Consistency of combinatorial substitutions

Timo Jolivet, en collaboration avec Jarkko Kari
LIAFA, Université Paris Diderot
FUNDIM, Université de Turku, Finlande

Séminaire Combinatoire et Algorithmes
LITIS, Université de Rouen
29 mars 2012

Substitutions: replace letters by patterns.

Substitutions: replace letters by patterns.

Examples. . .

$$1 \mapsto \begin{smallmatrix} 1 & 2 \\ 2 & 1 \end{smallmatrix} \qquad 2 \mapsto \begin{smallmatrix} 2 & 1 \\ 1 & 2 \end{smallmatrix}$$

1

$$1 \mapsto \begin{smallmatrix} 1 & 2 \\ 2 & 1 \end{smallmatrix} \qquad 2 \mapsto \begin{smallmatrix} 2 & 1 \\ 1 & 2 \end{smallmatrix}$$

$$1 \mapsto \begin{smallmatrix} 1 & 2 \\ 2 & 1 \end{smallmatrix}$$

$$1 \mapsto \begin{matrix} 1 & 2 \\ 2 & 1 \end{matrix} \qquad\qquad 2 \mapsto \begin{matrix} 2 & 1 \\ 1 & 2 \end{matrix}$$

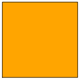
$$1 \mapsto \begin{matrix} 1 & 2 \\ 2 & 1 \end{matrix} \mapsto \begin{matrix} 1 & 2 & 2 & 1 \\ 2 & 1 & 1 & 2 \\ 2 & 1 & 1 & 2 \\ 1 & 2 & 2 & 1 \end{matrix}$$

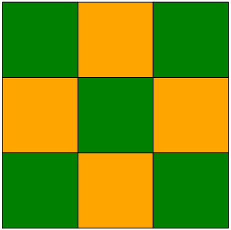
$$1 \mapsto \begin{matrix} 1 & 2 \\ 2 & 1 \end{matrix} \qquad\qquad 2 \mapsto \begin{matrix} 2 & 1 \\ 1 & 2 \end{matrix}$$

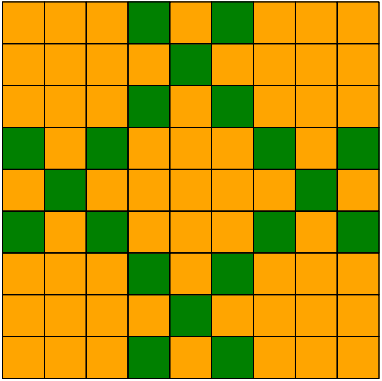
$$1 \mapsto \begin{matrix} 1 & 2 \\ 2 & 1 \end{matrix} \mapsto \begin{matrix} 1 & 2 & 2 & 1 \\ 2 & 1 & 1 & 2 \\ 2 & 1 & 1 & 2 \\ 1 & 2 & 2 & 1 \end{matrix} \mapsto \begin{matrix} 1 & 2 & 2 & 1 & 2 & 1 & 1 & 2 \\ 2 & 1 & 1 & 2 & 1 & 2 & 2 & 1 \\ 2 & 1 & 1 & 2 & 1 & 2 & 2 & 1 \\ 1 & 2 & 2 & 1 & 2 & 1 & 1 & 2 \\ 1 & 2 & 2 & 1 & 2 & 1 & 1 & 2 \\ 2 & 1 & 1 & 2 & 1 & 2 & 2 & 1 \\ 2 & 1 & 1 & 2 & 1 & 2 & 2 & 1 \\ 1 & 2 & 2 & 1 & 2 & 1 & 1 & 2 \end{matrix}$$

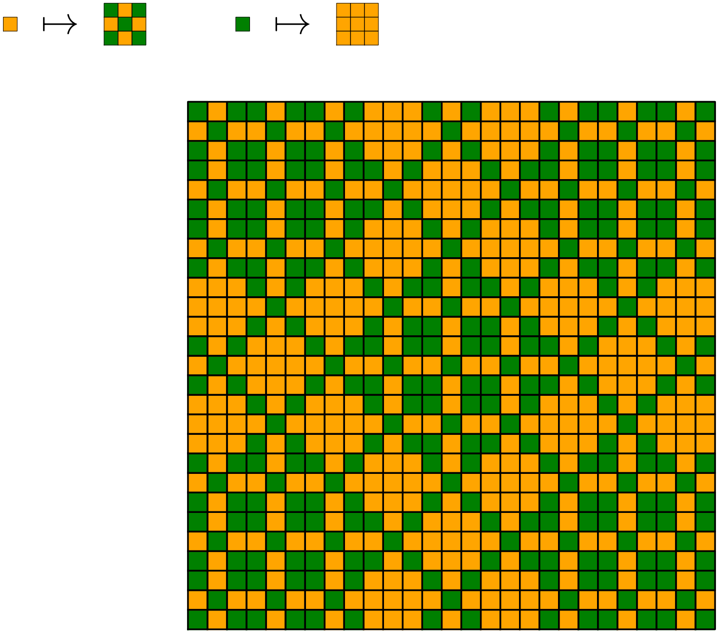
$$1 \mapsto \begin{matrix} 1 & 2 \\ 2 & 1 \end{matrix} \qquad\qquad 2 \mapsto \begin{matrix} 2 & 1 \\ 1 & 2 \end{matrix}$$

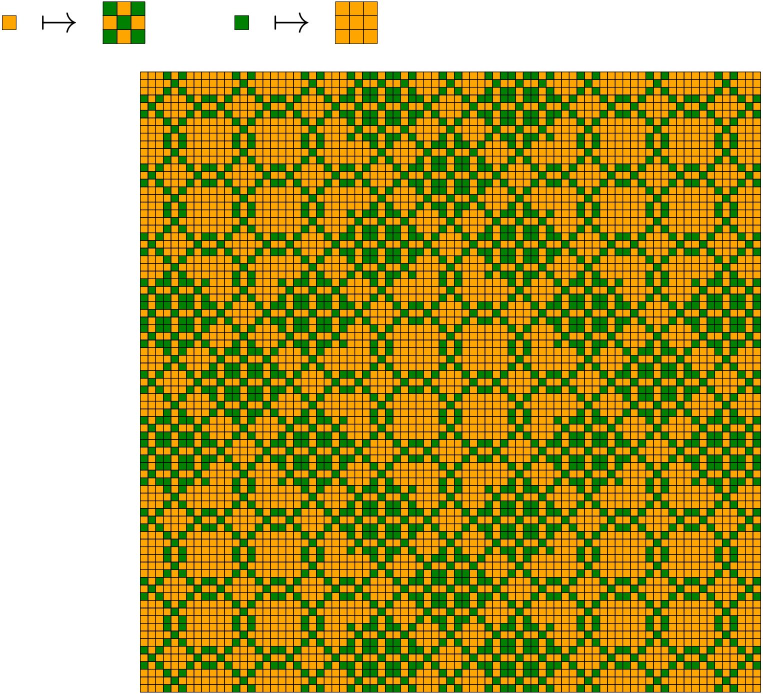
$$1 \mapsto \begin{matrix} 1 & 2 \\ 2 & 1 \end{matrix} \mapsto \begin{matrix} 1 & 2 & 2 & 1 \\ 2 & 1 & 1 & 2 \\ 2 & 1 & 1 & 2 \\ 1 & 2 & 2 & 1 \end{matrix} \mapsto \begin{matrix} 1 & 2 & 2 & 1 & 2 & 1 & 1 & 2 \\ 2 & 1 & 1 & 2 & 1 & 2 & 2 & 1 \\ 2 & 1 & 1 & 2 & 1 & 2 & 2 & 1 \\ 1 & 2 & 2 & 1 & 2 & 1 & 1 & 2 \\ 1 & 2 & 2 & 1 & 2 & 1 & 1 & 2 \\ 2 & 1 & 1 & 2 & 1 & 2 & 2 & 1 \\ 2 & 1 & 1 & 2 & 1 & 2 & 2 & 1 \\ 1 & 2 & 2 & 1 & 2 & 1 & 1 & 2 \end{matrix} \mapsto \begin{matrix} 1 & 2 & 2 & 1 & 2 & 1 & 1 & 2 & 2 & 1 & 1 & 2 & 1 & 2 & 2 & 1 \\ 2 & 1 & 1 & 2 & 1 & 2 & 2 & 1 & 1 & 2 & 2 & 1 & 2 & 1 & 1 & 2 \\ 2 & 1 & 1 & 2 & 1 & 2 & 2 & 1 & 1 & 2 & 2 & 1 & 2 & 1 & 1 & 2 \\ 1 & 2 & 2 & 1 & 2 & 1 & 1 & 2 & 2 & 1 & 1 & 2 & 1 & 2 & 2 & 1 \\ 2 & 1 & 1 & 2 & 1 & 2 & 2 & 1 & 1 & 2 & 2 & 1 & 2 & 1 & 1 & 2 \\ 1 & 2 & 2 & 1 & 2 & 1 & 1 & 2 & 2 & 1 & 1 & 2 & 1 & 2 & 2 & 1 \\ 1 & 2 & 2 & 1 & 2 & 1 & 1 & 2 & 2 & 1 & 1 & 2 & 1 & 2 & 2 & 1 \\ 2 & 1 & 1 & 2 & 1 & 2 & 2 & 1 & 1 & 2 & 2 & 1 & 2 & 1 & 1 & 2 \\ 2 & 1 & 1 & 2 & 1 & 2 & 2 & 1 & 1 & 2 & 2 & 1 & 2 & 1 & 1 & 2 \\ 1 & 2 & 2 & 1 & 2 & 1 & 1 & 2 & 2 & 1 & 1 & 2 & 1 & 2 & 2 & 1 \\ 1 & 2 & 2 & 1 & 2 & 1 & 1 & 2 & 2 & 1 & 1 & 2 & 1 & 2 & 2 & 1 \\ 2 & 1 & 1 & 2 & 1 & 2 & 2 & 1 & 1 & 2 & 2 & 1 & 2 & 1 & 1 & 2 \\ 1 & 2 & 2 & 1 & 2 & 1 & 1 & 2 & 2 & 1 & 1 & 2 & 1 & 2 & 2 & 1 \\ 2 & 1 & 1 & 2 & 1 & 2 & 2 & 1 & 1 & 2 & 2 & 1 & 2 & 1 & 1 & 2 \\ 2 & 1 & 1 & 2 & 1 & 2 & 2 & 1 & 1 & 2 & 2 & 1 & 2 & 1 & 1 & 2 \\ 1 & 2 & 2 & 1 & 2 & 1 & 1 & 2 & 2 & 1 & 1 & 2 & 1 & 2 & 2 & 1 \end{matrix}$$











$$\boxed{1} \;\longmapsto\; \begin{array}{|cc|}\hline 2 & 4 \\ 1 & 3 \\ \hline\end{array} \qquad \boxed{2} \;\longmapsto\; \boxed{1\;\;3} \qquad \boxed{3} \;\longmapsto\; \begin{array}{|c|}\hline 2 \\ 1 \\ \hline\end{array} \qquad \boxed{4} \;\longmapsto\; \boxed{1}$$

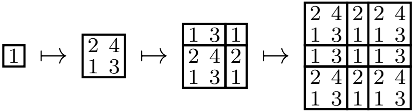
$$\boxed{1}$$

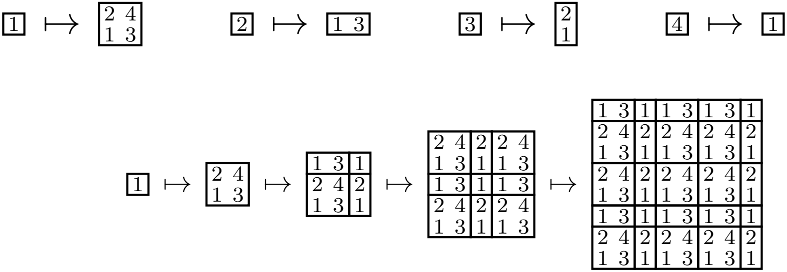
$$\boxed{1} \longmapsto \boxed{\begin{matrix} 2 & 4 \\ 1 & 3 \end{matrix}} \qquad \boxed{2} \longmapsto \boxed{1 \ \ 3} \qquad \boxed{3} \longmapsto \boxed{\begin{matrix} 2 \\ 1 \end{matrix}} \qquad \boxed{4} \longmapsto \boxed{1}$$

$$\boxed{1} \mapsto \boxed{\begin{matrix} 2 & 4 \\ 1 & 3 \end{matrix}}$$

$$\boxed{1} \longmapsto \boxed{\begin{array}{cc} 2 & 4 \\ 1 & 3 \end{array}} \qquad \boxed{2} \longmapsto \boxed{\begin{array}{cc} 1 & 3 \end{array}} \qquad \boxed{3} \longmapsto \boxed{\begin{array}{c} 2 \\ 1 \end{array}} \qquad \boxed{4} \longmapsto \boxed{1}$$

$$\boxed{1} \mapsto \boxed{\begin{array}{cc} 2 & 4 \\ 1 & 3 \end{array}} \mapsto \begin{array}{|cc|c|} \hline 1 & 3 & 1 \\ \hline 2 & 4 & 2 \\ 1 & 3 & 1 \\ \hline \end{array}$$





"Compatibles" rectangles.

11 letters, compatible rectangles (from M. Rigo)

$$a \mapsto \begin{array}{|c|c|}\hline a & b \\ \hline c & d \\ \hline\end{array} \quad b \mapsto \begin{array}{|c|}\hline i \\ \hline e \\ \hline\end{array} \quad c \mapsto \begin{array}{|c|c|}\hline i & j \\ \hline\end{array} \quad d \mapsto \begin{array}{|c|}\hline i \\ \hline\end{array} \quad e \mapsto \begin{array}{|c|c|}\hline f & b \\ \hline\end{array}$$

$$f \mapsto \begin{array}{|c|c|}\hline g & b \\ \hline h & d \\ \hline\end{array} \quad g \mapsto \begin{array}{|c|c|}\hline f & b \\ \hline h & d \\ \hline\end{array} \quad h \mapsto \begin{array}{|c|c|}\hline i & m \\ \hline\end{array} \quad i \mapsto \begin{array}{|c|c|}\hline i & m \\ \hline h & d \\ \hline\end{array}$$

$$j \mapsto \begin{array}{|c|}\hline k \\ \hline c \\ \hline\end{array} \quad k \mapsto \begin{array}{|c|c|}\hline l & m \\ \hline c & d \\ \hline\end{array} \quad l \mapsto \begin{array}{|c|c|}\hline k & m \\ \hline c & d \\ \hline\end{array} \quad m \mapsto \begin{array}{|c|}\hline i \\ \hline h \\ \hline\end{array}$$

26 letters, compatible rectangles (from M. Rigo)

$$a \mapsto \begin{array}{|c|c|}\hline a & b \\ \hline c & d \\ \hline\end{array} \quad b \mapsto \begin{array}{|c|}\hline e \\ \hline f \\ \hline\end{array} \quad c \mapsto \begin{array}{|c|c|}\hline e & h \\ \hline\end{array} \quad d \mapsto \begin{array}{|c|}\hline i \\ \hline\end{array} \quad e \mapsto \begin{array}{|c|c|}\hline j & k \\ \hline l & m \\ \hline\end{array}$$

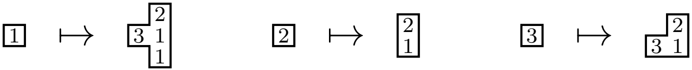
$$f \mapsto \begin{array}{|c|c|}\hline g & b \\ \hline\end{array} \quad g \mapsto \begin{array}{|c|c|}\hline y & b \\ \hline o & t \\ \hline\end{array} \quad h \mapsto \begin{array}{|c|}\hline z \\ \hline c \\ \hline\end{array} \quad i \mapsto \begin{array}{|c|c|}\hline i & n \\ \hline o & d \\ \hline\end{array}$$

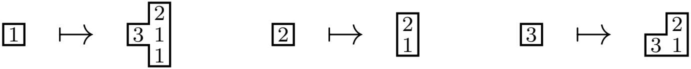
$$j \mapsto \begin{array}{|c|c|}\hline e & p \\ \hline q & r \\ \hline\end{array} \quad k \mapsto \begin{array}{|c|}\hline e \\ \hline s \\ \hline\end{array} \quad l \mapsto \begin{array}{|c|c|}\hline e & u \\ \hline\end{array} \quad m \mapsto \begin{array}{|c|}\hline e \\ \hline\end{array}$$

$$n \mapsto \begin{array}{|c|}\hline i \\ \hline o \\ \hline\end{array} \quad o \mapsto \begin{array}{|c|c|}\hline i & n \\ \hline\end{array} \quad p \mapsto \begin{array}{|c|}\hline e \\ \hline q \\ \hline\end{array} \quad q \mapsto \begin{array}{|c|c|}\hline e & p \\ \hline\end{array} \quad r \mapsto \begin{array}{|c|}\hline e \\ \hline\end{array}$$

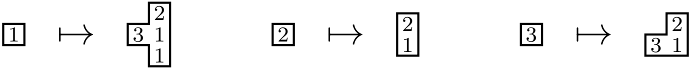
$$s \mapsto \begin{array}{|c|c|}\hline v & k \\ \hline\end{array} \quad t \mapsto \begin{array}{|c|}\hline i \\ \hline\end{array} \quad u \mapsto \begin{array}{|c|}\hline w \\ \hline l \\ \hline\end{array} \quad v \mapsto \begin{array}{|c|c|}\hline w & p \\ \hline l & r \\ \hline\end{array} \quad w \mapsto \begin{array}{|c|c|}\hline v & k \\ \hline q & r \\ \hline\end{array}$$

$$x \mapsto \begin{array}{|c|c|}\hline z & n \\ \hline c & d \\ \hline\end{array} \quad y \mapsto \begin{array}{|c|c|}\hline g & b \\ \hline o & d \\ \hline\end{array} \quad z \mapsto \begin{array}{|c|c|}\hline x & n \\ \hline c & t \\ \hline\end{array}$$



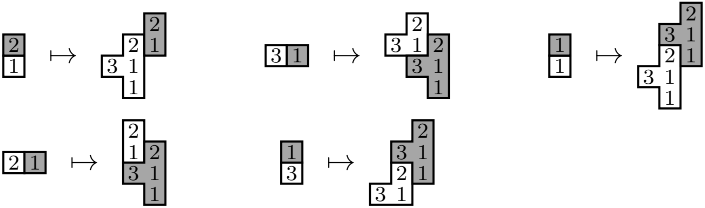


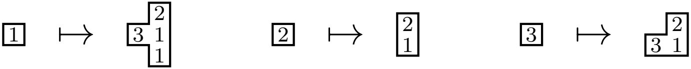
How can we “concatenate” these ?



How can we “concatenate” these ?

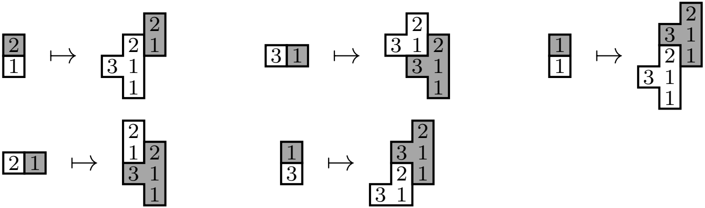
➥ We must define some **rules:**

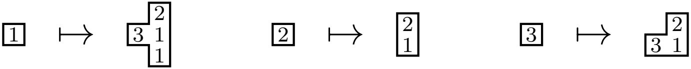




How can we “concatenate” these ?

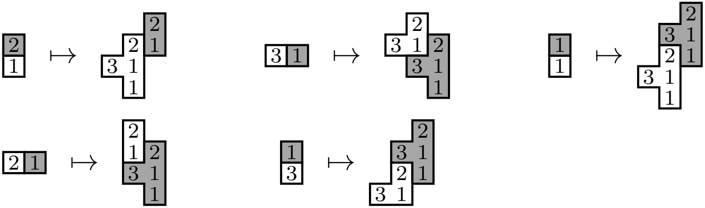
➥ We must define some **rules:**

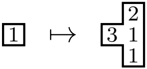


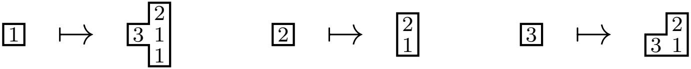


How can we “concatenate” these ?

➥ We must define some **rules:**

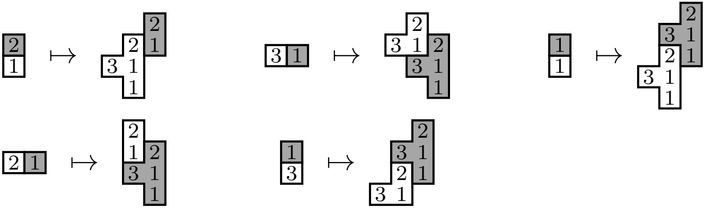


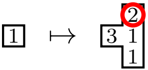


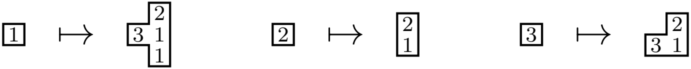


How can we “concatenate” these ?

➥ We must define some **rules:**

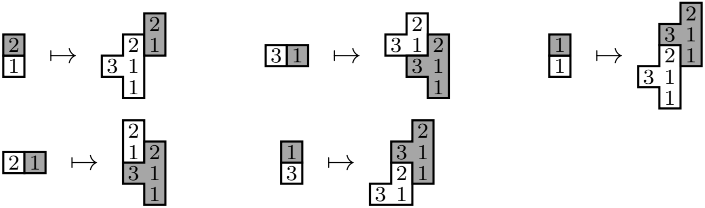


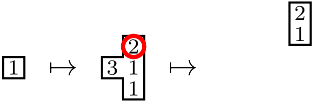


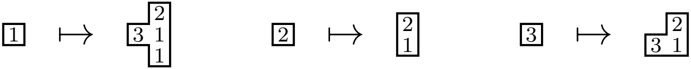


How can we “concatenate” these ?

➥ We must define some **rules:**

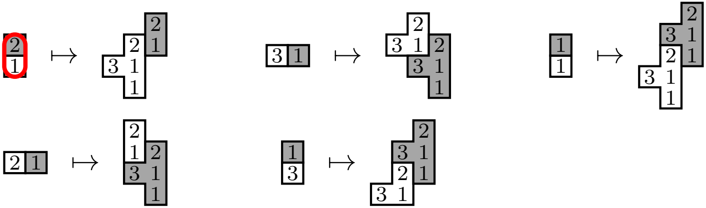


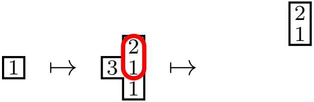


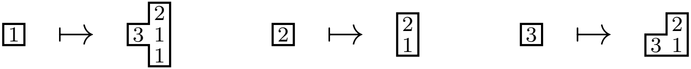


How can we “concatenate” these ?

➥ We must define some **rules:**

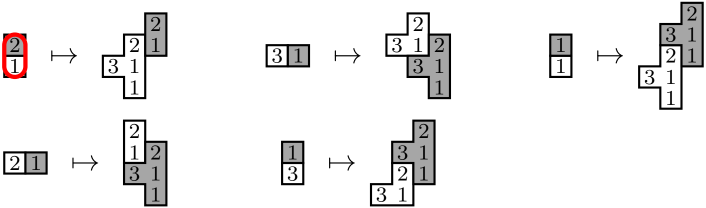


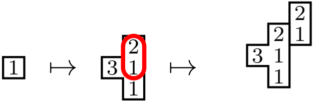


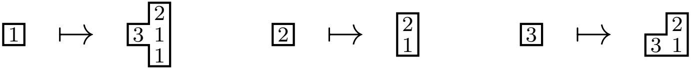


How can we “concatenate” these ?

➥ We must define some **rules:**

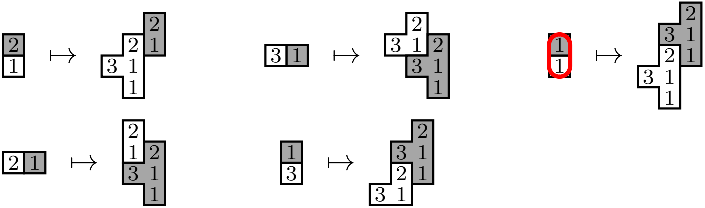


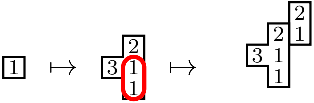


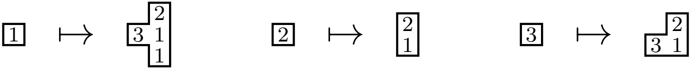


How can we “concatenate” these ?

➥ We must define some **rules:**

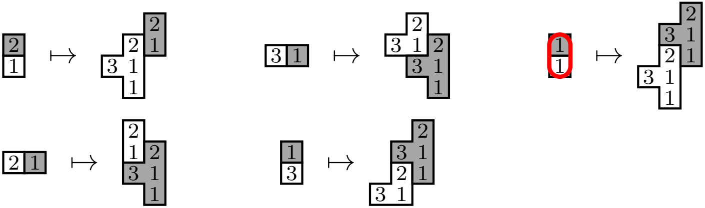


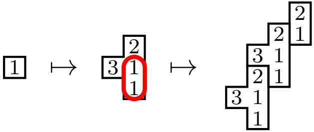


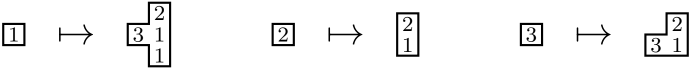


How can we “concatenate” these ?

➥ We must define some **rules:**

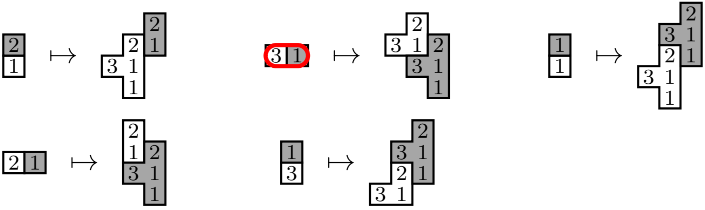


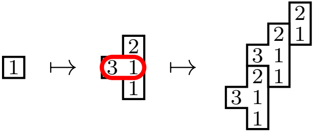


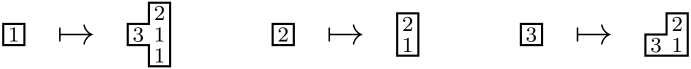


How can we “concatenate” these ?

➥ We must define some **rules:**

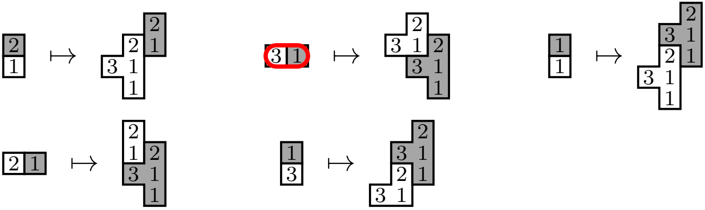


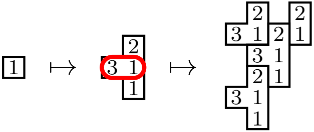


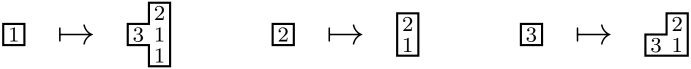


How can we “concatenate” these ?

➥ We must define some **rules:**

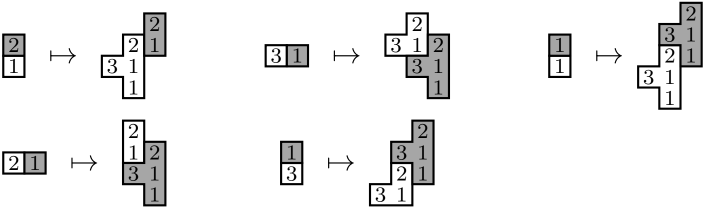


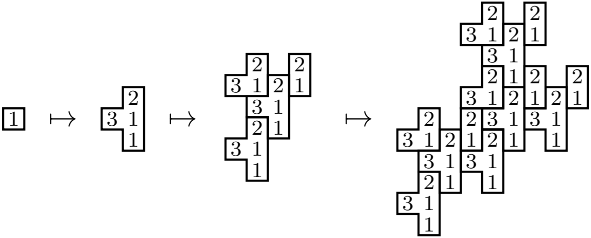


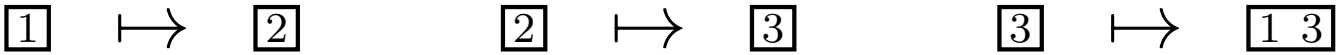


How can we “concatenate” these ?

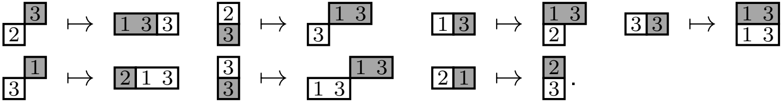
➥ We must define some **rules:**

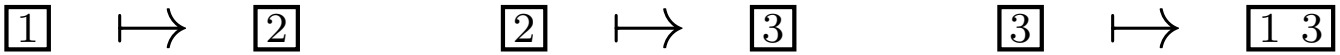




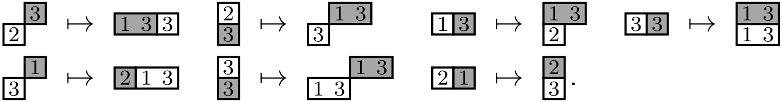


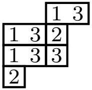
Concatenation rules:

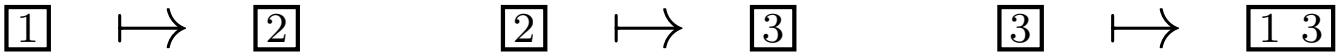




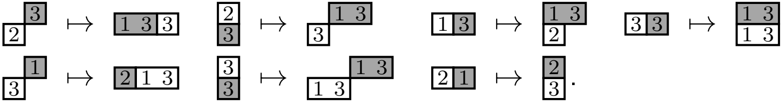
Concatenation rules:

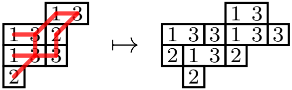


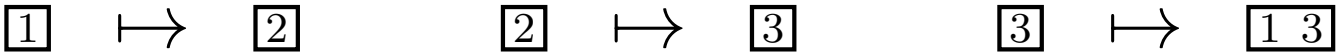




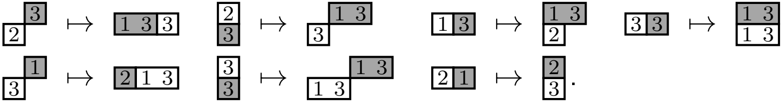
Concatenation rules:

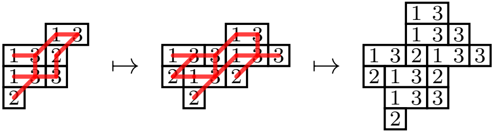






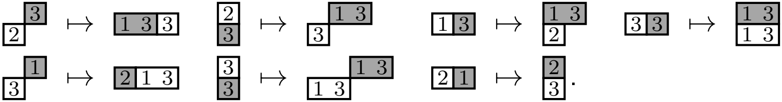
Concatenation rules:

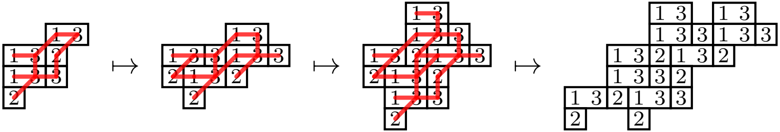




1 ↦ 2   2 ↦ 3   3 ↦ 1 3

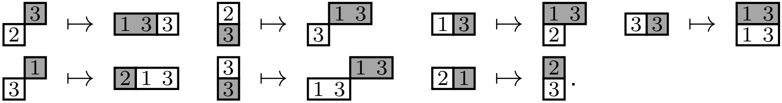
Concatenation rules:

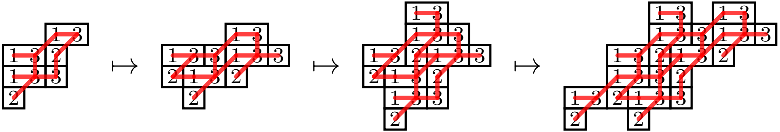




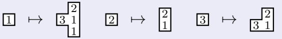
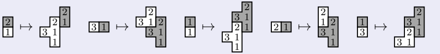
$\boxed{1} \mapsto \boxed{2}$ $\quad$ $\boxed{2} \mapsto \boxed{3}$ $\quad$ $\boxed{3} \mapsto \boxed{1\ 3}$

Concatenation rules:

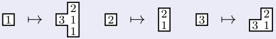
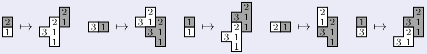




Definition: a **combinatorial substitution** is given by:

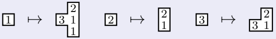
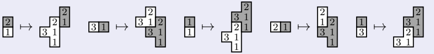
1. base rules (),
2. concatenation rules ().

Definition: a **combinatorial substitution** is given by:

1. base rules (),
2. concatenation rules ().

- Nice and very general definition. . .

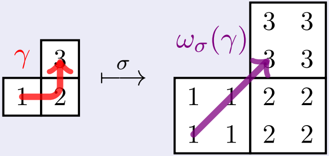
Definition: a **combinatorial substitution** is given by:

1. base rules (),
2. concatenation rules ().

- Nice and very general definition. . .
- **But:**
  - Such substitutions are not necessarily consistent.
  - Such substitutions can produce overlapping patterns.

Consistency

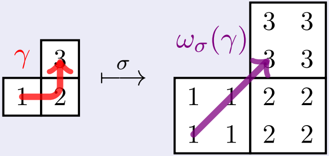
Notation: if γ is a σ-path, let $\omega_\sigma(\gamma)$ be the **image vector** of γ.



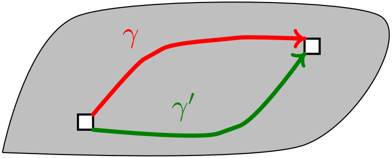
Meaning:
$\sigma(1)$ and $\sigma(3)$
separated by $\omega_\sigma(\gamma)$

Consistency

Notation: if γ is a σ-path, let $\omega_\sigma(\gamma)$ be the **image vector** of γ.

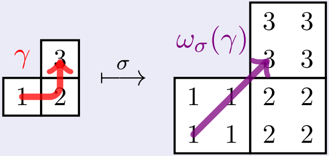


Meaning:
$\sigma(1)$ and $\sigma(3)$
separated by $\omega_\sigma(\gamma)$

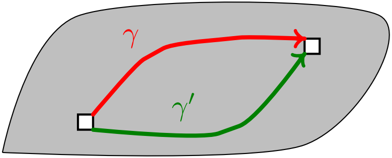


Consistency

Notation: if γ is a σ-path, let $\omega_\sigma(\gamma)$ be the **image vector** of γ.



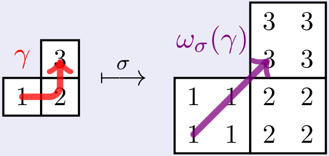
Meaning:
$\sigma(1)$ and $\sigma(3)$
separated by $\omega_\sigma(\gamma)$



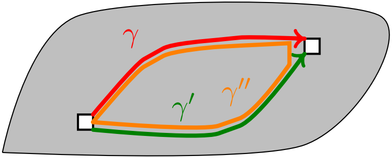
Consistent:
$\omega_\sigma(\gamma) = \omega_\sigma(\gamma')$

Consistency

Notation: if γ is a σ-path, let $\omega_\sigma(\gamma)$ be the **image vector** of γ.



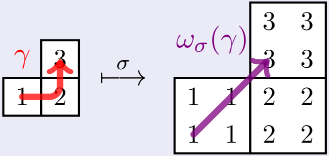
Meaning:
$\sigma(1)$ and $\sigma(3)$
separated by $\omega_\sigma(\gamma)$



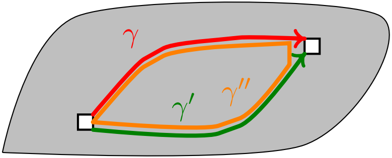
Consistent:
$\omega_\sigma(\gamma) = \omega_\sigma(\gamma')$
$\Leftrightarrow \omega_\sigma(\gamma'') = 0$

Consistency

Notation: if γ is a σ-path, let $\omega_\sigma(\gamma)$ be the **image vector** of γ.



Meaning:
$\sigma(1)$ and $\sigma(3)$
separated by $\omega_\sigma(\gamma)$

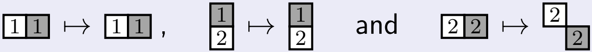


Consistent:
$\omega_\sigma(\gamma) = \omega_\sigma(\gamma')$
$\Leftrightarrow \omega_\sigma(\gamma'') = 0$

Property: σ is consistent $\iff$ $\omega_\sigma(\gamma) = 0$ for all loops γ.

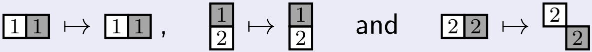
Example: inconsistent substitution

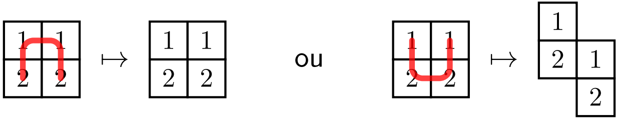
- Base: $\boxed{1} \mapsto \boxed{1}$ and $\boxed{2} \mapsto \boxed{2}$
- Concatenation rules:



Example: inconsistent substitution

- Base: $\boxed{1} \mapsto \boxed{1}$ and $\boxed{2} \mapsto \boxed{2}$
- Concatenation rules:





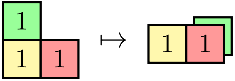
The pattern has two different images, depending on the chosen rules.

Example: overlapping substitution

- Base: $\boxed{1} \mapsto \boxed{1}$
- Concatenation rules: $\boxed{1}\boxed{1} \mapsto \boxed{1}\boxed{1}$ and $\begin{array}{|c|}\hline 1 \\ \hline 1 \\ \hline \end{array} \mapsto \boxed{1}\boxed{1}$

Example: overlapping substitution

- Base: $\boxed{1} \mapsto \boxed{1}$
- Concatenation rules: $\boxed{1}\boxed{1} \mapsto \boxed{1}\boxed{1}$ and $\begin{array}{c}\boxed{1}\\\boxed{1}\end{array} \mapsto \boxed{1}\boxed{1}$



Patterns overlap in the image.

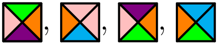
Can **inconsistency** and **overlapping** be detected algorithmically ?

Can **inconsistency** and **overlapping** be detected algorithmically ?

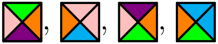
➥ **No.**

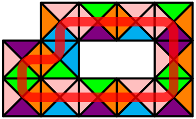
Proof methods: tiling problems.

Wang tiles

- **Wang tiles** $=$ unit squares with colored edges.
  Example: $T = \Big\{$  $\Big\}$.

Wang tiles

- **Wang tiles** = unit squares with colored edges.
  Example: $T = \{$  $\}$.
- **Valid cycle of T:** sequence of tiles $(a_1, \ldots, a_n)$ such that
  1. a_i is a translated copy of a tile of T (no rotations);
  2. a_i and a_{i+1} share one edge and agree in color on it;
  3. $a_1 = a_n$.



Input: a set T of Wang tiles.
Question: Does there exist a valid cycle of T?

➥ Undecidable [Kari 2002].
➥ Let's use it to prove the undecidability of consistency.

Undecidability of consistency

The reduction: $T \ \mapsto \ \sigma_T$

- Alphabet: $T \times \{\rightarrow, \uparrow, \leftarrow, \downarrow\}$

Undecidability of consistency

The reduction: $T \;\mapsto\; \sigma_T$

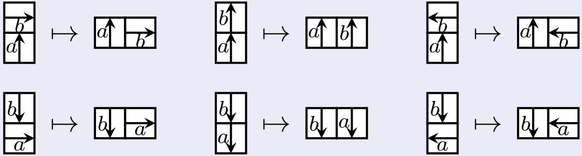
- Alphabet: $T \times \{\rightarrow, \uparrow, \leftarrow, \downarrow\}$
- Base: 

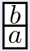
$\forall a \in T$

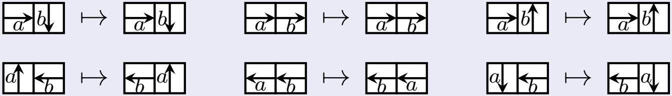
Undecidability of consistency

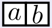
The reduction: $T \;\mapsto\; \sigma_T$

- Alphabet: $T \times \{\rightarrow, \uparrow, \leftarrow, \downarrow\}$
- Base:  $\forall a \in T$
- Concatenation rules:



for tiles $a, b \in T$ such that  match in color, and



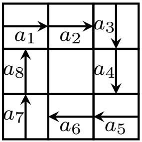
for $a, b \in T$ such that  match in color.

Undecidability of consistency

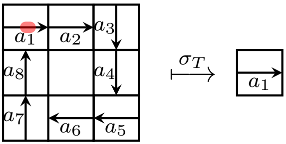
What does σ_T do?

➥ The pointed tile is placed at the right.

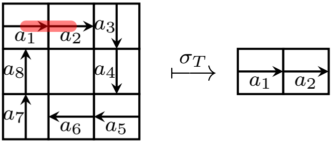
Undecidability of consistency



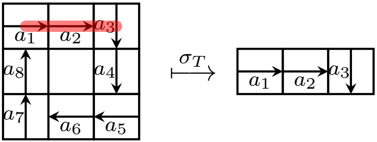
Undecidability of consistency



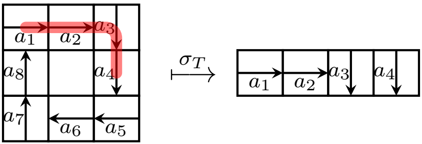
Undecidability of consistency



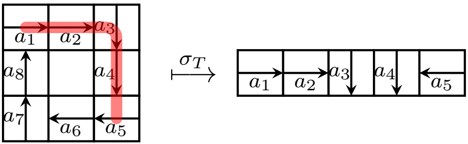
Undecidability of consistency



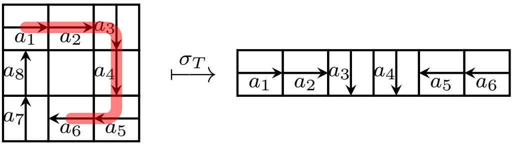
Undecidability of consistency



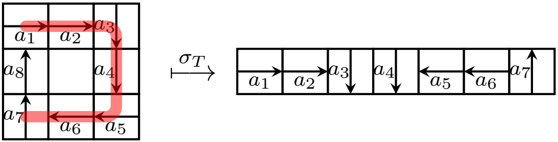
Undecidability of consistency



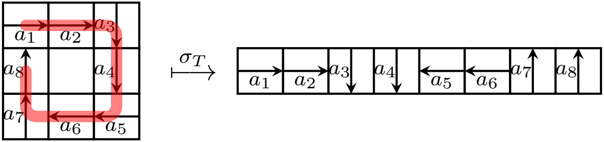
Undecidability of consistency



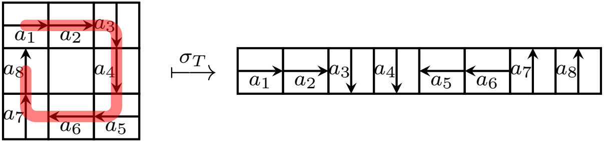
Undecidability of consistency

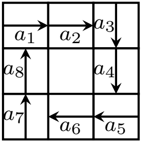


Undecidability of consistency

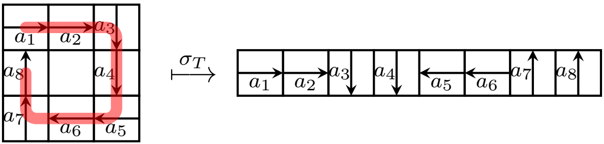


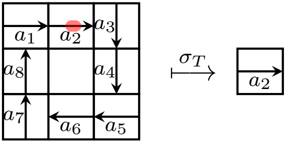
Undecidability of consistency



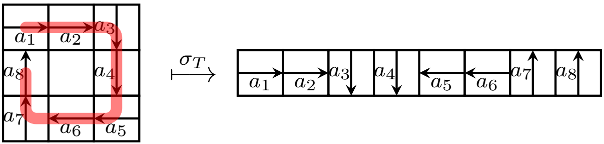


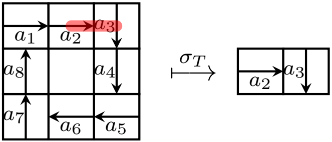
Undecidability of consistency



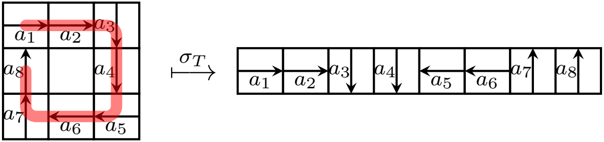


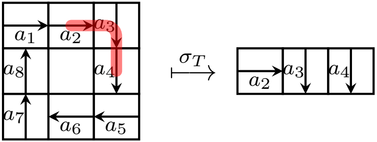
Undecidability of consistency



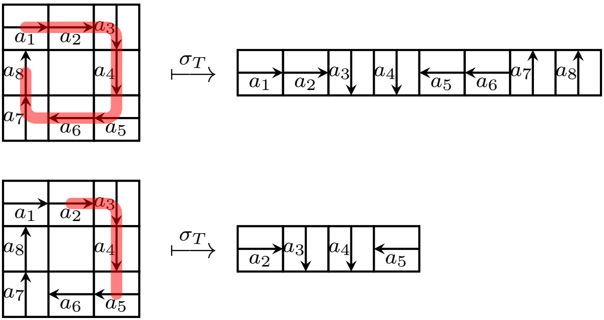


Undecidability of consistency

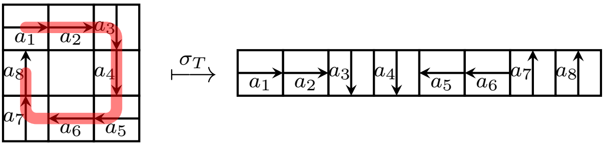


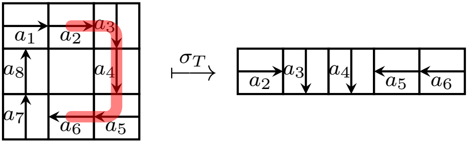


Undecidability of consistency

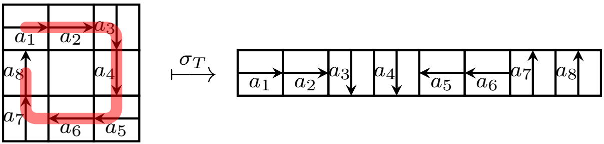


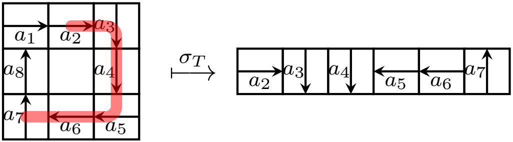
Undecidability of consistency



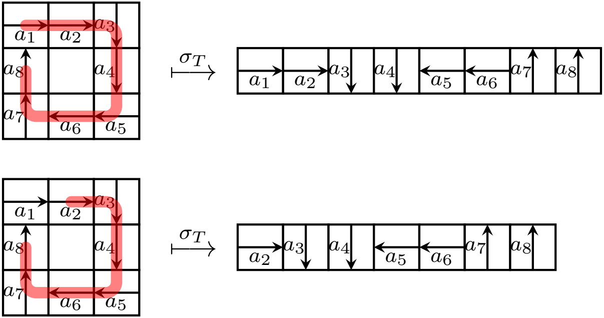


Undecidability of consistency

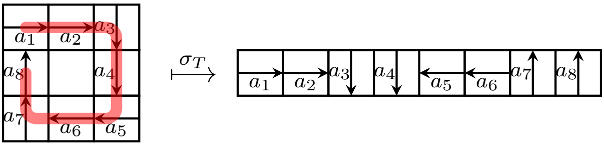


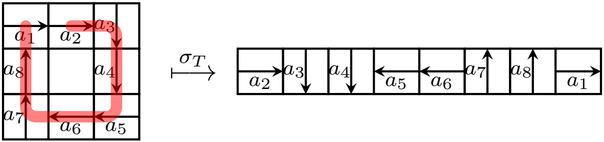


Undecidability of consistency

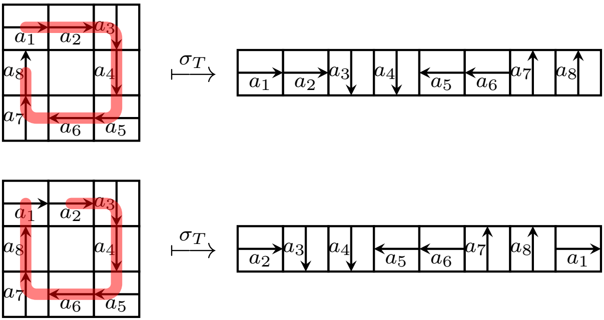


Undecidability of consistency



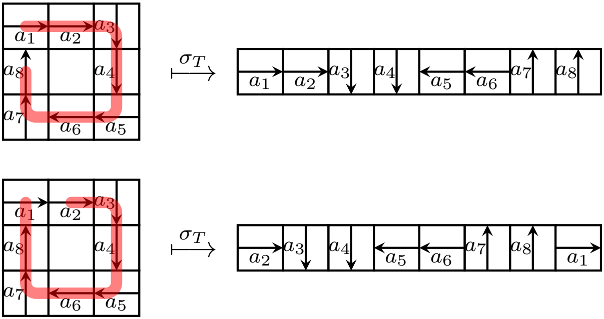


Undecidability of consistency



- **Hence:** $\exists$ a valid cycle of $T \implies \sigma_T$ is inconsistent

Undecidability of consistency



- **Hence:** $\exists$ a valid cycle of $T \implies \sigma_T$ is inconsistent
- (**Conversely:** T is inconsistent
  $\implies$ $\exists$ a simple loop γ
  $\implies$ $\exists$ a valid cycle of T)

➥ **The reduction is complete.** □

Theorem [J.-Kari 2011]

- Consistency is undecidable.
  (We just proved it.)

Theorem [J.-Kari 2011]

- Consistency is undecidable.
  (We just proved it.)
- Overlapping is undecidable.
  (Using another reduction of the same problem.)

Theorem [J.-Kari 2011]

- Consistency is undecidable.
  (We just proved it.)
- Overlapping is undecidable.
  (Using another reduction of the same problem.)

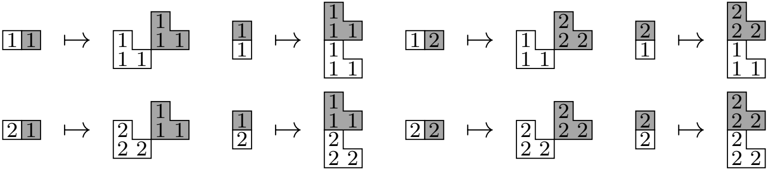
➥ Holds for "reasonnable" classes of substitutions ("domino-to-domino" rules).

Particular case: domino-complete substitutions

Definition

σ is **domino-complete** if there is one rule for every possible domino.

Example:

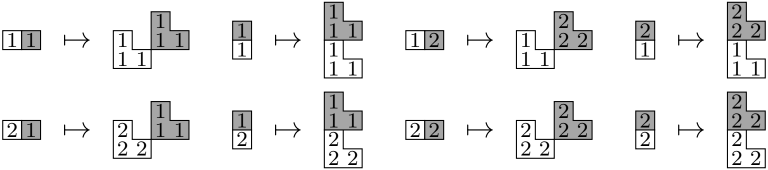


Particular case: domino-complete substitutions

Definition

σ is **domino-complete** if there is one rule for every possible domino.

Example:



We can **decide consistency** for this class.

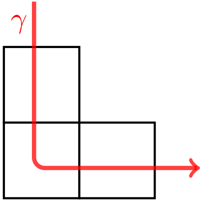
Domino-complete substitutions: consistency

- σ **in**consistent $\Rightarrow$ $\exists$ loop γ such that $\omega_\sigma(\gamma) \neq 0$.

Domino-complete substitutions: consistency

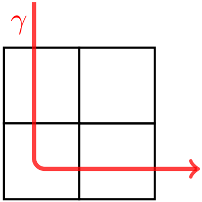
- σ **in**consistent $\Rightarrow$ $\exists$ loop γ such that $\omega_\sigma(\gamma) \neq 0$.

- **Reduce** γ step by step:



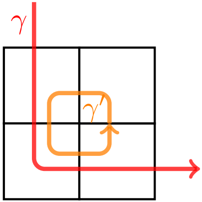
Domino-complete substitutions: consistency

- σ **in**consistent $\Rightarrow$ $\exists$ loop γ such that $\omega_\sigma(\gamma) \neq 0$.
- **Reduce** γ step by step:



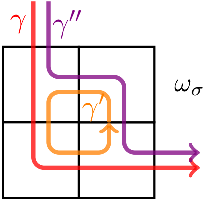
Domino-complete substitutions: consistency

- σ **in**consistent $\Rightarrow$ $\exists$ loop γ such that $\omega_\sigma(\gamma) \neq 0$.
- **Reduce** γ step by step:



Domino-complete substitutions: consistency

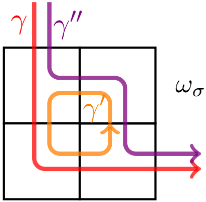
- σ **in**consistent $\Rightarrow$ $\exists$ loop γ such that $\omega_\sigma(\gamma) \neq 0$.
- **Reduce** γ step by step:



$$\omega_\sigma(\gamma) = \omega_\sigma(\gamma') + \omega_\sigma(\gamma'')$$

Domino-complete substitutions: consistency

- σ **in**consistent $\Rightarrow$ $\exists$ loop γ such that $\omega_\sigma(\gamma) \neq 0$.
- **Reduce** γ step by step:

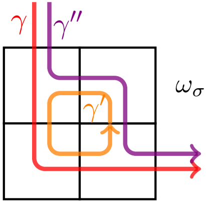


$$\omega_\sigma(\gamma) = \omega_\sigma(\gamma') + \omega_\sigma(\gamma'')$$

- Stop when γ' or γ'' is a 2×2 loop with image vector $\neq 0$.

Domino-complete substitutions: consistency

- σ **in**consistent $\Rightarrow$ $\exists$ loop γ such that $\omega_\sigma(\gamma) \neq 0$.
- **Reduce** γ step by step:



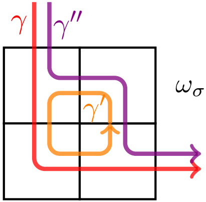
$\omega_\sigma(\gamma) =$
$\omega_\sigma(\gamma') + \omega_\sigma(\gamma'')$

- Stop when γ' or γ'' is a 2×2 loop with image vector $\neq 0$.

So: σ inconsistent $\Longrightarrow$ σ inconsistent on a 2×2 pattern, so:

Domino-complete substitutions: consistency

- σ **in**consistent $\Rightarrow$ $\exists$ loop γ such that $\omega_\sigma(\gamma) \neq 0$.
- **Reduce** γ step by step:



$$\omega_\sigma(\gamma) = \omega_\sigma(\gamma') + \omega_\sigma(\gamma'')$$

- Stop when γ' or γ'' is a 2×2 loop with image vector $\neq 0$.

So: σ inconsistent $\Longrightarrow$ σ inconsistent on a 2×2 pattern, so:

Theorem [J.-Kari 2011]

A domino-complete substitution is consistent
if and only if it is on the 2×2 patterns: **decidable**.

What about overlapping?

We also have:

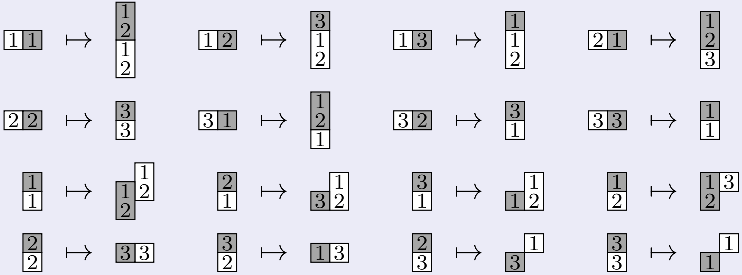
Theorem [J.-Kari 2011]

It is decidable if a consistent domino-complete substitution is overlapping.

Now we study a particular example σ:

Base: $\boxed{1} \mapsto \boxed{\begin{smallmatrix}1\\2\end{smallmatrix}}$, $\quad \boxed{2} \mapsto \boxed{3}$, $\quad \boxed{3} \mapsto \boxed{1}$

Concatenation rules:



➥ "Almost" domino-complete: only $\boxed{2\;3}$ and $\boxed{\begin{smallmatrix}1\\3\end{smallmatrix}}$ are omitted.

Iterations of σ

$\boxed{1}$

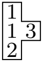
Iterations of σ

1
2

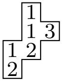
Iterations of σ

1	3
2	

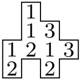
Iterations of σ



Iterations of σ



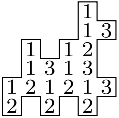
Iterations of σ



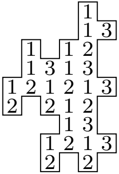
Iterations of σ

		1	
		1	3
	1	2	
	1	3	
1	2	1	3
2		2	

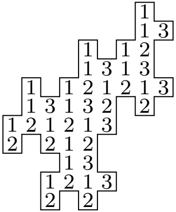
Iterations of σ



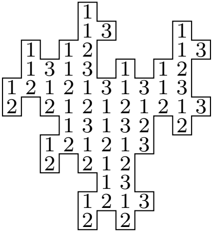
Iterations of σ



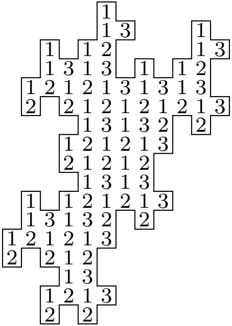
Iterations of σ



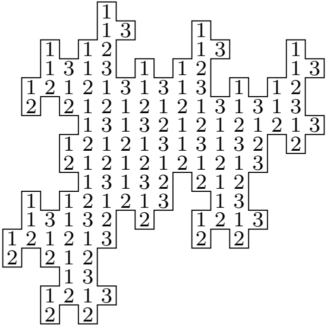
Iterations of σ



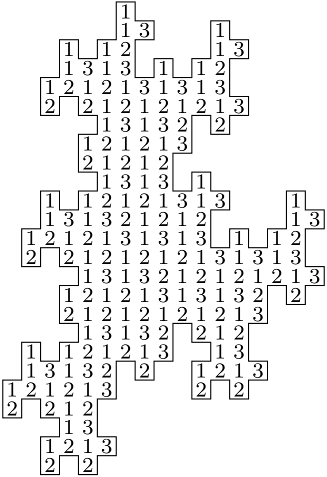
Iterations of σ



Iterations of σ



Iterations of σ



Consistency of σ?

σ is **not** domino-complete: we can't use the previous algorithm. . .

Consistency of σ?

σ is **not** domino-complete: we can't use the previous algorithm...

But, similar reduction arguments yield:

Consistency of σ?

σ is **not** domino-complete: we can't use the previous algorithm...

But, similar reduction arguments yield:

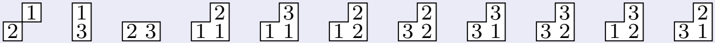
σ is consistent

Consistency of σ?

σ is **not** domino-complete: we can't use the previous algorithm...

But, similar reduction arguments yield:

σ is consistent on every pattern which is contained in a full configuration of $\{1,2,3\}^{\mathbb{Z}^2}$ avoiding the following patterns:



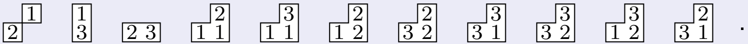
.

Consistency of σ?

σ is **not** domino-complete: we can't use the previous algorithm...

But, similar reduction arguments yield:

σ is consistent on every pattern which is contained in a full configuration of $\{1, 2, 3\}^{\mathbb{Z}^2}$ avoiding the following patterns:

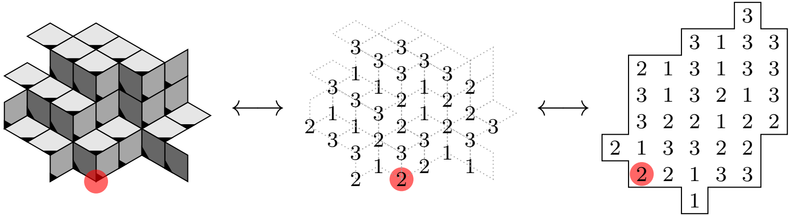


➥ **What?!**

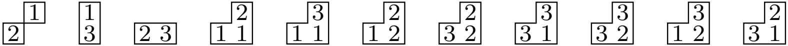
History. . .

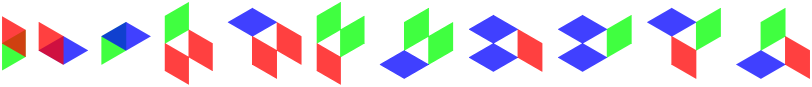
σ is strongly linked with discrete surfaces.
(Ito-Ohtsuki, Arnoux, Berthé, Siegel, Jamet, Fernique,. . .)

Discrete surfaces $\Longleftrightarrow$ configurations of $\{1,2,3\}^{\mathbb{Z}^2}$:



With 11 forbidden patterns:





σ is consistent on every pattern which is contained in a full configuration of $\{1,2,3\}^{\mathbb{Z}^2}$ avoiding the following patterns:

$$\begin{array}{cc} & 1 \\ 2 & \end{array} \quad \begin{array}{c} 1 \\ 3 \end{array} \quad \begin{array}{cc} 2 & 3 \end{array} \quad \begin{array}{cc} & 2 \\ 1 & 1 \end{array} \quad \begin{array}{cc} & 3 \\ 1 & 1 \end{array} \quad \begin{array}{cc} & 2 \\ 1 & 2 \end{array} \quad \begin{array}{cc} & 2 \\ 3 & 2 \end{array} \quad \begin{array}{cc} & 3 \\ 3 & 1 \end{array} \quad \begin{array}{cc} & 3 \\ 3 & 2 \end{array} \quad \begin{array}{cc} & 3 \\ 1 & 2 \end{array} \quad \begin{array}{cc} & 2 \\ 3 & 1 \end{array} .$$

➥ Purely combinatorial proof (without using general results about discrete surfaces).

➥ Decidability of consistency for “almost” domino-complete substitutions.

σ is consistent on every pattern which is contained in a full configuration of $\{1,2,3\}^{\mathbb{Z}^2}$ avoiding the following patterns:

$$\begin{smallmatrix} & 1 \\ 2 & \end{smallmatrix} \quad \begin{smallmatrix} 1 \\ 3 \end{smallmatrix} \quad \begin{smallmatrix} 2 & 3 \end{smallmatrix} \quad \begin{smallmatrix} & 2 \\ 1 & 1 \end{smallmatrix} \quad \begin{smallmatrix} & 3 \\ 1 & 1 \end{smallmatrix} \quad \begin{smallmatrix} & 2 \\ 1 & 2 \end{smallmatrix} \quad \begin{smallmatrix} & 2 \\ 3 & 2 \end{smallmatrix} \quad \begin{smallmatrix} & 3 \\ 3 & 1 \end{smallmatrix} \quad \begin{smallmatrix} & 3 \\ 3 & 2 \end{smallmatrix} \quad \begin{smallmatrix} & 3 \\ 1 & 2 \end{smallmatrix} \quad \begin{smallmatrix} & 2 \\ 3 & 1 \end{smallmatrix} \ .$$

➥ Purely combinatorial proof (without using general results about discrete surfaces).

➥ Decidability of consistency for “almost” domino-complete substitutions.

Non-overlapping ? : not so easy because of the complicated “algebra” of discrete surface σ-paths.

Questions and problems

- Find new examples of substitutions (no "rectangloid" or discrete surfaces substitutions).

Questions and problems

- Find new examples of substitutions (no "rectangloid" or discrete surfaces substitutions).
- Decision problems:
  - Can we iterate σ indefinitely starting from a given pattern?

Questions and problems

- Find new examples of substitutions (no "rectangloid" or discrete surfaces substitutions).
- Decision problems:
  - Can we iterate σ indefinitely starting from a given pattern?
  - Are the images connected? simply connected?

Questions and problems

- Find new examples of substitutions (no "rectangloid" or discrete surfaces substitutions).
- Decision problems:
  - Can we iterate σ indefinitely starting from a given pattern?
  - Are the images connected? simply connected?
  - Do the images contain arbitrarily large squares?
  - ...

Questions and problems

- Find new examples of substitutions (no "rectangloid" or discrete surfaces substitutions).
- Decision problems:
  - Can we iterate σ indefinitely starting from a given pattern?
  - Are the images connected? simply connected?
  - Do the images contain arbitrarily large squares?
  - ...

Thank you