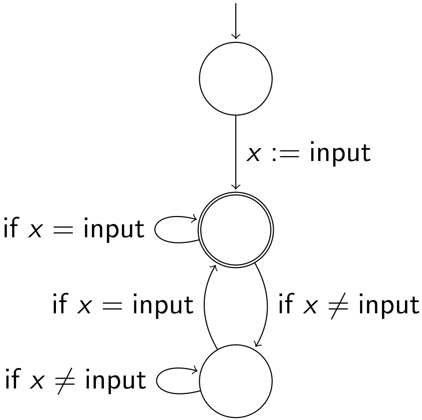
Universality of unambiguous register automata

Corentin Barloy - University of Lille
Joint work with Lorenzo Clemente - MIMUW (Warsaw)

STACS 2021

Non-guessing register automata

x is a register that stores values from $\mathbb{N}$

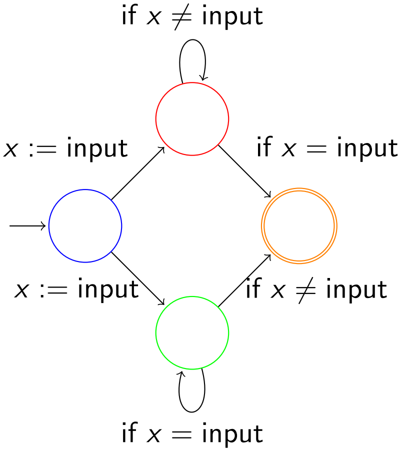


The unambiguity property

Unambiguity: there is a most one accepting run for each word.

The unambiguity property

Unambiguity: there is a most one accepting run for each word.



Decision problems

The universality problem asks whether a given unambiguous register automaton accepts every word.

Decision problems

The universality problem asks whether a given unambiguous register automaton accepts every word.

Inclusion $\Longleftrightarrow$ Equivalence $\Longleftrightarrow$ Universality

Decision problems

The universality problem asks whether a given unambiguous register automaton accepts every word.

$$\text{Inclusion} \iff \text{Equivalence} \iff \text{Universality}$$

This is quite surprising: for DCFG, Universality is decidable but not Inclusion!

Main result

Theorem [Mottet, Quaas STACS'19]

The universality problem for unambiguous register automata in decidable in 2EXP-SPACE.

Main result

Theorem [Mottet, Quaas STACS'19]

The universality problem for unambiguous register automata in decidable in 2EXP-SPACE.

Theorem [B, Clemente]

The universality problem for unambiguous register automata in decidable in 2EXP-TIME.

Main result

Theorem [Mottet, Quaas STACS'19]

The universality problem for unambiguous register automata in decidable in 2EXP-SPACE.

Theorem [B, Clemente]

The universality problem for unambiguous register automata in decidable in 2EXP-TIME.

Theorem [Bojańczyk, Klin, Moerman]

The universality problem for unambiguous register automata **with guessing**, and over equality or **order**, is decidable in **EXP-TIME**.

Main result

Theorem [Mottet, Quaas STACS'19]

The universality problem for unambiguous register automata in decidable in 2EXP-SPACE.

Theorem [B, Clemente]

The universality problem for unambiguous register automata in decidable in 2EXP-TIME.

Theorem [Bojańczyk, Klin, Moerman]

The universality problem for unambiguous register automata **with guessing**, and over equality or **order**, is decidable in **EXP-TIME**.

(Undecidable in general)

Proof sketch

1. Counting of the number of accepted orbits.

Proof sketch

1. Counting of the number of accepted orbits.

$\text{orbit}(u) = \{\text{all renaming of } u\}$

Proof sketch

1. Counting of the number of accepted orbits.

$\text{orbit}(u) = \{\text{all renaming of } u\}$

$$F_s(n,k) = \text{number of orbits of runs that end in } s \\ \text{with length } n \text{ and with } k \text{ different letters}$$

Proof sketch

1. Counting of the number of accepted orbits.
2. Reduction to zeroness of linrec sequence.

Proof sketch

1. Counting of the number of accepted orbits.
2. Reduction to zeroness of linrec sequence.

$$\begin{cases} f_{\color{blue}\bullet}(n+1,k+1) = & 0 \\ f_{\color{red}\bullet}(n+1,k+1) = & f_{\color{blue}\bullet}(n,k) + (k+1)\cdot f_{\color{blue}\bullet}(n,k+1) + f_{\color{red}\bullet}(n,k) + k\cdot f_{\color{red}\bullet}(n,k+1) \\ f_{\color{green}\bullet}(n+1,k+1) = & f_{\color{blue}\bullet}(n,k) + (k+1)\cdot f_{\color{blue}\bullet}(n,k+1) + f_{\color{green}\bullet}(n,k+1) \\ f_{\color{orange}\bullet}(n+1,k+1) = & f_{\color{red}\bullet}(n,k+1) + f_{\color{green}\bullet}(n,k) + k\cdot f_{\color{green}\bullet}(n,k+1) \\ S(n+1,k+1) = & S(n,k) + (k+1)\cdot S(n,k+1) \\ g(n+1,k+1) = & S(n,k) - f_{\color{orange}\bullet}(n,k) \end{cases}$$

Proof sketch

1. Counting of the number of accepted orbits.
2. Reduction to zeroness of linrec sequence.
3. Modelling using Ore polynomials.

Proof sketch

1. Counting of the number of accepted orbits.
2. Reduction to zeroness of linrec sequence.
3. Modelling using Ore polynomials.

$$\left\{\begin{array}{rrrrrl}
(\partial_1\partial_2)\cdot f_{\color{blue}\bullet} & & & & & =0\\
-\partial_2\cdot f_{\color{blue}\bullet} & +(\partial_1\partial_2-(k+1)\partial_2-1)\cdot f_{\color{red}\bullet} & & & & =0\\
-\partial_2\cdot f_{\color{blue}\bullet} & & + & (\partial_1\partial_2-\partial_2)\cdot f_{\color{green}\bullet} & & =0\\
 & -\partial_2\cdot f_{\color{red}\bullet} & + & (-(k-1)\partial_2-1)\cdot f_{\color{green}\bullet} & +(\partial_1\partial_2)\cdot f_{\color{orange}\bullet} & =0\\
 & (\partial_1\partial_2-(k+1)\partial_2-1)\cdot S & & & & =0\\
 & \partial_1\partial_2 g-S+f_{\color{orange}\bullet} & & & & =0
\end{array}\right.$$

Proof sketch

1. Counting of the number of accepted orbits.
2. Reduction to zeroness of linrec sequence.
3. Modelling using Ore polynomials.
4. Perform elimination.

Proof sketch

1. Counting of the number of accepted orbits.
2. Reduction to zeroness of linrec sequence.
3. Modelling using Ore polynomials.
4. Perform elimination.

$$((k^2 - 5k + 6)\partial_1^3\partial_2^3 + (2k + 2)\partial_1^3\partial_2^2 - 2\partial_1^2\partial_2 + \partial_2^2 - (3k + 3)) \cdot g = 0$$

Proof sketch

1. Counting of the number of accepted orbits.
2. Reduction to zeroness of linrec sequence.
3. Modelling using Ore polynomials.
4. Perform elimination.

Yields a 4-EXP-SPACE algorithm.

Improving the complexity

Idea: Instead of removing one variable at a time, invert the matrix using (non-commutative) linear algebra.

Improving the complexity

Idea: Instead of removing one variable at a time, invert the matrix using (non-commutative) linear algebra.

- Put matrices of operators in a triangular form (Hermite form):

$$\begin{pmatrix} (\partial_1 - 1)\partial_2 & -\partial_2 \\ -k\partial_2 - 1 & \partial_1\partial_2 \end{pmatrix} \Rightarrow \begin{pmatrix} 1 & (\frac{k}{\partial_1 - 1} - \partial_1)\partial_2 \\ 0 & \partial_2^2 - \frac{1}{\partial_1^2 - \partial_1 - (k+1)}\partial_2 \end{pmatrix}$$

Improving the complexity

Idea: Instead of removing one variable at a time, invert the matrix using (non-commutative) linear algebra.

- Put matrices of operators in a triangular form (Hermite form):

$$\begin{pmatrix} (\partial_1 - 1)\partial_2 & -\partial_2 \\ -k\partial_2 - 1 & \partial_1\partial_2 \end{pmatrix} \Rightarrow \begin{pmatrix} 1 & (\frac{k}{\partial_1 - 1} - \partial_1)\partial_2 \\ 0 & \partial_2^2 - \frac{1}{\partial_1^2 - \partial_1 - (k+1)}\partial_2 \end{pmatrix}$$

- Exponential bound on the coefficient of the Hermite form [Giesbrecht,Kim 11].

Main theorem

Theorem [B.,Clemente 20]

The universality problem for unambiguous register automata is decidable in 2EXP-TIME.

Conclusion

- Improving the complexity. Monicity conjecture: monic cancelling relations suffice:

$$(\cancel{(k^2-5k+6)}\partial_1^3\partial_2^3 + (2k+2)\partial_1^3\partial_2^2 - 2\partial_1^2\partial_2 + \partial_2^2 - (3k+3)) \cdot g = 0$$

This is false for linrec in general, we have to restrict to equations arising from automata. This would give an EXP-TIME bound.

Conclusion

- Improving the complexity. Monicity conjecture: monic cancelling relations suffice:

$$(\cancel{(k^2-5k+6)}\partial_1^3\partial_2^3 + (2k+2)\partial_1^3\partial_2^2 - 2\partial_1^2\partial_2 + \partial_2^2 - (3k+3)) \cdot g = 0$$

  This is false for linrec in general, we have to restrict to equations arising from automata. This would give an EXP-TIME bound.
- Extend to other structures: other atoms, timed automata, pushdown automata...

Conclusion

- Improving the complexity. Monicity conjecture: monic cancelling relations suffice:

$$(\cancel{(k^2 - 5k + 6)}\partial_1^3\partial_2^3 + (2k+2)\partial_1^3\partial_2^2 - 2\partial_1^2\partial_2 + \partial_2^2 - (3k+3)) \cdot g = 0$$

  This is false for linrec in general, we have to restrict to equations arising from automata. This would give an EXP-TIME bound.

- Extend to other structures: other atoms, timed automata, pushdown automata. . .
- Extend to weighted automata.

M. Giesbrecht and M. S. Kim.
Computing the Hermite form of a matrix of Ore polynomials.
Journal of Algebra, 376:341–362, 2013.

A. Mottet and K. Quaas.
The containment problem for unambiguous register automata and unambiguous timed automata.
Theory of Computing Systems, 2020.

O. Ore.
Theory of non-commutative polynomials.
Annals of Mathematics, 34(3):480–508, 1933.

R. Stearns and H. Hunt.
On the equivalence and containment problems for unambiguous regular expressions, grammars, and automata.
In *Proc. of SFCS'81*, pages 74–81, Washington, DC, USA, 1981. IEEE Computer Society.