Compressed decision problems in relatively hyperbolic groups

Sarah Rees

University of Newcastle, UK

York, 8th Feb 2023

Plan for today

- I'll explain the words in my title, in particular I'll tell you what the compressed word and conjugacy problems are, why they're interesting.
- I'll state my recent work with Derek Holt that I want to discuss, as well as the results of Lohrey and Holt,Lohrey&Schleimer for free and hyperbolic groups that it extends.
- I'll explain (but without too much technical detail) how the constructions of polynomial time solutions for the compressed word problems for free and hyperbolic groups work, and then how we can adapt those proofs to the more general case of groups hyperbolic relative to free abelian groups. The basic ideas and constructions of our work all come from HLS; we simply adapt them to make them work in a more general case.
- I'll explain very briefly how we can also extend other results (in particular the linear time solution of the compressed conjugacy problem) of HLS from hyperbolic to rel. hyperbolic groups.

Word problem and compressed word problem

Let $G = \langle X \rangle$ be a finitely generated group, where $X \supseteq X^{-1}$.
The **word problem** for G, $\mathtt{WP}(G)$ asks if $\exists$ an algorithm that, for any **word** w over X, decides if $w =_G 1$. We express the time complexity of the algorithm as a function of the (string) **length** n of w (called $|w|$).

For G free, hyperbolic (Alonso et al.; Holt), or hyperbolic relative to (virtually) abelian subgroups (Ciobanu, Holt, Rees), $\mathtt{WP}(G)$ has linear time (and in fact real time) complexity.

The **compressed word problem** $\mathtt{CWP}(G)$ asks the same question as $\mathtt{WP}(G)$, but with input 'word' given in compressed format defined by a **straight-line program**, SLP (defn to follow). Time complexity is expressed in terms of the size $|\mathrm{SLP}|$ of the SLP (which typically is logarithmic in $|w|$).

It's known that, for G free (Lohrey, 2006) or hyperbolic (Holt,Lohrey &Schleimer, STACS & ArXiv 2019), $\mathtt{CWP}(G)$ is soluble in poly. time.

I'll talk about a recent result (Holt, Rees, 2022) that the same is true for G hyperbolic rel. to free abelian subgroups.

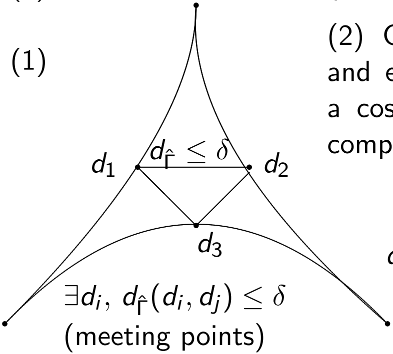
Relative hyperbolicity (à la Osin)

A group $G = \langle X \rangle$ is hyperbolic relative to a finite collection $\{H_i : i \in \Omega\}$ of *parabolic* subgroups of G, if, where $\mathcal{H} = \bigcup_i (H_i \setminus \{1\})$ and $\widehat{X} = X \cup \mathcal{H}$,

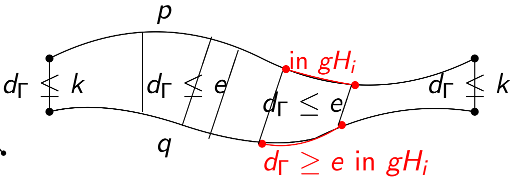
(1) the Cayley graph $\widehat{\Gamma} = \widehat{\Gamma}(G, \widehat{X})$ is δ-hyperbolic for some δ and

(2) G has 'bounded coset penetration' (bcp relates $\Gamma = \Gamma(G, X)$ to $\hat{\Gamma}$).

(1)



(2) Given p, q, $\hat{\Gamma}$-quasi-geodesics, that start and end close in Γ, don't backtrack (re-enter a coset), & st q contains a long enough 'i-component' (over gens. in H_i, shown in red):



Rel. hyp. generalises hyp., admits egs such as $\pi_1(M)$, M fin. vol. hyp.
If $\Omega = \emptyset$, then $\Gamma = \widehat{\Gamma}$ and G is (δ-)hyperbolic. If also $\delta = 0$, then G is free.

Relating Γ and $\widehat{\Gamma}$: derived words

Notn: A word of length n over X or $\widehat{X}$ (assumed inverse closed) is a string $\alpha = a_0 \cdots a_{n-1}$ of elts of X or $\widehat{X}$. ε denotes the empty word, $\alpha[i:j)$ the subword $a_i \cdots a_{j-1}$; we abbreviate $\alpha[0:j)$ and $\alpha[i:n)$ as $\alpha[:j)$ and $\alpha[i:)$.

We relate each word over X to its **derived word over $\widehat{X}$** as follows:

Given w expressed as a concatenation $\alpha_0\beta_1\alpha_1\beta_2\cdots\beta_n\alpha_n$ of words in X^*, with components $\beta_1,\ldots,\beta_n$, with β_i over gens of H_{j_i}, we define the derived word $\widehat{w} = \alpha_0 h_1 \alpha_1 h_2 \cdots h_n \alpha_n$, with h_i the elt of H_{j_i} rep. by β_i.

We need to deal with the relationship between words w over X (paths in Γ) and the corresponding derived words $\widehat{w}$ over $\widehat{X}$ (paths in $\widehat{\Gamma}$), and subwords/subpaths of both.

We're particularly interested in subwords w' of w that **don't split components**, i.e. both start and finish with α-subwords. In that case we also find $\widehat{w'}$ as a subword of $\widehat{w}$. So we have i,j with $w' = w[i,j)$, but also k,l with $\widehat{w'} = \widehat{w}[k,l)$. Sometimes we shall use the notation $w[[k,l))$ to denote w', in order to associate w' with the subword $\widehat{w'}$ of $\widehat{w}$.

Straight-line programs

A **straight-line program** (SLP) over X is a triple $\mathcal{G} = (V, S, \rho)$, where X, V are a finite alphabet and variable set, $S \in V$ is the **start variable**, ρ is a map $\rho : V \to (V \cup X)^*$, whose domain extends naturally to $(V \cup X)^*$, such that $\forall A, k$, A cannot occur in $\rho^k(A)$ (acyclicity).

We define the **size** of $\mathcal{G}$, $|\mathcal{G}|$, to be $\sum_{A \in V} |\rho(A)|$.

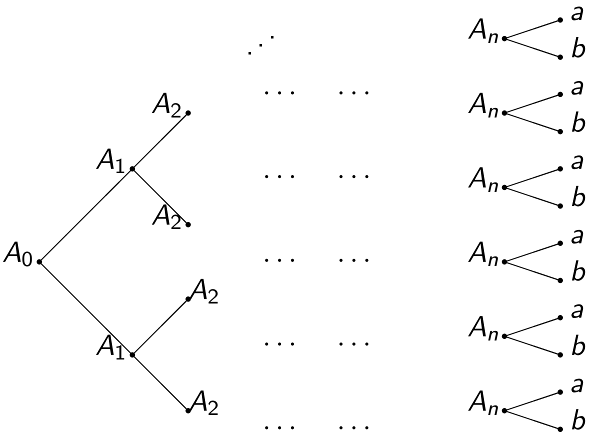
In effect, $\mathcal{G}$ 'is' a **context-free grammar** generating a single string $\mathtt{val}(\mathcal{G})$ in X^*:

$$\mathtt{val}(\mathcal{G}) := \rho^{\mathtt{ht}(S)}(S), \text{ where, for } A \in V,\ \mathtt{ht}(A) := \min\{k : \rho^k(A) \in X^*\}.$$

SLPs, like grammars, can be put into Chomsky normal form (in poly-time).
For each $A \in V$, a subgrammar $\mathcal{G}_A = (V, A, \rho)$ generates the subword $\mathtt{val}(\mathcal{G}_A) = \mathtt{val}(A) := \rho^{\mathtt{ht}(A)}(A)$ of $\rho(\mathcal{G})$.
Note that if $\rho(A) = BC$ then $\mathtt{val}(A) = \mathtt{val}(B)\mathtt{val}(C)$.

An example of an SLP

Where $X = \{a, b\}$, $\mathcal{G} = (\{A_0, \ldots, A_n\}, \rho, A_0)$ with $\rho(A_n) = ab$, $\rho(A_{i-1}) = A_i A_i$, $0 < i \leq n$ is an SLP of size $2(n+1)$ for the word $(ab)^{2^n}$. For each i, $\mathcal{G}_{A_i}$ is an SLP for $(ab)^{2^{n-i}}$.



Motivation: $\mathtt{CWP}(G)$ relates to $\mathtt{WP}$

Schleimer 2008: If $X = \{x_1, \ldots, x_M\}$, $G = \langle X \rangle$, $A = \langle \alpha_1, \ldots, \alpha_N \rangle$, $A < Aut(G)$, then $\mathtt{WP}(A)$ is poly-time reducible to $\mathtt{CWP}(G)$. Hence $\mathtt{WP}(\mathtt{Aut}(F_m))$ is soluble in poly-time.

We observe that $\xi =_A 1 \iff$ for each $i = 1, \ldots, M$, $x_i^{-1}\xi(x_i) =_G 1$, And then, given an expression for $\xi \in A$ as a word $\alpha_{i_1} \cdots \alpha_{i_n}$, we define (following here Lohrey's 2014 book rather than Schleimer), for each i, an SLP $\mathcal{G}_i = (\{A_{k,a} : a \in X \cup X^{-1}, k = 1, \ldots, n\}, \rho, A_{n,x_i})$ for $\xi(x_i)$ (from which we can easily deduce one for $x_i^{-1}\xi(x_i)$) with the following defn. of ρ:

$$\rho(A_{k,a}) = \begin{cases} a & (k = 0) \\ A_{k-1,a_1} \cdots A_{k-1,a_{m_k}} & (0 < k \leq n) \quad \text{where } \alpha_{i_k}(a) = a_1 \cdots a_{m_k} \end{cases}$$

Each SLP $\mathcal{G}_i$ $(i = 1, \ldots, M)$ has size $K|\xi|$, $K = K(G, A)$, and so we reduce $\mathtt{WP}(A)$ to $\mathtt{CWP}(G)$ in poly-time.

When $G = F_m$, then $A = Aut(F_m)$ is $\mathtt{fg}$, and $\mathtt{CWP}(F_m)$ is soluble in poly-time (Lohrey). It follows that $\mathtt{WP}(Aut(F_m))$ is soluble in poly-time.

Motivation: $\texttt{CWP}(G)$ relates to $\texttt{WP}$ (2)

We also have:

$\texttt{WP}(K \rtimes_\phi Q)$ reduces in log-space (and hence in poly-time) to a combination of $\texttt{WP}(Q)$ and $\texttt{CWP}(K)$ (Lohrey&Schleimer, 2007)

We note that all the following have poly-time $\texttt{CWP}$:

- free, word hyperbolic groups (Lohrey, Holt,Lohrey&Schleimer)
- finitely generated nilpotent groups
- all virtually special groups, ie finite extensions of subgroups of RAAGs and so
  - Coxeter groups
  - fully residually free groups
  - fundamental groups of hyperbolic 3-manifolds

Certainly $\texttt{CWP}(G)$ is **not always** easy, even if $\texttt{WP}(G)$ **is**.

Eg Thompson's group F has $\texttt{CWP}$ that is co-NP hard, **but** its word problem is in the subclass AC^1 of P (Lohrey). The hardness of $\texttt{CWP}(G)$ is further explored in Bartholdi,Figelius,Lohrey&Weiß, LiPics & ArXiv 2020.

Our result

Theorem (Holt, Rees, 2022)

The compressed word problem for a group that is hyperbolic relative to a finite collection of free abelian subgroups is soluble in polynomial time.

We extend the proofs of Lohrey and Holt,Lohrey&Schleimer for free and hyperbolic groups.

We need to work harder to get the geometry we need to make it all work in polynomial time. But the basic ideas are in the proofs for free and hyperbolic groups.

The basic idea of the proof

We have a good computable normal form map nf() for our group G, with $\mathsf{nf}(1) = \epsilon$. Our input is an SLP $\mathcal{G}$ generating our 'input word' w. We can assume $\mathcal{G}$ is in Chomsky normal form, and generally well structured.

- We aim to construct from $\mathcal{G}$ an SLP $\mathcal{S}$ that generates $\mathsf{nf}(w)$. Note that $w =_G 1 \iff \mathsf{nf}(w) = \epsilon \iff \mathtt{val}(\mathcal{S}) = \epsilon$.
- We work from the leaves to the root of the production tree for $\mathcal{G}$. The basic step deals with productions of the form $A \to BC$.
- For this basic step we need to build an SLP with value $\mathsf{nf}(\mathsf{nf}(B)\mathsf{nf}(C))$ out of SLPs for $\mathsf{nf}(B)$ and $\mathsf{nf}(C)$, i.e we need to find $\mathsf{nf}(v_1 v_2)$, given v_1, v_2 in normal form. In the free group we have

  $$\mathsf{nf}(v_1 v_2) = v_{11} v_{22}, \text{ where } v_1 = v_{11} v_{12},\ v_2 = v_{21} v_{22},\ v_{21} = v_{12}^{-1}.$$

  In hyperbolic and relatively hyperbolic groups it is more complicated, but we have hyperbolic geometry to help us.
- We save time and space in our construction by building first a TCSLP for $\mathsf{nf}(w)$, which allows more sophisticated productions than an SLP, then modifying it (in two stages) to get an SLP with the same value.

Cut-SLPs

We extend the defn. of SLP to that of **cut straight-line program** (CSLP), allowing additional productions using the **cut operator**, of the form

$$A \to B[i : j),$$

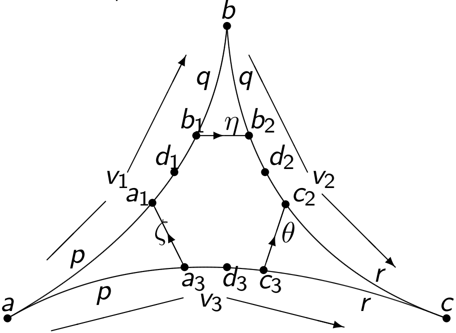
where, for a production of this type, we define $\mathtt{val}(A) := \mathtt{val}(B)[i : j)$. Cut operators are exactly what's needed for $\mathtt{CWP}(F_n)$. We already observed that for freely reduced v_1, v_2 of lengths n_1, n_2, for some k,

$$\mathsf{nf}(v_1 v_2) = v_1[1 : n_1 - k)v_2[k : n_2).$$

Cut operators were used by Lohrey (2006), introduced by Gasieniec et al.(1996), studied by Hagenah (2000). For relatively hyperbolic groups, we shall need to allow cut operators of the form $B[[k : l))$ as well as $B[i, j)$; in this case we say that the CSLP is specified **relative to compression**.
But for hyperbolic and relatively hyperbolic groups adding cut operators is not enough on its own, since in these cases the word $\mathsf{nf}(v_1 v_2)$ is only (in some sense) **close** to the paths labelled v_1, v_2 in the Cayley graph. So now we need the extra power of **tethering** (TCSLPs, to be defined soon).

G hyperbolic: exploiting geometry of Γ, defining $\mathtt{val}_{\mathcal{G}}(A)$

For hyp. G, we choose $\mathrm{nf}(w)$ to be its shortlex min. rep. $\mathrm{slex}(w)$, (a selected geodesic in Γ, giving G a **biautomatic structure**). Let $\mathcal{G}$ contain a production $A \to BC$, where $v_1 = \mathrm{nf}(\mathtt{val}_{\mathcal{G}}(B)), v_2 = \mathrm{nf}(\mathtt{val}_{\mathcal{G}}(C))$. We need to find $v_3 := \mathrm{nf}(v_1 v_2)$. We consider the hyperbolic triangle in Γ with sides γ_{v_i} as shown, **δ-thin**, meeting pts d_i. Given **corresponding** vtces



b_i st $|\eta| \leq \delta$ (maximising $d(b_1, b)$, found using **binary search**), **can find** a_1, c_2 on v_1, v_2 corresp. to a_3, c_3 on v_3 st $|\zeta|, |\theta| \leq \delta$. Then v_3 is concat. of $\mathrm{nf}(v_1[1\!:p)\zeta^{-1})$, $\mathrm{nf}(\zeta v_1[p\!:n_1\!-\!q)\eta v_2[q\!:\!n_2\!-\!r)\theta^{-1})$ and $\mathrm{nf}(\theta v_2[n_2 - r : n_2))$. We search exhaustively for ζ, θ. Choice is correct $\iff$ concatenation is in slex.

In some cases (when $\nexists b_i$, see later) we have a different triangle, but this is the basic idea. **But** to do this, we need to be able to build SLPs for words like $\mathrm{nf}(v_1[1 : p)\zeta^{-1})$ for short ζ.

Tethered-SLPs

Given a normal form nf() for words over X, we extend the definition of SLP to define a **tethered straight-line program** (TSLP) by allowing additional productions that use the **tethered operator**, of the form

$$A \to B\langle \alpha, \beta \rangle$$

for selected words α, β of length bounded by a constant J, where, for a production of this type, we define $\mathtt{val}(A) := \mathsf{nf}(\alpha\, \mathtt{val}(B)\, \beta^{-1})$

Similarly we define **tethered cut straight-line programs** (TCSLPs) as extensions of CSLPs.

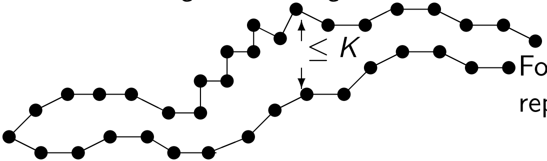
We see that we can find the word $v_3 := \mathsf{nf}(v_1 v_2)$ from the previous slide as the value of a production with `rhs` that is the concatenation of 3 productions, each involving both cut and tether operators:

$$(B[1:p))\langle \epsilon, \zeta \rangle,\ (B[p:n_1-q)\eta C[q:n_2-r))\langle \zeta, \theta \rangle,\ (C[n_2-r:n_2))\langle \theta, \epsilon \rangle$$

Tethered-SLPs and tethered-CSLPs were used by Holt,Lohrey&Schleimer when they dealt with hyperbolic groups.

For G rel. hyperbolic: relating the geometry of Γ and $\widehat{\Gamma}$

For relatively hyperbolic G, we have to deal with the fact that negative curvature is visible in $\widehat{\Gamma}$, over the infinite set $\widehat{X}$, rather than in Γ, over X.
We have to relate Γ and $\widehat{\Gamma}$, and paths w and $\widehat{w}$ within them.
We find a good normal form nf() for G via an **asynchronous biautomatic structure**, i.e. a regular set L of words over X (recognised by an FSA), one rep. per group element, satisfying an **asynchronous fellow traveller property** (appropriate paths in Γ labelled by u, v within L for which $u =_G vx$ or $u =_G xv$ must fellow-travel asynchronously).

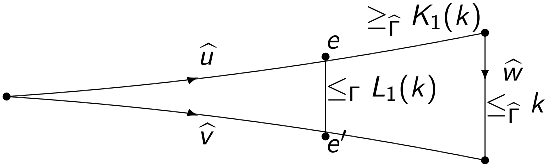


For $u \in L$, η short, can find reps of $u\eta$ and ηu in L quickly.

Synch. biautomatic structures for rel. hyp. groups were built by Antolin& Ciobanu (2016), but don't have all properties we need. So we build our own asynch. structure, for **very well chosen** X, with $\widehat{\mathrm{nf}(w)}$ geodesic in $\widehat{\Gamma}$, each H_i component of $\mathrm{nf}(w)$ within a specified biautomatic structure for H_i, and more ... (e.g. $u = \mathrm{nf}(u)$ for appropriate $u \subseteq \mathrm{nf}(w)$).

Relating the geometry of Γ and $\widehat{\Gamma}$ (2)

Where $\widehat{u}, \widehat{v}$ are geodesic, and $uw =_G v$, with $|\widehat{w}| \leq k$, then $\exists K_1(k), L_1(k)$ st, for any vertex e at distance at least $K_1(k)$ from the end of $\widehat{u}$, $\exists$ a vertex e' on $\widehat{v}$ with $d_\Gamma(e, e') \leq L_1(k)$; we say that e, e' are **corresponding** vertices.



It's important that corresponding vertices e, e' are close wrt the metric of Γ, not just wrt the metric of $\hat{G}$. That's because our constructions are with SLPs over X rather than SLPs over $\widehat{X}$.

The poly-time constituents of our construction

Given SLPs $\mathcal{G}_1, \mathcal{G}_2$ it's straightforward (and very quick) to compute an SLP with value $\mathtt{val}(\mathcal{G}_1)\mathtt{val}(\mathcal{G}_2)$.

We use various further constructions that can be done in poly-time, that is, whose time is bounded by a function of the size $|\mathcal{G}|$ of an input SLP.

In poly-time, by standard SLP results, given an SLP $\mathcal{G}$ for w we can

- compute w,
- test if w is in the language of a specified *fsa*,
- construct $\mathcal{G}'$ with value w that is **trimmed** (no unnecessary variables), and in Chomsky form,
- for any $i, j, k, l \geq 0$, $i \leq j$, $k \leq l$, construct $\mathcal{G}[i : j)$ or $\mathcal{G}[[k : l))$, with value $w[i : j)$ or $w[[k : l))$,
- test if $\mathcal{G}$ has the same value as a second SLP, $\mathcal{H}$.

The polynomial-time constituents (2)

We need specific poly-time constructions that deal with an SLP $\mathcal{G}$ for $\texttt{fg}$ G that is rel. hyp, or (free) abelian (to deal with parabolics).

In poly-time, by our results, given an SLP $\mathcal{G}$ for fg G, with value w,

if G is fg abelian over Z, we can

- construct a **compact** SLP $\mathcal{G}'$ over $Y \supseteq X$ with value $\mathsf{slex}(w)$ (we call $\mathcal{G}'$ compact if $|\mathcal{G}'| \leq \max(C\log(|\mathtt{val}(\mathcal{G}')|), 1)$),

or if G is rel. hyp, we can pause

- modify $\mathcal{G}$ to an SLP $\mathcal{G}'$ with value w, st
(1) every component of w has a **root** in $\mathcal{G}'$, i.e. is $\mathtt{val}_{\mathcal{G}'}(A)$, some A,
(2) given $\mathcal{G}'$, can easily write down an SLP for $\widehat{w}$,
- construct an SLP with value $\mathsf{nf}(w)$, of size at most $C|\widehat{w}|\log(|w|)$, in time bounded by a polynomial in $|\widehat{w}|$ and $|\mathcal{G}|$. (This is a **useful lemma** for when $|\widehat{w}|$ is bounded.)

Poly time construction of SLP for $\mathsf{nf}(\mathtt{val}(\mathcal{G}))$, for G relhyp

Input: an SLP $\mathcal{G}$ for G, over a 'nice' generating set X.

We imitate HLS's construction for hyperbolic groups, with 3 poly-time steps, each built out of poly-time components, mostly basic ops on SLPs.

Step 1: construct a nice 'non-splitting' TCSLP $\mathcal{T}$, specified rel. to compression, st $\mathtt{val}(\mathcal{T}) = \mathsf{nf}(\mathtt{val}(\mathcal{G}))$, $J_{\mathcal{T}} \leq L$, $|\mathcal{T}| \leq p_1(|\mathcal{G}|)$.

Non-splitting means: if $\rho_{\mathcal{T}}(A) = BC$ then $\mathtt{val}_{\mathcal{T}}(B), \mathtt{val}_{\mathcal{T}}(C)$ don't split components of $\mathtt{val}_{\mathcal{T}}(A)$.

Step 2: construct a 'non-splitting' TSLP $\mathcal{U}$ st $\mathtt{val}(\mathcal{U}) = \mathtt{val}(\mathcal{T})$, $J_{\mathcal{U}} \leq L$, $|\mathcal{U}| \leq p_2(|\mathcal{T}|)$.

Step 3: construct an SLP $\mathcal{S}$ st $\mathtt{val}(\mathcal{S}) = \mathtt{val}(\mathcal{U})$, $|\mathcal{S}| \leq p_3(|\mathcal{U}|)$.

The point of the route SLP $\to$ TCSLP $\to$ TSLP $\to$ SLP (inherited from HLS) is to limit size and time.

All 3 steps imitate the analogous steps in HLS' construction. I'll focus here on Step 1. The basic problem is to use the hyperbolic geometry of $\widehat{\Gamma}$ within Γ.

Step 1: constructing a TCSLP accepting $\mathsf{nf}(w)$

Input: an SLP $\mathcal{G}$ over X generating w, and a big enough integer L.

We can assume that X is **nice**, so that we have a **nice** asynchronous biautomatic structure giving normal forms $\mathsf{nf}()$.

And we can assume that $\mathcal{G}$ is in Chomsky form, **trimmed**, and that any **component** of w (maximal H_i subword) has the form $\mathtt{val}(A)$ for a variable A.

Aim: to construct a TCSLP $\mathcal{T}$ generating $\mathsf{nf}(w)$, with $J_\mathcal{T} \leq L$, which is nf-reduced (for any A, $\mathtt{val}_\mathcal{T}(A) = \mathsf{nf}(\mathtt{val}_\mathcal{T}(A))$).

The procedure:

Work through variables of $\mathcal{G}$ in order of increasing height, and use induction on height.

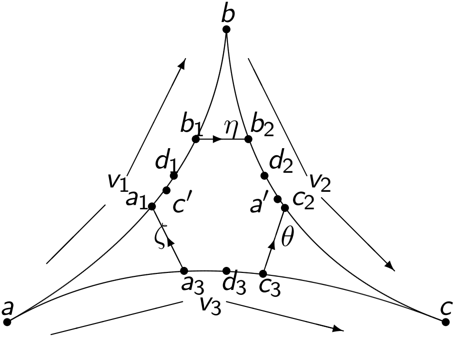
If $\mathtt{ht}(A) = 1$, then $\rho_\mathcal{T}(A) = \mathsf{nf}(\rho_\mathcal{G}(A))$.

Otherwise, $\rho_\mathcal{G}(A) = BC$, where $\mathtt{ht}(B), \mathtt{ht}(C) < \mathtt{ht}(A)$, and by induction we can find $\mathcal{T}_B, \mathcal{T}_C$ in poly-time with values $\mathsf{nf}(\mathtt{val}_\mathcal{G}(B)), \mathsf{nf}(\mathtt{val}_\mathcal{G}(C))$.

Step 1: modifying production on A when $\rho_{\mathcal{G}}(A) = BC$

We have $v_1 = \mathrm{nf}(\mathtt{val}_{\mathcal{G}}(B))$, $v_2 = \mathrm{nf}(\mathtt{val}_{\mathcal{G}}(C))$, and **nice** SLPs $\mathcal{S}_B$, $\mathcal{S}_C$, derived in poly-time (Steps 2,3) from $\mathcal{T}_B, \mathcal{T}_C$ with values v_1, v_2.

Where $v_3 := \mathrm{nf}(v_1 v_2)$, we consider the hyperbolic triangle in $\widehat{\Gamma}$ with vtces a, b, c, sides the paths $\gamma_{\widehat{v_1}}, \gamma_{\widehat{v_2}}, \gamma_{\widehat{v_3}}$, joining a to b, b to c, and a to c.

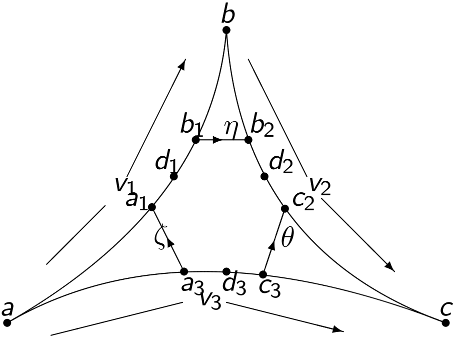


The triangle is 'thin', meeting pts d_i (close in $\widehat{\Gamma}$). For e on $\gamma_{\hat{v_1}}$ (or $\gamma_{\hat{v_2}}$), we use $\mathcal{S}_B$ & $\mathcal{S}_C$ **to find** corresp. e' on $\gamma_{\hat{v_2}}$ (or $\gamma_{\hat{v_1}}$), $d_\Gamma(e, e') \leq L$, if e' exists. In poly-time, **we find either** (1a) a' on $\gamma_{\widehat{v_2}}$ corresp. to a on $\gamma_{\widehat{v_1}}$, **or** (1b) c' on $\gamma_{\widehat{v_1}}$ corresp. to c on $\gamma_{\widehat{v_2}}$, **or** (2) corresp. b_1, b_2 (maximising $d_{\hat{\Gamma}}(b, b_1)$), st $|\eta|_X \leq L$ and a_1, c_2 corresp. to a_3, c_3 via ζ, θ st $|\zeta|_X, |\theta|_X \leq L$.

In each case, we then construct a TCSLP for v_3, combining concatenations, cuts and tethering ops. on $\mathcal{S}_B$, $\mathcal{S}_C$, as follows.

Step 1: Building the TCSLP for v_3 (in case 2 above)

Our choice of b_1, b_2 ensures b_1 close to d_1, then **we locate** a_1 the other side of d_1 on $\gamma_{\hat{v}_1}$. The words ζ, θ are found via exhaustive searches. The construction verifies when they're correct.



For a given selection, suppose that $v_1[[k_1 : l_1))$ and $v_2[[k_2 : l_2))$ are subwords of v_1, v_2 from a_1 to b_1, b_2 to c_2. In poly time we construct SLPs $\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_3$, out of $\mathcal{S}_B$ and $\mathcal{S}_C$, whose values are the words $\mathrm{nf}(v_1[[: k_1))\zeta^{-1})$, $\mathrm{nf}(\zeta v_1[[k_1 : l_1))\eta v_2[[k_2 : l_2))\theta^{-1})$ and $\mathrm{nf}(\theta v_2[[l_2 :)))$. In poly-time we check if $\mathcal{S}_1\mathcal{S}_2\mathcal{S}_3$ is nf-reduced. If so, it has value v_3, so chosen ζ, θ are correct.

We need a TCSLP (smaller) for v_3, not an SLP, so now with this ζ, θ, we construct $\mathcal{T}_1 := \mathcal{T}_B[[: k_1))\langle\varepsilon, \zeta\rangle$ and $\mathcal{T}_3 := \mathcal{T}_C[[l_2 :))\langle\theta, \varepsilon\rangle$ as single variable extensions of $\mathcal{T}_B, \mathcal{T}_C$, and insert the TCSLP $\mathcal{T}_1\mathcal{S}_2\mathcal{T}_3$ into $\mathcal{T}$ to define $\rho_{\mathcal{T}}(A)$.

Steps 2 and 3: converting a TCSLP $\mathcal{T}$ to a TSLP $\mathcal{U}$ and then an SLP $\mathcal{S}$, each in poly-time

For **Step 2,** $\mathcal{T} \to \mathcal{U}$, we imitate HLS's proof for hyperbolic groups, which in turn imitates Hagenah's construction of an SLP from a CSLP.

We process variables A of $\mathcal{T}$ in order of increasing height, eliminate productions $\rho_{\mathcal{T}}(A)$ that involve cut operators, adding at most $\mathtt{ht}(\mathcal{T})$ new variables, pushing cut ops. towards lower ht. variables, use induction.

For **Step 3,** $\mathcal{U} \to \mathcal{S}$, again we imitate HLS's proof for hyperbolic groups.

We process variables of $\mathcal{U}$ in order of increasing height. As each variable A of $\mathcal{U}$ is processed, either we define a new copy of A within $\mathcal{S}$, or a set of at most L^2 new variables.

Of course we need the negative curvature of $\widehat{\Gamma}$ (and its relationship to Γ) to make these steps work.

What more can we do?

We'd like to be able to deal with a wider class of parabolic subgroups.

We ought to be able to generalise to parabolics that are abelian with torsion; then we have to deal with the possibility $|H_i \cap H_j| > 1$.

The arguments that deal with free abelian are already very technical, but we believe it should be possible to extend them to allow torsion in abelian parabolics.

But generalising to virtually abelian parabolics might be impossible, because of the difficulty of constructing an appropriate asynchronously biautomatic structure wrt the right generating set.

We can also solve $\mathtt{CCP}(G)$ for G rel hyp in poly time

How? Given input SLPs $\mathcal{G}_1, \mathcal{G}_2$, we should either find $\mathcal{G}$ s.t

$$\mathtt{val}(\mathcal{G})\mathtt{val}(\mathcal{G}_2)\mathtt{val}(\mathcal{G})^{-1} =_G \mathtt{val}(\mathcal{G}_1)$$

or report that no such $\mathcal{G}$ exists.

For G hyp., HLS solved this as a conversion to compressed setting of the Epstein&Holt (2006) lin. time soln. to $\mathtt{CP}(G)$. That algorithm converts because reduces to testing if one word is a cyclic conjugate of another; other algorithms examine all cyclic conjugates of one or both input words.

We can solve this for rel. hyp. G, using similar methods.

- We use ‘look-up tables’ to deal with the cases where the words $u = \mathtt{val}(\mathcal{G}_1)$ and $v = \mathtt{val}(\mathcal{G}_2)$ are short.
- If both derived words $\widehat{u}, \widehat{v}$ are short, we use the Antolin&Ciobanu (2016) solution of $\mathtt{CP}(G)$ for G rel. hyp; if conjugators exist, we find one via a **minimal bounded conjugacy diagram**
- If at least one of $\widehat{u}, \widehat{v}$ is longer, we imitate HLS, adapting EH algorithm to compressed setting, in this case for rel. hyp G.

Lin. time soln. to $\mathtt{CP}(G)$ for G hyp. (Epstein&Holt, 2006)

Given u, v find g st $gvg^{-1} =_G u$ $(u \sim_G v)$ or report that $\nexists g$ $(u \not\sim_G v)$.

- Step 1 In linear time, find $\mathrm{slex}(u), \mathrm{slex}(v)$, replace u, v by these.
- Step 2 In linear time, replace u, v by $\mathrm{slex}(u_c), \mathrm{slex}(v_c)$, where u_c, v_c are cyclic conjugates of u, c through half their lengths. Now all powers of u, v are L-local quasigeodesics, for some L.
- Step 3 In linear time, find h, M st $z := hu^M h^{-1}$ is slex-straight (Delzant); w is **slex-straight** if for all $k > 0$, $w^k = \mathrm{slex}(w^k)$.
- Step 4 In linear time, test if $\exists$ short h' st $(v^M)^{h'} =_G$ a cyclic conjugate z^{z_1} of z.
  If **no**, then $u^M \not\sim_G v^M$ and so $u \not\sim_G v$.
  If **yes**, replace v by $\mathrm{slex}(v^{h' z_1^{-1}})$, so that $u^M =_G v^M$. Then $u \sim_G v \iff u \sim_{C_G(z)} v$.
- Step 5 Check whether $u =_{C_G(z)} v^g$ for g from a bounded set of potential conjugators.

For G relhyp, we do much the same, with slex and slex-straight replaced by nf and nf-straight.