Topics in Logic and Automata Theory

Logic and Automata over Graphs

Abdullah Abdul Khadir

Chennai Mathematical Institute
abdullah@cmi.ac.in

May 14, 2010

Notations and Symbols

Henceforth we assume the following :-

- σ is the vocabulary σ=(R_1,...,R_m,c_1,...,c_s) where, $\forall i \in \{1, .., m\}, R_i$ is a relation symbol of arity k_i, for some $k_i \in \mathbb{N}$ and $\forall i \in \{1, .., s\}, c_i$ is a unique constant symbol.

Notations and Symbols

Henceforth we assume the following :-

- σ is the vocabulary σ=(R_1,...,R_m,c_1,...,c_s) where, $\forall i \in \{1, .., m\}, R_i$ is a relation symbol of arity k_i, for some $k_i \in \mathbb{N}$ and $\forall i \in \{1, .., s\}, c_i$ is a unique constant symbol.
- $\mathcal{A}$=(A,R_1^A,...,R_m^A,c_1^A,...,c_s^A) and $\mathcal{B}$=(B,R_1^B,...,R_m^B,c_1^B,...,c_s^B) are two structures interpreting σ over the domains A and B respectively.

Notations and Symbols

Henceforth we assume the following :-

- σ is the vocabulary σ=(R_1,...,R_m,c_1,...,c_s) where, $\forall i \in \{1, .., m\}, R_i$ is a relation symbol of arity k_i, for some $k_i \in \mathbb{N}$ and $\forall i \in \{1, .., s\}, c_i$ is a unique constant symbol.
- $\mathcal{A}$=(A,R_1^A,...,R_m^A,c_1^A,...,c_s^A) and $\mathcal{B}$=(B,R_1^B,...,R_m^B,c_1^B,...,c_s^B) are two structures interpreting σ over the domains A and B respectively.
- $\mathcal{G}_A$ and $\mathcal{G}_B$ are the gaiffman graphs (explained in the next slide) for $\mathcal{A}$ and $\mathcal{B}$ respectively.

Notations and Symbols

Henceforth we assume the following :-

- σ is the vocabulary σ=(R_1,...,R_m,c_1,...,c_s) where, $\forall i \in \{1, .., m\}, R_i$ is a relation symbol of arity k_i, for some $k_i \in \mathbb{N}$ and $\forall i \in \{1, .., s\}, c_i$ is a unique constant symbol.
- $\mathcal{A}$=(A,R_1^A,...,R_m^A,c_1^A,...,c_s^A) and $\mathcal{B}$=(B,R_1^B,...,R_m^B,c_1^B,...,c_s^B) are two structures interpreting σ over the domains A and B respectively.
- $\mathcal{G}_A$ and $\mathcal{G}_B$ are the gaiffman graphs (explained in the next slide) for $\mathcal{A}$ and $\mathcal{B}$ respectively.
- Given an element a $\in A$, N($\mathcal{A}$,a)$\upharpoonright$d is the neighbourhood or sphere or subgraph of the Gaiffman graph of $\mathcal{A}$, $\mathcal{G}_A$, with a as center and a radius of d.

Gaiffman Graph

Given a structure $\mathcal{A}$=(A,R_1^A,...,R_m^A,c_1^A,...,c_s^A), the Gaiffman graph is the undirected graph $\mathcal{G}_A = (A$,E) where

Gaiffman Graph

Given a structure $\mathcal{A}$=(A,R_1^A,...,R_m^A,c_1^A,...,c_s^A), the Gaiffman graph is the undirected graph $\mathcal{G}_A = (A, \mathrm{E})$ where

- A is the domain of the structure $\mathcal{A}$.

Gaiffman Graph

Given a structure $\mathcal{A}$=(A,R_1^A,...,R_m^A,c_1^A,...,c_s^A), the Gaiffman graph is the undirected graph $\mathcal{G}_A = (A,\mathrm{E})$ where

- A is the domain of the structure $\mathcal{A}$.
- E is a binary relation on A such that for any two elements a,b $\in A$, E(a,b) holds iff $\exists$ a relation R_i^A of arity k_i and k_i variables $\{x_1, ..., x_{k_i}\} \in A$ such that $\mathrm{R}(x_1, ..., x_{k_i})$ holds. Moreover a and b are two of the k_i variables.

Gaiffman Graph

Given a structure $\mathcal{A}$=(A,R_1^A,...,R_m^A,c_1^A,...,c_s^A), the Gaiffman graph is the undirected graph $\mathcal{G}_A$ = (A,E) where

- A is the domain of the structure $\mathcal{A}$.
- E is a binary relation on A such that for any two elements a,b $\in A$, E(a,b) holds iff $\exists$ a relation R_i^A of arity k_i and k_i variables $\{x_1, ..., x_{k_i}\} \in A$ such that R$(x_1, ..., x_{k_i})$ holds. Moreover a and b are two of the k_i variables.

Example : $\mathcal{A} = (\{1,2,3,4\},\leq)$

Gaiffman Graph

Given a structure $\mathcal{A}$=(A,R_1^A,...,R_m^A,c_1^A,...,c_s^A), the Gaiffman graph is the undirected graph $\mathcal{G}_A$ = (A,E) where

- A is the domain of the structure $\mathcal{A}$.
- E is a binary relation on A such that for any two elements a,b $\in A$, E(a,b) holds iff $\exists$ a relation R_i^A of arity k_i and k_i variables $\{x_1, ..., x_{k_i}\} \in A$ such that $R(x_1, ..., x_{k_i})$ holds. Moreover a and b are two of the k_i variables.

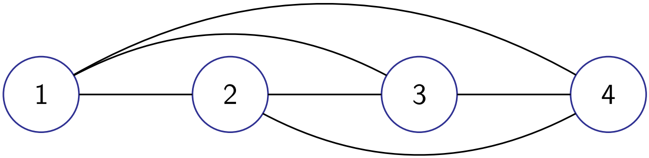
Example : $\mathcal{A} = (\{1,2,3,4\},\leq)$

Gaiffman Graph

Given a structure $\mathcal{A}$=(A,R_1^A,...,R_m^A,c_1^A,...,c_s^A), the Gaiffman graph is the undirected graph $\mathcal{G}_A$ = (A,E) where

- A is the domain of the structure $\mathcal{A}$.
- E is a binary relation on A such that for any two elements a,b $\in A$, E(a,b) holds iff $\exists$ a relation R_i^A of arity k_i and k_i variables $\{x_1, ..., x_{k_i}\} \in A$ such that R($x_1, ..., x_{k_i}$) holds. Moreover a and b are two of the k_i variables.

Example : $\mathcal{A}$ = ({1,2,3,4},$\leq$)



Local Equivalence

Some points and terms to note related to graphs :

- For any d $\in \mathbb{N}$, the number of spheres of radius d is finite.
- Let n $\in \mathbb{N}$, be the number of spheres for a fixed radius d.
- Then we can talk of a type signature of a graph given by ($\#\text{Type}_1$,...,$\#\text{Type}_n$) which is the number of spheres of radius d that are of Type_1,Type_2 ... Type_n.

Local Equivalence

Some points and terms to note related to graphs :

- For any d $\in \mathbb{N}$, the number of spheres of radius d is finite.
- Let n $\in \mathbb{N}$, be the number of spheres for a fixed radius d.
- Then we can talk of a type signature of a graph given by ($\#\text{Type}_1$,...,$\#\text{Type}_n$) which is the number of spheres of radius d that are of Type_1,Type_2 ... Type_n.

Definition (Local d-Equivalence)

Two structures $\mathcal{A}$ and $\mathcal{B}$ are said to be locally d-equivalent for some d $\in \mathbb{N}$, iff both $\mathcal{A}$ and $\mathcal{B}$ have the same type signature of radius d. Let it be denoted by $\mathcal{A} \sim_d \mathcal{B}$.

Logical Equivalence

Definition (Logical r-Equivalence)

Two structures $\mathcal{A}$ and $\mathcal{B}$ are said to be logically r-equivalent for some r $\in \mathbb{N}$, iff they satisfy the same first order formulae of quantifier depth r. Let it be denoted by $\mathcal{A} \equiv_r \mathcal{B}$.

Logical Equivalence

Definition (Logical r-Equivalence)

Two structures $\mathcal{A}$ and $\mathcal{B}$ are said to be logically r-equivalent for some r $\in \mathbb{N}$, iff they satisfy the same first order formulae of quantifier depth r. Let it be denoted by $\mathcal{A} \equiv_r \mathcal{B}$.

We recall that :

- If $\mathcal{A} \equiv_r \mathcal{B}$ then the Duplicator has a winning strategy for the r-round EF-game.

Logical Equivalence

Definition (Logical r-Equivalence)

Two structures $\mathcal{A}$ and $\mathcal{B}$ are said to be logically r-equivalent for some r $\in \mathbb{N}$, iff they satisfy the same first order formulae of quantifier depth r. Let it be denoted by $\mathcal{A} \equiv_r \mathcal{B}$.

We recall that :

- If $\mathcal{A} \equiv_r \mathcal{B}$ then the Duplicator has a winning strategy for the r-round EF-game.
- The above statement holds in both directions namely, $\mathcal{A} \equiv_r \mathcal{B} \Longleftrightarrow$ Duplicator has a winning strategy for the r-round EF game.

Hanf's theorem

Theorem (Hanf's)

Let $d,r \in \mathbb{N}$ *such that* $d \geq 3^{r-1}$.
Then, $\mathcal{A} \sim_d \mathcal{B} \Longrightarrow \mathcal{A} \equiv_r \mathcal{B}$.

Hanf's theorem

Theorem (Hanf's)

Let $d, r \in \mathbb{N}$ *such that* $d \geq 3^{r-1}$.
Then, $\mathcal{A} \sim_d \mathcal{B} \implies \mathcal{A} \equiv_r \mathcal{B}$.

Essentially, what Hanf's theorem states is that for sufficiently large radius d, local equivalence is the same as logical equivalence.

Hanf's theorem

Theorem (Hanf's)

Let d,r $\in \mathbb{N}$ *such that* $d \geq 3^{r-1}$.
Then, $\mathcal{A} \sim_d \mathcal{B} \Longrightarrow \mathcal{A} \equiv_r \mathcal{B}$.

Essentially, what Hanf's theorem states is that for sufficiently large radius d, local equivalence is the same as logical equivalence.

Proof sketch :-

Hanf's theorem

Theorem (Hanf's)

Let d,r $\in \mathbb{N}$ *such that* $d \geq 3^{r-1}$.
Then, $\mathcal{A} \sim_d \mathcal{B} \Longrightarrow \mathcal{A} \equiv_r \mathcal{B}$.

Essentially, what Hanf's theorem states is that for sufficiently large radius d, local equivalence is the same as logical equivalence.

Proof sketch :-

- The duplicator's strategy in Round 1

Hanf's theorem

Theorem (Hanf's)

Let $d, r \in \mathbb{N}$ such that $d \geq 3^{r-1}$.
Then, $\mathcal{A} \sim_d \mathcal{B} \Longrightarrow \mathcal{A} \equiv_r \mathcal{B}$.

Essentially, what Hanf's theorem states is that for sufficiently large radius d, local equivalence is the same as logical equivalence.

Proof sketch :-

- The duplicator's strategy in Round 1
- The duplicator's strategy in Round i

Hanf's theorem

Theorem (Hanf's)

Let $d, r \in \mathbb{N}$ *such that* $d \geq 3^{r-1}$.
Then, $\mathcal{A} \sim_d \mathcal{B} \Longrightarrow \mathcal{A} \equiv_r \mathcal{B}$.

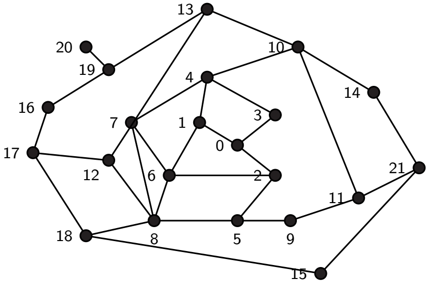
Essentially, what Hanf's theorem states is that for sufficiently large radius d, local equivalence is the same as logical equivalence.

Proof sketch :-

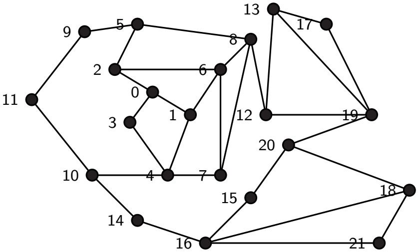
- The duplicator's strategy in Round 1
- The duplicator's strategy in Round i
- Variations of hanf's theorem

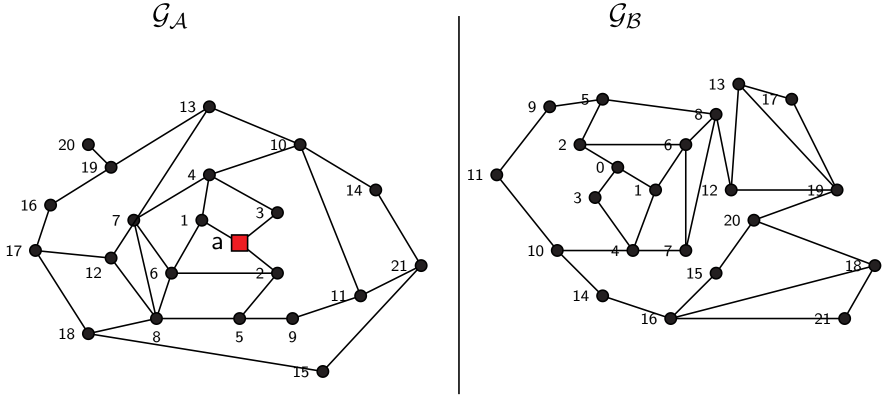
The EF-Game Graphs

$\mathcal{G}_{\mathcal{A}}$



$\mathcal{G}_{\mathcal{B}}$

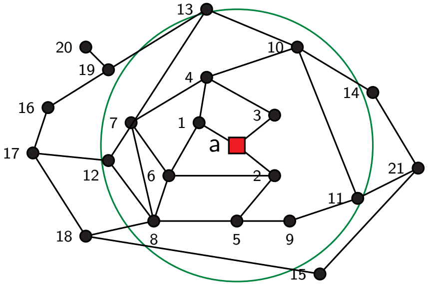




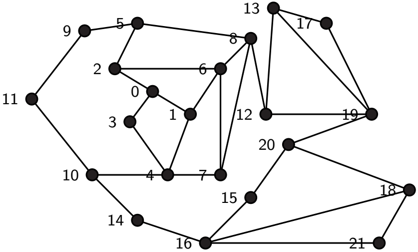
- The Spoiler chooses a vertex from any graph (here, $\mathcal{A}$)

Round 1

$\mathcal{G}_{\mathcal{A}}$



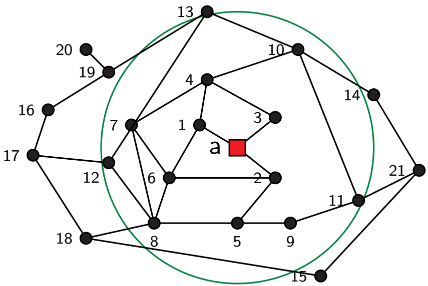
$\mathcal{G}_{\mathcal{B}}$



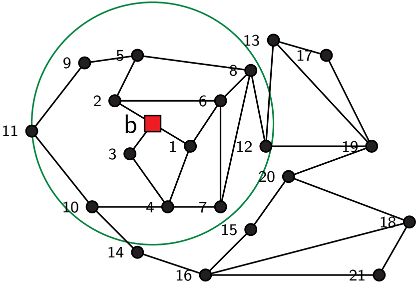
- The Spoiler chooses a vertex from any graph (here, $\mathcal{A}$)
- The d-neighbourhood of a in $\mathcal{A}$, denoted N($\mathcal{A}$,a)$\restriction$d is of one of the types $\{\text{Type}_1, \ldots \text{Type}_n\}$

Round 1

$\mathcal{G}_{\mathcal{A}}$

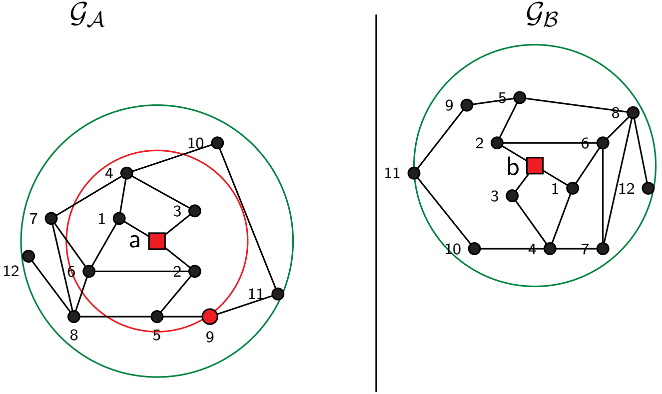


$\mathcal{G}_{\mathcal{B}}$



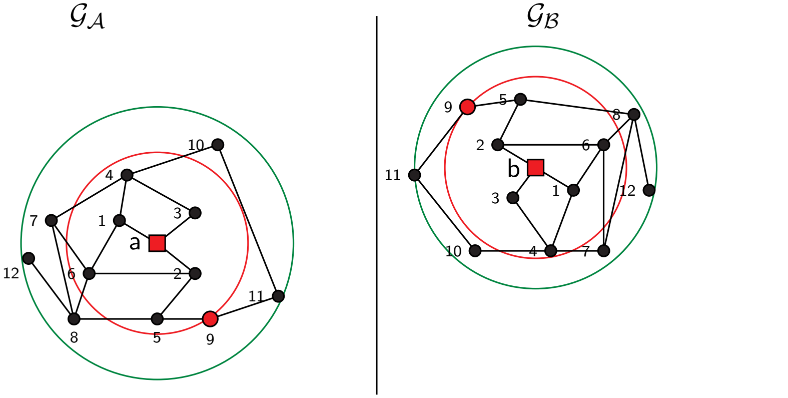
- The Spoiler chooses a vertex from any graph (here, $\mathcal{A}$)
- The d-neighbourhood of a in $\mathcal{A}$, denoted N($\mathcal{A}$,a)$\restriction$d is of one of the types $\{\text{Type}_1, \ldots\ \text{Type}_n\}$
- The Duplicator picks an element from the other structure (here, $b \in \mathcal{B}$) such that N($\mathcal{A}$,a)$\restriction$d $\cong$ N($\mathcal{B}$,b)$\restriction$d

Round i, Case 1



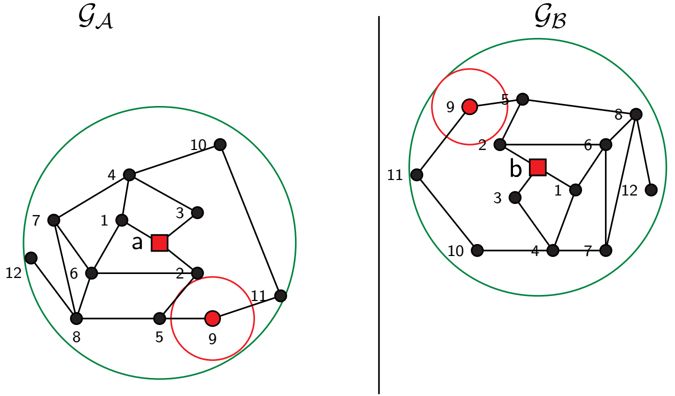
- In Round i, the first case is when the Spoiler picks a vertex that is within $2*3^{i-2}$ of any previously selected point (maybe more than one).

Round i, Case 1



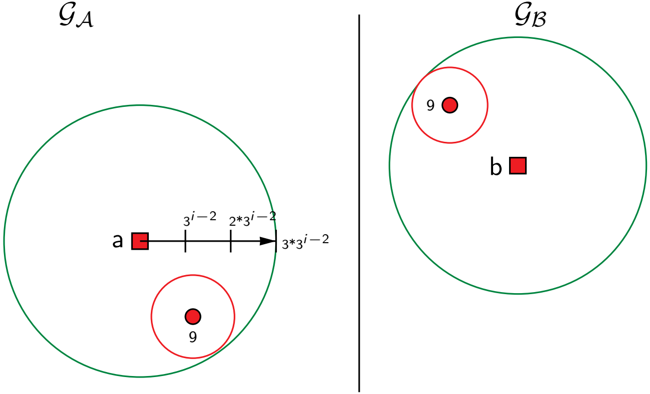
- In Round i, the first case is when the Spoiler picks a vertex that is within $2*3^{i-2}$ of any previously selected point (maybe more than one).
- Then the Duplicator will use the isomorphism of the d-radius sphere around any one of the centres to obtain a similar vertex on the other graph.

Round i, Case 1



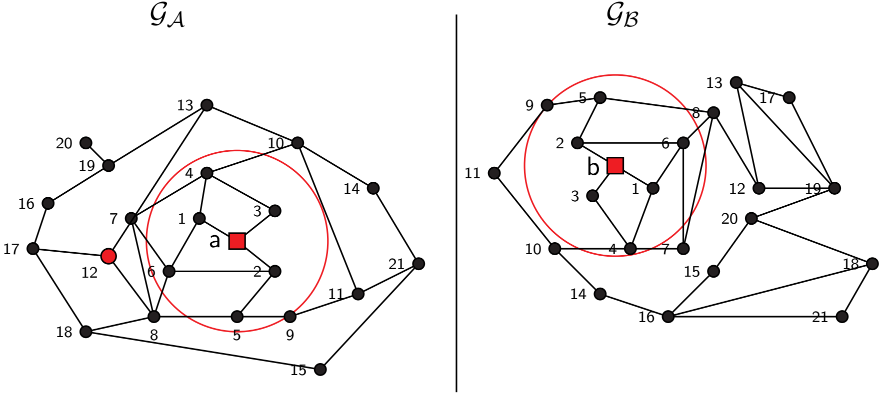
- The reason why it works is because, as shown in the figure, in subsequent i-1 rounds the Spoiler will not be able to get out of the isomorphism of the d-sphere around the previously selected points.

Round i, Case 1



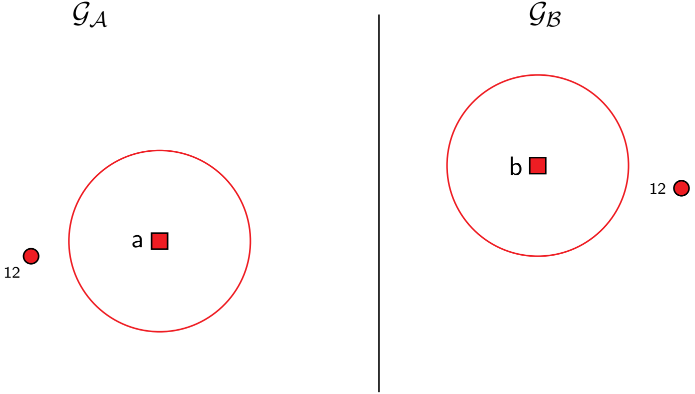
- The reason why it works is because, as shown in the figure, in subsequent i-1 rounds the Spoiler will not be able to get out of the isomorphism of the d-sphere around the respective previously selected points.
- This is due to the fact that $\forall$ i $\in \mathbb{N}$, $3^{i-2} \geq$ (i-1).

Round i, Case 2



- Now, the other case is if the Spoiler picks a vertex that is outside 2*3^{i-2} of **all** previously selected points.

Round i, Case 2



- Now, the other case is if the Spoiler picks a vertex that is outside $2*3^{i-2}$ of **all** previously selected points.
- Then, as in Round 1, the Duplicator will be able to pick a vertex $b \in \mathcal{B}$ such that $N(\mathcal{A},a)\restriction d \cong N(\mathcal{B},b)\restriction d$. Also, this particular point b is not in the range of $2*3^{i-2}$ distance of any other previously selected point in $\mathcal{G}_{\mathcal{B}}$.

Variations of Hanf's theorem

Definition

Given d,t $\in \mathbb{N}$, we can define the concept of type signatures of radius d with threshold t such that the values (#Type_1,...,#Type_n) are counted only upto a threshold t and anything $\geq$ t is considered ∞. Two structures $\mathcal{A}$ and $\mathcal{B}$, are said to be d-equivalent with threshold t if their type signatures with radius d are equal. It is denoted $\mathcal{A} \sim_{d,t} \mathcal{B}$.

Variations of Hanf's theorem

Definition

Given d,t $\in \mathbb{N}$, we can define the concept of type signatures of radius d with threshold t such that the values (#Type_1,...,#Type_n) are counted only upto a threshold t and anything $\geq$ t is considered ∞. Two structures $\mathcal{A}$ and $\mathcal{B}$, are said to be d-equivalent with threshold t if their type signatures with radius d are equal. It is denoted $\mathcal{A} \sim_{d,t} \mathcal{B}$.

Theorem

Given $d \in \mathbb{N}$ and two structures $\mathcal{A}$ and $\mathcal{B}$, if $\mathcal{A} \sim_d \mathcal{B}$ then there exists a fixed $t \in \mathbb{N}$ such that $\mathcal{A} \sim_{d,t} \mathcal{B}$.

Variations of Hanf's theorem

Definition

Given d,t $\in \mathbb{N}$, we can define the concept of type signatures of radius d with threshold t such that the values ($\#\text{Type}_1, \ldots, \#\text{Type}_n$) are counted only upto a threshold t and anything $\geq$ t is considered ∞. Two structures $\mathcal{A}$ and $\mathcal{B}$, are said to be d-equivalent with threshold t if their type signatures with radius d are equal. It is denoted $\mathcal{A} \sim_{d,t} \mathcal{B}$.

Theorem

Given $d \in \mathbb{N}$ and two structures $\mathcal{A}$ and $\mathcal{B}$, if $\mathcal{A} \sim_d \mathcal{B}$ then there exists a fixed $t \in \mathbb{N}$ such that $\mathcal{A} \sim_{d,t} \mathcal{B}$.

Theorem (Hanf's theorem for EMSO)

- *Let ϕ be an EMSO formula with n second-order quantifiers given by, $\phi = \exists X_1 \ldots \exists X_n \psi(X_1, \ldots, X_n)$, where ψ is a pure first order sentence.*

Variations of Hanf's theorem

Definition

Given d,t $\in \mathbb{N}$, we can define the concept of type signatures of radius d with threshold t such that the values (#Type$_1$,...,#Type$_n$) are counted only upto a threshold t and anything $\geq$ t is considered ∞. Two structures $\mathcal{A}$ and $\mathcal{B}$, are said to be d-equivalent with threshold t if their type signatures with radius d are equal. It is denoted $\mathcal{A} \sim_{d,t} \mathcal{B}$.

Theorem

Given $d \in \mathbb{N}$ and two structures $\mathcal{A}$ and $\mathcal{B}$, if $\mathcal{A} \sim_d \mathcal{B}$ then there exists a fixed $t \in \mathbb{N}$ such that $\mathcal{A} \sim_{d,t} \mathcal{B}$.

Theorem (Hanf's theorem for EMSO)

- *Let ϕ be an EMSO formula with n second-order quantifiers given by, $\phi = \exists X_1 \ldots \exists X_n \psi(X_1, \ldots, X_n)$, where ψ is a pure first order sentence.*
- *If we consider the extended models of $\mathcal{A}$, $\mathcal{A}' = (A \times 2^{\{0,1,\ldots k\}}, R_1^A, \ldots, R_m^A, c_1^A, \ldots, c_s^A)$, then we can reuse Hanf's theorem as only the ψ part remains to be interpreted over these modified structures.*

MSO and automata over pictures

MSO and automata over pictures

Definition (Pictures, Picture Languages)

- A picture, p, over an alphabet Σ is basically a function of the form
  $$p : \{1,2,\ldots, n\} \times \{1,2,\ldots,m\} \rightarrow \Sigma, \text{ for any } n,m \in \mathbb{N}$$
- The set of all pictures (over Σ) is the set of all possible functions p for every n,m $\in \mathbb{N}$. It is denoted by Σ^{**}
- A picture language is a subset of Σ^{**}

MSO and automata over pictures

Definition (Pictures, Picture Languages)

- A picture, p, over an alphabet Σ is basically a function of the form
  $$p : \{1,2,..., n\} \times \{1,2,...,m\} \to \Sigma, \text{ for any } n,m \in \mathbb{N}$$
- The set of all pictures (over Σ) is the set of all possible functions p for every n,m $\in \mathbb{N}$. It is denoted by Σ^{**}
- A picture language is a subset of Σ^{**}

p :

a	b
b	a

q :

a	a	b
a	b	a

r :

a	a	a	a	a	a	a
b	b	b	b	b	b	b
a	a	a	a	a	a	a
b	b	b	b	b	b	b
a	a	a	a	a	a	a

MSO and automata over pictures

Definition (Pictures, Picture Languages)

- A picture, p, over an alphabet Σ is basically a function of the form

  $$p : \{1,2,\dots, n\} \times \{1,2,\dots,m\} \rightarrow \Sigma, \text{ for any } n,m \in \mathbb{N}$$

- The set of all pictures (over Σ) is the set of all possible functions p for every n,m $\in \mathbb{N}$. It is denoted by Σ^{**}
- A picture language is a subset of Σ^{**}
- If p is a picture of size (m,n), then $\widehat{p}$ is the picture p surrounded by a special boundary symbol $\# \notin \Sigma$

$\hat{p}$:

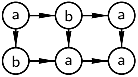
#	#	#	#
#	a	b	#
#	b	a	#
#	#	#	#

$\hat{r}$:

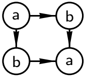
#	#	#	#	#	#	#	#	#
#	a	a	a	a	a	a	a	#
#	b	b	b	b	b	b	b	#
#	a	a	a	a	a	a	a	#
#	b	b	b	b	b	b	b	#
#	a	a	a	a	a	a	a	#
#	#	#	#	#	#	#	#	#

Row Concatenation of 2 pictures

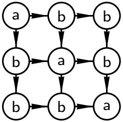
p :



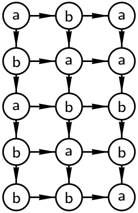
q :



r :



$p \ominus r$:

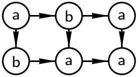


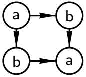
$p \ominus q$: Undefined

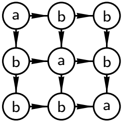
$q \ominus r$: Undefined

(As the number of columns are incompatible)

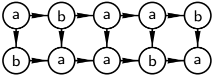
Column Concatenation of 2 pictures

p :


q :


r :


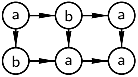
p ⊘ r : Undefined
q ⊘ r : Undefined

p ⊘ q:


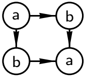
(As the number of rows are incompatible)

Row and Column Concatenation of 2 pictures

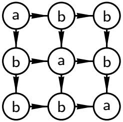
p :



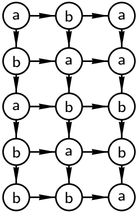
q :



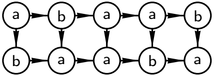
r :



$p \ominus r$:



$p \oslash q$:



Operations on Picture Languages

Definition

- Let L, L_1 and L_2 be 3 picture languages (subsets of Σ^{**})

Operations on Picture Languages

Definition

- Let L, L_1 and L_2 be 3 picture languages (subsets of Σ^{**})
- Then, $L_1 \oslash L_2 = \{x \oslash y \parallel x \in L_1 \text{ and } y \in L_2\}$. Similarly for $L_1 \ominus L_2$.

Operations on Picture Languages

Definition

- Let L, L_1 and L_2 be 3 picture languages (subsets of Σ^{**})
- Then, $L_1 \oslash L_2 = \{x \oslash y \parallel x \in L_1 \text{ and } y \in L_2\}$. Similarly for $L_1 \ominus L_2$.
- $L^{\oslash 1} = L$; $L^{\oslash n} = L^{\oslash (n-1)} \oslash L$. Similarly for $L^{\ominus n}$.

Operations on Picture Languages

Definition

- Let L, L_1 and L_2 be 3 picture languages (subsets of Σ^{**})
- Then, $L_1 \oslash L_2 = \{x \oslash y \parallel x \in L_1 \text{ and } y \in L_2\}$. Similarly for $L_1 \ominus L_2$.
- $L^{\oslash 1} = L$; $L^{\oslash n} = L^{\oslash (n-1)} \oslash L$. Similarly for $L^{\ominus n}$.
- Column Kleene Closure of L, $L^{*\oslash} = \cup_i L^{\oslash i}$
- Row Kleene Closure of L, $L^{*\ominus} = \cup_i L^{\ominus i}$

Operations on Picture Languages

Definition

- Let L,L_1 and L_2 be 3 picture languages (subsets of Σ^{**})
- Then, $L_1 \oslash L_2 = \{x \oslash y \parallel x \in L_1 \text{ and } y \in L_2\}$. Similarly for $L_1 \ominus L_2$.
- $L^{\oslash 1} = L$; $L^{\oslash n} = L^{\oslash (n-1)} \oslash L$. Similarly for $L^{\ominus n}$.
- Column Kleene Closure of L, $L^{*\oslash} = \cup_i L^{\oslash i}$
- Row Kleene Closure of L, $L^{*\ominus} = \cup_i L^{\ominus i}$

Definition (Projections)

- Let Σ_1 and Σ_2 be two finite alphabets such that $| \Sigma_1 | \geq | \Sigma_2 |$ and $\pi : \Sigma_1 \rightarrow \Sigma_2$ is a mapping.

Operations on Picture Languages

Definition

- Let L,L_1 and L_2 be 3 picture languages (subsets of Σ^{**})
- Then, $L_1 \oslash L_2 = \{x \oslash y \parallel x \in L_1 \text{ and } y \in L_2\}$. Similarly for $L_1 \ominus L_2$.
- $L^{\oslash 1} = L$; $L^{\oslash n} = L^{\oslash (n-1)} \oslash L$. Similarly for $L^{\ominus n}$.
- Column Kleene Closure of L, $L^{*\oslash} = \cup_i L^{\oslash i}$
- Row Kleene Closure of L, $L^{*\ominus} = \cup_i L^{\ominus i}$

Definition (Projections)

- Let Σ_1 and Σ_2 be two finite alphabets such that $\mid \Sigma_1 \mid \geq \mid \Sigma_2 \mid$ and $\pi : \Sigma_1 \rightarrow \Sigma_2$ is a mapping.
- Then given p $\in \Sigma_1^{**}$, $\pi(p)$ is the picture p' $\in \Sigma_2^{**}$ such that p'(i,j) $= \pi(p(i,j))\ \forall 1 \leq i \leq l_1(p), 1 \leq j \leq l_2(p)$

Operations on Picture Languages

Definition

- Let L, L_1 and L_2 be 3 picture languages (subsets of Σ^{**})
- Then, $L_1 \oslash L_2 = \{x \oslash y \parallel x \in L_1 \text{ and } y \in L_2\}$. Similarly for $L_1 \ominus L_2$.
- $L^{\oslash 1} = L$; $L^{\oslash n} = L^{\oslash (n-1)} \oslash L$. Similarly for $L^{\ominus n}$.
- Column Kleene Closure of L, $L^{*\oslash} = \cup_i L^{\oslash i}$
- Row Kleene Closure of L, $L^{*\ominus} = \cup_i L^{\ominus i}$

Definition (Projections)

- Let Σ_1 and Σ_2 be two finite alphabets such that $|\Sigma_1| \geq |\Sigma_2|$ and $\pi : \Sigma_1 \rightarrow \Sigma_2$ is a mapping.
- Then given p $\in \Sigma_1^{**}$, $\pi(p)$ is the picture p' $\in \Sigma_2^{**}$ such that p'(i,j) $= \pi(p(i,j))\ \forall 1 \leq i \leq l_1(p), 1 \leq j \leq l_2(p)$
- Similarly, given a picture language $L \subseteq \Sigma_1^{**}$, the projection of L by $\pi : \Sigma_1 \rightarrow \Sigma_2$ is defined as $\pi(L) = \{\pi(p) \mid p \in L\} \subseteq \Sigma_2^{**}$

Operations on Picture Languages

Definition

- Let L,L_1 and L_2 be 3 picture languages (subsets of Σ^{**})
- Then, $L_1 \oslash L_2 = \{x \oslash y \parallel x \in L_1 \text{ and } y \in L_2\}$. Similarly for $L_1 \ominus L_2$.
- $L^{\oslash 1} = L$; $L^{\oslash n} = L^{\oslash(n-1)} \oslash L$. Similarly for $L^{\ominus n}$.
- Column Kleene Closure of L, $L^{*\oslash} = \cup_i L^{\oslash i}$
- Row Kleene Closure of L, $L^{*\ominus} = \cup_i L^{\ominus i}$

Definition (Projections)

- Let Σ_1 and Σ_2 be two finite alphabets such that $|\Sigma_1| \geq |\Sigma_2|$ and $\pi : \Sigma_1 \to \Sigma_2$ is a mapping.
- Then given p $\in \Sigma_1^{**}$, $\pi(p)$ is the picture p' $\in \Sigma_2^{**}$ such that p'(i,j) $= \pi(p(i,j))\ \forall 1 \leq i \leq l_1(p), 1 \leq j \leq l_2(p)$
- Similarly, given a picture language $L \subseteq \Sigma_1^{**}$, the projection of L by $\pi : \Sigma_1 \to \Sigma_2$ is defined as $\pi(L) = \{\pi(p) \mid p \in L\} \subseteq \Sigma_2^{**}$
- Given a picture p of size (m,n), if $h \leq m$, $k \leq n$, we denote by $T_{h,k}(p)$ the set of all subpictures (contiguous rectangular subblocks) of p of size (h,k).

Local Picture Languages(LOC)

Definition

A picture language L $\subseteq \Gamma^{**}$ is local if there exists a set Δ of pictures (or "tiles") of size **(2,2)** over $\Gamma \cup \{\#\}$, such that L$=\{p \in \Gamma^{**} \mid T_{2,2}(\widehat{p}) \subseteq \Delta\}$

Local Picture Languages(LOC)

Definition

A picture language $L \subseteq \Gamma^{**}$ is local if there exists a set Δ of pictures (or "tiles") of size **(2,2)** over $\Gamma \cup \{\#\}$, such that $L=\{p \in \Gamma^{**} \mid T_{2,2}(\widehat{p}) \subseteq \Delta\}$

- If $L = \{p \in \Gamma^{**} \mid T_{2,2}(\widehat{p}) \subseteq \Delta\}$, then we call Δ a *local representation by tiles* for the language L.

Local Picture Languages(LOC)

Definition

A picture language $L \subseteq \Gamma^{**}$ is local if there exists a set Δ of pictures (or "tiles") of size **(2,2)** over $\Gamma \cup \{\#\}$, such that $L = \{p \in \Gamma^{**} \mid T_{2,2}(\widehat{p}) \subseteq \Delta\}$

- If $L = \{p \in \Gamma^{**} \mid T_{2,2}(\widehat{p}) \subseteq \Delta\}$, then we call Δ a *local representation by tiles* for the language L.
- We denote by LOC the family of local picture languages.

Local Picture Languages(LOC)

Definition

A picture language L $\subseteq \Gamma^{**}$ is local if there exists a set Δ of pictures (or "tiles") of size **(2,2)** over $\Gamma \cup \{\#\}$, such that L=$\{p \in \Gamma^{**} \mid T_{2,2}(\widehat{p}) \subseteq \Delta\}$

- If L $= \{p \in \Gamma^{**} \mid T_{2,2}(\widehat{p}) \subseteq \Delta\}$, then we call Δ a *local representation by tiles* for the language L.
- We denote by LOC the family of local picture languages.
- An example of a picture language in LOC, consider $L_0 \subseteq \{0,1\}^{**}$ of square pictures (of size at least (2,2)) in which all nondiagonal positions carry symbol 0 whereas the diagonal positions carry symbol 1.

Local Picture Languages(LOC)

Definition

A picture language $L \subseteq \Gamma^{**}$ is local if there exists a set Δ of pictures (or "tiles") of size **(2,2)** over $\Gamma \cup \{\#\}$, such that $L = \{p \in \Gamma^{**} \mid T_{2,2}(\widehat{p}) \subseteq \Delta\}$

- If $L = \{p \in \Gamma^{**} \mid T_{2,2}(\widehat{p}) \subseteq \Delta\}$, then we call Δ a *local representation by tiles* for the language L.
- We denote by LOC the family of local picture languages.
- An example of a picture language in LOC, consider $L_0 \subseteq \{0,1\}^{**}$ of square pictures (of size at least (2,2)) in which all nondiagonal positions carry symbol 0 whereas the diagonal positions carry symbol 1.
- An appropriate set of tiles for L_0 consists of the 16 different (2,2)-subblocks of the picture on the right.

Local Picture Languages(LOC)

Definition

A picture language L $\subseteq \Gamma^{**}$ is local if there exists a set Δ of pictures (or "tiles") of size **(2,2)** over $\Gamma \cup \{\#\}$, such that $L = \{p \in \Gamma^{**} \mid T_{2,2}(\widehat{p}) \subseteq \Delta\}$

- If $L = \{p \in \Gamma^{**} \mid T_{2,2}(\widehat{p}) \subseteq \Delta\}$, then we call Δ a *local representation by tiles* for the language L.
- We denote by LOC the family of local picture languages.
- An example of a picture language in LOC, consider $L_0 \subseteq \{0,1\}^{**}$ of square pictures (of size at least (2,2)) in which all nondiagonal positions carry symbol 0 whereas the diagonal positions carry symbol 1.
- An appropriate set of tiles for L_0 consists of the 16 different (2,2)-subblocks of the picture on the right.

#	#	#	#	#	#
#	1	0	0	0	#
#	0	1	0	0	#
#	0	0	1	0	#
#	0	0	0	1	#
#	#	#	#	#	#

Recognizable Picture Languages(REC)

Definition

A picture language L $\subseteq \Sigma^{**}$ is recognizable if there exists a local language L' over an alphabet Γ and a mapping

$\pi : \Gamma \rightarrow \Sigma$ such that L$=\pi(L')$

- As an example of such a language is the set of squares over $\Sigma = \{a\}$ and a suitable local language would be L_0 considered previously and the mapping, $\pi : \{0, 1\} \rightarrow \{a\}$

Recognizable Picture Languages(REC)

Definition

A picture language $L \subseteq \Sigma^{**}$ is recognizable if there exists a local language L' over an alphabet Γ and a mapping
$\pi : \Gamma \rightarrow \Sigma$ such that $L = \pi(L')$

- As an example of such a language is the set of squares over $\Sigma = \{a\}$ and a suitable local language would be L_0 considered previously and the mapping, $\pi : \{0, 1\} \rightarrow \{a\}$
- We consider only $\Gamma = \Sigma \times Q$ and $\pi : \Sigma \times Q \rightarrow \Sigma$ as the alphabet and mapping respectively for the local language L' of L.

Recognizable Picture Languages(REC)

Definition

A picture language L $\subseteq \Sigma^{**}$ is recognizable if there exists a local language L' over an alphabet Γ and a mapping
$\pi : \Gamma \to \Sigma$ such that L=$\pi(L')$

- As an example of such a language is the set of squares over $\Sigma = \{a\}$ and a suitable local language would be L_0 considered previously and the mapping, $\pi : \{0, 1\} \to \{a\}$
- We consider only $\Gamma = \Sigma\times$ Q and $\pi : \Sigma\times$ Q $\to \Sigma$ as the alphabet and mapping respectively for the local language L' of L.
- It is sufficient to consider Local languages of the type above as every local language L' given in the definition, with an arbitrary alphabet may be modified into a local language with the alphabet, $\Gamma = \Sigma\times$ Q.

Recognizable Picture Languages(REC)

Definition

A picture language L $\subseteq \Sigma^{**}$ is recognizable if there exists a local language L' over an alphabet Γ and a mapping

$\pi : \Gamma \rightarrow \Sigma$ such that L$=\pi(L')$

- As an example of such a language is the set of squares over $\Sigma = \{a\}$ and a suitable local language would be L_0 considered previously and the mapping, $\pi : \{0, 1\} \rightarrow \{a\}$
- We consider only $\Gamma = \Sigma \times$ Q and $\pi : \Sigma \times$ Q $\rightarrow \Sigma$ as the alphabet and mapping respectively for the local language L' of L.
- It is sufficient to consider Local languages of the type above as every local language L' given in the definition, with an arbitrary alphabet may be modified into a local language with the alphabet, $\Gamma = \Sigma \times$ Q.
- Under the above considerations, the tiling System is denoted by the triple (Σ,Q, Δ).

Automata Theoretic Approach to Picture Languages

- Row and column concatenation.

Automata Theoretic Approach to Picture Languages

- Row and column concatenation.
- Row and column closure.

Automata Theoretic Approach to Picture Languages

- Row and column concatenation.
- Row and column closure.
- Projections of picture languages.

Automata Theoretic Approach to Picture Languages

- Row and column concatenation.
- Row and column closure.
- Projections of picture languages.
- Local Picture languages (LOC).

Automata Theoretic Approach to Picture Languages

- Row and column concatenation.
- Row and column closure.
- Projections of picture languages.
- Local Picture languages (LOC).
- Recognizable picture languages (REC).

Automata Theoretic Approach to Picture Languages

- Row and column concatenation.
- Row and column closure.
- Projections of picture languages.
- Local Picture languages (LOC).
- Recognizable picture languages (REC).
- REC is closed with respect to
  - projection,

Automata Theoretic Approach to Picture Languages

- Row and column concatenation.
- Row and column closure.
- Projections of picture languages.
- Local Picture languages (LOC).
- Recognizable picture languages (REC).
- REC is closed with respect to
  - projection,
  - row and column concatenation,

Automata Theoretic Approach to Picture Languages

- Row and column concatenation.
- Row and column closure.
- Projections of picture languages.
- Local Picture languages (LOC).
- Recognizable picture languages (REC).
- REC is closed with respect to
  - projection,
  - row and column concatenation,
  - row and column closure,

Automata Theoretic Approach to Picture Languages

- Row and column concatenation.
- Row and column closure.
- Projections of picture languages.
- Local Picture languages (LOC).
- Recognizable picture languages (REC).
- REC is closed with respect to
  - projection,
  - row and column concatenation,
  - row and column closure,
  - Boolean union,

Automata Theoretic Approach to Picture Languages

- Row and column concatenation.
- Row and column closure.
- Projections of picture languages.
- Local Picture languages (LOC).
- Recognizable picture languages (REC).
- REC is closed with respect to
  - projection,
  - row and column concatenation,
  - row and column closure,
  - Boolean union,
  - Boolean intersection.

Automata Theoretic Approach to Picture Languages

- Row and column concatenation.
- Row and column closure.
- Projections of picture languages.
- Local Picture languages (LOC).
- Recognizable picture languages (REC).
- REC is closed with respect to
  - projection,
  - row and column concatenation,
  - row and column closure,
  - Boolean union,
  - Boolean intersection.
- REC is not closed with respect to Boolean complementation.

Automata Theoretic Approach to Picture Languages

- Row and column concatenation.
- Row and column closure.
- Projections of picture languages.
- Local Picture languages (LOC).
- Recognizable picture languages (REC).
- REC is closed with respect to
  - projection,
  - row and column concatenation,
  - row and column closure,
  - Boolean union,
  - Boolean intersection.
- REC is not closed with respect to Boolean complementation.

  Proof idea :

Automata Theoretic Approach to Picture Languages

- Row and column concatenation.
- Row and column closure.
- Projections of picture languages.
- Local Picture languages (LOC).
- Recognizable picture languages (REC).
- REC is closed with respect to
  - projection,
  - row and column concatenation,
  - row and column closure,
  - Boolean union,
  - Boolean intersection.
- REC is not closed with respect to Boolean complementation.

  Proof idea :

  - Let Σ be an alphabet and let L be a language over Σ given by
    $L=\{p \in \Sigma^{**} \mid p = s \ominus s$ where s is a square $\}$

Automata Theoretic Approach to Picture Languages

- Row and column concatenation.
- Row and column closure.
- Projections of picture languages.
- Local Picture languages (LOC).
- Recognizable picture languages (REC).
- REC is closed with respect to
  - projection,
  - row and column concatenation,
  - row and column closure,
  - Boolean union,
  - Boolean intersection.
- REC is not closed with respect to Boolean complementation.

  Proof idea :

  - Let Σ be an alphabet and let L be a language over Σ given by
    L=$\{$p$\in \Sigma^{**}$ | p=s$\ominus$s where s is a square $\}$
  - The claim is that L $\notin$ REC while $\overline{L} \in$ REC.

Logical definability of Picture Languages

A few notations and terminologies :

- Given a picture $p \in \Sigma^{**}$, we can identify the structure $\underline{p} = (\text{dom}(p), S_1, S_2, (P_a)_{a \in \Sigma})$,

Logical definability of Picture Languages

A few notations and terminologies :

- Given a picture $p \in \Sigma^{**}$, we can identify the structure
  $\underline{p} = (\text{dom}(p), S_1, S_2, (P_a)_{a \in \Sigma})$,
- x, y, z, x_1, x_2, ..., are first-order variables for points of dom(p) while X, Y, Z, X_1, X_2, ..., are MSO variables denoting sets of positions.

Logical definability of Picture Languages

A few notations and terminologies :

- Given a picture $p \in \Sigma^{**}$, we can identify the structure
  $\underline{p} = (\mathrm{dom}(p), S_1, S_2, (P_a)_{a \in \Sigma})$,
- x, y, z, x_1, x_2, ..., are first-order variables for points of dom(p) while X, Y, Z, X_1, X_2, ..., are MSO variables denoting sets of positions.
- Atomic formulas are of the form $x{=}y$, xS_iy, $X(x)$ and $P_a(x)$

Logical definability of Picture Languages

A few notations and terminologies :

- Given a picture p $\in \Sigma^{**}$, we can identify the structure
  $\underline{p} = (\mathrm{dom(p)}, S_1, S_2, (P_a)_{a\in\Sigma})$,
- $x, y, z, x_1, x_2, \ldots$, are first-order variables for points of dom(p) while $X, Y, Z, X_1, X_2, \ldots$, are MSO variables denoting sets of positions.
- Atomic formulas are of the form x=y, xS_iy, X(x) and $P_a(x)$ interpreted as equality between x and y, $(x,y) \in S_i$, $x \in X$, $x \in P_a$ respectively.

Logical definability of Picture Languages

A few notations and terminologies :

- Given a picture $p \in \Sigma^{**}$, we can identify the structure
  $\underline{p} = (\text{dom(p)}, S_1, S_2, (P_a)_{a \in \Sigma})$,
- x, y, z, x_1, x_2, ..., are first-order variables for points of dom(p) while X, Y, Z, X_1, X_2, ..., are MSO variables denoting sets of positions.
- Atomic formulas are of the form $x=y$, xS_iy, $X(x)$ and $P_a(x)$ interpreted as equality between x and y, $(x,y) \in S_i$, $x \in X$, $x \in P_a$ respectively.
- Formulas are built up from atomic formulas by means of the Boolean connectives and the quantifiers $\exists$ and $\forall$ applicable to first-order as well as second-order variables.

Logical definability of Picture Languages

A few notations and terminologies :

- Given a picture p $\in \Sigma^{**}$, we can identify the structure
  $\underline{p} = (\text{dom}(p), S_1, S_2, (P_a)_{a \in \Sigma})$,
- $x, y, z, x_1, x_2, \ldots,$ are first-order variables for points of dom(p) while $X, Y, Z, X_1, X_2, \ldots,$ are MSO variables denoting sets of positions.
- Atomic formulas are of the form $x=y$, xS_iy, $X(x)$ and $P_a(x)$ interpreted as equality between x and y, $(x,y) \in S_i$, $x \in X$, $x \in P_a$ respectively.
- Formulas are built up from atomic formulas by means of the Boolean connectives and the quantifiers $\exists$ and $\forall$ applicable to first-order as well as second-order variables.
- If $\phi(X_1, \ldots, X_n)$ is a formula with at most $X_1, \ldots, X_n$ occurring free in ϕ, p is a picture, and $Q_1, \ldots, Q_n$ are subsets of dom(p), we write

Logical definability of Picture Languages

A few notations and terminologies :

- Given a picture $p \in \Sigma^{**}$, we can identify the structure
  $\underline{p} = (\text{dom}(p), S_1, S_2, (P_a)_{a \in \Sigma})$,
- x, y, z, x_1, x_2, ..., are first-order variables for points of dom(p) while X, Y, Z, X_1, X_2, ..., are MSO variables denoting sets of positions.
- Atomic formulas are of the form $x=y$, xS_iy, $X(x)$ and $P_a(x)$ interpreted as equality between x and y, $(x,y) \in S_i$, $x \in X$, $x \in P_a$ respectively.
- Formulas are built up from atomic formulas by means of the Boolean connectives and the quantifiers $\exists$ and $\forall$ applicable to first-order as well as second-order variables.
- If $\phi(X_1, ..., X_n)$ is a formula with at most $X_1, ..., X_n$ occurring free in ϕ, p is a picture, and $Q_1, ..., Q_n$ are subsets of dom(p), we write

$$((\underline{p}), Q_1, ..., Q_n) \models \phi(X_1, ..., X_n)$$

Logical definability of Picture Languages

A few notations and terminologies :

- Given a picture p $\in \Sigma^{**}$, we can identify the structure
  $\underline{p} = (\text{dom}(p), S_1, S_2, (P_a)_{a\in\Sigma})$,
- x, y, z, x_1, x_2, ..., are first-order variables for points of dom(p) while X, Y, Z, X_1, X_2, ..., are MSO variables denoting sets of positions.
- Atomic formulas are of the form $x=y$, xS_iy, $X(x)$ and $P_a(x)$ interpreted as equality between x and y, $(x,y) \in S_i$, $x \in X$, $x \in P_a$ respectively.
- Formulas are built up from atomic formulas by means of the Boolean connectives and the quantifiers $\exists$ and $\forall$ applicable to first-order as well as second-order variables.
- If $\phi(X_1, ..., X_n)$ is a formula with at most $X_1, ..., X_n$ occurring free in ϕ, p is a picture, and $Q_1, ..., Q_n$ are subsets of dom(p), we write

  $$((\underline{p}), Q_1, ..., Q_n) \models \phi(X_1, ..., X_n)$$

  if p satisfies ϕ under the above mentioned interpretation where Q_i is taken as interpretation of X_i.

Logical definability of Picture Languages

A few notations and terminologies :

- Given a picture $p \in \Sigma^{**}$, we can identify the structure
  $\underline{p} = (\text{dom}(p), S_1, S_2, (P_a)_{a\in\Sigma})$,
- x, y, z, x_1, x_2, ..., are first-order variables for points of dom(p) while X, Y, Z, X_1, X_2, ..., are MSO variables denoting sets of positions.
- Atomic formulas are of the form $x=y$, xS_iy, $X(x)$ and $P_a(x)$ interpreted as equality between x and y, $(x,y) \in S_i$, $x \in X$, $x \in P_a$ respectively.
- Formulas are built up from atomic formulas by means of the Boolean connectives and the quantifiers $\exists$ and $\forall$ applicable to first-order as well as second-order variables.
- If $\phi(X_1, ..., X_n)$ is a formula with at most X_1, ..., X_n occurring free in ϕ, p is a picture, and Q_1, ...,Q_n are subsets of dom(p), we write

  $$((\underline{p}), Q_1, ..., Q_n) \models \phi(X_1, ..., X_n)$$

  if p satisfies ϕ under the above mentioned interpretation where Q_i is taken as interpretation of X_i.
- If ϕ is a sentence we write $\underline{p} \models \phi$.

Logical Definability of Picture Languages

Definition (MSO-definable)

A picture language L is monadic second-order definable (L $\in$ MSO), if there is a monadic second-order sentence ϕ with L $=$ L(ϕ).

Logical Definability of Picture Languages

Definition (MSO-definable)

A picture language L is monadic second-order definable ($L \in MSO$), if there is a monadic second-order sentence ϕ with $L = L(\phi)$.

Definition (FO-definable)

A picture language L is first-order definable ($L \in FO$), if there is a sentence ϕ conatining only first-order quantifiers such that $L = L(\phi)$.

Logical Definability of Picture Languages

Definition (MSO-definable)

A picture language L is monadic second-order definable ($L \in MSO$), if there is a monadic second-order sentence ϕ with $L = L(\phi)$.

Definition (FO-definable)

A picture language L is first-order definable ($L \in FO$), if there is a sentence ϕ conatining only first-order quantifiers such that $L = L(\phi)$.

Definition (EMSO-definable)

Finally, A picture language L is existential monadic second-order definable ($L \in EMSO$), if there is a sentence of the form

$$\phi = \exists X_1 \ \ldots \ \exists X_n \psi(X_1, \ldots, X_n) \text{ where } \psi \text{ contains only first-order}$$

quantifiers such that $L = L(\phi)$.

REC $\iff$ EMSO

Now, the main theorem concerning picture languages

Theorem

For any picture language L:L$\in$ REC iff L $\in$ EMSO.

REC $\iff$ EMSO

Now, the main theorem concerning picture languages

Theorem

For any picture language L:L$\in$ REC iff L $\in$ EMSO.

Proof Idea :

- The direction (REC $\implies$ EMSO) is the easy one, and all we have to do is prove the following lemma

REC $\iff$ EMSO

Now, the main theorem concerning picture languages

Theorem

For any picture language L:L$\in$ REC iff L $\in$ EMSO.

Proof Idea :

- The direction (REC $\implies$ EMSO) is the easy one, and all we have to do is prove the following lemma

Definition

p$\in$ L iff $\exists$ picture c $\in$ Q** of the same size as p such that $\widehat{p \times c}$ is tilable by Δ

REC $\Longleftrightarrow$ EMSO

Now, the main theorem concerning picture languages

Theorem

For any picture language L:L$\in$ REC iff L $\in$ EMSO.

Proof Idea :

- The direction (REC $\Longrightarrow$ EMSO) is the easy one, and all we have to do is prove the following lemma

Definition

p$\in$ L iff $\exists$ picture c $\in$ Q^{**} of the same size as p such that $\widehat{p \times c}$ is tilable by Δ

- We capture the tiling of the picture $\widehat{p \times c}$ by the EMSO formula
  $\phi = \exists X_1 \ldots \exists X_k(\phi_{partition} \wedge \forall\ x_1 \ldots x_4$
  $(\chi_m \wedge \chi_t \wedge \chi_b \wedge \chi_l \wedge \chi_r \wedge \chi_{tl} \wedge \chi_{tr} \wedge \chi_{bl} \wedge \chi_{br}))$

REC $\iff$ EMSO

Now, the main theorem concerning picture languages

Theorem

For any picture language L:L$\in$ REC iff L $\in$ EMSO.

Proof Idea :

- The direction (REC $\Longrightarrow$ EMSO) is the easy one, and all we have to do is prove the following lemma

Definition

p$\in L$ iff $\exists$ picture c $\in$ Q** of the same size as p such that $\widehat{p \times c}$ is tilable by Δ

- We capture the tiling of the picture $\widehat{p \times c}$ by the EMSO formula
  $\phi = \exists X_1 ... \exists X_k(\phi_{partition} \wedge \forall\ x_1 ... x_4$
  $(\chi_m \wedge \chi_t \wedge \chi_b \wedge \chi_l \wedge \chi_r \wedge \chi_{tl} \wedge \chi_{tr} \wedge \chi_{bl} \wedge \chi_{br}))$
- $\phi_{partition}(X_1,...,X_k)$:
  $\forall z(X_1(z) \vee ... \vee X_k(z)) \wedge \bigwedge_{i \neq j} \neg(X_i(z) \wedge X_j(z))$.

REC $\iff$ EMSO

Now, the main theorem concerning picture languages

Theorem

For any picture language L:L$\in$ REC iff L $\in$ EMSO.

Proof Idea :

- The direction (REC $\Longrightarrow$ EMSO) is the easy one, and all we have to do is prove the following lemma

Definition

p$\in$ L iff $\exists$ picture c $\in$ Q^{**} of the same size as p such that $\widehat{p \times c}$ is tilable by Δ

- We capture the tiling of the picture $\widehat{p \times c}$ by the EMSO formula
$\phi = \exists X_1 \ldots \exists X_k(\phi_{partition} \wedge \forall\, x_1 \ldots x_4$
$(\chi_m \wedge \chi_t \wedge \chi_b \wedge \chi_l \wedge \chi_r \wedge \chi_{tl} \wedge \chi_{tr} \wedge \chi_{bl} \wedge \chi_{br}))$
- $\phi_{partition}(X_1,\ldots,X_k)$:
$\forall z(X_1(z) \vee \ldots \vee X_k(z)) \wedge \bigwedge_{i \neq j} \neg(X_i(z) \wedge X_j(z))$.
- While $\chi_m,\chi_t,\chi_b,\chi_l,\chi_r,\chi_{tl},\chi_{tr},\chi_{bl},\chi_{br}$ refer to the formulae describing (2,2) local neighbourhoods.

EMSO $\implies$ REC

For this direction, we use the variation of hanf's theorem for EMSO.

EMSO $\implies$ REC

For this direction, we use the variation of hanf's theorem for EMSO.

- Given two pictures, p_1,p_2, $d,t \in \mathbb{N}$, if $T_{(i,j)}(p_1)\restriction(d,t)=T_{(i,j)}(p_2)\restriction(d,t)$ $\forall$ $i,j \leq d$, then we say that p_1 is d,t-equivalent to p_2 denoted $p_1 \sim_{d,t} p_2$.
- For pictures, we consider only rectangles and not spheres.

EMSO $\implies$ REC

For this direction, we use the variation of hanf's theorem for EMSO.

- Given two pictures, p_1,p_2, d,t $\in \mathbb{N}$, if $T_{(i,j)}(p_1)\restriction(d,t) = T_{(i,j)}(p_2)\restriction(d,t)$ $\forall$ i,j $\leq d$, then we say that p_1 is d,t-equivalent to p_2 denoted $p_1 \sim_{d,t} p_2$.
- For pictures, we consider only rectangles and not spheres.
- For the same parameters above, if $T_{(i,j)}(p_1)\restriction(d,t) = T_{(i,j)}(p_2)\restriction(d,t)$ $\forall$ i,j $= d$, then we say that p_1 is exactly d,t-equivalent to p_2 denoted $p_1 \simeq_{d,t} p_2$.

EMSO $\Longrightarrow$ REC

For this direction, we use the variation of hanf's theorem for EMSO.

- Given two pictures, p_1,p_2, d,t $\in \mathbb{N}$, if $T_{(i,j)}(p_1)\upharpoonright(d,t) = T_{(i,j)}(p_2)\upharpoonright(d,t)$ $\forall$ i,j $\leq d$, then we say that p_1 is d,t-equivalent to p_2 denoted $p_1 \sim_{d,t} p_2$.
- For pictures, we consider only rectangles and not spheres.
- For the same parameters above, if $T_{(i,j)}(p_1)\upharpoonright(d,t) = T_{(i,j)}(p_2)\upharpoonright(d,t)$ $\forall$ i,j $= d$, then we say that p_1 is exactly d,t-equivalent to p_2 denoted $p_1 \simeq_{d,t} p_2$.
- A picture language,L, is called **locally d-testable with threshold t** if L is a union of $\sim_{d,t}$-equivalence classes

EMSO $\implies$ REC

For this direction, we use the variation of hanf's theorem for EMSO.

- Given two pictures, p_1,p_2, d,t $\in \mathbb{N}$, if $T_{(i,j)}(p_1)\restriction(d,t)=T_{(i,j)}(p_2)\restriction(d,t)$ $\forall$ i,j $\leq d$, then we say that p_1 is d,t-equivalent to p_2 denoted $p_1 \sim_{d,t} p_2$.
- For pictures, we consider only rectangles and not spheres.
- For the same parameters above, if $T_{(i,j)}(p_1)\restriction(d,t)=T_{(i,j)}(p_2)\restriction(d,t)$ $\forall$ i,j $= d$, then we say that p_1 is exactly d,t-equivalent to p_2 denoted $p_1 \simeq_{d,t} p_2$.
- A picture language,L, is called **locally d-testable with threshold t** if L is a union of $\sim_{d,t}$-equivalence classes
- If it holds for some t,we say that L is **locally threshold d-testable**.

EMSO $\implies$ REC

For this direction, we use the variation of hanf's theorem for EMSO.

- Given two pictures, p_1,p_2, d,t $\in \mathbb{N}$, if $T_{(i,j)}(p_1)\upharpoonright(d,t) = T_{(i,j)}(p_2)\upharpoonright(d,t)$ $\forall$ i,j $\leq d$, then we say that p_1 is d,t-equivalent to p_2 denoted $p_1 \sim_{d,t} p_2$.
- For pictures, we consider only rectangles and not spheres.
- For the same parameters above, if $T_{(i,j)}(p_1)\upharpoonright(d,t) = T_{(i,j)}(p_2)\upharpoonright(d,t)$ $\forall$ i,j $= d$, then we say that p_1 is exactly d,t-equivalent to p_2 denoted $p_1 \simeq_{d,t} p_2$.
- A picture language,L, is called **locally d-testable with threshold t** if L is a union of $\sim_{d,t}$-equivalence classes
- If it holds for some t,we say that L is **locally threshold d-testable**.
- If it holds for some d and t then we say L is **locally threshold testable**.

EMSO $\Longrightarrow$ REC

For this direction, we use the variation of hanf's theorem for EMSO.

- Given two pictures, p_1,p_2, d,t $\in \mathbb{N}$, if $T_{(i,j)}(p_1)\restriction(d,t)=T_{(i,j)}(p_2)\restriction(d,t)$ $\forall$ i,j $\leq d$, then we say that p_1 is d,t-equivalent to p_2 denoted $p_1 \sim_{d,t} p_2$.
- For pictures, we consider only rectangles and not spheres.
- For the same parameters above, if $T_{(i,j)}(p_1)\restriction(d,t)=T_{(i,j)}(p_2)\restriction(d,t)$ $\forall$ i,j $= d$, then we say that p_1 is exactly d,t-equivalent to p_2 denoted $p_1 \simeq_{d,t} p_2$.
- A picture language,L, is called **locally d-testable with threshold t** if L is a union of $\sim_{d,t}$-equivalence classes
- If it holds for some t,we say that L is **locally threshold d-testable**.
- If it holds for some d and t then we say L is **locally threshold testable**.
- Finally, if L is a union of $\simeq_{d,t}$-classes for some t, L is called **locally strictly threshold d-testable**.

EMSO $\implies$ REC

Theorem (Theorem 1)

A picture language is first-order definable iff it is locally threshold testable.

EMSO $\implies$ REC

Theorem (Theorem 1)

A picture language is first-order definable iff it is locally threshold testable.

Theorem (Theorem 2)

Using theorem 1, we claim that if $L \in EMSO$ then L is a projection of a locally threshold testable picture language.

EMSO $\Longrightarrow$ REC

Theorem (Theorem 1)

A picture language is first-order definable iff it is locally threshold testable.

Theorem (Theorem 2)

Using theorem 1, we claim that if $L \in$ EMSO then L is a projection of a locally threshold testable picture language.

Thus, Theorem 1 $\Longrightarrow$ Theorem 2 $\Longrightarrow$ (EMSO $\Longrightarrow$ REC)

We only need to prove Theorem 1 now.

EMSO $\Longrightarrow$ REC

Theorem (Theorem 1)

A picture language is first-order definable iff it is locally threshold testable.

Theorem (Theorem 2)

Using theorem 1, we claim that if $L \in$ EMSO then L is a projection of a locally threshold testable picture language.

Thus, Theorem 1 $\Longrightarrow$ Theorem 2 $\Longrightarrow$ (EMSO $\Longrightarrow$ REC)

We only need to prove Theorem 1 now.

Proof sketch of Theorem 1 :-

- The proof for the direction $\Longleftarrow$ is by an adaptation of Hanf's theorem to pictures. We can use the bound as $d=2*3^n+1$ and $t=n*3^{2n}$ for n-equivalence.

EMSO $\implies$ REC

Theorem (Theorem 1)

A picture language is first-order definable iff it is locally threshold testable.

Theorem (Theorem 2)

Using theorem 1, we claim that if $L \in$ EMSO then L is a projection of a locally threshold testable picture language.

Thus, Theorem 1 $\implies$ Theorem 2 $\implies$ (EMSO $\implies$ REC)

We only need to prove Theorem 1 now.

Proof sketch of Theorem 1 :-

- The proof for the direction $\impliedby$ is by an adaptation of Hanf's theorem to pictures. We can use the bound as d=2*3^n+1 and t=n*3^{2n} for n-equivalence.
- So that completes the proof for $\impliedby$ of theorem 1 and we only need to prove the reverse direction.

First-order definable $\Longrightarrow$ Locally threshold testable

Proof Sketch :-

Theorem

Each locally threshold d-testable language L can be decomposed into $L_0 \cup L_1 \cup \ldots \cup L_{d-2}$ *where* $L_i \subseteq \Sigma_i^{**}$ *(0≤i≤d-2) is locally strictly threshold (i+2)-testable.*

First-order definable $\Longrightarrow$ Locally threshold testable

Proof Sketch :-

Theorem

Each locally threshold d-testable language L can be decomposed into $L_0 \cup L_1 \cup \ldots \cup L_{d-2}$ *where* $L_i \subseteq \Sigma_i^{**}$ *(*$0 \leq i \leq d-2$*) is locally strictly threshold (i+2)-testable.*

Definition

Let $d \geq 2$ be a positive integer. A picture language $L \subseteq \Sigma_{d-2}^{**}$ is d-local if there exists a set $\Delta_{(d)}$ of pictures of size (d,d) (or "d-tiles") over $\Sigma \cup \{\#\}$, such that $L = \{p \in \Sigma^{**} \mid T_{d,d}(\widehat{p}) \subseteq \Delta^{(d)}\}$

First-order definable $\Longrightarrow$ Locally threshold testable

Proof Sketch :-

Theorem

Each locally threshold d-testable language L can be decomposed into $L_0 \cup L_1 \cup \ldots \cup L_{d-2}$ where $L_i \subseteq \Sigma_i^{**}$ ($0 \le i \le d-2$) is locally strictly threshold (i+2)-testable.

Definition

Let $d \geq 2$ be a positive integer. A picture language $L \subseteq \Sigma_{d-2}^{**}$ is d-local if there exists a set $\Delta_{(d)}$ of pictures of size (d,d) (or "d-tiles") over $\Sigma \cup \{\#\}$, such that $L = \{p \in \Sigma^{**} \mid T_{d,d}(\widehat{p}) \subseteq \Delta^{(d)}\}$

Theorem

Let $d \geq 3$ be a positive integer. A locally strictly threshold d-testable picture language $L \subseteq \Sigma_{d-2}^{**}$ is the projection of a d-local language.

First-order definable $\Longrightarrow$ Locally threshold testable

Proof Sketch :-

Theorem

Each locally threshold d-testable language L can be decomposed into $L_0 \cup L_1 \cup \ldots \cup L_{d-2}$ where $L_i \subseteq \Sigma_i^{**}$ ($0 \leq i \leq d-2$) is locally strictly threshold (i+2)-testable.

Definition

Let $d \geq 2$ be a positive integer. A picture language $L \subseteq \Sigma_{d-2}^{**}$ is d-local if there exists a set $\Delta_{(d)}$ of pictures of size (d,d) (or "d-tiles") over $\Sigma \cup \{\#\}$, such that $L = \{p \in \Sigma^{**} \mid T_{d,d}(\widehat{p}) \subseteq \Delta^{(d)}\}$

Theorem

Let $d \geq 3$ be a positive integer. A locally strictly threshold d-testable picture language $L \subseteq \Sigma_{d-2}^{**}$ is the projection of a d-local language.

Theorem

A d-local picture language is a projection of a local language.

Alternation hierarchy of MSO over grids and graphs

Alternation hierarchy of MSO over grids and graphs

- The signature of grids is given by $\tau_{Grid} = ([m,n], S_1^{m,n}, S_2^{m,n})$
  where $[m,n] = [m] \times [n]$
- The signature of t-bit grids for some $t \in \mathbb{N}$ is given by
  $\tau_{t-Grid} = ([m,n], S_1^{m,n}, S_2^{m,n}, X_1, \ldots, X_t)$

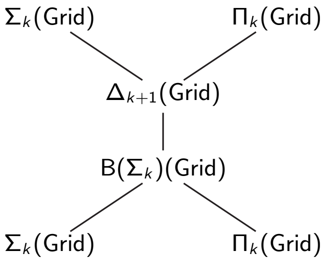
Alternation hierarchy of MSO over grids and graphs

- The signature of grids is given by $\tau_{Grid} = ([m,n], S_1^{m,n}, S_2^{m,n})$ where $[m,n] = [m] \times [n]$
- The signature of t-bit grids for some $t \in \mathbb{N}$ is given by $\tau_{t-Grid} = ([m,n], S_1^{m,n}, S_2^{m,n}, X_1, \ldots, X_t)$

Theorem

$\forall\ k \geq 1,\ B(\Sigma_k)(Grids) \subsetneq \Delta_{k+1}(Grids)$

The inclusion results are as shown the diagram in the right with undirected edges indicating strict inclusion.



$B(\Sigma_k)(\text{Grids}) \subsetneq \Delta_{k+1}(\text{Grids})$

The basis of the theorem is definability results for sets of grids.

- For a function f :$\mathbb{N} \to \mathbb{N}$ we denote by L_f the set of grids whose size is (m,f(m)) for m $\geq$ 1.

$B(\Sigma_k)(\text{Grids}) \subsetneq \Delta_{k+1}(\text{Grids})$

The basis of the theorem is definability results for sets of grids.

- For a function f $:\mathbb{N} \rightarrow \mathbb{N}$ we denote by L_f the set of grids whose size is (m,f(m)) for m $\geq$ 1.
- A formula ϕ over τ_{Grid} defines the function f$:\mathbb{N} \rightarrow \mathbb{N}$ iff $\text{Mod}_{Grid}(\phi) = L_f$.

$B(\Sigma_k)(\text{Grids}) \subsetneq \Delta_{k+1}(\text{Grids})$

The basis of the theorem is definability results for sets of grids.

- For a function f :$\mathbb{N} \rightarrow \mathbb{N}$ we denote by L_f the set of grids whose size is (m,f(m)) for m $\geq$ 1.
- A formula ϕ over τ_{Grid} defines the function f:$\mathbb{N} \rightarrow \mathbb{N}$ iff $Mod_{Grid}(\phi) = L_f$.
- A function is at most k-fold exponential if f(m) is $s_k(O(m))$,
  where $s_0(m) = m$ and $s_{k+1}(m) = 2^{s_k(m)}$.

$B(\Sigma_k)(\text{Grids}) \subsetneq \Delta_{k+1}(\text{Grids})$

The basis of the theorem is definability results for sets of grids.

- For a function f :$\mathbb{N} \rightarrow \mathbb{N}$ we denote by L_f the set of grids whose size is (m,f(m)) for $m \geq 1$.
- A formula ϕ over τ_{Grid} defines the function f:$\mathbb{N} \rightarrow \mathbb{N}$ iff $Mod_{Grid}(\phi) = L_f$.
- A function is at most k-fold exponential if f(m) is $s_k(O(m))$,
  where $s_0(m) = m$ and $s_{k+1}(m) = 2^{s_k(m)}$.

Theorem

Every $B(\Sigma_k)$-definable function is at most k-fold exponential.

$B(\Sigma_k)(\text{Grids}) \subsetneq \Delta_{k+1}(\text{Grids})$

The basis of the theorem is definability results for sets of grids.

- For a function f $:\mathbb{N} \to \mathbb{N}$ we denote by L_f the set of grids whose size is (m,f(m)) for m ≥ 1.
- A formula ϕ over τ_{Grid} defines the function f$:\mathbb{N} \to \mathbb{N}$ iff $Mod_{Grid}(\phi) = L_f$.
- A function is at most k-fold exponential if f(m) is $s_k(O(m))$,
  where $s_0(m) = m$ and $s_{k+1}(m) = 2^{s_k(m)}$.

Theorem

Every $B(\Sigma_k)$-definable function is at most k-fold exponential.

Theorem

Let $f_1(m) = 2^m$, $f_{k+1}(m) = f_k(m)2^{f_k(m)}$ for m,k $\geq$ 1.
$\forall k \geq 1$,the function f_k is definable in Σ_k and Π_k over τ_{Grid}

Complexity of a $B(\Sigma_k)$-definable function

Proof Sketch:

- For a picture language L over alphabet Γ and an integer m$\geq$ 1, we denote by L(m) the word language L restricted to $\Gamma^{m,1}$.

Complexity of a $B(\Sigma_k)$-definable function

Proof Sketch:

- For a picture language L over alphabet Γ and an integer m$\geq$ 1, we denote by L(m) the word language L restricted to $\Gamma^{m,1}$.
- $\forall$ t $\geq$ 0 and for every $\phi \in \Sigma_1$ with free variables among X_1, ... ,X_t $\exists$c$\geq$1 such that for all m$\geq$1, there is an NFA with 2^{cm} states that recognises the word language $\text{Mod}_t(\phi)$(m) over $(\{0,1\}^t)^{m,1}$.

Complexity of a $B(\Sigma_k)$-definable function

Proof Sketch:

- For a picture language L over alphabet Γ and an integer $m \geq 1$, we denote by L(m) the word language L restricted to $\Gamma^{m,1}$.
- $\forall$ $t \geq 0$ and for every $\phi \in \Sigma_1$ with free variables among X_1, ... ,X_t $\exists c \geq 1$ such that for all $m \geq 1$, there is an NFA with 2^{cm} states that recognises the word language $Mod_t(\phi)(m)$ over $(\{0,1\}^t)^{m,1}$.
- $\forall$ $k \geq 1$ and for every $\phi \in \Sigma_k$ with free variables among X_1, ... ,X_t $\exists c \geq 1$ such that for all $m \geq 1$, there is an NFA with $s_k(cm)$ states that recognises the word language $Mod_t(\phi)(m)$ over $(\{0,1\}^t)^{m,1}$.

Complexity of a $B(\Sigma_k)$-definable function

Proof Sketch:

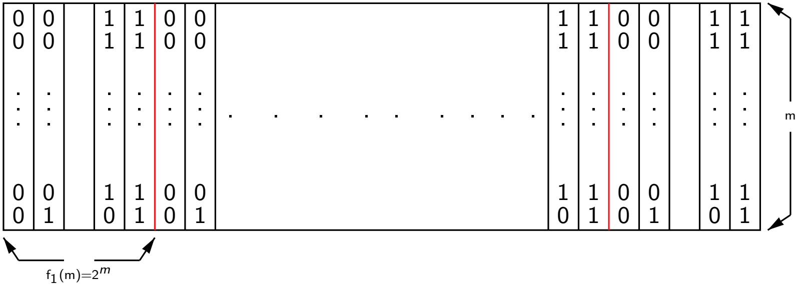
- For a picture language L over alphabet Γ and an integer m$\geq$ 1, we denote by L(m) the word language L restricted to $\Gamma^{m,1}$.
- $\forall$ t $\geq$ 0 and for every $\phi \in \Sigma_1$ with free variables among X_1, ... ,X_t $\exists$c$\geq$1 such that for all m$\geq$1, there is an NFA with 2^{cm} states that recognises the word language $Mod_t(\phi)(m)$ over $(\{0,1\}^t)^{m,1}$.
- $\forall$ k $\geq$ 1 and for every $\phi \in \Sigma_k$ with free variables among X_1, ... ,X_t $\exists$c$\geq$1 such that for all m$\geq$1, there is an NFA with $s_k(cm)$ states that recognises the word language $Mod_t(\phi)(m)$ over $(\{0,1\}^t)^{m,1}$.
- Let N $\subseteq \mathbb{N}$ be recognizable by some n-state NFA. Then $\exists$ k $\leq (n+2)^2$ and an integer p such that N is recognized by a DFA A with states 0,...,k+(p-1) such that A reaches the state k+((l-k)mod p) after reading an input of length l$\geq$k.($\implies$ N is $(n+2)^2$-periodic).

Complexity of a $B(\Sigma_k)$-definable function

Proof Sketch:

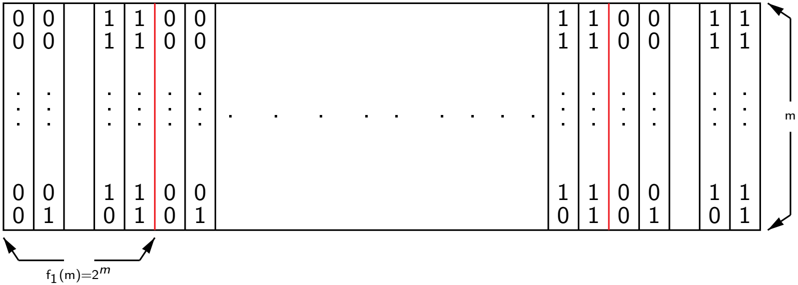
- For a picture language L over alphabet Γ and an integer $m \geq 1$, we denote by L(m) the word language L restricted to $\Gamma^{m,1}$.
- $\forall$ $t \geq 0$ and for every $\phi \in \Sigma_1$ with free variables among $X_1, \ldots, X_t$ $\exists c \geq 1$ such that for all $m \geq 1$, there is an NFA with 2^{cm} states that recognises the word language $Mod_t(\phi)(m)$ over $(\{0,1\}^t)^{m,1}$.
- $\forall$ $k \geq 1$ and for every $\phi \in \Sigma_k$ with free variables among $X_1, \ldots, X_t$ $\exists c \geq 1$ such that for all $m \geq 1$, there is an NFA with $s_k(cm)$ states that recognises the word language $Mod_t(\phi)(m)$ over $(\{0,1\}^t)^{m,1}$.
- Let $N \subseteq \mathbb{N}$ be recognizable by some n-state NFA. Then $\exists$ $k \leq (n+2)^2$ and an integer p such that N is recognized by a DFA A with states $0,\ldots,k+(p-1)$ such that A reaches the state $k+((l-k) \bmod p)$ after reading an input of length $l \geq k$.($\implies$ N is $(n+2)^2$-periodic).
- Let ϕ be a $B(\Sigma_k)$-sentence. There is a constant $c \geq 1$ such that for every $m \geq 1$ the set $Mod_0(\phi)(m)$ is $s_k(cm)$-periodic. Thus from all the above statements we get that every $B(\Sigma_k)$-definable function is at most k-fold exponential.

$f_k(m) \in \Delta_k(\text{Grid})$

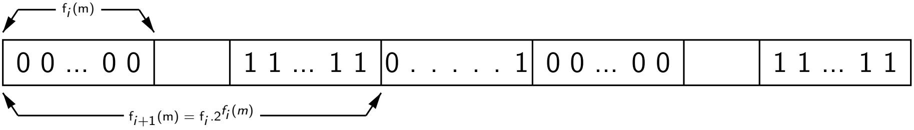


The Complete column-numbering of a grid

$f_k(m) \in \Delta_k(Grid)$



The Complete column-numbering of a grid

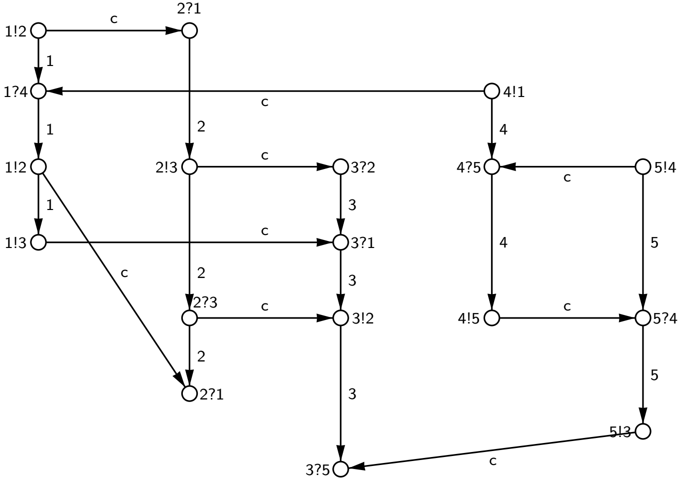


Complete f_i-numbering along the top row of the grid

MSCs, MPA and EMSO_{MSC}

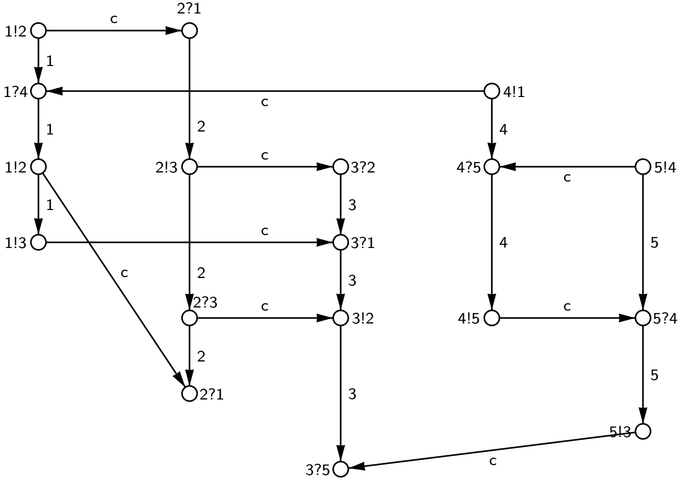
MSCs, MPA and EMSO$_{MSC}$

Message Sequence Charts :-



MSCs, MPA and EMSO$_{MSC}$

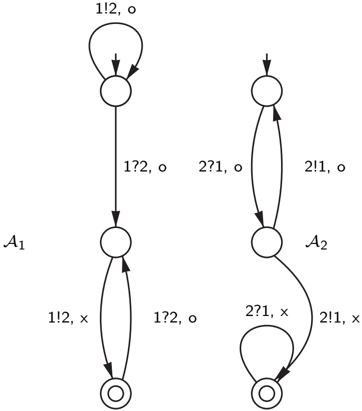
Message Sequence Charts :-



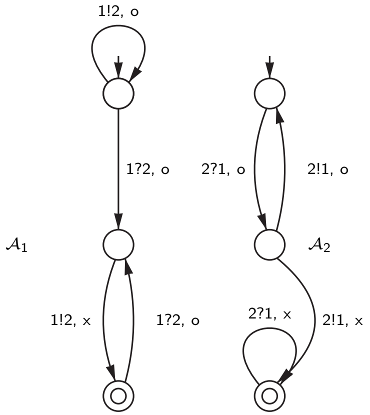
MSC (over P) is a graph M $= (E, \{\Delta_p\}_{p \in P}, \Delta_c, \lambda)$

MSCs, MPA and EMSO$_{MSC}$

Message Passing Automata :-



Message Passing Automata :-



MPA (over P) is a structure $(A) = (((A)_p)_{p \in P}, \mathcal{D}, s^{-in}, F)$

Definitions

Definition

An MSC (over P) is a graph $M = (E,\{\Delta_p\}_{p \in P},\Delta_c,\lambda) \in \mathbb{DG}(Act,P_c)$ such that

- Δ_p is a total order.
- $\Delta_c \subseteq E \times E$ is the set of edges connecting messages.
- Number of messages sent is equal to the number of messages received

Definitions

Definition

An MSC (over P) is a graph M $= (\text{E},\{\Delta_p\}_{p\in P},\Delta_c,\lambda) \in \mathbb{DG}(\text{Act},\text{P}_c)$ such that

- Δ_p is a total order.
- $\Delta_c \subseteq \text{E} \times \text{E}$ is the set of edges connecting messages.
- Number of messages sent is equal to the number of messages received

Definition

MPA (over P) is a structure $(A) = (((A)_p)_{p\in P},\mathcal{D},\text{s}^{-in},\text{F})$ such that

- $\mathcal{D}$ is a set of synchronization messages.
- for each p∈P ,$\mathcal{A}$ is a pair (S_p, δ_p) where
  - S_p is a set of local states
  - $\delta_p \subseteq S_p \times \text{Act}_p \times \mathcal{D} \times S_p$
- $\text{s}^{-in} \in \Pi_{p\in P}\text{S}_p$ is the global initial state.
- $\text{F} \subseteq \Pi_{p\in P}\text{S}_p$ is the set of global final states.

Logical Definability

Definition (MSO(Σ,C))

MSO(Σ,C) over the class $\mathbb{DG}$ are built up from the atomic formulas $\lambda(x) = a$ (for $a \in \Sigma$), $x\Delta_c y$ (for $c \in C$), $x \in X$ and $x = y$.

Logical Definability

Definition (MSO(Σ,C))

MSO(Σ,C) over the class $\mathbb{DG}$ are built up from the atomic formulas λ(x) $=$ a (for a $\in \Sigma$), xΔ_cy (for c$\in$ C), x$\in$ X and x=y.

Definition

A graph acceptor over (Σ,C) is a structure $\mathcal{B} = (\mathcal{Q}, \mathcal{R}, \widehat{S}$,Occ) where

- $\mathcal{Q}$ is its nonempty finite set of states
- $\mathcal{B} \in \mathbb{N}$ is the radius
- $\widehat{S}$ is a finite set of R-spheres over ($\Sigma \times \mathcal{Q}$,C) and
- Occ is a boolean combinations of conditions of the form "sphere H $\in \widehat{S}$ occurs at least n times" where n $\in \mathbb{N}$.

The Theorems

Theorem

$MPA \equiv EMSO_{MSC}$

The Theorems

Theorem

$MPA \equiv EMSO_{MSC}$

Theorem

The monadic quantifier alternation hierarchy over MSC is infinite.

Summary

Summary

- Hanf's theorem
- Picture languages
  - Local picture Languages
  - Recognizable picture languages
  - REC $\iff$ EMSO
- Monadic quantifier Alternation hierarchy over Grids and graphs
- MSC $\iff$ EMSO_{MSC}