Two Dimensional Picture Languages: Tiling Systems, Domino Systems and Weighted Finite Automata

Benedikt Lüken-Winkels

September 2020

Introduction

This report provides an overview and a summary of the works of Gimmarresi and Restivo [1] and Culik and Kari [2] on two-dimensional languages, especially focussing on Tiling Systems, Domino Systems and Weighted Finite Automata.

1 Tiling Systems

The basic idea of a tiling system is to use the ability of finite automata to recognize a string language by projecting a local language in the two-dimensional world. Therefore the local language can be obtained from a finite set Θ of square *tiles* of size 2×2. Then a two-dimensional language is 'tiling recognizable', if it can be projected from the local picture language.

A tiling system $\mathcal{T}$ is defined as a 4-tuple $\mathcal{T} = \{\Sigma, \Gamma, \Theta, \pi\}$. Let Σ and Γ be finite alphabets. A language $L \subseteq \Sigma^{**}$ is tiling recognizable, if there is a local language $L' \subseteq \Gamma^{**}$ that can be obtained exactly from tiles taken from the finite set of tiles Θ over the alphabet $\Gamma \cup \{\#\}$ and a projection $\pi : \Theta \to \Gamma$. When L' can be obtained from Θ it can be written as $L' = L(\Theta)$. A computational approach on verifying the locality of L' is to scan a picture from L' with a 2×2 window and checking, if every tile is included in Θ.

The recognition of a Language L by a tiling system $\mathcal{T}$ can be written as $L = L(\mathcal{T})$, when L is a projection of some local language. and $L \in \mathcal{L}(TS)$ (L lies in the family of two-dimensional languages recognizable by tiling systems).

Example 1 Let $\Sigma = \{a\}$ be a one-letter alphabet and let L be the language of all pictures over Σ with 3 rows.
Claim: Language L is tiling recognizable.

$$\textit{Let } \Theta = \left\{ \begin{array}{c} \boxed{\begin{array}{cc} \# & \# \\ \# & 1 \end{array}}, \boxed{\begin{array}{cc} \# & \# \\ 1 & 1 \end{array}}, \boxed{\begin{array}{cc} \# & \# \\ 1 & \# \end{array}} \\ \boxed{\begin{array}{cc} \# & 1 \\ \# & 2 \end{array}}, \boxed{\begin{array}{cc} 1 & 1 \\ 2 & 2 \end{array}}, \boxed{\begin{array}{cc} 1 & \# \\ 2 & \# \end{array}} \\ \boxed{\begin{array}{cc} \# & 2 \\ \# & 3 \end{array}}, \boxed{\begin{array}{cc} 2 & 2 \\ 3 & 3 \end{array}}, \boxed{\begin{array}{cc} 2 & \# \\ 3 & \# \end{array}} \\ \boxed{\begin{array}{cc} \# & 3 \\ \# & \# \end{array}}, \boxed{\begin{array}{cc} 3 & 3 \\ \# & \# \end{array}}, \boxed{\begin{array}{cc} 3 & \# \\ \# & \# \end{array}} \end{array} \right\}$$

Then a picture $p \in L(\Theta)$ can look like this:

$$\boxed{\begin{array}{ccccccc} \# & \# & \# & \# & \# & \# & \# \\ \# & 1 & 1 & 1 & 1 & 1 & \# \\ \# & 2 & 2 & 2 & 2 & 2 & \# \\ \# & 3 & 3 & 3 & 3 & 3 & \# \\ \# & \# & \# & \# & \# & \# & \# \end{array}}$$

Using the previously explained method a 2×2 window traversing p will always contain a tile from Θ. Now with π being defined as $\pi(1) = \pi(2) = \pi(3) = a$, one can see, that L is *tiling recognizable*, so $L \in \mathcal{L}(TS)$. $\square$

This approach works for languages with any number of rows, with a corresponding size of Γ, since for each row there has to be a dedicated symbol in the alphabet to keep track of the number of rows in each picture from $L' = L(\Theta)$.

1.1 Closure Properties

Projection Let Σ_1 and Σ_2 be finite alphabets and $\rho : \Sigma_1 \to \Sigma_2$ a projection. If $L_1 \subseteq \Sigma_1^{**}$ is tiling recognizable, then $L_2 = \rho(L_1)$ ($L_2 \subseteq \Sigma_2^{**}$) is too. Let $\mathcal{T}_1 = (\Sigma_1, \Gamma, \Theta, \pi_1)$ be recognizing tiling system of L_1 and $\mathcal{T}_2 = (\Sigma_2, \Gamma, \Theta, \pi_2)$. Now if π_2 is defined as $\pi_2 = \rho \circ \pi_1 : \Gamma \to \Sigma_2$, one can see, that L_2 is recognized by $\mathcal{T}_2$. Therefore $L_1, L_2 \in \mathcal{L}(TS)$.

$\Rightarrow \mathcal{L}(TS)$ is closed under projection.

Row and column concatenation Let L_1 and L_2 be picture languages over an alphabet Σ and let $L = L_1 \ominus L_2$ be the language corresponding to the row concatenation of L_1 and L_2. Furthermore let $\mathcal{T}_1 = (\Sigma, \Gamma_1, \Theta_1, \pi_1)$ and $\mathcal{T}_2 = (\Sigma, \Gamma_2, \Theta_2, \pi_2)$ be tiling systems for L_1 and L_2. In a tiling system $\mathcal{T} = (\Sigma, \Gamma, \Theta, \pi)$ for L with $\Gamma = \Gamma_1 \cup \Gamma_2$, Θ has to contain

each tile from Θ_1 without its bottom borders

$$\Theta_1' = \left\{ \boxed{\begin{array}{cc} a_1 & b_1 \\ c_1 & d_1 \end{array}} \mid \boxed{\begin{array}{cc} a_1 & b_1 \\ c_1 & d_1 \end{array}} \in \Theta_1 \textit{ and } c_1, d_1 \neq \# \right\},$$

each tile from Θ_2 without its upper borders

$$\Theta_2' = \left\{ \boxed{\begin{array}{cc} a_1 & b_1 \\ c_1 & d_1 \end{array}} \mid \boxed{\begin{array}{cc} a_1 & b_1 \\ c_1 & d_1 \end{array}} \in \Theta_2 \ and \ a_1, b_1 \neq \# \right\} \text{ and}$$

connecting (gluing) tiles

$$\Theta_{12} = \left\{ \boxed{\begin{array}{cc} \# & a_1 \\ \# & a_2 \end{array}}, \boxed{\begin{array}{cc} b_1 & \# \\ b_2 & \# \end{array}}, \boxed{\begin{array}{cc} c_1 & d_1 \\ c_2 & d_2 \end{array}} \mid \begin{array}{l} \boxed{\begin{array}{cc} \# & a_1 \\ \# & \# \end{array}}, \boxed{\begin{array}{cc} b_1 & \# \\ \# & \# \end{array}}, \boxed{\begin{array}{cc} c_1 & d_1 \\ \# & \# \end{array}} \in \Theta_1 \\ \boxed{\begin{array}{cc} \# & \# \\ \# & a_2 \end{array}}, \boxed{\begin{array}{cc} \# & \# \\ b_2 & \# \end{array}}, \boxed{\begin{array}{cc} \# & \# \\ c_2 & d_2 \end{array}} \in \Theta_2 \end{array} \right\}.$$

Then $\Theta = \Theta_1 \cup \Theta_2 \cup \Theta_{12}$ and $\pi : \Gamma \to \Sigma$ maps the tile's symbols corresponding to their 'origins':

$$\forall a \in \Gamma, \qquad \pi(a) = \begin{cases} d \ \pi_1(a) & \text{if } a \in \Gamma_1 \\ \ \ \ \pi_2(a) & \text{if } a \in \Gamma_2 \backslash \{\Gamma_1 \cap \Gamma_2\} \end{cases}$$

For the *column concatenation* $L = L_1 \oplus L_2$ there is a similar approach, but in this case with Θ_1 including all but the right and Θ_2 all but the left bordered tiles. The 'gluing' set of tiles Θ_{12} corresponds to the columns where the pictures are concatenated.

$\Rightarrow \mathcal{L}(TS)$ is closed row and column concatenation.

Row and column closure For a tiling system $(\Sigma, \Gamma, \Theta, \pi)$ for $L^{*\ominus}$, two different tiling systems can be used to build Γ as the union of sets of tiles without upper and lower borders and the set of gluing tiles, as shown above. With the same technique a tiling system for $L^{*\oplus}$ can be defined.

$\Rightarrow \mathcal{L}(TS)$ is closed row and column closure operations.

Union and intersection Let L_1 and L_2 be picture languages over an alphabet Σ and let $L = L_1 \cup L_2$ be the language corresponding to the union of L_1 and L_2. Furthermore let $(\Sigma, \Gamma_1, \Theta_1, \pi_1)$ and $(\Sigma, \Gamma_2, \Theta_2, \pi_2)$ be tiling systems for L_1 and L_2, respectively. A tiling system $(\Sigma, \Gamma, \Theta, \pi)$ for L has $\Gamma = \Gamma_1 \cup \Gamma_2$ with $\Theta = \Theta_1 \cup \Theta_2$. The projection π is defined corresponding to π_1 and π_2 as above.

$(\Sigma, \Gamma, \Theta, \pi)$ for the language $L = L_1 \cap L_2$ uses as local alphabet Γ a subset of $\Gamma_1 \times \Gamma_2$ to identify tiles that occur in both Θ_1 and Θ_2. Two symbols that map to the same symbol in Σ create a pair, that belongs to Γ:

$$(a_1, a_2) \in \Gamma \Leftrightarrow \pi_1(a_1) = \pi_2(a_2)$$

Now Θ consists of tiles, where these pairs of symbols correspond to their origins:

$$\Theta = \left\{ \boxed{\begin{array}{cc} (a_1, a_2) & (b_1, b_2) \\ (c_1, c_2) & (d_1, d_2) \end{array}} \mid \boxed{\begin{array}{cc} a_1 & b_1 \\ c_1 & d_1 \end{array}} \in \Theta_1 \ and \ \boxed{\begin{array}{cc} a_2 & b_2 \\ c_2 & d_2 \end{array}} \in \Theta_2 \right\},$$

$\Rightarrow \mathcal{L}(TS)$ is closed under union and intersection.

Complement Let $\Sigma = \{a, b\}$ and let $L = \{p \in \Sigma^{**} | p = s \ominus s$ wheres is a square$\}$ be the language of rectangles with identical upper and lower halfs. If $L \in \mathcal{L}(TS)$, then there is a local L' over an alphabet Γ from which L can be obtained by projection. In general $|\Sigma| \leq |\Gamma|$, as seen in previous examples.

$\Rightarrow \mathcal{L}(TS)$ is not closed under complement.

Rotation To obtain a tiling system for the rotation L^R of a language L with tiling system $(\Sigma, \Gamma, \Theta, \pi)$, the tiles in Θ are rotated. $(\Sigma, \Gamma, \Theta^R, \pi)$ recognizes L^R.

$\Rightarrow \mathcal{L}(TS)$ is closed under rotation.

2 Domino Systems

3 Weighted Finite (Picture) Automata

References

[1] Dora Giammarresi and Antonio Restivo. Two-dimensional languages. In Grzegorz Rozenberg and Arto Salomaa, editors, *Handbook of Formal Languages, Volume 3: Beyond Words*, pages 215–267. Springer, 1997.

[2] Karel Culik II and Jarkko Kari. Image compression using weighted finite automata. In Andrzej M. Borzyszkowski and Stefan Sokolowski, editors, *Mathematical Foundations of Computer Science 1993, 18th International Symposium, MFCS'93, Gdansk, Poland, August 30 - September 3, 1993, Proceedings*, volume 711 of *Lecture Notes in Computer Science*, pages 392–402. Springer, 1993.