Knapsack problems in hyperbolic groups

Andrey Nikolaev
(Stevens Institute)

GAGTA, May 2013

Based on joint work with A.Miasnikov and A.Ushakov

Non-commutative discrete optimization

Basic idea:

Take a classical algorithmic problem from computer science (traveling salesman, Post correspondence, knapsack, . . .) and translate it into group-theoretic setting.

Non-commutative discrete optimization

The classical subset sum problem (SSP):

Given $a_1, \ldots, a_k, a \in \mathbb{Z}$ decide if

$$\varepsilon_1 a_1 + \ldots + \varepsilon_k a_k = a$$

for some $\varepsilon_1, \ldots, \varepsilon_k \in \{0, 1\}$.

SSP for a group G:

Given $g_1, \ldots, g_k, g \in G$ decide if

$$g_1^{\varepsilon_1} \ldots g_k^{\varepsilon_k} = g$$

for some $\varepsilon_1, \ldots, \varepsilon_k \in \{0, 1\}$.

Elements in G are given as words in a fixed set of generators of G.

Non-commutative discrete optimization

The classical subset sum problem (SSP):

Given $a_1, \ldots, a_k, a \in \mathbb{Z}$ decide if

$$\varepsilon_1 a_1 + \ldots + \varepsilon_k a_k = a$$

for some $\varepsilon_1, \ldots, \varepsilon_k \in \{0, 1\}$.

SSP for a group G:

Given $g_1, \ldots, g_k, g \in G$ decide if

$$g_1^{\varepsilon_1} \ldots g_k^{\varepsilon_k} = g$$

for some $\varepsilon_1, \ldots, \varepsilon_k \in \{0, 1\}$.

Elements in G are given as words in a fixed set of generators of G.

Non-commutative discrete optimization

The classical subset sum problem (SSP):

Given $a_1, \ldots, a_k, a \in \mathbb{Z}$ decide if

$$\varepsilon_1 a_1 + \ldots + \varepsilon_k a_k = a$$

for some $\varepsilon_1, \ldots, \varepsilon_k \in \{0, 1\}$.

SSP for a group G:

Given $g_1, \ldots, g_k, g \in G$ decide if

$$g_1^{\varepsilon_1} \ldots g_k^{\varepsilon_k} = g$$

for some $\varepsilon_1, \ldots, \varepsilon_k \in \{0, 1\}$.

Elements in G are given as words in a fixed set of generators of G.

Non-commutative discrete optimization

In the classical (commutative) case, **SSP** is pseudo-polynomial.

Classical SSP

- If input is given in unary, **SSP** is in **P**,
- if input is given in binary, **SSP** is **NP**-complete.

The situation is quite more involved in non-commutative case.

In the classical (commutative) case, **SSP** is pseudo-polynomial.

Classical SSP

- If input is given in unary, **SSP** is in **P**,
- if input is given in binary, **SSP** is **NP**-complete.

The situation is quite more involved in non-commutative case.

Non-commutative discrete optimization

Group	Complexity	Why
Nilpotent	**P**	Poly growth
$\mathbb{Z} \wr \mathbb{Z}$	**NP**-complete	**ZOE**
Free metabelian	**NP**-complete	$\mathbb{Z} \wr \mathbb{Z}$
Thompson's F	**NP**-complete	$\mathbb{Z} \wr \mathbb{Z}$
$BS(1, p)$	**NP**-complete	Binary **SSP**($\mathbb{Z}$)
Hyperbolic	**P**	Later in the talk

Note that the **NP**-completeness is despite unary input.

Non-commutative discrete optimization

Group	Complexity	Why
Nilpotent	**P**	Poly growth
$\mathbb{Z} \wr \mathbb{Z}$	**NP**-complete	**ZOE**
Free metabelian	**NP**-complete	$\mathbb{Z} \wr \mathbb{Z}$
Thompson's F	**NP**-complete	$\mathbb{Z} \wr \mathbb{Z}$
$BS(1, p)$	**NP**-complete	Binary **SSP**($\mathbb{Z}$)
Hyperbolic	**P**	Later in the talk

Note that the **NP**-completeness is despite unary input.

Three principle Knapsack type (decision) problems in groups:

SSP subset sum,

KP knapsack,

SMP submonoid membership.

Variations of **SSP**, **KP**, **SMP**:

- search,
- optimization,
- bounded.

Three principle Knapsack type (decision) problems in groups:

SSP subset sum,

KP knapsack,

SMP submonoid membership.

Variations of **SSP**, **KP**, **SMP**:

- search,
- optimization,
- bounded.

Three principle Knapsack type (decision) problems in groups:

SSP subset sum,

KP knapsack,

SMP submonoid membership.

Variations of **SSP**, **KP**, **SMP**:

- search,
- optimization,
- bounded.

Knapsack problems in groups

Three principle Knapsack type (decision) problems in groups:

SSP subset sum,

KP knapsack,

SMP submonoid membership.

Variations of **SSP**, **KP**, **SMP**:

- search,
- optimization,
- bounded.

The knapsack problem (KP) for G:

Given $g_1, \ldots, g_k, g \in G$ decide if

$$g_1^{\varepsilon_1} \ldots g_k^{\varepsilon_k} = g$$

for some non-negative integers $\varepsilon_1, \ldots, \varepsilon_k$.

There are minor variations of this problem, for instance, integer **KP**, when ε_i are arbitrary integers. They are all similar, we omit them here.

The subset sum problem sometimes is called $0-1$ knapsack.

The knapsack problem in groups

The knapsack problem (KP) for G:

Given $g_1, \ldots, g_k, g \in G$ decide if

$$g_1^{\varepsilon_1} \ldots g_k^{\varepsilon_k} = g$$

for some non-negative integers $\varepsilon_1, \ldots, \varepsilon_k$.

There are minor variations of this problem, for instance, integer **KP**, when ε_i are arbitrary integers. They are all similar, we omit them here.

The subset sum problem sometimes is called $0-1$ knapsack.

The knapsack problems in groups is closely related to the big powers method, which appeared long before any complexity considerations (Baumslag, 1962).

Submonoid membership problem (SMP):

Given a finite set $A = \{g_1, \ldots, g_k, g\}$ of elements of G decide if g belongs to the submonoid generated by A, i.e., if $g = g_{i_1}, \ldots, g_{i_s}$ for some $g_{i_j} \in A$.

If the set A is closed under inversion then we have the subgroup membership problem in G.

It makes sense to consider the bounded versions of **KP** and **SMP**, they are always decidable in groups with decidable word problem.

The bounded knapsack problem (BKP) for G:

decide, when given $g_1, \ldots, g_k, g \in G$ and $1^m \in \mathbb{N}$, if $g =_G g_1^{\varepsilon_1} \ldots g_k^{\varepsilon_k}$ for some $\varepsilon_i \in \{0, 1, \ldots, m\}$.

BKP is **P**-time equivalent to **SSP** in G.

Bounded variations

It makes sense to consider the bounded versions of **KP** and **SMP**, they are always decidable in groups with decidable word problem.

The bounded knapsack problem (BKP) for G:

decide, when given $g_1, \ldots, g_k, g \in G$ and $1^m \in \mathbb{N}$, if $g =_G g_1^{\varepsilon_1} \ldots g_k^{\varepsilon_k}$ for some $\varepsilon_i \in \{0, 1, \ldots, m\}$.

BKP is **P**-time equivalent to **SSP** in G.

Bounded variations

Bounded submonoid membership problem (BSMP) for G:

Given $g_1, \ldots g_k, g \in G$ and $1^m \in \mathbb{N}$ (in unary) decide if g is equal in G to a product of the form $g = g_{i_1} \cdots g_{i_s}$, where $g_{i_1}, \ldots, g_{i_s} \in \{g_1, \ldots, g_k\}$ and $s \leq m$.

Hyperbolic groups

Theorem

Let G be a hyperbolic group then all the problems $\mathbf{SSP}(G), \mathbf{KP}(G), \mathbf{BSMP}(G)$, as well as their search and optimization versions are in $\mathbf{P}$.

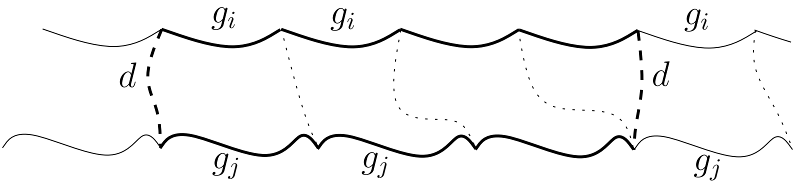
$\mathbf{KP}(G) \in \mathbf{P}$, sketch of proof

Draw equality

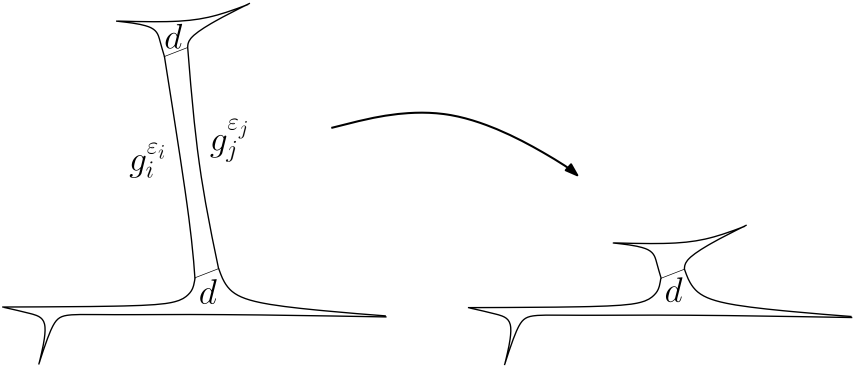
$$g_1^{\varepsilon_1} \dots g_k^{\varepsilon_k} = g$$

in the Cayley graph. If one of ε_i's is large, we can cut some powers out.

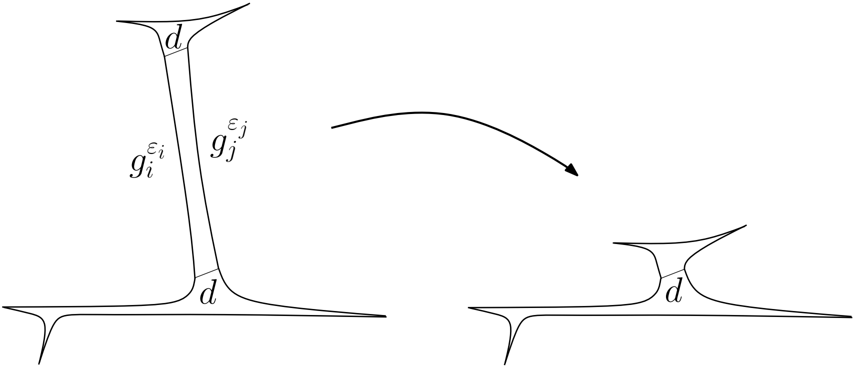
$\mathbf{KP}(G) \in \mathbf{P}$, sketch of proof



$\mathbf{KP}(G) \in \mathbf{P}$, sketch of proof



Now we only need to solve $\mathbf{SSP}(G)$.



Now we only need to solve $\mathbf{SSP}(G)$.

$\mathbf{SSP}(G) \in \mathbf{P}$, sketch of proof

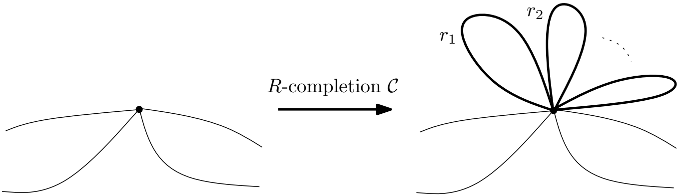
$w_1, w_2, \ldots, w_k, w$ is a positive instance of **SSP** iff a word equal to 1 in G is readable in the following graph:

$$
\begin{tikzcd}
\alpha\bullet \arrow[r, bend left, "w_1"] \arrow[r, "\varepsilon"'] & \bullet \arrow[r, bend left, "w_2"] \arrow[r, "\varepsilon"'] & \bullet & \cdots & \bullet \arrow[r, bend left, "w_k"] \arrow[r, "\varepsilon"'] & \bullet \arrow[r, "w^{-1}"'] & \bullet\omega
\end{tikzcd}
$$

To recognize whether a word equal to 1 in G is readable, we perform two operations, so called *R-completion* and *folding*.

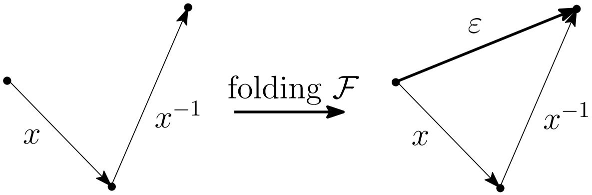
$\mathbf{SSP}(G) \in \mathbf{P}$, sketch of proof

For a symmetrized presentation $\langle X \mid R\rangle$ and a graph Γ labeled by X, at each vertex of Γ we add a loop labeled by r, for each $r \in R$:



$\mathbf{SSP}(G) \in \mathbf{P}$, sketch of proof

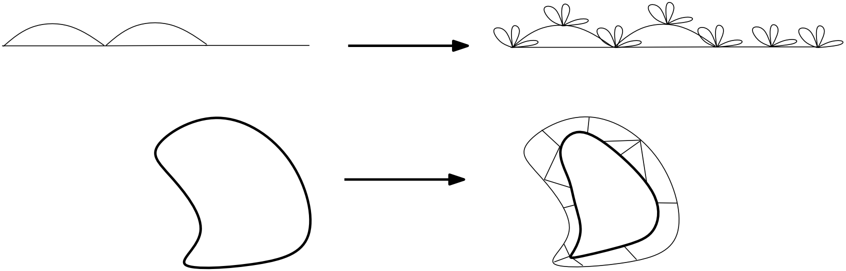
For each “foldable” pair of consecutive edges we add a new edge:



“Foldable” pairs:

$s_1 \xrightarrow{x} s_2 \xrightarrow{x^{-1}} s_3$	$s_1 \xrightarrow{\varepsilon} s_3$
$s_1 \xrightarrow{x} s_2 \xrightarrow{\varepsilon} s_3$	$s_1 \xrightarrow{x} s_3$
$s_1 \xrightarrow{\varepsilon} s_2 \xrightarrow{x} s_3$	$s_1 \xrightarrow{x} s_3$
$s_1 \xrightarrow{\varepsilon} s_2 \xrightarrow{\varepsilon} s_3$	$s_1 \xrightarrow{\varepsilon} s_3$.

One application of completion and folding corresponds to "peeling off" one layer of cells in van Kampen diagram:



$\mathbf{SSP}(G) \in \mathbf{P}$, sketch of proof

Lemma

Let $\langle X \mid R \rangle$ be a finite presentation of a hyperbolic group G. Let Γ be an acyclic automaton over $X \cup X^{-1}$ with at most m nontrivially labeled edges. Then $1 \in \overline{L(\Gamma)}$ if and only if $\mathcal{F}(\mathcal{C}^{O(\log m)}(\Gamma))$ contains an edge $\alpha \xrightarrow{\varepsilon} \omega$.

Proof: in a hyperbolic group G, the depth of van Kampen diagrams is *logarithmic* in perimeter (Druțu 2001).

SSP$(G) \in$ **P**, sketch of proof

Lemma

Let $\langle X \mid R \rangle$ be a finite presentation of a hyperbolic group G. Let Γ be an acyclic automaton over $X \cup X^{-1}$ with at most m nontrivially labeled edges. Then $1 \in \overline{L(\Gamma)}$ if and only if $\mathcal{F}(\mathcal{C}^{O(\log m)}(\Gamma))$ contains an edge $\alpha \xrightarrow{\varepsilon} \omega$.

Proof: in a hyperbolic group G, the depth of van Kampen diagrams is *logarithmic* in perimeter (Druțu 2001).

$\mathbf{SSP}(G) \in \mathbf{P}$, the algorithm

To solve **SSP** in a hyperbolic group G, given words $w_1, w_2, \ldots, w$, we construct the graph Γ as above

$$\begin{tikzcd} \alpha \arrow[r, "\varepsilon"'] \arrow[r, bend left, "w_1"] & \bullet \arrow[r, "\varepsilon"'] \arrow[r, bend left, "w_2"] & \bullet & \cdots & \bullet \arrow[r, "\varepsilon"'] \arrow[r, bend left, "w_k"] & \bullet \arrow[r, "w^{-1}"'] & \omega \end{tikzcd}$$

Figure : Graph $\Gamma = \Gamma(w_1, \ldots, w_k, w)$.

and apply $O(\log(|w| + \sum |w_i|))$ R-completions and then the (non-Stallings) folding to construct the graph $\mathcal{F}(\mathcal{C}^{O(\log m)}(\Gamma))$:

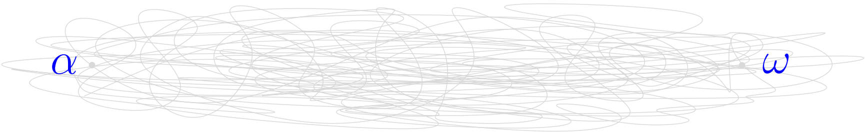


Figure : Graph $\mathcal{F}(\mathcal{C}^{O(\log(|w|+\sum |w_i|))}(\Gamma))$

and check whether the resulting graph contains the edge $\alpha \xrightarrow{\varepsilon} \omega$.

and apply $O(\log(|w| + \sum |w_i|))$ R-completions and then the (non-Stallings) folding to construct the graph $\mathcal{F}(\mathcal{C}^{O(\log m)}(\Gamma))$:

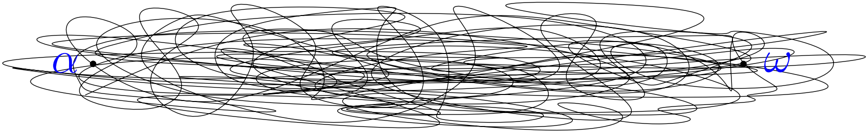


Figure : Graph $\mathcal{F}(\mathcal{C}^{O(\log(|w|+\sum |w_i|))}(\Gamma))$

and check whether the resulting graph contains the edge $\alpha \xrightarrow{\varepsilon} \omega$.

The same argument can be used to show that search and optimization variations of **SSP**, **KP** are in **P** for a hyperbolic group G.

The big finish

The same argument can be also used to show that **BSMP**(G) (together with its search and optimization variations) for a hyperbolic group is in **P**.

Surprise

The bounded **SMP** is polynomial time decidable in any hyperbolic group, while there are hyperbolic groups with undecidable **SMP**.

The big finish

The same argument can be also used to show that **BSMP**(G) (together with its search and optimization variations) for a hyperbolic group is in **P**.

Surprise

The bounded **SMP** is polynomial time decidable in any hyperbolic group, while there are hyperbolic groups with undecidable **SMP**.

The big finish

The same argument can be also used to show that **BSMP**(G) (together with its search and optimization variations) for a hyperbolic group is in **P**.

Surprise

The bounded **SMP** is polynomial time decidable in any hyperbolic group, while there are hyperbolic groups with undecidable **SMP**.