Realizing Finitely Presented Groups as Fundamental Groups of SFTs

Léo Paviet Salomon, Pascal Vanier

GREYC
Université Caen-Normandie

August 29, 2023
MFCS 2023

Tilings

Alphabet: finite set of colours

$\Sigma = \{$ ,  $\}$

Tilings

Alphabet: finite set of colours

$\Sigma = \{$ ,  $\}$

A set $\mathcal{F}$ of forbidden, finite patterns:

Tilings

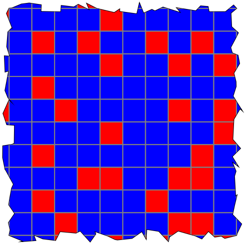
Alphabet: finite set of colours

$\Sigma = \{$ ,  $\}$

A set $\mathcal{F}$ of forbidden, finite patterns:

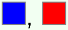
$$\mathcal{F} = \emptyset$$

Configuration on the alphabet Σ $\iff$ Σ-colouring of $\mathbb{Z}^2$ that does not contain any forbidden pattern



Tilings

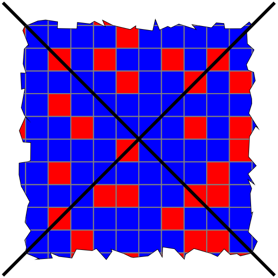
Alphabet: finite set of colours

$\Sigma = \{$  $\}$

A set $\mathcal{F}$ of forbidden, finite patterns:

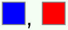
$\mathcal{F} = \{$  $\}$

Configuration on the alphabet Σ $\iff$ Σ-colouring of $\mathbb{Z}^2$ that does not contain any forbidden pattern



Tilings

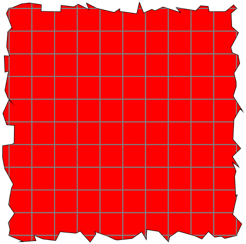
Alphabet: finite set of colours

$\Sigma = \{$  $\}$

A set $\mathcal{F}$ of forbidden, finite patterns:

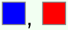
$\mathcal{F} = \{$  $\}$

Configuration on the alphabet Σ $\iff$ Σ-colouring of $\mathbb{Z}^2$ that does not contain any forbidden pattern



Tilings

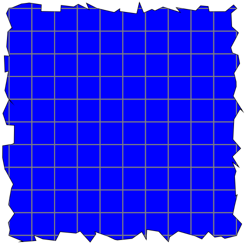
Alphabet: finite set of colours

$\Sigma = \{$  $\}$

A set $\mathcal{F}$ of forbidden, finite patterns:

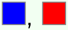
$\mathcal{F} = \{$  $\}$

Configuration on the alphabet Σ $\iff$ Σ-colouring of $\mathbb{Z}^2$ that does not contain any forbidden pattern



Tilings

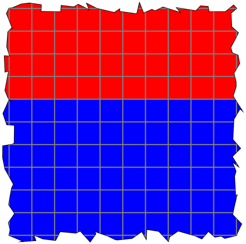
Alphabet: finite set of colours

$\Sigma = \{$  $\}$

A set $\mathcal{F}$ of forbidden, finite patterns:

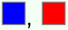
$\mathcal{F} = \{$  $\}$

Configuration on the alphabet Σ $\iff$ Σ-colouring of $\mathbb{Z}^2$ that does not contain any forbidden pattern



Tilings

Alphabet: finite set of colours

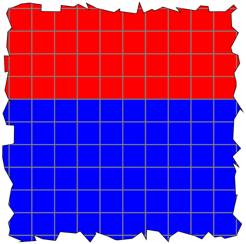
$\Sigma = \{$  $\}$

A set $\mathcal{F}$ of forbidden, finite patterns:

$\mathcal{F} = \{$  $\}$

The set of all the valid configurations is called a **subshift**, denoted by $X_{\mathcal{F}}$

Configuration on the alphabet Σ $\iff$ Σ-colouring of $\mathbb{Z}^2$ that does not contain any forbidden pattern



SFT

If the family of forbidden patterns $\mathcal{F}$ is finite, then $X_{\mathcal{F}}$ is a **subshift of finite type** (SFT).

SFT

If the family of forbidden patterns $\mathcal{F}$ is finite, then $X_{\mathcal{F}}$ is a **subshift of finite type** (SFT).
Despite being described by finite information, they are quite complicated objects; given a SFT X, all those problems are undecidable:

SFT

If the family of forbidden patterns $\mathcal{F}$ is finite, then $X_{\mathcal{F}}$ is a **subshift of finite type** (SFT).
Despite being described by finite information, they are quite complicated objects; given a SFT X, all those problems are undecidable:

- Is X empty ?

SFT

If the family of forbidden patterns $\mathcal{F}$ is finite, then $X_{\mathcal{F}}$ is a **subshift of finite type** (SFT).
Despite being described by finite information, they are quite complicated objects; given a SFT X, all those problems are undecidable:

- Is X empty ?
- Does X contain an aperiodic configuration ?

SFT

If the family of forbidden patterns $\mathcal{F}$ is finite, then $X_{\mathcal{F}}$ is a **subshift of finite type** (SFT).
Despite being described by finite information, they are quite complicated objects; given a SFT X, all those problems are undecidable:

- Is X empty ?
- Does X contain an aperiodic configuration ?
- How many patterns of size $n \times n$ are there in X ?

Projective Fundamental Group

We study a conjugacy (= isomorphism for subshifts) invariant: the **projective fundamental group**, defined by W.Geller and J.Propp in 1995 ([GP95]).

Projective Fundamental Group

We study a conjugacy (= isomorphism for subshifts) invariant: the **projective fundamental group**, defined by W.Geller and J.Propp in 1995 ([GP95]).
Idea: mimic the definition of the fundamental group in topology.

Paths

Let X a subshift. Fix a finite $B \subset \mathbb{Z}^2$. We define two things: **paths**, and what it means for paths to be **homotopic** (or deformation-equivalent)
A path is:

Paths

Let X a subshift. Fix a finite $B \subset \mathbb{Z}^2$. We define two things: **paths**, and what it means for paths to be **homotopic** (or deformation-equivalent)

A path is:

- a sequence $(\mathrm{v}_i, P_i)_{0 \leq i < t}$ of centered patterns, where

Paths

Let X a subshift. Fix a finite $B \subset \mathbb{Z}^2$. We define two things: **paths**, and what it means for paths to be **homotopic** (or deformation-equivalent)
A path is:

- a sequence $(\mathrm{v}_i, P_i)_{0 \leq i < t}$ of centered patterns, where
- (v_i), the **trajectory**, is a walk in $\mathbb{Z}^2$ (=a sequence of adjacent points)

Paths

Let X a subshift. Fix a finite $B \subset \mathbb{Z}^2$. We define two things: **paths**, and what it means for paths to be **homotopic** (or deformation-equivalent)
A path is:

- a sequence $(\mathrm{v}_i, P_i)_{0 \leq i < t}$ of centered patterns, where
- (v_i), the **trajectory**, is a walk in $\mathbb{Z}^2$ (=a sequence of adjacent points)
- (P_i) is a sequence of patterns of X of support B centered in v_i

Paths

Let X a subshift. Fix a finite $B \subset \mathbb{Z}^2$. We define two things: **paths**, and what it means for paths to be **homotopic** (or deformation-equivalent)
A path is:

- a sequence $(\mathrm{v}_i, P_i)_{0 \leq i < t}$ of centered patterns, where
- (v_i), the **trajectory**, is a walk in $\mathbb{Z}^2$ (=a sequence of adjacent points)
- (P_i) is a sequence of patterns of X of support B centered in v_i

Local consistency:

Paths

Let X a subshift. Fix a finite $B \subset \mathbb{Z}^2$. We define two things: **paths**, and what it means for paths to be **homotopic** (or deformation-equivalent)

A path is:

- a sequence $(\mathrm{v}_i, P_i)_{0 \leq i < t}$ of centered patterns, where
- (v_i), the **trajectory**, is a walk in $\mathbb{Z}^2$ (=a sequence of adjacent points)
- (P_i) is a sequence of patterns of X of support B centered in v_i

Local consistency:

Paths

Let X a subshift. Fix a finite $B \subset \mathbb{Z}^2$. We define two things: **paths**, and what it means for paths to be **homotopic** (or deformation-equivalent)
A path is:

- a sequence $(\mathrm{v}_i, P_i)_{0 \leq i < t}$ of centered patterns, where
- (v_i), the **trajectory**, is a walk in $\mathbb{Z}^2$ (=a sequence of adjacent points)
- (P_i) is a sequence of patterns of X of support B centered in v_i

Local consistency:

Paths

Let X a subshift. Fix a finite $B \subset \mathbb{Z}^2$. We define two things: **paths**, and what it means for paths to be **homotopic** (or deformation-equivalent)
A path is:

- a sequence $(\mathrm{v}_i, P_i)_{0 \leq i < t}$ of centered patterns, where
- (v_i), the **trajectory**, is a walk in $\mathbb{Z}^2$ (=a sequence of adjacent points)
- (P_i) is a sequence of patterns of X of support B centered in v_i

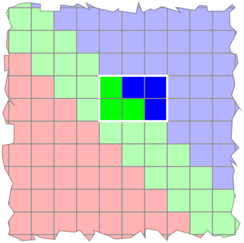
Local consistency:

Paths

Let X a subshift. Fix a finite $B \subset \mathbb{Z}^2$. We define two things: **paths**, and what it means for paths to be **homotopic** (or deformation-equivalent)
A path is:

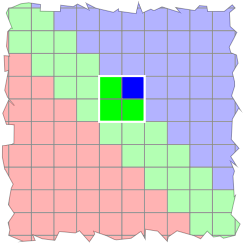
- a sequence $(\mathrm{v}_i, P_i)_{0 \leq i < t}$ of centered patterns, where
- (v_i), the **trajectory**, is a walk in $\mathbb{Z}^2$ (=a sequence of adjacent points)
- (P_i) is a sequence of patterns of X of support B centered in v_i

Local consistency:

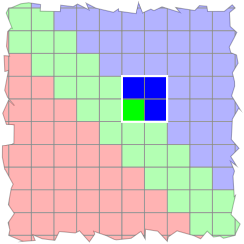


A path is not required to stay in the same configuration:

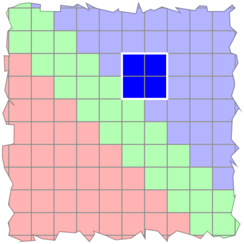
A path is not required to stay in the same configuration:



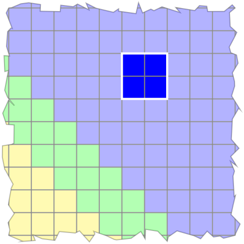
A path is not required to stay in the same configuration:



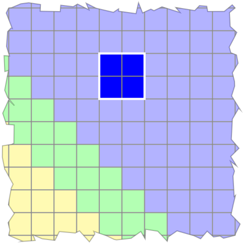
A path is not required to stay in the same configuration:



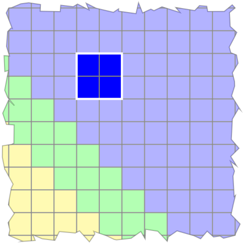
A path is not required to stay in the same configuration:



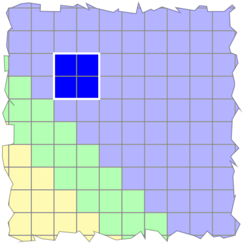
A path is not required to stay in the same configuration:



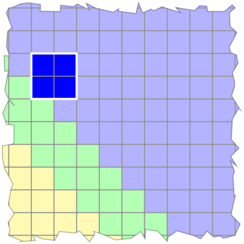
A path is not required to stay in the same configuration:



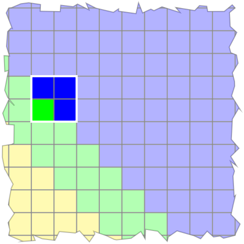
A path is not required to stay in the same configuration:



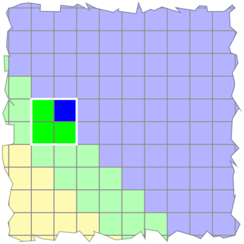
A path is not required to stay in the same configuration:



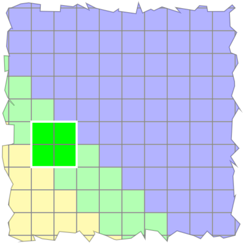
A path is not required to stay in the same configuration:



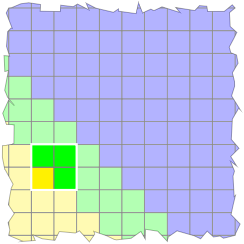
A path is not required to stay in the same configuration:



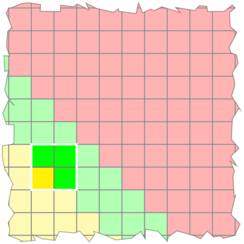
A path is not required to stay in the same configuration:



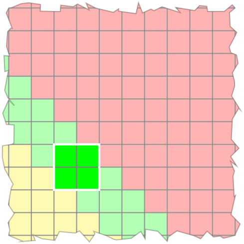
A path is not required to stay in the same configuration:



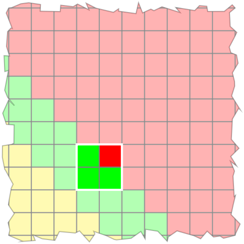
A path is not required to stay in the same configuration:



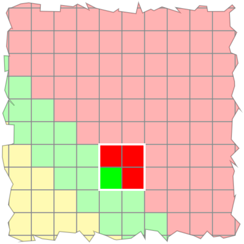
A path is not required to stay in the same configuration:



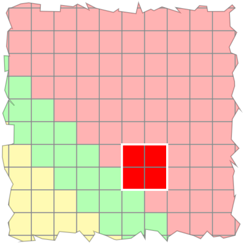
A path is not required to stay in the same configuration:



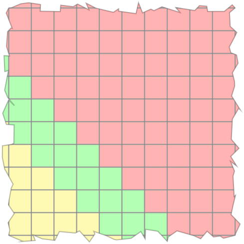
A path is not required to stay in the same configuration:



A path is not required to stay in the same configuration:



A path is not required to stay in the same configuration:



A path that stays within a single configuration is **coherent**.

Deformations

Can deform any *coherent (sub)path* by taking any other trajectory *inside a configuration* containing it.

Deformations

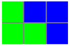
Can deform any *coherent (sub)path* by taking any other trajectory *inside a configuration* containing it.
This path ...

Deformations

Can deform any *coherent (sub)path* by taking any other trajectory *inside a configuration* containing it.
This path ...

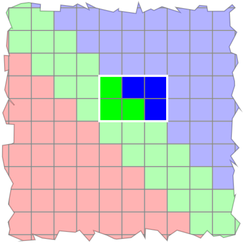
Deformations

Can deform any *coherent (sub)path* by taking any other trajectory *inside a configuration* containing it.
This path ...



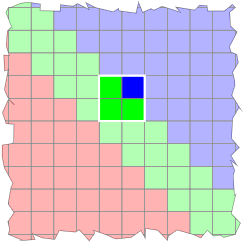
Deformations

Can deform any *coherent (sub)path* by taking any other trajectory *inside a configuration* containing it.
This path ...



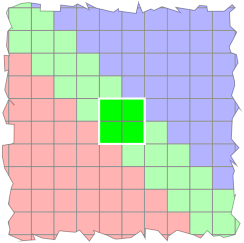
Deformations

Can deform any *coherent (sub)path* by taking any other trajectory *inside a configuration* containing it.
This path ... can be deformed into this one



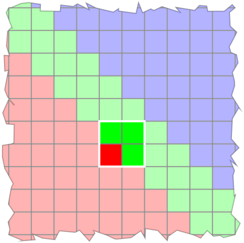
Deformations

Can deform any *coherent (sub)path* by taking any other trajectory *inside a configuration* containing it.
This path ... can be deformed into this one



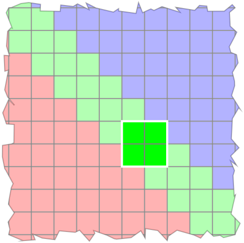
Deformations

Can deform any *coherent (sub)path* by taking any other trajectory *inside a configuration* containing it.
This path ... can be deformed into this one



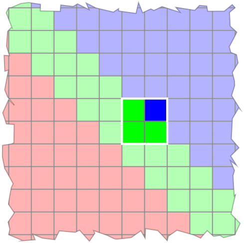
Deformations

Can deform any *coherent (sub)path* by taking any other trajectory *inside a configuration* containing it.
This path ... can be deformed into this one



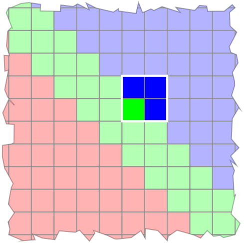
Deformations

Can deform any *coherent (sub)path* by taking any other trajectory *inside a configuration* containing it.
This path ... can be deformed into this one



Deformations

Can deform any *coherent (sub)path* by taking any other trajectory *inside a configuration* containing it.
This path ... can be deformed into this one



Homotopic paths: paths that be deformed into one another with a finite sequence of such elementary deformations.

Homotopic paths: paths that be deformed into one another with a finite sequence of such elementary deformations.
We can therefore define *for each window* $B \subset \mathbb{Z}^2$ a fundamental group $\pi_1^B(X)$:

Homotopic paths: paths that be deformed into one another with a finite sequence of such elementary deformations.
We can therefore define *for each window* $B \subset \mathbb{Z}^2$ a fundamental group $\pi_1^B(X)$:

- Group elements are (equivalence classes of) loops

Homotopic paths: paths that be deformed into one another with a finite sequence of such elementary deformations.
We can therefore define *for each window* $B \subset \mathbb{Z}^2$ a fundamental group $\pi_1^B(X)$:

- Group elements are (equivalence classes of) loops
- The group operation is the concatenation of loops

Homotopic paths: paths that be deformed into one another with a finite sequence of such elementary deformations.
We can therefore define *for each window* $B \subset \mathbb{Z}^2$ a fundamental group $\pi_1^B(X)$:

- Group elements are (equivalence classes of) loops
- The group operation is the concatenation of loops

(To get a single object, we need a way to combine all those groups $\pi_1^B(X)$ in a single group $\pi_1(X)$: we take the projective limit of those groups – not important here)

A natural question is to determine if we can realize arbitrary groups as fundamental groups of some subshifts.

A natural question is to determine if we can realize arbitrary groups as fundamental groups of some subshifts.
A first answer in the original article: we can at least make all the **finite groups**.

A natural question is to determine if we can realize arbitrary groups as fundamental groups of some subshifts.
A first answer in the original article: we can at least make all the **finite groups**.
We show that we can in fact realize (at least) all the **finitely presented** groups.

Group presentation: example

Point of view about group presentations: **rewriting theory**

Group presentation: example

Point of view about group presentations: **rewriting theory**

- Group elements are words on an alphabet (generators)

Group presentation: example

Point of view about group presentations: **rewriting theory**

- Group elements are words on an alphabet (generators)
- Two words are the same if we can rewrite one into the other by inserting/deleting relators

Group presentation: example

Point of view about group presentations: **rewriting theory**

- Group elements are words on an alphabet (generators)
- Two words are the same if we can rewrite one into the other by inserting/deleting relators

$$\mathbb{Z}_2 \times \mathbb{Z}_3 = \langle a, b, a', b' \mid \underbrace{a^2}_{r_1}, \underbrace{b^3}_{r_2}, \underbrace{aba'b'}_{r_3}, aa', a'a, bb', b'b \rangle :$$

Group presentation: example

Point of view about group presentations: **rewriting theory**

- Group elements are words on an alphabet (generators)
- Two words are the same if we can rewrite one into the other by inserting/deleting relators

$$\mathbb{Z}_2 \times \mathbb{Z}_3 = \langle a, b, a', b' \mid \underbrace{a^2}_{r_1}, \underbrace{b^3}_{r_2}, \underbrace{aba'b'}_{r_3}, aa', a'a, bb', b'b \rangle:$$

$$\begin{aligned} abbab &= abb\color{red}{bb'}\color{black}ab \\ &\underset{r_2}{=} ab'ab \\ &\underset{r_3}{=} ab'a\color{red}{aba'b'}\color{black}b \\ &\underset{r_1,r_2}{=} ab'ba' \\ &= aa' \\ &= 1 \end{aligned}$$

Main theorem

Theorem (P., Vanier [PSV23])

Every finitely presented group is the projective fundamental group of some SFT.

Main theorem

Theorem (P., Vanier [PSV23])

Every finitely presented group is the projective fundamental group of some SFT.

The idea of the construction is to see paths in the subshift as words on the alphabet of generators, and their deformation as relations. More precisely:

- Some tiles will correspond to generators
- Paths will correspond to words on those generators
- Local rules will ensure that deformations correspond to insertion/deletion of relators

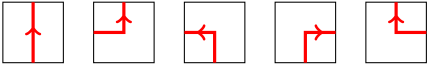
A set of tiles

Let $G = \langle S \mid R \rangle$ a finitely presented group. Consider the tiles:

A set of tiles

Let $G = \langle S \mid R \rangle$ a finitely presented group. Consider the tiles:

For each generator $s \in S$:

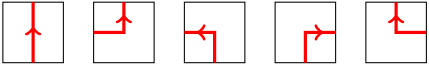


and a single empty tile

A set of tiles

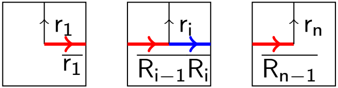
Let $G = \langle S \mid R \rangle$ a finitely presented group. Consider the tiles:

For each generator $s \in S$:



and a single empty tile

For each $r_1 \dots r_n \in R$:

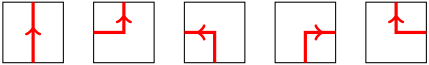


$\overline{R_i}$ = fresh colour corresponding to the prefix $r_1 \dots r_i$:

A set of tiles

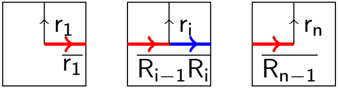
Let $G = \langle S \mid R \rangle$ a finitely presented group. Consider the tiles:

For each generator $s \in S$:



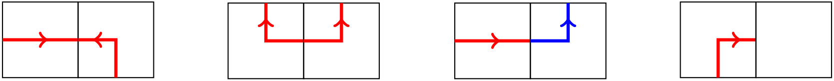
and a single empty tile

For each $r_1 \dots r_n \in R$:



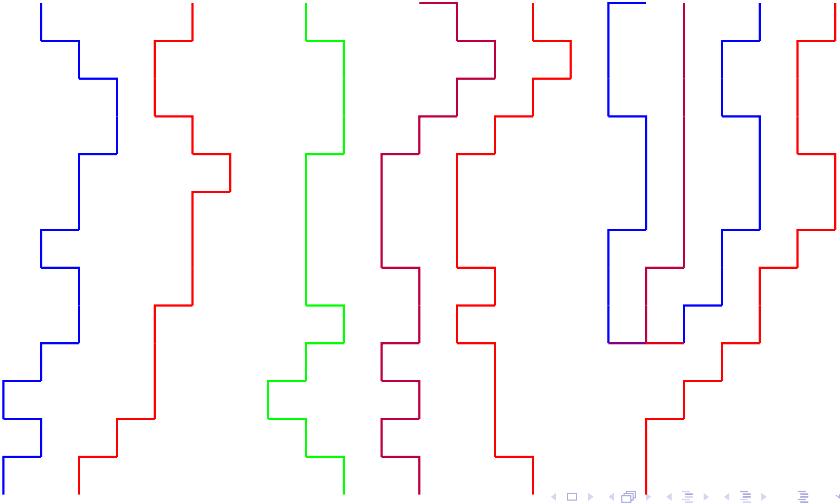
$\overline{R_i}$ = fresh colour corresponding to the prefix $r_1 \dots r_i$:

Matching rules: wires can't stop, *i.e.* all the following are **forbidden**



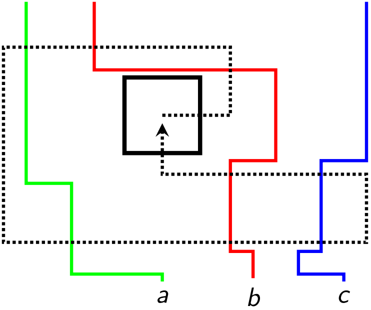
Configurations

Configurations look like this:



From paths to words

To a path p, we associate a unique word on the alphabet $S \cup S^{-1}$, denoted by $[\![p]\!]$, defined by the wires that it crosses.



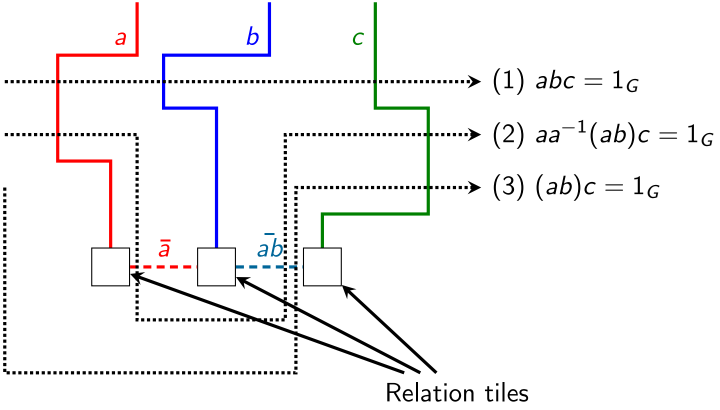
For this path p, $[\![p]\!] \equiv bb^{-1}a^{-1}abcc^{-1}b^{-1} =_G 1_G$.

Homotopic implies equal

Claim: in a given configuration and between 2 given endpoints, all the paths are associated to the same element of G

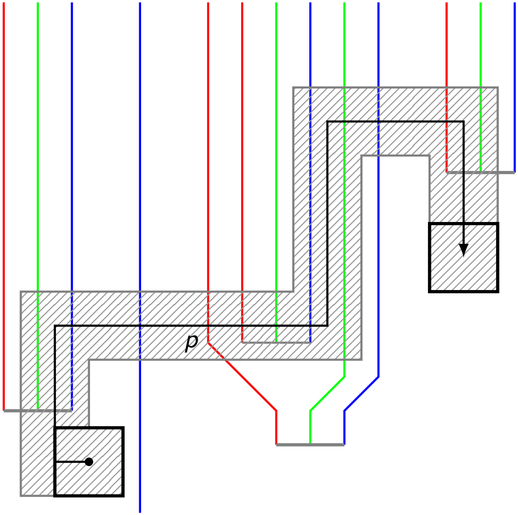
Homotopic implies equal

Claim: in a given configuration and between 2 given endpoints, all the paths are associated to the same element of G

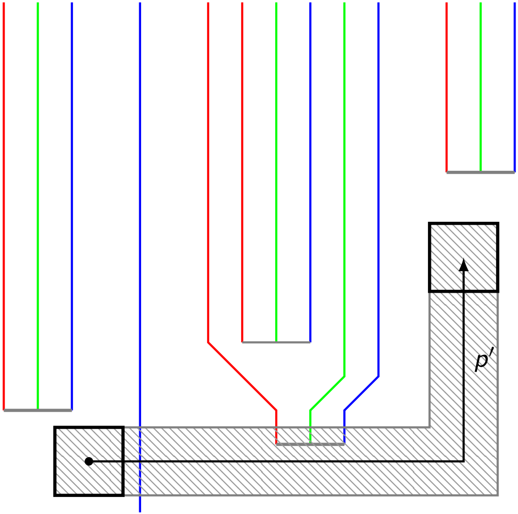


Relation $abc = 1_G$

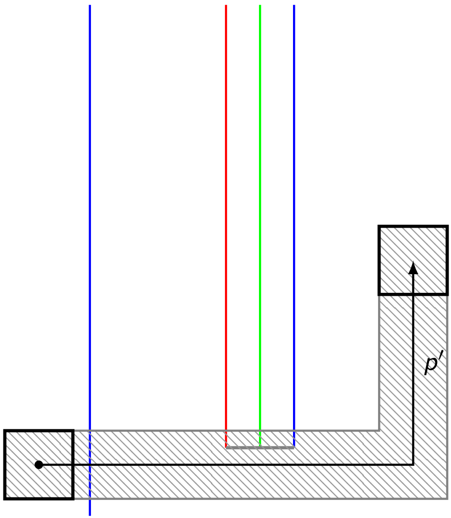
Avoid all the relation tiles



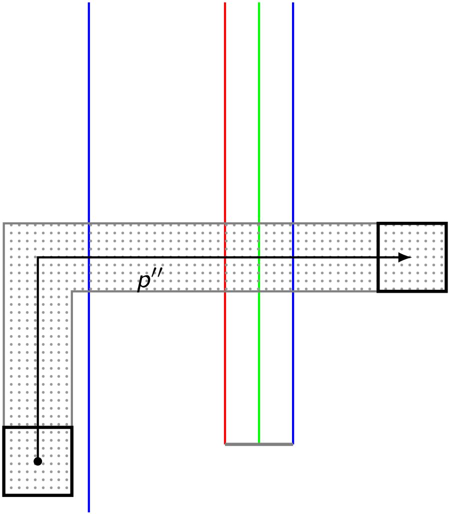
Avoid all the relation tiles



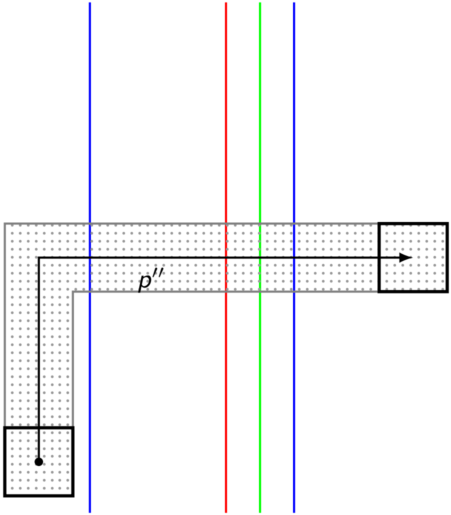
Avoid all the relation tiles



Avoid all the relation tiles

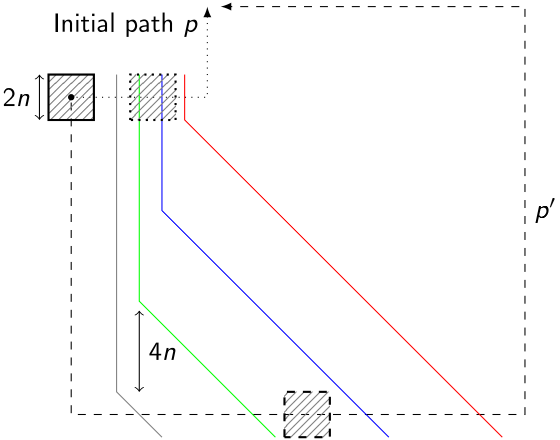


Avoid all the relation tiles



Only a single wire at a time

Deformation of p into p' in a single configuration to see only one wire per pattern.



Series of such tricks:

- All the equivalent loops are associated to the same group element
- All the loops associated to the same group element are equivalent

Therefore the fundamental group is $G = \langle S \mid R \rangle$

Bibliography

William Geller and James Propp.
The projective fundamental group of a $\mathbb{Z}^2$-shift.
Ergodic Theory and Dynamical Systems, 15(6):1091–1118, 1995.

Léo Paviet Salomon and Pascal Vanier.
Realizing Finitely Presented Groups as Projective Fundamental Groups of SFTs.
In Jérôme Leroux, Sylvain Lombardy, and David Peleg, editors, *48th International Symposium on Mathematical Foundations of Computer Science (MFCS 2023)*, volume 272 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 75:1–75:15, Dagstuhl, Germany, 2023. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

Normalizing coherent paths

The previous remarks mean that a coherent path can be put into some kind of normal form. However, all loops need not be coherent – even when the fundamental group is trivial !

Normalizing coherent paths

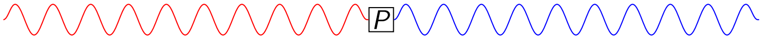
The previous remarks mean that a coherent path can be put into some kind of normal form. However, all loops need not be coherent – even when the fundamental group is trivial !
Solution: decompose a loop into a sequence of coherent paths, and add contractible loops at the junction.

Path decomposition

Whenever we have a incoherent path, we can split it in *normalized* coherent paths:

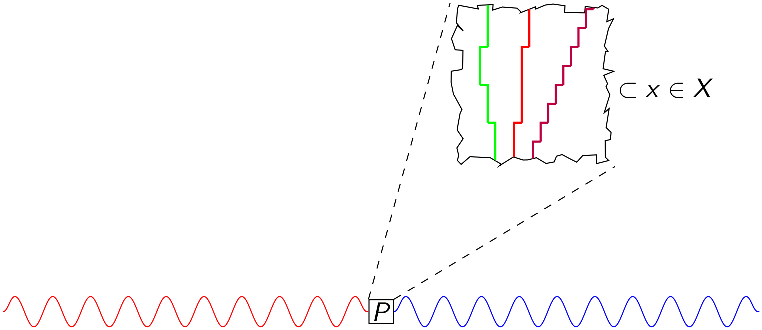
Path decomposition

Whenever we have a incoherent path, we can split it in *normalized* coherent paths:



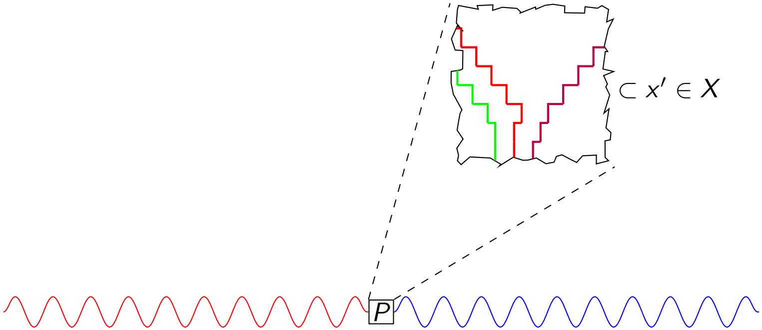
Path decomposition

Whenever we have a incoherent path, we can split it in *normalized* coherent paths:



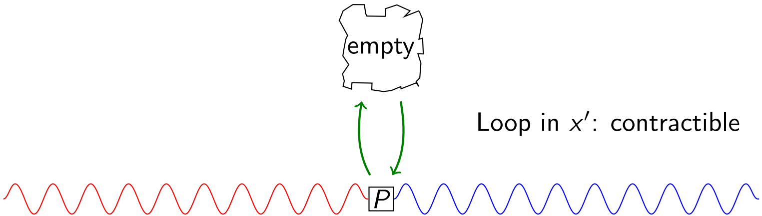
Path decomposition

Whenever we have a incoherent path, we can split it in *normalized* coherent paths:



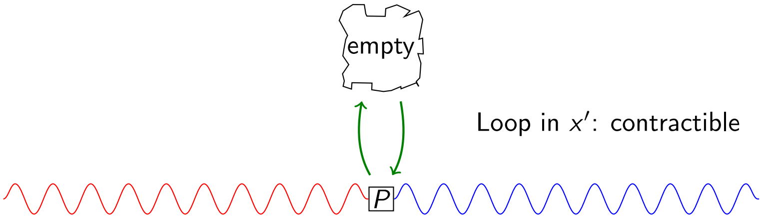
Path decomposition

Whenever we have a incoherent path, we can split it in *normalized* coherent paths:



Path decomposition

Whenever we have a incoherent path, we can split it in *normalized* coherent paths:



This means that we can deform the path so that each sub-coherent path starts and ends with empty patterns: can deform them independently.