On a Class of Rational Functions for Pictures

František Mráz[1] Friedrich Otto[2] Daniel Průša[3]

[1]Charles University, Prague, Czech Republic

[2]Universität Kassel, Kassel, Germany

[3]Czech Technical University, Prague, Czech Republic

NCMA 2015

Porto

August 31 – September 1, 2015

1. Introduction

2. Pictures and Picture Languages

A picture P over Σ is a finite two-dimensional array of symbols from Σ.
$\mathrm{row}(P)$ ($\mathrm{col}(P)$) denotes the number of rows (columns) of P,
$P_{i,j}$ is the symbol at position (i,j), $1 \leq i \leq \mathrm{row}(P)$, $1 \leq j \leq \mathrm{col}(P)$.

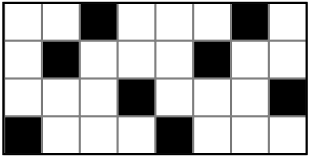


Figure: A sample picture of size $(4, 8)$ over $\Sigma = \{\square, \blacksquare\}$.

By $\Sigma^{m,n}$ we denote the set of all pictures of size (m, n) over Σ,
Λ is the only picture of size $(0, 0)$, and $\Sigma^{*,*}$ is the set of pictures over Σ.

2. Pictures and Picture Languages

A picture P over Σ is a finite two-dimensional array of symbols from Σ.
$\mathrm{row}(P)$ ($\mathrm{col}(P)$) denotes the number of rows (columns) of P,
$P_{i,j}$ is the symbol at position (i, j), $1 \leq i \leq \mathrm{row}(P)$, $1 \leq j \leq \mathrm{col}(P)$.

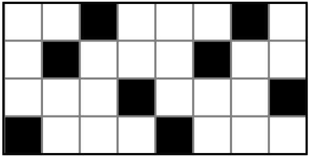


Figure: A sample picture of size $(4, 8)$ over $\Sigma = \{\square, \blacksquare\}$.

By $\Sigma^{m,n}$ we denote the set of all pictures of size (m, n) over Σ,
Λ is the only picture of size $(0, 0)$, and $\Sigma^{*,*}$ is the set of pictures over Σ.

Let $\mathcal{S} = \{\vdash, \dashv, \top, \bot, \#\}$ be a set of five special markers (sentinels).

In order to enable an automaton to detect the border of a picture $P \in \Sigma^{m,n}$ easily, we define the boundary picture $\widehat{P}$ over $\Sigma \cup \mathcal{S}$ of size $(m+2, n+2)$:

$\#$	$\top \; \top \; \cdots \; \top \; \top$	$\#$
$\vdash \; \vdots \; \vdash$	P	$\dashv \; \vdots \; \dashv$
$\#$	$\bot \; \bot \; \cdots \; \bot \; \bot$	$\#$

Figure: The boundary picture $\widehat{P}$.

Definition 1

A deterministic two-dimensional two-way ordered restarting automaton (det-2D-2W-ORWW) *is given through* $M = (Q, \Sigma, \Gamma, \mathcal{S}, q_0, \delta, >)$, *where*

- Q *is a finite set of states containing the initial state* q_0,
- Σ *is a finite input alphabet,* Γ *is a finite tape alphabet containing* Σ *such that* $\Gamma \cap \mathcal{S} = \emptyset$, *and* $>$ *is a partial ordering on* Γ, *and*
- $\delta : Q \times (\Gamma \cup \mathcal{S})^{3,3} \to (Q \times \{\mathrm{R}, \mathrm{D}\}) \cup \Gamma \cup \{\mathrm{Accept}\}$ *satisfies the following restrictions for all* $q \in Q$ *and all* $C \in (\Gamma \cup \mathcal{S})^{3,3}$:
  1. *if* $C_{2,3} = \dashv$, *then* $\delta(q, C) \neq (q', \mathrm{R})$ *for all* $q' \in Q$,
  2. *if* $C_{3,2} = \bot$, *then* $\delta(q, C) \neq (q', \mathrm{D})$ *for all* $q' \in Q$,
  3. *if* $\delta(q, C) = b \in \Gamma$, *then* $C_{2,2} > b$ *with respect to the ordering* $>$.

If $Q = \{q_0\}$, *then* M *is called stateless* (*stl-det-2D-2W-ORWW*).

Definition 1

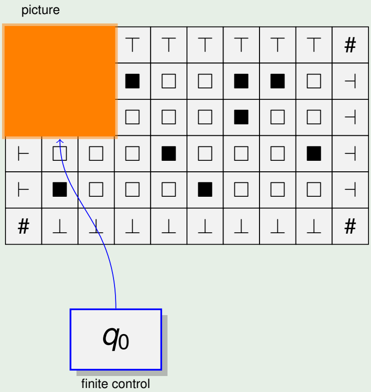
A deterministic two-dimensional two-way ordered restarting automaton (det-2D-2W-ORWW) is given through $M = (Q, \Sigma, \Gamma, \mathcal{S}, q_0, \delta, >)$, *where*

- Q *is a finite set of states containing the initial state* q_0,
- Σ *is a finite input alphabet,* Γ *is a finite tape alphabet containing* Σ *such that* $\Gamma \cap \mathcal{S} = \emptyset$, *and* $>$ *is a partial ordering on* Γ, *and*
- $\delta : Q \times (\Gamma \cup \mathcal{S})^{3,3} \to (Q \times \{\mathrm{R}, \mathrm{D}\}) \cup \Gamma \cup \{\mathsf{Accept}\}$ *satisfies the following restrictions for all* $q \in Q$ *and all* $C \in (\Gamma \cup \mathcal{S})^{3,3}$:
  1. *if* $C_{2,3} = \dashv$, *then* $\delta(q, C) \neq (q', \mathrm{R})$ *for all* $q' \in Q$,
  2. *if* $C_{3,2} = \perp$, *then* $\delta(q, C) \neq (q', \mathrm{D})$ *for all* $q' \in Q$,
  3. *if* $\delta(q, C) = b \in \Gamma$, *then* $C_{2,2} > b$ *with respect to the ordering* $>$.

If $Q = \{q_0\}$, *then* M *is called stateless (stl-det-2D-2W-ORWW).*

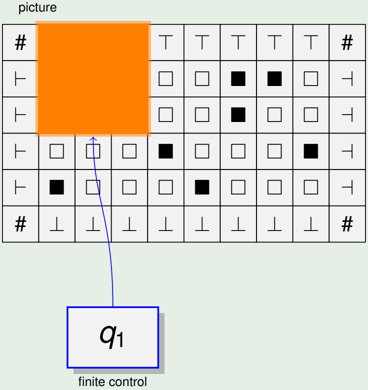
Example:

The possible moves of a det-2D-2W-ORWW-automaton $\mathcal{M}$:



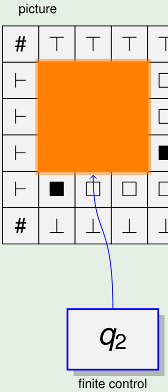
Example:

The possible moves of a det-2D-2W-ORWW-automaton $\mathcal{M}$:



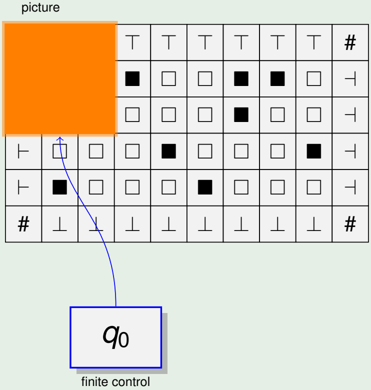
Example:

The possible moves of a det-2D-2W-ORWW-automaton $\mathcal{M}$:



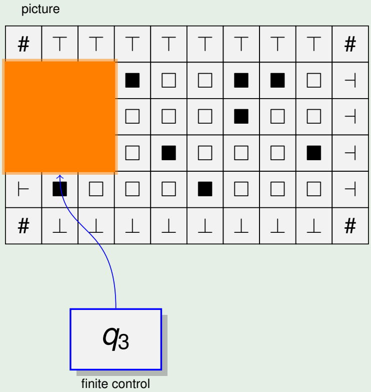
Example:

The possible moves of a det-2D-2W-ORWW-automaton $\mathcal{M}$:



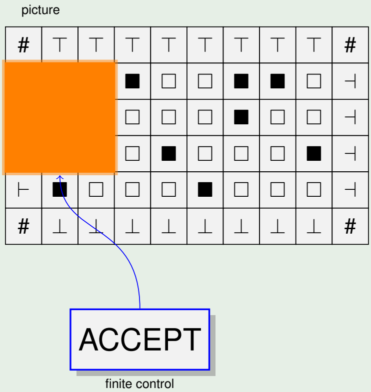
Example:

The possible moves of a det-2D-2W-ORWW-automaton $\mathcal{M}$:



Example:

The possible moves of a det-2D-2W-ORWW-automaton $\mathcal{M}$:



Observation (O. CIAA 2014)

$$\text{DREC} \subsetneq \mathcal{L}(\text{stl-det-2D-2W-ORWW}) \subseteq \mathcal{L}(\text{det-2D-2W-ORWW}) \subsetneq \text{P},$$

where

- DREC denotes the determ. recognizable 2-dim. languages (Anselmo et. al. 2007),
- and P denotes the polynomial-time recognizable languages.

Theorem 2

From a given det-2D-2W-ORWW-automaton $\mathcal{M}$, one can construct a stateless det-2D-2W-ORWW-automaton $\mathcal{M}_0$ such that $L(\mathcal{M}_0) = L(\mathcal{M})$.

Observation (O. CIAA 2014)

$$\text{DREC} \subsetneq \mathcal{L}(\text{stl-det-2D-2W-ORWW}) \subseteq \mathcal{L}(\text{det-2D-2W-ORWW}) \subsetneq \text{P},$$

where

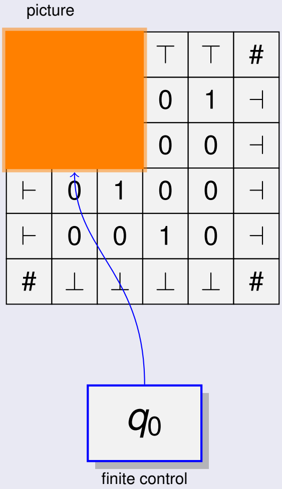
- DREC denotes the determ. recognizable 2-dim. languages (Anselmo et. al. 2007),
- and P denotes the polynomial-time recognizable languages.

Theorem 2

From a given det-2D-2W-ORWW-automaton $\mathcal{M}$, one can construct a stateless det-2D-2W-ORWW-automaton $\mathcal{M}_0$ such that $L(\mathcal{M}_0) = L(\mathcal{M})$.

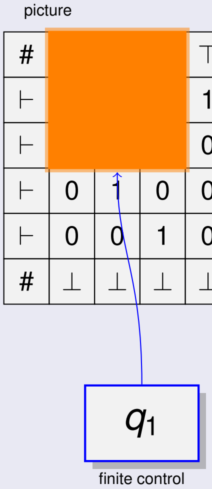
Proof of Theorem 2:

An example computation of a det-2D-2W-ORWW-automaton M:



Proof of Theorem 2:

An example computation of a det-2D-2W-ORWW-automaton M:

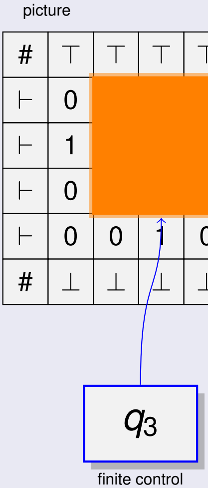


Proof of Theorem 2:

An example computation of a det-2D-2W-ORWW-automaton M:

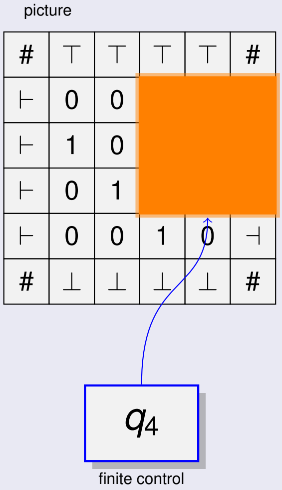
Proof of Theorem 2:

An example computation of a det-2D-2W-ORWW-automaton M:



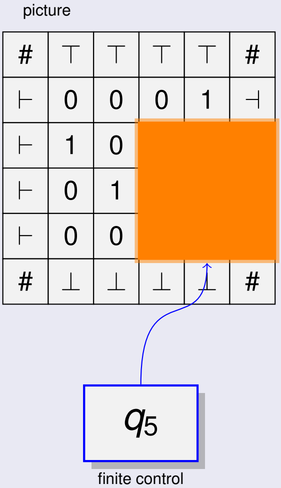
Proof of Theorem 2:

An example computation of a det-2D-2W-ORWW-automaton M:



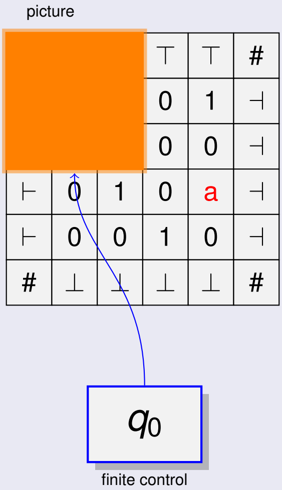
Proof of Theorem 2:

An example computation of a det-2D-2W-ORWW-automaton M:



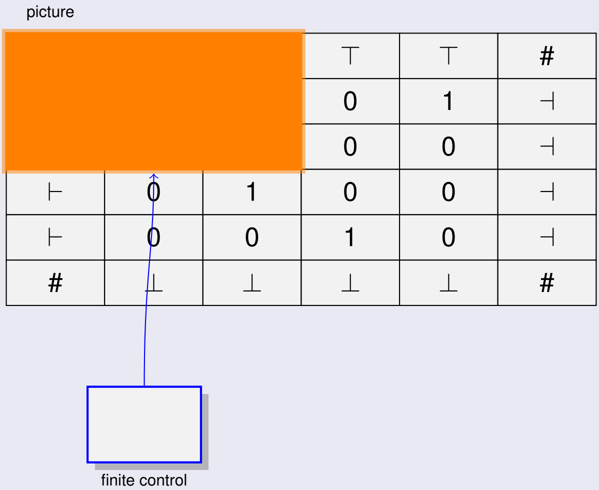
Proof of Theorem 2:

An example computation of a det-2D-2W-ORWW-automaton *M*:



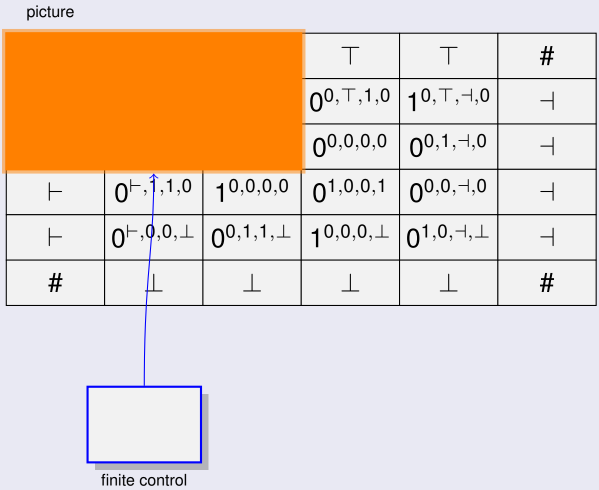
Proof of Theorem 2 (cont.):

Simulation by a stl-det-2D-2W-ORWW-automaton M_0:



Proof of Theorem 2 (cont.):

Simulation by a stl-det-2D-2W-ORWW-automaton M_0:

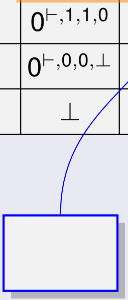


Proof of Theorem 2 (cont.):

Simulation by a stl-det-2D-2W-ORWW-automaton M_0:

picture

#				$\top$	#
$\vdash$				$1^{0,\top,\dashv,0}$	$\dashv$
$\vdash$				$0^{0,1,\dashv,0}$	$\dashv$
$\vdash$	$0^{\vdash,1,1,0}$	$1^{0,0,0,0}$	$0^{1,0,0,1}$	$0^{0,0,\dashv,0}$	$\dashv$
$\vdash$	$0^{\vdash,0,0,\bot}$	$0^{0,1,1,\bot}$	$1^{0,0,0,\bot}$	$0^{1,0,\dashv,\bot}$	$\dashv$
#	$\bot$	$\bot$	$\bot$	$\bot$	#



finite control

Proof of Theorem 2 (cont.):

Simulation by a stl-det-2D-2W-ORWW-automaton M_0:

picture

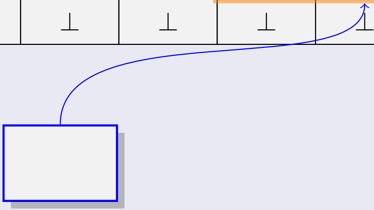
#	$\top$	$\top$	$\top$	$\top$	#
$\vdash$	$0^{\vdash,\top,0,1}_{q_0}$	$0^{0,\top,0,0}_{q_1}$			
$\vdash$	$1^{\vdash,0,0,0}$	$0^{1,0,0,1}$			
$\vdash$	$0^{\vdash,1,1,0}$	$1^{0,0,0,0}$			
$\vdash$	$0^{\vdash,0,0,\bot}$	$0^{0,1,1,\bot}$	$1^{0,0,0,\bot}$	$0^{1,0,\dashv,\bot}$	$\dashv$
#	$\bot$	$\bot$	$\bot$	$\bot$	#

Proof of Theorem 2 (cont.):

Simulation by a stl-det-2D-2W-ORWW-automaton M_0:

picture

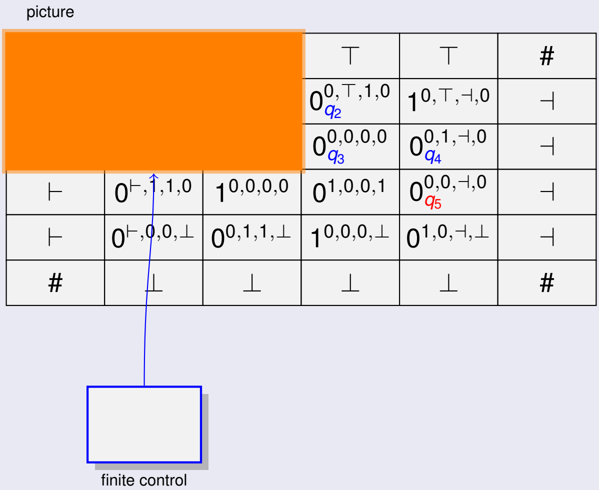
#	$\top$	$\top$	$\top$	$\top$	#
$\vdash$	$0^{\vdash,\top,0,1}_{q_0}$	$0^{0,\top,0,0}_{q_1}$	$0^{0,\top,1,0}_{q_2}$	$1^{0,\top,\dashv,0}$	$\dashv$
$\vdash$	$1^{\vdash,0,0,0}$	$0^{1,0,0,1}$			
$\vdash$	$0^{\vdash,1,1,0}$	$1^{0,0,0,0}$			
$\vdash$	$0^{\vdash,0,0,\bot}$	$0^{0,1,1,\bot}$			
#	$\bot$	$\bot$	$\bot$	$\bot$	#



finite control

Proof of Theorem 2 (cont.):

Simulation by a stl-det-2D-2W-ORWW-automaton M_0:

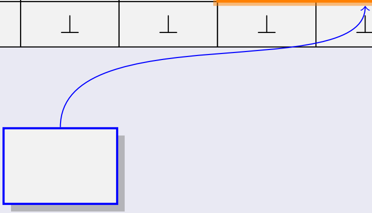


Proof of Theorem 2 (cont.):

Simulation by a stl-det-2D-2W-ORWW-automaton M_0:

picture

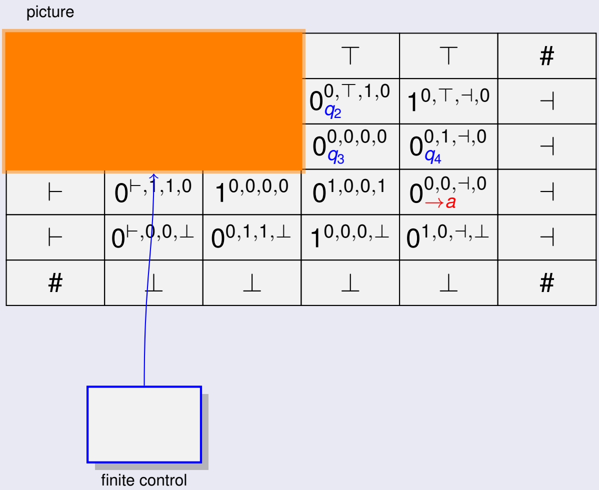
#	$\top$	$\top$	$\top$	$\top$	#
$\vdash$	$0^{\vdash,\top,0,1}_{q_0}$	$0^{0,\top,0,0}_{q_1}$	$0^{0,\top,1,0}_{q_2}$	$1^{0,\top,\dashv,0}$	$\dashv$
$\vdash$	$1^{\vdash,0,0,0}$	$0^{1,0,0,1}$			
$\vdash$	$0^{\vdash,1,1,0}$	$1^{0,0,0,0}$			
$\vdash$	$0^{\vdash,0,0,\bot}$	$0^{0,1,1,\bot}$			
#	$\bot$	$\bot$	$\bot$	$\bot$	#



finite control

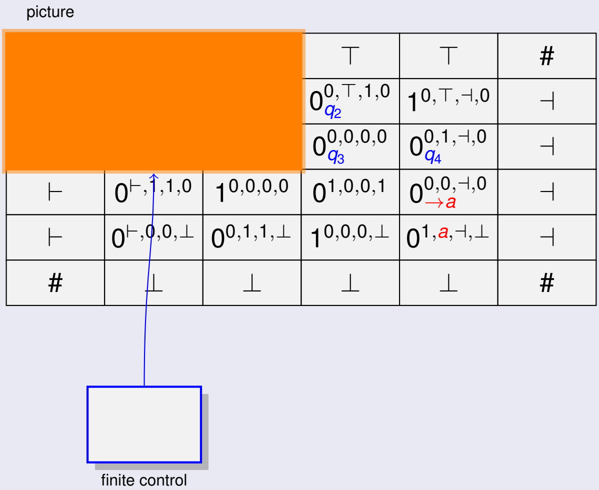
Proof of Theorem 2 (cont.):

Simulation by a stl-det-2D-2W-ORWW-automaton M_0:



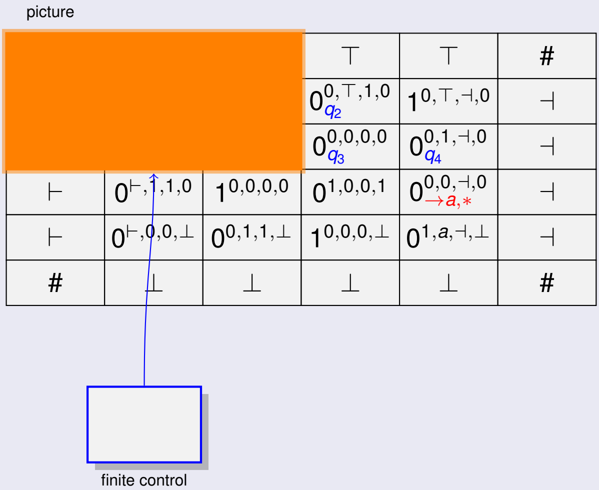
Proof of Theorem 2 (cont.):

Simulation by a stl-det-2D-2W-ORWW-automaton M_0:



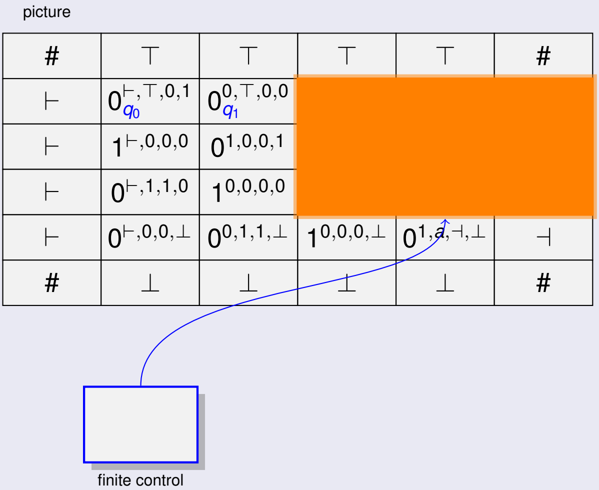
Proof of Theorem 2 (cont.):

Simulation by a stl-det-2D-2W-ORWW-automaton M_0:



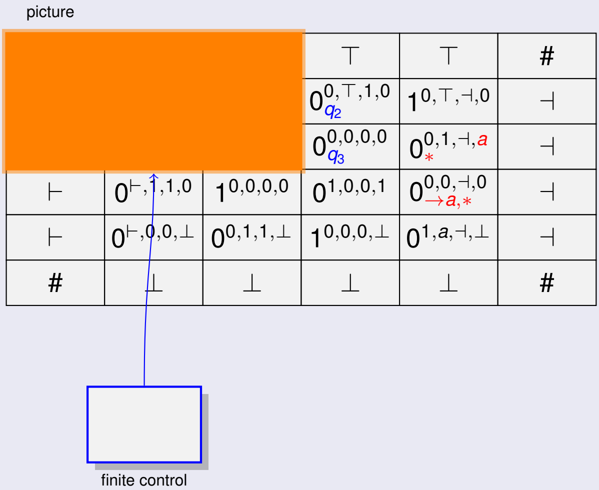
Proof of Theorem 2 (cont.):

Simulation by a stl-det-2D-2W-ORWW-automaton M_0:



Proof of Theorem 2 (cont.):

Simulation by a stl-det-2D-2W-ORWW-automaton M_0:

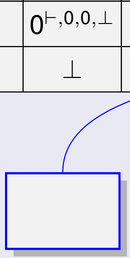


Proof of Theorem 2 (cont.):

Simulation by a stl-det-2D-2W-ORWW-automaton M_0:

picture

#	$\top$	$\top$	$\top$	$\top$	#
$\vdash$	$0^{\vdash,\top,0,1}_{q_0}$	$0^{0,\top,0,0}_{q_1}$			
$\vdash$	$1^{\vdash,0,0,0}$	$0^{1,0,0,1}$			
$\vdash$	$0^{\vdash,1,1,0}$	$1^{0,0,0,0}$			
$\vdash$	$0^{\vdash,0,0,\bot}$	$0^{0,1,1,\bot}$	$1^{0,0,0,\bot}$	$0^{1,a,\dashv,\bot}$	$\dashv$
#	$\bot$	$\bot$	$\bot$	$\bot$	#



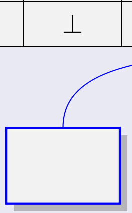
finite control

Proof of Theorem 2 (cont.):

Simulation by a stl-det-2D-2W-ORWW-automaton M_0:

picture

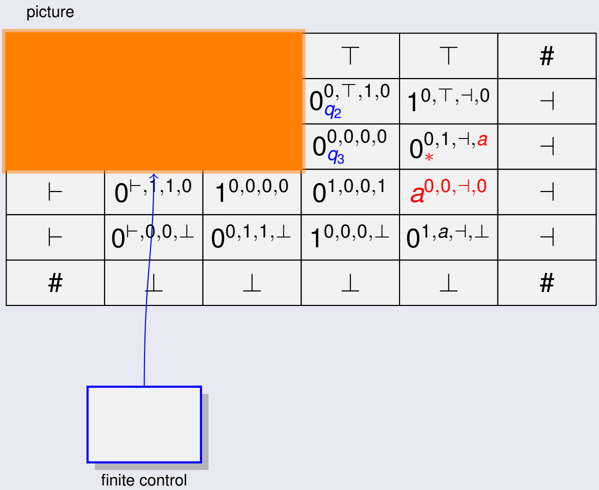
#	$\top$	$\top$	$\top$	$\top$	#
$\vdash$	$0^{\vdash,\top,0,1}_{q_0}$	$0^{0,\top,0,0}_{q_1}$	$0^{0,\top,1,0}_{q_2}$	$1^{0,\top,\dashv,0}$	$\dashv$
$\vdash$	$1^{\vdash,0,0,0}$	$0^{1,0,0,1}$			
$\vdash$	$0^{\vdash,1,1,0}$	$1^{0,0,0,0}$			
$\vdash$	$0^{\vdash,0,0,\bot}$	$0^{0,1,1,\bot}$			
#	$\bot$	$\bot$	$\bot$	$\bot$	#



finite control

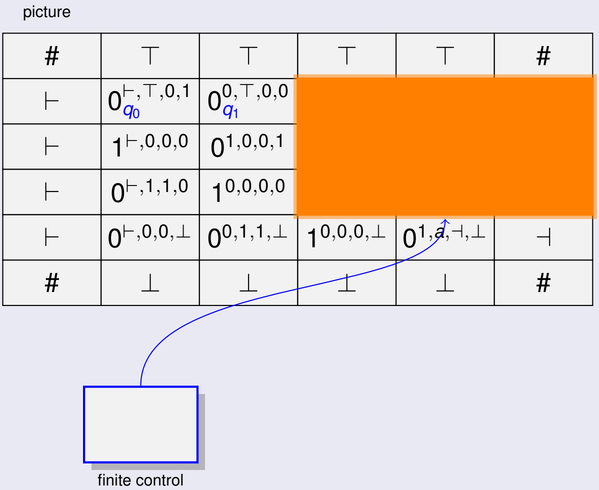
Proof of Theorem 2 (cont.):

Simulation by a stl-det-2D-2W-ORWW-automaton M_0:



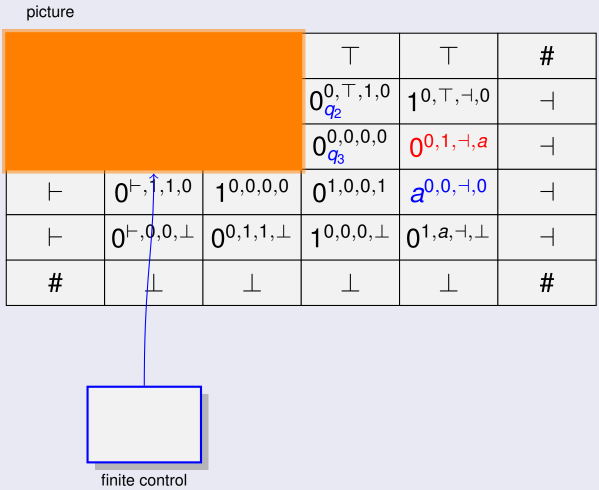
Proof of Theorem 2 (cont.):

Simulation by a stl-det-2D-2W-ORWW-automaton M_0:



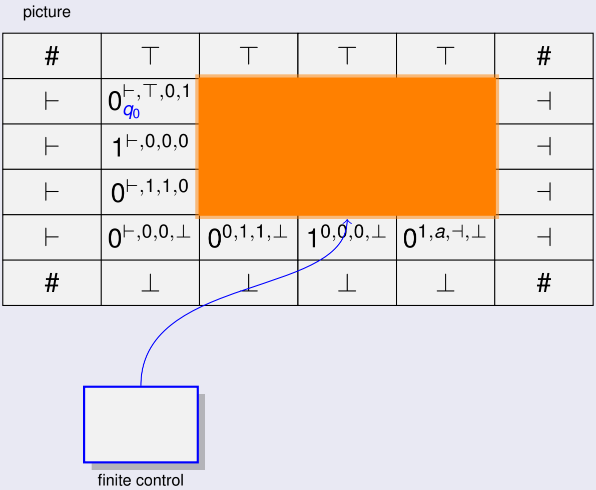
Proof of Theorem 2 (cont.):

Simulation by a stl-det-2D-2W-ORWW-automaton M_0:



Proof of Theorem 2 (cont.):

Simulation by a stl-det-2D-2W-ORWW-automaton M_0:



3. Computing Transductions

Definition 3

(a) *A homogeneous morphism from $\Gamma^{*,*}$ into $\Delta^{*,*}$ is defined by two integers $b, h \geq 1$ and a mapping $\varphi : \Gamma \to \Delta^{h,b}$. Then φ extends to a morphism $\varphi : \Gamma^{*,*} \to \Delta^{*,*}$ that maps a picture $P \in \Gamma^{m,n}$ into a picture $\varphi(P) \in \Delta^{m \cdot h, n \cdot b}$.*

(b) *Let $\mathcal{M} = (Q, \Sigma, \Gamma, \mathcal{S}, q_0, \delta, >)$ be a det-2D-2W-ORWW-automaton, let Δ be a finite (output) alphabet, and let $\varphi : \Gamma^{*,*} \to \Delta^{*,*}$ be a homogeneous morphism.*
For $P \in L(\mathcal{M})$, $\widetilde{P}$ denotes the final tape inscription that $\mathcal{M}$ produces during its accepting computation on input P.
With P we associate the output picture $\varphi(\widetilde{P})$.
Thus, $(\mathcal{M}, \varphi)$ defines a transduction $\varphi_{\mathcal{M}} : L(\mathcal{M}) \to \Delta^{*,*}$.

3. Computing Transductions

Definition 3

(a) *A homogeneous morphism from* $\Gamma^{*,*}$ *into* $\Delta^{*,*}$ *is defined by two integers* $b, h \geq 1$ *and a mapping* $\varphi : \Gamma \to \Delta^{h,b}$*. Then* φ *extends to a morphism* $\varphi : \Gamma^{*,*} \to \Delta^{*,*}$ *that maps a picture* $P \in \Gamma^{m,n}$ *into a picture* $\varphi(P) \in \Delta^{m \cdot h, n \cdot b}$*.*

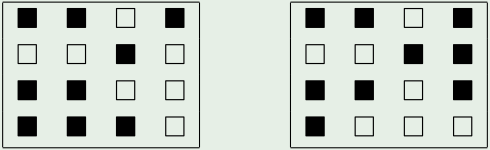
(b) *Let* $\mathcal{M} = (Q, \Sigma, \Gamma, \mathcal{S}, q_0, \delta, >)$ *be a det-2D-2W-ORWW-automaton, let* Δ *be a finite (output) alphabet, and let* $\varphi : \Gamma^{*,*} \to \Delta^{*,*}$ *be a homogeneous morphism.*
For $P \in L(\mathcal{M})$, $\widetilde{P}$ *denotes the final tape inscription that* $\mathcal{M}$ *produces during its accepting computation on input* P*.*
With P *we associate the output picture* $\varphi(\widetilde{P})$*.*
Thus, $(\mathcal{M}, \varphi)$ *defines a transduction* $\varphi_{\mathcal{M}} : L(\mathcal{M}) \to \Delta^{*,*}$*.*

Example:

Let $\Sigma = \{\square, \blacksquare\}$ and $L_{\mathrm{sq}} = \{ P \in \Sigma^{*,*} \mid \mathrm{rows}(P) = \mathrm{cols}(P) \geq 1 \}$.
We define a transformation τ on L_{sq} as follows:

$$\tau(P) = Q \in \Sigma^{*,*}, \quad \text{where } Q_{\mathrm{rows}(P),i} = P_{i,\mathrm{cols}(P)},$$
$$Q_{i,\mathrm{cols}(P)} = P_{\mathrm{rows}(P),i},\ 1 \leq i \leq \mathrm{rows}(P), \text{ and}$$
$$Q_{i,j} = P_{i,j},\ 1 \leq i,j < \mathrm{rows}(P),$$

that is, $\tau(P)$ is obtained from P by interchanging the last column with the last row, leaving all other entries untouched.



(a) (b)

Figure: An example picture P_0 from L_{sq} (a) and the picture $\tau(P_0)$ (b).

Example (cont.):

We present a det-2D-2W-ORWW-automaton $\mathcal{M}_\tau = (Q, \Sigma, \Gamma, \mathcal{S}, q_0, \delta, >)$ and a morphism $\varphi : \Gamma^{*,*} \to \Sigma^{*,*}$ such that $(\mathcal{M}_\tau, \varphi)$ realizes τ.
We take $\Gamma = \Sigma \cup \{ a, [a, b]_i \mid a, b \in \Sigma, 1 \leq i \leq 5 \}$, we define a partial ordering $>$ by taking $a > [a, b]_1 > [a, c]_2 > [a, d]_3 > [a, e]_4 > [a, f]_5$ for all $a, b, c, d, e, f \in \Sigma$, and we define the transition function δ in such a way that $\mathcal{M}_\tau$ proceeds as follows given a picture $P \in \Sigma^{n,n}$ as input:

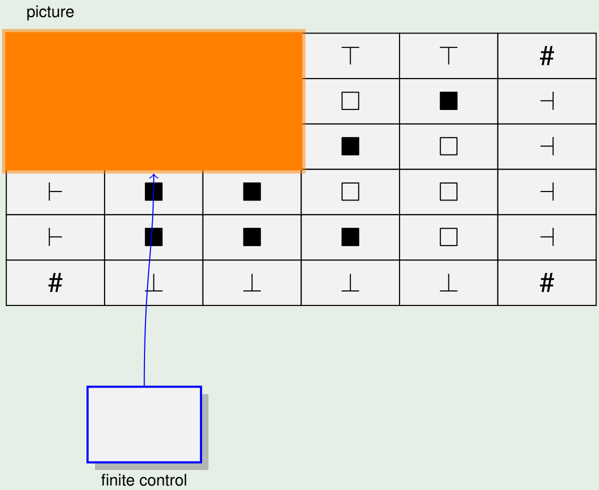
1. The information on the last column is moved to the main diagonal.
2. Then this information is moved to the bottom row.
3. The information on the bottom row is moved to the main diagonal.
4. Then this information is moved to the last column.

The morphism φ is defined as follows:

$$\varphi : [a, b]_2 \to b, [a, b]_3 \to a, [a, b]_4 \to a, [a, b]_5 \to b.$$

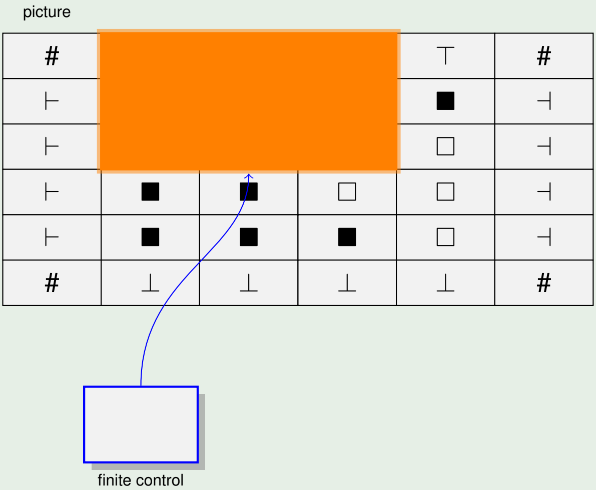
Example (cont.):

Computation of the det-2D-2W-ORWW-automaton $\mathcal{M}_\tau$:



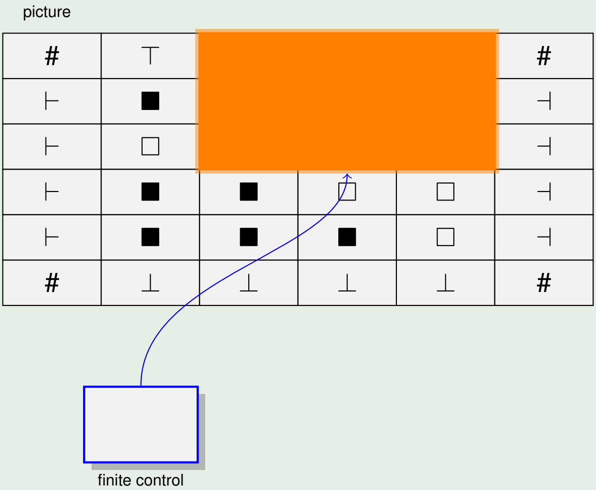
Example (cont.):

Computation of the det-2D-2W-ORWW-automaton $\mathcal{M}_\tau$:



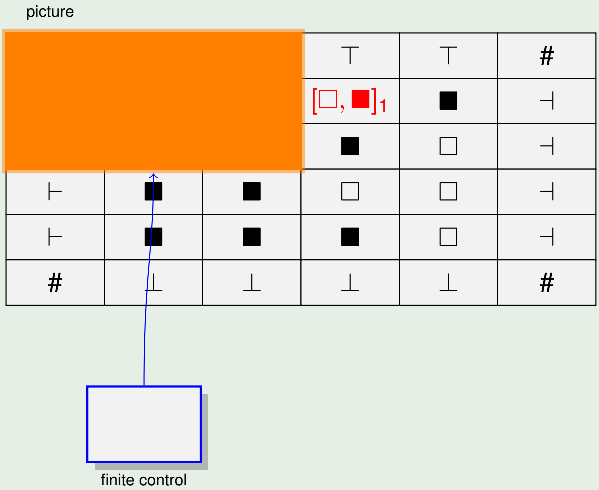
Example (cont.):

Computation of the det-2D-2W-ORWW-automaton $\mathcal{M}_\tau$:



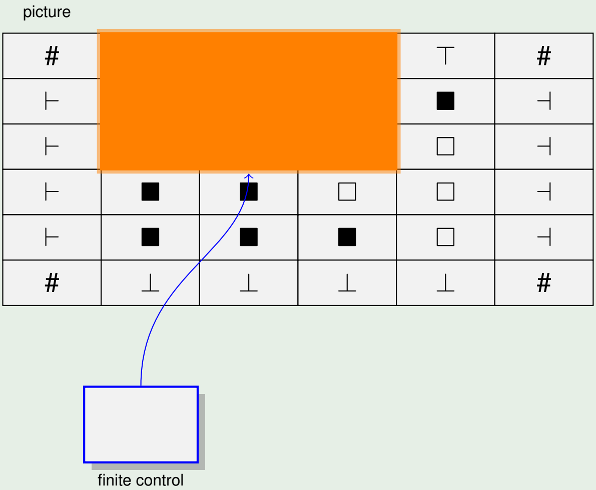
Example (cont.):

Computation of the det-2D-2W-ORWW-automaton $\mathcal{M}_\tau$:



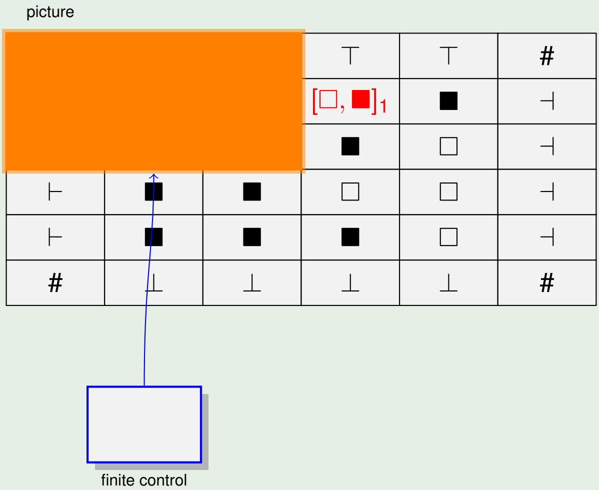
Example (cont.):

Computation of the det-2D-2W-ORWW-automaton $\mathcal{M}_\tau$:



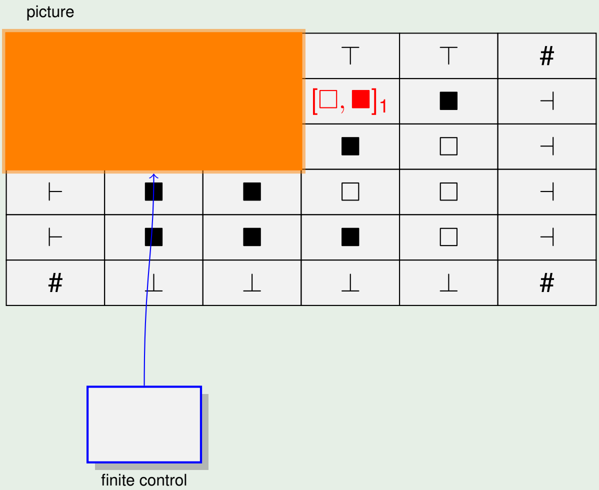
Example (cont.):

Computation of the det-2D-2W-ORWW-automaton $\mathcal{M}_\tau$:



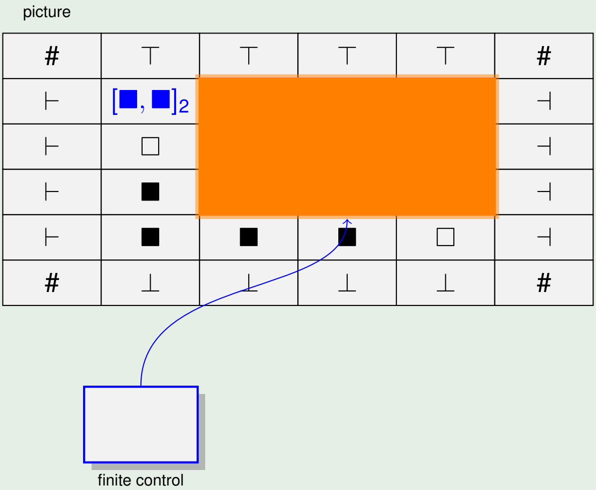
Example (cont.):

Computation of the det-2D-2W-ORWW-automaton $\mathcal{M}_\tau$:



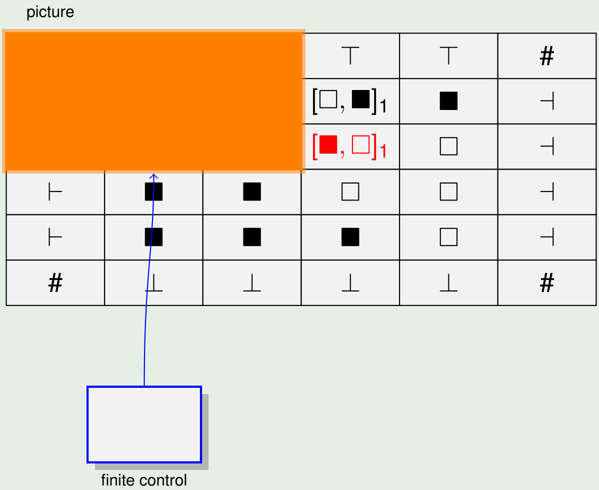
Example (cont.):

Computation of the det-2D-2W-ORWW-automaton $\mathcal{M}_\tau$:



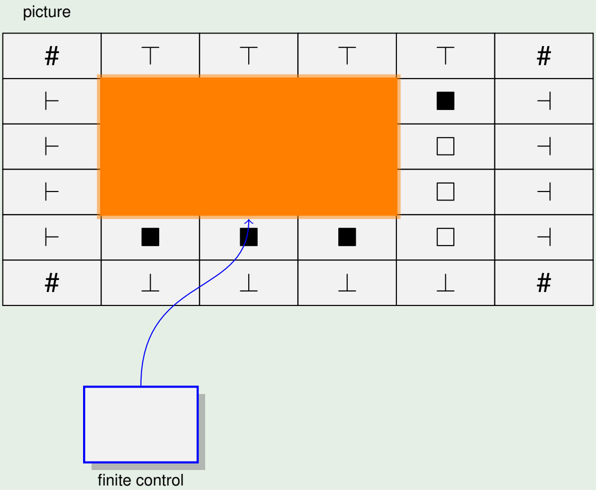
Example (cont.):

Computation of the det-2D-2W-ORWW-automaton $\mathcal{M}_\tau$:



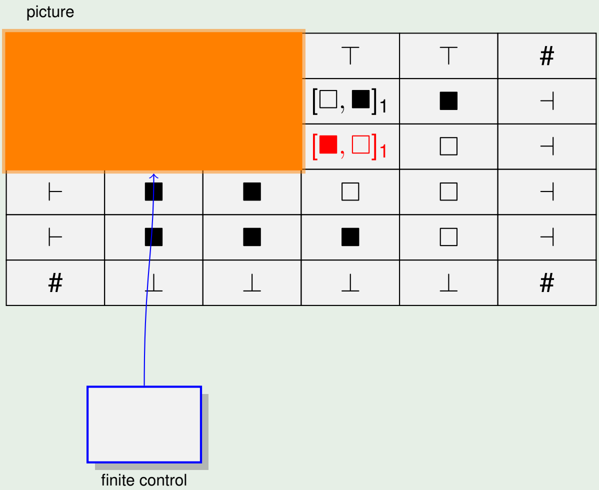
Example (cont.):

Computation of the det-2D-2W-ORWW-automaton $\mathcal{M}_\tau$:



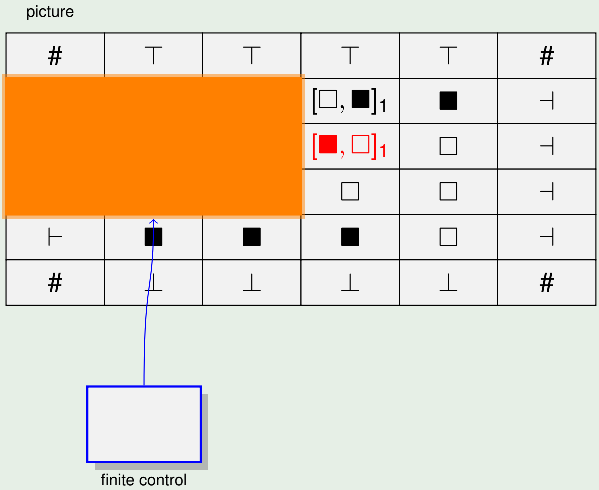
Example (cont.):

Computation of the det-2D-2W-ORWW-automaton $\mathcal{M}_\tau$:



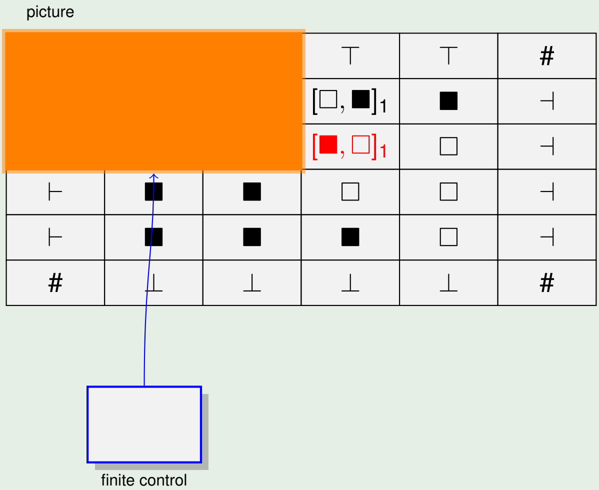
Example (cont.):

Computation of the det-2D-2W-ORWW-automaton $\mathcal{M}_\tau$:



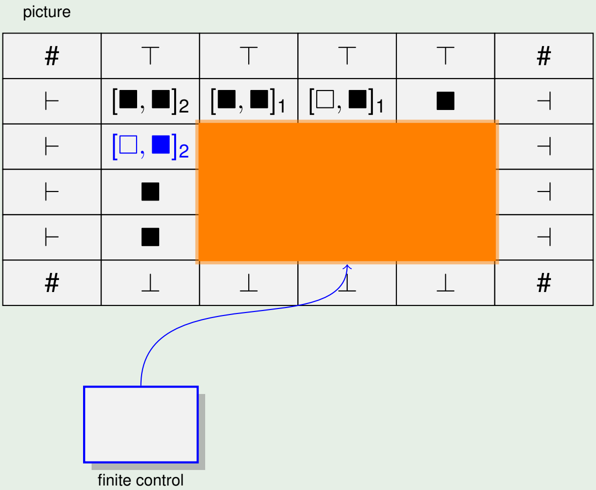
Example (cont.):

Computation of the det-2D-2W-ORWW-automaton $\mathcal{M}_\tau$:



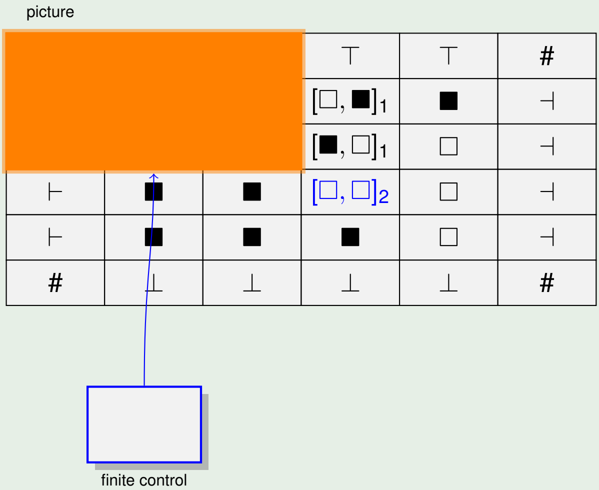
Example (cont.):

Computation of the det-2D-2W-ORWW-automaton $\mathcal{M}_\tau$:



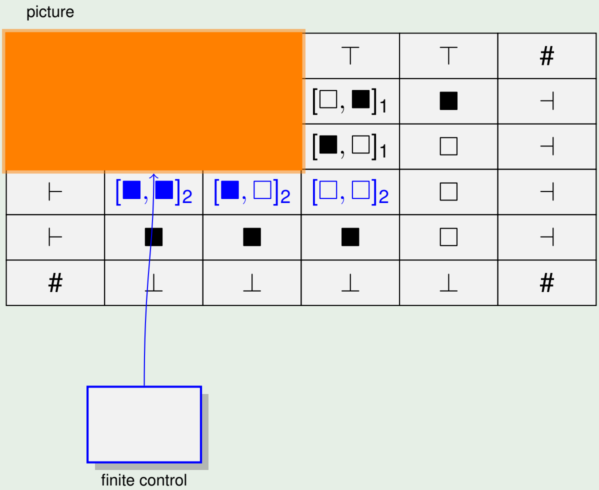
Example (cont.):

Computation of the det-2D-2W-ORWW-automaton $\mathcal{M}_\tau$:



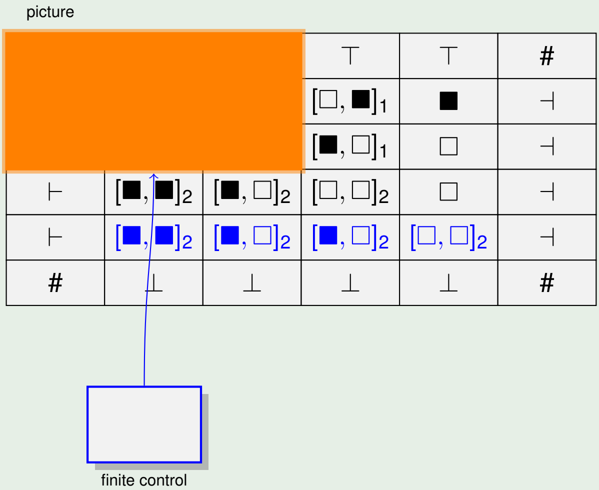
Example (cont.):

Computation of the det-2D-2W-ORWW-automaton $\mathcal{M}_\tau$:



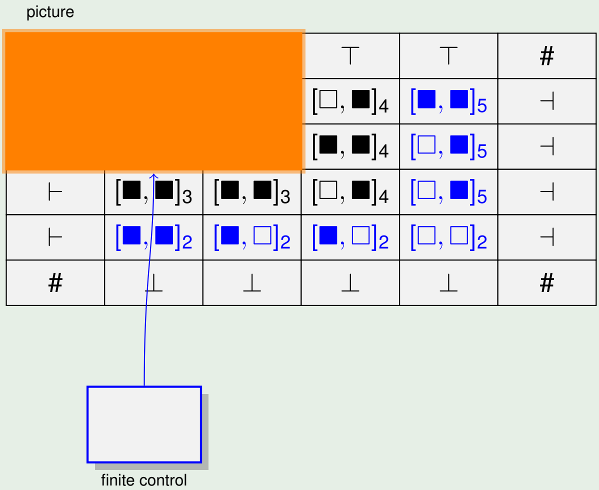
Example (cont.):

Computation of the det-2D-2W-ORWW-automaton $\mathcal{M}_\tau$:



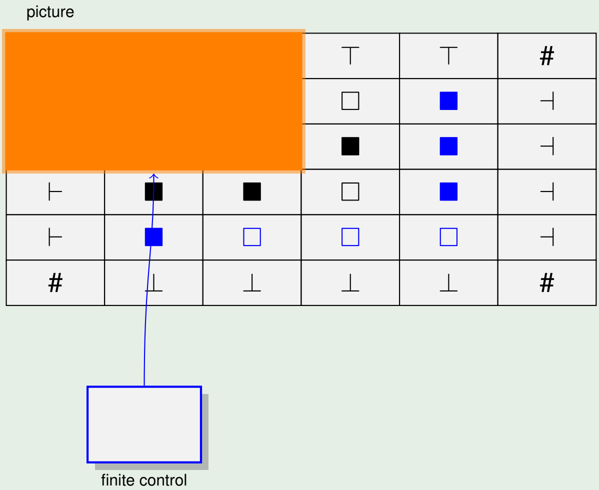
Example (cont.):

Computation of the det-2D-2W-ORWW-automaton $\mathcal{M}_\tau$:



Example (cont.):

Computation of the det-2D-2W-ORWW-automaton $\mathcal{M}_\tau$:



4. On the Class $\mathcal{F}$(det-2D-2W-ORWW)

5. Decision Problems

6. Conclusion

Thank you for your attention!