Parallel Hyperedge Replacement

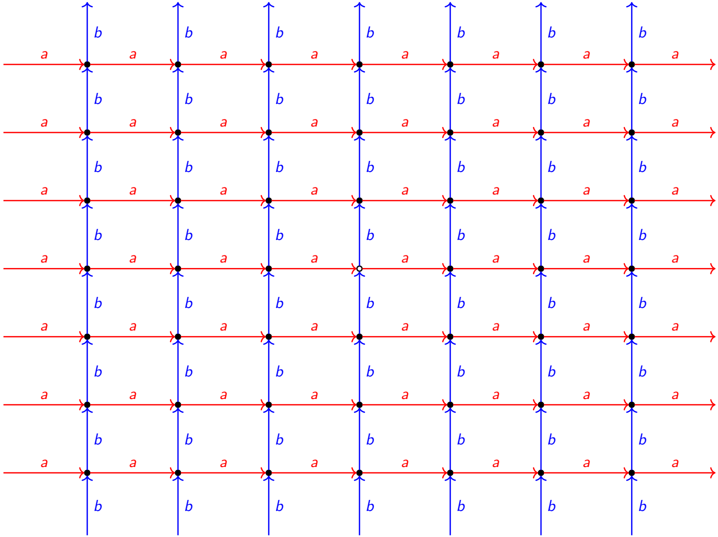
Newcastle Geometry & Algebra Seminar

Graham Campbell

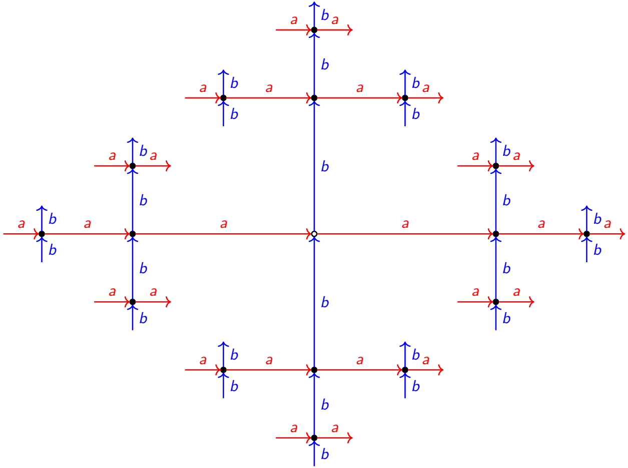
School of Mathematics, Statistics and Physics, Newcastle University, UK

May 2020

Cayley Graph of $\mathbb{Z}^2$



Cayley Graph of F_2



The Word Problem

Definition 1 (Word Problem)

Given a group presentation $\langle X \mid R \rangle$ for a group G, the word problem is the membership problem for the string language:

$$\mathrm{WP}_X(G) = \{w \in (X \cup X^{-1})^* \mid w =_G 1_G\}.$$

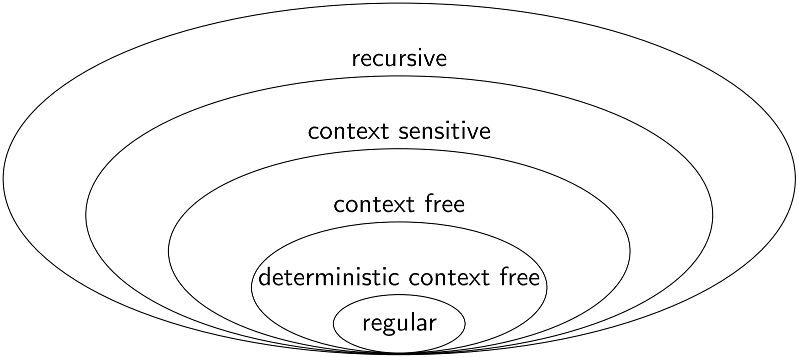
Question: How hard is the word problem?

Answer: It is impossible! There is a finite group presentation with an undecidable word problem (Novikov 1955).

Question: The Chomsky hierarchy is a notion of complexity of a language. Can we characterise the finitely generated groups in terms of language families?

Answer: Maybe...

Context-Free Groups



Theorem 2 (Anisimov (1971) and Muller and Schupp (1983))

1. A presentation defines a finite group iff it has regular WP;
2. A presentation defines a virtually free group iff it has (D)CF WP.

Multiple Context-Free

Question: how “far away” are interesting families of groups from the context-free languages in the formal language formal language hierarchy?

Multiple context-free (MCF) languages are a conservative extension of the context-free languages, introduced in the late 80s (Seki et al. 1991).

Theorem 3 (Ho (2018))

If a presentation defines a virtually Abelian group, then it has MCF WP.

Theorem 4 (Gilman, Kropholler, and Schleimer (2018))

The fundamental group of a hyperbolic three-manifold does not admit a MCF WP.

Theorem 5 (Engelfriet and Heyker (1991) and Weir (1992))

The MCF languages are exactly the string languages generated by HR grammars (Drewes, Kreowski, and Habel 1997).

Lindenmayer Systems

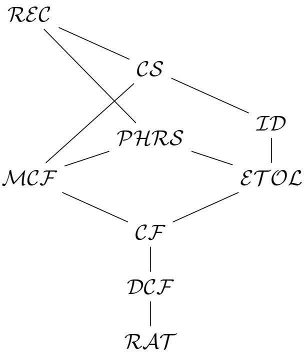
There lots of other well-behaved language classes sitting in between the CF and CS classes, such as the indexed languages (Aho 1968) and their subclass of ET0L languages (Rozenberg and Salomaa 1980).

It is not known if there are any groups with indexed word problems other than the virtually free groups.

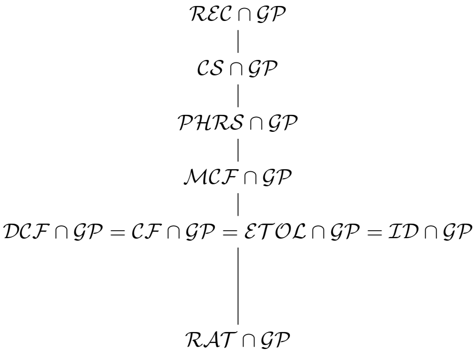
In particular, we don't know if any hyperbolic groups (other than the virtually free groups) have ET0L word problems (Ciobanu, Elder, and Ferov 2018), such as the fundamental group of the double torus.

What if we tried to mix together ideas from MCF and ET0L... parallel hyperedge replacement! Do the word problems of hyperbolic groups lie within this class? I should mention forms of parallel HR have been considered before (Habel 1992; Kreowski 1993), the work is not extensive and does not consider rational control or string generational power.

Taking Stock

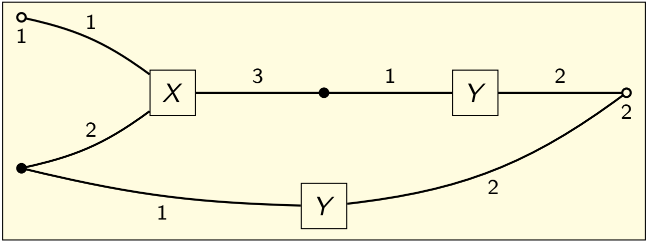


(a) Proved String Language Hierarchy



(b) Conjectured Group Language Hierarchy

Hypergraphs I



Hypergraphs II

Definition 6 (Signature)

A signature is a pair $\mathcal{C} = (\Sigma, \mathrm{type})$ where Σ is some finite label set, and $\mathrm{type} : \Sigma \to \mathbb{N}$ is a typing function which assigns to each label an arity.

Definition 7 (Hypergraph)

A hypergraph over $\mathcal{C}$ is a tuple $H = (V_H, E_H, \mathrm{att}_H, \mathrm{lab}_H, \mathrm{ext}_H)$ where:

1. V_H is a finite set of nodes;
2. E_H is a finite set of hyperedges;
3. $\mathrm{att}_H : E_H \to \mathrm{iseq}(V_H)$ is the attachment function;
4. $\mathrm{lab}_H : E_H \to \Sigma$ is the labelling function;
5. $\mathrm{ext}_H : \mathrm{iseq}(V_H)$ are the external nodes;

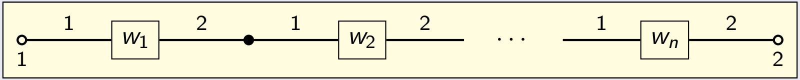
such that labelling is compatible with typing $(\mathrm{type} \circ \mathrm{lab}_H = |\cdot| \circ \mathrm{att}_H)$.

The class of all hypergraphs over $\mathcal{C}$ is denoted $\mathcal{H}_\mathcal{C}$.

Strings and Handles

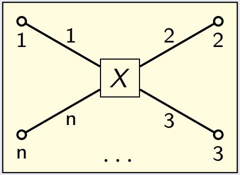
Definition 8 (String Graph)

Given a non-empty word $w = w_1 w_2 \cdots w_n$, we define its string graph $w^\bullet$:

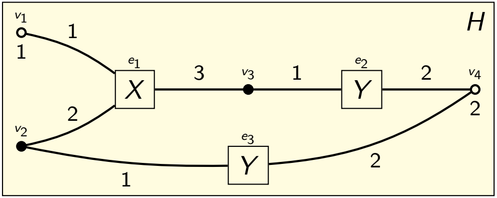


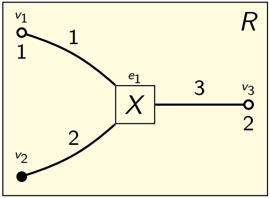
Definition 9 (Handle)

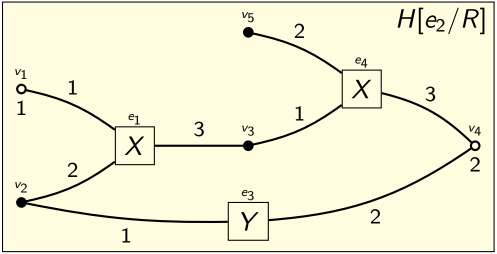
Given a label X of type n, we define its handle $X^\bullet$:



Replacement







Rules and Derivations

Let $\mathcal{C} = (\Sigma, \mathrm{type})$ be a signature and $N \subseteq \Sigma$ a set of non-terminals.

Definition 10 (Rule)

A rule over N is a pair (L, R) with $L \in N$, $R \in \mathcal{H}_{\mathcal{C}}$, $\mathrm{type}(L) = \mathrm{type}(R)$.

Definition 11 (Direct Derivation)

Given $H \in \mathcal{H}_{\mathcal{C}}$ and $\mathcal{R}$ a set of rules, if $e \in E_H$ and $(\mathrm{lab}_H(e), R) \in \mathcal{R}$, then we say that H directly derives $H' \cong H[e/R]$, and write $H \Rightarrow_{\mathcal{R}} H'$.

Definition 12 (Derivation)

H derives H' if there is a sequence $H \Rightarrow_{\mathcal{R}} H_1 \Rightarrow_{\mathcal{R}} \cdots \Rightarrow_{\mathcal{R}} H_k$ for some $k \in \mathbb{N}$, with $H_k = H'$. We write $H \Rightarrow_{\mathcal{R}}^k H'$ or $H \Rightarrow_{\mathcal{R}}^* H'$.

HR Grammars

Definition 13 (HR Grammar)

A HR grammar of order k is a system $\mathcal{G} = (\mathcal{C}, N, S, \mathcal{R})$ where:

1. $\mathcal{C} = (\Sigma, \mathrm{type})$ is a signature;
2. $N \subseteq \Sigma$ is the set of non-terminal labels;
3. $S \in N$ is the start symbol;
4. $\mathcal{R}$ is a finite set of rules over N;

with $\max(\{|\mathrm{type}(R)| \mid (L, R) \in \mathcal{R}\}) \leq k$. The generated language is:

$$\mathrm{L}(\mathcal{G}) = \{H \in \mathcal{H}_{\mathcal{C}} \mid S^\bullet \Rightarrow_{\mathcal{R}}^* H \text{ with } \mathrm{lab}_H^{-1}(N) = \emptyset\} \subseteq \mathcal{H}_{\mathcal{C}}.$$

$L \subseteq \mathcal{H}_{\mathcal{C}}$ is called a HR language of order k (k-HR language) if there is a k-HR grammar such that $\mathrm{L}(\mathcal{G}) = L$.

The class of HR languages is the union of all k-HR languages for $k \in \mathbb{N}$.

HRS Languages

Definition 14 (String Graph Language)

A hypergraph language that only consists of string graphs is called a string graph language. We can identify the language members with the strings they represent.

Given a HR grammar $\mathcal{G}$ that generates a string graph language, we write $\mathrm{STR}(\mathrm{L}(\mathcal{G}))$ for the actual string language it generates.

Thus, a string language L is called a HRS language if, up to treatment of the empty string, there is a HR grammar that generates the language of string graphs that represent exactly the strings in L. The class of HRS languages is the union of all k-HRS languages for $k \in \mathbb{N}$.

Theorem 15 (Precise Theorem 5)

For all $k \geq 1$, $\mathcal{HRS}_{2k} = \mathcal{HRS}_{2k+1} = \mathcal{MCF}_k$.

Parallel Derivations

Definition 16 (Parallel Direct Derivation)

Let $H \in \mathcal{H}_\mathcal{C}$ with $E_H = \{e_1, \dots e_n\}$, and $\mathcal{R}$ be a set of rules. If for each $e_i \in E_H$, there is an $R_i \in \mathcal{H}_\mathcal{C}$ such that $(\mathrm{lab}_H(e_i), R_i) \in \mathcal{R}$, then H parallelly directly derives $H' \cong H[e_1/R_1] \cdots [e_n/R_n]$, and write $H \Rrightarrow_\mathcal{R} H'$.

Definition 17 (Parallel Derivation)

Let $\mathcal{S} = \{\mathcal{R}_i \mid i \in I\}$ be a finite set of rule sets indexed by I, and $\mathcal{M}$ an FSA over I. Then H ($\mathcal{M}$-)parallelly derives H' if there is a sequence $H \Rrightarrow_{\mathcal{R}_{i_1}} H_1 \Rrightarrow_{\mathcal{R}_{i_2}} \cdots \Rrightarrow_{\mathcal{R}_{i_k}} H_k$ ($k \in \mathbb{N}$) such that $i_1 i_2 \cdots i_k \in \mathrm{L}(\mathcal{M})$ and $H' = H_k$. We write $H \Rrightarrow_\mathcal{S}^\mathcal{M} H'$, $H \Rrightarrow_\mathcal{S}^{i_1 i_2 \cdots i_k} H'$ or $H \Rrightarrow_\mathcal{S}^k H'$.

Definition 18 (Table)

A table T is a finite set of rules over Σ such that for each $L \in \Sigma$, there is at least one $R \in \mathcal{H}_\mathcal{C}$ such that $(L, R) \in T$.

PHR Grammars

Definition 19 (PHR Grammar)

A (k-PHR grammar is a system $\mathcal{G} = (\mathcal{C}, A, S, \mathcal{T}, \mathcal{M})$ where:

1. $\mathcal{C} = (\Sigma, \mathrm{type})$ is a signature;
2. $A \subseteq \Sigma$ is the set of terminal labels;
3. $S \in \Sigma \setminus A$ is the start symbol;
4. $\mathcal{T} = \{T_i \mid i \in I\}$ is a finite set of tables indexed by I;
5. $\mathcal{M} = (Q, I, \delta, i, F)$ is an FSA over I;

with $\max(\{|\mathrm{type}(R)| \mid (L, R) \in \bigcup_{T_i \in \mathcal{T}} T_i\}) \leq k$. The generated language is:

$$\mathrm{L}(\mathcal{G}) = \{H \in \mathcal{H}_\mathcal{C} \mid S^\bullet \Rrightarrow_\mathcal{T}^\mathcal{M} H \text{ with } \mathrm{lab}_H^{-1}(A) = E_H\} \subseteq \mathcal{H}_\mathcal{C}.$$

L is called a k-PHR language if there is a k-PHR grammar $\mathcal{G}$ s.t. $\mathrm{L}(\mathcal{G}) = L$. The class of PHR languages is the union of all k-PHR languages.

Control Removal

Inspired by Asveld (1977) and Ciobanu, Elder, and Ferov (2018) for ET0L grammars.

Definition 20 (PHR Grammar Without Control)

A k-PHR grammar without control is a tuple $\mathcal{G} = (\mathcal{C}, A, S, \mathcal{T})$ such that $(\mathcal{C}, A, S, \mathcal{T}, \mathcal{M})$ is a k-PHR grammar where $\mathcal{M}$ is an FSA which accepts everything. Its generated language is defined in the obvious way.

Lemma 21 (Control Removal)

Given a k-PHR grammar $\mathcal{G}$, one can effectively construct a k-PHR grammar $\mathcal{G}'$ without control such that $\mathrm{L}(\mathcal{G}) = \mathrm{L}(\mathcal{G}')$.

Proof: Encode control in the labels! Make a copy of all the labels for all of the states in the FSA, and moving between control states is synchronized with moving between the copies of labels. □

HR Generalisation

Proposition 22

$L = \{a^{2^n} \mid n \in \mathbb{N}\}$ is an ET0L language but not MCF. Moreover, it is not semilinear.

Proof: It is easy to see that $\mathcal{G} = (\{a\}, \{a\}, a, \{\{(a, aa)\}\})$ is an ET0L grammar with $\mathrm{L}(\mathcal{G}) = L$. Recall that a language is semilinear if and only if it is letter-equivalent to a regular language (Parikh 1966). Since L is a language on only one symbol it must be semilinear if and only if it is a regular language, but clearly it is not a regular language! But all MCF languages are semilinear, so it must be the case that L is not. □

Theorem 23 (PHR Generalises HR)

For $k \geq 0$, $\mathcal{HR}_k \subsetneq \mathcal{PHR}_k$.

Proof: Inclusion is by induction on derivation length, simulating sequential derivations. Strictness (roughly!) by Proposition 22. □

PHRS I

We define k-PHRS languages in the obvious way. This our new interesting string language class!

Proposition 24

$\mathcal{HRS}_0 = \mathcal{HRS}_1 = \mathcal{PHRS}_0 = \mathcal{PHRS}_1 = \{\emptyset, \{\epsilon\}\}$.

Proposition 25

Given a k-PHR grammar $\mathcal{G}$ which generates a string language, then one can effectively construct a k-PHR grammar $\mathcal{G}'$ such that all labels have type at least 2 and $\mathrm{L}(\mathcal{G}) = \mathrm{L}(\mathcal{G}')$.

Lemma 26 (PHRS Generalises ET0L)

$\mathcal{ET0L} = \mathcal{PHRS}_2$ and for $k \geq 4$, $\mathcal{ET0L} \subsetneq \mathcal{PHRS}_k$.

Corollary 27

There are 2-PHRS languages that are not semilinear.

PHRS II

Lemma 28 (PHRS Generalises MCF)

For $k \geq 2$, $\mathcal{HRS}_k \subsetneq \mathcal{PHRS}_k$.

Proof: The equivalence of MCF and HRS due to Theorem 15. To see the remainder follows from Theorem 23 and its proof. We get strictness from Proposition 22 together with Lemma 26. □

Conjecture 29 (PHRS Refines CS)

$\mathcal{PHRS} \subsetneq \mathcal{CS}$.

Conjecture 30 (Substitution-Closed Full AFL)

For $k \geq 2$, $\mathcal{PHRS}_k$ and $\mathcal{PHRS}$ are substitution-closed full abstract families of languages.

PHRS III

If Conjecture 30 is true, then we have the useful corollary (which is also true of regular, (D)CF, MCF and ET0L languages:

Corollary 31

For $k \geq 2$, $\mathcal{PHRS}_k$ and $\mathcal{PHRS}$ are closed under inverse homomorphisms. Moreover, if a group has a PHRS word problem for some given presentation, then all presentations necessarily do.

And finally:

Conjecture 32 (WP Double Torus)

The fundamental group of the double torus admits a PHRS word problem with is neither a MCF nor ET0L language.

More in Campbell (2020), submitted to TERMGRAPH 2020.

References I

Aho, Alfred (1968). "Indexed Grammars – An Extension of Context-Free Grammars". In: *Journal of the ACM* 15.4, pp. 647–671. DOI: 10.1145/321479.321488.

Anisimov, Anatoly (1971). "Group Languages". In: *Kibernetika* 4, pp. 18–24.

Asveld, Peter (1977). "Controlled iteration grammars and full hyper-AFL's". In: *Information and Control* 34.3, pp. 248–269. DOI: 10.1016/S0019-9958(77)90308-4.

Campbell, Graham (2020). *Parallel Hyperedge Replacement String Languages*. Submitted for publication. School of Mathematics, Statistics and Physics, Newcastle University, UK. URL: https://cdn.gjcampbell.co.uk/2020/PHRS-Languages-Preprint.pdf.

Ciobanu, Laura, Murray Elder, and Michal Ferov (2018). "Applications of L systems to group theory". In: *International Journal of Algebra and Computation* 28.2, pp. 309–329. DOI: 10.1142/S0218196718500145.

Drewes, Frank, Hans-Jörg Kreowski, and Annegret Habel (1997). "Hyperedge Replacement Graph Grammars". In: *Handbook of Graph Grammars and Computing by Graph Transformation, Volume 1: Foundations*. World Scientific, pp. 95–162. DOI: 10.1142/9789812384720_0002.

Engelfriet, Joost and Linda Heyker (1991). "The string generating power of context-free hypergraph grammars". In: *Journal of Computer and System Sciences* 43.2, pp. 328–360. DOI: 10.1016/0022-0000(91)90018-Z.

Gilman, Robert, Robert Kropholler, and Saul Schleimer (2018). "Groups whose word problems are not semilinear". In: *Groups Complexity Cryptology* 10.2, pp. 53–62. DOI: 10.1515/gcc-2018-0010.

Habel, Annegret (1992). *Hyperedge Replacement: Grammars and Languages*. Vol. 643. Lecture Notes in Computer Science. Springer. DOI: 10.1007/BFb0013875.

References II

Ho, Meng-Che (2018). "The word problem of $\mathbb{Z}^n$ is a multiple context-free language". In: *Groups Complexity Cryptology* 10.1, pp. 9–15. DOI: 10.1515/gcc-2018-0003.

Kreowski, Hans-Jörg (1993). "Five facets of hyperedge replacement beyond context-freeness". In: *Proc. 9th International Conference on Fundamentals of Computation Theory (FCT 1993)*. Ed. by Zoltán Ésik. Vol. 710. Lecture Notes in Computer Science, pp. 69–86. DOI: 10.1007/3-540-57163-9_5.

Muller, David and Paul Schupp (1983). "Groups, the Theory of Ends, and Context-Free Languages". In: *Journal of Computer and System Sciences* 26.3, pp. 295–310. DOI: 10.1016/0022-0000(83)90003-X.

Novikov, Pyotr (1955). "Über die algorithmische Unentscheidbarkeit des Wortproblems in der Gruppentheorie". In: *Trudy Matematicheskogo Instituta imeni V.A. Steklova* 44, pp. 1–143.

Parikh, Rohit (1966). "On Context-Free Languages". In: *Journal of the ACM* 13.4, pp. 570–581. DOI: 10.1145/321356.321364.

Rozenberg, Grzegorz and Arto Salomaa (1980). *The Mathematical Theory of L Systems*. Vol. 90. Pure and Applied Mathematics. Academic Press.

Seki, Hiroyuki et al. (1991). "On multiple context-free grammars". In: *Theoretical Computer Science* 88.2, pp. 191–229. DOI: 10.1016/0304-3975(91)90374-B.

Weir, David (1992). "Linear context-free rewriting systems and deterministic tree-walking transducers". In: *Proc. 30th Annual Meeting of the Assoc. for Comput. Linguist.* Pp. 136–143. DOI: 10.3115/981967.981985.