Higman type theorems for subshifts

Emmanuel Jeandel, Pascal Vanier

Laboratoire d’Algorithmique Complexité et Logique, UPEC
Puzzle

What could this be?

Given an infinite number of puzzle pieces can we tile infinitely in all directions with them?
This was a very hard puzzle...
This was a very hard puzzle...
Answer

This was a very hard puzzle...
A finite alphabet:

\[ \Sigma = \{\text{□}, \text{●}\} \]
Subshifts and subshifts of finite type

A finite alphabet:

\[ \Sigma = \{ \square, \Box \} \]

A tiling or configuration is a coloring of \( \mathbb{Z}^d \):
Subshifts and subshifts of finite type

A finite alphabet:

\[ \Sigma = \{ \text{red}, \text{blue} \} \]

A finite number of forbidden patterns:

\[ \mathcal{F} = \left\{ \text{red, red, red} \right\} \]

A tiling or configuration is a coloring of \( \mathbb{Z}^d \):

![Tiling example]
Subshifts and subshifts of finite type

A finite alphabet:

\[ \Sigma = \{\text{red}, \text{blue}\} \]

A finite number of forbidden patterns:

\[ \mathcal{F} = \{\text{red, red, blue}\} \]

A tiling or configuration is a coloring of \( \mathbb{Z}^d \):
A finite alphabet:

\[ \Sigma = \{\text{red, blue}\} \]

A finite number of forbidden patterns:

\[ \mathcal{F} = \{\text{red, blue, red}\} \]

A tiling or configuration is a coloring of \( \mathbb{Z}^d \):
Subshifts and subshifts of finite type

A finite alphabet:

\[ \Sigma = \{\text{red square, blue square}\} \]

A finite number of forbidden patterns:

\[ \mathcal{F} = \{ \text{red square, blue square, red square} \} \]

A tiling or configuration is a coloring of \( \mathbb{Z}^d \):
Subshifts and subshifts of finite type

A finite alphabet:

\[ \Sigma = \{ \square, \square \} \]

A finite number of forbidden patterns:

\[ F = \{ \square, \square, \square \} \]

A tiling or configuration is a coloring of \( \mathbb{Z}^d \):

3/30
Subshifts and subshifts of finite type

A finite alphabet:

\[ \Sigma = \{1,2\} \]

A finite number of forbidden patterns:

\[ F = \{12, 21, 123\} \]

A tiling or configuration is a coloring of \( \mathbb{Z}^d \):

\[ \text{tiling or configuration} \]
Subshifts and subshifts of finite type

A finite alphabet:

\[ \Sigma = \{\text{\large \textcolor{red}{\text{\textcircled{}}}}, \text{\large \textcolor{blue}{\text{\textcircled{}}}}\} \]

A finite number of forbidden patterns:

\[ \mathcal{F} = \left\{ \begin{array}{c}
\mathbf{1}, \mathbf{2}, \mathbf{3}
\end{array} \right\} \]

Subshift of finite type (SFT): set of configurations avoiding \( \mathcal{F} \). We note \( \mathcal{X}_\mathcal{F} \):

\[ \mathcal{X}_\mathcal{F} = \left\{ \begin{array}{c}
\begin{array}{c}
\text{\large \textcolor{red}{\text{\textcircled{}}}}, \text{\large \textcolor{blue}{\text{\textcircled{}}}}, \text{\large \textcolor{red}{\text{\textcircled{}}}}
\end{array}
\end{array} \right\} \]

A tiling or configuration is a coloring of \( \mathbb{Z}^d \):
Subshifts and subshifts of finite type

A finite alphabet:

\[ \Sigma = \{ \square, \blacksquare \} \]

A finite number of forbidden patterns:

\[ \mathcal{F} = \{ \text{patterns} \} \]

Subshift of finite type (SFT): set of configurations avoiding \( \mathcal{F} \). We note \( \mathcal{X}_\mathcal{F} : \)

\[ \mathcal{X}_\mathcal{F} = \{ \text{configurations} \} \]

The family may also be infinite we then talk about subshifts.
Another example

Let \( \mathcal{F} = \{ab, ba\} \) and \( \Sigma = \{a, b\} \):

\[
X_{\mathcal{F}} =
\cdots aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa \cdots
\]
\[
\cdots bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb \cdots
\]
Finitely generated groups

**Finitely generated group:** $S$ finite set of generators and $R$ a set of relations, $\langle S \mid R \rangle$ is the largest group generated by $S$ in which all of $R$ holds.

**Example:** $G = \langle a, b \mid aba^{-1}b^{-1} \rangle$

- $G$ is the largest group in which $ab = ba$, i.e. $aba^{-1}b^{-1} = 1$
- $G \simeq \mathbb{Z}^2$
Examples of groups and subshifts

Example 1: groups
• $G = \langle a, b \mid \rangle$: the free group with two generators.
• $G = \langle a, b \mid ab, ba \rangle$
  Reduced words of the form: $a^n, b^n$

Example 2: subshifts
• $\Sigma = \{a, b\}$ and $\mathcal{F} = \emptyset$: the full shift over two symbols.
• $\Sigma = \{a, b\}$ and $\mathcal{F} = \{ab, ba\}$
  Configurations: $\omega b^\omega, \omega a^\omega$
Notations

Alphabet and

- *Set of relations* defines a group.
- *Set of forbidden patterns* defines a subshift.

Group: \[ \langle S \mid R \rangle \]

Subshift of dimension \( d \): \[ \langle \Sigma \mid \mathcal{F} \rangle^d \]
Language/Word problem

More generally:

**Word Problem:**
\[ WP(G) = \{ w | w = 1_G \} \]

\( WP(G) \) is recursively enumerable from the set of relations.

\[ G = \langle S_G \mid WP(G) \rangle \]

**Complement of the language:**
\[ \mathcal{L}(X)^c = \{ m \mid \forall x \in X, m \not\in x \} \]

\( \mathcal{L}(X)^c \) is recursively enumerable from the set of forbidden patterns.

\[ X = \langle \Sigma_X \mid \mathcal{L}(X)^c \rangle \]
Similar definitions

**Finitely many** relations/forbidden patterns

- Subshifts of finite type
- Finitely presented groups

**Recursively enumerable** set of relations/forbidden patterns

- Effective subshifts
- Recursively presented groups

\( WP(G) \) and \( \mathcal{L}(X)^c \) are recursively enumerable in both cases.
1. Analogies

2. Higman embedding theorem

3. Relative Higman embedding theorem

4. Boone-Higman-Thompson theorem

5. Conclusion
Adding relations

Let $X = \langle A \mid R \rangle$ and $Y = \langle A \mid R \cup Q \rangle$:

**Groups:**
$Y$ is a quotient subgroup of $X$ by some normal subgroup.

**Subshifts:**
$Y$ is a subshift of $X$. 
Adding relations

$T = \langle A \mid R \rangle$ becomes trivial if we add any relation/pattern to $R$.

Groups:
$T$ is simple.

Subshifts:
$T$ is minimal.
Restricting

Let $T = \langle A \mid R \rangle$ and $S = \langle B \mid R \rangle$ with $B \subsetneq A$.

**Groups:**
- $S$ is a subgroup of $T$

Not all subgroups are of this form.

**Subshifts:**
- $\mathcal{L}(S) = \mathcal{L}(T) \cap B^d$
Restricting: example

\[ X = \left\{ a, b \left| (a\ a), (b\ b), \frac{a}{b}, \frac{b}{a} \right. \right\}^2 \]

\[ \ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots \]

\[ \ldots \ a \ b \ a \ b \ a \ b \ \ldots \]

\[ \ldots \ a \ b \ a \ b \ a \ b \ \ldots \]

\[ \ldots \ a \ b \ a \ b \ a \ b \ \ldots \]

\[ \ldots \ a \ b \ a \ b \ a \ b \ \ldots \]

\[ \ldots \ a \ b \ a \ b \ a \ b \ \ldots \]

\[ \ldots \ \ldots \ \ldots \ \ldots \ \ldots \ \ldots \ \ldots \ \ldots \]

\[ S \text{ s.t. } \mathcal{L}(S) = \mathcal{L}(X) \cap \{a, b\}^\ast : \]

\[ \cdots \text{ababababababababababababababab} \cdots \]

\[ S' \text{ s.t. } \mathcal{L}(S') = \mathcal{L}(X) \cap \{a\}^\ast : \]

\[ \emptyset \]
Free product

Definition  The **free product** of $F = \langle A \mid R \rangle$ and $G = \langle B \mid Q \rangle$ is:

$$F \ast G = \langle A \cup B \mid R \cup Q \rangle$$

Remark  **Adding symbols** corresponds to the **free product by a free group/full shift.**
**Free product: example**

\[ X = \langle a, b \mid (a \ a), (b \ b), \begin{pmatrix} a \\ b \end{pmatrix} \rangle^2 \]

\[ Y = \langle c, d \mid (c \ d), (d, c), \begin{pmatrix} c \\ d \end{pmatrix} \rangle^2 \]

\[ \in X \ast Y \]
The rosetta stone

<table>
<thead>
<tr>
<th><strong>Group $G$</strong></th>
<th><strong>Subshift $X$</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Group with $n$ generators</td>
<td>Subshift on $n$ symbols</td>
</tr>
<tr>
<td>Free group with $n$ generators</td>
<td>Full shift on $n$ symbols</td>
</tr>
<tr>
<td>Word problem $WP(G)$</td>
<td>co-language $L(X)^c$</td>
</tr>
<tr>
<td>Finitely presented group</td>
<td>SFT</td>
</tr>
<tr>
<td>Recursively presented group</td>
<td>Effectively closed subshift</td>
</tr>
<tr>
<td>Simple group</td>
<td>Minimal subshift</td>
</tr>
<tr>
<td>$G_1$ is a quotient of $G_2$</td>
<td>$X_1 \subseteq X_2$</td>
</tr>
<tr>
<td>$G_1 \subseteq G_2$ by restricting the generators</td>
<td>$L(X_1) = L(X_2) \cap \Sigma^*$</td>
</tr>
</tbody>
</table>

Analogies 16/30
Some decidability theorems

Theorem: It is **undecidable** whether a f.p. group is trivial.

Theorem: f.p. simple groups have **computable word problem**.

Theorem: It is **undecidable** whether an SFT is empty.

Theorem: A minimal SFT has **computable language**.
Take $X$ an effective subshift on some alphabet $\Sigma$.

There exists a **computable family of forbidden patterns** $\mathcal{F}$ generating it.

$X' = \langle \Sigma \cup \{\#\} | \mathcal{F} \rangle$ has a computable language:

Any pattern on $\Sigma$ containing no pattern of $\mathcal{F}$ may appear between $\#$.

The restriction of $X'$ to $\Sigma$ is $X$ which may not have a computable language.
1. Analogies

2. Higman embedding theorem

3. Relative Higman embedding theorem

4. Boone-Higman-Thompson theorem

5. Conclusion
Higman embedding theorem for groups

**Theorem** [Higman 1961] $G$ is **recursively presented** iff it is a subgroup of some **finitely presented** group $H$.

The proof is stronger actually: $G$ is obtained by restriction of $H$.

Statement reminiscent of:

**Theorem** [Hochman 2009, Aubrun Sablik 2013, Durand Romashchenko Shen 2011]
A **subshift** of dimension $d$ is **effective** iff it is a **projective subaction** of some **SFT** of dimension $d + 1$. 
Theorem  $X$ is an effective subshift iff it can be obtained by restriction of some SFT $Y$.

Proof.
AS 2013, DRS 2011 proof:

Small modification:

$X$ is then obtaining by restricting $Y$ to $X$’s alphabet.
1. Analogies

2. Higman embedding theorem

3. Relative Higman embedding theorem

4. Boone-Higman-Thompson theorem

5. Conclusion
Relative embedding theorem for groups

**Definition**  
$G$ is **finitely presented** over $H$ if it can be obtained by **adding finitely many generators and relations** to $H$.

**Definition**  
$A \leq_e B$ iff there exists $f : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{P}_{finite}(\mathbb{N})$ computable s.t.:

$$x \in A \iff \exists n \in \mathbb{N}, f(n,x) \subseteq B$$

$A$ is uniformly enumerable from any enumeration of $B$.

**Theorem** [Higman Scott 1988] $K$ is a subgroup of a finitely presented group over $G$ iff $WP(K) \leq_e WP(G)$.
Relative embedding theorem for subshifts

Definition \( X \) is an \textbf{SFT over} \( Y \) if it can be obtained by \textbf{adding finitely many symbols and forbidden patterns to} \( Y \).

Theorem \( X \) is a restriction of an SFT over \( Y \) iff \( \mathcal{L}(X)^c \leq_e \mathcal{L}(Y)^c \).
Proof idea

Make a subshift containing $\mathcal{L}(Y)$ in at least one configuration:

Insert lines of $X$ inbetween the lines: if a pattern $m$ is not allowed, then $\exists n, f(n, m) \notin \mathcal{L}(Y)$, forbid that.
1. Analogies

2. Higman embedding theorem

3. Relative Higman embedding theorem

4. Boone-Higman-Thompson theorem

5. Conclusion
For groups:

**Theorem** [Boone-Higman 1974, Thompson 1980]

\( WP(G) \) is computable iff \( G \) is a subgroup of a simple finitely presented group.

For subshifts:

**Theorem** \( \mathcal{L}(X) \) is computable iff \( X \) is a restriction of a minimal effective subshift.
Proof

Clear:

\[ X \text{ minimal effectively closed } \Rightarrow \mathcal{L}(X) \cap \Sigma^* \text{ computable}. \]

Not so clear:

\[ \mathcal{L}(X) \text{ computable } \Rightarrow \text{there exists } X' \text{ minimal effectively closed such that } \mathcal{L}(X) = \mathcal{L}(X') \cap \Sigma^*. \]
Proof: Two steps

Theorem A subshift $X$ has a computable language iff it is the restriction of some minimal effective subshift $Y$.

Theorem [Durand Romashchenko 2018] An minimal effective subshift can be realized as a subaction of some minimal SFT.
Minimality

**Definition** A subshift $X$ is **minimal** iff there is no subshift $Y$ s.t. $Y \subsetneq X$.

For each pattern there is a *window* in which it always appears.

**Example:**

![Diagram of a minimal subshift]

4. Boone-Higman-Thompson theorem
Minimality

**Definition** A subshift $X$ is **minimal** iff there is no subshift $Y$ s.t. $Y \subset X$.

For each pattern there is a *window* in which it always appears.

**Example:**
Minimality

**Definition** A subshift $X$ is **minimal** iff there is no subshift $Y$ s.t. $Y \subsetneq X$.

For each pattern there is a *window* in which it always appears.

Example:

4. Boone-Higman-Thompson theorem
Minimality

**Definition** A subshift $X$ is *minimal* iff there is no subshift $Y$ s.t. $Y \subsetneq X$.

For each pattern there is a *window* in which it always appears.

Example:
Definition  A subshift $X$ is **minimal** iff there is no subshift $Y$ s.t. $Y \subsetneq X$.

For each pattern there is a *window* in which it always appears.

Example:
Proof: minimal effective construction

<table>
<thead>
<tr>
<th></th>
<th>(w_1^1)</th>
<th>(w_0^1)</th>
<th>(w_1^1)</th>
<th>(w_0^1)</th>
<th>(w_1^1)</th>
<th>(w_0^1)</th>
<th>(w_1^1)</th>
<th>(w_0^1)</th>
<th>(w_1^1)</th>
<th>(w_0^1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(w_1^2)</td>
<td>#</td>
<td>(w_2^2)</td>
<td>#</td>
<td>(w_3^2)</td>
<td>#</td>
<td>(w_0^2)</td>
<td>#</td>
<td>(w_1^2)</td>
<td>#</td>
<td>(w_0^2)</td>
</tr>
<tr>
<td>#</td>
<td>(w_1^1)</td>
<td>#</td>
<td>(w_0^1)</td>
<td>#</td>
<td>(w_1^1)</td>
<td>#</td>
<td>(w_0^1)</td>
<td>#</td>
<td>(w_1^1)</td>
<td>#</td>
</tr>
<tr>
<td>#</td>
<td>#</td>
<td>(w_2^2)</td>
<td>#</td>
<td>(w_3^2)</td>
<td>#</td>
<td>(w_0^2)</td>
<td>#</td>
<td>(w_1^2)</td>
<td>#</td>
<td>(w_0^2)</td>
</tr>
<tr>
<td>#</td>
<td>#</td>
<td>(w_1^1)</td>
<td>#</td>
<td>(w_0^1)</td>
<td>#</td>
<td>(w_1^1)</td>
<td>#</td>
<td>(w_0^1)</td>
<td>#</td>
<td>(w_1^1)</td>
</tr>
<tr>
<td>#</td>
<td>#</td>
<td>#</td>
<td>(w_1^1)</td>
<td>#</td>
<td>(w_0^1)</td>
<td>#</td>
<td>(w_1^1)</td>
<td>#</td>
<td>(w_0^1)</td>
<td>#</td>
</tr>
<tr>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>(w_0^1)</td>
<td>#</td>
<td>(w_1^1)</td>
<td>#</td>
<td>(w_0^1)</td>
<td>#</td>
<td>(w_1^1)</td>
</tr>
<tr>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>(w_0^1)</td>
<td>#</td>
<td>(w_0^1)</td>
<td>#</td>
<td>(w_1^1)</td>
<td>#</td>
</tr>
<tr>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>(w_0^1)</td>
<td>#</td>
<td>(w_1^1)</td>
<td>#</td>
<td>(w_0^1)</td>
</tr>
<tr>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>(w_1^1)</td>
<td>#</td>
<td>(w_0^1)</td>
<td>#</td>
</tr>
<tr>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>#</td>
<td>(w_0^1)</td>
<td>#</td>
<td>(w_1^1)</td>
</tr>
</tbody>
</table>

4. Boone-Higman-Thompson theorem
Proof: minimal effective construction

<table>
<thead>
<tr>
<th>$w^1_1$</th>
<th>$w^0_0$</th>
<th>$w^1_1$</th>
<th>$w^1_0$</th>
<th>$w^1_0$</th>
<th>$w^1_1$</th>
<th>$w^0_0$</th>
<th>$w^1_0$</th>
<th>$w^1_0$</th>
<th>$w^1_0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$w^2_1$</td>
<td>$w^2_0$</td>
<td>$w^2_0$</td>
<td>$w^2_1$</td>
<td>$w^2_0$</td>
<td>$w^2_1$</td>
<td>$w^2_0$</td>
<td>$w^2_1$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$w^2_0$</td>
<td>$w^2_0$</td>
<td>$w^2_0$</td>
<td>$w^2_0$</td>
<td>$w^2_0$</td>
<td>$w^2_0$</td>
<td>$w^2_0$</td>
<td>$w^2_0$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$w^3_1$</td>
<td>$w^3_0$</td>
<td>$w^3_0$</td>
<td>$w^3_1$</td>
<td>$w^3_0$</td>
<td>$w^3_1$</td>
<td>$w^3_0$</td>
<td>$w^3_1$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

4. Boone-Higman-Thompson theorem
Proof: minimal effective construction

4. Boone-Higman-Thompson theorem
Proof: minimal effective construction

\[
\begin{array}{cccccccccccccccc}
\# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 \\
w_1^1 & \# & w_2^2 & \# & w_3^2 & \# & w_0^2 & \# & w_1^2 & \# & w_0^2 & \# & w_1^2 & \# & w_0^2 & \# & w_1^2 \\
\# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 \\
\# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 \\
\# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 \\
\# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 \\
\end{array}
\]

\[
\begin{array}{cccccccccccccccc}
\rightarrow & \# \\
\# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 \\
w_1^2 & \# & w_2^2 & \# & w_3^2 & \# & w_0^2 & \# & w_1^2 & \# & w_0^2 & \# & w_1^2 & \# & w_0^2 & \# & w_1^2 \\
\# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 \\
\# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 \\
\# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 \\
\# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 \\
\end{array}
\]

\[
\begin{array}{cccccccccccccccc}
\rightarrow & \# \\
\# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 \\
w_1^2 & \# & w_2^2 & \# & w_3^2 & \# & w_0^2 & \# & w_1^2 & \# & w_0^2 & \# & w_1^2 & \# & w_0^2 & \# & w_1^2 \\
\# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 \\
\# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 \\
\# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 \\
\# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 & \# & w_0^1 & \# & w_1^1 \\
\end{array}
\]

4. Boone-Higman-Thompson theorem

4. Boone-Higman-Thompson theorem

4. Boone-Higman-Thompson theorem

4. Boone-Higman-Thompson theorem
Proof: minimal effective construction

Let’s make a computable configuration:

- Every row has a level \( l \) and contains all pairs of words of \( \mathcal{L}(X) \) of length \( p_l \) separated by \# periodically.
- A row of level \( l \) appears every \( 2^l \) lines and contains words of length \( p_l \) the period of the previous level.
- The row of level 0 appears at most once and contains no \#.

The subshift thus generated is minimal and effective.
1. Analogies

2. Higman embedding theorem

3. Relative Higman embedding theorem

4. Boone-Higman-Thompson theorem

5. Conclusion
Conclusion

- Dictionary between subshifts/groups: not perfect, room for improvement.
- Intuition for proving theorems on subshifts
- What about theorems on groups?