# Knapsack problems for wreath products*

**Moses Ganardi[1], Daniel König[1], Markus Lohrey[1], and Georg Zetzsche[2]**

1 **Universität Siegen, Germany**
{ganardi,koenig,lohrey}@eti.uni-siegen.de

2 **LSV, CNRS & ENS Paris-Saclay, France**
zetzsche@lsv.fr

## Abstract

In recent years, knapsack problems for (in general non-commutative) groups have attracted attention. In this paper, the knapsack problem for wreath products is studied. It turns out that decidability of knapsack is not preserved under wreath product. On the other hand, the class of knapsack-semilinear groups, where solutions sets of knapsack equations are effectively semilinear, is closed under wreath product. As a consequence, we obtain the decidability of knapsack for free solvable groups. Finally, it is shown that for every non-trivial abelian group $G$, knapsack (as well as the related subset sum problem) for the wreath product $G \wr \mathbb{Z}$ is NP-complete.

**1998 ACM Subject Classification** F.2.2 Nonnumerical Algorithms and Problems

**Keywords and phrases** knapsack, wreath products, decision problems in group theory

**Digital Object Identifier** 10.4230/LIPIcs.STACS.2018.33

## 1 Introduction

In [15], Myasnikov, Nikolaev, and Ushakov began the investigation of classical discrete optimization problems, which are formulated over the integers, for arbitrary (possibly non-commutative) groups. The general goal of this line of research is to study to what extent results from the commutative setting can be transferred to the non-commutative setting. Among other problems, Myasnikov et al. introduced for a finitely generated group $G$ the *knapsack problem* and the *subset sum problem*. The input for the knapsack problem is a sequence of group elements $g_1, \ldots, g_k, g \in G$ (specified by finite words over the generators of $G$) and it is asked whether there exists a solution $(x_1, \ldots, x_k) \in \mathbb{N}^k$ of the equation $g_1^{x_1} \cdots g_k^{x_k} = g$. For the subset sum problem one restricts the solution to $\{0,1\}^k$. For the particular case $G = \mathbb{Z}$ (where the additive notation $x_1 \cdot g_1 + \cdots + x_k \cdot g_k = g$ is usually preferred) these problems are NP-complete (resp., $\mathsf{TC}^0$-complete) if the numbers $g_1, \ldots, g_k, g$ are encoded in binary representation [7, 6] (resp., unary notation [2]).

Another motivation is that decidability of knapsack for a group $G$ implies that the membership problem for every fixed polycyclic subgroup of $G$ is decidable. This follows from the well-known fact that every polycyclic group $A$ has a generating set $\{a_1, \ldots, a_k\}$ such that every element of $A$ can be written as $a_1^{n_1} \cdots a_k^{n_k}$ for $n_1, \ldots, n_k \in \mathbb{N}$, see e.g. [17, Chapter 9].

In [15], Myasnikov et al. encode elements of the finitely generated group $G$ by words over the group generators and their inverses, which corresponds to the unary encoding of integers. There is also an encoding of words that corresponds to the binary encoding of integers, so

[*] The fourth author is supported by a fellowship within the Postdoc-Program of the German Academic Exchange Service (DAAD) and by Labex DigiCosme, Univ. Paris-Saclay, project VERICONISS.

35th Symposium on Theoretical Aspects of Computer Science (STACS 2018).
Editors: Rolf Niedermeier and Brigitte Vallée; Article No. 33; pp. 33:1–33:13
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

called straight-line programs, and knapsack problems under this encodings have been studied in [11]. In this paper, we only consider the case where input words are explicitly represented. Here is a (non-complete) list of known results concerning knapsack and subset sum problems:

- Subset sum and knapsack can be solved in polynomial time for every hyperbolic group [15]. In [3] this result was extended to free products of any number of hyperbolic groups and finitely generated abelian groups.
- For every virtually nilpotent group, subset sum belongs to nondeterministic logspace [8]. On the other hand, there are nilpotent groups of class 2 for which knapsack is undecidable. Examples are direct products of sufficiently many copies of the discrete Heisenberg group $H_3(\mathbb{Z})$ [8], and free nilpotent groups of class 2 and sufficiently high rank [14].
- Knapsack for $H_3(\mathbb{Z})$ is decidable [8]. In particular, together with the previous point it follows that decidability of knapsack is not preserved under direct products.
- For the following groups, subset sum is $\mathsf{NP}$-complete (whereas the word problem can be solved in polynomial time): free metabelian non-abelian groups of finite rank, the wreath product $\mathbb{Z} \wr \mathbb{Z}$, Thompson's group $F$, the Baumslag-Solitar group $\mathrm{BS}(1,2)$ [15], and every polycyclic group that is not virtually nilpotent [16].
- Knapsack is decidable for every co-context-free group [8].
- Knapsack belongs to $\mathsf{NP}$ for all virtually special groups (finite extensions of subgroups of graph groups) [11]. For graph groups (also known as right-angled Artin groups) a complete classification of the complexity of knapsack was obtained in [12]: If the underlying graph contains an induced path or cycle on 4 nodes, then knapsack is $\mathsf{NP}$-complete; in all other cases knapsack can be solved in polynomial time (even in $\mathsf{LogCFL}$).
- Decidability of knapsack is preserved under finite extensions, HNN-extensions over finite associated subgroups and amalgamated free products over finite subgroups [11].

In this paper, we study the knapsack problem for wreath products. The wreath product is a fundamental construction in group theory and semigroup theory, see Section 4 for the definition. An important application of wreath products in group theory is the Magnus embedding theorem [18], which allows to embed the quotient group $F_k/[N,N]$ into the wreath product $\mathbb{Z}^k \wr (F_k/N)$, where $F_k$ is a free group of rank $k$ and $N$ is a normal subgroup of $F_k$. In particular, free solvable groups can be embedded into iterated wreath products of free abelian groups; a fact that we will use in this paper. Wreath products also have some nice algorithmic properties: The word problem for a wreath product $G \wr H$ is $\mathsf{AC}^0$-reducible to the word problems for the factors $G$ and $H$, and the conjugacy problem for $G \wr H$ is $\mathsf{TC}^0$-reducible to the conjugacy problems for $G$ and $H$ and the so called power problem for $H$ [13].

As in the case of direct products, it turns out that decidability of knapsack is not preserved under wreath products: For this we consider direct products of the form $H_3(\mathbb{Z}) \times \mathbb{Z}^\ell$, where $H_3(\mathbb{Z})$ is the discrete 3-dimensional Heisenberg group. It was shown in [8] that for every $\ell \geq 0$, knapsack is decidable for $H_3(\mathbb{Z}) \times \mathbb{Z}^\ell$. We prove that for every non-trivial group $G$ and every sufficiently large $\ell$, knapsack for $G \wr (H_3(\mathbb{Z}) \times \mathbb{Z}^\ell)$ is undecidable.

By the above discussion, we need stronger assumptions on $G$ and $H$ to obtain decidability of knapsack for $G \wr H$. We exhibit a very weak condition on $G$ and $H$, knapsack-semilinearity, which is sufficient for decidability of knapsack for $G \wr H$. A finitely generated group $G$ is knapsack-semilinear if for every knapsack equation, the set of all solutions (a solution can be seen as an vector of natural numbers) is effectively semilinear.

Clearly, for every knapsack-semilinear group, the knapsack problem is decidable. While the converse is not true, the class of knapsack-semilinear groups is extraordinarily wide. The simplest examples are finitely generated abelian groups, but it also includes the rich class of virtually special groups [11], all hyperbolic groups [4], and all co-context-free groups [8].

Furthermore, it is known to be closed under direct products (an easy observation), going to a finitely generated subgroup, going to a finite extension, HNN-extensions over finite associated subgroups and amalgamated free products over finite subgroups (the last three closure properties are simple extensions of the transfer theorems in [11]). In fact, the only non-knapsack-semilinear groups with a decidable knapsack problem that we are aware of are the groups $H_3(\mathbb{Z}) \times \mathbb{Z}^n$ for $n \geq 0$.

We prove in Section 6 that the class of knapsack-semilinear groups is closed under wreath products. As a direct consequence of the Magnus embedding, it follows that knapsack is decidable for every free solvable group. Recall that, in contrast, knapsack for free nilpotent groups is in general undecidable [14].

Finally, we consider the complexity of knapsack for wreath products. We prove that for every non-trivial finitely generated abelian group $G$, knapsack for $G \wr \mathbb{Z}$ is $\mathsf{NP}$-complete (the hard part is membership in $\mathsf{NP}$). This result includes important special cases like for instance the lamplighter group $\mathbb{Z}_2 \wr \mathbb{Z}$ and $\mathbb{Z} \wr \mathbb{Z}$. Wreath products of the form $G \wr \mathbb{Z}$ with $G$ abelian turn out to be important in connection with subgroup distortion [1]. Our proof also shows that for every non-trivial finitely generated abelian group $G$, the subset sum problem for $G \wr \mathbb{Z}$ is $\mathsf{NP}$-complete. In [15] this result is only shown for infinite abelian groups $G$.

Missing proofs can be found in the full version [4].

## 2 Preliminaries

We assume standard notions concerning groups. A group $G$ is *finitely generated* if there exists a finite subset $\Sigma \subseteq G$ such that every element $g \in G$ can be written as $g = a_1 a_2 \cdots a_n$ with $a_1, a_2, \ldots, a_n \in \Sigma$. We also say that the word $a_1 a_2 \cdots a_n \in \Sigma^*$ evaluates to $g$ (or represents $g$). The set $\Sigma$ is called a finite generating set of $G$. We always assume that $\Sigma$ is symmetric in the sense that $a \in \Sigma$ implies $a^{-1} \in \Sigma$. Elements of $G$ will be represented by words from $\Sigma^*$. An element $g \in G$ is called *torsion element* if there is an $n \geq 1$ with $g^n = 1$. The smallest such $n$ is the *order* of $g$ and is denoted by $\mathrm{ord}(g)$. If $g$ is not a torsion element, we set $\mathrm{ord}(g) = \infty$.

A set $A \subseteq \mathbb{N}^k$ is *linear* if $A = \{v_0 + \lambda_1 \cdot v_1 + \cdots + \lambda_n \cdot v_n \mid \lambda_1, \ldots, \lambda_n \in \mathbb{N}\}$ for vectors $v_0, \ldots, v_n \in \mathbb{N}^k$. The tuple of vectors $(v_0, \ldots, v_n)$ is a *linear representation* of $A$. A set $A \subseteq \mathbb{N}^k$ is *semilinear* if it is a finite union of linear sets $A_1, \ldots, A_m$. A *semilinear representation* of $A$ is a list of linear representations for the linear sets $A_1, \ldots, A_m$. It is well-known that the semilinear subsets of $\mathbb{N}^k$ are exactly the sets definable in *Presburger arithmetic*. These are those sets that can be defined with a first-order formula $\varphi(x_1, \ldots, x_k)$ over the structure $(\mathbb{N}, 0, +, \leq)$ [5]. Moreover, the transformations between such a first-order formula and an equivalent semilinear representation are effective. In particular, the semilinear sets are effectively closed under Boolean operations.

## 3 Knapsack for groups

Let $G$ be a finitely generated group with the finite symmetric generating set $\Sigma$. Moreover, let $V$ be a set of formal variables that take values from $\mathbb{N}$. For a subset $U \subseteq V$, we use $\mathbb{N}^U$ to denote the set of maps $\nu \colon U \to \mathbb{N}$, which we call *valuations*. An *exponent expression* over $G$ is a formal expression of the form $E = v_0 u_1^{x_1} v_1 u_2^{x_2} v_2 \cdots u_k^{x_k} v_k$ with $k \geq 0$ and words $u_i, v_i \in \Sigma^*$. Here, the variables do not have to be pairwise distinct. If every variable in an exponent expression occurs at most once, it is called a *knapsack expression*. Let $V_E = \{x_1, \ldots, x_k\}$ be the set of variables that occur in $E$. For a valuation $\nu \in \mathbb{N}^U$ such that $V_E \subseteq U$ (in which case we also say that $\nu$ is a valuation for $E$), we define $\nu(E) = v_0 u_1^{\nu(x_1)} v_1 u_2^{\nu(x_2)} v_2 \cdots u_k^{\nu(x_k)} v_k \in \Sigma^*$. We say that $\nu$ is a *solution* of the equation $E = 1$ if $\nu(E)$ evaluates to the identity element 1 of $G$. With $\mathsf{Sol}(E)$ we denote the set of all solutions $\nu \in \mathbb{N}^{V_E}$ of $E$. We can view $\mathsf{Sol}(E)$ as a subset of $\mathbb{N}^k$. The *length* of $E$ is defined as $|E| = |v_0| + \sum_{i=1}^{k} |u_i| + |v_i|$, whereas $k$ is its *depth*. If the length of a knapsack expression is not needed, we will write an exponent expression over $G$ also as $E = h_0 g_1^{x_1} h_1 g_2^{x_2} h_2 \cdots g_k^{x_k} h_k$ where $g_i, h_i \in G$. We define *solvability of exponent equations over* $G$, $\textsc{ExpEq}(G)$ for short, as the following decision problem:

**Input:** A finite list of exponent expressions $E_1, \dots, E_n$ over $G$.

**Question:** Is $\bigcap_{i=1}^{n} \mathsf{Sol}(E_i)$ non-empty?

The knapsack problem for $G$, $\mathrm{KP}(G)$ for short, is the following decision problem:

**Input:** A single knapsack expression $E$ over $G$.

**Question:** Is $\mathsf{Sol}(E)$ non-empty?

We also consider the uniform knapsack problem for powers $G^m = \prod_{i=1}^{m} G_i$ with $G_i \cong G$. We denote this problem with $\mathrm{KP}(G^*)$. Formally, it is defined as follows:

**Input:** A number $m \geq 0$ (in unary notation) and a knapsack expression $E$ over $G^m$.

**Question:** Is $\mathsf{Sol}(E)$ non-empty?

It turns out that the problems $\mathrm{KP}(G^*)$ and $\textsc{ExpEq}(G)$ are inter-reducible:

▶ **Proposition 3.1.** *$\mathrm{KP}(G^*)$ is decidable if and only if $\textsc{ExpEq}(G)$ is decidable.*

Note that the exponent equation $v_0 u_1^{x_1} v_1 u_2^{x_2} v_2 \cdots u_k^{x_k} v_k = 1$ is equivalent to the exponent equation $(v_0 u_1 v_0^{-1})^{x_1} (v_0 v_1 u_2 v_1^{-1} v_0^{-1})^{x_2} \cdots (v_0 \cdots v_{k-1} u_k v_{k-1}^{-1} \cdots v_0^{-1})^{x_k} (v_0 \cdots v_k) = 1$. Hence, it suffices to consider exponent expressions of the form $u_1^{x_1} u_2^{x_2} \cdots u_k^{x_k} v$.

The group $G$ is called *knapsack-semilinear* if for every knapsack expression $E$ over $G$, the set $\mathsf{Sol}(E)$ is a semilinear set of vectors and a semilinear representation can be effectively computed from $E$. The following classes of groups only contain knapsack-semilinear groups:

- virtually special groups [11]: these are finite extensions of subgroups of graph groups (aka right-angled Artin groups). The class of virtually special groups is very rich. It contains all Coxeter groups, one-relator groups with torsion, fully residually free groups, and fundamental groups of hyperbolic 3-manifolds.
- hyperbolic groups [4]
- co-context-free groups [8], i.e., groups where the set of all words over the generators that do not represent the identity is a context-free language. Lehnert and Schweitzer [9] have shown that the Higman-Thompson groups are co-context-free.

Since emptiness of the intersection of finitely many semilinear sets is decidable, we have:

▶ **Lemma 3.2.** *If $G$ is knapsack-semilinear, then $\mathrm{KP}(G^*)$ and $\textsc{ExpEq}(G)$ are decidable.*

An example of a group $G$, where $\mathrm{KP}(G)$ is decidable, but $\mathrm{KP}(G^*)$ and $\textsc{ExpEq}(G)$ are undecidable is the Heisenberg group $H_3(\mathbb{Z})$ (the group of all upper triangular $(3 \times 3)$-matrices over $\mathbb{Z}$ with all diagonal entries equal to 1) [8]. Hence, $H_3(\mathbb{Z})$ is not knapsack-semilinear.

# 4 Wreath products

Let $G$ and $H$ be groups. Consider the direct sum $K = \bigoplus_{h \in H} G_h$, where $G_h$ is a copy of $G$. We view $K$ as the set $G^{(H)}$ of all mappings $f \colon H \to G$ such that $\mathsf{supp}(f) = \{h \in H \mid f(h) \neq 1\}$ is finite, together with pointwise multiplication as the group operation. The set $\mathsf{supp}(f) \subseteq H$ is called the *support* of $f$. The group $H$ has a natural left action on $G^{(H)}$ given by $hf(a) = f(h^{-1}a)$, where $f \in G^{(H)}$ and $h, a \in H$. The corresponding semidirect product $G^{(H)} \rtimes H$ is the *wreath product* $G \wr H$. In other words:

- Elements of $G \wr H$ are pairs $(f, h)$, where $h \in H$ and $f \in G^{(H)}$.
- The multiplication in $G \wr H$ is defined as follows: Let $(f_1, h_1), (f_2, h_2) \in G \wr H$. Then $(f_1, h_1)(f_2, h_2) = (f, h_1h_2)$, where $f(a) = f_1(a)f_2(h_1^{-1}a)$.

The following intuition might be helpful: An element $(f, h) \in G \wr H$ can be thought of as a finite multiset of elements of $G \setminus \{1_G\}$ that are sitting at certain elements of $H$ (the map $f$) together with the distinguished element $h \in H$, which can be thought of as a cursor moving in $H$. The product $(f_1, h_1)(f_2, h_2)$ is computed as follows. First, we shift the finite collection of $G$-elements (that corresponds to the mapping $f_2$) by $h_1$: If $g \in G \setminus \{1_G\}$ is sitting at $a \in H$ (i.e., $f_2(a) = g$), then we remove $g$ from $a$ and put it to the new location $h_1a \in H$. This new collection corresponds to the mapping $f_2' \colon a \mapsto f_2(h_1^{-1}a)$. After this shift, the two collections of $G$-elements are multiplied pointwise: If in $a \in H$ the elements $g_1$ and $g_2$ are sitting (i.e., $f_1(a) = g_1$ and $f_2'(a) = g_2$), then we put the product $g_1g_2$ into the location $a$. Finally, the new distinguished $H$-element (the new cursor position) becomes $h_1h_2$.

By identifying $f \in G^{(H)}$ with $(f, 1_H) \in G \wr H$ and $h \in H$ with $(1_{G^{(H)}}, h)$, we regard $G^{(H)}$ and $H$ as subgroups of $G \wr H$. Hence, for $f \in G^{(H)}$ and $h \in H$, we have $fh = (f, 1_H)(1_{G^{(H)}}, h) = (f, h)$. There are two natural projection maps $\sigma_{G\wr H} \colon G \wr H \to H$ (which is a morphism) and $\tau_{G\wr H} \colon G \wr H \to G^{(H)}$ with $\sigma_{G\wr H}(f, h) = h$ and $\tau_{G\wr H}(f, h) = f$ . If $G$ (resp. $H$) is generated by the set $\Sigma$ (resp. $\Gamma$) with $\Sigma \cap \Gamma = \emptyset$, then $G \wr H$ is generated by the set $\{(f_a, 1_H) \mid a \in \Sigma\} \cup \{(f_{1_G}, b) \mid b \in \Gamma\}$, where for $g \in G$, the mapping $f_g : H \to G$ is defined by $f_g(1_H) = g$ and $f_g(x) = 1_G$ for $x \in H \setminus \{1_H\}$. We identify this generating set with $\Sigma \uplus \Gamma$.

## 5 Main results

In this section, we state the main results of the paper. We begin with a general necessary condition for knapsack to be decidable for a wreath product. Note that if $H$ is finite, then $G \wr H$ is a finite extension of $G^{|H|}$ [10, Proposition 1], meaning that $\mathrm{KP}(G \wr H)$ is decidable if and only if $\mathrm{KP}(G^{|H|})$ is decidable [11, Theorem 11] (in [11], preservation of NP-membership was shown, but the proof also yields preservation of decidability). Therefore, we are only interested in the case that $H$ is infinite.

▶ **Proposition 5.1.** *Suppose $H$ is infinite. If $\mathrm{KP}(G \wr H)$ is decidable, then $\mathrm{KP}(H)$ and $\mathrm{KP}(G^*)$ are decidable.*

Of course, as a subgroup of $G \wr H$, $H$ inherits decidability of knapsack. On the other hand, given $m \in \mathbb{N}$, one can easily compute an embedding of $G^m$ into $G \wr H$ and thus solve knapsack instances over $G^m$ uniformly in $m$. Proposition 5.1 shows that $\mathrm{KP}(H_3(\mathbb{Z}) \wr \mathbb{Z})$ is undecidable: It was shown in [8] that $\mathrm{KP}(H_3(\mathbb{Z}))$ is decidable, whereas for some $m > 1$, the problem $\mathrm{KP}(H_3(\mathbb{Z})^m)$ is undecidable.

Proposition 5.1 raises the question whether decidability of $\mathrm{KP}(H)$ and $\mathrm{KP}(G^*)$ implies decidability of $\mathrm{KP}(G \wr H)$. We disprove this in the following theorem. The second statement is due to the fact that for every $\ell \in \mathbb{N}$, $\mathrm{KP}(H_3(\mathbb{Z}) \times \mathbb{Z}^\ell)$ is decidable, as shown in [8].

▶ **Theorem 5.2.** *There is an $\ell \in \mathbb{N}$ such that for every $G \neq 1$, $\mathrm{KP}(G \wr (H_3(\mathbb{Z}) \times \mathbb{Z}^\ell))$ is undecidable. In particular, there are groups $G$, $H$ such that $\mathrm{KP}(G^*)$ and $\mathrm{KP}(H)$ are decidable and $\mathrm{KP}(G \wr H)$ is undecidable.*

We therefore need to strengthen the assumptions on $H$ in order to show decidability of $\mathrm{KP}(G \wr H)$. Under the weak additional assumption of knapsack-semilinearity for $H$, we obtain a partial converse to Proposition 5.1. In Section 6 we prove:

▶ **Theorem 5.3.** *Let $H$ be knapsack-semilinear and infinite. Then* $\mathrm{KP}(G \wr H)$ *is decidable if and only if* $\mathrm{KP}(G^*)$ *is decidable.*

If $G$ is knapsack-semilinear, the solution sets are effectively semilinear:

▶ **Theorem 5.4.** *The group $G \wr H$ is knapsack-semilinear if and only if both $G$ and $H$ are knapsack-semilinear.*

Since every free abelian group is clearly knapsack-semilinear, it follows that the iterated wreath products $G_{1,r} = \mathbb{Z}^r$ and $G_{d+1,r} = \mathbb{Z}^r \wr G_{d,r}$ are knapsack-semilinear. By the well-known Magnus embedding, the free solvable group $S_{d,r}$ embeds into $G_{d,r}$. Hence, we get:

▶ **Corollary 5.5.** *Every free solvable group is knapsack-semilinear. Hence, solvability of exponent equations is decidable for free solvable groups.*

Finally, we consider the complexity of knapsack for wreath products. In Section 7 we prove NP-completeness for an important special case:

▶ **Theorem 5.6.** *For every finitely generated abelian group $G \neq 1$,* $\mathrm{KP}(G \wr \mathbb{Z})$ *is* NP*-complete.*

## 6 (Un)decidability: Proofs of Theorems 5.2, 5.3, and 5.4

**Undecidability.** We begin with a proof sketch for Theorem 5.2. Here, the only property of $H_3(\mathbb{Z})$ that we use is that solvability of knapsack instances of some fixed depth $k$ over the group $H_3(\mathbb{Z})^m$ is undecidable for some $m \geq 0$, which was shown in [8]. Using this property, we prove that $\mathrm{KP}(G^m \wr (H_3(\mathbb{Z}) \times \mathbb{Z}^{k \cdot m}))$ is undecidable. Since every group $G^n \wr H$ embeds into $G \wr (H \times \mathbb{Z})$, this implies undecidability of $\mathrm{KP}(G \wr (H_3(\mathbb{Z}) \times \mathbb{Z}^{k \cdot m+1}))$.

Undecidability of $\mathrm{KP}(G^m \wr (H_3(\mathbb{Z}) \times \mathbb{Z}^{k \cdot m}))$ is shown as follows. Consider a knapsack expression $E$ over $H_3(\mathbb{Z})^m$ of depth $k$. We turn $E$ into $m$ knapsack expressions $E_1, \ldots, E_m$ over $H_3(\mathbb{Z})$ of depth $k$ so that $E$ has a solution if and only if there is a common solution to $E_1, \ldots, E_m$ so that for every $i \in [1, k]$, the $i$-th variable for each expression has the same value. We construct a knapsack expression over $G^m \wr (H_3(\mathbb{Z}) \times \mathbb{Z}^{k \cdot m})$ as follows. We pick an $a \in G \setminus \{1\}$ and use it as a "breadcrumb": It is placed at a particular cursor position in $H_3(\mathbb{Z}) \times \mathbb{Z}^{k \cdot m}$ in one of the $m$ coordinates of $G^m$ and is later collected by multiplying $a^{-1}$ in the same coordinate of $G^m$. Fix a correspondence between the $k \cdot m$ variables in $E_1, \ldots, E_m$ and the coordinates of $\mathbb{Z}^{k \cdot m}$. Our new expression operates in three phases. The first phase performs for each $j = 1, \ldots, m$ the following. It places a breadcrumb in the $j$-th coordinate of $G^m$ and then moves the cursor by some value of $E_j$. At the same time, for each variable in $E_j$, it moves the cursor in the corresponding coordinate of $\mathbb{Z}^{k \cdot m}$ by the value of that variable.

In the second phase, we check that for each $i \in [1, k]$, the $i$-th variable for each expression has the same value and move the cursor back to the origin. To this end, we move the cursor in the $\mathbb{Z}^{k \cdot m}$-coordinates so that two coordinates that correspond to two variables that are to be compared are decremented simultaneously. After this, we collect the first breadcrumb. In the third phase, it remains to check that each $E_j$ evaluates to $1 \in H_3(\mathbb{Z})$. Since we are already in the origin, this amounts to checking that we can collect the remaining breadcrumbs $2, \ldots, m$ by moving just in the $\mathbb{Z}^{k \cdot m}$-coordinates. The full proof can be found in [4].

**Decidability.** The rest of this section is devoted to the positive results, Theorems 5.3 and 5.4. Let us fix a wreath product $G \wr H$. Recall the projections $\sigma = \sigma_{G \wr H} \colon G \wr H \to H$ and $\tau = \tau_{G \wr H} \colon G \wr H \to G^{(H)}$ from section 4. For $g \in G \wr H$ we write $\mathsf{supp}(g)$ for $\mathsf{supp}(\tau(g))$.

A knapsack expression $E = h_0 g_1^{x_1} h_1 \cdots g_k^{x_k} h_k$ over $G \wr H$ is called *torsion-free* if for each $i \in [1,k]$, either $\sigma(g_i) = 1$ or $\sigma(g_i)$ has infinite order. A simple conjugation argument shows that it suffices to prove Theorem 5.3 and 5.4 for torsion-free knapsack expressions. For the rest of this section let us fix a torsion-free knapsack expression $E$ over $G \wr H$. We can assume that $E = g_1^{x_1} g_2^{x_2} \cdots g_k^{x_k} g_{k+1}$ (note that if $g$ has infinite order than also $c^{-1}gc$ has infinite order). We partition the set $V_E = \{x_1, \ldots, x_k\}$ of variables in $E$ as $V_E = S \uplus M$, where $S = \{x_i \in V_E \mid \sigma(g_i) = 1\}$ and $M = \{x_i \in V_E \mid \mathrm{ord}(\sigma(g_i)) = \infty\}$. In this situation, the following notation will be useful. If $U = A \uplus B$ for a set of variables $U \subseteq V$ and $\mu \in \mathbb{N}^A$ and $\kappa \in \mathbb{N}^B$, then we write $\mu \oplus \kappa \in \mathbb{N}^U$ for the valuation with $(\mu \oplus \kappa)(x) = \mu(x)$ for $x \in A$ and $(\mu \oplus \kappa)(x) = \kappa(x)$ for $x \in B$.

**Computing powers.** A key observation in our proof is that in order to compute the group element $\tau(g^m)(h)$ (in the cursor intuition, this is the element labeling the point $h \in H$ in the wreath product element $g^m$) for $h \in H$ and $g \in G \wr H$, where $\sigma(g)$ has infinite order, one only has to perform at most $|\mathsf{supp}(g)|$ many multiplications in $G$, yielding a bound independent of $m$. We begin by introducing a partial order on $H$. Suppose $h \in H$ has infinite order (i.e. $\mathrm{ord}(h) = \infty$). For $h', h'' \in H$, we write $h' \preccurlyeq_h h''$ if there is an $n \geq 0$ with $h' = h^n h''$. Then, $\preccurlyeq_h$ is transitive. Moreover, since $h$ has infinite order, $\preccurlyeq_h$ is also anti-symmetric and thus a partial order. Observe that if knapsack is decidable for $H$, given $h, h', h'' \in H$, we can decide whether $h$ has infinite order and whether $h' \preccurlyeq_h h''$. This notion is used because for $g \in G \wr H$, the order $\preccurlyeq_{\sigma(g)}$ determines how to evaluate the mapping $\tau(g^m)$ at a certain element of $H$. We will sometimes want to multiply all elements $a_i$ for $i \in I$ such that the order in which we multiply is specified by some linear order on $I$. If $(I, \leq)$ is a finite linearly ordered set with $I = \{i_1, \ldots, i_n\}$, $i_1 < i_2 < \ldots < i_n$, then we write $\prod_{i \in I}^{\leq} a_i$ for $\prod_{j=1}^{n} a_{i_j}$. If the order $\leq$ is clear from the context, we just write $\prod_{i \in I} a_i$.

**Addresses.** A central concept in our proof is that of an address. A solution to the equation $E = 1$ can be thought of as a sequence of instructions on how to walk through the Cayley graph of $H$ and place elements of $G$ at those nodes. Here, being a solution means that in the end, all the nodes contain the identity of $G$. In order to express that every node carries 1 in the end, we want to talk about at which points in the product $E = g_1^{x_1} g_2^{x_2} \cdots g_k^{x_k} g_{k+1}$ a particular node is visited. An address is a datum that contains just enough information about such a point to determine which element of $G$ has been placed during that visit.

A pair $(i, h)$ with $i \in [1, k+1]$, and $h \in H$ is called an *address* if $h \in \mathsf{supp}(g_i)$. The set of addresses of the expression $E$ is denoted by $A$. Note that $A$ is finite and computable. To each address $(i, h)$, we associate the group element $\gamma(i, h) = g_i$ of the expression $E$.

**A linear order on addresses.** We will see that if a node is visited more than once, then (i) each time[1] it does so at a different address and (ii) the order of these visits only depends on the addresses. To capture the order of these visits, we define a linear order on addresses.

We partition $A = \bigcup_{i \in [1,k+1]} A_i$, where $A_i = \{(i, h) \mid h \in \mathsf{supp}(g_i)\}$ for $i \in [1, k+1]$. Then, for $a \in A_i$ and $a' \in A_j$ with $i < j$, we let $a < a'$. It remains to order addresses within each $A_i$. Within $A_{k+1}$, we pick an arbitrary order. If $i \in [1, k]$ and $\sigma(g_i) = 1$, we also order $A_i$ arbitrarily. Finally, if $i \in [1, k]$ and $\sigma(g_i)$ has infinite order, then we pick a linear order $\leq$ on $A_i$ so that for $h, h' \in \mathsf{supp}(g_i)$, $h \preccurlyeq_{\sigma(g_i)} h'$ implies $(i, h) \leq (i, h')$. Note that this is possible since $\preccurlyeq_{\sigma(g_i)}$ is a partial order on $H$.

[1] Here, we count two visits inside the same factor $g_i$, $i \in [1, k]$, with $\sigma(g_i) = 1$ as one visit.

**Cancelling profiles.** In order to express that a solution for $E$ yields the identity at every node of the Cayley graph of $H$, we need to compute the element of $G$ that is placed after the various visits at a particular node. We therefore associate to each address an expression over $G$ that yields the element placed during a visit at this address $a \in A$. In analogy to $\tau(g)$ for $g \in G \wr H$, we denote this expression by $\tau(a)$. If $a = (k+1, h)$, then we set $\tau(a) = \tau(g_{k+1})(h)$. Now, let $a = (i, h)$ for $i \in [1, k]$. If $\sigma(g_i) = 1$, then $\tau(a) = \tau(g_i)(h)^{x_i}$. Finally, if $\sigma(g_i)$ has infinite order, then $\tau(a) = \tau(g_i)(h)$.

This allows us to express the element of $G$ that is placed at a node $h \in H$ if $h$ has been visited with a particular set of addresses. To each subset $C \subseteq A$, we assign the expression $E_C = \prod_{a \in C} \tau(a)$, where the order of multiplication is given by the linear order on $A$. Observe that only variables in $S \subseteq \{x_1, \ldots, x_k\}$ occur in $E_C$. Therefore, given $\kappa \in \mathbb{N}^S$, we can evaluate $\kappa(E_C) \in G$. We say that $C \subseteq A$ is *$\kappa$-cancelling* if $\kappa(E_C) = 1$.

In order to record which sets of addresses can cancel simultaneously (meaning: for the same valuation), we use profiles. A *profile* is a subset of $\mathcal{P}(A)$ (the power set of $A$). A profile $P \subseteq \mathcal{P}(A)$ is said to be *$\kappa$-cancelling* if every $C \in P$ is $\kappa$-cancelling. A profile is *cancelling* if it is $\kappa$-cancelling for some $\kappa \in \mathbb{N}^S$.

**Clusters.** We also need to express that there is a node $h \in H$ that is visited with a particular set of addresses. To this end, we associate to each address $a \in A$ another expression $\sigma(a)$. As opposed to $\tau(a)$, the expression $\sigma(a)$ is over $H$ and variables $M' = M \cup \{y_i \mid x_i \in M\}$. Let $a = (i, h) \in A$. When we define $\sigma(a)$, we will also include factors $\sigma(g_j)^{x_j}$ and $\sigma(g_j)^{y_j}$ where $\sigma(g_j) = 1$. However, since these factors do not affect the evaluation of the expression, this should be interpreted as leaving out such factors.

1. If $i = k+1$ then $\sigma(a) = \sigma(g_1)^{x_1} \cdots \sigma(g_k)^{x_k} h$.
2. If $i \in [1, k]$ then $\sigma(a) = \sigma(g_1)^{x_1} \cdots \sigma(g_{i-1})^{x_{i-1}} \sigma(g_i)^{y_i} h$.

We now want to express that when multiplying $g_1^{\nu(x_1)} \cdots g_k^{\nu(x_k)} g_{k+1}$, there is a node $h \in H$ such that the set of addresses with which one visits $h$ is precisely $C \subseteq A$. In this case, we will call $C$ a cluster. Let $\mu \in \mathbb{N}^M$ and $\mu' \in \mathbb{N}^{M'}$. We write $\mu' \sqsubset \mu$ if $\mu'(x_i) = \mu(x_i)$ for $x_i \in M$ and $\mu'(y_i) \in [0, \mu(x_i) - 1]$ for every $y_i \in M'$. We can now define the set of addresses at which one visits $h \in H$: For $h \in H$, let $A_{\mu,h} = \{a \in A \mid \mu'(\sigma(a)) = h \text{ for some } \mu' \in \mathbb{N}^{M'} \text{ with } \mu' \sqsubset \mu\}$. A subset $C \subseteq A$ is called a *$\mu$-cluster* if $C \neq \emptyset$ and there is an $h \in H$ such that $C = A_{\mu,h}$. It can now be shown that if $\nu = \mu \oplus \kappa$ for $\kappa \in \mathbb{N}^S$ and $\mu \in \mathbb{N}^M$, evaluating $\tau(\nu(E))$ at a node $h \in H$ amounts to evaluating $\kappa$ on the expression $E_C$ where $C$ is the $\mu$-cluster $A_{\mu,h}$. In other words, we have $\tau(\nu(E))(h) = \kappa(E_C)$. From this, we obtain a characterization of solutions of the knapsack expression $E$.

▶ **Proposition 6.1.** *Let $\nu \in \mathbb{N}^{V_E}$ with $\nu = \mu \oplus \kappa$ for $\mu \in \mathbb{N}^M$ and $\kappa \in \mathbb{N}^S$. Then $\nu(E) = 1$ if and only if $\sigma(\nu(E)) = 1$ and there is a $\kappa$-cancelling profile $P$ such that every $\mu$-cluster is contained in $P$.*

This allows us to decompose a knapsack instance for $G \wr H$ into two tasks: determining which profiles are cancelling and finding a $\mu \in \mathbb{N}^M$ such that all $\mu$-clusters are contained in a given profile. The first task can be reduced to solving exponent equation systems over $G$: For each profile $P \subseteq \mathcal{P}(A)$, let $K_P \subseteq \mathbb{N}^S$ be the set of all $\kappa \in \mathbb{N}^S$ such that $P$ is $\kappa$-cancelling.

▶ **Lemma 6.2.** *Let $P$ be a given profile. If $\mathrm{KP}(G^*)$ is decidable, then it is decidable whether $K_P \neq \emptyset$. If $G$ is knapsack-semilinear, then $K_P \subseteq \mathbb{N}^S$ is effectively semilinear.*

For our second task, we employ the effective semilinearity of knapsack solution sets for $H$. Let $L_P \subseteq \mathbb{N}^M$ be the set of all $\mu \in \mathbb{N}^M$ such that every $\mu$-cluster belongs to $P$.

▶ **Lemma 6.3.** *Let $H$ be knapsack-semilinear. For every profile $P \subseteq \mathcal{P}(A)$, the set $L_P$ is effectively semilinear.*

We can define $L_P$ in Presburger arithmetic: In order to express that a given $C \subseteq A$ is a $\mu$-cluster, we employ universal quantification to state that no other address $a \in A \setminus C$ is visited at the same node as the addresses in $C$. This leads to a $\Pi_2$-formula.

We can now prove Theorem 5.3 and 5.4. Let $H$ be knapsack-semilinear and let $\mathrm{KP}(G^*)$ be decidable. Observe that for $\nu = \mu \oplus \kappa$, where $\mu \in \mathbb{N}^M$ and $\kappa \in \mathbb{N}^S$, the value of $\sigma(\nu(E))$ only depends on $\mu$. The set $T \subseteq \mathbb{N}^M$ of all $\mu$ such that $\sigma(\nu(E)) = 1$ is effectively semilinear by knapsack-semilinearity of $H$. Proposition 6.1 tells us that $\mathsf{Sol}(E) = \bigcup_{P \subseteq \mathcal{P}(A)} K_P \oplus (L_P \cap T)$ and $L_P$ is effectively semilinear by Lemma 6.3. This implies Theorem 5.3: We can decide solvability of $E$ by checking, for each profile $P \subseteq \mathcal{P}(A)$, whether $K_P \neq \emptyset$ (which is decidable by Lemma 6.2) and whether $L_P \cap T \neq \emptyset$. Moreover, if $G$ is knapsack-semilinear, then $K_P$ and thus $\mathsf{Sol}(E)$ are effectively semilinear as well. This proves Theorem 5.4.

## 7 Complexity: Proof of Theorem 5.6

Throughout the section we fix a finitely generated group $G$. The goal of this section is to show that if $G$ is abelian and non-trivial, then $\mathrm{KP}(G \wr \mathbb{Z})$ is $\mathsf{NP}$-complete.

### 7.1 Periodic words over groups

With $G^\omega$ we denote the group of all mappings $f \colon \mathbb{N} \to G$ with the pointwise multiplication $(fg)(n) = f(n)g(n)$. The identity element is the mapping id with $\mathrm{id}(n) = 1$ for all $n \in \mathbb{N}$. If $G$ is abelian, then also $G^\omega$ is abelian and we write $\sum_{i=1}^n f_i$ for $f_1 \cdots f_n$ with $f_i \in G^\omega$. A function $f \in G^\omega$ is *periodic with period* $q \geq 1$ if $f(k) = f(k+q)$ for all $k \geq 0$. Note that $q$ is not assumed to be minimal. Let $G^\rho$ be the set of all periodic functions from $G^\omega$. With $G^+$ we denote the set of all tuples $(g_0, \dots, g_{q-1})$ over $G$ of arbitrary length $q \geq 1$. A periodic function $f \in G^\rho$ with period $q$ can be specified by the tuple $(f(0), \dots, f(q-1)) \in G^+$. Vice versa, a tuple $u = (g_0, \dots, g_{q-1}) \in G^+$ defines the periodic function $f_u \in G^\omega$ with $f_u(n \cdot q + r) = g_r$ for $n \geq 0$ and $0 \leq r < q$. One can view this mapping as the sequence $u^\omega$ obtained by taking infinitely many repetitions of $u$. If $f_1$ is periodic with period $q_1$ and $f_2$ is periodic with period $q_2$, then $f_1 f_2$ is periodic with period $q_1 q_2$ (in fact, $\mathrm{lcm}(q_1, q_2)$). Hence, $G^\rho$ forms a countable subgroup of $G^\omega$. Note that $G^\rho$ is not finitely generated: The subgroup generated by elements $f_i \in G^\rho$ with period $q_i$ $(1 \leq i \leq n)$ contains only functions with period $\mathrm{lcm}(q_1, \dots, q_n)$. For $n \geq 0$ let $G_n^\rho \leq G^\rho$ be the subgroup of all $f \in G^\rho$ with $f(k) = 1$ for all $0 \leq k \leq n-1$. The *uniform membership problem for subgroups* $G_n^\rho$, $\textsc{Membership}(G_*^\rho)$, is the following problem:

**Input:** Tuples $u_1, u_2, \dots, u_n \in G^+$ and a binary encoded number $m$.
**Question:** Does the product $f_{u_1} f_{u_2} \cdots f_{u_n} \in G^\rho$ belong to $G_m^\rho$?

▶ **Theorem 7.1.** *For every finitely generated abelian group $G$, $\textsc{Membership}(G_*^\rho)$ can be solved in polynomial time.*

**Proof.** We use the additive notation for $G^\omega$. Let $u_1, \dots, u_n \in G^+$, $q_i = |u_i|$, and $f = \sum_{i=1}^n f_{u_i}$. We show that if there exists a position $m$ such that $f(m) \neq 0$, then there exists a position $m < \sum_{i=1}^n q_i$ such that $f(m) \neq 0$. This suffices to prove the theorem since the word problem for a finitely generated abelian group can be solved in polynomial time.

Let $m \geq \sum_{i=1}^n q_i$ and assume that $f(j) = 0$ for all $j$ with $m - \sum_{i=1}^n q_i \leq j < m$. We show that $f(m) = 0$, which proves the above claim.

Note that $f_{u_i}(j) = f_{u_i}(j - q_i)$ for all $j \geq q_i$, $1 \leq i \leq n$. For $M \subseteq [1, n]$ let $q_M = \sum_{i \in M} q_i$. Moreover, for $1 \leq k \leq n$ let $\mathcal{M}_k = \{M \subseteq [1, n], |M| = k\}$. For $1 \leq k \leq n-1$ we get

$$\begin{aligned}\sum_{M \in \mathcal{M}_k} \sum_{i \in M} f_{u_i}(m - q_M) \ &= \sum_{M \in \mathcal{M}_k} \sum_{i \in [1,n] \setminus M} -f_{u_i}(m - q_M) \ = \sum_{M \in \mathcal{M}_k} \sum_{i \in [1,n] \setminus M} -f_{u_i}(m - q_M - q_i) \\ &= \sum_{i=1}^{n} \sum_{M \in \mathcal{M}_k, i \notin M} -f_{u_i}(m - q_{M \cup \{i\}}) \ = \sum_{M \in \mathcal{M}_{k+1}} \sum_{i \in M} -f_{u_i}(m - q_M).\end{aligned}$$

We can write

$$f(m) \ = \ \sum_{i=1}^{n} f_{u_i}(m) \ = \ \sum_{i=1}^{n} f_{u_i}(m - q_i) \ = \ \sum_{M \in \mathcal{M}_1} \sum_{i \in M} f_{u_i}(m - q_M).$$

From the above identities we get by induction:

$$f(m) \ = \ \pm \sum_{M \in \mathcal{M}_n} \sum_{i \in M} f_{u_i}(m - q_M) \ = \ \pm \sum_{i \in [1,n]} f_{u_i}(m - q_{[1,n]}) \ = \ \pm f(m - \sum_{i=1}^{n} q_i) = 0.$$

This proves the claim and hence the theorem. ◀

## 7.2 Automata for Cayley representations

The main technical result of this section is:

▶ **Proposition 7.2.** *Let $G$ be a finitely generated abelian group. If* $\textsc{ExpEq}(G) \in \mathsf{NP}$ *and* $\textsc{Membership}(G_*^{\rho}) \in \mathsf{NP}$*, then also* $\mathrm{KP}(G \wr \mathbb{Z}) \in \mathsf{NP}$.

We start with some definitions. An interval $[a, b] \subseteq \mathbb{Z}$ *supports* an element $(f, d) \in G \wr \mathbb{Z}$ if $\{0, d\} \cup \mathsf{supp}(f) \subseteq [a, b]$. If $(f, d) \in G \wr \mathbb{Z}$ is a product of length $n$ over the generators, then the minimal interval $[a, b]$ which supports $(f, d)$ satisfies $b - a \leq n$. A knapsack expression $E = v_0 u_1^{x_1} v_1 \cdots u_k^{x_k} v_k$ is called *rigid* if each $u_i$ evaluates to an element $(f_i, 0) \in G \wr \mathbb{Z}$. Intuitively, the movement of the cursor is independent from the values of the variables $x_i$ up to repetition of loops. In particular, every variable-free expression is rigid.

In the following we define the so called Cayley representation of a rigid knapsack expression. This is a finite word, where every symbol is a marked knapsack expression over $G$. A marked knapsack expression over $G$ is of the form $E$, $\overline{E}$, $\underline{E}$, or $\underline{\overline{E}}$, where $E$ is a knapsack expression over $G$. We say that $\overline{E}$ and $\underline{\overline{E}}$ (resp., $\underline{E}$ and $\underline{\overline{E}}$) are top-marked (resp., bottom-marked).

Let $E = v_0 u_1^{x_1} v_1 \cdots u_k^{x_k} v_k$ be a rigid knapsack expression over $G \wr \mathbb{Z}$. For an assignment $\nu$ let $(f_\nu, d) \in G \wr \mathbb{Z}$ be the element to which $\nu(E)$ evaluates, i.e. $(f_\nu, d) = \nu(E)$. Note that $d$ does not depend on $\nu$. Because of the rigidity of $E$, there is an interval $[a, b] \subseteq \mathbb{Z}$ that supports $(f_\nu, d)$ for all assignments $\nu$. For each $j \in [a, b]$ let $E_j$ be a knapsack expression over $G$ with the variables $x_1, \ldots, x_k$ such that $f_\nu(j) = \nu(E_j)$ for all assignments $\nu$. Then we call the formal expression

$$r = \begin{cases} E_a \, E_{a+1} \, \cdots \, E_{-1} \, \overline{E_0} \, E_1 \, \cdots \, E_{d-1} \, \underline{E_d} \, E_{d+1} \, \cdots \, E_b & \text{if } d > 0 \\ E_a \, E_{a+1} \, \cdots \, E_{-1} \, \underline{\overline{E_0}} \, E_1 \, \cdots \, E_b & \text{if } d = 0 \; . \\ E_a \, E_{a+1} \, \cdots \, E_{d-1} \, \underline{E_d} \, E_{d+1} \, \cdots \, E_{-1} \, \overline{E_0} \, E_1 \, \cdots \, E_b & \text{if } d < 0 \end{cases}$$

a *Cayley representation* of $E$ (or $E$ is *represented* by $r$). Formally, a Cayley representation $r$ is a sequence of marked knapsack expressions, and the length of this sequence is denote

| -1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $a^x$ | $\underline{\overline{a^x}}$ | $b^x$ | | | | | | | | | | | |
| | $\overline{1}$ | $\underline{1}$ | | | | | | | | | | | |
| | | $\underline{\overline{b^y}}$ | $b^y$ | $b^y$ | | | | | | | | | |
| | $b$ | $\overline{a}$ | $b^{-1}$ | $\underline{a}$ | $a$ | | | | | | | | |
| | | | $b$ | $\overline{a}$ | $b^{-1}$ | $\underline{a}$ | $a$ | | | | | | |
| | | | | | $b$ | $\overline{a}$ | $b^{-1}$ | $\underline{a}$ | $a$ | | | | |
| | | | | | | | $b$ | $\overline{a}$ | $b^{-1}$ | $\underline{a}$ | $a$ | | |
| | | | | | | | | | $b$ | $\overline{a}$ | $b^{-1}$ | $\underline{a}$ | $a$ |
| $a^x$ | $\overline{a^x b}$ | $b^x b^y a$ | $b^y$ | $b^y a^2$ | $a$ | $a^2$ | $a$ | $a^2$ | $a$ | $a^2$ | $ab^{-1}$ | $\underline{a}$ | $a$ |

**Figure 1** Cayley representation

by $|r|$. In our examples, we separate for better readability consecutive marked knapsack expressions in $r$ by commas. By the above definition, $r$ depends on the chosen supporting interval $[a, b]$. However, compared to the representation of the minimal supporting interval, any other Cayley representation differs only by adding 1's (i.e., trivial knapsack expressions over $G$) at the left and right end of $r$.

A Cayley representation of $E$ records for each point in $\mathbb{Z}$ an expression that describes which element will be placed at that point. Multiplying an element of $G \wr \mathbb{Z}$ always begins at a particular cursor position; in a Cayley representation, the marker on top specifies the expression that is placed at the cursor position in the beginning. Moreover, a Cayley representation describes how the cursor changes when multiplying $\nu(E)$: The marker on the bottom specifies where the cursor is located in the end.

▶ **Example 7.3.** Consider the wreath product $F_2 \wr \mathbb{Z}$, where $F_2$ is the free group generated by $\{a, b\}$ and $\mathbb{Z}$ is generated by $t$, and the rigid knapsack expression $E = u_1^x u_2 u_3^y u_4^5$ with $u_1 = at^{-1}at^2bt^{-1}$ (represented by $a\,\underline{\overline{a}}\,b$), $u_2 = t$ (represented by $\overline{1}\,\underline{1}$), $u_3 = btbtbt^{-2}$ (represented by $\underline{\overline{b}}\,b\,b$), and $u_4 = at^{-1}bt^2b^{-1}tatat^{-1}$ (represented by $b\,\overline{a}\,b^{-1}\,\underline{a}\,a$).

A Cayley representation of $u_1^x$ is $a^x\,\underline{\overline{a^x}}\,b^{-1}$ and a Cayley representation of $u_3^y$ is $\underline{\overline{b^y}}\,b^y\,b^y$. The diagram in Figure 1 illustrates how to compute a Cayley representation $r$ of $E$, which is shown in the bottom line. Here, we have chosen the supporting interval minimal. Note that if we replace the exponents 5 in $u_4^5$ by a larger number, then we only increase the number of repetitions of the factor $a, a^2$ in the Cayley representation.

Let $E$ be an arbitrary knapsack expression over $G \wr \mathbb{Z}$. We can assume that $E$ has the form $u_1^{x_1} \cdots u_k^{x_k} u_{k+1}$. Let $X_0$ be the set of all variables $x_i$ where $u_i$ evaluates to an element $(f, 0) \in G \wr \mathbb{Z}$, and let $X_1 = \{x_1, \ldots, x_k\} \setminus X_0$. For a partial assignment $\nu\colon X_1 \to \mathbb{N}$ we obtain a rigid knapsack expression $E_\nu$ by replacing in $E$ every variable $x_i \in X_1$ by $\nu(x_i)$. A set $R$ of Cayley representations is a *set representation* of $E$ if

- for each assignment $\nu\colon X_1 \to \mathbb{N}$ there exists $r \in R$ such that $r$ represents $E_\nu$,
- for each $r \in R$ there exists an assignment $\nu\colon X_1 \to \mathbb{N}$ such that $r$ represents $E_\nu$ and $\nu(x) \leq |r|$ for all $x \in X_1$.

▶ **Example 7.4.** Consider the non-rigid knapsack expression $E' = u_1^x u_2 u_3^y u_4^z$ over $F_2 \wr \mathbb{Z}$, where $u_1, u_2, u_3, u_4$ are taken from Example 7.3. We have $X_0 = \{x, y\}$ and $X_1 = \{z\}$. A set representation $R$ of $E'$ consists of the following Cayley representations: $a^x, \overline{a^x}, \underline{b^x b^y}, b^y, b^y$ for $\nu(z) = 0$, $a^x, \overline{a^x b}, b^x b^y a, b^y b^{-1}, \underline{b^y a}, a$ for $\nu(z) = 1$, and

$$a^x, \overline{a^x b}, b^x b^y a, b^y, b^y a^2, (a, a^2)^{\nu(z)-2}, ab^{-1}, \underline{a}, a \quad \text{for } \nu(z) \geq 2.$$

Only finitely many different marked knapsack expressions appear in this set representation $R$, and $R$ is clearly a regular language over the finite alphabet consisting of this finitely many marked knapsack expressions.

We can now sketch the proof of Proposition 7.2. The main idea is to construct a nondeterministic finite automaton (NFA) that accepts a set representation of $E = u_1^{x_1} \cdots u_k^{x_k} u_{k+1}$. Let $n = |E|$. First, we compute polynomial-size NFAs $\mathcal{A}_i$ ($i \in [1, k+1]$), where $\mathcal{A}_i$ accepts a set representation of $u_i^{x_i}$ (or $u_{k+1}$ for $i = k+1$). For $u_{k+1}$ and expressions $u_i^{x_i}$ with $x_i \in X_0$ these set representations are singletons and the construction of $\mathcal{A}_i$ is straightforward, see e.g. Example 7.3. For expressions $u_i^{x_i}$ with $x_i \in X_1$ one has to construct an NFA that accepts a set containing a Cayley representation of every $u_i^m$ (a variable-free knapsack expression over $G$) for $m \geq 0$. The main observation is that all these Cayley representations are periodic (except for a short prefix and suffix) with the same period.

From the NFAs $\mathcal{A}_i$ one obtains an NFA $\mathcal{A}$ accepting a set representation of $E$ using a simple product construction. This NFA $\mathcal{A}$ has exponential size in $n$, so we cannot construct it. However, its exponential size bound on $\mathcal{A}$ yields that $E = 1$ has a solution if and only if there exists a solution $\nu$ such that $\nu(x)$ is exponentially bounded in $n$ for all $x \in X_1$. Since each $\mathcal{A}_i$ accepts a set representation of $u_i^{x_i}$, $i \in [1, k]$ or of $u_{k+1}$, this implies that solvability of $E$ is witnessed by words $\alpha_i \in L(\mathcal{A}_i)$ for $i \in [1, k+1]$ whose length is bounded exponentially in $n$. The periodic nature of the words $\alpha_i$ allows to represent these words in polynomial space as a concatenation of powers $\beta^m$ for binary encoded numbers $m$. We guess such representations of the $\alpha_i$.

It remains to verify that the guessed words $\alpha_i$ indeed witness a solution of $E = 1$. This means that there exists a valuation $\nu \colon X_0 \to \mathbb{N}$ such that for every position $p$ the $(k+1)$-tuple consisting of the $p$-th entries of the $\alpha_i$ evaluates to 1 under $\nu$. There exist only polynomially many positions $p$ where an expression $u^x$ with $x \in X_0$ occurs in some $\alpha_i$. Thus, we can construct from all these positions an instance of ExpEq$(G)$. The remaining pieces of the $\alpha_i$ only contain group elements from $G$ and are periodic. The question, whether they cancel at all remaining positions is an instance of Membership$(G_*^\rho)$.

Proposition 7.2 yields the NP upper bound for Theorem 5.6: If $G$ is a finitely generated abelian group, then ExpEq$(G)$ corresponds to the solvability problem for linear equation systems over the integers, possibly with modulo-constraints. This problem is well known to be in NP. Moreover, Membership$(G_*^\rho)$ can be solved in polynomial time by Theorem 7.1.

It remains to prove the NP-hardness part of Theorem 5.6. Using a reduction from 3-dimensional matching, one can show the following [4]:

▶ **Theorem 7.5.** *If $G$ is non-trivial and $H$ contains an element of infinite order, then knapsack and subset sum for $G \wr H$ are* NP*-hard.*

## 8 Open problems

The main open problem is to characterize those $G$ and $H$ for which KP$(G \wr H)$ is decidable. Concerning complexity, we are confident that our NP upper bound for KP$(G \wr \mathbb{Z})$, where $G$ is finitely generated abelian, can be extended to KP$(G \wr H)$, where $H$ is a finitely generated free group or $\mathbb{Z}^n$ for some $n \geq 0$. Another question is whether the assumption on $G$ being abelian can be weakened. In particular, we want to investigate whether polynomial time algorithms exist for Membership$(G_*^\rho)$ for certain non-abelian groups $G$.

## References

1 T. C. Davis and A. Yu. Olshanskii. Subgroup distortion in wreath products of cyclic groups. *Journal of Pure and Applied Algebra*, 215(12):2987–3004, 2011.

2 M. Elberfeld, A. Jakoby, and T. Tantau. Algorithmic meta theorems for circuit classes of constant and logarithmic depth. *Electronic Colloquium on Computational Complexity (ECCC)*, 18:128, 2011.

3 E. Frenkel, A. Nikolaev, and A. Ushakov. Knapsack problems in products of groups. *Journal of Symbolic Computation*, 74:96–108, 2016.

4 M. Ganardi, D. König, M. Lohrey, and G. Zetzsche. Knapsack problems for wreath products. *CoRR*, abs/1709.09598, 2017. URL: `http://arxiv.org/abs/1709.09598`.

5 S. Ginsburg and E. H. Spanier. Semigroups, Presburger formulas, and languages. *Pacific Journal of Mathematics*, 16(2):285–296, 1966. `doi:10.2140/pjm.1966.16.285`.

6 C. Haase. *On the complexity of model checking counter automata.* PhD thesis, University of Oxford, St Catherine's College, 2011.

7 R. M. Karp. Reducibility among combinatorial problems. In R. E. Miller and J. W. Thatcher, editors, *Complexity of Computer Computations*, pages 85–103. Plenum Press, 1972.

8 D. König, M. Lohrey, and G. Zetzsche. Knapsack and subset sum problems in nilpotent, polycyclic, and co-context-free groups. In *Algebra and Computer Science*, volume 677 of *Contemporary Mathematics*, pages 138–153. American Mathematical Society, 2016.

9 J. Lehnert and P. Schweitzer. The co-word problem for the Higman-Thompson group is context-free. *Bulletin of the London Mathematical Society*, 39(2):235–241, 2007.

10 M. Lohrey, B. Steinberg, and G. Zetzsche. Rational subsets and submonoids of wreath products. *Information and Computation*, 243:191–204, 2015.

11 M. Lohrey and G. Zetzsche. Knapsack in graph groups, HNN-extensions and amalgamated products. In Nicolas Ollinger and Heribert Vollmer, editors, *Proc. of the 33rd International Symposium on Theoretical Aspects of Computer Science (STACS 2016)*, volume 47 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 50:1–50:14, Dagstuhl, Germany, 2016. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

12 M. Lohrey and G. Zetzsche. The complexity of knapsack in graph groups. In *Proceedings of the 34th Symposium on Theoretical Aspects of Computer Science, STACS 2017*, volume 66 of *LIPIcs*, pages 52:1–52:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017.

13 A. Miasnikov, S. Vassileva, and A. Weiß. The conjugacy problem in free solvable groups and wreath products of abelian groups is in $\mathrm{TC}^0$. In *Computer Science – Theory and Applications – 12th International Computer Science Symposium in Russia, CSR 2017, Proceedings*, volume 10304 of *Lecture Notes in Computer Science*, pages 217–231. Springer, 2017.

14 A. Mishchenko and A. Treier. Knapsack problem for nilpotent groups. *Groups Complexity Cryptology*, 9(1):87–98, 2017.

15 A. Myasnikov, A. Nikolaev, and A. Ushakov. Knapsack problems in groups. *Mathematics of Computation*, 84:987–1016, 2015.

16 A. Nikolaev and A. Ushakov. Subset sum problem in polycyclic groups. *Journal of Symbolic Computation*, 84:84–94, 2018.

17 C. Sims. *Computation with finitely presented groups.* Cambridge University Press, 1994.

18 M. Wilhelm. On a theorem of Marshall Hall. *Annals of Mathematics. Second Series*, 40:764–768, 1939.