One-deterministic-counter automata

https://arxiv.org/abs/2301.13456

Prince Mathew

*School of Mathematics & Computer Science
Indian Institute of Technology Goa, India
prince@iitgoa.ac.in[1]

Joint work with Dr. Prakash Saivasan, Dr. Sreejith A.V., and Dr. Vincent Penelle

One-deterministic-counter automata (ODCA)

Semantic definition

An OCA where all runs of a word lead to the same counter value.

One-deterministic-counter automata (ODCA)

Semantic definition

An OCA where all runs of a word lead to the same counter value.

Syntactic definition

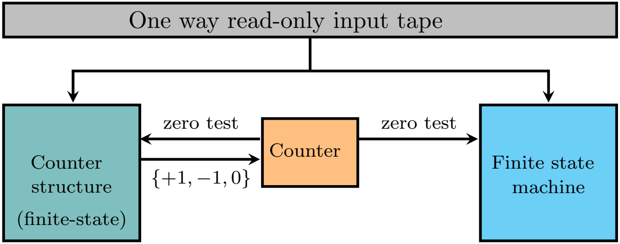


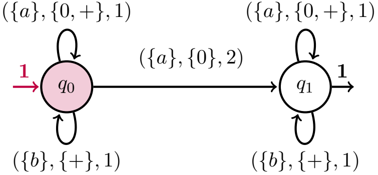
Figure: One-deterministic-counter automata

Example - Weighted ODCA (visibly)

Example

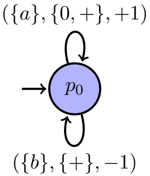
$$\texttt{equalPrefix}(w) = \begin{cases} \mathbf{2} \cdot \mathbf{k}, \text{where } k \text{ is the \#proper prefixes of } w \text{ with} \\ \qquad \#a\text{'s} = \ \#b\text{'s, if } \#a\text{'s} \geq \#b\text{'s for all prefixes.} \\ \mathbf{0}, \ \text{otherwise} \end{cases}$$

.



Finite state machine

Counter structure



a	a	b	b	a	b	a

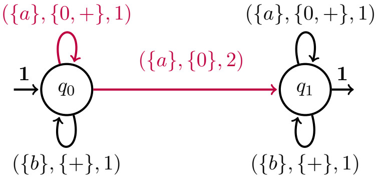
Input tape

0

Counter

1	0

Initial vector



Finite state machine

a	a	b	b	a	b	a

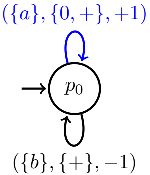
Input tape

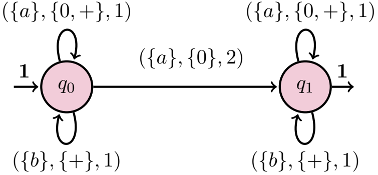
0

Counter

1	0

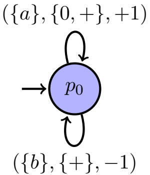
Counter structure





Finite state machine

Counter structure



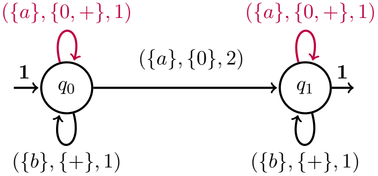
a	a	b	b	a	b	a

Input tape

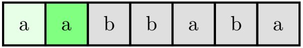
1

Counter

1	2

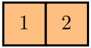


Finite state machine

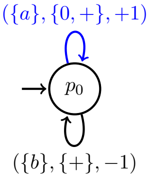


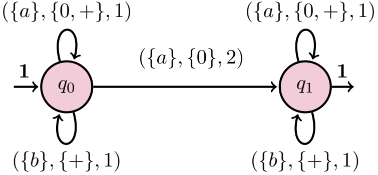
Input tape

Counter



Counter structure





Finite state machine

a	a	b	b	a	b	a

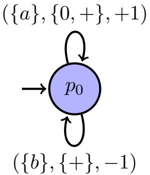
Input tape

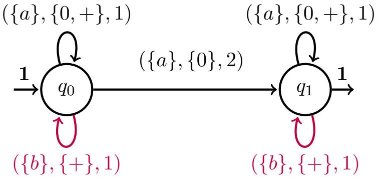
2

Counter

1	2

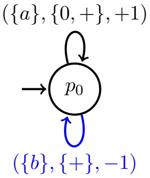
Counter structure





Finite state machine

Counter structure



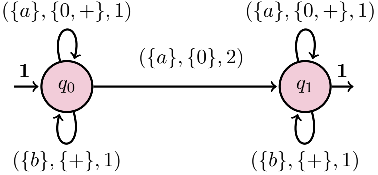
a	a	b	b	a	b	a

Input tape

2

Counter

1	2



Finite state machine

a	a	b	b	a	b	a

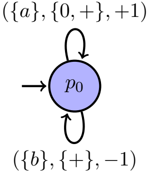
Input tape

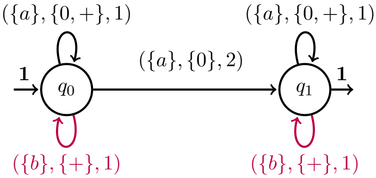
1

Counter

1	2

Counter structure





Finite state machine

a	a	b	b	a	b	a

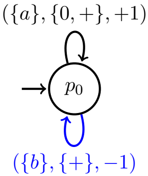
Input tape

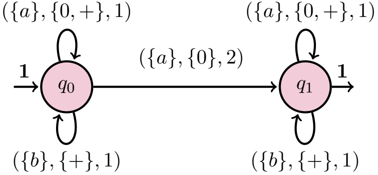
1

Counter

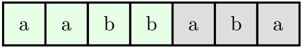
1	2

Counter structure

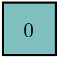




Finite state machine

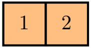


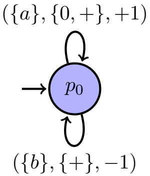
Input tape

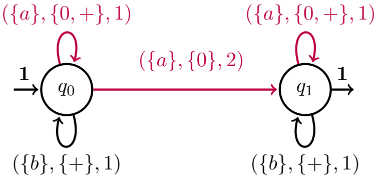


Counter

Counter structure

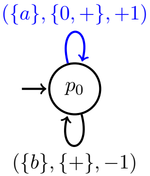






Finite state machine

Counter structure



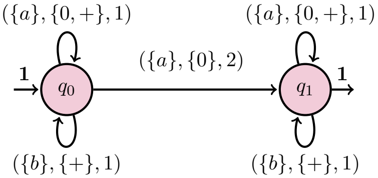
a	a	b	b	a	b	a

Input tape

0

Counter

1	2



Finite state machine

a	a	b	b	a	b	a

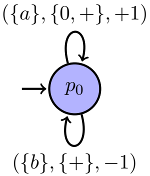
Input tape

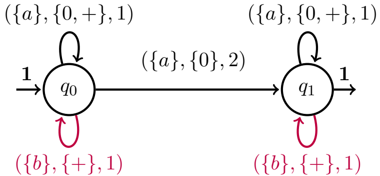
1

Counter

1	4

Counter structure





Finite state machine

a	a	b	b	a	b	a

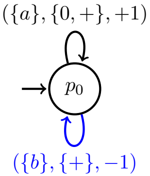
Input tape

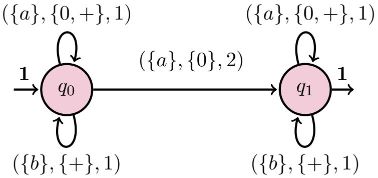
1

Counter

1	4

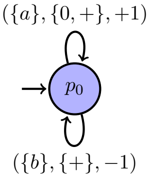
Counter structure





Finite state machine

Counter structure



a	a	b	b	a	b	a

Input tape

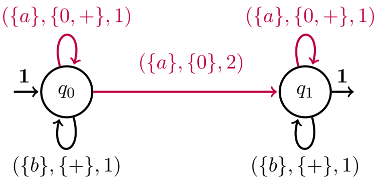
0

Counter

1	4

a	a	b	b	a	b	a

Input tape



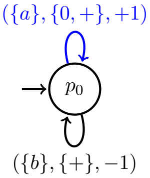
Finite state machine

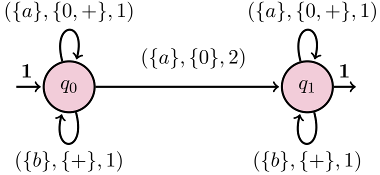
0

Counter

1	4

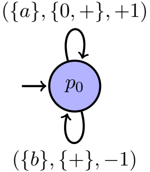
Counter structure





Finite state machine

Counter structure



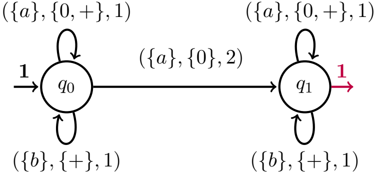
a	a	b	b	a	b	a

Input tape

1

Counter

1	6



Finite state machine

a	a	b	b	a	b	a

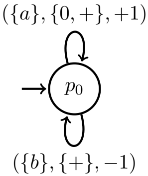
Input tape

1

Counter

1	6

Counter structure



Motivation

- Decidability of equivalence of pPDA is an open problem (Forejt et al., 2014).
- Equivalence of probabilistic OCA is also not known.
- Probabilistic ODCA is a class of probabilistic OCA for which equivalence is decidable.
- This is a strict super class of visibly probabilistic OCA.

Our result

Theorem

Equivalence of weighted ODCAs *(weights from a field) is in* P.

Reachability problem

co-VS Reachability problem

INPUT:

- A weighted visibly OCA $\mathcal{A}$ over a field,
- an initial configuration c,
- a vector space $\mathcal{V}$, and
- counter value m.

Reachability

co-VS Reachability problem

INPUT:

- A weighted visibly OCA $\mathcal{A}$ over a field,
- an initial configuration c,
- a vector space $\mathcal{V}$, and
- counter value m.

OUTPUT:

- *Yes*, if there exists a run $\mathsf{c} \xrightarrow{*} \overline{\mathcal{V}} \times \{m\}$ in $\mathcal{A}$, and
- *No*, otherwise.

Reachability

co-VS Reachability problem

INPUT:

- A weighted visibly OCA $\mathcal{A}$ over a field,
- an initial configuration c,
- a vector space $\mathcal{V}$, and
- counter value m.

OUTPUT:

- *Yes*, if there exists a run $\mathsf{c} \xrightarrow{*} \overline{\mathcal{V}} \times \{m\}$ in $\mathcal{A}$, and
- *No*, otherwise.

- $z \in \Sigma^*$ is a reachability witness for $(\mathsf{c}, \overline{\mathcal{V}}, m)$ if $\mathsf{c} \xrightarrow{z} \overline{\mathcal{V}} \times \{m\}$.

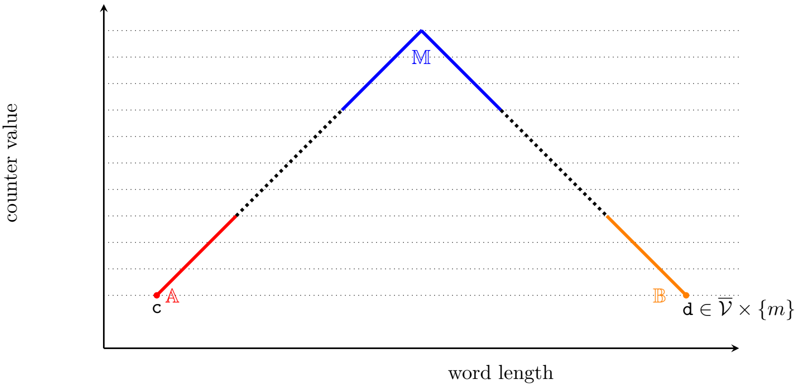
co-VS reachability

Theorem - co-VS reachability

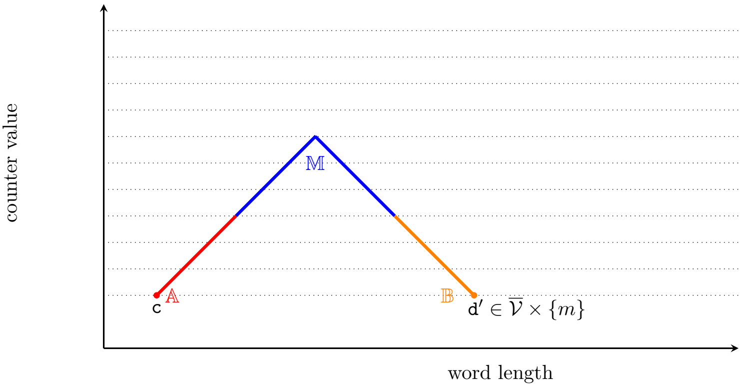
co-VS reachability is decidable in polynomial time.

- We prove this by showing a pseudo-pumping lemma.

Pseudo-pumping lemma (pumping down)



Pseudo-pumping lemma (pumping down)



Special word Lemma

Lemma- Special word

The lexicographically minimal reachability witness z, if it exists, satisfies the following conditions :

1. $z = uy_1^{r_1}w_1y_2^{r_2}w_2$ such that $|uy_1w_1y_2w_2|$ is polynomially bounded in input size, and
2. r_1 and r_2 are polynomially bounded in input size and the input counter values.

Equivalence

Equivalence

Lemma - Witness bound

If two weighted ODCAs $\mathcal{A}_1$ and $\mathcal{A}_2$ are not equivalent, then there exists a distinguishing word z such that the counter values encountered during the run of z are less than a polynomial in the input size.

Equivalence

Lemma - Witness bound

If two weighted ODCAs $\mathcal{A}_1$ and $\mathcal{A}_2$ are not equivalent, then there exists a distinguishing word z such that the counter values encountered during the run of z are less than a polynomial in the input size.

Theorem - Equivalence

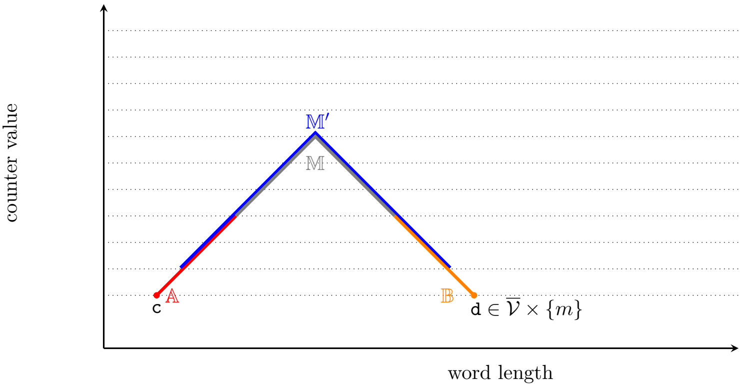
There is a polynomial time algorithm to check the equivalence of two weighted ODCAs (weights from a field).

Other results

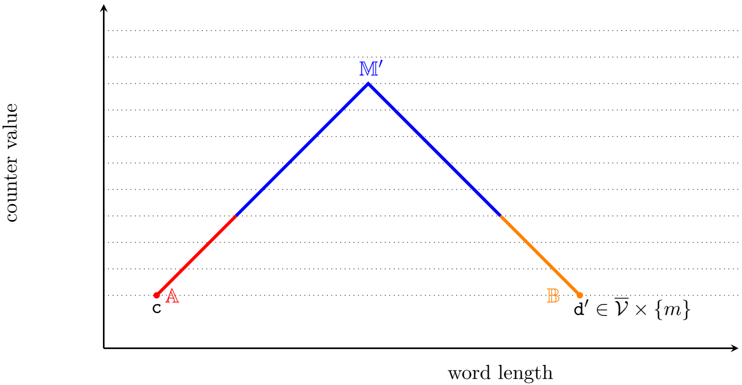
Theorem - Regularity

Given a weighted ODCA (weights from a field), determining whether there exists a weighted automata recognising the same function is in P.

Regularity - pumping up



Regularity - pumping up



Summary

- Equivalence of weighted ODCAs is in P.
- Regularity of weighted ODCAs is in P.

References

Valiant and Paterson.
Deterministic one-counter automata.
JCSS: Journal of Computer and System Sciences, 10, 1975.

Stanislav Böhm and Stefan Göller.
Language equivalence of deterministic real-time one-counter automata is NL-Complete.
In *MFCS*, Lecture Notes in Computer Science, pages 194–205. Springer, 2011. ISBN 978-3-642-22992-3.

Vojtech Forejt, Petr Jancar, Stefan Kiefer, and James Worrell.
Language equivalence of probabilistic pushdown automata.
Inf. Comput, 237:1–11, 2014.

Thank You!