Knapsack in Graph Groups, HNN-Extensions and Amalgamated Products

Markus Lohrey[1] Georg Zetzsche[2]*

[1]Department für Elektrotechnik und Informatik
Universität Siegen

[2]LSV, CNRS & ENS Cachan
Université Paris-Saclay

STACS 2016

*Supported by a fellowship within the Postdoc-Program of the German Academic Exchange Service (DAAD).

The knapsack problem in groups

Definition (Myasnikov, Nikolaev, and Ushakov)

Let G be a finitely generated group. The *knapsack problem for G* is the following decision problem:

Given: Elements $g_1, \ldots, g_k, g \in G$.

Question: Are there $x_1, \ldots, x_k \in \mathbb{N}$ with $g_1^{x_1} \cdots g_k^{x_k} = g$?

The knapsack problem in groups

Definition (Myasnikov, Nikolaev, and Ushakov)

Let G be a finitely generated group. The *knapsack problem for G* is the following decision problem:

Given: Elements $g_1, \ldots, g_k, g \in G$.

Question: Are there $x_1, \ldots, x_k \in \mathbb{N}$ with $g_1^{x_1} \cdots g_k^{x_k} = g$?

The knapsack problem

- If $G = \mathbb{Z}$ and elements are encoded in binary: NP-complete.

The knapsack problem in groups

Definition (Myasnikov, Nikolaev, and Ushakov)

Let G be a finitely generated group. The *knapsack problem for G* is the following decision problem:

Given: Elements $g_1, \ldots, g_k, g \in G$.

Question: Are there $x_1, \ldots, x_k \in \mathbb{N}$ with $g_1^{x_1} \cdots g_k^{x_k} = g$?

The knapsack problem

- If $G = \mathbb{Z}$ and elements are encoded in binary: NP-complete.
- For which groups is knapsack decidable?

The knapsack problem in groups

Definition (Myasnikov, Nikolaev, and Ushakov)

Let G be a finitely generated group. The *knapsack problem for G* is the following decision problem:

Given: Elements $g_1, \ldots, g_k, g \in G$.

Question: Are there $x_1, \ldots, x_k \in \mathbb{N}$ with $g_1^{x_1} \cdots g_k^{x_k} = g$?

The knapsack problem

- If $G = \mathbb{Z}$ and elements are encoded in binary: NP-complete.
- For which groups is knapsack decidable?
- What is the complexity?

Graph Groups

Definition

Let A be an alphabet and $I \subseteq A \times A$ be irreflexive and symmetric.

Graph Groups

Definition

Let A be an alphabet and $I \subseteq A \times A$ be irreflexive and symmetric. The group $\mathbb{G}(A, I)$ is defined as

$$\mathbb{G}(A, I) = \langle A \mid ab = ba \ ((a, b) \in I) \rangle.$$

Graph Groups

Definition

Let A be an alphabet and $I \subseteq A \times A$ be irreflexive and symmetric. The group $\mathbb{G}(A, I)$ is defined as

$$\mathbb{G}(A, I) = \langle A \mid ab = ba \ ((a, b) \in I)\rangle.$$

Groups of the form $\mathbb{G}(A, I)$ are called *graph groups*.

Graph Groups

Definition

Let A be an alphabet and $I \subseteq A \times A$ be irreflexive and symmetric. The group $\mathbb{G}(A, I)$ is defined as

$$\mathbb{G}(A, I) = \langle A \mid ab = ba \ ((a, b) \in I) \rangle.$$

Groups of the form $\mathbb{G}(A, I)$ are called *graph groups*.

Theorem

For each graph group $\mathbb{G}(A, I)$, knapsack is in NP.

Virtually special groups

A group is *virtually special* if it is a finite extension of a subgroup of a graph group.

Virtually special groups

A group is *virtually special* if it is a finite extension of a subgroup of a graph group.
Class turned out to be very rich:

- Coxeter groups
- one-relator groups with torsion
- fully residually free groups
- fundamental groups of hyperbolic 3-manifolds

Virtually special groups

A group is *virtually special* if it is a finite extension of a subgroup of a graph group.
Class turned out to be very rich:

- Coxeter groups
- one-relator groups with torsion
- fully residually free groups
- fundamental groups of hyperbolic 3-manifolds

Not hard to see: finite extensions inherit NP-membership.

Virtually special groups

A group is *virtually special* if it is a finite extension of a subgroup of a graph group.
Class turned out to be very rich:

- Coxeter groups
- one-relator groups with torsion
- fully residually free groups
- fundamental groups of hyperbolic 3-manifolds

Not hard to see: finite extensions inherit NP-membership.

Theorem

For every virtually special group, knapsack is in NP.

Virtually special groups

A group is *virtually special* if it is a finite extension of a subgroup of a graph group.

Class turned out to be very rich:

- Coxeter groups
- one-relator groups with torsion
- fully residually free groups
- fundamental groups of hyperbolic 3-manifolds

Not hard to see: finite extensions inherit NP-membership.

Theorem

For every virtually special group, knapsack is in NP.

Did we make the problem easy by using unary encoding?

Succinct encoding of strings

A *straight-line program (SLP)* is a context-free grammar that generates exactly one string.

Succinct encoding of strings

A *straight-line program (SLP)* is a context-free grammar that generates exactly one string.

Example

$A_0 \to a, \;\; A_i \to A_{i-1}A_{i-1}$

Succinct encoding of strings

A *straight-line program (SLP)* is a context-free grammar that generates exactly one string.

Example

$A_0 \to a, \;\; A_i \to A_{i-1}A_{i-1}$: Start symbol A_n generates a^{2^n}.

Succinct encoding of strings

A *straight-line program (SLP)* is a context-free grammar that generates exactly one string.

Example

$A_0 \to a, \;\; A_i \to A_{i-1}A_{i-1}$: Start symbol A_n generates a^{2^n}.

- For strings over one letter, SLPs are essentially binary encodings.

Succinct encoding of strings

A *straight-line program (SLP)* is a context-free grammar that generates exactly one string.

Example

$A_0 \to a, \;\; A_i \to A_{i-1}A_{i-1}$: Start symbol A_n generates a^{2^n}.

- For strings over one letter, SLPs are essentially binary encodings.
- By a *compressed string*, we mean one given as an SLP.

Succinct encoding of strings

A *straight-line program (SLP)* is a context-free grammar that generates exactly one string.

Example

$A_0 \to a, \;\; A_i \to A_{i-1}A_{i-1}$: Start symbol A_n generates a^{2^n}.

- For strings over one letter, SLPs are essentially binary encodings.
- By a *compressed string*, we mean one given as an SLP.

Theorem

For every virtually special group, compressed knapsack is in NP*.*

Algorithm

Algorithm in $\mathbb{Z}$ case: Guess binary representations and verify.

Algorithm

Algorithm in $\mathbb{Z}$ case: Guess binary representations and verify.
Possible because:

- We have an exponential bound on solution
- Verification can be done in polynomial time

Algorithm

Algorithm in $\mathbb{Z}$ case: Guess binary representations and verify.
Possible because:

- We have an exponential bound on solution
- Verification can be done in polynomial time

Verification in Graph Groups

- Suppose we have an exponential bound on solutions.
- Construct SLP for $g_1^{x_1} \cdots g_k^{x_k}$: $B_0 \to g_k, \quad B_i \to B_{i-1} B_{i-1}$

Algorithm

Algorithm in $\mathbb{Z}$ case: Guess binary representations and verify.
Possible because:

- We have an exponential bound on solution
- Verification can be done in polynomial time

Verification in Graph Groups

- Suppose we have an exponential bound on solutions.
- Construct SLP for $g_1^{x_1} \cdots g_k^{x_k}$: $B_0 \to g_k, \quad B_i \to B_{i-1}B_{i-1}$

Theorem (Lohrey, Schleimer 2007)

For every fixed graph group, the compressed word problem belongs to P.

Algorithm

Algorithm in $\mathbb{Z}$ case: Guess binary representations and verify.
Possible because:

- We have an exponential bound on solution
- Verification can be done in polynomial time

Verification in Graph Groups

- Suppose we have an exponential bound on solutions.
- Construct SLP for $g_1^{x_1} \cdots g_k^{x_k}$: $B_0 \to g_k, \quad B_i \to B_{i-1}B_{i-1}$

Theorem (Lohrey, Schleimer 2007)

For every fixed graph group, the compressed word problem belongs to P.

Task

Show: If there is a solution, then there is an exponential one.

Trace monoids

Definition

- Let A be an alphabet and $I \subseteq A \times A$ irreflexive and symmetric.

Trace monoids

Definition

- Let A be an alphabet and $I \subseteq A \times A$ irreflexive and symmetric.
- Let $\equiv_I$ be the smallest congruence on A^* with $ab \equiv_I ba$ for all $(a, b) \in I$.

Trace monoids

Definition

- Let A be an alphabet and $I \subseteq A \times A$ irreflexive and symmetric.
- Let $\equiv_I$ be the smallest congruence on A^* with $ab \equiv_I ba$ for all $(a, b) \in I$.
- The *trace monoid* $\mathbb{M}(A, I)$ is defined as

$$\mathbb{M}(A, I) \;=\; A^*/{\equiv_I}.$$

Trace monoids

Definition

- Let A be an alphabet and $I \subseteq A \times A$ irreflexive and symmetric.
- Let $\equiv_I$ be the smallest congruence on A^* with $ab \equiv_I ba$ for all $(a, b) \in I$.
- The *trace monoid* $\mathbb{M}(A, I)$ is defined as

$$\mathbb{M}(A, I) \;=\; A^*/{\equiv_I}.$$

- $[u]_I$ denotes the congruence class of $u \in A^*$.

Trace monoids

Definition

- Let A be an alphabet and $I \subseteq A \times A$ irreflexive and symmetric.
- Let $\equiv_I$ be the smallest congruence on A^* with $ab \equiv_I ba$ for all $(a, b) \in I$.
- The *trace monoid* $\mathbb{M}(A, I)$ is defined as

$$\mathbb{M}(A, I) \;=\; A^*/{\equiv_I}.$$

- $[u]_I$ denotes the congruence class of $u \in A^*$.
- We consider $\mathbb{M}(A^{\pm 1}, I^{\pm 1})$, where

$$A^{\pm 1} = \{a^{+1}, a^{-1} \mid a \in A\}, \qquad I^{\pm} = \{(a^{\pm 1}, b^{\pm 1}) \mid (a, b) \in I\}.$$

Trace monoids

Definition

- Let A be an alphabet and $I \subseteq A \times A$ irreflexive and symmetric.
- Let $\equiv_I$ be the smallest congruence on A^* with $ab \equiv_I ba$ for all $(a, b) \in I$.
- The *trace monoid* $\mathbb{M}(A, I)$ is defined as

$$\mathbb{M}(A, I) \;=\; A^*/{\equiv_I}.$$

- $[u]_I$ denotes the congruence class of $u \in A^*$.
- We consider $\mathbb{M}(A^{\pm 1}, I^{\pm 1})$, where

$$A^{\pm 1} = \{a^{+1}, a^{-1} \mid a \in A\}, \qquad I^{\pm} = \{(a^{\pm 1}, b^{\pm 1}) \mid (a, b) \in I\}.$$

- A trace t is *irreducible* if there is no decomposition $t = [uaa^{-1}v]_I$ for $a \in A^{\pm 1}, u, v \in (A^{\pm 1})^*$.

We call a trace t *connected* if there is no factorization $t = uv$ with $u \neq 1 \neq v$ and uIv.

We call a trace t *connected* if there is no factorization $t = uv$ with $u \neq 1 \neq v$ and uIv.

We call a trace t *connected* if there is no factorization $t = uv$ with $u \neq 1 \neq v$ and uIv.

Lemma

Fix the alphabet A. Let $p, q, u, v, s, t \in \mathbb{M}(A, I)$ with $u \neq 1$ and $v \neq 1$ connected. Then the set

$$L(p, u, s, q, v, t) := \{(x, y) \in \mathbb{N} \times \mathbb{N} \mid pu^x s = qv^y t\}$$

is a union of linear sets of the form $\{(a + bz, c + dz) \mid z \in \mathbb{N}\}$ where a, b, c, d are polynomial in the lengths of p, q, u, v, s, t..

We call a trace t *connected* if there is no factorization $t = uv$ with $u \neq 1 \neq v$ and uIv.

Lemma

Fix the alphabet A. Let $p, q, u, v, s, t \in \mathbb{M}(A, I)$ with $u \neq 1$ and $v \neq 1$ connected. Then the set

$$L(p, u, s, q, v, t) := \{(x, y) \in \mathbb{N} \times \mathbb{N} \mid pu^x s = qv^y t\}$$

is a union of linear sets of the form $\{(a + bz, c + dz) \mid z \in \mathbb{N}\}$ where a, b, c, d are polynomial in the lengths of p, q, u, v, s, t..

- Techniques from recognizable trace languages:
- Construct poly-size automaton for $L = [pu^* s]_I \cap [qv^* t]_I$.

We call a trace t *connected* if there is no factorization $t = uv$ with $u \neq 1 \neq v$ and uIv.

Lemma

Fix the alphabet A. Let $p, q, u, v, s, t \in \mathbb{M}(A, I)$ with $u \neq 1$ and $v \neq 1$ connected. Then the set

$$L(p, u, s, q, v, t) := \{(x, y) \in \mathbb{N} \times \mathbb{N} \mid pu^x s = qv^y t\}$$

is a union of linear sets of the form $\{(a + bz, c + dz) \mid z \in \mathbb{N}\}$ where a, b, c, d are polynomial in the lengths of p, q, u, v, s, t..

- Techniques from recognizable trace languages:
- Construct poly-size automaton for $L = [pu^* s]_I \cap [qv^* t]_I$.
- Possible because u and v are connected.

We call a trace t *connected* if there is no factorization $t = uv$ with $u \neq 1 \neq v$ and uIv.

Lemma

Fix the alphabet A. Let $p, q, u, v, s, t \in \mathbb{M}(A, I)$ with $u \neq 1$ and $v \neq 1$ connected. Then the set

$$L(p, u, s, q, v, t) := \{(x, y) \in \mathbb{N} \times \mathbb{N} \mid pu^x s = qv^y t\}$$

is a union of linear sets of the form $\{(a + bz, c + dz) \mid z \in \mathbb{N}\}$ where a, b, c, d are polynomial in the lengths of p, q, u, v, s, t..

- Techniques from recognizable trace languages:
- Construct poly-size automaton for $L = [pu^*s]_I \cap [qv^*t]_I$.
- Possible because u and v are connected.
- Apply results for unary finite automata to set of lengths of L:
- Known size bounds for semilinear representation

Levi's Lemma

Lemma

Let $u_1, \ldots, u_m, v_1, \ldots, v_n \in \mathbb{M}(A, I)$. *Then* $u_1 u_2 \cdots u_m = v_1 v_2 \cdots v_n$ *if and only if there exist* $w_{i,j} \in \mathbb{M}(A, I)$ $(1 \leqslant i \leqslant m,\ 1 \leqslant j \leqslant n)$ *such that*

- $u_i = w_{i,1} w_{i,2} \cdots w_{i,n}$ *for every* $1 \leqslant i \leqslant m$,
- $v_j = w_{1,j} w_{2,j} \cdots w_{m,j}$ *for every* $1 \leqslant j \leqslant n$, *and*
- $(w_{i,j}, w_{k,\ell}) \in I$ *if* $1 \leqslant i < k \leqslant m$ *and* $n \geqslant j > \ell \geqslant 1$.

v_n	$w_{1,n}$	$w_{2,n}$	$w_{3,n}$	$\ldots$	$w_{m,n}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
v_3	$w_{1,3}$	$w_{2,3}$	$w_{3,3}$	$\ldots$	$w_{m,3}$
v_2	$w_{1,2}$	$w_{2,2}$	$w_{3,2}$	$\ldots$	$w_{m,2}$
v_1	$w_{1,1}$	$w_{2,1}$	$w_{3,1}$	$\ldots$	$w_{m,1}$
	u_1	u_2	u_3	$\ldots$	u_m

Levi's Lemma

Lemma

Let $u_1, \ldots, u_m, v_1, \ldots, v_n \in \mathbb{M}(A, I)$. *Then* $u_1 u_2 \cdots u_m = v_1 v_2 \cdots v_n$ *if and only if there exist* $w_{i,j} \in \mathbb{M}(A, I)$ $(1 \leqslant i \leqslant m,\ 1 \leqslant j \leqslant n)$ *such that*

- $u_i = w_{i,1} w_{i,2} \cdots w_{i,n}$ *for every* $1 \leqslant i \leqslant m$,
- $v_j = w_{1,j} w_{2,j} \cdots w_{m,j}$ *for every* $1 \leqslant j \leqslant n$, *and*
- $(w_{i,j}, w_{k,\ell}) \in I$ *if* $1 \leqslant i < k \leqslant m$ *and* $n \geqslant j > \ell \geqslant 1$.

v_n	$w_{1,n}$	$w_{2,n}$	$w_{3,n}$	$\ldots$	$w_{m,n}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
v_3	$w_{1,3}$	$w_{2,3}$	$w_{3,3}$	$\ldots$	$w_{m,3}$
v_2	$w_{1,2}$	$w_{2,2}$	$w_{3,2}$	$\ldots$	$w_{m,2}$
v_1	$w_{1,1}$	$w_{2,1}$	$w_{3,1}$	$\ldots$	$w_{m,1}$
	u_1	u_2	u_3	$\ldots$	u_m

Let $u_1, u_2, \ldots, u_n \in \mathrm{IRR}(A^{\pm 1}, I)$ be irreducible traces.
The sequence $u_1, u_2, \ldots, u_n$ is *I-freely reducible* if it can be reduced to the empty sequence ε by the following rules:

- $u_i, u_j \to u_j, u_i$ if $u_i I u_j$
- $u_i, u_j \to \varepsilon$ if $u_i = u_j^{-1}$ in $\mathbb{G}(A, I)$
- $u_i \to \varepsilon$ if $u_i = \varepsilon$.

Let $u_1, u_2, \ldots, u_n \in \mathrm{IRR}(A^{\pm 1}, I)$ be irreducible traces.
The sequence $u_1, u_2, \ldots, u_n$ is *I-freely reducible* if it can be reduced to the empty sequence ε by the following rules:

- $u_i, u_j \to u_j, u_i$ if $u_i I u_j$
- $u_i, u_j \to \varepsilon$ if $u_i = u_j^{-1}$ in $\mathbb{G}(A, I)$
- $u_i \to \varepsilon$ if $u_i = \varepsilon$.

Lemma

Let $n \geqslant 2$ and $u_1, u_2, \ldots, u_n \in \mathrm{IRR}(A^{\pm 1}, I)$. If $u_1 u_2 \cdots u_n = 1$ in $\mathbb{G}(A, I)$, then there exist factorizations $u_i = u_{i,1} \cdots u_{i,k_i}$ such that the sequence

$$u_{1,1}, \ldots, u_{1,k_1},\ u_{2,1}, \ldots, u_{2,k_2},\ \ldots, u_{n,1}, \ldots, u_{n,k_n}$$

is I-freely reducible. Moreover, $\sum_{i=1}^{n} k_i \leqslant 2^n - 2$.

Lemma

Let $u^x = y_1 \cdots y_m$ be an equation where u is a concrete connected trace. It is equivalent to a disjunction of statements

$$\exists x_i > 0 \; (i \in K) : x = \sum_{i \in K} x_i + c \wedge \bigwedge_{i \in K} y_i = p_i u^{x_i} s_i \wedge \bigwedge_{i \in [1,m] \setminus K} y_i = p_i s_i,$$

where

- $K \subseteq [1, m]$
- p_i, s_i *are concrete traces of length polynomial in m and $|u|$*
- c *is a concrete number, polynomial in m*

Theorem

Let $u_1, u_2, \dots, u_n \in \mathbb{G}(A, I)\setminus\{1\}$, $v_0, v_1, \dots, v_n \in \mathbb{G}(A, I)$ and let $x_1, \dots, x_n$ be variables ranging over $\mathbb{N}$. Then, the set of solutions of the exponent equation

$$v_0 u_1^{x_1} v_1 u_2^{x_2} v_2 \cdots u_n^{x_n} v_n = 1$$

is semilinear. Moreover, if there is a solution, then there is a solution where the x_i are exponential in n and $|u_1|, |u_2|, \dots, |u_n|, |v_0|, |v_1|, \dots, |v_n|$.

Theorem

Let $u_1, u_2, \ldots, u_n \in \mathbb{G}(A, I) \setminus \{1\}$, $v_0, v_1, \ldots, v_n \in \mathbb{G}(A, I)$ and let $x_1, \ldots, x_n$ be variables ranging over $\mathbb{N}$. Then, the set of solutions of the exponent equation

$$v_0 u_1^{x_1} v_1 u_2^{x_2} v_2 \cdots u_n^{x_n} v_n = 1$$

is semilinear. Moreover, if there is a solution, then there is a solution where the x_i are exponential in n and $|u_1|, |u_2|, \ldots, |u_n|, |v_0|, |v_1|, \ldots, |v_n|$.

Preprocessing:

- Make sure $u_i^{x_i}$ is reduced: "cyclically reduce" each u_i.
- If u_i not connected with $u_i = u_{i,1} u_{i,2}$, $u_{i,1} I u_{i,2}$, then replace $u_i^{x_i}$ with $u_{i,1}^{x_i} u_{i,2}^{x_i}$.

- Consider $v_0 \cdot u_1^{x_1} \cdot v_1 \cdot u_2^{x_2} \cdot v_2 \cdots u_n^{x_n} \cdot v_n = 1$
- By preprocessing, all factors $u_1^{x_1}, u_2^{x_2}, \ldots, u_n^{x_n}, v_0, \ldots, v_n$ are irreducible

- Consider $v_0 \cdot u_1^{x_1} \cdot v_1 \cdot u_2^{x_2} \cdot v_2 \cdots u_n^{x_n} \cdot v_n = 1$
- By preprocessing, all factors $u_1^{x_1}, u_2^{x_2}, \ldots, u_n^{x_n}, v_0, \ldots, v_n$ are irreducible
- Apply exponential refinement to obtain I-freely reducible sequence.

- Consider $v_0 \cdot u_1^{x_1} \cdot v_1 \cdot u_2^{x_2} \cdot v_2 \cdots u_n^{x_n} \cdot v_n = 1$
- By preprocessing, all factors $u_1^{x_1}, u_2^{x_2}, \ldots, u_n^{x_n}, v_0, \ldots, v_n$ are irreducible
- Apply exponential refinement to obtain I-freely reducible sequence.
- Consider all possible refinements and all possible reduction sequences.

- Consider $v_0 \cdot u_1^{x_1} \cdot v_1 \cdot u_2^{x_2} \cdot v_2 \cdots u_n^{x_n} \cdot v_n = 1$
- By preprocessing, all factors $u_1^{x_1}, u_2^{x_2}, \ldots, u_n^{x_n}, v_0, \ldots, v_n$ are irreducible
- Apply exponential refinement to obtain I-freely reducible sequence.
- Consider all possible refinements and all possible reduction sequences.
- We obtain a disjunction of statements:
  - (a) $u_i^{x_i} = y_{i,1} \cdots y_{i,k_i}$ $(1 \leqslant i \leqslant n)$
  - (b) $v_i = z_{i,1} \cdots z_{i,l_i}$ $(0 \leqslant i \leqslant n)$
  - (c) $y_{i,j} I y_{k,l}$ for all $(i,j,k,l) \in J_1$
  - (d) $y_{i,j} I z_{k,l}$ for all $(i,j,k,l) \in J_2$
  - (e) $z_{i,j} I z_{k,l}$ for all $(i,j,k,l) \in J_3$
  - (f) $y_{i,j} = y_{k,l}^{-1}$ for all $(i,j,k,l) \in M_1$
  - (g) $y_{i,j} = z_{k,l}^{-1}$ for all $(i,j,k,l) \in M_2$
  - (h) $z_{i,j} = z_{k,l}^{-1}$ for all $(i,j,k,l) \in M_3$

- Consider $v_0 \cdot u_1^{x_1} \cdot v_1 \cdot u_2^{x_2} \cdot v_2 \cdots u_n^{x_n} \cdot v_n = 1$
- By preprocessing, all factors $u_1^{x_1}, u_2^{x_2}, \ldots, u_n^{x_n}, v_0, \ldots, v_n$ are irreducible
- Apply exponential refinement to obtain I-freely reducible sequence.
- Consider all possible refinements and all possible reduction sequences.
- We obtain a disjunction of statements:
  - (a) $u_i^{x_i} = y_{i,1} \cdots y_{i,k_i}$ $(1 \leqslant i \leqslant n)$
  - (b) $v_i = z_{i,1} \cdots z_{i,l_i}$ $(0 \leqslant i \leqslant n)$
  - (c) $y_{i,j} I y_{k,l}$ for all $(i,j,k,l) \in J_1$
  - (d) $y_{i,j} I z_{k,l}$ for all $(i,j,k,l) \in J_2$
  - (e) $z_{i,j} I z_{k,l}$ for all $(i,j,k,l) \in J_3$
  - (f) $y_{i,j} = y_{k,l}^{-1}$ for all $(i,j,k,l) \in M_1$
  - (g) $y_{i,j} = z_{k,l}^{-1}$ for all $(i,j,k,l) \in M_2$
  - (h) $z_{i,j} = z_{k,l}^{-1}$ for all $(i,j,k,l) \in M_3$
- Replace $z_{k,l}$ by concrete traces.

- Consider $v_0 \cdot u_1^{x_1} \cdot v_1 \cdot u_2^{x_2} \cdot v_2 \cdots u_n^{x_n} \cdot v_n = 1$
- By preprocessing, all factors $u_1^{x_1}, u_2^{x_2}, \ldots, u_n^{x_n}, v_0, \ldots, v_n$ are irreducible
- Apply exponential refinement to obtain I-freely reducible sequence.
- Consider all possible refinements and all possible reduction sequences.
- We obtain a disjunction of statements:
  - (a) $u_i^{x_i} = y_{i,1} \cdots y_{i,k_i}$ $(1 \leqslant i \leqslant n)$
  - (b) $v_i = z_{i,1} \cdots z_{i,l_i}$ $(0 \leqslant i \leqslant n)$
  - (c) $y_{i,j} I y_{k,l}$ for all $(i,j,k,l) \in J_1$
  - (d) $y_{i,j} I z_{k,l}$ for all $(i,j,k,l) \in J_2$
  - (e) $z_{i,j} I z_{k,l}$ for all $(i,j,k,l) \in J_3$
  - (f) $y_{i,j} = y_{k,l}^{-1}$ for all $(i,j,k,l) \in M_1$
  - (g) $y_{i,j} = z_{k,l}^{-1}$ for all $(i,j,k,l) \in M_2$
  - (h) $z_{i,j} = z_{k,l}^{-1}$ for all $(i,j,k,l) \in M_3$
- Replace $z_{k,l}$ by concrete traces.

- Consider $v_0 \cdot u_1^{x_1} \cdot v_1 \cdot u_2^{x_2} \cdot v_2 \cdots u_n^{x_n} \cdot v_n = 1$
- By preprocessing, all factors $u_1^{x_1}, u_2^{x_2}, \ldots, u_n^{x_n}, v_0, \ldots, v_n$ are irreducible
- Apply exponential refinement to obtain I-freely reducible sequence.
- Consider all possible refinements and all possible reduction sequences.
- We obtain a disjunction of statements:
  - (a) $u_i^{x_i} = y_{i,1} \cdots y_{i,k_i}$ $(1 \leqslant i \leqslant n)$
  - (b) $v_i = z_{i,1} \cdots z_{i,l_i}$ $(0 \leqslant i \leqslant n)$
  - (c) $y_{i,j} I y_{k,l}$ for all $(i,j,k,l) \in J_1$
  - (d) $y_{i,j} I z_{k,l}$ for all $(i,j,k,l) \in J_2$
  - (e) $z_{i,j} I z_{k,l}$ for all $(i,j,k,l) \in J_3$
  - (f) $y_{i,j} = y_{k,l}^{-1}$ for all $(i,j,k,l) \in M_1$
  - (g) $y_{i,j} = z_{k,l}^{-1}$ for all $(i,j,k,l) \in M_2$
  - (h) $z_{i,j} = z_{k,l}^{-1}$ for all $(i,j,k,l) \in M_3$
- Replace $z_{k,l}$ by concrete traces.

- Consider $v_0 \cdot u_1^{x_1} \cdot v_1 \cdot u_2^{x_2} \cdot v_2 \cdots u_n^{x_n} \cdot v_n = 1$
- By preprocessing, all factors $u_1^{x_1}, u_2^{x_2}, \ldots, u_n^{x_n}, v_0, \ldots, v_n$ are irreducible
- Apply exponential refinement to obtain I-freely reducible sequence.
- Consider all possible refinements and all possible reduction sequences.
- We obtain a disjunction of statements:
  - (a) $u_i^{x_i} = y_{i,1} \cdots y_{i,k_i}$ $(1 \leqslant i \leqslant n)$

    (f) $y_{i,j} = y_{k,l}^{-1}$ for all $(i,j,k,l) \in M_1$
  - (c) $y_{i,j} I y_{k,l}$ for all $(i,j,k,l) \in J_1$
  - (d) $y_{i,j} I z_{k,l}$ for all $(i,j,k,l) \in J_2$
- Replace $z_{k,l}$ by concrete traces.

- Consider $v_0 \cdot u_1^{x_1} \cdot v_1 \cdot u_2^{x_2} \cdot v_2 \cdots u_n^{x_n} \cdot v_n = 1$
- By preprocessing, all factors $u_1^{x_1}, u_2^{x_2}, \ldots, u_n^{x_n}, v_0, \ldots, v_n$ are irreducible
- Apply exponential refinement to obtain I-freely reducible sequence.
- Consider all possible refinements and all possible reduction sequences.
- We obtain a disjunction of statements:
  - (a) $u_i^{x_i} = y_{i,1} \cdots y_{i,k_i}$ $(1 \leqslant i \leqslant n)$

    (f) $y_{i,j} = y_{k,l}^{-1}$ for all $(i,j,k,l) \in M_1$
  - (c) $y_{i,j} I y_{k,l}$ for all $(i,j,k,l) \in J_1$
  - (d) $y_{i,j} I z_{k,l}$ for all $(i,j,k,l) \in J_2$
- Replace $z_{k,l}$ by concrete traces.
- Replace $u_i^{x_i} = y_{i,1} \cdots y_{i,k_i}$. For some $x_{i,j} > 0$:
  - $x_i = c_i + \sum_{j \in K_i} x_{i,j}$ for all $1 \leqslant i \leqslant n$,
  - $y_{i,j} = p_{i,j} u_i^{x_{i,j}} s_{i,j}$ for all $1 \leqslant i \leqslant n$, $j \in K_i$,
  - $y_{i,j} = p_{i,j} s_{i,j}$ for all $1 \leqslant i \leqslant n$, $j \in [1, k_i] \setminus K_i$.

- Consider $v_0 \cdot u_1^{x_1} \cdot v_1 \cdot u_2^{x_2} \cdot v_2 \cdots u_n^{x_n} \cdot v_n = 1$
- By preprocessing, all factors $u_1^{x_1}, u_2^{x_2}, \ldots, u_n^{x_n}, v_0, \ldots, v_n$ are irreducible
- Apply exponential refinement to obtain I-freely reducible sequence.
- Consider all possible refinements and all possible reduction sequences.
- We obtain a disjunction of statements:

(f) $y_{i,j} = y_{k,l}^{-1}$ for all $(i,j,k,l) \in M_1$

(c) $y_{i,j} I y_{k,l}$ for all $(i,j,k,l) \in J_1$

(d) $y_{i,j} I z_{k,l}$ for all $(i,j,k,l) \in J_2$

- Replace $z_{k,l}$ by concrete traces.
- Replace $u_i^{x_i} = y_{i,1} \cdots y_{i,k_i}$. For some $x_{i,j} > 0$:
  - $x_i = c_i + \sum_{j \in K_i} x_{i,j}$ for all $1 \leqslant i \leqslant n$,
  - $y_{i,j} = p_{i,j} u_i^{x_{i,j}} s_{i,j}$ for all $1 \leqslant i \leqslant n$, $j \in K_i$,
  - $y_{i,j} = p_{i,j} s_{i,j}$ for all $1 \leqslant i \leqslant n$, $j \in [1, k_i] \backslash K_i$.
- Now we know alphabet of $y_{i,j}$

- Consider $v_0 \cdot u_1^{x_1} \cdot v_1 \cdot u_2^{x_2} \cdot v_2 \cdots u_n^{x_n} \cdot v_n = 1$
- By preprocessing, all factors $u_1^{x_1}, u_2^{x_2}, \ldots, u_n^{x_n}, v_0, \ldots, v_n$ are irreducible
- Apply exponential refinement to obtain I-freely reducible sequence.
- Consider all possible refinements and all possible reduction sequences.
- We obtain a disjunction of statements:

(f) $y_{i,j} = y_{k,l}^{-1}$ for all $(i, j, k, l) \in M_1$

- Replace $z_{k,l}$ by concrete traces.
- Replace $u_i^{x_i} = y_{i,1} \cdots y_{i,k_i}$. For some $x_{i,j} > 0$:
  - $x_i = c_i + \sum_{j \in K_i} x_{i,j}$ for all $1 \leqslant i \leqslant n$,
  - $y_{i,j} = p_{i,j} u_i^{x_{i,j}} s_{i,j}$ for all $1 \leqslant i \leqslant n$, $j \in K_i$,
  - $y_{i,j} = p_{i,j} s_{i,j}$ for all $1 \leqslant i \leqslant n$, $j \in [1, k_i] \backslash K_i$.
- Now we know alphabet of $y_{i,j}$

- The only remaining statements are of the form:
  - a') $x_i = c_i + \sum_{j \in K_i'} x_{i,j}$ for all $1 \leqslant i \leqslant n$, and
  - b') $p_{i,j} u_i^{x_{i,j}} s_{i,j} = s_{k,l}^{-1} (u_k^{-1})^{x_{k,l}} p_{k,l}^{-1}$ for all $(i,j,k,l) \in M$.

- The only remaining statements are of the form:
  - a' $x_i = c_i + \sum_{j \in K_i'} x_{i,j}$ for all $1 \leqslant i \leqslant n$, and
  - b' $p_{i,j} u_i^{x_{i,j}} s_{i,j} = s_{k,l}^{-1} (u_k^{-1})^{x_{k,l}} p_{k,l}^{-1}$ for all $(i,j,k,l) \in M$.
- Now we apply the fact that sets

$$L(p, u, s, q, v, t) := \{(x, y) \in \mathbb{N} \times \mathbb{N} \mid pu^x s = qv^y t\}$$

are semilinear (with small representations).

- The only remaining statements are of the form:
  - (a') $x_i = c_i + \sum_{j \in K'_i} x_{i,j}$ for all $1 \leqslant i \leqslant n$, and
  - (b') $p_{i,j} u_i^{x_{i,j}} s_{i,j} = s_{k,l}^{-1} (u_k^{-1})^{x_{k,l}} p_{k,l}^{-1}$ for all $(i,j,k,l) \in M$.
- Now we apply the fact that sets

$$L(p,u,s,q,v,t) := \{(x,y) \in \mathbb{N} \times \mathbb{N} \mid pu^x s = qv^y t\}$$

  are semilinear (with small representations).
- Replace equations (a') by linear diophantine equations.

- The only remaining statements are of the form:
  - a') $x_i = c_i + \sum_{j \in K'_i} x_{i,j}$ for all $1 \leqslant i \leqslant n$, and
  - b') $p_{i,j} u_i^{x_{i,j}} s_{i,j} = s_{k,l}^{-1} (u_k^{-1})^{x_{k,l}} p_{k,l}^{-1}$ for all $(i,j,k,l) \in M$.
- Now we apply the fact that sets

$$L(p,u,s,q,v,t) := \{(x,y) \in \mathbb{N} \times \mathbb{N} \mid pu^x s = qv^y t\}$$

  are semilinear (with small representations).
- Replace equations (a') by linear diophantine equations.
- Coefficients are linear in the length of $p_{i,j}, s_{i,j}$, hence exponential in n.

- The only remaining statements are of the form:
  - (a') $x_i = c_i + \sum_{j \in K_i'} x_{i,j}$ for all $1 \leqslant i \leqslant n$, and
  - (b') $p_{i,j} u_i^{x_{i,j}} s_{i,j} = s_{k,l}^{-1} (u_k^{-1})^{x_{k,l}} p_{k,l}^{-1}$ for all $(i,j,k,l) \in M$.
- Now we apply the fact that sets

$$L(p, u, s, q, v, t) := \{(x, y) \in \mathbb{N} \times \mathbb{N} \mid pu^x s = qv^y t\}$$

  are semilinear (with small representations).
- Replace equations (a') by linear diophantine equations.
- Coefficients are linear in the length of $p_{i,j}, s_{i,j}$, hence exponential in n.
- This yields small enough linear equation system for the x_i.
- Well-known result of von zur Gathen and Sieveking yields a small solution.

Transfer results

Theorem

The class of groups with knapsack in NP *is closed under*

- *Taking finite extensions*
- *HNN-extensions over finite associated subgroups*
- *Amalgamated products with finite identified groups*

Transfer results

Theorem

The class of groups with knapsack in NP *is closed under*

- *Taking finite extensions*
- *HNN-extensions over finite associated subgroups*
- *Amalgamated products with finite identified groups*

For finite extensions:

- Guess cosets of $v_0 u_1^{x_1} v_1 \cdots u_i^{x_i} v_i$ for all $i \in [1, n]$

Transfer results

Theorem

The class of groups with knapsack in NP *is closed under*

- *Taking finite extensions*
- *HNN-extensions over finite associated subgroups*
- *Amalgamated products with finite identified groups*

For finite extensions:

- Guess cosets of $v_0 u_1^{x_1} v_1 \cdots u_i^{x_i} v_i$ for all $i \in [1, n]$
- Create modified instance that has solution iff instance has solution with these cosets

Transfer results

Theorem

The class of groups with knapsack in NP *is closed under*

- *Taking finite extensions*
- *HNN-extensions over finite associated subgroups*
- *Amalgamated products with finite identified groups*

For finite extensions:

- Guess cosets of $v_0 u_1^{x_1} v_1 \cdots u_i^{x_i} v_i$ for all $i \in [1, n]$
- Create modified instance that has solution iff instance has solution with these cosets

For other transformations:

- Saturation procedure that successively adds transitions to automaton

Transfer results

Theorem

The class of groups with knapsack in NP *is closed under*

- *Taking finite extensions*
- *HNN-extensions over finite associated subgroups*
- *Amalgamated products with finite identified groups*

For finite extensions:

- Guess cosets of $v_0 u_1^{x_1} v_1 \cdots u_i^{x_i} v_i$ for all $i \in [1, n]$
- Create modified instance that has solution iff instance has solution with these cosets

For other transformations:

- Saturation procedure that successively adds transitions to automaton
- Choose suitable class of automata such that adding transitions still leads to knapsack instances: knapsack automata.