Tile-rewriting grammars for picture languages and associated parsing techniques

PhD Minor research

Student : Daniele Paolo Scarpazza
Affiliation : Politecnico di Milano
Advisor : Prof. Stefano Crespi Reghizzi
Date : February 16th, 2005

In this presentation:

1) What are pictures languages

2) What are Tile-Rewriting Grammars

3) We have a polynomial parsing technique for Tile-Rewriting Grammars!

In this presentation:

1) What are pictures languages

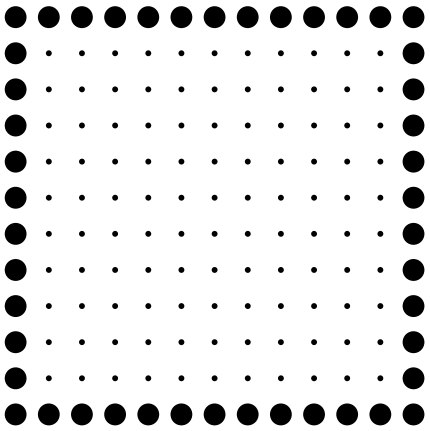
2) What are Tile-Rewriting Grammars

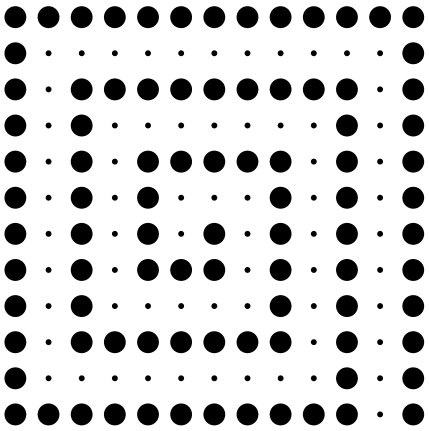
3) We have a polynomial parsing technique for Tile-Rewriting Grammars!

Picture

Given a finite alphabet Σ, a *picture* over Σ is
a rectangular array of elements of Σ.

Example: $\Sigma = \{\bullet, \cdot\}$





We call Σ^{**} the set of pictures is over Σ.

For $h, k \geq 1$, $\Sigma^{(h,k)}$ denotes the set of pictures of size (h, k).

We will use the notation $|p| = (h, k), |p|_{row} = h, |p|_{col} = k$.

A *pixel* is an element $p(i, j)$.

If all pixels are identical to $C \in \Sigma$ the picture is called *homogeneous* and denoted as C-picture.

Subpicture

Let p and q be pictures. q can be a *subpicture* of p at position (i, j), and we write:

$$q \trianglelefteq_{(i,j)} p.$$

Example: if $p = \begin{pmatrix} a & d & g & j & m \\ b & e & h & k & n \\ c & f & i & l & o \end{pmatrix}$ and $q = \begin{pmatrix} e & h & k \\ f & i & l \end{pmatrix}$, then: $q \trianglelefteq_{(2,2)} p$.

Substitution

If p, q, q' are pictures, $q \trianglelefteq_{(i,j)} p$, and q, q' have the same size, then

$$p[q'/q]_{(i,j)}$$

is the picture obtained by replacing in p the occurrence of q at (i, j) with q'.

If $p = \begin{pmatrix} a & d & g & j & m \\ b & e & h & k & n \\ c & f & i & l & o \end{pmatrix}$, $q = \begin{pmatrix} e & h & k \\ f & i & l \end{pmatrix}$, $q' = \begin{pmatrix} Z & Z & Z \\ Z & Z & Z \end{pmatrix}$,
then:

$$p[q'/q]_{(2,2)} = \begin{pmatrix} a & d & g & j & m \\ b & Z & Z & Z & n \\ c & Z & Z & Z & o \end{pmatrix}.$$

Coordinates

A *coordinate* is a couple of positive integers.

If $q \trianglelefteq_{(i,j)} p$, we call $coor_{(i,j)}(q,p)$
the set of coordinates in p where q is located.

$$p = \begin{pmatrix} a & d & g & j & m \\ b & e & h & k & n \\ c & f & i & l & o \end{pmatrix}, \quad q = \begin{pmatrix} e & h & k \\ f & i & l \end{pmatrix},$$

$$coor_{(2,2)}(q,p) = \{(2,2),(2,3),(2,4),(3,2),(3,3),(3,4)\}.$$

Rectangle

Intuitively, we call *rectangle* a set of coordinates of a rectangular area.

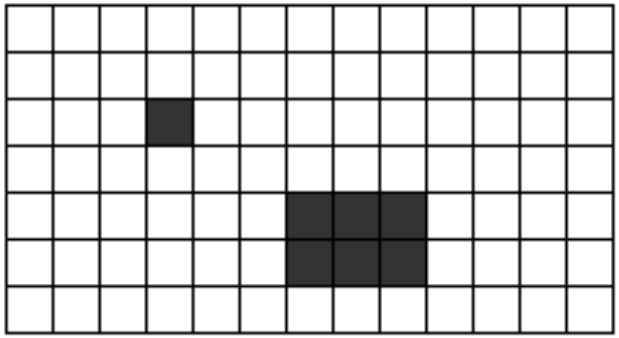
We write rectangles in the following form $r \boxtimes c \boxplus (i, j)$.
That rectangle is r rows high, c columns wide and starts at position (i, j).

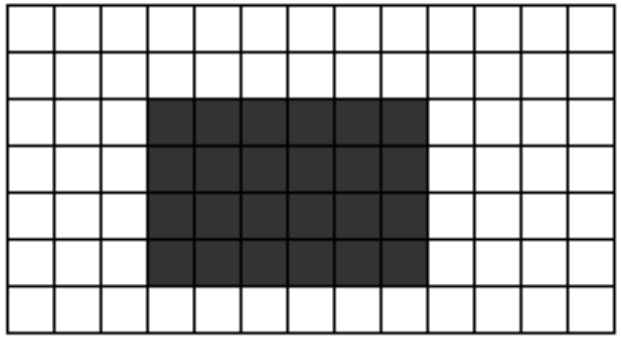
$$p = \begin{pmatrix} a & d & g & j & m \\ b & e & h & k & n \\ c & f & i & l & o \end{pmatrix}, \quad q = \begin{pmatrix} e & h & k \\ f & i & l \end{pmatrix},$$

$coor_{(2,2)}(q, p) = \{(2, 2), (2, 3), (2, 4), (3, 2), (3, 3), (3, 4)\} = 2 \boxtimes 3 \boxplus (2, 2)$

Ceiling

Given a set of coordinates C, the *ceiling* of C, denoted as $\lceil C \rceil$, is the smallest rectangle which is either a superset or equal to C.





$$\lceil (1 \boxtimes 1 \boxplus (3,4)) \cup (2 \boxtimes 3 \boxplus (5,7))) \rceil =$$
$$= \lceil \{(3,4),(5,7),(5,8),(5,9),(6,7),(6,8),(6,9)\} \rceil =$$
$$= 4 \boxtimes 6 \boxplus (3,4)$$

Locally testable language (LOC).

Given a finite set of tiles $\omega = \{t_1, t_2, ...\} \subseteq \Sigma^{(i,j)}$,
$LOC(\omega)$ is the set of those pictures which use all the tiles in ω at least once.

$$\omega = \left\{ \begin{array}{cc} A & A \\ A & A \end{array}, \begin{array}{cc} A & B \\ A & B \end{array}, \begin{array}{cc} B & B \\ B & B \end{array} \right\},$$

$$\begin{array}{ccccc} A & A & B & B & B \\ A & A & B & B & B \end{array} \in LOC(\omega)$$

$$\begin{array}{ccccc} A & A & A & B & B \\ A & A & B & B & B \end{array} \notin LOC(\omega), \qquad \begin{array}{cccc} A & A & A & B \\ A & A & A & B \end{array} \notin LOC(\omega)$$

In this presentation:

1) What are pictures languages

2) What are Tile-Rewriting Grammars

3) We have a polynomial parsing technique for Tile-Rewriting Grammars!

Tile Rewriting Grammar (TRG)

It is a tuple (Σ, N, S, R), where Σ is the *terminal* alphabet, N is a set of *nonterminal* symbols, $S \in N$ is the *starting symbol*, R is a set of *rules*. R may contain two kinds of rules:

Fixed size: $A \to t$

Variable size: $A \to \omega$

A fixed size rule rewrites a A-homogeneous subpicture as t. A variable size rule rewrites a A-homogeneous subpicture as one of the pictures in $LOC(\omega)$.

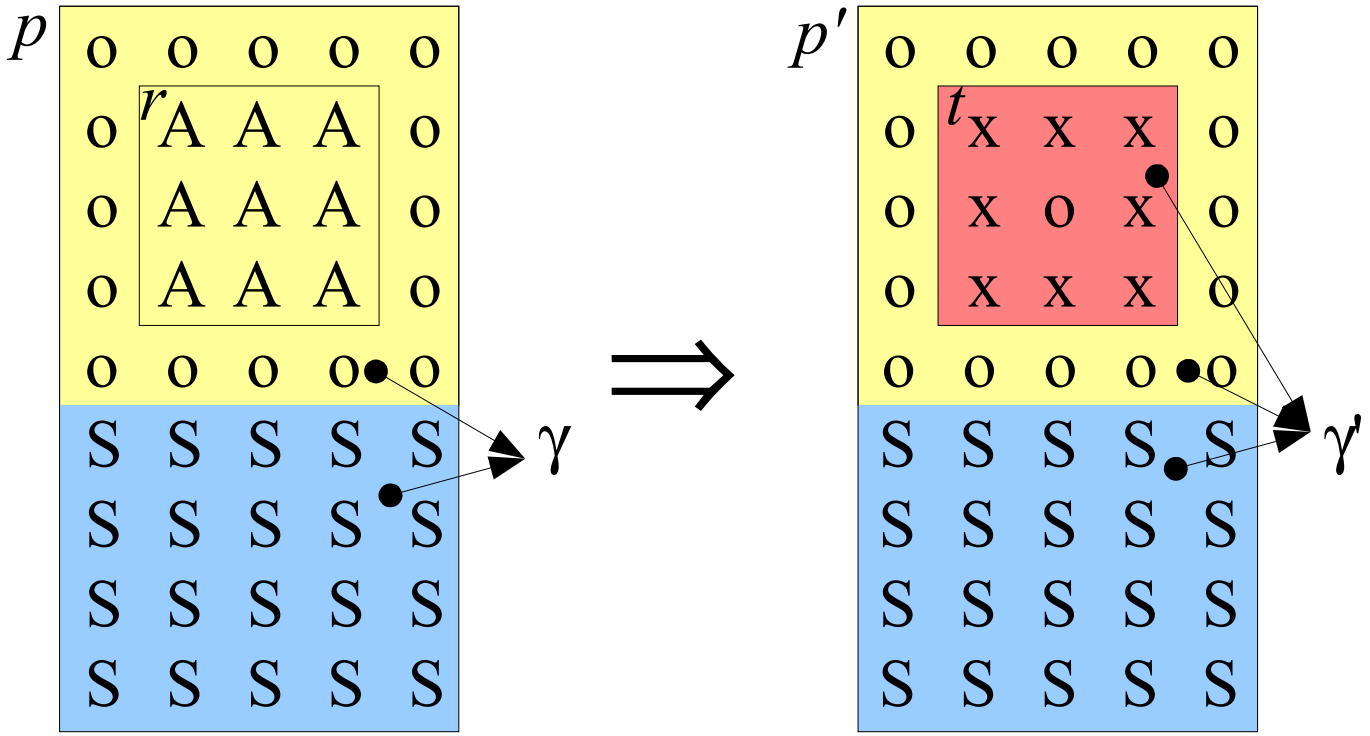
Fixed size rules are not a special case of variable size rules.

Equivalence relation, Maximal subpicture

Let γ be an equivalence relation on $coor(p)$, written $(x, y) \overset{\gamma}{\sim} (x', y')$.

Two subpictures q and q' are equivalent with respect to γ
iff their coordinates are equivalent: $(x, y) \overset{\gamma}{\sim} (x', y')$.

A homogeneous C-subpicture $q \trianglelefteq p$ is *maximal with respect to* γ
iff every equivalent C-subpicture q'
is completely included in q or does not overlap with q.



Derivation in one step $(p, \gamma) \Rightarrow_G (p', \gamma')$:
– there is a rule $A \rightarrow t$ or $A \rightarrow \omega$ in grammar G;
– there is an A-homogeneous subpicture $r \trianglelefteq p$, maximal with respect to γ
– p' is obtained substituting r with picture t, i.e. $p' = p[t/r]$
– in case of variable size rule $t \in LOC(\omega)$
– γ' is equal γ except for the eq. class containing $z = coor_{(i,j)}(r, p)$,
split into z and its complement w.r.t. its equivalence class.

The subpicture r is named the *application area* in the derivation step.

Derivation in n steps is a trivial extension.

The *picture language* defined by a grammar G (written $L(G)$) is the set of $p \in \Sigma^{**}$ such that, if $|p| = (h, k)$, then

$$\left(S^{(h,k)}, coor(p) \times coor(p)\right) \stackrel{*}{\Rightarrow}_G (p, \gamma) \tag{1}$$

where γ is arbitrary.

For short we write $S \stackrel{*}{\Rightarrow}_G p$.

Example: Chinese boxes.

$G = (\Sigma, N, S, R)$, where $\Sigma = \{\ulcorner, \urcorner, \llcorner, \lrcorner, \circ\}$, $N = \{S\}$,
and R consists of the following rules:

$$S \to \begin{array}{cc} \ulcorner & \urcorner \\ \llcorner & \lrcorner \end{array};$$

$$S \to \left\{ \begin{array}{cccccccccccccccccccc} \ulcorner & \circ & , & \circ & S & , & \circ & S & , & S & S & , & \circ & \circ & , & S & S & , & \circ & \urcorner \\ \circ & S & , & \llcorner & \circ & , & \circ & S & , & \circ & \circ & , & S & S & , & S & S & , & S & \circ \\ & & & & & & & & & & & & & & & & & & & \lrcorner \end{array} \right\}$$

Example picture in $L(G)$:

$$\begin{array}{cccccc} \ulcorner & \circ & \circ & \circ & \circ & \urcorner \\ \circ & \ulcorner & \circ & \circ & \urcorner & \circ \\ \circ & \circ & \ulcorner & \urcorner & \circ & \circ \\ \circ & \circ & \llcorner & \lrcorner & \circ & \circ \\ \circ & \llcorner & \circ & \circ & \lrcorner & \circ \\ \llcorner & \circ & \circ & \circ & \circ & \lrcorner \end{array}$$

For convenience, we will often specify a set of tiles by a sample picture exhibiting the tiles as its subpictures.

We write | to separate alternative right parts of rules.

The previous grammar becomes:

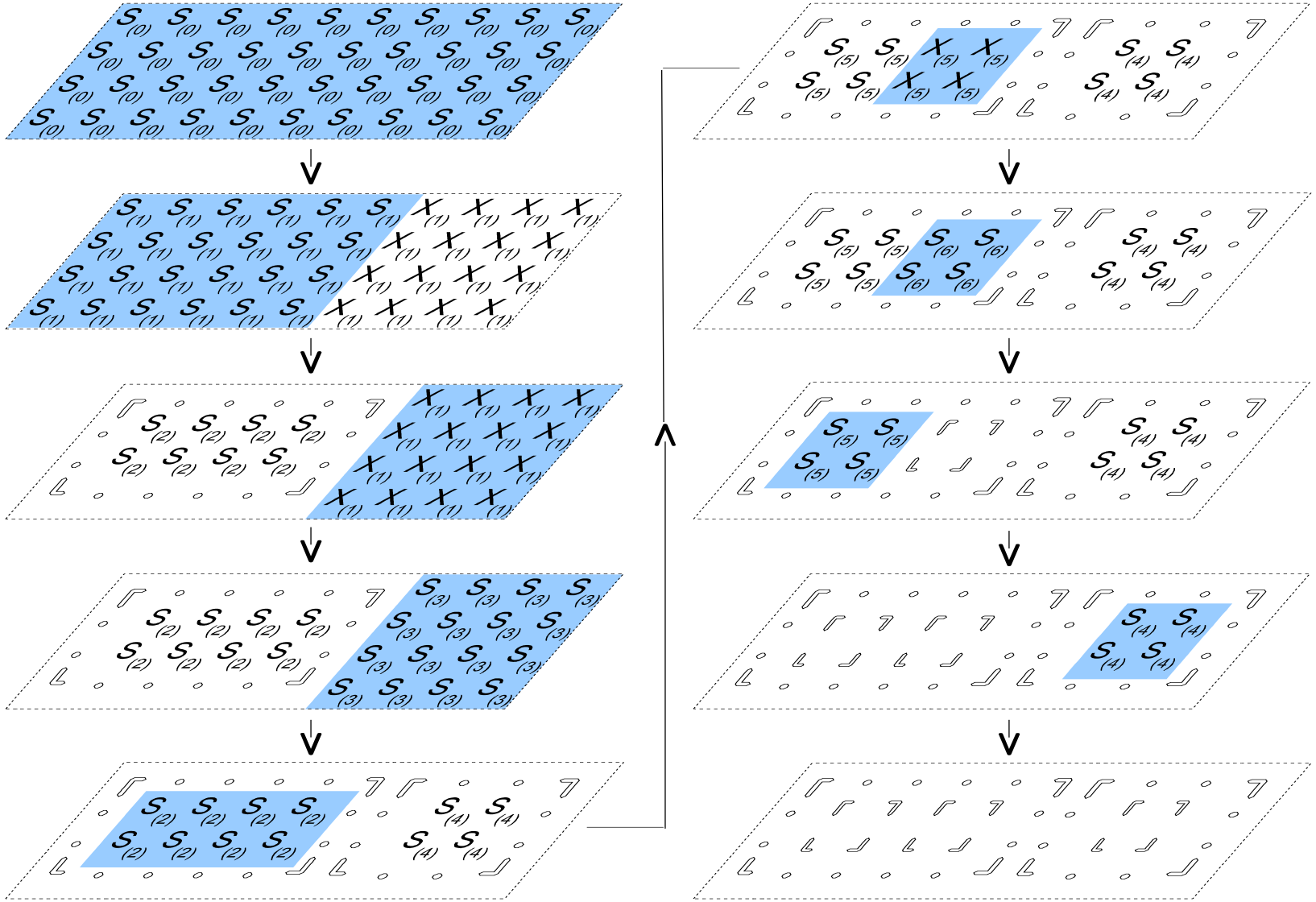
$$S \to \begin{array}{|cc|}\hline \ulcorner & \urcorner \\ \llcorner & \lrcorner \\ \hline \end{array} \mid B_{2,2}\begin{pmatrix} \ulcorner & \circ & \circ & \urcorner \\ \circ & S & S & \circ \\ \circ & S & S & \circ \\ \llcorner & \circ & \circ & \lrcorner \end{pmatrix}$$

Example: 2D Dyck analogue.

$$S \to \begin{matrix} \ulcorner & \urcorner \\ \llcorner & \lrcorner \end{matrix} \mid \left[\!\!\left[\begin{matrix} \ulcorner & \circ & \circ & \urcorner \\ \circ & S & S & \circ \\ \circ & S & S & \circ \\ \llcorner & \circ & \circ & \lrcorner \end{matrix} \right]\!\!\right] \mid \left[\!\!\left[\begin{matrix} S & S & X & X \\ S & S & X & X \end{matrix} \right]\!\!\right] \mid \left[\!\!\left[\begin{matrix} S & S \\ S & S \\ X & X \\ X & X \end{matrix} \right]\!\!\right]$$

$$X \to \left[\!\!\left[\begin{matrix} S & S \\ S & S \end{matrix} \right]\!\!\right]$$

where $[\![p]\!]$ is a shorthand for the set of 2×2 tiles of p.



In this presentation:

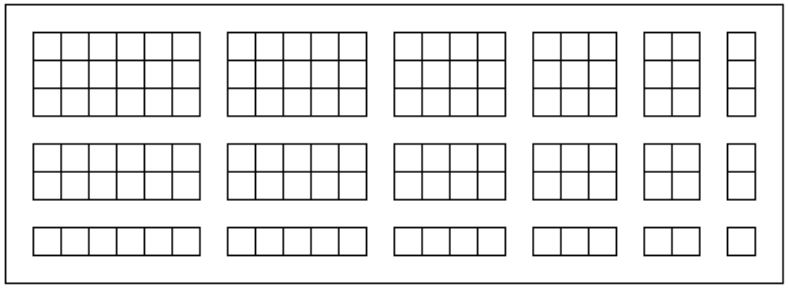
1) What are pictures languages

2) What are Tile-Rewriting Grammars

3) We have a polynomial parsing technique for Tile-Rewriting Grammars!

Tableau

- A *tableau* is a matrix of variable-size matrices.
  A $m \times n$ tableau T contains $m \times n$ matrices, each denoted by $T_{i,j}$.
- Matrix $T_{i,j}$ has size $(m - i + 1, n - j + 1)$.
  The notation $T[i \boxtimes j \boxplus (a, b)]$ indicates the (a, b) element of matrix $T_{i,j}$, or $(T_{i,j})_{a,b}$.
- Example: the following figure shows a 3×6 tableau.



- each elementary cell in a tableau corresponds to a subpicture.

Candidates

- our algorithm uses one tableau T, to store candidates;
- *candidate* is a triple (R_e, ω_x, α) such that
  R_e is a rule, ω_x is a set of missing tiles, $\alpha \subseteq \mathcal{R}(m \boxtimes n)$ is a rectangle;
- Example: element $(R_3, \{t_{3,2}, t_{3,3}, t_{3,5}\}, ...)$ in a tableau cell indicates that the subpicture corresponding to that tableau cell only uses tiles present in the right-part of rule R_3, and tiles $t_{3,2}, t_{3,3}$, are $t_{3,5}$ not used in the picture.

Monopicture, multipicture

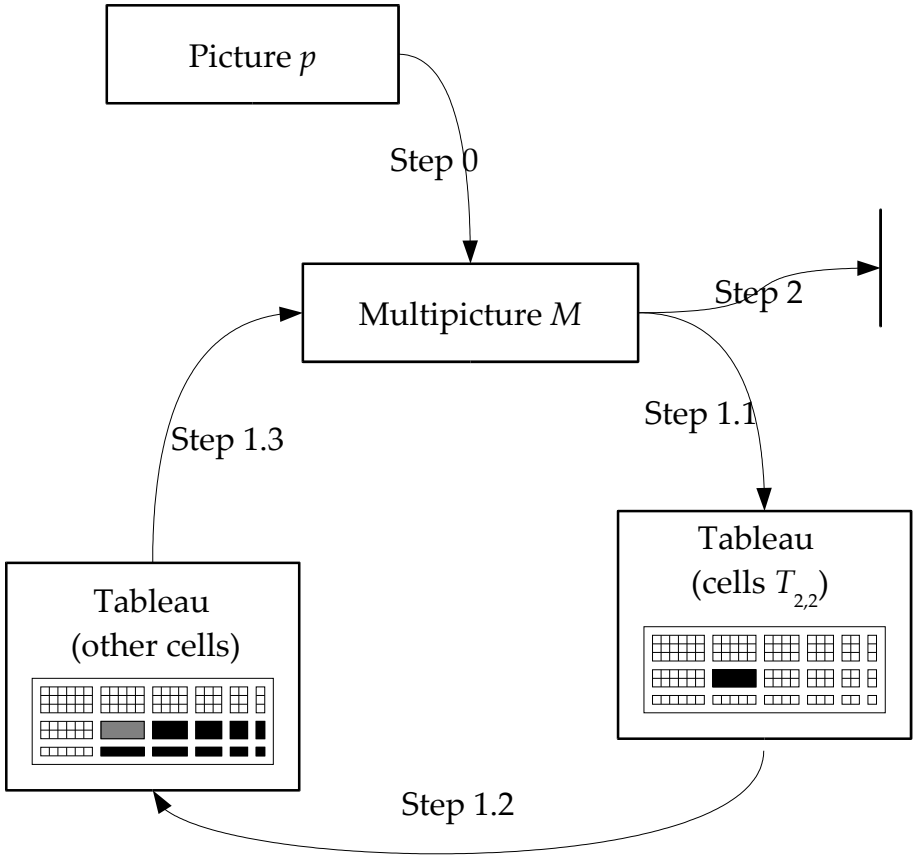
- Each cell of a monopicture M of size (m, n) over alphabet Σ contains exactly one couple (symbol, rectangle) such as (A, u), where $A \in \Sigma$, and u is a rectangle contained in $m \boxtimes n$.

- We call u the *scope* of symbol A.

- In multipictures, each cell contains zero or more such couples.

- our algorithm uses one multipicture M, to store recognized application areas;

The parsing technique at a glance



Step 0: initialize T and M;
Step 1: repeat until fixed point:
– in step 1.1 *match* we examine M and add candidates to $T_{2,2}$;
– in step 1.2 *grow* we examine $T_{2,2}$ and add candidates to $T_{*,*}$;
– in step 1.3 *recognize* we examine $T_{*,*}$ and add couples to M.
Step 2: *declare*;

T: which subpictures are tileable by which rule;
A: which activation areas have been recognized so far;

Our algorithm – step # 1.1

Step 1.1: update each cell $T[2 \boxtimes 2 \boxplus (i,j)]$, adding elements: if

$$R_e = (Q \to \omega) \in R, \qquad t \in \omega, \qquad t' \in (t :: M[2 \boxtimes 2 \boxplus (i,j)]),$$

$$t' = \begin{pmatrix} (t_{1,1}, \alpha_{1,1}) & (t_{1,2}, \alpha_{1,2}) \\ (t_{2,1}, \alpha_{2,1}) & (t_{2,2}, \alpha_{2,2}) \end{pmatrix}$$

then:

$$(R_e, \omega - t, \lceil \alpha_{1,1} \cup \alpha_{1,2} \cup \alpha_{2,1} \cup \alpha_{2,2} \rceil) \in T[2 \boxtimes 2 \boxplus (i,j)] \ ;$$

Our algorithm – step # 1.2

Step 1.2: update T adding elements:
$\forall (r,c) \mid 2 < r \leq m, 2 < c \leq n$ in lexicographical order: if

$$(R_e, \omega_1, \alpha_1) \in T[r \boxtimes c-1 \boxplus (i,j)] \quad \wedge \quad (R_e, \omega_2, \alpha_2) \in T[r \boxtimes c-1 \boxplus (i,j+1)] \text{ or}$$
$$(R_e, \omega_1, \alpha_1) \in T[r-1 \boxtimes c \boxplus (i,j)] \quad \wedge \quad (R_e, \omega_2, \alpha_2) \in T[r-1 \boxtimes c \boxplus (i+1,j)]$$

then

$$(R_e, \omega_1 \cap \omega_2, \lceil \alpha_1 \cup \alpha_2 \rceil) \in T[r \boxtimes c \boxplus (i,j)] \ ;$$

Comment: elements of two horizontally adjacent tableau cells $m \boxtimes n$ can be merged into a corresponding $m \boxtimes (n+1)$ cell. Similarly for vertically adjacent cells. Given the same rule, the set of missing tiles is the intersection of the sets of missing tiles, and the scope is the ceiling of the union of scopes.

Our algorithm – step # 1.3

Step 1.3: update M adding elements: for each element such that

$$(R_e, \emptyset, \alpha) \in T[r \boxtimes c \boxplus (i,j)] \;\mid\; \alpha \subseteq r \boxtimes c \boxplus (i,j),\; R_e = (A \rightarrow ...)$$

add elements:

$$\forall (i,j) \in r \boxtimes c \boxplus (i,j) \qquad (A, r \boxtimes c \boxplus (i,j)) \in M_{i,j}\ ;$$

Comment: whenever all the tiles of subpicture $r \boxtimes c \boxplus (i,j)$ appear in the right part of rule R_e, and no tile is missing $(R_e, \emptyset, ...)$ and the scope α of all the recognized symbols is inside the current subpicture, then recognize the subpicture as application area of rule R_e. Update multipicture, adding a couple $(A, r \boxtimes c \boxplus (i,j))$ in every cell of sub-multipicture $r \boxtimes c \boxplus (i,j)$.

Our algorithm – final step

Step 2: to be executed when fixed point is reached:
if $\forall (i,j) \in r \boxtimes c \qquad (S, m \boxtimes n \boxplus (1,1)) \in M_{i,j}$ then:

$$\text{declare} \qquad p \in \mathcal{L}(G)$$

else

$$\text{declare} \qquad p \notin \mathcal{L}(G).$$

Comment: the recognition of a rule which has the starting symbol S as the left-hand side, and the whole picture as application area ($m \boxtimes n \boxplus (1,1)$), indicates that the picture is recognized. In short, the multipicture must contain element $(S, m \times n)$ in every cell. If such area is not recognized and a fixed point is reached, the picture is not in the language.

Compatibility

Given a picture p and a multipicture M,
the *compatibility* between p and M (denoted as $p :: M$) is
the set of monopictures p' such that:

for each cell: $p'_{i,j} = (p_{i,j}, \alpha_{i,j}) \in M_{i,j}$

for each pair: $(\alpha_{i,j} = \alpha_{k,l} \wedge p_{i,j} = p_{k,l}) \vee (\alpha_{i,j} \cap \alpha_{k,l} = \emptyset)$.

We say that p is *compatible* with M iff $p :: M \neq \emptyset$.

p

x	**x**	**x**	x	x	x
x	**o**	**o**	o	o	x
x	**o**	**o**	o	o	x
x	**x**	**x**	x	x	x

>

p'

A	A	A	x	x	x
A	A	A	o	o	x
A	A	A	o	o	x
A	A	A	x	x	x

M

x, 1x1+(1,1)	**x, 1x1+(1,2)**	**x, 1x1+(1,3)**	x, 1x1+(1,4)	x, 1x1+(1,5)	x, 1x1+(1,6)
x, 1x1+(2,1)	**o, 1x1+(2,2)**	**o, 1x1+(2,3)**	o, 1x1+(2,4)	o, 1x1+(2,5)	x, 1x1+(2,6)
x, 1x1+(3,1)	**o, 1x1+(3,2)**	**o, 1x1+(3,3)**	o, 1x1+(3,4)	o, 1x1+(3,5)	x, 1x1+(3,6)
x, 1x1+(4,1)	**x, 1x1+(4,2)**	**x, 1x1+(4,3)**	x, 1x1+(4,4)	x, 1x1+(4,5)	x, 1x1+(4,6)

>

M'

x, 1x1+(1,1) **A, 4x3+(1,1)**	**x, 1x1+(1,2)** **A, 4x3+(1,1)**	x, 1x1+(1,3) A, 4x3+(1,1)	x, 1x1+(1,4)	x, 1x1+(1,5)	x, 1x1+(1,6)
x, 1x1+(2,1) **A, 4x3+(1,1)**	**o, 1x1+(2,2)** **A, 4x3+(1,1)**	o, 1x1+(2,3) A, 4x3+(1,1)	o, 1x1+(2,4)	o, 1x1+(2,5)	x, 1x1+(2,6)
x, 1x1+(3,1) A, 4x3+(1,1)	o, 1x1+(3,2) A, 4x3+(1,1)	o, 1x1+(3,3) A, 4x3+(1,1)	o, 1x1+(3,4)	o, 1x1+(3,5)	x, 1x1+(3,6)
x, 1x1+(4,1) A, 4x3+(1,1)	x, 1x1+(4,2) A, 4x3+(1,1)	x, 1x1+(4,3) A, 4x3+(1,1)	x, 1x1+(4,4)	x, 1x1+(4,5)	x, 1x1+(4,6)

Example: pictures (p, p') and corresponding multipictures (M, M') in the first step of a recognition chain. Please consider the sub-multipicture $M'[2 \boxtimes 2 \boxplus (1,1)]$, bordered in the figure. For it, compatibilities with two tiles are reported:

$$M'[2 \boxtimes 2 \boxplus (1,1)] \quad :: \quad \begin{matrix} A & A \\ A & A \end{matrix} = \left\{ \begin{matrix} (A, 4\boxtimes 3 \boxplus (1,1)) & (A, 4\boxtimes 3 \boxplus (1,1)) \\ (A, 4\boxtimes 3 \boxplus (1,1)) & (A, 4\boxtimes 3 \boxplus (1,1)) \end{matrix} \right\}$$

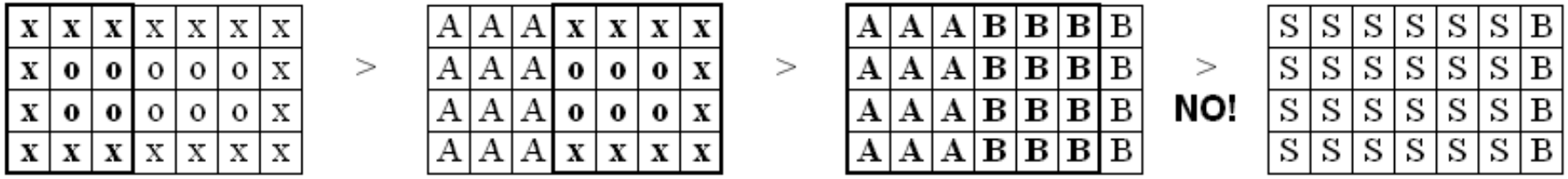
$$M'[2 \boxtimes 2 \boxplus (1,1)] \quad :: \quad \begin{matrix} x & x \\ x & o \end{matrix} = \left\{ \begin{matrix} (x, 1\boxtimes 1 \boxplus (1,1)) & (x, 1\boxtimes 1 \boxplus (1,2)) \\ (x, 1\boxtimes 1 \boxplus (2,1)) & (o, 1\boxtimes 1 \boxplus (2,2)) \end{matrix} \right\}$$

Compatibility 2

- In a derivation, each application areas is disjoint or included in a previous one:

$$\alpha_1, \ldots, \alpha_i, \alpha_{i+1}, \ldots, \alpha_{i+j}, \ldots \qquad (\alpha_i \cap \alpha_{i+j} = \emptyset) \quad \vee \quad (\alpha_i \supseteq \alpha_{i+j}) \ .$$

- In bottom-up recognition, enforcing the reversed version: the *all-or-nothing* principle.
- Example of violation:



- We enforce the principle with *multipictures*, i.e. pictures containing couples (symbol, area). Example, each B would be replaced by $(B, 4 \boxtimes 4 \boxplus (1,4))$.
  The last step would be illegal because:

$$(4 \boxtimes 6 \boxplus (1,1)) \not\supseteq (4 \boxtimes 4 \boxplus (1,4)), \qquad (4 \boxtimes 6 \boxplus (1,1)) \cap (4 \boxtimes 4 \boxplus (1,4) \neq \emptyset).$$

Informal comments

- The algorithm scans the tableau and the multipicture several times;
  at each scan, one or more disjoint application areas are recognized;
- each tableau cell $T[r \boxtimes c \boxplus (i,j)]$ is filled with candidates (R_e, ω, α)
  where if rule $R_e = A \longrightarrow \omega_e$ exists, then

$$\omega = \omega_e - B_{2,2}(M[r \boxtimes c \boxplus (i,j)]).$$

- When $(R_e, \emptyset, \alpha) \in T[r \boxtimes c \boxplus (i,j)]$, then
  exactly all the tiles in rule R_e are used in subpicture $M[r \boxtimes c \boxplus (i,j)]$;

- moreover α is the ceiling of the activation areas of the symbols in the multipicture which comply with rule R_e.
- If α is not smaller or equal to the subrectangle $r \boxtimes c \boxplus (i,j)$ we discard the rule,
  since it would violate the theorem of disjointness of application areas.

- Every time a rule $R_e = (A \rightarrow \ldots)$ is recognized over an activation area q, a couple (A, q) is added to every cell of $M[q]$, so that the next iteration of the algorithm can use the newly added symbol to recognize larger activation areas.

Copy rules are not needed

If $G = (\Sigma, N, S, R)$ is a TRG, and $A, B \in N$, a rule in the form

$$A \rightarrow \left\{ \begin{array}{cc} B & B \\ B & B \end{array} \right\}$$

is said to be a *copy rule*;

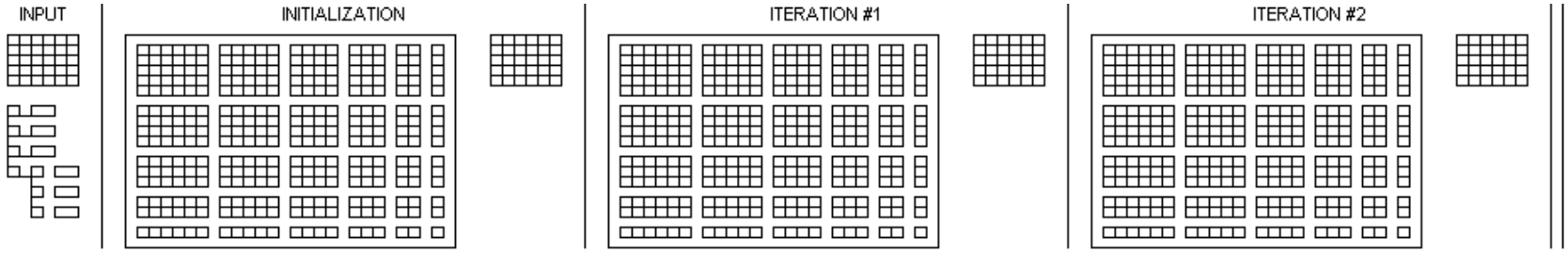
For every TRG G with copy rules,
$\exists G' \mid \mathcal{L}(G) = \mathcal{L}(G')$ and G' is free from copy rules;

With no loss of generality, our algorithm operates
on TRG grammars free from copy rules.

Time complexity

- input picture has dimensions $m \times n$, the input size is $N = m \cdot n$;
- complexity is dominated by step 1.2;
  - this step fills a $m \times n$ tableau which has $\frac{m(m-1)n(n-1)}{4} \leq N^2$ cells;
  - for each cell, it compares couples of elements of exactly two other cells;
  - elements are in the form $(R_e, \omega, \alpha) \Rightarrow$ they can be at most $|R| \cdot max_i|\omega_{R_i}| \cdot |\mathcal{R}(m \times n)|$,
    where $k = |R| \cdot max_i|\omega_{R_i}|$ is fixed, while $|\mathcal{R}(m \times n)| \leq N^2$;
  - therefore, the time complexity of each comparison is $\leq k^2 N^4$,
    and the overall complexity of step 1.2 is $\leq k^2 N^6$.
- Steps 1.1–1.3 (therefore 1.2) are iterated until fixed point.
  At every iteration, one or more application areas detected.
  No copy rules $\Rightarrow$ an application area can appear at most once in a derivation.
- Therefore # of iterations $\leq$ # application areas $\leq N^2$.
- Therefore, the time complexity of the whole algorithm is $O(N^8)$.

A full example

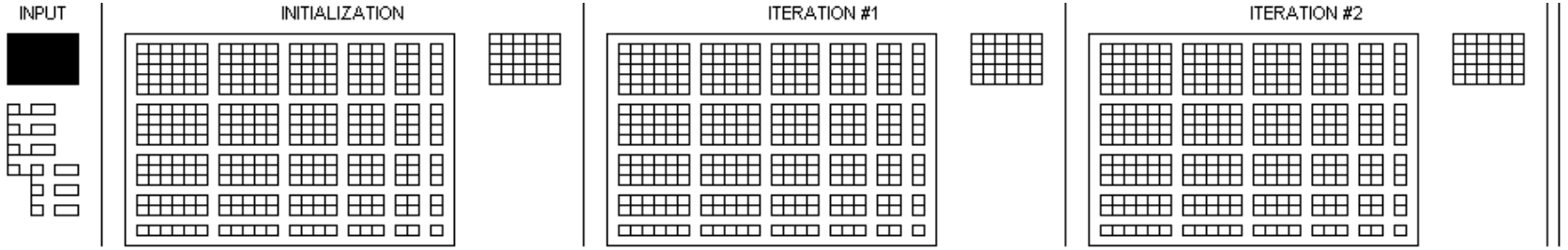


In the next slides we fully develop an example, with the help of the above storyboard. Elements updated in each slide will be marked in black. Elements will be illustrated in this order:

- the input picture and the input grammar;
- the result of initialization on the tableau T and on the multipicture M will be shown;
- the effects of the first iteration;
- the effects of the second iteration (and conclusion).

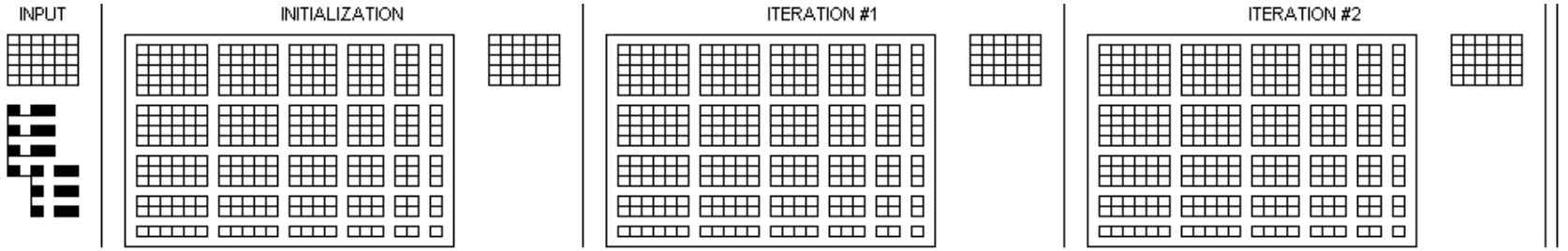
For sake of clarity, T and M at initialization, iteration 1 and 2 are represented as separate entities.

A full example: the input picture



$$p = \begin{array}{|c|c|c|c|c|c|} \hline x & x & x & x & x & x \\ \hline x & o & o & o & o & x \\ \hline x & o & o & o & o & x \\ \hline x & o & o & o & o & x \\ \hline x & x & x & x & x & x \\ \hline \end{array}$$

A full example: the input grammar (1/2)



$$G = (\Sigma, N, S, R)$$

$$\Sigma = \{x, o\}$$

$$N = \{S, A, B\}$$

$$R_1 : S \to B_{2,2} \begin{pmatrix} A & A & B & B \\ A & A & B & B \end{pmatrix}$$

$$R_2 : A \to B_{2,2} \begin{pmatrix} x & x & x \\ x & o & o \\ x & o & o \\ x & x & x \end{pmatrix}$$

$$R_3 : B \to B_{2,2} \begin{pmatrix} x & x & x \\ o & o & x \\ o & o & x \\ x & x & x \end{pmatrix}$$

A full example: the input grammar (2/2)

For ease of reference, we give names to the tiles in each right-hand side.

$$R_1 \quad : \quad S \to B_{2,2} \begin{pmatrix} A & A & B & B \\ A & A & B & B \end{pmatrix} = \left\{ t_{1,1} = \begin{pmatrix} A & A \\ A & A \end{pmatrix}, t_{1,2} = \begin{pmatrix} A & B \\ A & B \end{pmatrix}, t_{1,3} = \begin{pmatrix} B & B \\ B & B \end{pmatrix} \right\}$$

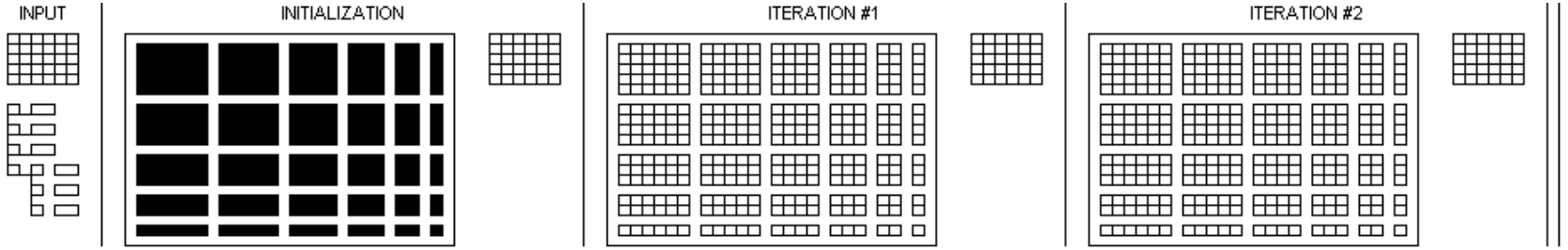
$$R_2 \quad : \quad A \to B_{2,2} \begin{pmatrix} x & x & x \\ x & o & o \\ x & o & o \\ x & x & x \end{pmatrix} = \left\{ t_{2,1} = \begin{pmatrix} x & x \\ x & o \end{pmatrix}, t_{2,2} = \begin{pmatrix} x & x \\ o & o \end{pmatrix}, t_{2,3} = \begin{pmatrix} x & o \\ x & o \end{pmatrix}, \right.$$

$$\left. t_{2,4} = \begin{pmatrix} x & o \\ x & x \end{pmatrix}, t_{2,5} = \begin{pmatrix} o & o \\ x & x \end{pmatrix}, t_{2,6} = \begin{pmatrix} o & o \\ o & o \end{pmatrix} \right\}$$

$$R_3 \quad : \quad B \to B_{2,2} \begin{pmatrix} x & x & x \\ o & o & x \\ o & o & x \\ x & x & x \end{pmatrix} = \left\{ t_{3,1} = \begin{pmatrix} x & x \\ o & x \end{pmatrix}, t_{3,2} = \begin{pmatrix} x & x \\ o & o \end{pmatrix}, t_{3,3} = \begin{pmatrix} o & x \\ o & x \end{pmatrix}, \right.$$

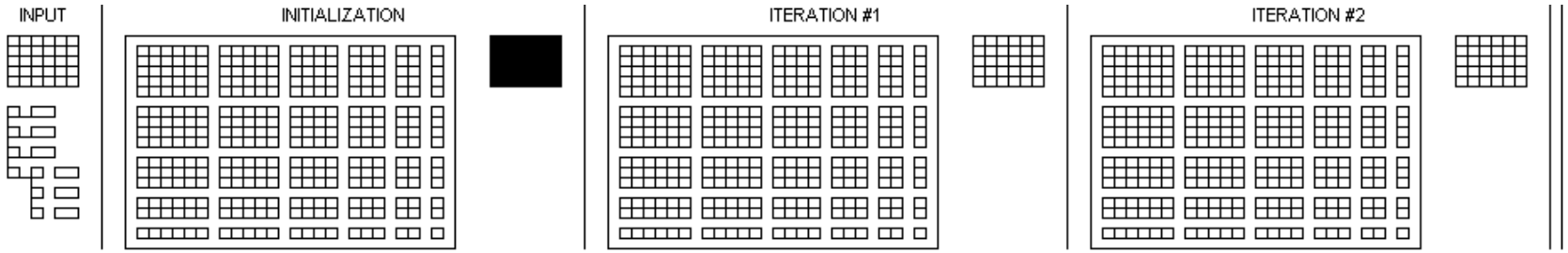
$$\left. t_{3,4} = \begin{pmatrix} o & x \\ x & x \end{pmatrix}, t_{3,5} = \begin{pmatrix} o & o \\ x & x \end{pmatrix}, t_{3,6} = \begin{pmatrix} o & o \\ o & o \end{pmatrix} \right\}$$

A full example: tableau initialization



- At initialization time, the tableau is empty.
- Being the input picture dimensions equal to 6×5, also the tableau T will have 6×5 cells.
- The first cell, $T_{1,1}$, will be a 6×5 matrix.
  Cells in this matrix represent each 1×1 tile in the input picture.
- $T_{1,2}$, will be a 6×4 matrix, i.e. with one less column, and so on, up to $T_{1,6}$, which is a 6×1 matrix. Same considerations apply to the second and following rows.
- The last cell, $T_{5,6}$, is a matrix composed by a single cell, representing the whole input picture.

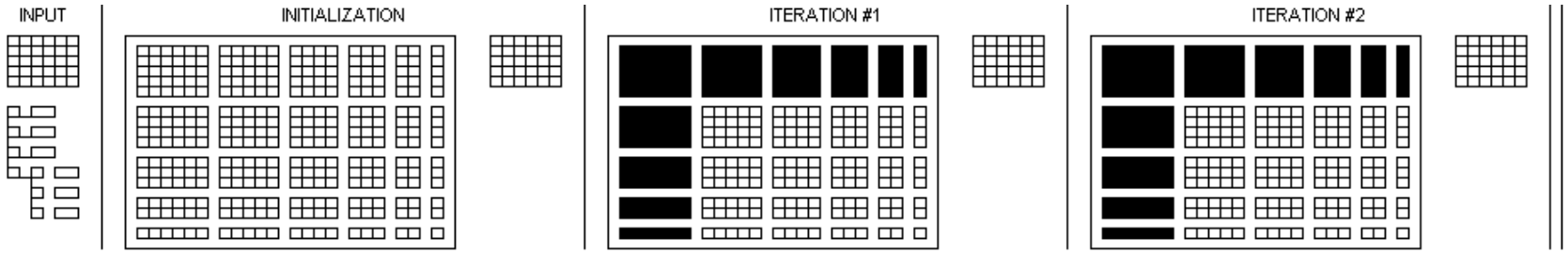
A full example: multipicture initialization



The multipicture is initially set to the contents of the picture. The *scope* of each terminal symbol is set to the 1×1 cell where the symbol belong.

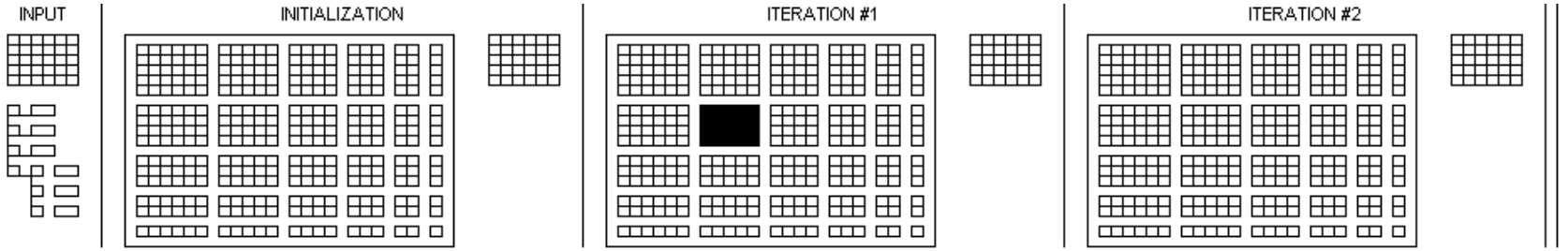
$$M = \begin{array}{|c|c|c|c|c|c|} \hline (x, 1\boxtimes 1 \boxplus (1,1)) & (x, 1\boxtimes 1 \boxplus (1,2)) & (x, 1\boxtimes 1 \boxplus (1,3)) & (x, 1\boxtimes 1 \boxplus (1,4)) & (x, 1\boxtimes 1 \boxplus (1,5)) & (x, 1\boxtimes 1 \boxplus (1,6)) \\ \hline (x, 1\boxtimes 1 \boxplus (2,1)) & (o, 1\boxtimes 1 \boxplus (2,2)) & (o, 1\boxtimes 1 \boxplus (2,3)) & (o, 1\boxtimes 1 \boxplus (2,4)) & (o, 1\boxtimes 1 \boxplus (2,5)) & (x, 1\boxtimes 1 \boxplus (2,6)) \\ \hline (x, 1\boxtimes 1 \boxplus (3,1)) & (o, 1\boxtimes 1 \boxplus (3,2)) & (o, 1\boxtimes 1 \boxplus (3,3)) & (o, 1\boxtimes 1 \boxplus (3,4)) & (o, 1\boxtimes 1 \boxplus (3,5)) & (x, 1\boxtimes 1 \boxplus (3,6)) \\ \hline (x, 1\boxtimes 1 \boxplus (4,1)) & (o, 1\boxtimes 1 \boxplus (4,2)) & (o, 1\boxtimes 1 \boxplus (4,3)) & (o, 1\boxtimes 1 \boxplus (4,4)) & (o, 1\boxtimes 1 \boxplus (4,5)) & (x, 1\boxtimes 1 \boxplus (4,6)) \\ \hline (x, 1\boxtimes 1 \boxplus (5,1)) & (x, 1\boxtimes 1 \boxplus (5,2)) & (x, 1\boxtimes 1 \boxplus (5,3)) & (x, 1\boxtimes 1 \boxplus (5,4)) & (x, 1\boxtimes 1 \boxplus (5,5)) & (x, 1\boxtimes 1 \boxplus (5,6)) \\ \hline \end{array}$$

A full example: iteration #1



All tableau cells corresponding to $1 \times n$ and $n \times 1$ cells are never used by the algorithm, and always remain empty.

A full example: iteration #1, step 1.1



Add elements to cells $T[2 \boxtimes 2 \boxplus (i,j)]$ in the tableau: informally, for each of the 2×2 tiles t in the multipicture which appear in the right-hand size of a rule, add a $(R_i, \omega_i - t, 2 \boxtimes 2 \boxplus (i,j))$ entry.

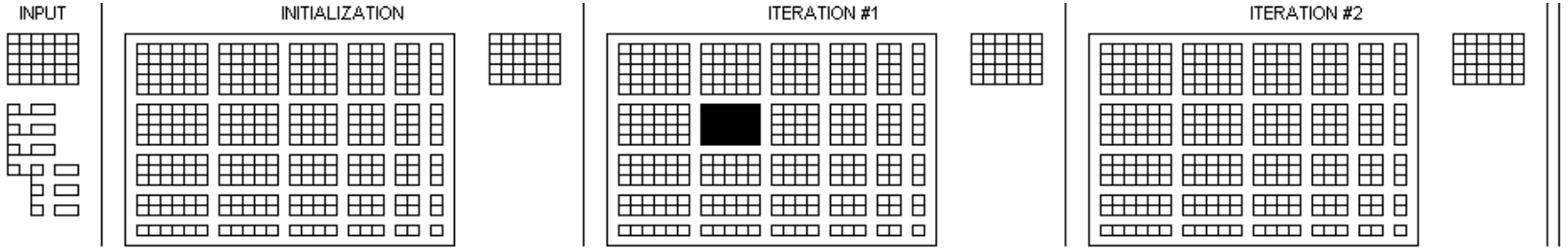
For the first cell, $T[2 \boxtimes 2 \boxplus (1,1)]$:

$$M = \begin{array}{|c|c|c|} \hline (\mathbf{x}, 1\boxtimes 1 \boxplus (\mathbf{1},\mathbf{1})) & (\mathbf{x}, 1\boxtimes 1 \boxplus (\mathbf{1},\mathbf{2})) & \dots \\ \hline (\mathbf{x}, 1\boxtimes 1 \boxplus (\mathbf{2},\mathbf{1})) & (\mathbf{o}, 1\boxtimes 1 \boxplus (\mathbf{2},\mathbf{2})) & \dots \\ \hline \dots & \dots & \dots \\ \hline \end{array}$$

$$\Downarrow$$

$$T_{2,2} = \begin{array}{|c|c|} \hline (\mathbf{R_2}, \{\mathbf{t_{2,2}}, \mathbf{t_{2,3}}, \mathbf{t_{2,4}}, \mathbf{t_{2,5}}, \mathbf{t_{2,6}}\}, 2\boxtimes 2 \boxplus (\mathbf{1},\mathbf{1})) & \dots \\ \hline \dots & \dots \\ \hline \end{array}$$

A full example: iteration #1, step 1.1 (continued)



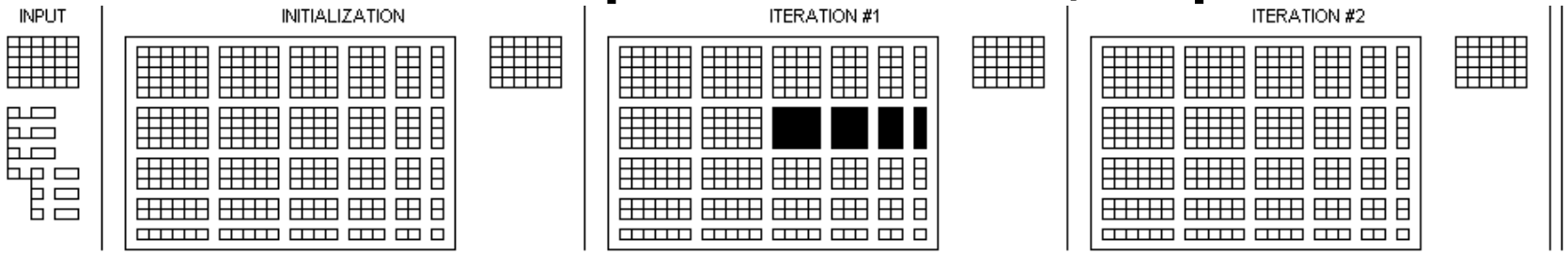
Same for the second cell, $T[2 \boxtimes 2 \boxplus (i,j)]$. Note that the corresponding tile appears in the right-hand side of two rules, these two elements are added to the tableau:

$$M = \begin{array}{|c|c|c|c|} \hline (x, 1\boxtimes 1 \boxplus (1,1)) & (\mathbf{x}, 1\boxtimes 1 \boxplus (\mathbf{1},\mathbf{2})) & (\mathbf{x}, 1\boxtimes 1 \boxplus (\mathbf{1},\mathbf{3})) & \dots \\ \hline (x, 1\boxtimes 1 \boxplus (2,1)) & (\mathbf{o}, 1\boxtimes 1 \boxplus (\mathbf{2},\mathbf{2})) & (\mathbf{o}, 1\boxtimes 1 \boxplus (\mathbf{2},\mathbf{3})) & \dots \\ \hline \dots & \dots & \dots & \\ \hline \end{array}$$

$$\Downarrow$$

$$T_{2,2} = \begin{array}{|c|c|c|} \hline (R_2, \{t_{2,2}, t_{2,3}, t_{2,4}, t_{2,5}, t_{2,6}\}, 2\boxtimes 2 \boxplus (1,1)) & (\mathbf{R_2}, \{\mathbf{t_{2,1}}, \mathbf{t_{2,3}}, \mathbf{t_{2,4}}, \mathbf{t_{2,5}}, \mathbf{t_{2,6}}\}, 2\boxtimes 2 \boxplus (\mathbf{1},\mathbf{2})) & \dots \\ & (\mathbf{R_3}, \{\mathbf{t_{3,1}}, \mathbf{t_{3,3}}, \mathbf{t_{3,4}}, \mathbf{t_{3,5}}, \mathbf{t_{3,6}}\}, 2\boxtimes 2 \boxplus (\mathbf{1},\mathbf{2})) & \dots \\ \hline \dots & \dots & \\ \hline \end{array}$$

A full example: iteration #1, step 1.2



For all the matrices in the tableau from $T_{2,3}$ to $T_{2,6}$, each cell can be filled by "merging" the contents of a couple of cells in the tableau matrix immediately on the left:

$$\left.\begin{array}{l}(R_e, \omega_1, \alpha_1) \in T[r \boxtimes c-1 \boxplus (i,j)] \wedge \\ (R_e, \omega_2, \alpha_2) \in T[r \boxtimes c-1 \boxplus (i,j+1)]\end{array}\right\} \Rightarrow (R_e, \omega_1 \cap \omega_2, \lceil \alpha_1 \cup \alpha_2 \rceil) \in T[r \boxtimes c \boxplus (i,j)]$$

$T_{2,2} =$

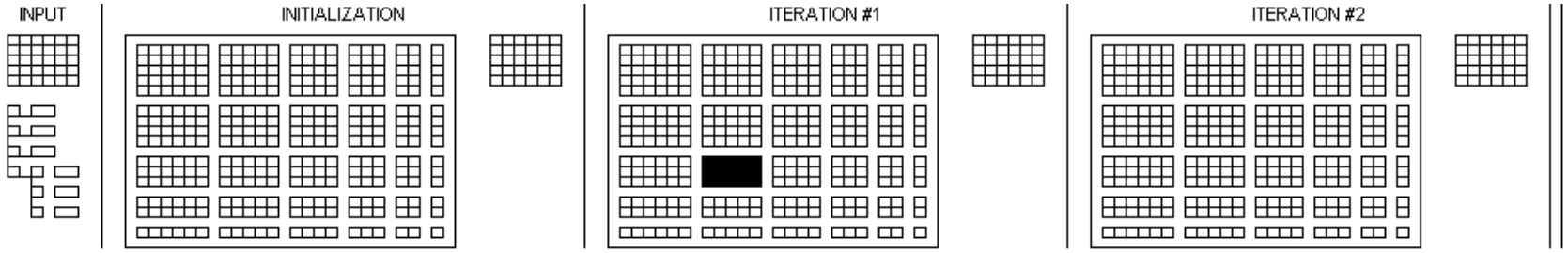
$(\mathbf{R_2}, \{\mathbf{t_{2,2}}, \mathbf{t_{2,3}}, \mathbf{t_{2,4}}, \mathbf{t_{2,5}}, \mathbf{t_{2,6}}\}, \mathbf{2 \boxtimes 2 \boxplus (1,1)})$	$(\mathbf{R_2}, \{\mathbf{t_{2,1}}, \mathbf{t_{2,3}}, \mathbf{t_{2,4}}, \mathbf{t_{2,5}}, \mathbf{t_{2,6}}\}, \mathbf{2 \boxtimes 2 \boxplus (1,2)})$ $(R_3, \{t_{3,1}, t_{3,3}, t_{3,4}, t_{3,5}, t_{3,6}\}, 2 \boxtimes 2 \boxplus (1,2))$	… …
…	…	

$\Downarrow$

$T_{2,3} =$

$(\mathbf{R_2}, \{\mathbf{t_{2,3}}, \mathbf{t_{2,4}}, \mathbf{t_{2,5}}, \mathbf{t_{2,6}}\}, \mathbf{2 \boxtimes 3 \boxplus (1,1)})$	…
…	…

A full example: iteration #1, step 1.2 (continued)



Also cells in matrix $T_{3,2}$ can be derived from cells in $T_{2,2}$:

$$\left.\begin{array}{l}(R_e, \omega_1, \alpha_1) \in T[r-1 \boxtimes c \boxplus (i,j)] \quad \wedge \\ (R_e, \omega_2, \alpha_2) \in T[r-1 \boxtimes c \boxplus (i+1,j)]\end{array}\right\} \Rightarrow (R_e, \omega_1 \cap \omega_2, \lceil \alpha_1 \cup \alpha_2 \rceil) \in T[r \boxtimes c \boxplus (i,j)]$$

$T_{2,2} =$

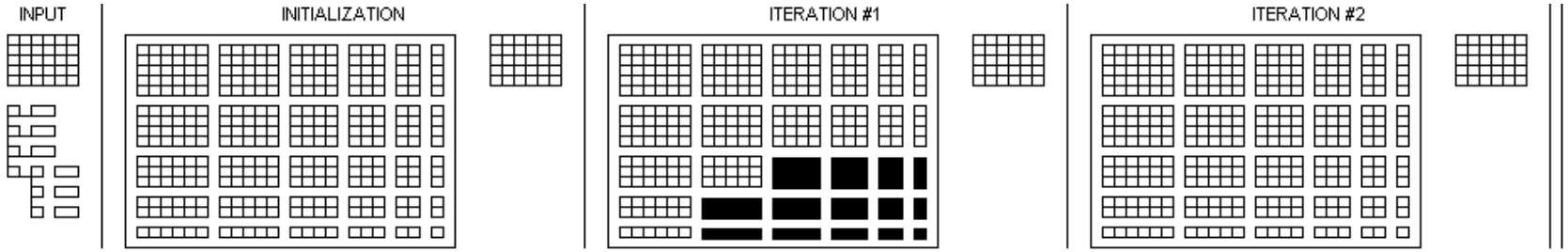
$(\mathbf{R_2}, \{\mathbf{t_{2,2}}, \mathbf{t_{2,3}}, \mathbf{t_{2,4}}, \mathbf{t_{2,5}}, \mathbf{t_{2,6}}\}, \mathbf{2 \boxtimes 2 \boxplus (1,1)})$	$(R_2, \{t_{2,1}, t_{2,3}, t_{2,4}, t_{2,5}, t_{2,6}\}, 2 \boxtimes 2 \boxplus (1,2))$ $(R_3, \{t_{3,1}, t_{3,3}, t_{3,4}, t_{3,5}, t_{3,6}\}, 2 \boxtimes 2 \boxplus (1,2))$	$\ldots$ $\ldots$
$(\mathbf{R_2}, \{\mathbf{t_{2,1}}, \mathbf{t_{2,2}}, \mathbf{t_{2,4}}, \mathbf{t_{2,5}}, \mathbf{t_{2,6}}\}, \mathbf{2 \boxtimes 2 \boxplus (2,1)})$	$\ldots$	
$\ldots$	$\ldots$	

$\Downarrow$

$T_{2,3} =$

$(\mathbf{R_2}, \{\mathbf{t_{2,2}}, \mathbf{t_{2,4}}, \mathbf{t_{2,5}}, \mathbf{t_{2,6}}\}, \mathbf{3 \boxtimes 2 \boxplus (1,1)})$	$\ldots$
$\ldots$	$\ldots$

A full example: iteration #1, step 1.2 (continued)

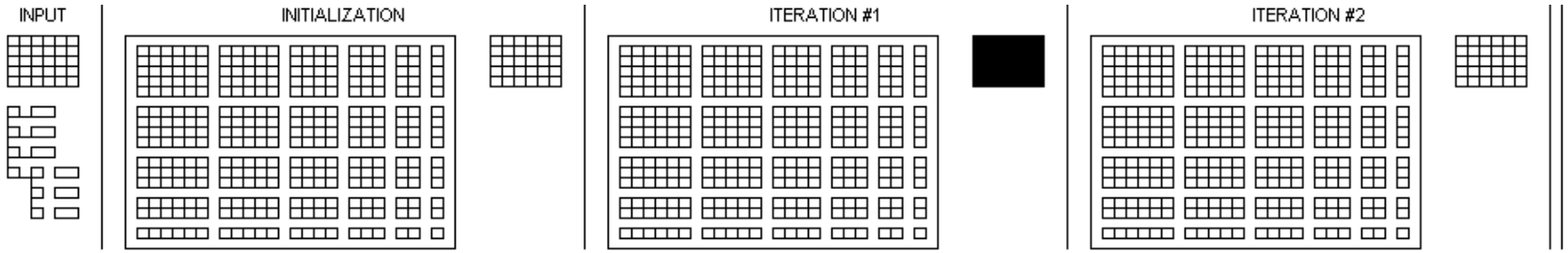


The contents of the remaining cells in the tableau can be derived by applying the same horizontal and vertical merging rules shown in the last two slides:

$$\left.\begin{array}{l}(R_e, \omega_1, \alpha_1) \in T[r \boxtimes c - 1 \boxplus (i,j)] \wedge \\ (R_e, \omega_2, \alpha_2) \in T[r \boxtimes c - 1 \boxplus (i,j+1)]\end{array}\right\} \quad \Rightarrow \quad (R_e, \omega_1 \cap \omega_2, \lceil \alpha_1 \cup \alpha_2 \rceil) \in T[r \boxtimes c \boxplus (i,j)]$$

$$\left.\begin{array}{l}(R_e, \omega_1, \alpha_1) \in T[r - 1 \boxtimes c \boxplus (i,j)] \quad \wedge \\ (R_e, \omega_2, \alpha_2) \in T[r - 1 \boxtimes c \boxplus (i+1,j)]\end{array}\right\} \quad \Rightarrow \quad (R_e, \omega_1 \cap \omega_2, \lceil \alpha_1 \cup \alpha_2 \rceil) \in T[r \boxtimes c \boxplus (i,j)]$$

A full example: iteration #1, step 1.3



$(\mathbf{R2}, \emptyset, \mathbf{5\boxtimes 3\boxplus(1,1)})$	$(R2, \{t_{2,1}, t_{2,3}, t_{2,4},\} 5\boxtimes 3\boxplus(1,2))$ $(R3, \{t_{3,1}, t_{3,3}, t_{3,4}\}, 5\boxtimes 3\boxplus(1,2))$	$(R2, \{t_{2,1}, t_{2,3}, t_{2,4}\}, 5\boxtimes 3\boxplus(1,3))$ $(R3, \{t_{3,1}, t_{3,3}, t_{3,4}\}, 5\boxtimes 3\boxplus(1,3))$	$\mathbf{R3}, \emptyset, \mathbf{5\boxtimes 3\boxplus(1,4)})$

Let's consider the final state of tableau cell $T_{5,3}$: rules $R2$ and R_3 were recognized:

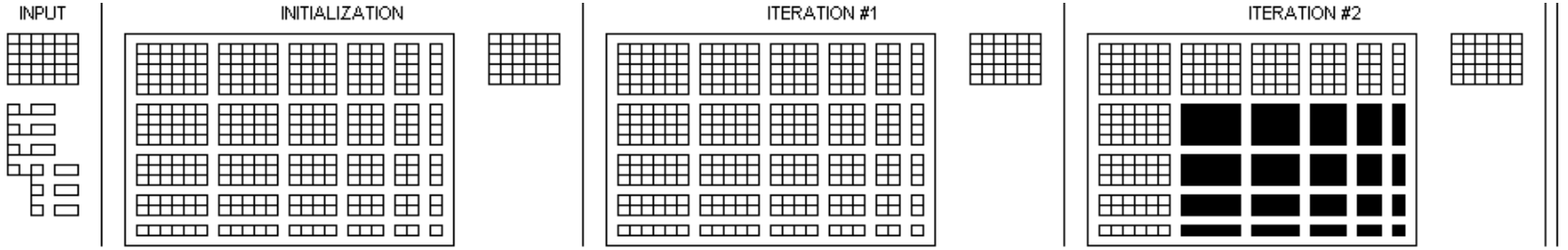
- an entry $(A, 5\boxtimes 3\boxplus(1,1))$ is added to all the pixels in the $5\boxtimes 3\boxplus(1,1)$ subpicture of the multipicture M;
- an entry $(B, 5\boxtimes 3\boxplus(1,4))$ is added to all the pixels in the $5\boxtimes 3\boxplus(1,4)$ subpicture of the multipicture M.

The final state of the multipicture at the end of iteration 1 is shown in the next slide.

$M =$

$(x, 1\boxtimes 1 \boxplus (1,1))$ $(A, 5\boxtimes 3 \boxplus (1,1))$ $(A, 5\boxtimes 4 \boxplus (1,1))$ $(A, 5\boxtimes 5 \boxplus (1,1))$	$(x, 1\boxtimes 1 \boxplus (1,2))$ $(A, 5\boxtimes 3 \boxplus (1,1))$ $(A, 5\boxtimes 4 \boxplus (1,1))$ $(A, 5\boxtimes 5 \boxplus (1,1))$ $(B, 5\boxtimes 5 \boxplus (1,2))$	$(x, 1\boxtimes 1 \boxplus (1,3))$ $(A, 5\boxtimes 3 \boxplus (1,1))$ $(A, 5\boxtimes 4 \boxplus (1,1))$ $(A, 5\boxtimes 5 \boxplus (1,1))$ $(B, 5\boxtimes 4 \boxplus (1,3))$ $(B, 5\boxtimes 5 \boxplus (1,1))$	$(x, 1\boxtimes 1 \boxplus (1,4))$ $(A, 5\boxtimes 4 \boxplus (1,1))$ $(A, 5\boxtimes 5 \boxplus (1,1))$ $(B, 5\boxtimes 3 \boxplus (1,4))$ $(B, 5\boxtimes 4 \boxplus (1,3))$ $(B, 5\boxtimes 5 \boxplus (1,2))$	$(x, 1\boxtimes 1 \boxplus (1,5))$ $(A, 5\boxtimes 5 \boxplus (1,1))$ $(B, 5\boxtimes 3 \boxplus (1,4))$ $(B, 5\boxtimes 4 \boxplus (1,3))$ $(B, 5\boxtimes 5 \boxplus (1,2))$	$(x, 1\boxtimes 1 \boxplus (1,6))$ $(B, 5\boxtimes 3 \boxplus (1,4))$ $(B, 5\boxtimes 4 \boxplus (1,3))$ $(B, 5\boxtimes 5 \boxplus (1,2))$
$(x, 1\boxtimes 1 \boxplus (2,1))$ $(A, 5\boxtimes 3 \boxplus (1,1))$ $(A, 5\boxtimes 4 \boxplus (1,1))$ $(A, 5\boxtimes 5 \boxplus (1,1))$	$(o, 1\boxtimes 1 \boxplus (2,2))$ $(A, 5\boxtimes 3 \boxplus (1,1))$ $(A, 5\boxtimes 4 \boxplus (1,1))$ $(A, 5\boxtimes 5 \boxplus (1,1))$ $(B, 5\boxtimes 5 \boxplus (1,2))$	$(o, 1\boxtimes 1 \boxplus (2,3))$ $(A, 5\boxtimes 3 \boxplus (1,1))$ $(A, 5\boxtimes 4 \boxplus (1,1))$ $(A, 5\boxtimes 5 \boxplus (1,1))$ $(B, 5\boxtimes 4 \boxplus (1,3))$ $(B, 5\boxtimes 5 \boxplus (1,1))$	$(o, 1\boxtimes 1 \boxplus (2,5))$ $(A, 5\boxtimes 4 \boxplus (1,1))$ $(A, 5\boxtimes 5 \boxplus (1,1))$ $(B, 5\boxtimes 3 \boxplus (1,4))$ $(B, 5\boxtimes 4 \boxplus (1,3))$ $(B, 5\boxtimes 5 \boxplus (1,2))$	$(o, 1\boxtimes 1 \boxplus (2,5))$ $(A, 5\boxtimes 5 \boxplus (1,1))$ $(B, 5\boxtimes 3 \boxplus (1,4))$ $(B, 5\boxtimes 4 \boxplus (1,3))$ $(B, 5\boxtimes 5 \boxplus (1,2))$	$(x, 1\boxtimes 1 \boxplus (2,6))$ $(B, 5\boxtimes 3 \boxplus (1,4))$ $(B, 5\boxtimes 4 \boxplus (1,3))$ $(B, 5\boxtimes 5 \boxplus (1,2))$
$(x, 1\boxtimes 1 \boxplus (3,1))$ $(A, 5\boxtimes 3 \boxplus (1,1))$ $(A, 5\boxtimes 4 \boxplus (1,1))$ $(A, 5\boxtimes 5 \boxplus (1,1))$	$(o, 1\boxtimes 1 \boxplus (3,2))$ $(A, 5\boxtimes 3 \boxplus (1,1))$ $(A, 5\boxtimes 4 \boxplus (1,1))$ $(A, 5\boxtimes 5 \boxplus (1,1))$ $(B, 5\boxtimes 5 \boxplus (1,2))$	$(o, 1\boxtimes 1 \boxplus (3,3))$ $(A, 5\boxtimes 3 \boxplus (1,1))$ $(A, 5\boxtimes 4 \boxplus (1,1))$ $(A, 5\boxtimes 5 \boxplus (1,1))$ $(B, 5\boxtimes 4 \boxplus (1,3))$ $(B, 5\boxtimes 5 \boxplus (1,1))$	$(o, 1\boxtimes 1 \boxplus (3,4))$ $(A, 5\boxtimes 4 \boxplus (1,1))$ $(A, 5\boxtimes 5 \boxplus (1,1))$ $(B, 5\boxtimes 3 \boxplus (1,4))$ $(B, 5\boxtimes 4 \boxplus (1,3))$ $(B, 5\boxtimes 5 \boxplus (1,2))$	$(o, 1\boxtimes 1 \boxplus (3,5))$ $(A, 5\boxtimes 5 \boxplus (1,1))$ $(B, 5\boxtimes 3 \boxplus (1,4))$ $(B, 5\boxtimes 4 \boxplus (1,3))$ $(B, 5\boxtimes 5 \boxplus (1,2))$	$(x, 1\boxtimes 1 \boxplus (3,6))$ $(B, 5\boxtimes 3 \boxplus (1,4))$ $(B, 5\boxtimes 4 \boxplus (1,3))$ $(B, 5\boxtimes 5 \boxplus (1,2))$
$(x, 1\boxtimes 1 \boxplus (4,1))$ $(A, 5\boxtimes 3 \boxplus (1,1))$ $(A, 5\boxtimes 4 \boxplus (1,1))$ $(A, 5\boxtimes 5 \boxplus (1,1))$	$(o, 1\boxtimes 1 \boxplus (4,2))$ $(A, 5\boxtimes 3 \boxplus (1,1))$ $(A, 5\boxtimes 4 \boxplus (1,1))$ $(A, 5\boxtimes 5 \boxplus (1,1))$ $(B, 5\boxtimes 5 \boxplus (1,2))$	$(o, 1\boxtimes 1 \boxplus (4,3))$ $(A, 5\boxtimes 3 \boxplus (1,1))$ $(A, 5\boxtimes 4 \boxplus (1,1))$ $(A, 5\boxtimes 5 \boxplus (1,1))$ $(B, 5\boxtimes 4 \boxplus (1,3))$ $(B, 5\boxtimes 5 \boxplus (1,1))$	$(o, 1\boxtimes 1 \boxplus (4,4))$ $(A, 5\boxtimes 4 \boxplus (1,1))$ $(A, 5\boxtimes 5 \boxplus (1,1))$ $(B, 5\boxtimes 3 \boxplus (1,4))$ $(B, 5\boxtimes 4 \boxplus (1,3))$ $(B, 5\boxtimes 5 \boxplus (1,2))$	$(o, 1\boxtimes 1 \boxplus (4,5))$ $(A, 5\boxtimes 5 \boxplus (1,1))$ $(B, 5\boxtimes 3 \boxplus (1,4))$ $(B, 5\boxtimes 4 \boxplus (1,3))$ $(B, 5\boxtimes 5 \boxplus (1,2))$	$(x, 1\boxtimes 1 \boxplus (4,6))$ $(B, 5\boxtimes 3 \boxplus (1,4))$ $(B, 5\boxtimes 4 \boxplus (1,3))$ $(B, 5\boxtimes 5 \boxplus (1,2))$
$(x, 1\boxtimes 1 \boxplus (5,1))$ $(A, 5\boxtimes 3 \boxplus (1,1))$ $(A, 5\boxtimes 4 \boxplus (1,1))$ $(A, 5\boxtimes 5 \boxplus (1,1))$	$(x, 1\boxtimes 1 \boxplus (5,2))$ $(A, 5\boxtimes 3 \boxplus (1,1))$ $(A, 5\boxtimes 4 \boxplus (1,1))$ $(A, 5\boxtimes 5 \boxplus (1,1))$ $(B, 5\boxtimes 5 \boxplus (1,2))$	$(x, 1\boxtimes 1 \boxplus (5,3))$ $(A, 5\boxtimes 3 \boxplus (1,1))$ $(A, 5\boxtimes 4 \boxplus (1,1))$ $(A, 5\boxtimes 5 \boxplus (1,1))$ $(B, 5\boxtimes 4 \boxplus (1,3))$ $(B, 5\boxtimes 5 \boxplus (1,1))$	$(x, 1\boxtimes 1 \boxplus (5,4))$ $(A, 5\boxtimes 4 \boxplus (1,1))$ $(A, 5\boxtimes 5 \boxplus (1,1))$ $(B, 5\boxtimes 3 \boxplus (1,4))$ $(B, 5\boxtimes 4 \boxplus (1,3))$ $(B, 5\boxtimes 5 \boxplus (1,2))$	$(x, 1\boxtimes 1 \boxplus (5,5))$ $(A, 5\boxtimes 5 \boxplus (1,1))$ $(B, 5\boxtimes 3 \boxplus (1,4))$ $(B, 5\boxtimes 4 \boxplus (1,3))$ $(B, 5\boxtimes 5 \boxplus (1,2))$	$(x, 1\boxtimes 1 \boxplus (5,6))$ $(B, 5\boxtimes 3 \boxplus (1,4))$ $(B, 5\boxtimes 4 \boxplus (1,3))$ $(B, 5\boxtimes 5 \boxplus (1,2))$

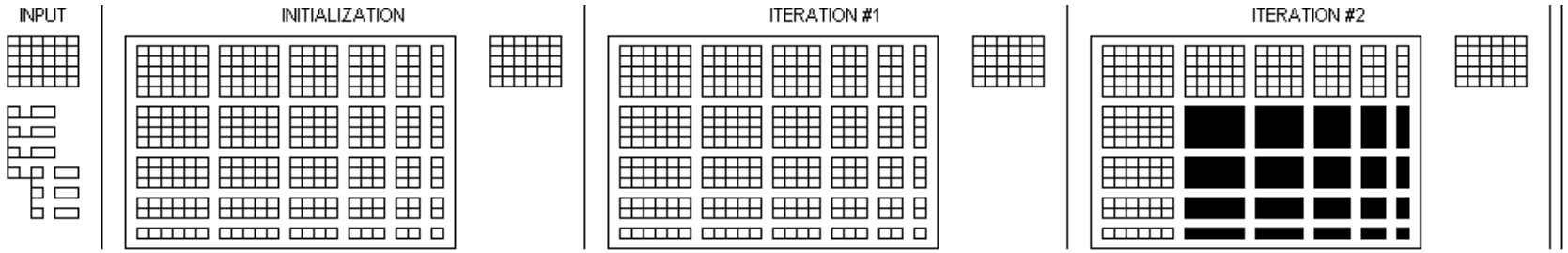
A full example: iteration #2, step 1.1



Step #1 is applied on the new multipicture. Entire subrectangles of A's and B's are recognized by rule R_1.

$$M = \begin{array}{c|c|c|c} \ldots. & \ldots & \ldots & \ldots \\ \hline & \ldots & \ldots & \\ & (A, 5\boxtimes 3 \boxplus (1,1)) & (A, 5\boxtimes 3 \boxplus (1,1)) & \\ \ldots & (A, 5\boxtimes 4 \boxplus (1,1)) & (A, 5\boxtimes 4 \boxplus (1,1)) & \ldots \\ & (A, 5\boxtimes 5 \boxplus (1,1)) & (A, 5\boxtimes 5 \boxplus (1,1)) & \\ & \ldots & \ldots & \\ \hline & \ldots & \ldots & \\ & (A, 5\boxtimes 3 \boxplus (1,1)) & (A, 5\boxtimes 3 \boxplus (1,1)) & \\ \ldots & (A, 5\boxtimes 4 \boxplus (1,1)) & (A, 5\boxtimes 4 \boxplus (1,1)) & \ldots \\ & (A, 5\boxtimes 5 \boxplus (1,1)) & (A, 5\boxtimes 5 \boxplus (1,1)) & \\ & \ldots & \ldots & \\ \hline \ldots & \ldots & \ldots & \ldots \end{array} \Rightarrow T_{2,2} = \begin{array}{c|c|c} \ldots. & \ldots & \ldots \\ \hline & \ldots & \\ & (R_1, \{t_{1,2}, t_{1,3}\}, 5\boxtimes 3 \boxplus (1,1)) & \\ \ldots & (R_1, \{t_{1,2}, t_{1,3}\}, 5\boxtimes 4 \boxplus (1,1)) & \ldots \\ & (R_1, \{t_{1,2}, t_{1,3}\}, 5\boxtimes 5 \boxplus (1,1)) & \\ & \ldots & \ldots \\ \hline \ldots & \ldots & \ldots \end{array}$$

A full example: iteration #2, step 1.2

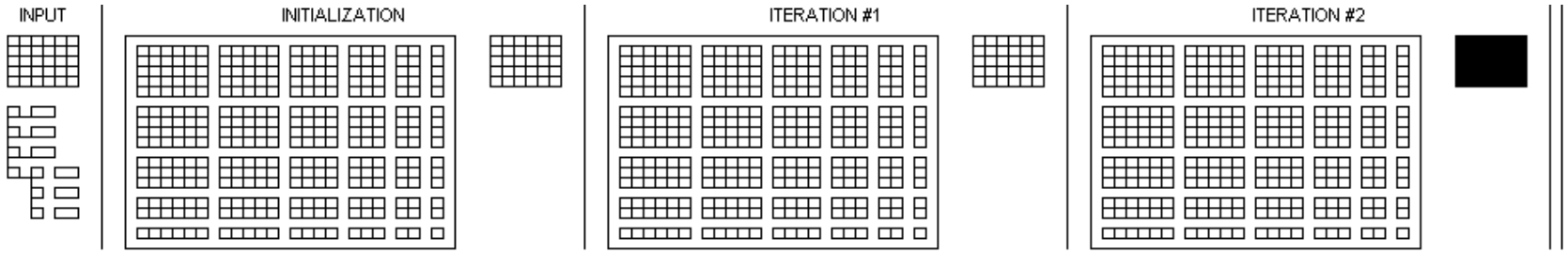


At the end of step 2, cell $T_{5,6}$ contains the following value:

$$T_{2,2} = \boxed{(R_1, \emptyset, 5 \boxtimes 6 \boxplus (1,1))}\,.$$

Thus, an application area for rule R_1 was recognized over the entire multipicture.

A full example: iteration #2, step 1.3



- An element $(S, 5\boxtimes 6 \boxplus (1,1))$ is added to every cell in the multipicture belonging to the rectangle $5\boxtimes 6 \boxplus (1,1)$, which is the whole picture.
- The picture is therefore recognized.
- The final state of the multipicture is shown in the next slide.

$M =$

$(x, 1\boxtimes 1\boxplus(1,1))$ $(A, 5\boxtimes 3\boxplus(1,1))$ $(A, 5\boxtimes 4\boxplus(1,1))$ $(A, 5\boxtimes 5\boxplus(1,1))$ $(S, 5\boxtimes 6\boxplus(1,1))$	$(x, 1\boxtimes 1\boxplus(1,2))$ $(A, 5\boxtimes 3\boxplus(1,1))$ $(A, 5\boxtimes 4\boxplus(1,1))$ $(A, 5\boxtimes 5\boxplus(1,1))$ $(B, 5\boxtimes 5\boxplus(1,2))$ $(S, 5\boxtimes 6\boxplus(1,1))$	$(x, 1\boxtimes 1\boxplus(1,3))$ $(A, 5\boxtimes 3\boxplus(1,1))$ $(A, 5\boxtimes 4\boxplus(1,1))$ $(A, 5\boxtimes 5\boxplus(1,1))$ $(B, 5\boxtimes 4\boxplus(1,3))$ $(B, 5\boxtimes 5\boxplus(1,1))$ $(S, 5\boxtimes 6\boxplus(1,1))$	$(x, 1\boxtimes 1\boxplus(1,4))$ $(A, 5\boxtimes 4\boxplus(1,1))$ $(A, 5\boxtimes 5\boxplus(1,1))$ $(B, 5\boxtimes 3\boxplus(1,4))$ $(B, 5\boxtimes 4\boxplus(1,3))$ $(B, 5\boxtimes 5\boxplus(1,2))$ $(S, 5\boxtimes 6\boxplus(1,1))$	$(x, 1\boxtimes 1\boxplus(1,5))$ $(A, 5\boxtimes 5\boxplus(1,1))$ $(B, 5\boxtimes 3\boxplus(1,4))$ $(B, 5\boxtimes 4\boxplus(1,3))$ $(B, 5\boxtimes 5\boxplus(1,2))$ $(S, 5\boxtimes 6\boxplus(1,1))$	$(x, 1\boxtimes 1\boxplus(1,6))$ $(B, 5\boxtimes 3\boxplus(1,4))$ $(B, 5\boxtimes 4\boxplus(1,3))$ $(B, 5\boxtimes 5\boxplus(1,2))$ $(S, 5\boxtimes 6\boxplus(1,1))$
$(x, 1\boxtimes 1\boxplus(2,1))$ $(A, 5\boxtimes 3\boxplus(1,1))$ $(A, 5\boxtimes 4\boxplus(1,1))$ $(A, 5\boxtimes 5\boxplus(1,1))$ $(S, 5\boxtimes 6\boxplus(1,1))$	$(o, 1\boxtimes 1\boxplus(2,2))$ $(A, 5\boxtimes 3\boxplus(1,1))$ $(A, 5\boxtimes 4\boxplus(1,1))$ $(A, 5\boxtimes 5\boxplus(1,1))$ $(B, 5\boxtimes 5\boxplus(1,2))$ $(S, 5\boxtimes 6\boxplus(1,1))$	$(o, 1\boxtimes 1\boxplus(2,3))$ $(A, 5\boxtimes 3\boxplus(1,1))$ $(A, 5\boxtimes 4\boxplus(1,1))$ $(A, 5\boxtimes 5\boxplus(1,1))$ $(B, 5\boxtimes 4\boxplus(1,3))$ $(B, 5\boxtimes 5\boxplus(1,1))$ $(S, 5\boxtimes 6\boxplus(1,1))$	$(o, 1\boxtimes 1\boxplus(2,5))$ $(A, 5\boxtimes 4\boxplus(1,1))$ $(A, 5\boxtimes 5\boxplus(1,1))$ $(B, 5\boxtimes 3\boxplus(1,4))$ $(B, 5\boxtimes 4\boxplus(1,3))$ $(B, 5\boxtimes 5\boxplus(1,2))$ $(S, 5\boxtimes 6\boxplus(1,1))$	$(o, 1\boxtimes 1\boxplus(2,5))$ $(A, 5\boxtimes 5\boxplus(1,1))$ $(B, 5\boxtimes 3\boxplus(1,4))$ $(B, 5\boxtimes 4\boxplus(1,3))$ $(B, 5\boxtimes 5\boxplus(1,2))$ $(S, 5\boxtimes 6\boxplus(1,1))$	$(x, 1\boxtimes 1\boxplus(2,6))$ $(B, 5\boxtimes 3\boxplus(1,4))$ $(B, 5\boxtimes 4\boxplus(1,3))$ $(B, 5\boxtimes 5\boxplus(1,2))$ $(S, 5\boxtimes 6\boxplus(1,1))$
$(x, 1\boxtimes 1\boxplus(3,1))$ $(A, 5\boxtimes 3\boxplus(1,1))$ $(A, 5\boxtimes 4\boxplus(1,1))$ $(A, 5\boxtimes 5\boxplus(1,1))$ $(S, 5\boxtimes 6\boxplus(1,1))$	$(o, 1\boxtimes 1\boxplus(3,2))$ $(A, 5\boxtimes 3\boxplus(1,1))$ $(A, 5\boxtimes 4\boxplus(1,1))$ $(A, 5\boxtimes 5\boxplus(1,1))$ $(B, 5\boxtimes 5\boxplus(1,2))$ $(S, 5\boxtimes 6\boxplus(1,1))$	$(o, 1\boxtimes 1\boxplus(3,3))$ $(A, 5\boxtimes 3\boxplus(1,1))$ $(A, 5\boxtimes 4\boxplus(1,1))$ $(A, 5\boxtimes 5\boxplus(1,1))$ $(B, 5\boxtimes 4\boxplus(1,3))$ $(B, 5\boxtimes 5\boxplus(1,1))$ $(S, 5\boxtimes 6\boxplus(1,1))$	$(o, 1\boxtimes 1\boxplus(3,4))$ $(A, 5\boxtimes 4\boxplus(1,1))$ $(A, 5\boxtimes 5\boxplus(1,1))$ $(B, 5\boxtimes 3\boxplus(1,4))$ $(B, 5\boxtimes 4\boxplus(1,3))$ $(B, 5\boxtimes 5\boxplus(1,2))$ $(S, 5\boxtimes 6\boxplus(1,1))$	$(o, 1\boxtimes 1\boxplus(3,5))$ $(A, 5\boxtimes 5\boxplus(1,1))$ $(B, 5\boxtimes 3\boxplus(1,4))$ $(B, 5\boxtimes 4\boxplus(1,3))$ $(B, 5\boxtimes 5\boxplus(1,2))$ $(S, 5\boxtimes 6\boxplus(1,1))$	$(x, 1\boxtimes 1\boxplus(3,6))$ $(B, 5\boxtimes 3\boxplus(1,4))$ $(B, 5\boxtimes 4\boxplus(1,3))$ $(B, 5\boxtimes 5\boxplus(1,2))$ $(S, 5\boxtimes 6\boxplus(1,1))$
$(x, 1\boxtimes 1\boxplus(4,1))$ $(A, 5\boxtimes 3\boxplus(1,1))$ $(A, 5\boxtimes 4\boxplus(1,1))$ $(A, 5\boxtimes 5\boxplus(1,1))$ $(S, 5\boxtimes 6\boxplus(1,1))$	$(o, 1\boxtimes 1\boxplus(4,2))$ $(A, 5\boxtimes 3\boxplus(1,1))$ $(A, 5\boxtimes 4\boxplus(1,1))$ $(A, 5\boxtimes 5\boxplus(1,1))$ $(B, 5\boxtimes 5\boxplus(1,2))$ $(S, 5\boxtimes 6\boxplus(1,1))$	$(o, 1\boxtimes 1\boxplus(4,3))$ $(A, 5\boxtimes 3\boxplus(1,1))$ $(A, 5\boxtimes 4\boxplus(1,1))$ $(A, 5\boxtimes 5\boxplus(1,1))$ $(B, 5\boxtimes 4\boxplus(1,3))$ $(B, 5\boxtimes 5\boxplus(1,1))$ $(S, 5\boxtimes 6\boxplus(1,1))$	$(o, 1\boxtimes 1\boxplus(4,4))$ $(A, 5\boxtimes 4\boxplus(1,1))$ $(A, 5\boxtimes 5\boxplus(1,1))$ $(B, 5\boxtimes 3\boxplus(1,4))$ $(B, 5\boxtimes 4\boxplus(1,3))$ $(B, 5\boxtimes 5\boxplus(1,2))$ $(S, 5\boxtimes 6\boxplus(1,1))$	$(o, 1\boxtimes 1\boxplus(4,5))$ $(A, 5\boxtimes 5\boxplus(1,1))$ $(B, 5\boxtimes 3\boxplus(1,4))$ $(B, 5\boxtimes 4\boxplus(1,3))$ $(B, 5\boxtimes 5\boxplus(1,2))$ $(S, 5\boxtimes 6\boxplus(1,1))$	$(x, 1\boxtimes 1\boxplus(4,6))$ $(B, 5\boxtimes 3\boxplus(1,4))$ $(B, 5\boxtimes 4\boxplus(1,3))$ $(B, 5\boxtimes 5\boxplus(1,2))$ $(S, 5\boxtimes 6\boxplus(1,1))$
$(x, 1\boxtimes 1\boxplus(5,1))$ $(A, 5\boxtimes 3\boxplus(1,1))$ $(A, 5\boxtimes 4\boxplus(1,1))$ $(A, 5\boxtimes 5\boxplus(1,1))$ $(S, 5\boxtimes 6\boxplus(1,1))$	$(x, 1\boxtimes 1\boxplus(5,2))$ $(A, 5\boxtimes 3\boxplus(1,1))$ $(A, 5\boxtimes 4\boxplus(1,1))$ $(A, 5\boxtimes 5\boxplus(1,1))$ $(B, 5\boxtimes 5\boxplus(1,2))$ $(S, 5\boxtimes 6\boxplus(1,1))$	$(x, 1\boxtimes 1\boxplus(5,3))$ $(A, 5\boxtimes 3\boxplus(1,1))$ $(A, 5\boxtimes 4\boxplus(1,1))$ $(A, 5\boxtimes 5\boxplus(1,1))$ $(B, 5\boxtimes 4\boxplus(1,3))$ $(B, 5\boxtimes 5\boxplus(1,1))$ $(S, 5\boxtimes 6\boxplus(1,1))$	$(x, 1\boxtimes 1\boxplus(5,4))$ $(A, 5\boxtimes 4\boxplus(1,1))$ $(A, 5\boxtimes 5\boxplus(1,1))$ $(B, 5\boxtimes 3\boxplus(1,4))$ $(B, 5\boxtimes 4\boxplus(1,3))$ $(B, 5\boxtimes 5\boxplus(1,2))$ $(S, 5\boxtimes 6\boxplus(1,1))$	$(x, 1\boxtimes 1\boxplus(5,5))$ $(A, 5\boxtimes 5\boxplus(1,1))$ $(B, 5\boxtimes 3\boxplus(1,4))$ $(B, 5\boxtimes 4\boxplus(1,3))$ $(B, 5\boxtimes 5\boxplus(1,2))$ $(S, 5\boxtimes 6\boxplus(1,1))$	$(x, 1\boxtimes 1\boxplus(5,6))$ $(B, 5\boxtimes 3\boxplus(1,4))$ $(B, 5\boxtimes 4\boxplus(1,3))$ $(B, 5\boxtimes 5\boxplus(1,2))$ $(S, 5\boxtimes 6\boxplus(1,1))$

In this presentation:

1) What are pictures languages

2) What are Tile-Rewriting Grammars

3) We have a polynomial parsing technique for Tile-Rewriting Grammars!