Degrees of Restriction for Two-Dimensional Automata

CIAA 2021

Taylor J. Smith

Joint work with K. Salomaa

School of Computing
Queen's University
Kingston, Ontario, Canada

July 21, 2021

Table of Contents

Table of Contents

Two-Dimensional Automata

- A two-dimensional (2D) automaton is a generalization of a one-dimensional automaton.
- Two major differences:
  1. Different input word
  2. Different transition function

- A two-dimensional (2D) automaton is a generalization of a one-dimensional automaton.
- Two major differences:
  1. **Different input word**
  2. Different transition function

$$
\begin{matrix}
\# & \# & \# & \cdots & \# & \# \\
\# & a_{1,1} & a_{1,2} & \cdots & a_{1,n} & \# \\
\# & a_{2,1} & a_{2,2} & \cdots & a_{2,n} & \# \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
\# & a_{m,1} & a_{m,2} & \cdots & a_{m,n} & \# \\
\# & \# & \# & \cdots & \# & \#
\end{matrix}
$$

- A two-dimensional (2D) automaton is a generalization of a one-dimensional automaton.
- Two major differences:
  1. Different input word
  2. **Different transition function**

$$\delta : (Q \setminus q_{\text{accept}}) \times (\Sigma \cup \{\#\}) \to Q \times \{U, D, L, R\}$$

Deterministic four-way (2DFA-4W)

$$\delta : (Q \setminus q_{\text{accept}}) \times (\Sigma \cup \{\#\}) \to 2^{Q \times \{U,D,L,R\}}$$

Nondeterministic four-way (2NFA-4W)

Restricted 2D Automata

- 2D automata do not have to be four-way automata.
- Restrict the transition function to get:
  - Three-way (3W) automata: $\{D, L, R\}$
  - Two-way (2W) automata: $\{D, R\}$
- Three-way automata cannot return to a row after moving downward, but they can read symbols multiple times in a row.
- Two-way automata are "read-once".
  - Similar to a one-way one-dimensional automaton.

Allowing Restricted Moves

- If we permit a bounded number of restricted moves, we get a new family of 2D automata.
- We call these models **extended 2D automata**.

Allowing Restricted Moves

- If we permit a bounded number of restricted moves, we get a new family of 2D automata.
- We call these models extended 2D automata.
  - ***i*-extended three-way** 2D automaton:
    Makes at most i upward moves.

Allowing Restricted Moves

- If we permit a bounded number of restricted moves, we get a new family of 2D automata.
- We call these models extended 2D automata.
  - *i*-extended three-way 2D automaton:
    Makes at most *i* upward moves.
  - **(i, j)-extended two-way** 2D automaton:
    Makes at most *i* upward moves and at most *j* leftward moves.

Allowing Restricted Moves

- If we permit a bounded number of restricted moves, we get a new family of 2D automata.
- We call these models extended 2D automata.
  - i-extended three-way 2D automaton:
    Makes at most i upward moves.
  - (i, j)-extended two-way 2D automaton:
    Makes at most i upward moves and at most j leftward moves.
- Morita et al. (2003) studied the similar notion of **input head reversal-bounded 2D TMs**.
  - I.e., 2D TMs whose input heads may switch vertical direction of movement some bounded number of times.

Notation

Two-Dimensional Automata

2DFA-nW

dimension of input word

\# of input head moves

Notation

Two-Dimensional Automata

2DFA-nW

dimension of input word

# of input head moves

Extended Two-Dimensional Automata

2DFA-3W[i] 2DFA-2W[i, j]

number of upward moves

number of leftward moves

Table of Contents

Recognition Properties

- First motivation: to examine relationships between 2D automata and their extended variants.
- Does an additional (constant) number of moves give additional recognition power?

Recognition Properties

- First motivation: to examine relationships between 2D automata and their extended variants.
- Does an additional (constant) number of moves give additional recognition power?
- Yes. . . we can induce an **infinite** hierarchy of inclusions!

Theorem

2NFA-3W[0] $\subset$ 2NFA-3W[1].

Proof

Let $\Sigma = \{0, 1\}$. Define

$$L_1 = \{\text{two-row 2D words } w \text{ with at least two columns } j \text{ where } w[1,j] = w[2,j]\}.$$

$$\begin{matrix} \# & \# & \# & \# & \# & \# & \# & \# & \# & \# & \# & \# & \# \\ \# & * & \cdots & * & 1 & * & \cdots & * & 1 & * & \cdots & * & \# \\ \# & * & \cdots & * & 1 & * & \cdots & * & 1 & * & \cdots & * & \# \\ \# & \# & \# & \# & \# & \# & \# & \# & \# & \# & \# & \# & \# \end{matrix}$$

"Stacked 1s"

Theorem

2NFA-3W[0] $\subset$ 2NFA-3W[1].

Proof (cont'd)

$\mathcal{A} \in$ 2NFA-3W[1] recognizes words in L_1:

- Move rightward and nondeterministically select an occurrence of 1.
- Move downward and check that the symbol read is 1.
- Move rightward and nondeterministically select another occurrence of 1.
- Move upward and check that the symbol read is 1.

Theorem

2NFA-3W[0] $\subset$ 2NFA-3W[1].

Proof (cont'd)

$\mathcal{A}' \in$ 2NFA-3W[0] cannot recognize words in L_1:

- Let $u(i,j,z)$, $1 \leq i < j \leq z$, be a length-z 1D word where $u[i] = u[j] = 1$ and all other symbols are 0.
- Let $w(i,j,z) = u(i,j,z) \ominus u(i,j,z)$.
- For large enough z, $\mathcal{A}'$ accepts all words $w(i,j,z)$.
- We choose two computations $C(i,j,z)$ and $C(r,s,z)$ where $\mathcal{A}'$ moves downward to the second row in the same column and state. (Note: $i \neq r$ or $j \neq s$.)
- However, $\mathcal{A}'$ then accepts the word $u(i,j,z) \ominus u(r,s,z) \notin L_1$!

Recognition Properties

CIAA 2021 BREMEN

- The general proof technique is evident:
  1. Choose a witness language.
  2. Show how the "stronger" model can recognize the language.
  3. Show how the "weaker" model cannot recognize the language.
- I.e., a natural extension of the standard fooling set argument.
- Other separations follow by similar proofs.

Theorem

2NFA-3W[i] $\subset$ 2NFA-3W[$i+1$] for all $i \geq 2$.

Proof Sketch

Take L_1 and create a family of languages L_i, $i \geq 2$, by taking i copies of L_1 and concatenating words row-wise.
Each word in L_i consists of $2i$ rows where rows $2j$ and $2j+1$, $0 \leq j < i$, contain at least two occurrences of “stacked 1s”.

Corollary

For all $i \geq 1$,

$$2\text{NFA-3W} \subset \cdots \subset 2\text{NFA-3W}[i] \subset 2\text{NFA-3W}[i{+}1] \subset \cdots \subset 2\text{NFA-4W}.$$

Using different witness languages, we get:

Theorem

For all $i \geq 1$,

$$2\text{DFA-3W} \subset \cdots \subset 2\text{DFA-3W}[i] \subset 2\text{DFA-3W}[i{+}1] \subset \cdots \subset 2\text{DFA-4W}.$$

Theorem

2NFA-2W$[0,0] \subset$ 2NFA-2W$[1,0]$.

Proof

Use the same language L_1.

- $\mathcal{C} \in$ 2NFA-2W$[1,0]$ recognizes words in L_1 in the same way as $\mathcal{A} \in$ 2NFA-3W$[1]$.
- $\mathcal{C}' \in$ 2NFA-2W$[0,0]$ cannot recognize words in L_1: for example, the accepting computation may be the same for both

$$\begin{matrix} 1 & 1 \\ 1 & 1 \end{matrix} \quad \text{and} \quad \begin{matrix} 1 & 0 \\ 1 & 1 \end{matrix}\,.$$

Theorem

2NFA-2W$[i, 0] \subset$ 2NFA-2W$[i+1, 0]$ for all $i \geq 2$.

Proof Sketch

Let $\Sigma = \{0, 1\}$. Define

$$K_{i+1} = \{\text{two-row 2D words } u \text{ such that } u \text{ contains at least } 2i+2 \text{ occurrences of “stacked 1s”}\}.$$

Corollary

For all $i \geq 1$,

$$\text{2NFA-2W} \subset \cdots \subset \text{2NFA-2W}[i, 0] \subset \text{2NFA-2W}[i{+}1, 0] \subset \cdots \subset \text{2NFA-3W}^{\circlearrowleft}.$$

Using different witness languages, we get:

Theorem

For all $i \geq 1$,

$$\text{2DFA-2W} \subset \cdots \subset \text{2DFA-2W}[i, 0] \subset \text{2DFA-2W}[i{+}1, 0] \subset \cdots \subset \text{2DFA-3W}^{\circlearrowleft}.$$

There are analogous hierarchies for 2NFA-2W$[0, i]$/2DFA-2W$[0, i]$.

Determinism vs. Nondeterminism

- Second motivation: to examine relationships between deterministic and nondeterministic models of extended 2D automata.
- From past work, we know that 2DFA-3W $\subset$ 2NFA-3W.
- Does a similar relationship exist for extended three-way 2D automata?

Determinism vs. Nondeterminism

- Second motivation: to examine relationships between deterministic and nondeterministic models of extended 2D automata.
- From past work, we know that 2DFA-3W $\subset$ 2NFA-3W.
- Does a similar relationship exist for extended three-way 2D automata?
- No. . . the models are **incomparable**!

Lemma

There exists a language recognized by an automaton $\mathcal{M} \in$ 2DFA-3W[1] and not by any automaton $\mathcal{N} \in$ 2NFA-3W[0].

Proof Sketch

Let $\Sigma = \{0, 1\}$. Define

$$M_1 = \{\text{two-row 2D words } w \text{ with exactly two columns } j \text{ where } w[1, j] = w[2, j], \text{ and all other symbols are } 0\}.$$

(Note: M_1 is similar to L_1 but with *exactly* two “stacked 1s” instead of *at least* two.)

Lemma

There exists a language recognized by an automaton $\mathcal{P} \in$ 2NFA-3W[0] and not by any automaton $\mathcal{Q} \in$ 2DFA-3W[1].

Proof

Let $\Sigma = \{0, 1\}$. Define

$$N_1 = \{\text{two-row 2D words } w \text{ with at least one column } j \text{ where } w[1, j] = w[2, j]\}.$$

Create another language N_2 by concatenating two copies of N_1 row-wise.

Lemma

There exists a language recognized by an automaton $\mathcal{P} \in$ 2NFA-3W[0] and not by any automaton $\mathcal{Q} \in$ 2DFA-3W[1].

Proof (cont'd)

$\mathcal{P} \in$ 2NFA-3W[0] recognizes words in N_2 via a straightforward procedure.

Lemma

There exists a language recognized by an automaton $\mathcal{P} \in$ 2NFA-3W[0] and not by any automaton $\mathcal{Q} \in$ 2DFA-3W[1].

Proof (cont'd)

$\mathcal{Q} \in$ 2DFA-3W[1] cannot recognize words in N_2:

- We show this by proving an intermediate claim: no automaton $\mathcal{Q}' \in$ 2DFA-3W[0] can recognize words in N_1.
- Thus, $\mathcal{Q}$ can't correctly verify the first two rows of its input word without moving upward.
- After using its single upward move, $\mathcal{Q}$ can't correctly verify the last two rows of its input word.

- What about other numbers of moves?
  - We have generalized the argument to find a language recognized by 2DFA-3W[$i + 1$] and not by 2NFA-3W[i] for all $i \geq 1$.
  - A different language (or approach) would be needed to obtain the generalization in the other direction.
- What about deterministic vs. nondeterministic extended two-way 2D automata?
- Some opportunities for future work...

Table of Contents

Closure Properties

- Three-way and four-way 2D automaton models share certain closure properties.
- Both 2NFA-3W and 2NFA-4W are closed under union.
- Both 2DFA-3W and 2DFA-4W are closed under complement.
- What is the status of union/complement closure for our "in between" models?

Closure Properties: Union

Theorem

For all $i \geq 1$, the class 2NFA-3W$[i]$ is closed under union.

Proof

Take the set of automata recognizing each word in the union. Nondeterministically choose which automaton to use on a given input word.

Theorem

For all $i \geq 1$, the class 2DFA-3W[i] is closed under complement.

Proof

If $\mathcal{A} \in$ 2DFA-3W[i] halts on every input word, then swap accepting and non-accepting states to get the complement automaton $\mathcal{A}'$.

Theorem

For all $i \geq 1$, the class 2DFA-3W[i] is closed under complement.

Proof (cont'd)

Otherwise, $\mathcal{A}$ loops infinitely on some input word.

- We must show that $\mathcal{A}'$ can simulate $\mathcal{A}$ while ensuring its input head moves upward/downward or halts in its current row.
- This is done using a modified form of a construction given in Theorem 1 of Szepietowski (1992).

Closure Properties

- What about other operations?
  - Both 2NFA-3W and 2NFA-4W are closed under reversal (row reflection).
  - A number of operations are closed for either three- or four-way 2D automata, but not both.
- Some opportunities for future work...

Table of Contents

Conclusions

- Restricting the input head movement of a 2D automaton restricts its recognition power.
- Allowing a constant number of restricted moves gives recognition power that falls "in between" models.
- Hierarchies exist for both three-way and two-way extended 2D automata, one each for deterministic and nondeterministic.
- Deterministic and nondeterministic extended three-way 2D automata are incomparable.
- Nondeterministic extended three-way 2D automata are closed under union, while the deterministic model is closed under complement.

Future Work

- Does some kind of two-way "sub-hierarchy" result if we modify the numbers of permitted upward/leftward moves simultaneously, rather than separately?
- What kind of relationship exists between 2DFA-2W[i, j] and 2NFA-2W[i, j]?
- Questions of closure status for other operations/models persist.
- What happens if we modify our three-way model to only switch its vertical input head direction at most some constant k number of times?
  - This model would be more closely related to the 2D TM model studied by Morita et al. (2003).

References

[1] M. Morita, K. Inoue, A. Ito, and Y. Wang. Some properties on input head reversal-bounded two-dimensional Turing machines. *IEICE Trans. Inf. Syst.*, E86-D(2):201–212, 2003.

[2] T. J. Smith and K. Salomaa. Degrees of restriction for two-dimensional automata. In S. Maneth, editor, *Proc. of CIAA 2021*, volume 12803 of *LNCS*, pages 77–89, Berlin Heidelberg, 2021. Springer-Verlag.

[3] A. Szepietowski. Some remarks on two-dimensional finite automata. *Inf. Sci.*, 63(1–2):183–189, 1992.