Parameterized complexity of the word search problem in the Baumslag–Gersten group

Andrey Nikolaev

Stevens Institute of Technology

Omsk Branch of Mathematical Center in Akademgorodok
June 26, 2020

Joint work with A. Miasnikov

Plan:

– Parameterized complexity. Fixed parameter tractability.

– Word problem, word search problem in groups.

– Fixed parameter tractability of the word problem.

– Word search problem in Baumslag–Solitar group.

– Word search problem in Baumslag–Gersten group.

– Further developments.

Parameterized complexity

Downey, Fellows 1995: study complexity as a function of the size of the input, n, and a parameter of input or output, k.

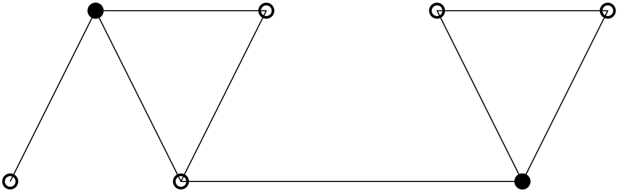
Example 1. Given a graph on n vertices, is there a dominating set of size k?
(Set $D \subseteq V$ s.t. every vertex is in D or has a neighbor in D.)

Can be solved in time $O(n^{1+k})$.

Parameterized complexity

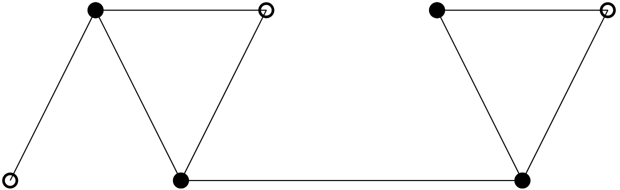
Downey, Fellows 1995: study complexity as a function of the size of the input, n, and a parameter of input or output, k.

Example 1. Given a graph on n vertices, is there a dominating set of size k?
(Set $D \subseteq V$ s.t. every vertex is in D or has a neighbor in D.)



Can be solved in time $O(n^{1+k})$.

Example 2. Given a graph on n vertices, is there a vertex cover of size k?
(Set $K \subseteq V$ s.t. every edge has a vertex in K.)



Can be solved in time $O(1.2738^k + kn)$.

Parameterized complexity

VERTEX COVER, solvable in time $O(1.2738^k + kn)$.
DOMINATING SET, solvable in time $O(n^{1+k})$.

Definition. A computational problem is fixed parameter tractable if there is an algorithm that solves the problem on input of size n with parameter k in time $f(k)n^c$, where $f : \mathbb{N} \to \mathbb{N}$ is computable and c is a constant.
Class of such problems is denoted FPT.

VERTEX COVER $\in$ FPT.
DOMINATING SET $\notin$ FPT (unless certain complexity hierarchy collapses).
Both are NP-complete.

VERTEX COVER, solvable in time $O(1.2738^k + kn)$.
DOMINATING SET, solvable in time $O(n^{1+k})$.

Definition. A computational problem is fixed parameter tractable if there is an algorithm that solves the problem on input of size n with parameter k in time $f(k)n^c$, where $f : \mathrm{N} \to \mathrm{N}$ is computable and c is a constant.
Class of such problems is denoted FPT.

VERTEX COVER $\in$ FPT.
DOMINATING SET $\notin$ FPT (unless certain complexity hierarchy collapses).
Both are NP-complete.

Vertex Cover, solvable in time $O(1.2738^k + kn)$.
Dominating Set, solvable in time $O(n^{1+k})$.

Definition. A computational problem is fixed parameter tractable if there is an algorithm that solves the problem on input of size n with parameter k in time $f(k)n^c$, where $f : \mathrm{N} \rightarrow \mathrm{N}$ is computable and c is a constant.
Class of such problems is denoted FPT.

Vertex Cover $\in$ FPT.
Dominating Set $\notin$ FPT (unless certain complexity hierarchy collapses).
Both are NP-complete.

Parameterized complexity allows:

– Finer classification of hard problems.

– Distinction between laborious and complicated problems.

Example. List all binary words of length n.

This is not a polynomial time solvable problem. If the size of answer $k = 2^n$ is taken as length of input, then the time is kn.

Parameterized complexity

Parameterized complexity allows:
– Finer classification of hard problems.
– Distinction between laborious and complicated problems.
Example. List all binary words of length n.
This is not a polynomial time solvable problem. If the size of answer $k = 2^n$ is taken as length of input, then the time is kn.

Parameterized complexity

Parameterized complexity allows:
– Finer classification of hard problems.
– Distinction between laborious and complicated problems.
Example. List all binary words of length n.
This is not a polynomial time solvable problem. If the size of answer $k = 2^n$ is taken as length of input, then the time is kn.

Parameterized complexity

Parameterized complexity allows:
– Finer classification of hard problems.
– Distinction between laborious and complicated problems.
Example. List all binary words of length n.
This is not a polynomial time solvable problem. If the size of answer $k = 2^n$ is taken as length of input, then the time is kn.

Parameterized complexity

Parameterized complexity allows:
– Finer classification of hard problems.
– Distinction between laborious and complicated problems.

Example. List all binary words of length n.

This is not a polynomial time solvable problem. If the size of answer $k = 2^n$ is taken as length of input, then the time is kn.

Parameterized complexity

Parameterized complexity allows:
– Finer classification of hard problems.
– Distinction between laborious and complicated problems.
Example. List all binary words of length n.
This is not a polynomial time solvable problem. If the size of answer $k = 2^n$ is taken as length of input, then the time is kn.

Word problem

Word problem in a group G with presentation $\langle X \mid R \rangle$:

Given a group word $w \in (X \cup X^{-1})^*$, decide if $w = 1$ in G.

Equivalently: decide if there are $r_1, \ldots, r_k \in R \cup R^{-1}$ and $u_1, \ldots, u_k \in (X \cup X^{-1})^*$ s.t.

$$\begin{aligned} w &= (u_1^{-1} r_1 u_1) \cdot (u_2^{-1} r_2 u_2) \cdots (u_k^{-1} r_k u_k) = \\ &= r_1^{u_1} \cdot r_2^{u_2} \cdots r_k^{u_k} \end{aligned}$$

after free cancellation.

Word problem

Word problem in a group G with presentation $\langle X \mid R \rangle$:

Given a group word $w \in (X \cup X^{-1})^*$, decide if $w = 1$ in G.

Equivalently: decide if there are $r_1, \ldots, r_k \in R \cup R^{-1}$ and $u_1, \ldots, u_k \in (X \cup X^{-1})^*$ s.t.

$$\begin{aligned} w &= (u_1^{-1} r_1 u_1) \cdot (u_2^{-1} r_2 u_2) \cdots (u_k^{-1} r_k u_k) = \\ &= r_1^{u_1} \cdot r_2^{u_2} \cdots r_k^{u_k} \end{aligned}$$

after free cancellation.

Word problem

Example. Consider abelian group $G = \langle a, b \mid a^{-1}b^{-1}ab \rangle$.
Input: $w = a^{-3}b^{-1}abab^{-2}ab^2$. In G, $w = 1$.
Indeed,

$$a^{-3}b^{-1}abab^{-2}ab^2 = a^{-2}(a^{-1}b^{-1}ab)a^2 \cdot (a^{-1}b^{-1}ab) \cdot b^{-1}(a^{-1}b^{-1}ab)b.$$

Let's make a picture!

Word problem

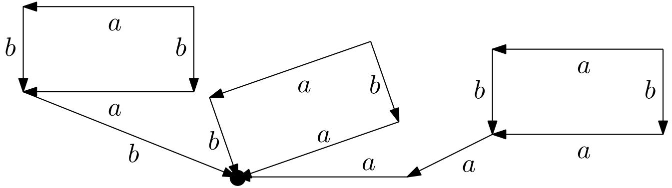
Example. Consider abelian group $G = \langle a, b \mid a^{-1}b^{-1}ab \rangle$.
Input: $w = a^{-3}b^{-1}abab^{-2}ab^2$. In G, $w = 1$.
Indeed,

$$a^{-3}b^{-1}abab^{-2}ab^2 = a^{-2}(a^{-1}b^{-1}ab)a^2 \cdot (a^{-1}b^{-1}ab) \cdot b^{-1}(a^{-1}b^{-1}ab)b.$$

Let's make a picture!

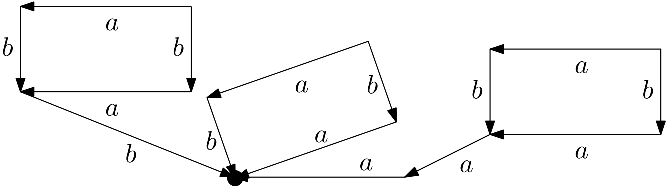
Word problem

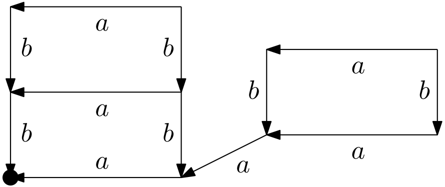
$G = \langle a, b \mid a^{-1}b^{-1}ab \rangle$.
$a^{-3}b^{-1}abab^{-2}ab^2 = a^{-2}(a^{-1}b^{-1}ab)a^2 \cdot (a^{-1}b^{-1}ab) \cdot b^{-1}(a^{-1}b^{-1}ab)b$.



Word problem

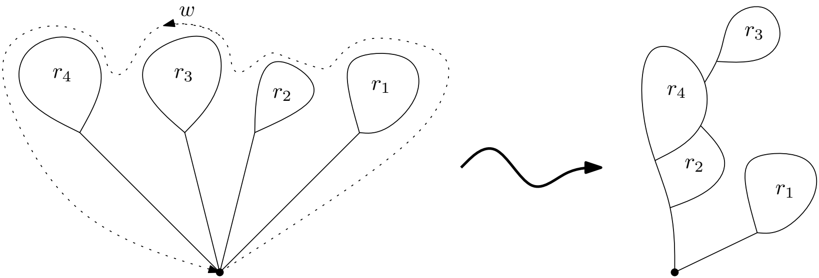
$G = \langle a, b \mid a^{-1}b^{-1}ab \rangle$.
$a^{-3}b^{-1}abab^{-2}ab^2 = a^{-2}(a^{-1}b^{-1}ab)a^2 \cdot (a^{-1}b^{-1}ab) \cdot b^{-1}(a^{-1}b^{-1}ab)b$.





Word problem

Van Kampen diagrams (disc diagrams):



Homotopically: a disc.
Homeomorphically: a tree made of discs.

Word search problem

Word search problem (*van Kampen diagram problem*) in a group G with presentation $\langle X \mid R \rangle$:

Given a group word $w \in (X \cup X^{-1})^*$ s.t. $w = 1$ in G, find $r_1, \ldots, r_k \in R \cup R^{-1}$ and $u_1, \ldots, u_k \in (X \cup X^{-1})^*$ s.t.

$$w = r_1^{u_1} \cdot r_2^{u_2} \cdots r_k^{u_k}$$

after free cancellation.
Equivalently, find a van Kampen diagram for w.

Intuitive viewpoint: this is the problem of "finding proof" that $w = 1$ in G.

Word search problem

Word search problem (*van Kampen diagram problem*) in a group G with presentation $\langle X \mid R \rangle$:

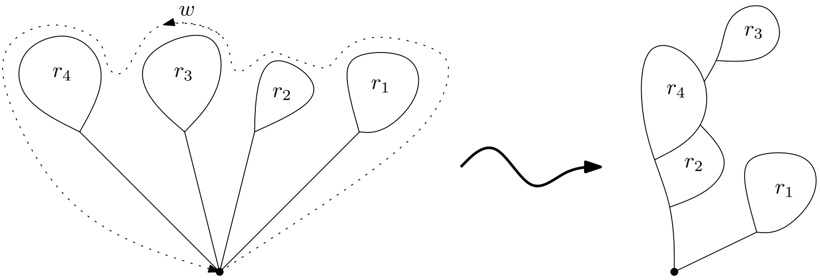
Given a group word $w \in (X \cup X^{-1})^*$ s.t. $w = 1$ in G, find $r_1, \ldots, r_k \in R \cup R^{-1}$ and $u_1, \ldots, u_k \in (X \cup X^{-1})^*$ s.t.

$$w = r_1^{u_1} \cdot r_2^{u_2} \cdots r_k^{u_k}$$

after free cancellation.
Equivalently, find a van Kampen diagram for w.

Intuitive viewpoint: this is the problem of "finding proof" that $w = 1$ in G.

Word search problem

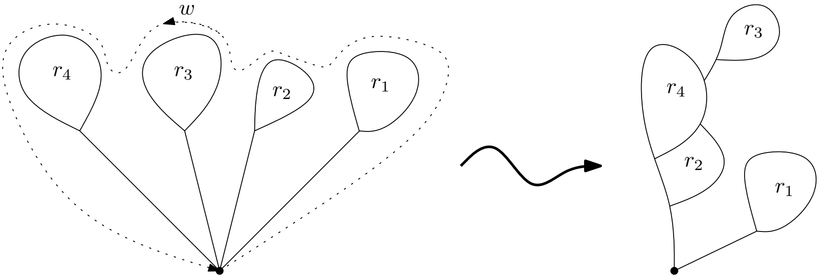


The minimal value k (area of the minimal van Kampen diagram) can serve as a parameter of the word (search) problem.
Length of the RHS in

$$w = r_1^{u_1} \cdot r_2^{u_2} \cdots r_k^{u_k}$$

is bounded by $C \cdot (n + k)^2$.

Word search problem



The minimal value k (area of the minimal van Kampen diagram) can serve as a parameter of the word (search) problem.
Length of the RHS in

$$w = r_1^{u_1} \cdot r_2^{u_2} \cdots r_k^{u_k}$$

is bounded by $C \cdot (n+k)^2$.

Dehn function

Dehn function of the presentation $\langle X \mid R\rangle$:

$$f(n) = \max\{k \mid w = 1, |w| \leq n\}.$$

Tells how large an area is required to show that $w = 1$ if $|w| \leq n$.

Effect of parameterization

There are finite presentations where the word problem is:
– undecidable (Novikov 1955, Boone 1958),
– NP-complete (Sapir 2002).

Theorem 1. In every finite presentation the word problem can be solved in time $f(k)n^2$, where n is the size of input, k is the area of a van Kampen diagram, and f depends on the presentation.

Effect of parameterization

There are finite presentations where the word problem is:
– undecidable (Novikov 1955, Boone 1958),
– NP-complete (Sapir 2002).

Theorem 1. In every finite presentation the word problem can be solved in time $f(k)n^2$, where n is the size of input, k is the area of a van Kampen diagram, and f depends on the presentation.

Effect of parameterization

Theorem 1. In every finite presentation the word problem can be solved in time $f(k)n^2$, where n is the size of input, k is the area of a van Kampen diagram, and f depends on the presentation.

Proof. – Fix k. Van Kampen diagram is a tree of $\leq k$ subdiagrams *homeomorphic to a disc*, of area $\leq k$ each.

– There are only finitely many, $g(k)$, van Kampen diagrams of area $\leq k$ homeomorphic to a disc.

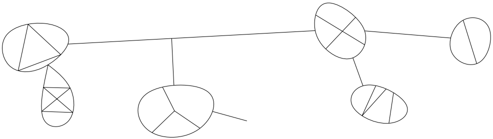
Algorithm: 1. Find a subword in w from the list of $g(k)$ boundary words of homeomorphic disc diagrams.

2. Remove the subword, perform free cancellation.

3. Repeat until whichever happens first: Step 1 fails, or the empty word is reached, or area runs out. □

Effect of parameterization

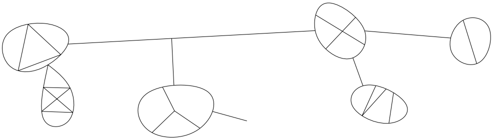
Theorem 1. In every finite presentation the word problem can be solved in time $f(k)n^2$, where n is the size of input, k is the area of a van Kampen diagram, and f depends on the presentation.
Proof. – Fix k. Van Kampen diagram is a tree of $\leq k$ subdiagrams *homeomorphic to a disc*, of area $\leq k$ each.



– There are only finitely many, $g(k)$, van Kampen diagrams of area $\leq k$ homeomorphic to a disc.
Algorithm: 1. Find a subword in w from the list of $g(k)$ boundary words of homeomorphic disc diagrams.
2. Remove the subword, perform free cancellation.
3. Repeat until whichever happens first: Step 1 fails, or the empty word is reached, or area runs out. □

Effect of parameterization

Theorem 1. In every finite presentation the word problem can be solved in time $f(k)n^2$, where n is the size of input, k is the area of a van Kampen diagram, and f depends on the presentation.
Proof. – Fix k. Van Kampen diagram is a tree of $\leq k$ subdiagrams *homeomorphic to a disc*, of area $\leq k$ each.

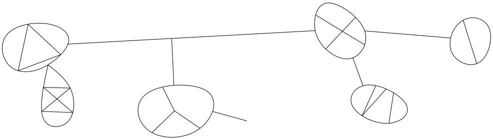


– There are only finitely many, $g(k)$, van Kampen diagrams of area $\leq k$ homeomorphic to a disc.
Algorithm: 1. Find a subword in w from the list of $g(k)$ boundary words of homeomorphic disc diagrams.
2. Remove the subword, perform free cancellation.
3. Repeat until whichever happens first: Step 1 fails, or the empty word is reached, or area runs out. □

Effect of parameterization

Theorem 1. In every finite presentation the word problem can be solved in time $f(k)n^2$, where n is the size of input, k is the area of a van Kampen diagram, and f depends on the presentation.

Proof. – Fix k. Van Kampen diagram is a tree of $\leq k$ subdiagrams *homeomorphic to a disc*, of area $\leq k$ each.



– There are only finitely many, $g(k)$, van Kampen diagrams of area $\leq k$ homeomorphic to a disc.

Algorithm: 1. Find a subword in w from the list of $g(k)$ boundary words of homeomorphic disc diagrams.

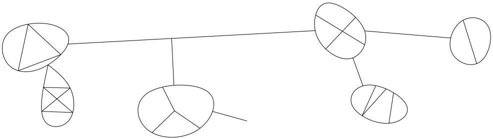
2. Remove the subword, perform free cancellation.

3. Repeat until whichever happens first: Step 1 fails, or the empty word is reached, or area runs out. □

Effect of parameterization

Theorem 1. In every finite presentation the word problem can be solved in time $f(k)n^2$, where n is the size of input, k is the area of a van Kampen diagram, and f depends on the presentation.

Proof. – Fix k. Van Kampen diagram is a tree of $\leq k$ subdiagrams *homeomorphic to a disc*, of area $\leq k$ each.



– There are only finitely many, $g(k)$, van Kampen diagrams of area $\leq k$ homeomorphic to a disc.

Algorithm: 1. Find a subword in w from the list of $g(k)$ boundary words of homeomorphic disc diagrams.

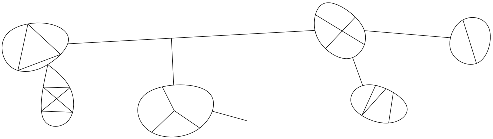
2. Remove the subword, perform free cancellation.

3. Repeat until whichever happens first: Step 1 fails, or the empty word is reached, or area runs out. □

Effect of parameterization

Theorem 1. In every finite presentation the word problem can be solved in time $f(k)n^2$, where n is the size of input, k is the area of a van Kampen diagram, and f depends on the presentation.

Proof. – Fix k. Van Kampen diagram is a tree of $\leq k$ subdiagrams *homeomorphic to a disc*, of area $\leq k$ each.



– There are only finitely many, $g(k)$, van Kampen diagrams of area $\leq k$ homeomorphic to a disc.

Algorithm: 1. Find a subword in w from the list of $g(k)$ boundary words of homeomorphic disc diagrams.

2. Remove the subword, perform free cancellation.

3. Repeat until whichever happens first: Step 1 fails, or the empty word is reached, or area runs out. $\square$

Effect of parameterization

Theorem 1. In every finite presentation the word problem can be solved in time $f(k)n^2$, where n is the size of input, k is the area of a van Kampen diagram, and f depends on the presentation.

Remark. $g(k)$, and therefore $f(k)$, is exponential even for $\langle a, b \mid a^{-1}b^{-1}ab \rangle$.

Therefore, the word problem and word search problem are FPT.

For specific groups, we can do better.

Effect of parameterization

Theorem 1. In every finite presentation the word problem can be solved in time $f(k)n^2$, where n is the size of input, k is the area of a van Kampen diagram, and f depends on the presentation.

Remark. $g(k)$, and therefore $f(k)$, is exponential even for $\langle a, b \mid a^{-1}b^{-1}ab \rangle$.
Therefore, the word problem and word search problem are FPT.
For specific groups, we can do better.

Baumslag–Solitar groups

Baumslag–Solitar group $BS(m,n) = \langle a, b \mid (a^m)^b = a^n \rangle$.
If $|m| \neq |n|$, then Dehn function of $BS(m,n)$ is exponential. For example, in $BS(1,2) = \langle a, b \mid a^b = a^2 \rangle$ the equality

$$b^{-N}ab^N \cdot a = a \cdot b^{-N}ab^N, \quad N = 1, 2, \ldots.$$

requires $O(2^N)$ cells.

The word search problem in $BS(1,2) = \langle a, b \mid a^b = a^2 \rangle$ requires exponential time.

This contrasts the simplicity of the algorithm to solve the word (search) problem.

$BS(1,2)$ can be viewed as an HNN-extension of $\langle a \rangle$ with a stable letter b and associated subgroups $\langle a \rangle$, $\langle a^2 \rangle$.

The algorithm follows Britton's lemma for HNN-extensions:

1. Look for a subword $b^{-1}a^n b$, replace with a^{2n}; or a subword $ba^{2n}b^{-1}$, replace with a^n.
2. That's it, there is no Step 2. Keep repeating Step 1.

The word search problem in $BS(1,2) = \langle a, b \mid a^b = a^2 \rangle$ requires exponential time.
This contrasts the simplicity of the algorithm to solve the word (search) problem.

$BS(1,2)$ can be viewed as an HNN-extension of $\langle a \rangle$ with a stable letter b and associated subgroups $\langle a \rangle$, $\langle a^2 \rangle$.

The algorithm follows Britton's lemma for HNN-extensions:
1. Look for a subword $b^{-1}a^n b$, replace with a^{2n}; or a subword $ba^{2n}b^{-1}$, replace with a^n.
2. That's it, there is no Step 2. Keep repeating Step 1.

Baumslag–Solitar groups

The word search problem in $BS(1,2) = \langle a, b \mid a^b = a^2 \rangle$ requires exponential time.
This contrasts the simplicity of the algorithm to solve the word (search) problem.

$BS(1,2)$ can be viewed as an HNN-extension of $\langle a \rangle$ with a stable letter b and associated subgroups $\langle a \rangle$, $\langle a^2 \rangle$.

The algorithm follows Britton's lemma for HNN-extensions:
1. Look for a subword $b^{-1}a^n b$, replace with a^{2n}; or a subword $ba^{2n}b^{-1}$, replace with a^n.
2. That's it, there is no Step 2. Keep repeating Step 1.

The word search problem in $BS(1,2) = \langle a, b \mid a^b = a^2 \rangle$ requires exponential time.
This contrasts the simplicity of the algorithm to solve the word (search) problem.

$BS(1,2)$ can be viewed as an HNN-extension of $\langle a \rangle$ with a stable letter b and associated subgroups $\langle a \rangle$, $\langle a^2 \rangle$.

The algorithm follows Britton's lemma for HNN-extensions:
1. Look for a subword $b^{-1}a^n b$, replace with a^{2n}; or a subword $ba^{2n}b^{-1}$, replace with a^n.
2. That's it, there is no Step 2. Keep repeating Step 1.

The word search problem in $BS(1,2) = \langle a, b \mid a^b = a^2 \rangle$ requires exponential time.
This contrasts the simplicity of the algorithm to solve the word (search) problem.

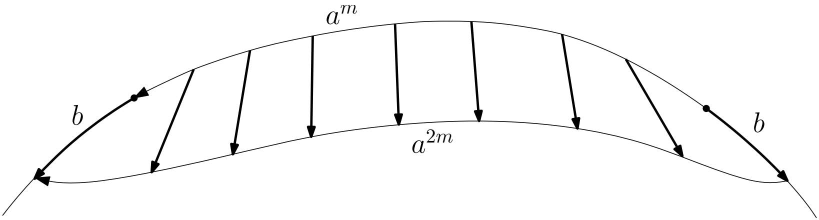
$BS(1,2)$ can be viewed as an HNN-extension of $\langle a \rangle$ with a stable letter b and associated subgroups $\langle a \rangle$, $\langle a^2 \rangle$.

The algorithm follows Britton's lemma for HNN-extensions:
1. Look for a subword $b^{-1}a^n b$, replace with a^{2n}; or a subword $ba^{2n}b^{-1}$, replace with a^n.
2. That's it, there is no Step 2. Keep repeating Step 1.

Baumslag–Solitar groups

Theorem 2. Parameterized complexity of the word search problem in Baumslag–Solitar group $BS(1,2)$ is polynomial in n and k, where n is the size of input and k is the area of a van Kampen diagram.

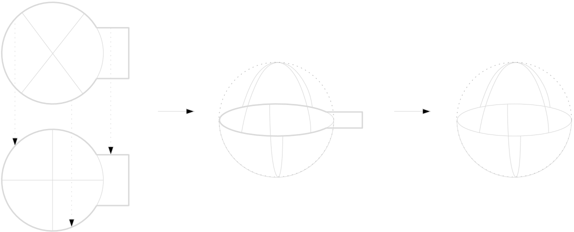
Proof. Each replacement $b^{-1}a^nb \to a^{2n}$ or $ba^{2n}b^{-1} \to a^n$ "peels off" a piece of van Kampen diagram.



Since area is k, this can happen $\leq k$ times.
If we count, we naively get time $O((n+k)^3)$. $\square$

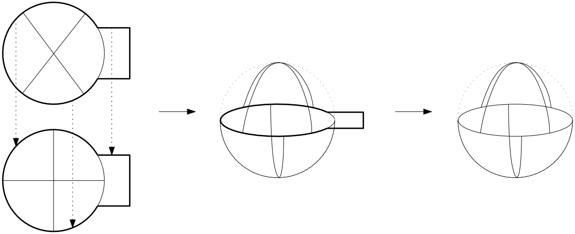
Fine print: how do we know that we chew through the *minimal* van Kampen diagram? Generally, the same word may have different van Kampen diagrams.

If the same word has two substantially different van Kampen diagrams, we can glue them in a sphere:



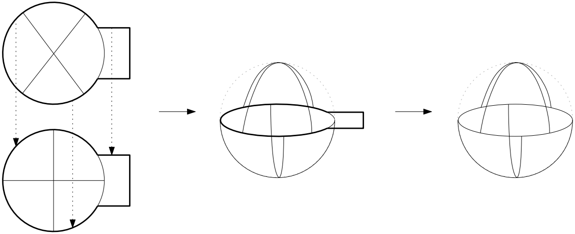
Good news: one-relator groups, in particular, $BS(1, 2)$, are *asperical*, that is, do not admit spherical diagrams. In other words, van Kampen diagrams in $BS(1, 2)$ are unique.

Fine print: how do we know that we chew through the *minimal* van Kampen diagram? Generally, the same word may have different van Kampen diagrams.
If the same word has two substantially different van Kampen diagrams, we can glue them in a sphere:



Good news: one-relator groups, in particular, $BS(1,2)$, are *asperical*, that is, do not admit spherical diagrams. In other words, van Kampen diagrams in $BS(1,2)$ are unique.

Fine print: how do we know that we chew through the *minimal* van Kampen diagram? Generally, the same word may have different van Kampen diagrams.
If the same word has two substantially different van Kampen diagrams, we can glue them in a sphere:



Good news: one-relator groups, in particular, $BS(1,2)$, are *asperical*, that is, do not admit spherical diagrams. In other words, van Kampen diagrams in $BS(1,2)$ are unique.

The Baumslag–Gersten group

Baumslag–Gersten group (Baumslag 1969):

$$\langle a, t \mid a^{a^t} = a^2 \rangle.$$

Rewrite as

$$GB = \langle a, b, t \mid a^b = a^2, a^t = b \rangle,$$

an HNN-extension of $BS(1,2) = \langle a, b \mid a^b = a^2 \rangle$ with a stable letter t and associated subgroups $\langle a \rangle$, $\langle b \rangle$.

Gersten 1992: GB has non-elementary Dehn function (in fact, $\underbrace{\exp \circ \exp \circ \cdots \circ \exp}_{\log n} 1$, Platonov 2004).

So the word search problem in this group is *non-elementary*.
However: word problem is polynomial (Myasnikov, Ushakov, Won 2011).

The Baumslag–Gersten group

Baumslag–Gersten group (Baumslag 1969):

$$\langle a, t \mid a^{a^t} = a^2 \rangle.$$

Rewrite as

$$GB = \langle a, b, t \mid a^b = a^2, a^t = b \rangle,$$

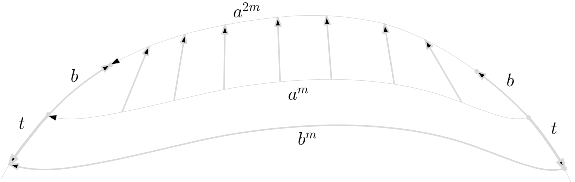
an HNN-extension of $BS(1,2) = \langle a, b \mid a^b = a^2 \rangle$ with a stable letter t and associated subgroups $\langle a \rangle$, $\langle b \rangle$.

Gersten 1992: GB has non-elementary Dehn function (in fact, $\underbrace{\exp \circ \exp \circ \cdots \circ \exp}_{\log n} 1$, Platonov 2004).

So the word search problem in this group is *non-elementary*.
However: word problem is polynomial (Myasnikov, Ushakov, Won 2011).

Theorem 3. Parameterized complexity of the word search problem in the Baumslag–Gersten group GB is polynomial in n and k, where n is the size of input and k is the area of a van Kampen diagram.

Proof. GB is aspherical and essentially the same algorithm works.



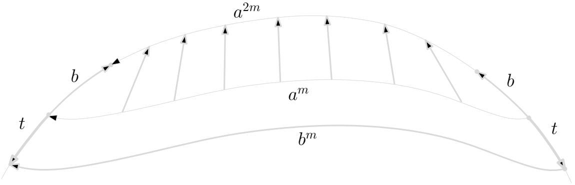
– We still apply Britton lemma repeatedly, looking for subwords $t^{-1} \dots t$ or $t \dots t^{-1}$.

– This time we have to find subwords that represent elements of $\langle a \rangle$ or $\langle b \rangle$ in $BS(1,2)$ (i.e., need to solve membership problem for those subgroups).

– Still polynomial time. □

Theorem 3. Parameterized complexity of the word search problem in the Baumslag–Gersten group GB is polynomial in n and k, where n is the size of input and k is the area of a van Kampen diagram.

Proof. GB is aspherical and essentially the same algorithm works.

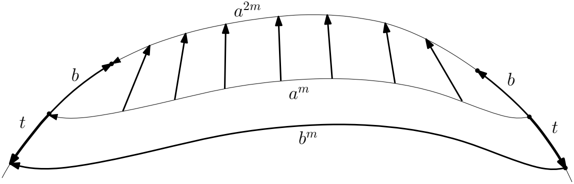


– We still apply Britton lemma repeatedly, looking for subwords $t^{-1} \dots t$ or $t \dots t^{-1}$.
– This time we have to find subwords that represent elements of $\langle a \rangle$ or $\langle b \rangle$ in $BS(1,2)$ (i.e., need to solve membership problem for those subgroups).
– Still polynomial time. □

The Baumslag–Gersten group

Theorem 3. Parameterized complexity of the word search problem in the Baumslag–Gersten group GB is polynomial in n and k, where n is the size of input and k is the area of a van Kampen diagram.

Proof. GB is aspherical and essentially the same algorithm works.



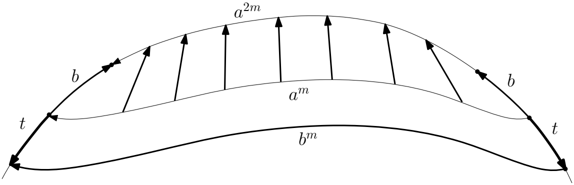
– We still apply Britton lemma repeatedly, looking for subwords $t^{-1} \ldots t$ or $t \ldots t^{-1}$.

– This time we have to find subwords that represent elements of $\langle a \rangle$ or $\langle b \rangle$ in $BS(1,2)$ (i.e., need to solve membership problem for those subgroups).

– Still polynomial time. □

Theorem 3. Parameterized complexity of the word search problem in the Baumslag–Gersten group GB is polynomial in n and k, where n is the size of input and k is the area of a van Kampen diagram.

Proof. GB is aspherical and essentially the same algorithm works.



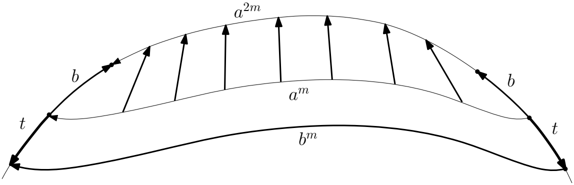
– We still apply Britton lemma repeatedly, looking for subwords $t^{-1} \dots t$ or $t \dots t^{-1}$.

– This time we have to find subwords that represent elements of $\langle a \rangle$ or $\langle b \rangle$ in $BS(1,2)$ (i.e., need to solve membership problem for those subgroups).

– Still polynomial time. □

Theorem 3. Parameterized complexity of the word search problem in the Baumslag–Gersten group GB is polynomial in n and k, where n is the size of input and k is the area of a van Kampen diagram.

Proof. GB is aspherical and essentially the same algorithm works.



– We still apply Britton lemma repeatedly, looking for subwords $t^{-1} \dots t$ or $t \dots t^{-1}$.
– This time we have to find subwords that represent elements of $\langle a \rangle$ or $\langle b \rangle$ in $BS(1,2)$ (i.e., need to solve membership problem for those subgroups).
– Still polynomial time. $\square$

– While the complexity of the word search problem in a finitely presented group GB is immense, the parameterized complexity shows that this complexity comes purely from the size of the answer, and nothing else.

– These results rehabilitate the classical decision algorithm for the word problem in HNN-extensions, which amounts to applying Britton's lemma repeatedly.

– The method may apply to arbitrary one-relator groups (unlike the method of Myasnikov, Ushakov, Won used to show that the word problem in GB is polynomial).

– While the complexity of the word search problem in a finitely presented group GB is immense, the parameterized complexity shows that this complexity comes purely from the size of the answer, and nothing else.

– These results rehabilitate the classical decision algorithm for the word problem in HNN-extensions, which amounts to applying Britton's lemma repeatedly.

– The method may apply to arbitrary one-relator groups (unlike the method of Myasnikov, Ushakov, Won used to show that the word problem in GB is polynomial).

Conclusion

– While the complexity of the word search problem in a finitely presented group GB is immense, the parameterized complexity shows that this complexity comes purely from the size of the answer, and nothing else.
– These results rehabilitate the classical decision algorithm for the word problem in HNN-extensions, which amounts to applying Britton's lemma repeatedly.
– The method may apply to arbitrary one-relator groups (unlike the method of Myasnikov, Ushakov, Won used to show that the word problem in GB is polynomial).

Further developments and current work

– *Word problem in one-relator groups.* It appears that the same approach works to show that the word in arbitrary one-relator groups.

Magnus breakdown: reduce the (extended) word problem in $\langle X \mid r \rangle$ to the extended word problem in $\langle X' \mid r' \rangle$, where $|r'| < |r|$.

Extended word problem in $\langle X \mid r \rangle$ is the membership problem in $\langle Y \rangle$, where $Y \subseteq X$.

– *Word problem in HNN-extensions.* It appears that the same works for a wide class of HNN-extensions. If $G = \langle X \mid R \rangle$, $H, K \leq G$, and $\varphi : H \to K$ is an isomorphism, the corresponding HNN-extension is

$$\langle X, t \mid R, h^t = \varphi(h), h \in H \rangle.$$

The main difficulty is to come up with the right kind of assumptions on the membership problem in the underlying group.

Further developments and current work

– *Word problem in one-relator groups.* It appears that the same approach works to show that the word in arbitrary one-relator groups.
Magnus breakdown: reduce the (extended) word problem in $\langle X \mid r \rangle$ to the extended word problem in $\langle X' \mid r' \rangle$, where $|r'| < |r|$.
Extended word problem in $\langle X \mid r \rangle$ is the membership problem in $\langle Y \rangle$, where $Y \subseteq X$.
– *Word problem in HNN-extensions.* It appears that the same works for a wide class of HNN-extensions. If $G = \langle X \mid R \rangle$, $H, K \leq G$, and $\varphi : H \to K$ is an isomorphism, the corresponding HNN-extension is

$$\langle X, t \mid R, h^t = \varphi(h), h \in H \rangle.$$

The main difficulty is to come up with the right kind of assumptions on the membership problem in the underlying group.

Further developments and current work

– *Word problem in one-relator groups.* It appears that the same approach works to show that the word in arbitrary one-relator groups.

Magnus breakdown: reduce the (extended) word problem in $\langle X \mid r \rangle$ to the extended word problem in $\langle X' \mid r' \rangle$, where $|r'| < |r|$.

Extended word problem in $\langle X \mid r \rangle$ is the membership problem in $\langle Y \rangle$, where $Y \subseteq X$.

– *Word problem in HNN-extensions.* It appears that the same works for a wide class of HNN-extensions. If $G = \langle X \mid R \rangle$, $H, K \leq G$, and $\varphi : H \to K$ is an isomorphism, the corresponding HNN-extension is

$$\langle X, t \mid R, h^t = \varphi(h), h \in H \rangle.$$

The main difficulty is to come up with the right kind of assumptions on the membership problem in the underlying group.

Further developments and current work

Conjugacy problem for a group G with presentation $\langle X \mid R \rangle$:
Given group words u, v in X, establish if there is $g \in G$ s.t. $u^g = v$.

What to use as a parameter?
– Length of conjugating element.
– Area of annular diagram. In this case, a result similar to Theorem 1 takes place.

Membership problem for a group G with presentation $\langle X \mid R \rangle$:
Given group words $u_1, \ldots, u_n, u$ in X, establish if $u \in \langle u_1, \ldots, u_n \rangle$.

What to use as a parameter?

Further developments and current work

Conjugacy problem for a group G with presentation $\langle X \mid R \rangle$:
Given group words u, v in X, establish if there is $g \in G$ s.t. $u^g = v$.

What to use as a parameter?
– Length of conjugating element.
– Area of annular diagram. In this case, a result similar to Theorem 1 takes place.

Membership problem for a group G with presentation $\langle X \mid R \rangle$:
Given group words $u_1, \ldots, u_n, u$ in X, establish if $u \in \langle u_1, \ldots, u_n \rangle$.

What to use as a parameter?

Further developments and current work

Conjugacy problem for a group G with presentation $\langle X \mid R \rangle$:
Given group words u, v in X, establish if there is $g \in G$ s.t. $u^g = v$.

What to use as a parameter?
– Length of conjugating element.
– Area of annular diagram. In this case, a result similar to Theorem 1 takes place.

Membership problem for a group G with presentation $\langle X \mid R \rangle$:
Given group words $u_1, \ldots, u_n, u$ in X, establish if $u \in \langle u_1, \ldots, u_n \rangle$.

What to use as a parameter?