Word Problem for the Baumslag-Gersten group $\left\langle a, b \mid a^{a^b} = a^2 \right\rangle$ is in **P**

Sasha Ushakov
Stevens Institute of Technology
sasha.ushakov@gmail.com

17th March 2007

Results

Let $G = \langle a, b \mid (b^{-1}ab)^{-1}a(b^{-1}ab) = a^2 \rangle$ (Baumslag-Gersten group).

A. Invent a new tool: **Power circuit!!!**

B. The Word problem for G is in P.

C. The Conjugacy **search** problem for G is generically P.

Some interesting properties of G

Super-exponential Dehn function $D(n) = tower_2(\log_2(n))$. Indeed, define

$$\begin{aligned} w_0 &= a \\ \ldots \\ w_i &= (b^{-1} w_{i+1} b) a (b^{-1} w_{i+1} b). \end{aligned}$$

Then for every $i \in \mathbb{Z}$,

$$w_i =_G a^{tower_2(i)}$$

and hence

$$u_i \equiv [w_i, a] =_G 1.$$

Since $|u_i| = 2^{i+3}$ and $Area(u_i) \approx tower_2(i)$ it follows that $D_G(n) \geq tower_2(\log_2 n)$.

Known approaches to the Word Problem

There are two standard approaches to the Word problem in G:

- Magnus break down method (general method for one-relator groups). Very complicated algorithm, impossible to analyze except trivial cases.
- Britton's Lemma (method for HNN-extensions of groups).

$$G = \langle a, b, t \mid b^{-1}ab = t,\ t^{-1}at = a^2 \rangle = \langle H, b \mid b^{-1}ab = t \rangle$$

which is an HNN-extension of Baumslag-Solitar group

$$H = \langle a, t \mid t^{-1}at = a^2 \rangle.$$

Power circuit (definition 1/2)

A **power circuit** is a quadruple $(\mathcal{P}, \mu, M, \nu)$ satisfying the conditions below:

- $\mathcal{P} = (V(\mathcal{P}), E(\mathcal{P}))$ a directed graph with no multiple edges and no directed cycles;
- $\mu : E(\mathcal{P}) \to \{1, -1\}$ a function called **the edge labelling function**;
- $M \subseteq V(\mathcal{P})$ a set of vertices called **the set of marked vertices**;
- and $\nu : M \to \{-1, 1\}$ a function called **the sign function**.

For an edge $e = v_1 \to v_2$ in $\mathcal{P}$ denote its **origin** v_1 by $\alpha(e)$ and its **terminus** v_2 by $\beta(e)$. For a vertex v in $\mathcal{P}$ define sets

$$In_v = \{e \in \mathcal{P} \mid \beta(e) = v\} \text{ and } Out_v = \{e \in \mathcal{P} \mid \alpha(e) = v\}.$$

A vertex v in $\mathcal{P}$ is called an **origin** if $In_v = \emptyset$.

Power circuit (definition 2/2)

Inductively define a function $\mathcal{E} : V(\mathcal{P}) \to \mathbb{R}$, for $v \in V(\mathcal{P})$ define

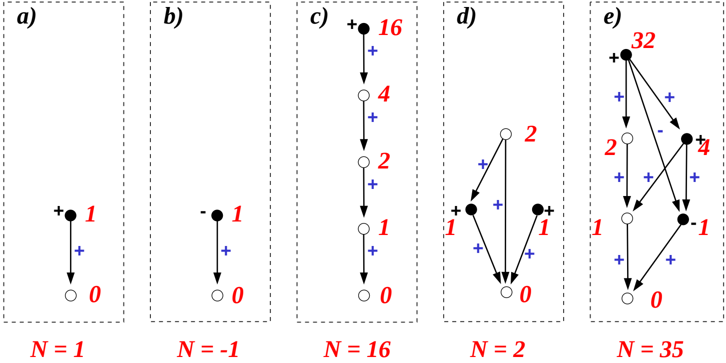
$$\mathcal{E}(\mathbf{v}) = \begin{cases} 0 & \text{if } Out_v = \emptyset; \\ 2^{\sum_{e \in Out_v} \mu(e)\mathcal{E}(\beta(e))} & \text{otherwise} \end{cases}$$

In this work we assume that $\mathcal{E}(v) \in \mathbb{Z}$ for each $v \in V(\mathcal{P})$.

Since $\mathcal{P}$ contains no cycles the function $\mathcal{E}$ is well-defined. Finally, assign a number $\mathcal{N}$ to a circuit $\mathcal{P} = (\mathcal{P}, \mu, M, \nu)$

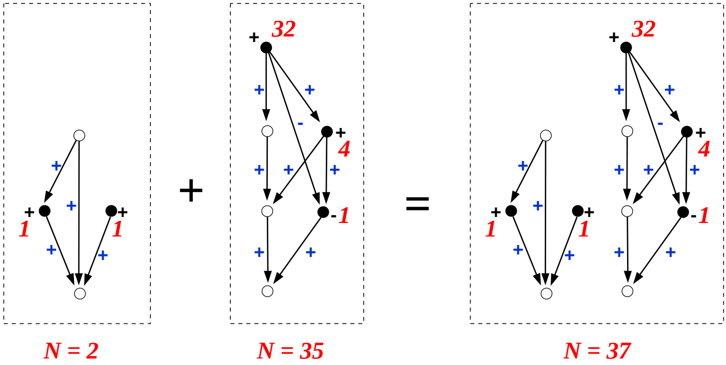
$$\mathcal{N} = \mathcal{N}(\mathcal{P}, \mu, M, \nu) = \sum_{v \in M} \nu(v)\mathcal{E}(v).$$

Examples of circuits



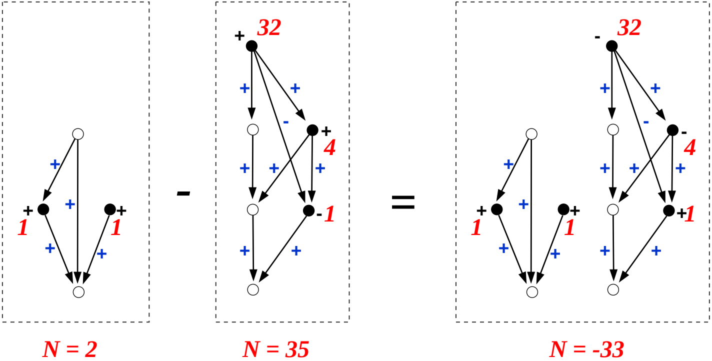
1. Black vertices denote the marked vertices.
2. An edge e is labeled with "+" if $\mu(e) = 1$.
3. A marked vertex v is labeled with "+" if $\nu(v) = 1$.

Addition



Let $\mathcal{P}_+ = \mathcal{P}_1 + \mathcal{P}_2$. Then $|V(\mathcal{P}_+)| = |V(\mathcal{P}_1)| + |V(\mathcal{P}_2)|$.

Subtraction



Let $\mathcal{P}_- = \mathcal{P}_1 + \mathcal{P}_2$. Then $|V(\mathcal{P}_-)| = |V(\mathcal{P}_1)| + |V(\mathcal{P}_2)|$.

Reduction (idea, main facts)

A circuit $\mathcal{P}$ is called **reduced** for all distinct $v_1, v_2 \in V(\mathcal{P})$, $\mathcal{E}(v_1) = \mathcal{E}(v_2)$ (plus a few other minor properties).

Theorem

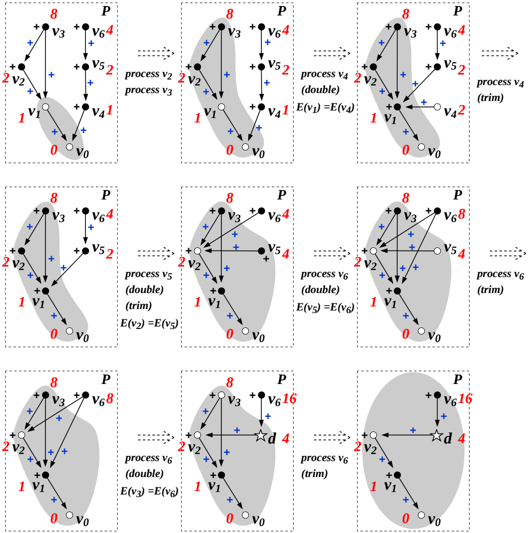
There exists an algorithm which for any circuit $\mathcal{P}$ finds and equivalent reduced circuit $\mathcal{P}'$. Moreover, $|V(\mathcal{P}')| \leq |V(\mathcal{P})| + 1$ and the time complexity of the algorithm is $O(|V(\mathcal{P})|^3)$.

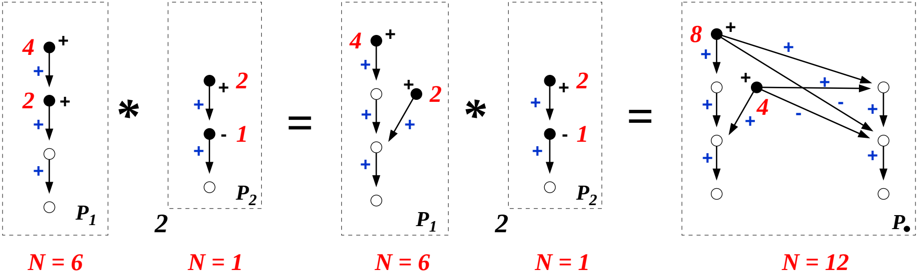
If $\mathcal{P}$ is a reduced circuit

- $\mathcal{N}(\mathcal{P}) = 0$ if and only if $\mathcal{P}$ has no marked vertices.
- If v_+ is the vertex with the greatest $\mathcal{E}$ value then $sign(\mathcal{N}(\mathcal{P})) = sign(\mathcal{E}(v_+))$.
- If v_- is the vertex with the smallest $\mathcal{E}$ value then 2^N divides $\mathcal{N}(\mathcal{P})$ if and only if 2^N divides $\mathcal{E}(v_-)$.

Reduction (example)

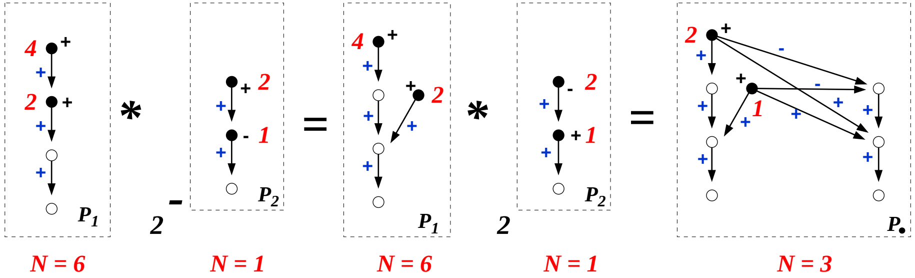
N = 17





Let $\mathcal{P}_\bullet = \mathcal{P}_1 \bullet \mathcal{P}_2 := \mathcal{P}_1 \cdot 2^{\mathcal{P}_2}$. Clearly, $|V(\mathcal{P}_\bullet)| = |V(\mathcal{P}_1)| + |V(\mathcal{P}_2)|$.

Division by a power of 2



Let $\mathcal{P}_\circ = \mathcal{P}_1 \bullet \mathcal{P}_2 := \mathcal{P}_1 \cdot 2^{-\mathcal{P}_2}$.
If all marked vertices in $\mathcal{P}_1$ are origins then $|V(\mathcal{P}_\bullet)| = |V(\mathcal{P}_1)| + |V(\mathcal{P}_2)|$.
Operation $\circ$ is not defined for all pairs $\mathcal{P}_1$, $\mathcal{P}_2$!!! For any pair of circuits $\mathcal{P}_1$, $\mathcal{P}_2$ it is possible to check if $\mathcal{P}_1 \circ \mathcal{P}_2$ is defined or not.

Our algorithm for the WP

Recall that

$$G = \langle a, b, t \mid b^{-1}ab = t,\ t^{-1}at = a^2 \rangle = \langle H, b \mid b^{-1}ab = t \rangle$$

HNN-extension of Baumslag-Solitar group

$$H = \langle a, t \mid t^{-1}at = a^2 \rangle.$$

Let

$$w = w(a, b, t) = g_0(a, t)b^{\varepsilon_1}g_1(a, t)b^{\varepsilon_2}g_2(a, t)\ldots b^{\varepsilon_n}g_n(a, t)$$

By Britton's Lemma if $n > 0$ and $w =_G 1$ then w contains a subword

- either $b^{-1}g(a, t)b$ where $g(a, t) =_H a^k$, in which case it can be replaced by t^k;
- or $bg(a, t)b^{-1}$ where $g(a, t) =_H t^k$, in which case it can be replaced by a^k.

Thus, the procedure makes up to $2n$ operations of checking if words $g_i(a, t)$ belong to $\langle a \rangle$ or $\langle t \rangle$, and if that is the case finding the exact power of a or t (the powers of which can grow super-exponentially fast).

Algorithm (stage 1)

From now on, assume that words $x_1^{p_1} x_2^{p_2} \dots x_n^{p_n}$ are given as sequences of pairs

$$(x_1, \mathcal{P}_1), \dots, (x_n, \mathcal{P}_n)$$

where $\mathcal{P}_i$ is a circuit representing p_i.
Consider the word over the alphabet of H

$$g(a,t) = a^{m_0} t^{\delta_1} a^{m_1} t^{\delta_2} a^{m_2} \dots t^{\delta_k} a^{m_k}$$

where $\delta_j \in \mathbb{Z}$ and $m_j \in \mathbb{Z}$.
Observe that for any positive k

$$t^{-k} a^m = a^{m2^k} t^{-k}$$

is a relation in G.

Algorithm (stage 1)

If $\delta_1 < 0$ then using the relation $t^{-k} a^m = a^{m2^k} t^{-k}$ obtain

$$a^{m_0} a^{m_1 2^{-\delta_1}} t^{\delta_1+\delta_2} a^{m_2} t^{\delta_3} a^{m_3} \dots t^{\delta_k} a^{m_k}.$$

If $\delta_1 + \delta_2 < 0$ then continue rewriting and obtain

$$a^{m_0} a^{m_1 2^{-\delta_1}} a^{m_2 2^{-\delta_1-\delta_2}} t^{\delta_1+\delta_2+\delta_3} a^{m_3} \dots t^{\delta_k} a^{m_k}.$$

If $\delta_1 + \delta_2 + \delta_3 > 0$ then we check the next power of t which is t^{δ_4}. And so on.

Algorithm (stage 1)

Finally, we obtain a word of one of the two types

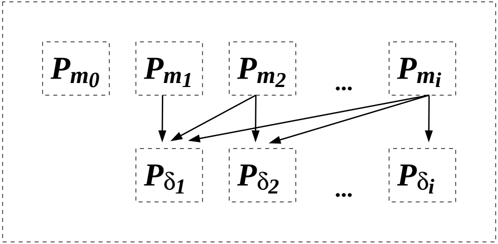
$$\begin{array}{c} a^{\left(m_0+m_1 2^{-\delta_1}+m_2 2^{-\delta_1-\delta_2}\right)} \cdot t^{(\delta_1+\delta_2+\delta_3)} \cdot \ldots \\ \ldots \cdot a^{\left(m_i+m_{i+1}2^{-\delta_{i+1}}+\ldots+m_{j-1}2^{-\delta_{i+1}-\cdots-\delta_{j-1}}\right)} \cdot t^{(\delta_{i+1}+\ldots+\delta_j)} \cdot \\ \cdot a^{\left(m_j+m_{j+1}2^{-\delta_{j+1}}+\ldots+m_{k-1}2^{-\delta_{j+1}-\cdots-\delta_{k-1}}\right)} \cdot t^{(\delta_{j+1}+\ldots+\delta_k)} a^{m_k} \end{array} \quad (1)$$

or

$$\begin{array}{c} a^{\left(m_0+m_1 2^{-\delta_1}+m_2 2^{-\delta_1-\delta_2}\right)} \cdot t^{(\delta_1+\delta_2+\delta_3)} \cdot \ldots \\ \ldots \cdot a^{\left(m_i+m_{i+1}2^{-\delta_{i+1}}+\ldots+m_{j-1}2^{-\delta_{i+1}-\cdots-\delta_{j-1}}\right)} \cdot t^{(\delta_{i+1}+\ldots+\delta_j)} \cdot \\ \cdot a^{\left(m_j+m_{j+1}2^{-\delta_{j+1}}+\ldots+m_{k-1}2^{-\delta_{j+1}-\cdots-\delta_{k-1}}+m_k 2^{-\delta_{j+1}-\cdots-\delta_k}\right)} \cdot t^{(\delta_{j+1}+\ldots+\delta_k)} \end{array} \quad (2)$$

Algorithm (stage 1)

The powers $m_0 + m_1 2^{-\delta_1} + m_2 2^{-\delta_1 - \delta_2} + \ldots m_i 2^{-\delta_1 - \cdots - \delta_i}$ of a can be realized compactly as circuits:



• Clearly, the operation can be performed in linear time and if $\mathcal{P}$ is the result then $|V(\mathcal{P})| = \sum_{j=0}^{m} |V(\mathcal{P}_{m_j})| + \sum_{j=1}^{m} |V(\mathcal{P}_{\delta_j})|$.
• Clearly, the size of powers $\delta_1 + \ldots + \delta_i$ of t is the sum of sizes of $\mathcal{P}_{\delta_j}$'s.
• At stage 1 we use up to n comparisons (cubic time). All other steps (constructing new powers of a and t) have linear time complexity.

Algorithm (stage 2)

Consider the terminal segment

$$a^{\sum_{a=i}^{j-1} m_a 2^{-\sum_{b=i+1}^{a} \delta_b}} \cdot t^{\delta_{i+1}+\ldots+\delta_j} \cdot a^{\sum_{a=j}^{k} m_a 2^{-\sum_{b=j+1}^{a} \delta_b}} \cdot t^{\delta_{j+1}+\ldots+\delta_k}$$

of (2). Further rewrites possible if and only if it satisfies 3 properties:

- $\delta_{j+1} + \ldots + \delta_k < 0$;
- $0 < \delta_{i+1} + \ldots + \delta_j \leq -(\delta_{j+1} + \ldots + \delta_k)$;
- the number $\sum_{a=j}^{k} m_a 2^{-\sum_{b=j+1}^{a} \delta_b}$ is divisible by $2^{\delta_{i+1}+\ldots+\delta_j}$.

Rewrite the last syllable to get a word

$$a^{\left(\sum_{a=i}^{j-1} m_a 2^{-\sum_{b=i+1}^{a} \delta_b}\right)+\left(2^{-\delta_{i+1}-\ldots-\delta_j}\right)\cdot\left(\sum_{a=j}^{k} m_a 2^{-\sum_{b=j+1}^{a} \delta_b}\right)} \cdot t^{\delta_{i+1}+\ldots+\delta_k}. \qquad (3)$$

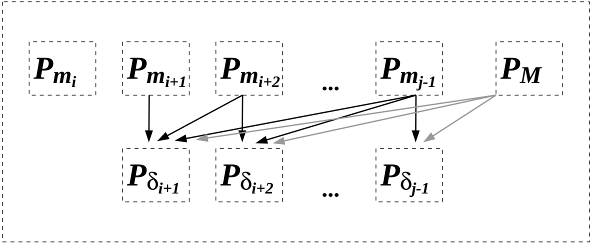
Continue the rewrites while possible...

Algorithm (stage 2)

If

$$M = \sum_{a=j}^{k} m_a 2^{-\sum_{b=j+1}^{a} \delta_b}$$

then the number $\left(\sum_{a=i}^{j-1} m_a 2^{-\sum_{b=i+1}^{a} \delta_b}\right) + \left(2^{-\delta_{i+1} - \ldots - \delta_j}\right) \cdot \left(\sum_{a=j}^{k} m_a 2^{-\sum_{b=j+1}^{a} \delta_b}\right)$ is constructed as follows:



Algorithm (stage 2)

- It requires cubic time to check 3 conditions above.
- It requires linear time to perform the rewrite.
- The size of circuits grows linearly!!!

Thus

1. We perform at most n "elementary" rewrites.
2. Each operation has complexity $O(k^3)$ in terms of the number of vertices in the corresponding circuits.
3. The number of vertices in circuits can be bounded by n^2.
4. The total complexity is $O(n^7)$.

Thank you.