

## **Learning Deterministic One-Counter Automata**

### Learning DOCA

Prince Mathew

DOCA
Active learning
State of the art

### OL

- 1. Behaviour DF
- 2. Partition  ${\cal A}$
- 4 DDEC on
- E Construct C
- 5. Construct 2
- Summary of OL

Conclusio



Prince Mathew, prince@iitgoa.ac.in
Indian Institute of Technology Goa, India



Vincent Penelle, vincent.penelle@u-bordeaux.fr
Univ. Bordeaux. CNRS, Bordeaux INP, LaBRI, France



A.V. Sreejith, sreejithav@iitgoa.ac.in Indian Institute of Technology Goa, India

2 October, 2025

RP 2025



## One-counter automata

### Learning DOCA

### Prince Mathew

### Introduction

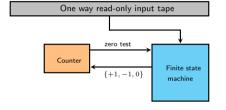
### DOCA

Active learning

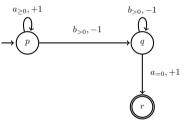
### OL

- 1. Behaviour DF/
- 2. Partition A
- 2. 1 311111011 2-4
- A DDEC on /
- - -
- 6. Construct L
- Summary of OL

Conclusion



**Counter**: Can be incremented, decremented or tested for zero.



OCA accepting  $\{a^nb^na\mid n\geq 0\}.$ 



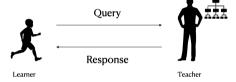
## Active learning framework

### Learning DOCA

### Prince Mathew

### Active learning

- 2. Partition A



- $\circ$  Teacher knows the language of a DOCA  $\mathcal{T}$ .
- $\circ$  Learner's aim: Learn a DOCA  $\mathcal{L}$  equivalent to  $\mathcal{T}$ .
- Learner can guery Teacher.
- Learner uses Teacher's response to learn.



## Queries

Learning DOCA

Prince Mathew

Active learning

2. Partition A

4. PBFS on A

 $\circ$  The Teacher knows the language of a DOCA  $\mathcal{T}$ .

 $\circ$  Learner's aim: Learn a DOCA  $\mathcal{L}$  equivalent to  $\mathcal{T}$ .

## Membership query

Learner: Is w in the language of  $\mathcal{T}$ ?

Teacher: Yes or No.

Minimal-equivalence query

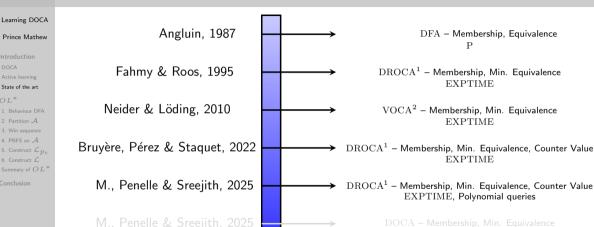
Learner: Is DOCA  $\mathcal{L}$  equivalent to  $\mathcal{T}$ ?

Teacher: Yes or "No and a minimal word

w that distinguishes  $\mathcal{L}$  and  $\mathcal{T}$ ".



## Literature review: Active learning of DOCA



 $^{2}$  VOCA: Visibly OCA

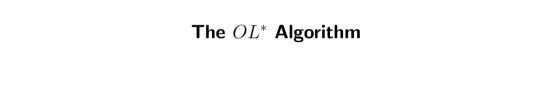
<sup>&</sup>lt;sup>1</sup>Realtime DOCA: strict subclass of DOCA



## Literature review: Active learning of DOCA



<sup>&</sup>lt;sup>1</sup>Realtime DOCA: strict subclass of DOCA





## $OL^*$ - Active learning of DOCA

### Learning DOCA

### Prince Mathew

DOCA
Active learning
State of the art

### $OL^*$

- Behaviour DFA
- o p .... A
- .....
- A DDEC on A
- \_\_\_\_\_
- 5. Construct L
- 6. Construct  $\mathcal{L}$ Summary of OL

Conclusion

## Theorem $(OL^* \text{ is in } P)$

- $\circ$  Let Teacher knows a language recognised by a DOCA  ${\cal T}.$
- $\circ$  The  $OL^*$  algorithm learns a DOCA  $\mathcal L$  that is equivalent to  $\mathcal T$  in time polynomial in  $|\mathcal T|$ , using membership and minimal-equivalence queries.



## Assumptions on DOCA

### Learning DOCA

### Prince Mathew

DOCA
Active learning
State of the art

### $OL^*$

- 1. Behaviour DFA
- 2. Partition A
- o. rriii sequei
- 4. PBFS on ✓
- 5. Construct L
- 6. Construct  $\mathcal{L}$ Summary of OL
- Conclusio

 $\circ$  The Teacher knows a language accepted by a DOCA  $\mathcal{T}$ .

- We denote by  $n = |\mathcal{T}|$ , the number of states.
- $\circ$  To make the presentation simpler, we assume the following about  $\mathcal{T}$ :
  - There are no  $\varepsilon$ -transitions.
  - In a transition, the counter is incremented or decremented at most by one.
- $\circ$  Learner wants to learn a DOCA  ${\cal L}$  equivalent to  ${\cal T}.$



## $OL^*$ : Step 1. Learning k-behaviour DFA

### Learning DOCA

### Prince Mathew

DOCA
Active learning
State of the art

### OL

- 1. Behaviour DFA
- - A
- 2. Partition A
- A DDEC on A
- E Construct C
- 5. Construct L
- 6. Construct  $\mathcal{L}$ Summary of OL

Conclusion

 $\circ$  A DFA  $\mathcal A$  is a k-behaviour **DFA** of  $\mathcal T$  if  $\mathcal A$  is k-equivalent to  $\mathcal T$ . That is,

w is accepted by  $\mathcal{A}$  iff w is accepted by  $\mathcal{T}$ , for all  $|w| \leq k$ .

- $\circ$  Angluin's  $L^*$  algorithm can learn a k-behaviour DFA in time polynomial in k and n.
  - This is done using membership and minimal-equivalence queries.

 $\circ$  Step 1. of learner is to **learn a** poly(n)-behaviour **DFA**.



## $OL^*$ : Step 2. Partitioning the behaviour DFA

### Learning DOCA

#### Prince Mathew

Introduction
DOCA
Active learning
State of the ar

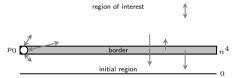
### OL

- 1. Behaviour DFA
- 2. Partition  ${\cal A}$
- J. Will sequen
- 4. PBFS on A
- 6. Construct  $\mathcal{L}_{p_0}$
- Summary of OI

#### Conclusio

### The DFA A is partitioned into:

- Initial region: States reachable by words of length  $< n^4$ .
- Border region: States reachable by words of length  $n^4$  but not less.
- Region of interest: Remaining states.



- o What next?
  - Learner constructs a partial OCA for all border states.
  - Combine all these partial OCA to get the final DOCA A.

poly(n)



## $OL^*$ : Step 2. Partitioning the behaviour DFA

Learning DOCA

Prince Mathew

DOCA
Active learning
State of the a

OL

1. Behaviour DF

2. Partition A

3. Win sequen

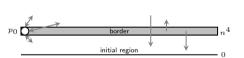
4. PBFS on A

6. Construct  $\mathcal{L}$ 

Conclusio

### The DFA $\mathcal{A}$ is partitioned into:

- Initial region: States reachable by words of length  $< n^4$ .
- Border region: States reachable by words of length  $n^4$  but not less.
- Region of interest: Remaining states.



region of interest

- What next?
  - Learner constructs a partial OCA for all border states.
  - Combine all these partial OCA to get the final DOCA A.

poly(n)



## $OL^*$ : Step 3. Finding a winning sequence

Learning DOCA

Prince Mathew

DOCA
Active learning

OL

1. Behaviour DFA

2. Partition A

3 Win sequence

4. PBFS on A

5. Construct C

6. Construct L

Conclusio

### Definition

We say  $w_0, w_1, w_2, \dots, w_K$  is a **winning sequence** for a state p if the run of these words on  $\mathcal T$  reach configurations

$$(p,i), (p,i+d), (p,i+2d), \dots, (p,i+Kd)$$

respectively, for some  $d \leq n^2$  and  $i > n^3$ .

Lemma (Winning sequence lemma)

For any state  $p_0$  in behaviour DFA A, one can enumerate polynomially many sequences of the form  $w_0, w_1, w_2, \ldots, w_K$  in polynomial time such that one of them is a winning sequence for  $p_0$ .



## $OL^*$ : Step 3. Finding a winning sequence

Learning DOCA

Prince Mathew

DOCA
Active learning
State of the ar

OL

1. Behaviour DF/

2. Partition  ${\cal A}$ 

3. Win sequence

4. PBFS on  ${\cal A}$ 

5. Construct  $\mathcal{L}_{\mathcal{I}}$ 

6. Construct L

Conclusio

### Definition

We say  $w_0, w_1, w_2, \dots, w_K$  is a **winning sequence** for a state p if the run of these words on  $\mathcal T$  reach configurations

$$(p,i), (p,i+d), (p,i+2d), \dots, (p,i+Kd)$$

respectively, for some  $d \le n^2$  and  $i > n^3$ .

### Lemma (Winning sequence lemma)

For any state  $p_0$  in behaviour DFA  $\mathcal{A}$ , one can enumerate polynomially many sequences of the form  $w_0, w_1, w_2, \ldots, w_K$  in polynomial time such that one of them is a winning sequence for  $p_0$ .



## $OL^*$ : Step 4. Parallel BFS on ${\mathcal A}$

Learning DOCA

Prince Mathew

DOCA
Active learning
State of the art

OL

- 1. Behaviour D
- 3. Win sequence
- 4. PBFS on A
- 4. PBF5 on 🎤
- 6. Construct  $\mathcal{L}$

Conclusio

o Consider a winning sequence

$$w_0, w_1, w_2, \ldots, w_K.$$

o Run these words on the behaviour DFA. We reach the state sequence

$$p_0, p_1, p_2, \ldots, p_K.$$

- $\circ$  Run parallel BFS (depth at most  $n^3$ ) from this sequence.
  - All distinct sequences identified.
  - At most  $n^3$  distinct sequences.
  - These sequences are the states of DOCA  $\mathcal{L}_{p_0}$ .



### Learning DOCA

### Prince Mathew

DOCA
Active learning



1. Behaviour DI

2. Partition  ${\cal A}$ 

2 Win sequence

4. PBFS on  ${\cal A}$ 

5. Construct  $\mathcal{L}_{\mathcal{P}_0}$ 

6. Construct  $\mathcal{L}$ Summary of OL













### Learning DOCA

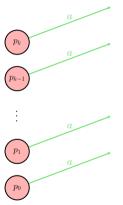
### Prince Mathew

DOCA Active learning

State of the

#### OL

- Behaviour DFA
- 2. Partition  ${\cal A}$
- 4. PBFS on A
- 4. FBF3 011 A
- 5. Construct  $\mathcal{L}_{\mathcal{P}_0}$ 6. Construct  $\mathcal{L}$
- Summary of OL





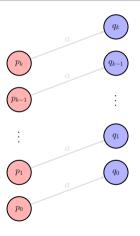
### Learning DOCA

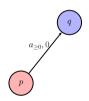
### Prince Mathew

DOCA
Active learning

### State of the

- 1. Behaviour DF
- 2. Partition A
- J. Will sequent
- 4. PBFS on  ${\cal A}$
- 5. Construct  $\mathcal{L}_{p_0}$
- 6. Construct  $\mathcal{L}$ Summary of OL







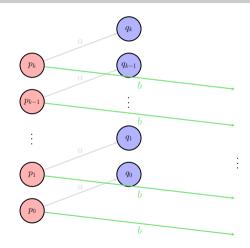
### Learning DOCA

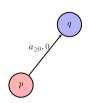
### Prince Mathew

DOCA
Active learning

#### $OL^{1}$

- 1. Behaviour DFA
- 2 Partition A
- Win sequen
- 4. PBFS on  ${\cal A}$
- 5. Construct L
- 6. Construct  $\mathcal{L}$







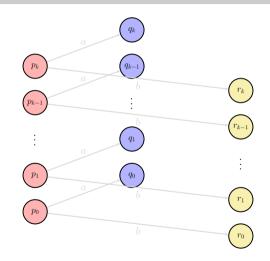
### Learning DOCA

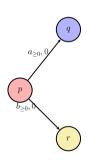
### Prince Mathew

DOCA
Active learning

#### OL

- 1. Behaviour DFA
- 2 Partition A
- 3 Win sequence
- 4. PBFS on  $\mathcal{A}$
- - -
- 6. Construct  $\mathcal{L}$
- Summary of OL







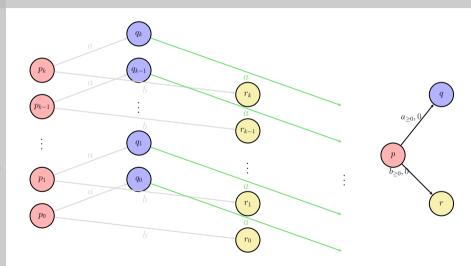
### Learning DOCA

### Prince Mathew

DOCA
Active learning

#### OL.

- 1. Behaviour DFA
- 2 Partition A
- 3. Win sequen
- 4. PBFS on  ${\cal A}$
- -
- 6. Construct  $\mathcal{L}$
- Summary of C





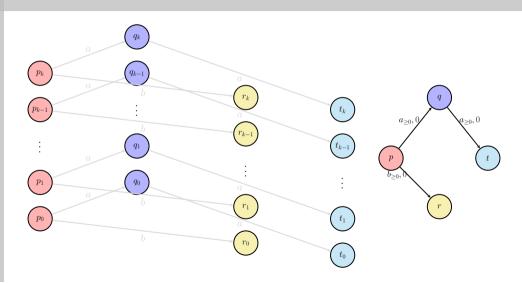
### Learning DOCA

### Prince Mathew

DOCA
Active learning

#### OI.

- 1. Behaviour DF
- 2 Partition A
- 2 14//-----
- 4. PBFS on A
- 4. FBF3 0II 🔑
- 5. Construct  $\mathcal{L}_{\mathcal{P}}$
- 6. Construct  $\mathcal{L}$ Summary of OL





### Learning DOCA

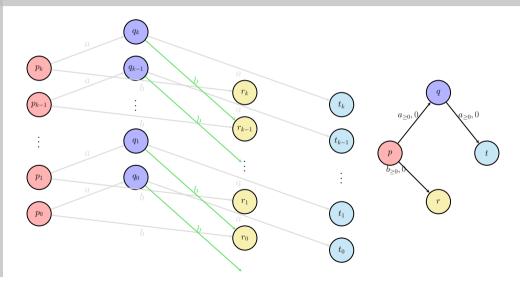
### Prince Mathew

Introduction
DOCA
Active learning

#### O.T.

- 1. Behaviour DFA
- 2. Partition A
- 2 Win sequence
- 4. PBFS on A
- 4. FBF3 0II 🔑
- 6. Construct  $\mathcal{L}_{p_i}$
- Summary of OL

Canalusia





### Learning DOCA

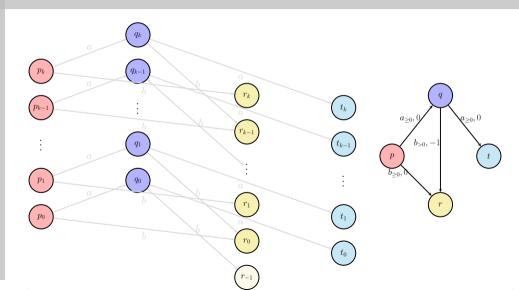
### Prince Mathew

DOCA
Active learning

#### OI.

- 1. Behaviour DFA
- 2. Partition  ${\cal A}$
- 4. PBFS on A
- 4. PBF5 on 🎤
- 6. Construct  $\mathcal{L}$

Construion





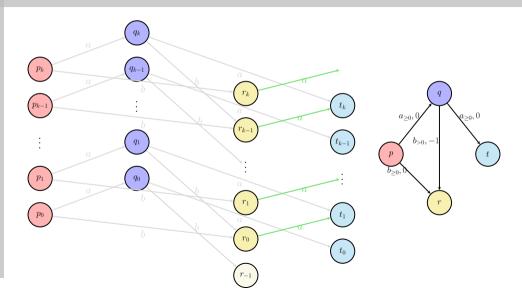
### Learning DOCA

### Prince Mathew

Introduction DOCA Active learning

#### O.T.

- 1. Behaviour DFA
- 2. Partition  ${\cal A}$
- 3. Win sequen
- 4. PBFS on  ${\cal A}$
- 5. Construct L
- 6. Construct L





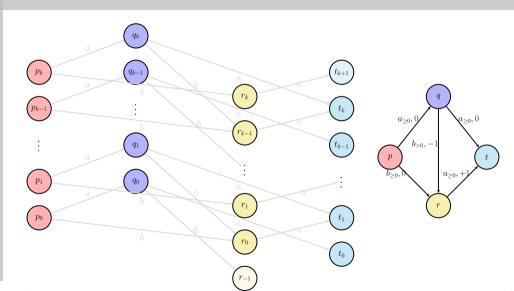
### Learning DOCA

### Prince Mathew

DOCA
Active learning

#### OI.

- 1. Behaviour DFA
- 2. Partition A
- 3. Win sequence
- 4. PBFS on  ${\cal A}$
- 5. Construct  $\mathcal{L}_{\mathcal{P}}$
- 6. Construct  $\mathcal{L}$ Summary of OL





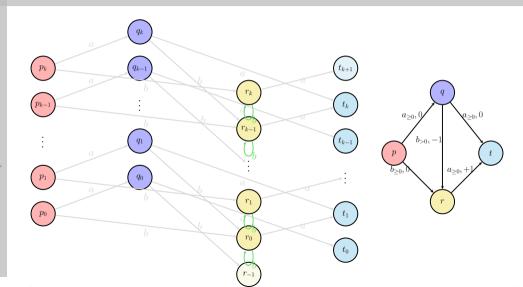
### Learning DOCA

### Prince Mathew

DOCA
Active learning

#### OI.

- 1. Behaviour DFA
- 2. Partition  ${\cal A}$
- J. Will sequen
- 4. PBFS on  ${\cal A}$
- 5. Construct L
- 6. Construct  $\mathcal{L}$ Summary of OL





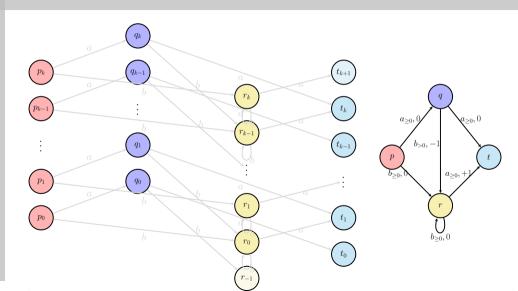
### Learning DOCA

### Prince Mathew

DOCA
Active learning

#### OI.

- 1. Behaviour DFA
- 2. Partition  ${\cal A}$
- J. TVIII sequein
- 4. PBFS on  ${\cal A}$
- 6. Construct  $\mathcal{L}$
- Summary of OL





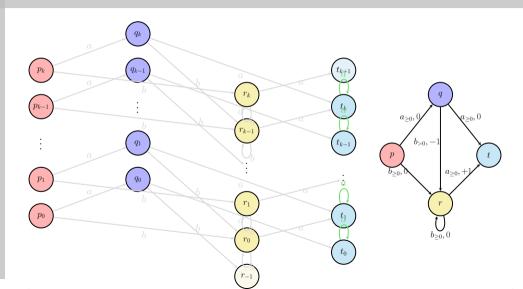
### Learning DOCA

### Prince Mathew

DOCA
Active learning

#### OI.

- 1. Behaviour DFA
- 2. Partition  ${\cal A}$
- 4. PBFS on A
- 4. FBF3 0II 🔑
- 6. Construct  $\mathcal{L}_{p_i}$
- 6. Construct  $\mathcal{L}$ Summary of OL





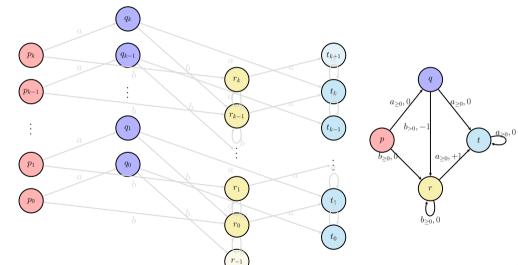
## Learning DOCA

### Prince Mathew

Introduction
DOCA
Active learning

#### OI.

- 1. Behaviour DFA
- 2. Partition A
- 5. vvin sequen
- 4. PBFS on  ${\cal A}$
- 5. Construct  $\mathcal{L}_{T}$
- 6. Construct  $\mathcal{L}$ Summary of OL





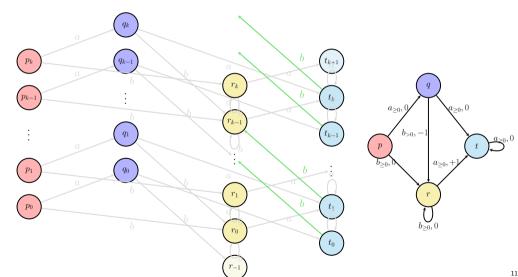
## Learning DOCA

### Prince Mathew

DOCA
Active learning

#### OI.

- 1. Behaviour DFA
- 2. Partition  ${\cal A}$
- Win sequence
- 4. PBFS on  ${\cal A}$
- E Construct (
- 6. Construct  $\mathcal{L}$
- Summary of OL





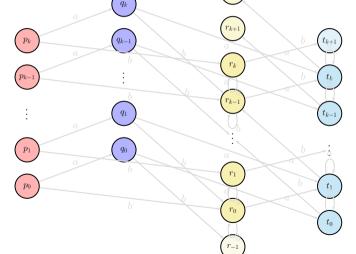
## Learning DOCA

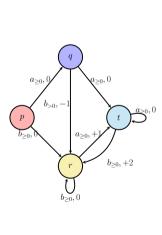
### Prince Mathew

DOCA
Active learning

### OL

- 1. Behaviour DFA
- 2. Partition  ${\cal A}$
- 3. Win sequence
- 4. PBFS on  ${\cal A}$
- E Construct /
- 6. Construct  $\mathcal{L}$ Summary of OL
- Camalinatan







# $OL^*$ : Step 5. Constructing $\mathcal{L}_{p_0}$

### Learning DOCA

### Prince Mathew

- 4. PBFS on A
- 5. Construct Lno.

- $\circ$  The parallel BFS colors "most" of the reachable states from border state p.
- $\circ$  However, upto  $n^3$  number of states are not colored.
  - eg. the state  $r_{-1}$  in the example, and some states reachable from  $r_{-1}$ .
- These are added to the partial OCA with zero counter value.
- The initial region is also added to the partial OCA with zero counter value.

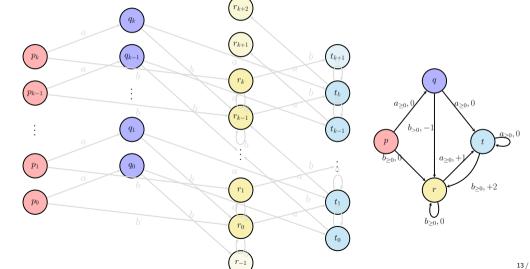


# OL\*: Step 5. Constructing $\mathcal{L}_{p_0}$

Learning DOCA

Prince Mathew

5. Construct  $\mathcal{L}_{\mathcal{D}_0}$ 



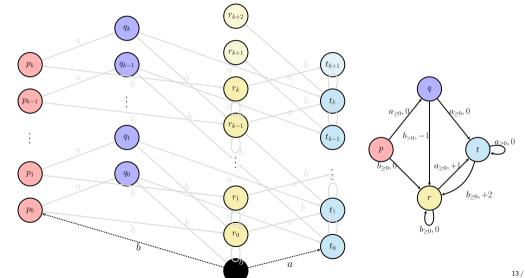


# OL\*: Step 5. Constructing $\mathcal{L}_{p_0}$

### Learning DOCA

### Prince Mathew

- 5. Construct  $\mathcal{L}_{\mathcal{D}_0}$





# OL\*: Step 5. Constructing $\mathcal{L}_{p_0}$

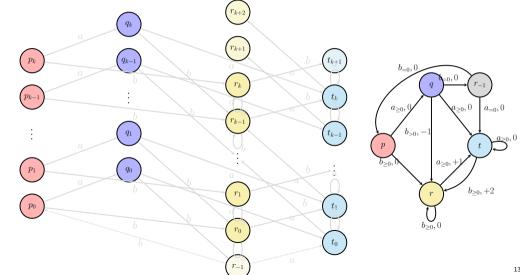
## Learning DOCA

### Prince Mathew

Introduction DOCA Active learning

### OL

- 1. Behaviour DFA
- a Dandelan A
- 3. Win sequence
- 4. PBFS on  ${\cal A}$
- 5. Construct  $\mathcal{L}_{\mathcal{D}_0}$
- 6. Construct  $\mathcal{L}$ Summary of OL
- C----!--





## $\mathit{OL}^*$ : Step 6. Constructing $\mathcal L$

### Learning DOCA

### Prince Mathew

DOCA
Active learning

### OL

- 1. Behaviour DFA
- 2. Partition A
- 3. vvin sequ
- 4. FDF3 OII A
- 5. Construct L
- 6. Construct  $\mathcal{L}$

Conclusio

 $\circ$  The final OCA  $\mathcal L$  is the union of partial OCA corresponding to all border states and the DFA corresponding to the initial region.

- $\circ$  Size of  $\mathcal{L}$  is  $\mathcal{O}(n^8)$
- $\circ$  There exists a poly(n) such that if  $\mathcal{L}$  is poly(n) equivalent to  $\mathcal{T}$ , then  $\mathcal{L}$  and  $\mathcal{T}$  are equivalent (Böhm, Göller & Jančar, 2013).
- Correctness:

```
\mathcal L is poly(n)-equivalent to \mathcal A (Step 0. Construction \mathcal A is poly(n)-equivalent to \mathcal T (Step 1. construction \mathcal L is poly(n)-equivalent to \mathcal T (from above
```

(Böhm, Göller & Jančar, 2013)



## $OL^*$ : Step 6. Constructing $\mathcal L$

### Learning DOCA

#### Prince Mathew

Introduction
DOCA
Active learning
State of the art

### OL

- 1. Behaviour DFA
- 0 D .... A
- 3. Win seq
- 4. PBFS on  ${\cal A}$
- 5. Construct L
- 6. Construct  $\mathcal{L}$

Summary of O I

Conclusio

- $\circ$  The final OCA  $\mathcal{L}$  is the union of partial OCA corresponding to all border states and the DFA corresponding to the initial region.
- $\circ$  Size of  $\mathcal{L}$  is  $\mathcal{O}(n^8)$ .
- $\circ$  There exists a poly(n) such that if  $\mathcal{L}$  is poly(n) equivalent to  $\mathcal{T}$ , then  $\mathcal{L}$  and  $\mathcal{T}$  are equivalent (Böhm, Göller & Jančar, 2013).
- o Correctness:

 $\mathcal L$  is poly(n)-equivalent to  $\mathcal A$  (Step 6. construction)  $\mathcal A$  is poly(n)-equivalent to  $\mathcal T$  (Step 1. construction)  $\mathcal L$  is poly(n)-equivalent to  $\mathcal T$  (from above)



## $OL^*$ : Step 6. Constructing $\mathcal{L}$

### Learning DOCA

### Prince Mathew

Introduction
DOCA
Active learning
State of the art

### OL

- 1. Behaviour DFA
- 0 D .... A
- Win sequ
- 4. PBF5 on A
- 5. Construct L<sub>1</sub>
- 6. Construct  $\mathcal{L}$ Summary of OL

Conclusion

 $\circ$  The final OCA  $\mathcal{L}$  is the union of partial OCA corresponding to all border states and the DFA corresponding to the initial region.

- $\circ$  Size of  $\mathcal{L}$  is  $\mathcal{O}(n^8)$ .
- $\circ$  There exists a poly(n) such that if  $\mathcal{L}$  is poly(n) equivalent to  $\mathcal{T}$ , then  $\mathcal{L}$  and  $\mathcal{T}$  are equivalent (Böhm, Göller & Jančar, 2013).
- Correctness:

 $\mathcal{L}$  is poly(n)-equivalent to  $\mathcal{A}$  (Step 6. construction)  $\mathcal{A}$  is poly(n)-equivalent to  $\mathcal{T}$  (Step 1. construction)  $\mathcal{L}$  is poly(n)-equivalent to  $\mathcal{T}$  (from above)  $\mathcal{L}$  is equivalent to  $\mathcal{T}$  (Böhm, Göller & Jančar, 2013)



## $\mathit{OL}^*$ : Step 6. Constructing $\mathcal L$

### Learning DOCA

### Prince Mathew

Introduction
DOCA
Active learning
State of the art

### OL

- 1. Behaviour DFA
- 1. Dellaviour D17
- 3. Win seq
- 4. PBFS on A
- 5. Construct L
- 6. Construct  $\mathcal{L}$ Summary of OL

Conclusio

 $\circ$  The final OCA  $\mathcal{L}$  is the union of partial OCA corresponding to all border states and the DFA corresponding to the initial region.

- $\circ$  Size of  $\mathcal{L}$  is  $\mathcal{O}(n^8)$ .
- $\circ$  There exists a poly(n) such that if  $\mathcal{L}$  is poly(n) equivalent to  $\mathcal{T}$ , then  $\mathcal{L}$  and  $\mathcal{T}$  are equivalent (Böhm, Göller & Jančar, 2013).
- Correctness:

 $\mathcal L$  is poly(n)-equivalent to  $\mathcal A$  (Step 6. construction)  $\mathcal A$  is poly(n)-equivalent to  $\mathcal T$  (Step 1. construction)  $\mathcal L$  is poly(n)-equivalent to  $\mathcal T$  (from above)



## $OL^*$ : Step 6. Constructing $\mathcal L$

### Learning DOCA

### Prince Mathew

DOCA
Active learning
State of the art

### OL

- 1. Behaviour DFA
- 2 Partition A
- 3. Win seque
- 4. PBFS on A
- 6. Construct L
- Summary of OL

Conclusion

 $\circ$  The final OCA  $\mathcal L$  is the union of partial OCA corresponding to all border states and the DFA corresponding to the initial region.

- $\circ$  Size of  $\mathcal{L}$  is  $\mathcal{O}(n^8)$ .
- $\circ$  There exists a poly(n) such that if  $\mathcal{L}$  is poly(n) equivalent to  $\mathcal{T}$ , then  $\mathcal{L}$  and  $\mathcal{T}$  are equivalent (Böhm, Göller & Jančar, 2013).
- Correctness:

 $\mathcal{L}$  is poly(n)-equivalent to  $\mathcal{A}$  (Step 6. construction)  $\mathcal{A}$  is poly(n)-equivalent to  $\mathcal{T}$  (Step 1. construction)  $\mathcal{L}$  is poly(n)-equivalent to  $\mathcal{T}$  (from above)

14 / 18



## $OL^*$ : Step 6. Constructing $\mathcal L$

### Learning DOCA

### Prince Mathew

DOCA
Active learning
State of the art

### OL

- 1. Behaviour DFA
- 2 Partition A
- Win sequ
- 4. PBFS on A
- 5. Construct L
- 6. Construct  $\mathcal{L}$ Summary of OL

Conclusio

 $\circ$  The final OCA  $\mathcal{L}$  is the union of partial OCA corresponding to all border states and the DFA corresponding to the initial region.

- $\circ$  Size of  $\mathcal{L}$  is  $\mathcal{O}(n^8)$ .
- $\circ$  There exists a poly(n) such that if  $\mathcal{L}$  is poly(n) equivalent to  $\mathcal{T}$ , then  $\mathcal{L}$  and  $\mathcal{T}$  are equivalent (Böhm, Göller & Jančar, 2013).
- Correctness:

 $\mathcal{L}$  is poly(n)-equivalent to  $\mathcal{A}$  (Step 6. construction)  $\mathcal{A}$  is poly(n)-equivalent to  $\mathcal{T}$  (Step 1. construction)  $\mathcal{L}$  is poly(n)-equivalent to  $\mathcal{T}$  (from above)  $\mathcal{L}$  is equivalent to  $\mathcal{T}$  (Böhm, Göller & Jančar, 2013)



## Summary of ${\cal O}L^*$

### Learning DOCA

### Prince Mathew

Introduction DOCA Active learning

### OI

- 1. Behaviour DF/
- o D .... A
- 3. Win segu
- 4. PBFS on  ${\cal A}$
- 5. Construct  $\mathcal{L}_7$
- 6. Construct L

### Summary of $OL^st$

- $\circ$  Construct poly(n)-behaviour DFA using  $L^*$  algorithm.
- o Partition the behaviour DFA into initial region, border, and region of interest.
- o For each border state:
  - Generate a winning sequence of words:  $w_0, w_1, \ldots, w_K$ .
  - Run these words on the DFA to get the sequence of states:  $p_0, p_1, \ldots, p_K$ .
  - Run parallel BFS from this sequence.
  - All reachable sequences of parallel BFS form states of partial OCA.
  - Counter values are incremented / decremented based on the shift in the sequence.
  - Add the missed states and initial region to get partial OCA.
- o Construct the final OCA by combining the partial OCA.



## Conclusion

### Learning DOCA

### Prince Mathew

Introduction
DOCA
Active learning
State of the ar

OI

### Behaviour D

- Behaviour DFA
- 3. Win sequence
- 4. PBFS on A
- 6. Construct  $\mathcal{L}$

### Summary of O

### Conclusion

### **Theorem**

 $OL^*$  learns a DOCA equivalent to the Teacher's DOCA using membership and minimal-equivalence queries, and in time polynomial in the size of a smallest DOCA recognising the language.

### Corollary

Polynomial approximation for minimisation of DOCA.



## Future work

### Learning DOCA

### Prince Mathew

Introduction DOCA Active learning

### O I

- 1. Behaviour DF/
- 1. Dellaviour Dr.
- 2. Partition A
- A DDEC on A
- 5. Construct Lp
- 6. Construct  $\mathcal{L}$ Summary of  $OL^3$

#### Conclusion

Replacing minimal-equivalence with equivalence query.

- $\circ$  Practical  $OL^*$  algorithm.
- Improving running time of equivalence.
- o Learning weighted models (like weighted visibly OCA).



### Learning DOCA

### Prince Mathew

Introduction

Active learning

- 1. Behaviour DF
- . . . . . .
- 2 Win sequence
- 5. vvin seque
- 5. Construct  $\mathcal{L}_{p_i}$

### Conclusion

# Thank You!