Universality Checking for Unambiguous VASSes

Wojciech Czerwiński

Diego Figueira

Piotr Hofman

Main message

unambiguous systems are worth studying

Main message

unambiguous systems are
often simpler than nondeterministic

unambiguous systems are worth studying

Unambiguity

Unambiguity

for each word there is at most one accepting run

Unambiguity

for each word there is at most one accepting run

many problems become simpler:

Unambiguity

for each word there is at most one accepting run

many problems become simpler:

universality for UFA (PTime)

Unambiguity

for each word there is at most one accepting run

many problems become simpler:

universality for UFA (PTime)

equivalence for UFA (PTime)

Unambiguity

for each word there is at most one accepting run

many problems become simpler:

universality for UFA (PTime)

equivalence for UFA (PTime)

universality for URA (2ExpSpace)

Universality

Universality

Emptiness not easier

Universality

Emptiness not easier

Universality is easier

Universality

Emptiness not easier

Universality is easier

Equivalence is easier

Universality

Emptiness not easier

Universality is easier

Equivalence is easier

Inclusion is easier

Universality

Emptiness not easier

Universality is easier

Equivalence is easier

Inclusion is easier

First step: universality problem

Which system?

Which system?

Universality in unambiguous case is:

Which system?

Universality in unambiguous case is:

in NC^2 for finite automata

Which system?

Universality in unambiguous case is:

in NC^2 for finite automata

in 2ExpSpace for register automata

Which system?

Universality in unambiguous case is:

in NC^2 for finite automata

in 2ExpSpace for register automata

not investigated for:

Which system?

Universality in unambiguous case is:

in NC^2 for finite automata

in 2ExpSpace for register automata

not investigated for:

pushdown automata

Which system?

Universality in unambiguous case is:

in NC^2 for finite automata

in 2ExpSpace for register automata

not investigated for:

pushdown automata

counter automata

What was known

What was known

Universality is:

What was known

Universality is:

- decidable for OCN and VASS (wqo)

What was known

Universality is:

- decidable for OCN and VASS (wqo)
- Ackermann-hard for OCN

What was known

Universality is:

- decidable for OCN and VASS (wqo)
- Ackermann-hard for OCN

Acceptance by states, ε-transitions allowed

Results

Results

Theorem

The universality problem in unambiguous case for

Results

Theorem

The universality problem in unambiguous case for

1) VASS is ExpSpace-complete

Results

Theorem

The universality problem in unambiguous case for

1) VASS is ExpSpace-complete

2) d-VASS is PSpace-complete for $d \geq 2$, binary

Results

Theorem

The universality problem in unambiguous case for

1) VASS is ExpSpace-complete

2) d-VASS is PSpace-complete for $d \geq 2$, binary

3) 1-VASS is coNP-hard, binary

Results

Theorem

The universality problem in unambiguous case for

1) VASS is ExpSpace-complete

2) d-VASS is PSpace-complete for $d \geq 2$, binary

3) 1-VASS is coNP-hard, binary

4) d-VASS is in NC^2, NL-hard, $d \geq 1$, unary

Open problems

Open problems

Complexity of universality for binary OCN
(coNP-complete? PSpace-complete?)

Open problems

Complexity of universality for binary OCN
(coNP-complete? PSpace-complete?)

equivalence, inclusion, co-finiteness problems

Open problems

Complexity of universality for binary OCN
(coNP-complete? PSpace-complete?)

equivalence, inclusion, co-finiteness problems

for unambiguous

Open problems

Complexity of universality for binary OCN
(coNP-complete? PSpace-complete?)

equivalence, inclusion, co-finiteness problems

for unambiguous

OCN, VASS, counter automata,
pushdown-automata, RA

Thank you!