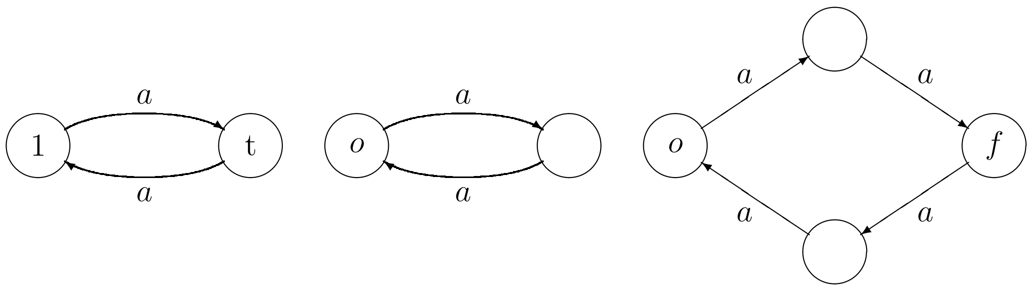

Figure 1

In Figure 1 above we have represented, in order, the Schreier graph $X(G, K, \psi)$ (which is nothing but the Cayley graph of G w.r. to ψ), and two automata $\mathcal{A}_1$ and $\mathcal{A}_2$. As usual o denotes the origin, while the sets of final states are $F_1 = \{o\}$ and $F_2 = \{o, f\}$, respectively. We have $L(\mathcal{A}_1) = L(\mathcal{A}_2) = L(G, K, \psi)$.

3. Pushdown automata

Besides grammars, we shall need another instrument for generating context-free languages. A *pushdown automaton* is a 7-tuple $\mathcal{A} = (\mathbf{Q}, \mathbf{\Sigma}, \mathbf{Z}, \delta, q_0, \mathbf{Q}_f, z_0)$, where $\mathbf{Q}$ is a finite set of *states,* $\mathbf{\Sigma}$ the input alphabet as above, $\mathbf{Z}$ a finite set of *stack symbols,* $q_0 \in Q$ the initial state, $\mathbf{Q}_f \subset \mathbf{Q}$ the set of final states, and $z_0 \in Z$ is the *start symbol.* Finally, the function $\delta : \mathbf{Q} \times (\mathbf{\Sigma} \cup \{\epsilon\}) \times (\mathbf{Z} \cup \{\epsilon\}) \to \mathcal{P}_{\mathrm{fin}}(\mathbf{Q} \times \mathbf{Z}^*)$ is the *transition function.* Here, $\mathcal{P}_{\mathrm{fin}}(\mathbf{Q} \times \mathbf{Z}^*)$ stands for the collection of all finite subsets of $\mathbf{Q} \times \mathbf{Z}^*$.

The autmaton works in the following way. At any time, it is in some state $p \in \mathbf{Q}$, and the stack contains a word $\zeta \in \mathbf{Z}^*$. The automaton reads a word $w \in \mathbf{\Sigma}^*$ from the "input tape" letter by letter from left to right. If the current letter of w is a, the state is p and the top (=rightmost) symbol of the stack word ζ is z, it performs one of the following transitions.

(i) $\mathcal{A}$ selects some $(q, \zeta') \in \delta(p, a, z)$, changes into state q, moves to the next position on the input tape (it may be empty if a was the last letter of w), and replaces the rightmost symbol z of ζ by ζ', or

(ii) $\mathcal{A}$ selects some $(q, \zeta') \in \delta(p, \epsilon, z)$, changes into state q, remains at the current position on the input tape (so that a has to be treated later), and replaces the rightmost symbol z of ζ by ζ'.

If both $\delta(p, a, z)$ and $\delta(p, \epsilon, z)$ are empty then $\mathcal{A}$ halts.

The automaton is also allowed to continue to work when the stack is empty, i.e., when $\zeta = \epsilon$. Then the automaton acts in the same way, by putting ζ' in the stack when it has selected $(q, \zeta') \in \delta(p, a, \epsilon)$ in case (i), resp. $(q, \zeta') \in \delta(p, \epsilon, \epsilon)$ in case (ii).

We say that $\mathcal{A}$ *accepts* a word $w \in \mathbf{\Sigma}^*$ if starting at the state q_0 with only z_0 in the stack and with w on the input tape, after finitely many transitions the automaton can reach a final state with empty stack and empty input tape. The language accepted by $\mathcal{A}$ is denoted $L(\mathcal{A})$.

The pushdown automaton is called *deterministic* if for any $p \in \mathbf{Q}$, $a \in \mathbf{\Sigma}$ and $z \in \mathbf{Z} \cup \{\epsilon\}$, it has at most one option what to do next, that is,

$$|\delta(p, a, z)| + |\delta(p, \epsilon, z)| \le 1\,.$$

(Here, $|\cdot|$ denotes cardinality.)

It is well known [5] that a language is context-free if and only if it is accepted by some pushdown automaton. A context-free language is called *deterministic* if it is accepted by a deterministic pushdown automaton. We also remark here that a deterministic context-free language L is *un-ambiguous*, which means that it is generated by some context-free grammar in which every word of L has precisely one rightmost derivation.

The following lemma is modelled after the indications of [10, Lemma 2]. For the sake of completeness, we include the full proof.

(3.1) Lemma. *Suppose that $G, K, \mathbf{\Sigma}$ and $\psi : \mathbf{\Sigma} \to G$ are as above. Let H be a finitely generated subgroup of G, and let $\mathbf{\Sigma}'$ be another alphabet and $\psi' : \mathbf{\Sigma}' \to H$ be such that $F' = \psi'(\mathbf{\Sigma}')$ generates H as a semigroup.*

Then, if $L(G, K, \psi)$ is context-free, also $L(H, K \cap H, \psi')$ is context-free, and if in addition $L(G, K, \psi)$ is deterministic, then so is $L(H, K \cap H, \psi')$.

Proof. We start with a pushdown automaton $\mathcal{A} = (\mathbf{Q}, \mathbf{\Sigma}, \mathbf{Z}, \delta, q_0, \mathbf{Q}_f, z_0)$ that accepts $L(G, K, \psi)$.

For each $b \in \mathbf{\Sigma}'$, there is $u(b) \in \mathbf{\Sigma}^*$ such that $\psi'(b) = \psi\big(u(b)\big)$, and we may choose $u(b)$ to have length ≥ 1. Thus,

$$w' = b_1 \cdots b_n \in L(H, K \cap H, \psi') \iff u(b_1) \cdots u(b_n) \in L(G, K, \psi)\,.$$

With this in mind, we modify $\mathcal{A}$ in order to obtain a pushdown automaton $\mathcal{A}'$ that accepts $L(H, K \cap H, \psi')$. Our $\mathcal{A}'$ has to translate any $w' = b_1 \cdots b_n \in (\mathbf{\Sigma}')^*$ into $w = u(b_1) \cdots u(b_n) \in \mathbf{\Sigma}^*$ and to use $\mathcal{A}$ in order to check whether $w \in L(G, K, \psi)$.

Let $m + 1 = \max\{|u(b)| : b \in \mathbf{\Sigma}'\}$. If $m = 0$ then the only modification of $\mathcal{A}$ needed is to replace $\mathbf{\Sigma}$ by its subset $\mathbf{\Sigma}'$ and to use the resulting restriction of the transition function.

Otherwise, we set $\mathbf{\Sigma}_m = \mathbf{\Sigma} \cup \mathbf{\Sigma}^2 \cup \cdots \cup \mathbf{\Sigma}^m$. For $v \in \mathbf{\Sigma}^+ = \mathbf{\Sigma}^* \setminus \{\epsilon\}$, we denote by v_+ its subword obtained by deleting the first letter. We define $\mathbf{Q}' = \mathbf{Q} \cup (\mathbf{Q} \times \mathbf{\Sigma}_m)$ and $\mathcal{A}' = (\mathbf{Q}', \mathbf{\Sigma}', \mathbf{Z}, \delta', q_0, \mathbf{Q}_f, z_0)$ with the transition function δ' as follows. For each $p \in \mathbf{Q}$

and $z \in \mathbf{Z}$,

$$\begin{aligned}
\delta'(p,\epsilon,z) &= \delta(p,\epsilon,z)\,,\\
\delta'(p,b,z) &= \delta\big(p,a,z\big)\,, \quad \text{if } u(b)=a\in\mathbf{\Sigma}\,,\\
\delta'(p,b,z) &= \Big\{\big((q,u(b)_+),\zeta\big) : (q,\zeta)\in\delta(p,a,z)\Big\}\,, \qquad\qquad \text{if } u(b)\in a\mathbf{\Sigma}^+\,,\\
\delta'\big((p,v),\epsilon,z\big) &= \Big\{\big((q,v),\zeta\big) : (q,\zeta)\in\delta(p,\epsilon,z)\Big\}\\
&\qquad \cup\ \Big\{\big((q,v_+),\zeta\big) : (q,\zeta)\in\delta(p,a,z)\Big\}\,, \qquad\qquad \text{if } v\in a\mathbf{\Sigma}^+\,,\\
\delta'\big((p,a),\epsilon,z\big) &= \Big\{\big((q,a),\zeta\big) : (q,\zeta)\in\delta(p,\epsilon,z)\Big\}\ \cup\ \delta(p,a,z)\,, \qquad \text{if } a\in\mathbf{\Sigma}\,.
\end{aligned}$$

Thus, the new states of the form (p,v) with $1 \le |v| < m$ serve to remember the terminal parts v of the words $u(b)$, $b \in \mathbf{\Sigma}'$. This automaton accepts $L(G,K,\psi')$, and it is deterministic, if $\mathcal{A}$ has this property. $\square$

(3.2) Corollary. *Being context-free is a property of the pair (G,K) that does not depend on the specific choice of the alphabet $\mathbf{\Sigma}$ and the map $\psi : \mathbf{\Sigma} \to G$ for which $\psi(\mathbf{\Sigma})$ generates G as a semigroup.*

Therefore, it is justified to refer to the contxt-free pair (G,K) rather than to the triple (G,K,ψ). Furthermore, whenever this is useful, we may restrict attention to the case when the graph $X(G,K,\psi)$ is symmetric: we say that ψ *is symmetric,* if there is a proper involution $a \mapsto a^{-1}$ of $\mathbf{\Sigma}$ such that $\psi(a^{-1}) = \psi(a)^{-1}$ in G. (Again, it is not necessary to assume that ψ is one-to-one, so that we have that $a^{-1} \ne a$ even when $\psi(a)^2 = 1_G$.)

(3.3) Proposition. *Let G be finitely generated, H be a subgroup with $[G:H] < \infty$. If K is a subgroup of H then (G,K) is context-free if and only if (H,K) is context-free.*

Proof. The "only if" is contained in Lemma 3.1. (Observe that H inherits finite generation from G, since $[G:H] < \infty$.)

For the converse, we assume that (H,K) is context-free and let $\psi : \mathbf{\Sigma} \to H$ and $\psi' : \mathbf{\Sigma}' \to G$ be semigroup presentations of H and G, respectively. There is a pushdown automaton $\mathcal{A} = (\mathbf{Q}, \mathbf{\Sigma}, \mathbf{Z}, \delta, q_0, \mathbf{Q}_f, z_0)$ that accepts $L(H,K,\psi)$.

Let F be a set of representatives of the right cosets of H in G, with $1_G \in F$. Thus, $|F| < \infty$, and

$$G = \biguplus_{g\in F} Hg\,,$$

For every $g \in F$ and $b \in \mathbf{\Sigma}'$ there is a unique $\bar{g} = \bar{g}(g,b) \in F$ such that $g\psi'(b) \in H\bar{g}$. Therefore there is a word $u = u(g,b) \in \mathbf{\Sigma}^*$ such that

$$g\psi'(b) = \psi\big(u(g,b)\big)\bar{g}(g,b)\,.$$

An input word $w = b_1 \cdots b_n$ is transformed recursively into $u_1 \cdots u_n$, along with the sequence $g_0, g_1, \ldots, g_n$ of elements of F that indicate the current H-coset at each step:

$$g_0 = 1_G\,; \quad u_k = u(g_{k-1}, b_k) \quad \text{and} \quad g_k = \bar{g}(g_{k-1}, b_k)\,.$$

Then $\psi'(w) \in K$ if and only if $g_n = 1_G$ and $\psi(u_1 \cdots u_n) \in K$.

Thus, our new automaton $\mathcal{A}'$ recalls at each step the current coset Hg_{k-1}, which is multiplied on the right by $\psi(b_k)$, where b_k is the next input letter. Then the new coset is $H\bar{g}(g_{k-1}, b_k)$, and $\mathcal{A}'$ simulates what $\mathcal{A}$ does next upon reading $u(g_{k-1}, b_k)$. Then w is accepted when at the end the coset is $H = H1_G$ and $\mathcal{A}$ is in a final state.

The simple task to write down this automaton in detail is left to the reader. $\square$

4. Context-free graphs

In this section, we assume that (X, E, ℓ) is symmetric. We may think of each pair of oppositely oriented edges (x, a, y) and (y, a^{-1}, x) as one non-oriented edge, so that X becomes an ordinary graph with symmetric neighbourhood relation, but possibly multiple edges and loops. If it is in addition fully deterministic, then X is a *regular graph*, that is, the number of outgoing edges (which coincides with the number of ingoing edges) at each vertex is $|\mathbf{\Sigma}|$. Attention: if we consider non-oriented edges, then each loop at x has to be counted twice, since it corresponds to two oriented edges of the form (x, a, x) and (x, a^{-1}, x). For all our purposes it is natural to require that X is *connected:* for any pair of vertices x, y there is a path from x to y. The distance $d(x, y)$ is the minimum length (number of edges) of a path from x to y, which defines the integer-valued *graph metric.* A *geodesic path* is one whose length is the distance between its endpoints.

We select a finite, non-empty subset F of X and consider the balls $B(F, n) = \{x : d(x, F) \leq n\}$ (where $d(x, F) = \min\{d(x, y) : y \in F\}$). If we delete $B(F, n)$ then the induced graph $X \setminus B(F, n)$ will fall apart into a finite number of connected components, called *cones* with respect to F. Each cone is a labelled, symmetric graph C with the *boundary* ∂C consisting of all vertices x in C having a neighbour outside C (i.e., in $B(F, n)$).

The following notion was introduced in [11] for symmetric, labelled graphs and $F = \{o\}$.

(4.1) Definition. The graph X is called context-free with respect to F if there is only a finite number of isomorphism types of the cones with respect to F as labelled graphs with boundary.

This means that there are finitley many cones $C_1, \dots, C_r$ (generally with respect to different radii n) such that for each cone C, we can fix a bijection ϕ_C from (the vertex set of) C to precisely one of the C_i to (the vertex set of) C, this bijection sends ∂C to ∂C_i, and (x, a, y) is an edge with both endpoints in C if and only if its image$\bigl(\phi_C(x), a, \phi_C(y)\bigr)$ is an edge of C_i. In this case, we say that C is a cone of *type* i.

Generally, as in [11], we are interested in the case when $F = \{o\}$ (or any other singleton), but there is at least one point where it will be useful to admit arbitrary finite, non-empty F.

Another natural notion of context-freeness of X with respect to o is to require that the language $L_{o,o}(X)$ is context-free. We shall see that for deterministic, symmetric graphs this is equivalent with context-freeness with respect to o in the sense of Definition 4.1. One direction of this equivalence is practically contained in [11], but not stated explicitly except for the case of Cayley graphs of groups. The other direction (that context-freeness of $L_{o,o}$ implies that of the graph) is shown in [11] only for Cayley graphs of groups, which is substantially simpler than the general case treated below in Theorem 4.7.

(4.2) Theorem. *If the symmetric, labelled graph (X, E, ℓ) with label alphabet $\mathbf{\Sigma}$ is context-free with respect to the finite, non-empty set $F \subset X$, then $L_{x,y}$ is a context-free language for all $x, y \in X$. Furthermore, if the graph X is deterministic, then so is the context-free language $L_{x,y}$.*

Proof. Just for the purpose of this proof, we write $x_0\,, y_0$ instead of x, y for the vertices for which L_{x_0,y_0} will be shown to be context-free. We may assume without loss of generality that $x_0\,, y_0$ in F. Indeed, if this is not the case, then we can replace F by $F' = B(F, n)$, which contains x_0 and y_0 when n is sufficiently large. The cones with respect to F' are also cones with respect to F, so that X is also context-free with respect to F'.

Similarly to [11, Lemma 2.3], we construct a deterministic pushdown automaton that accepts L_{x_0,y_0} .

We consider also the whole graph X as a cone C_0 with boundary F, which we keep apart from the other representatives $C_1, \dots, C_r$ of cones.

If C is a cone, then as a component of $X \setminus B(F, n)$ for some $n \geq 0$ it must be a *successor* of another cone C^-. The latter is the unique component of $X \setminus B(F, n-1)$ that contains C, when $n \geq 1$, while it is $C_0 = X$ when $n = 0$. We also call C^- the *predecessor* of C.

Different cones of type $j \in \{1, \dots, r\}$ may have predecessors of different types. Conversely, a cone C of type $i \in \{0, \dots, r\}$ may have none, one or more than one successors of type j, and the number $d_{i,j}$ of those successors depends only on i and j. In the representative cone C_i , we choose and fix a numbering of the distinct successors of type j as $C_{i,j}^k$, $k = 1, \dots, d_{i,j}$. If C is any cone with type i then we use the isomorphism $\phi_C : C \to C_i$ to transport this numering to the successors of C that have type j, which allows us to identify the k-th successor of C with type j.

One can visualize the cone structure by a finite, oriented graph Γ with multiple edges and root 0: the vertex set is the set of cone types $i \in \{0, \dots, r\}$, and there are $d_{i,j}$ oriented edges, which we denote by $t_{i,j}^k$ $(k = 1, \dots, d_{i,j})$ from vertex i to vertex j $(i \geq 0,\ j \geq 1)$.

Every vertex x of X belongs to the boundary of precisely one cone $C = C(x)$ with respect to F. We define the *type* i of x as the type of $C(x)$. Under the mapping ϕ_C, our x corresponds to precisely one element of ∂C_i. We write $\phi(x)$ for that element, without subscript C, so that ϕ maps X onto $\bigcup_i \partial C_i$. In particular, $\phi(x) = x$ for every $x \in F$.

Let $y \in X \setminus F$ with type j. Then there is i (depending on y) such that every neighbour x of y with $d(x, F) = d(y, F) - 1$ has type i, and there is precisely one successor cone $C_{i,j}^k$ of C_i that contains $\phi_{C(x)}(y)$. In this case, we write $\tau(y) = t_{i,j}^k$, the *second order type* of y. Compare with [11]. If y' is such that $C(y') = C(y)$ then $\tau(y') = \tau(y)$.

We now finally construct the required pushdown automaton $\mathcal{A}$. (Comparing with [11], we use more states and stack symbols, which facilitates the description.) The set of states and stack symbols are

$$\mathbf{Q} = \biguplus_{i=0}^{r} \partial C_i \quad \text{and} \quad \mathbf{Z} = F \cup \left\{ t_{i,j}^k : i = 1, \dots, r\,,\ j = 0, \dots, r\,,\ k = 1, \dots, d_{i,j} \right\}.$$

(When $d_{i,j} = 0$ then there is no $z^k_{i,j}$.) Note that both sets contain F. In order to generate the language L_{x_0,y_0}, where $x_0, y_0 \in F$, then we use x_0 as the initial state and y_0 as the (only) final state. We describe the transition function, which – like $\mathbf{Q}$ and $\mathbf{Z}$ – does not depend on x_0, y_0.

We want to read an input word, which has to correspond to the label starting at x_0. Inside the subgraph of X induced by F, our $\mathcal{A}$ behaves just like that subgraph, seen as a finite automaton.

Outside of F, it works as follows. At the m-th step, the automaton will be in a state that descibes the m-th vertex, say x, of that path, by identifying x as above with the element $\phi(x)$ of C_j, where j is the type of x. The current stack symbol is of the form $t^k_{i,j}$ and serves to recall that x lies in the k-th successor cone of type j of a cone with type i. If the next vertex along the path, say y, satisfies $d(y,F) = d(x,F) + 1$, and y has type j' then the state is changed to $\phi(y) \in C_{j'}$, and the symbol $z^{k'}_{j,j'} = \tau(y)$ is added to the stack. If $d(y,F) = d(x,F)$, then only the state is changed from $\phi(x)$ to $\phi(y)$. Finally, if $d(y,F) = d(x,F) - 1$ then the new state is again $\phi(y)$, while the top symbol in the stack is deleted. Formally, we get the following list of transition rules.

If $x \in F = \mathbf{Q} \cap \mathbf{Z}$:

$$\begin{aligned} \delta(x,a,x) \;&=\; \big\{(y,y) : (x,a,y) \in E\,,\; y \in F\big\} \\ &\quad\cup \big\{\big(\phi(y), x\tau(y)\big) : (x,a,y) \in E\,,\; d(y,F) = 1\big\}. \end{aligned}$$

If $x \in X \setminus F$:

$$\begin{aligned} \delta\big(\phi(x), a, \tau(x)\big) \;&=\; \big\{\big(\phi(y), a, \tau(x)\tau(y)\big) : (x,a,y) \in E\,,\; d(y,F) = d(x,F) + 1\big\} \\ &\quad\cup \big\{\big(\phi(y), \tau(y) = \tau(x)\big) : (x,a,y) \in E\,,\; d(y,F) = d(x,F)\big\} \\ &\quad\cup \big\{\big(\phi(y), \epsilon\big) : (x,a,y) \in E\,,\; d(y,F) = d(x,F) - 1\big\} \end{aligned}$$

This is a finite collection of transitions, since $\phi(\cdot)$ and $\tau(\cdot)$ can take only finitely many different values.

In view of the above explanations, $\mathcal{A}$ accepts L_{x_0,y_0}. Also, when the graph X is deterministic, then so is $\mathcal{A}$. □

Before proving a converse of Theorem 4.2, we first need some preliminaries, and start by recalling a fact proved in [10] and [11], see also Woess [18] and Berstel and Boasson [2].

(4.3) Lemma. *If $L_{o,o}$ is context-free then there is a constant M such that for each cone C with respect to o, one has* $\mathrm{diam}(\partial C) \le M$.

(The diameter is of course taken with respect to the graph metric.) We shall see below how to deduce this, but it is good to know it in advance.

A context-free grammar $\mathcal{C} = (\mathbf{V}, \mathbf{\Sigma}, \mathbf{P}, S)$ is said to have *Chomsky normal form (CNF)*, if (i) every production rule is of the form $T \vdash U\hat{U}$ or $T \vdash a$, where $U, \hat{U} \in \mathbf{V}$ (not necessariliy distinct), resp. $a \in \mathbf{\Sigma}$, and (ii) if $\epsilon \in L(\mathcal{C})$, then there is the rule $S \vdash \epsilon$, and S is not contained in the right hand side of any production rule.

With a slight deviation from [10], we associate with each $w = a_1 \cdots a_n \in L(\mathcal{C})$, $n \ge 2$ a labelled (closed) *polygon* $\mathsf{P}(w)$ with length $n+1$. As a directed graph, it has distinct vertices $t_0, t_1, \ldots, t_n$ and labelled edges (t_{i-1}, a_i, t_i), $i = 1, \ldots, n$, plus the edge (t_0, S, t_n). A *(diagonal) triangulation* of $\mathsf{P}(w)$ is a plane triangulation of $\mathsf{P}(w)$ obtained by inserting only diagonals. Here, we specify those diagonals as oriented, labelled edges (t_i, T, t_j), where t_i, t_j are not neighbours in $\mathsf{P}(w)$ and $T \in \mathbf{V}$. Furthermore, we will never have two diagonals between the same pair of vertices of $\mathsf{P}(w)$. (If $|w| \leq 2$ we consider $\mathsf{P}(w)$ itself triangulated.) The proof of the following Lemma may help to make the construction of [10] (used for Cayley graphs of groups) more transparent.

(4.4) Lemma. *If $\mathcal{C} = (\mathbf{V}, \mathbf{\Sigma}, \mathbf{P}, S)$ is in CNF and $w = a_1 \cdots a_n \in L(\mathcal{C})$ with $n \geq 2$ then there is a diagonal triangulation of $\mathsf{P}(w)$ with the property that whenever (t_i, T, t_j) is a diagonal edge, then T occurs in a derivation $S \stackrel{*}{\Longrightarrow} w$, $j - i \geq 2$ and $T \stackrel{*}{\Longrightarrow} a_{i+1} \cdots a_j$.*

Proof. We start with a fixed derivation $S \stackrel{*}{\Longrightarrow} w$, and explain how to build up the triangles step by step. Suppose that $T \in \mathbf{V}$ occurs in our derivation, and that we have a "sub-derivation" $T \vdash U\hat{U} \stackrel{*}{\Longrightarrow} a_{i+1} \cdots a_k$, where $U, \hat{U} \in \mathbf{V}$. Then there is $j \in \{i+1, \ldots, k-1\}$ such that $U \stackrel{*}{\Longrightarrow} a_{i+1} \cdots a_j$ and $\hat{U} \stackrel{*}{\Longrightarrow} a_{j+1} \cdots a_k$. In this case, we draw a triangle with three oriented, labelled edges, namely the *'old'* edge (t_i, T, t_k) and the two *'new'* edges (t_i, U, t_j) and $(t_j, \hat{U}, t_k)$.

If we have the derivation $S \stackrel{*}{\Longrightarrow} a_1 \cdots a_n$, then it uses successive steps of the form $T \vdash U\hat{U}$ with $U\hat{U} \stackrel{*}{\Longrightarrow} a_{i+1} \cdots a_k$ as above. We work through these steps one after the other, starting with $S \vdash T_1\hat{T}_1$, where $T_1 \stackrel{*}{\Longrightarrow} a_1 \ldots a_k$ and $\hat{T}_1 \stackrel{*}{\Longrightarrow} a_{k+1} \cdots a_n$. The first triangle has the 'old' edge (t_0, S, t_n) and the 'new' edges (t_0, T_1, t_k) and $(t_k, \hat{T}_1, t_n)$.

At any successive step, we take one of the 'new' edges (t_i, T, t_k), where $k - i \geq 2$ and proceed as explained at the beginning, so that we add two 'new' edges that make up a triangle together with (t_i, T, t_k), which is then declared 'old'. We continue until all derivation steps of the form $T \vdash U\hat{U}$ in our derivation $S \stackrel{*}{\Longrightarrow} w$ are exhausted. At this point, we have obtained a tiling of triangles that constitute a diagonal triangulation of its outer polygon, whose edges have the form (t_0, S, t_n) and (t_{i-1}, U_i, t_i) with $U_i \in \mathbf{V}$, $i = 1, \ldots, n$. The only steps of our derivation that we have not yet considered are the terminal ones $U_i \vdash a_i$. Thus, we conclude by replacing the label U_i of (t_{i-1}, U_i, t_i) by a_i. $\square$

The construction is best understood by considering an example: suppose our rightmost derivation is

$$
\begin{array}{lll}
S \vdash T_1\hat{T}_1 & \Longrightarrow T_1(T_2\hat{T}_2) & \Longrightarrow T_1(T_2(T_3\hat{T}_3)) \\
& \Longrightarrow T_1(T_2(T_3a_6)) & \Longrightarrow T_1(T_2((T_4\hat{T}_4)a_6)) \\
& \Longrightarrow T_1(T_2((T_4a_5)a_6)) & \Longrightarrow T_1(T_2((a_4a_5)a_6)) \\
& \Longrightarrow T_1(a_3((a_4a_5)a_6)) & \Longrightarrow T_1(a_3((a_4a_5)a_6)) \\
& \Longrightarrow (T_5\hat{T}_5)(a_3((a_4a_5)a_6)) & \Longrightarrow (T_5a_2)(a_3((a_4a_5)a_6)) \\
& \Longrightarrow (a_1a_2)(a_3((a_4a_5)a_6)) &
\end{array}
$$

(We have inserted the parentheses to make the rules that we used in each step more visible.) The associated triangulation is as follows.

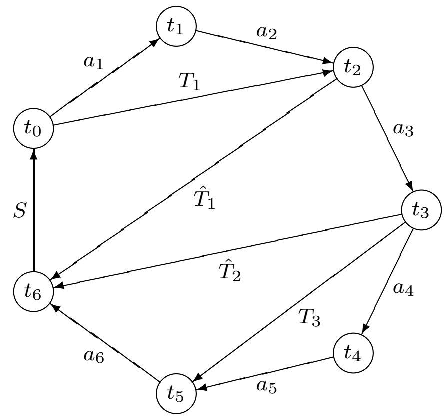

Figure 2

The variables of the terminal rules $T_5 \vdash a_1$, $\hat{T}_5 \vdash a_2$, $T_2 \vdash a_3$, $T_4 \vdash a_4$, $\hat{T}_4 \vdash a_5$ and $\hat{T}_3 \vdash a_6$ are not visible in this figure (but we might add them to the boundary edges). Apart from this, one can read the derivation $S \overset{*}{\Longrightarrow} w$ from the diagonalization in a similar way as it can be read from the so-called derivation tree (see e.g. [5, §1.6] for the latter.)

The following goes back to [10] in the case of (Cayley graphs of) finitely generated groups.

(4.5) Lemma. *Let $\mathcal{C} = (\mathbf{V}, \mathbf{\Sigma}, \mathbf{P}, S)$ be in CNF and $L(\mathcal{C}) = L_{x,y}(X)$, where X is a deterministic, symmetric graph. If $w = a_1 \cdots a_n \in L_{x,y}(X)$ and (t_i, T, t_j) is a diagonal edge in a triangulation of $\mathsf{P}(w)$ as in Lemma 4.4, then the vertices $\bar{x} = x^{a_1 \cdots a_i}$ and $\bar{y} = x^{a_1 \cdots a_j}$ of X satisfy $d(\bar{x}, \bar{y}) \le m(T)$, where*

$$d(\bar{x}, \bar{y}) \le m(T) = \min\{|w| : w \in L_T\}\,. \tag{4.6}$$

Proof. Since X is deterministic, Lemma 2.1 implies that $\pi_x(w)$ exists as the unique path with initial vertex x and label w. In particular, $\bar{x}$ and $\bar{y}$ lie on that path. Furthermore, we have $\bar{y} = y^{-a_{j+1} \cdots a_n}$.

Now let $v \in L_T$ with $|v| = m(T)$. Then by Lemma 4.4, T arises in a derivation $S \overset{*}{\Longrightarrow} a_1 \cdots a_i T a_{j+1} \cdots a_n \overset{*}{\Longrightarrow} w$. But then we also have $S \overset{*}{\Longrightarrow} a_1 \cdots a_i v a_{j+1} \cdots a_n$, a word in $L_{x,y}$. By Lemma 2.1, again using that X is symmetric and deterministic, $\bar{x}^v = y^{-a_{j+1} \cdots a_n} = \bar{y}$. Therefore, $\bar{x}$ and $\bar{y}$ are connected by a path with label v. Its length is $m(T)$. □

(4.7) Theorem. *Let (X, E, ℓ) be a fully deterministic, symmetric graph with label alphabet $\mathbf{\Sigma}$ and root o. If $L_{o,o}$ is a context-free language, then X is a context-free graph with respect to o, and in particular, $L_{o,o}$ is deterministic.*

Proof. There is a reduced grammar $\mathcal{C} = (\mathbf{V}, \mathbf{\Sigma}, \mathbf{P}, S)$ in CNF that generates $L_{o,o}$. Each of the languages L_T, $T \in \mathbf{V}$, is non-empty, only L_S contains ϵ, and we define

$$m = \max\{m(T) : T \in \mathbf{V}\}\,, \tag{4.8}$$

where $m(T)$ is as in (4.6).

Let C be a cone with respect to o such that $k = d(o, \partial C) > m$.

Construction of $\widetilde{D}(C)$. We define $D(C)$ as the subgraph of X induced by all vertices $y \in X$ with

$$d(o,x) = d(o,y) + d(x,y) \quad \text{and} \quad d(x,y) \le m \quad \text{for some } x \in \partial C\,.$$

In particular, y lies on some geodesic path from o to ∂C.

Now let $x_1, x_2 \in \partial C$, and consider some path $\pi \in \Pi_{x_1,x_2}(C)$ (i.e., it lies in C). Choose a geodesic path π_1 from o to x_1 and a geodesic path π_2 from x_2 to o. Then we can concatenate the three paths to a single path $\pi_1\pi\pi_2 \in \Pi_{o,o}$. Its label is the word $w = \ell(\pi_1)\ell(\pi)\ell(\pi_2) \in L_{o,o}$. Set $n = |w|$ and write

$$w = (a_1 \cdots a_k)(a_{k+1} \cdots a_{n-k})(a_{n-k+1} \cdots a_n)$$

where the 3 pieces in the parentheses are (in order) $\ell(\pi_1)$, $\ell(\pi)$ and $\ell(\pi_2)$. The words $\ell(\pi_1)$, $\ell(\pi)$ and $\ell(\pi_2)S$ are the labels of three consecutive arcs that fill the boundary of the polygon $\mathsf{P}(w)$. (To be precise, along the last edge of the 3rd arc, we are reading the label S in the reversed direction.) By [10, Lemma 5], its triangulation has a triangle which meets each of those arcs. (It may also occur that one corner of the triangle meets two arcs.) Thus, there are $i \in \{0,\dots,k\}$ and $i' \in \{k,\dots,n-k\}$ such that the vertices t_i and $t_{i'}$ of $\mathsf{P}(w)$ lie on that triangle. They correspond to the vertices $y_1 = o^{a_1\cdots a_i}$ and $y' = o^{a_1\cdots a_{i'}}$ of X. We either have $i' - i \le 1$, or else a diagonal $(t_i, U, t_{i'})$ is a side of our triangle. By Lemma 4.5, we get $d(y_1, y') \le m(U) \le m$. Thus $k \le i' \le d(o,y') \le i + m$, that is, $i \ge k - m > 0$. In particular, t_i does not lie on the third arc. In the same way, there is $j \in \{n-k,\dots,n-k+m\}$ (and not larger) such that t_j is a corner of our tiangle. This yields that there must be a "true" diagonal (t_i, T, t_j) of $\mathsf{P}(w)$. We set $v_1 = a_{i+1}\cdots a_k$ and $v_2 = a_{n-k+1}\cdots a_j$, so that $x_1 = y_1^{v_1}$, and let $y_2 = x_2^{a_{n-k+1}\cdots a_j}$. The points y_1 and y_2 are in $D(C)$, and by Lemma 4.4, $T \overset{*}{\Longrightarrow} v_1\ell(\pi)v_2$.

[It is here that we can see Lemma 4.3, since we deduce that $d(x_1,x_2) \le 3m$ for all $x_1, x_2 \in \partial C$.]

By Lemma 4.4, we also have

$$S \overset{*}{\Longrightarrow} a_1 \cdots a_i T a_{j+1} \cdots a_n\,,$$

so that $v \in L_T$ implies $a_1\cdots a_i v a_{j+1}\cdots a_n \in L_{o,o}$ and consequently $v \in L_{y_1,y_2}$, that is, $y_1^v = y_2$.

We now insert into $D(C)$ the additional labelled edge (y_1, v_1Tv_2, y_2), whose label is the word $v_1Tv_2 \in \mathbf{\Sigma}^*\mathbf{V}\mathbf{\Sigma}^*$. We insert all diagonals of the same type that can be obtained in the same way, and write $\widetilde{D}(C)$ for the resulting "edge-enrichment" of $D(C)$.

Subsuming, we have an edge (y_1, v_1Tv_2, y_2) in $\widetilde{D}(C)$ if and only if the following properties hold.

- $|v_i| \le m$ $(i = 1,2)$ and $T \in \mathbf{V}$,
- the path with label v_1 starting at y_1 and ending at $x_1 = y_1^{v_1} \in \partial C$ is part of a geodesic from o to x_1,
- the path with label v_2 starting at $x_2 = y_2^{-v_2} \in \partial C$ and ending at y_2 is part of a geodesic from x_2 to o, and
- there is a path π in C from x_1 to x_2 such that $T \overset{*}{\Longrightarrow} v_1\ell(\pi)v_2$,
- if $T \overset{*}{\Longrightarrow} v \in \mathbf{\Sigma}^*$ then v is the label of a path in Π_{y_1,y_2}.

Now, there are only finitely many cones C with respect to o with $d(\partial C, o) \le m$. On the other hand, for all cones C with $d(\partial C, o) \ge m$, there is a bound on the number of vertices of $\widetilde{D}(C)$, as well as on the number of possible labels on its edges. In particular, there are only finitely many possible isomorphism types of the labelled graphs $\bigl(\widetilde{D}(C), \partial C\bigr)$ with "marked" boundary $\partial C \subset \widetilde{D}(C)$.

We now suppose that C and C' are two cones at distance $\ge m$ from o, such that $\bigl(\widetilde{D}(C), \partial C\bigr)$ and $\bigl(\widetilde{D}(C'), \partial C'\bigr)$ are ismorphic. We claim that C and C' are isomorphic, and this will conclude the proof that there are only finitely many isomorphism types of cones with respect to o.

Let $\phi : \widetilde{D}(C) \to \widetilde{D}(C')$ be an isomorphism with $\phi(\partial C) = \partial C'$, and ϕ' its inverse mapping. We extend ϕ to a mapping from C to C', also denoted ϕ.

Claim 1. Let $x \in \partial C$ and $v \in \mathbf{\Sigma}^+$ such that the path $\pi_x(v)$ lies in C and meets ∂C only in its initial point x. Then the path $\pi_{x'}(v)$ lies in C' and meets $\partial C'$ only in its initial point $x' = \phi(x) \in \partial C'$.

Proof. If a is the initial letter of v then (always using the notation of Definition 2.1) the first edge of $\pi_x(v)$ is (x, a, x^a). We now consider the path $\pi_{x'}(v)$ with label v starting at $x' \in \partial C'$. We first claim that the latter lies in C' and only its initial point x' is in $\partial C'$. Let $(x', a, (x')^a)$ be the first edge of the path. Then $(x')^a$ cannot lie in $\widetilde{D}(C')$, since otherwise $(x, a, x^a) = \bigl(\phi'(x'), a, \phi'(x')^a\bigr)$ would be an edge in $\widetilde{D}(C)$, a contradiction. Thus, the path $\pi_{x'}(v)$ goes at least initally into $C' \setminus \partial C$.

So now suppose that $\pi_{x'}(v)$ ever returns to $\partial C'$, and let π' be its initial part up to the first return. Then $v' = \ell\bigl(\pi_{x'}(v)\bigr)$ is an initial part of v with $|v'| \ge 2$, and π' is a path within C' from $x_1' = x'$ to $x_2' = (x')^{v'} \in \partial C'$. But then, by construction, $\widetilde{D}(C')$ must contain an edge (y_1', v_1Tv_2, y_2') such that $x_1' = (y_1')^{v_1}$, $y_2' = (x_2')^{v_2}$, and $T \overset{*}{\Longrightarrow} v_1v'v_2$. Using the isomorphism $\phi' : \widetilde{D}(C') \to \widetilde{D}(C)$, we set $y_i = \phi'(y_i')$, $i = 1, 2$, and $x_2 = \phi'(x_2') \in \partial C$. We have of course $x_1 = \phi'(x_1')$. Now we must have the edge (y_1, v_1Tv_2, y_2) in $\widetilde{D}(C)$. But then $v_1v'v_2 \in L_{y_1,y_2}$, and consequently $v' \in L_{x_1,x_2}$, that is, $x_1^{v'} \in \partial C$. But this contradicts the fact that $\pi_x(v)$ meets ∂C only in its initial point. We conclude that also the path $\pi_{x'}(v)$ lies in C' and meets $\partial C'$ only in its inital point, and Claim 1 is verified.

Now let $z \in C \setminus \partial C$. Then there are $x \in \partial C$ and $v \in \mathbf{\Sigma}^+$ such that $z = x^v$ and the path $\pi_x(v)$ from x to z meets ∂C only in its initial point x. By Claim 1, the analogous statement holds for the path $\pi_{x'}(v)$ in C', where $x' = \phi(x)$. The only choice is to define $\phi(z) = z' = (x')^v$, which lies in $C' \setminus \partial C'$ as required. We have to show that ϕ is well-defined. This will follow from the next claim.

Claim 2. Let $x_1, x_2 \in \partial C$, $v, w \in \mathbf{\Sigma}^+$ such that the paths $\pi_{x_1}(v)$ and $\pi_{x_2}(w)$ lie in C, meet ∂C only in their initial points and end at the same point of $C \setminus \partial C$. Then, setting $x_i' = \phi(x_i)$, also $\pi_{x_1'}(v)$ and $\pi_{x_2'}(w)$ end at the same point of $C' \setminus \partial C'$.

Proof. Let w^{-1} be the "inverse" of w, as defined in Definition 2.1. Then $x_2^{-w^{-1}} = x_2^w$, and vw^{-1} is the label of the path from x_1 to x_2 that we obtain by first following $\pi_{x_1}(v)$ and then the "inverse" of $\pi_{x_2}(w)$. It lies entirely in C, and only its endpoints are in ∂C. By construction, $\widetilde{D}(C)$ has an edge (y_1, v_1Tv_2, y_2) such that $y_1^{v_1} = x_1$, $x_2^{v_2} = y_2$ and $T \overset{*}{\Longrightarrow} v_1vw^{-1}v_2$. We set $y_i' = \phi(y_i)$, $i = 1, 2$. Then (y_1', v_1Tv_2, y_2') is an edge of $\widetilde{D}(C')$.

Therefore $v_1 v w^{-1} v_2 \in L_{y_1',y_2'}$. But this implies that vw^{-1} is the label of a path from x_1' to x_2', and we know from Claim 1 that it lies in C and has only its endpoints in ∂C. Thus $(x_1')^v = (x_2')^{-w^{-1}} = (x_2')^w$, and Claim 2 is true.

Thus, ϕ is well defined, and the same works of course also for ϕ' by exchanging the roles of C and C'.

Claim 3. The map $\phi : C \to C'$ is bijective.

Proof. We know that $\phi : \partial C \to \partial C'$ is bijective and that $\phi(C \setminus \partial C) \subset C' \setminus \partial C$. Let $z \in C \setminus \partial C$, and let $x \in \partial C$, $v \in \mathbf{\Sigma}^+$ such that $\pi_x(v)$ is a path from x to z that intersects ∂C only at the initial point. Setting $x' = \phi(x)$, $z' = \phi(z)$, we know from the construction of ϕ and Claim 1 that $\pi_{x'}(v)$ is a path in C' from x' to z' that meets $\partial C'$ only in its initial point. Now the way how ϕ' is constructed yields that $\phi'(z') = z$. Therefore $\phi'\,\phi$ is the identity on C. Exchanging roles, we also get the $\phi\,\phi'$ is the identity on C'. This proves Claim 3.

It is now immediate from the construction that ϕ also preserves the edges and their labels, so that it is indeed an isomorphism between the labelled graphs C and C' that sends ∂C to $\partial C'$. $\square$

[11, Cor. 2.7] says that if a symmetric labelled graph is context-free with respect to one root o, then it is context-free with respect to any other vertex chosen as the root x. In view of Theorems 4.2 and 4.7, this is also obtained from the following, when the graph is fully deterministic.

(4.9) Corollary. *Let (X, E, ℓ) be a fully deterministic, strongly connected graph with label alphabet $\mathbf{\Sigma}$. If $L_{o,o}$ is context-free then $L_{x,y}$ is deterministic context-free for all $x, y \in X$.*

Theorems 4.2 and 4.7, together with Lemma 3.1 also imply the following.

(4.10) Corollary. *Let G be a finitely generated group and K a subgroup.*

(a) *The pair (G, K) is context-free if and only if for any symmetric $\psi : \mathbf{\Sigma} \to G$, the Schreier graph $X(G, K, \psi)$ is a context-free graph. In this case, the language $L(G, K, \psi)$ is deterministic for every (not necessarily symmetric) semigroup presentation $\psi : \mathbf{\Sigma} \to G$.*

(b) *If (G, K) is context-free, then also $(G, g^{-1}Kg)$ is context-free for every $g \in G$.*

Proof. (a) is clear. Regarding (b), for the Schreier graph $X(G, K, \psi)$, we have $L(G, K, \psi) = L_{o,o}$ and and $L(G, g^{-1}Kg, \psi) = L_{x,x}$ with $x = Kg$, $g \in G$. Thus, the statement follows from Corollary 4.9. $\square$

(4.11) Proposition. *Let G be a finitely generated group and K, H be subgroups with $K \le H$ and $[H : K] < \infty$.*

Then (G, K) is context-free if and only if (G, H) is context-free.

Proof. First, suppose that (G, K) is context-free. In the context-free graph $X(G, K)$, consider the finite set of vertices $F = \{Kh : h \in H\}$, containing the root vertex $o = o_K = K$. Then $L(G, H, \psi) = \bigcup_{x \in F} L_{o,x}$ is a finite (disjoint) union of context-free languages. Therefore it is context-free by standard facts.

Next, suppose that (G,H) is context-free. Since $[H:K]<\infty$, there is a subgroup N of K that has finite index and is normal in H. The finite group H/N acts on $X(G,N)$ by automorphisms of that labelled graph as follows: if $x=Ng$ and $[h]=hN\in H/N$ then $[h]x=hNg=N(hg)$. Every orbit of a vertex has $[H:N]$ elements. The finite set $F=\{Nh : h\in H\}$ is fixed (as a set, not elementwise) by H/N. In particular, for each $n\geq 0$, H/N permutes the cones which are components of $X(G,N)\setminus B(F,n)$. Therefore, each of those cones is isomorphic with a cone in the factor graph – which is just $X(G,H)$ – with respect to the root vertex $o_H=H$ of the latter. That root is the image of F under the natural graph homomorphism $X(G,H)\to X(G,N)$.

Since $X(G,H)$ is a context-free graph by Corollary 4.10(a), we infer that $X(G,N)$ is context-free with respect to F. But then the pair (G,N) is context-free, and the first part of the proof implies that also (G,K) is a context-free pair. □

5. Covers and Schreier graphs

We assume again that (X,E,ℓ) is symmetric and fully deterministic. Recall the involution $a\mapsto a^{-1}\neq a$ of $\mathbf{\Sigma}$. A word in $\mathbf{\Sigma}^*$ is called reduced if it contains no subword of the form aa^{-1}, where $a\in\mathbf{\Sigma}$. We write $\mathbb{T}_{\mathbf{\Sigma}}$ for the set of all reduced words in $\mathbf{\Sigma}^*$. We can equip $\mathbb{T}_{\mathbf{\Sigma}}$ with the structure of a labelled graph, whose edges are of the form

$$(5.1)\qquad (v,a,w)\quad\text{and}\quad (w,a^{-1},v)\,,\quad \text{where}\ v,w\in\mathbb{T}_{\mathbf{\Sigma}}\,,\ a\in\mathbf{\Sigma}\,,\ va=w\,.$$

Thus, the terminal letter of v must be different from a^{-1}. Then $\mathbb{T}_{\mathbf{\Sigma}}$ is fully deterministic, and it is a *tree,* that is, it has no closed path whose label is a (non-empty) reduced word. As the root of $\mathbb{T}_{\mathbf{\Sigma}}$, we choose the empty word ϵ. Then $\mathbb{T}_{\mathbf{\Sigma}}$ is the *universal cover* of X. Namely, if we choose (and fix) any vertex $o\in X$ as the root, then the mapping

$$(5.2)\qquad \Phi:\mathbb{T}_{\mathbf{\Sigma}}\to X\,,\quad \Phi(w)=o^w\,,$$

is a *covering map:* it is a surjective homomorphism between labelled graphs which is a local isomorphism, that is, it is one-to-one between the sets of outgoing (resp. ingoing) edges of any element $w\in\mathbb{T}_{\mathbf{\Sigma}}$ and its image $\Phi(w)$. (Note that this allows the image of an edge to be a loop.) "Universal" means that it covers every other cover of X, but this is not very important for us. The property of $w\in\mathbb{T}_{\mathbf{\Sigma}}$ to be reduced is equivalent with the fact that the path $\pi_o(w)$ in X is *non-backtracking,* that is, it does not contain two consecutive edges which are the reversal of each other.

We now realize that $\mathbb{T}_{\mathbf{\Sigma}}$ is the standard Cayley graph of the free group $\mathbb{F}_{\mathbf{\Sigma}}$, where $\mathbf{\Sigma}$ is the set of free generators together with their inverses. The group product is the following: if $v,w\in\mathbb{T}_{\mathbf{\Sigma}}\equiv\mathbb{F}_{\mathbf{\Sigma}}$, then $v\cdot w$ is obtained from the concatenated word vw by step after step deleting possible subwords of the form aa^{-1} that can arise from that concatenation. The group identity is ϵ, and the inverse of w is w^{-1} as at the end of Definition 2.1 With Φ as in (5.2), let

$$(5.3)\qquad \mathbb{K}=\mathbb{K}(X)=\Phi^{-1}(o)=\{w\in\mathbb{T}_{\mathbf{\Sigma}}:\pi_o(w)\ \text{is a closed path from}\ o\ \text{to}\ o\ \text{in}\ X\,\}\,.$$

Then, under the indentification $\mathbb{T}_{\mathbf{\Sigma}}\equiv\mathbb{F}_{\mathbf{\Sigma}}$, we clearly have that $\mathbb{K}$ is a subgroup of $\mathbb{F}_{\mathbf{\Sigma}}$. The following is known, see e.g. Lyndon and Schupp [9, Ch. III] or (our personal source) Imrich [7].

(5.4) Proposition. *The graph X is the Schreier graph of the pair of groups $\big(\mathbb{F}_{\boldsymbol{\Sigma}}, \mathbb{K}(X)\big)$ with respect to the semigroup presentation ψ given by $\psi(a) = a$, $a \in \boldsymbol{\Sigma}$.*

In $\psi(a) = a$, we interpret a simultaneously as a letter from the alphabet and as a generator of the free group.

Thus, in reality the study of context-free pairs of groups is the same as the study of fully deterministic, *symmetric* context-free graphs under a different viewpoint.

The same is not true without assuming symmetry. Indeed, given a semigroup presentation ψ of G, for every $a \in \Sigma$ there must be $w_a \in \Sigma^*$ such $\psi(w_a) = \psi(a)^{-1}$, the inverse in G. But then in the Schreier Graph $X(G, K, \psi)$, for any subgroup K of G, we have the following: if $(x, a, y) \in E$ then $y^{w_a} = x$, that is, there is the oriented path from y to x with label w_a. In a general fully deterministic graph this property does not necessarily hold, even if it has the additional property that for each $a \in \Sigma$, there is presicely one incoming edge with label a at every vertex. As an example, consider $X = \{x, y, z\}$, $\Sigma = \{a, b\}$ and labelled edges $(x, a, y), (x, b, y), (y, a, z), (y, b, x), (z, a, x), (z, b, z)$.

We return to the situation of Proposition 5.4. As a subgroup of the free group, the group $\mathbb{K}(X)$ is itself free. There is a method for finding a set of free generators. First recall the notion of a *spanning tree* of X. This is a tree T, which as subgraph of X is obtained by deleting edges (but no vertices) of X. Every connected (non-oriented) graph has a spanning tree, for locally finite graphs it can be constructed inductively. Now let T be a spanning tree of X, and consider all edges of X that are not edges of T. They must come in pairs (e, e^{-1}). For each pair, we choose one of the two partner edges, and we write E_0 for the chosen (oriented) edges. For each $e \in E_0$, we choose non-backtracking paths in T from o to e^- and from e^+ to o. Together with e (in the middle), they give rise to a non-backtracking path in X that starts and ends at o. Let $w(e)$ be the label on that path. Then the following holds [9], [7].

(5.5) Proposition. *As elements of $\mathbb{F}_{\boldsymbol{\Sigma}}$, the $w(e)$, $e \in E_0$, are free generators of $\mathbb{K}(X)$.*

(5.6) Corollary. *Let G be a virtually free group and K a finitely generated subgroup. Then (G, K) is context-free.*

Proof. Let $\mathbb{F} = \mathbb{F}_{\boldsymbol{\Sigma}}$ be a free subgroup of G with finite index. Then $\mathbb{K} = K \cap \mathbb{F}$ is a free subgroup of K with $[K : \mathbb{K}] < \infty$. Since K is finitely generated, also $\mathbb{K}$ is finitely generated. Since all sets of free generators of $\mathbb{K}$ must have the same cardinality, the set E_0 of edges of the Schreier graph X of $(\mathbb{F}, \mathbb{K})$ with respect to the standard labeling by $\boldsymbol{\Sigma}$ is finite. Thus, X is obtained by adding finitely many edges to a tree. If o is the root vertex of X and n is the largest distance between o and an endpoint of some edge in E_0, then every cone C of X whith $d(\partial C, o) > n$ is a rooted, labelled tree that is isomorphic with one of the cones of $\mathbb{T}_{\boldsymbol{\Sigma}}$. Thus, the Schreier graph, resp. $(\mathbb{F}, \mathbb{K})$ are context-free. It now follows from propositions 3.3 and 4.11.b that also (G, K) is context-free. □

We remark here that one can always reduce the study of context-free to free groups and their subgroups. Given (G, K), let $\mathbb{F}$ be a finitely generated free group that maps by a homomorphism onto G. Let $\mathbb{K}$ be the preimage of K under that homomorphism.

Then clearly (G, K) is context-free if and only $(\mathbb{F}, \mathbb{K})$ has this property. (This reduction, however, is not very instructive.)

Of course, there are context-free pairs beyond the situation of Corollary 5.6. For example, consider the free group $\mathbb{F} = \langle a, b \mid \rangle$ and the subgroup $\mathbb{K}$ with the infinite set of free generators $\{a^k b^l a b^{-l} a^{-k} : k, l \in \mathbb{Z}\,,\ l \neq 0\}$. The associated Schreier graph with respect to $\{a^{\pm 1}, b^{\pm 1}\}$ is the *comb lattice.* Its vertex set is the set of integer points in the plane. The edges labelled by a are along the x-axis, from $(k, 0)$ to $(k+1, 0)$, and there is a loop with label a at each point (k, l) with $l \neq 0$. The edges labelled by b are all the upward edges of the grid, that is, all edges from (k, l) to $(k, l+1)$, where $(k, l) \in \mathbf{Z}^2$. To these, we have to add the oppositley oriented edges whose labels are the respective inverses. The comb lattice is clearly a context-free graph (tree).

References

[1] Anisimov, A. V.: *Group languages,* Kibernetika **4** (1971) 18–24.

[2] Berstel, J., and Boasson, L. : *Context-free languages.* In: *Handbook of theoretical computer science*, Vol. B, 59–102, Elsevier, Amsterdam, 1990.

[3] Ceccherini-Silberstein, T., and Woess, W.: *Growth and ergodicity of context-free languages*, Trans. Amer. Math. Soc. **354** (2002) 4597–4625.

[4] Dunwoody, M. J.: *The accessibility of finitely presented groups*, Invent. Math. **81** (1985) 449–457.

[5] Harrison, M. A.: *Introduction to Formal Language Theory*, Addison-Wesley, Reading, MA, 1978.

[6] Holt, D., Rees, S., Röver, C. and Thomas, R.: *Groups with context-free co-word problem*, J. London Math. Soc. **71** (2005) 643–657.

[7] Imrich, W.: *Subgroup theorems and graphs.* In: *Combinatorial Mathematics V,* Lecture Notes in Mathematics **622**, 1 – 27, Springer, Berlin 1977.

[8] Lehnert, J., and Schweitzer, P.: *The co-word problem for the Higman-Thompson group is context-free*, Bull. Lond. Math. Soc. **39** (2007) 235–241.

[9] Lyndon, R. C., and Schupp, P. E.: *Combinatorial Group Theory.* Ergebnisse der Mathematik und ihrer Grenzgebiete **89**, Springer, Berlin, 1977.

[10] Muller, D. E., and Schupp, P. E.: *Groups, the theory of ends and context-free languages*, J. Comput. System Sc. **26** (1983) 295–310.

[11] Muller, D. E., and Schupp, P. E.: *The theory of ends, pushdown automata, and second-order logic*, Theoretical Computer Science **37** (1985) 51–75.

[12] Pélecq, L.: *Automorphism groups of context-free graphs*, Theoret. Comput. Sci. **165** (1996) 275–293.

[13] Sageev, M.: *Ends of group pairs and non-positively curved cube complexes*, Proc. London Math. Soc. **71** (1995) 585–617.

[14] Scott, P.: *Ends of pairs of groups*, J. Pure Appl. Algebra **11** (1977/78) 179–198.

[15] Sénizergues, G.: *Semi-groups acting on context-free graphs.* In: *Automata, languages and programming* (Paderborn, 1996), 206–218, Lecture Notes in Comput. Sci. **1099**, Springer, Berlin, 1996.

[16] Swarup, G. A.: *On the ends of pairs of groups*, J. Pure Appl. Algebra **87** (1993) 93–96.

[17] Woess, W.: *Context-free pairs of groups. II - cuts, tree sets, and random walks*, in preparation.

[18] Woess, W.: *Graphs and groups with tree-like properties*, J. Combin. Theory, Ser. B **47** (1989) 361–371.

Dipartimento di Ingegneria, Università del Sannio,
Corso Garibaldi 107, 82100 Benevento, Italy

E-mail address: tceccher@mat.uniroma1.it

Institut für Mathematische Strukturtheorie,
Technische Universität Graz, Steyrergasse 30, 8010 Graz, Austria

E-mail address: woess@TUGraz.at