Equations and logic on words

Sam van Gool

Utrecht University

WARU III, Prague

12 May 2019

Overview

Logic on words

Equations between languages

Equations between words

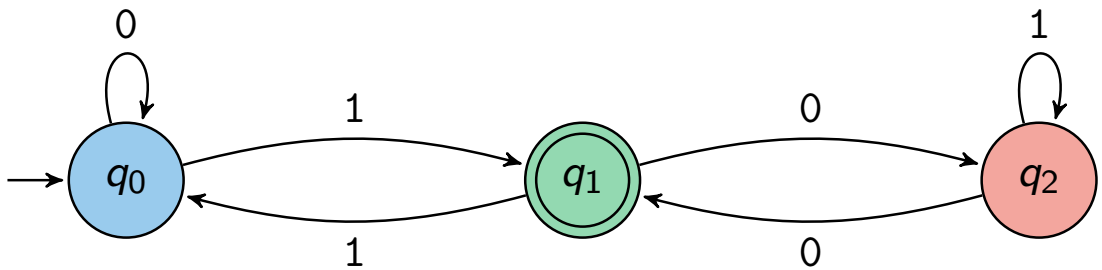
Overview

Regular languages: example

- A programming problem: given a natural number in binary, $w \in \{0,1\}^*$, determine if w is congruent 1 modulo 3.

Regular languages: example

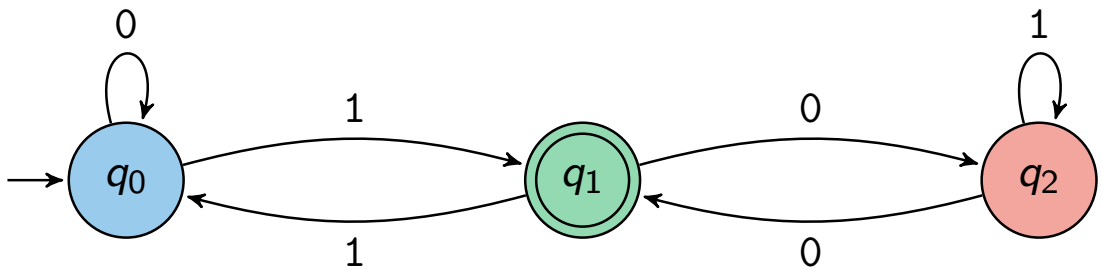
- A programming problem: given a natural number in binary, $w \in \{0,1\}^*$, determine if w is congruent 1 modulo 3.
- Solution 1: a (deterministic) automaton A:



Answer yes iff A accepts w.

Regular languages: example

- A programming problem: given a natural number in binary, $w \in \{0,1\}^*$, determine if w is congruent 1 modulo 3.
- Solution 1: a (deterministic) automaton A:



  Answer yes iff A accepts w.

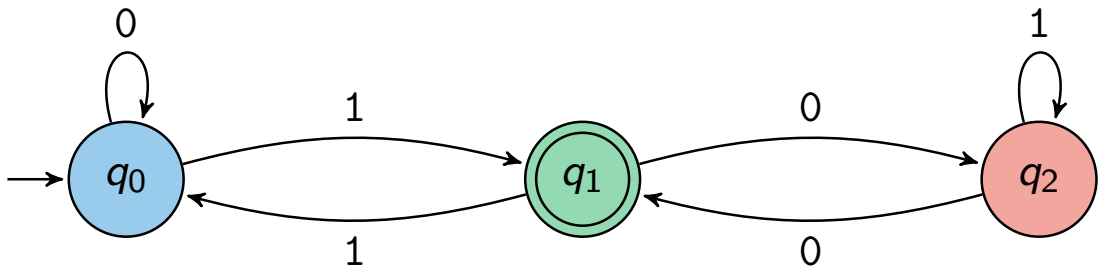
- Solution 2: a homomorphism $\varphi\colon \{0,1\}^* \to S_3$ defined by

$$0 \mapsto (1\,2), \quad 1 \mapsto (0\,1).$$

  Answer yes iff the permutation $\varphi(w)$ sends 0 to 1.

Regular languages: example

- A programming problem: given a natural number in binary, $w \in \{0,1\}^*$, determine if w is congruent 1 modulo 3.
- Solution 1: a (deterministic) automaton A:



  Answer yes iff A accepts w.

- Solution 3: an MSO sentence φ:

  $\exists Q_0 \exists Q_1 \exists Q_2 (Q_0(\texttt{first}) \wedge Q_1(\texttt{last}) \wedge$

  $\forall x[0(x) \wedge Q_0(x) \rightarrow Q_0(\mathrm{S}x)] \wedge [1(x) \wedge Q_0(x) \rightarrow Q_1(\mathrm{S}x)] \wedge \ldots).$

  Answer yes iff w satisfies the formula φ.

Regular languages

Regular languages are subsets $L \subseteq \Sigma^*$ which are ...

- recognizable by a finite automaton;
- invariant under a finite index monoid congruence;
- definable by a monadic second order sentence.

Myhill-Nerode 1958; Büchi 1960

Logic on words

- **Syntax.** Monadic Second Order (MSO) logic over $<$, Σ.
  - Basic propositional connectives: $\wedge$, $\neg$.
  - Quantification over first-order variables x, y, ... and monadic second-order variables P, Q, ...
  - Relational signature: $x < y$, $\mathrm{a}(x)$ for $a \in \Sigma$.

Logic on words

- **Syntax.** Monadic Second Order (MSO) logic over $<$, Σ.
- **Semantics.** A word $w = a_1 \dots a_n$ gives a structure W.
  - The underlying set of W is $\{1, \dots, n\}$.
  - The natural linear order $<^W$ interprets the binary predicate $<$.
  - For every letter $a \in \Sigma$, $\mathrm{a}^W := \{i \in \{1, \dots, n\} : a_i = a\}$.

Logic on words

- **Syntax.** Monadic Second Order (MSO) logic over $<$, Σ.
- **Semantics.** A word $w = a_1 \dots a_n$ gives a structure W.

- For a sentence φ, $L_\varphi := \{w \in \Sigma^* \mid w \models \varphi\}$.
- A language L is regular iff $L = L_\varphi$ for some φ in MSO.

- Shortcuts such as $\mathtt{S}(x)$, $\mathtt{first}$, $\mathtt{last}$, $\subseteq$, ... are MSO-definable.
- First Order (FO) logic is obtained by disallowing second order variables and second order quantifiers.

Logic on words: examples

$$\varphi\colon \exists P\Big[\ P(\texttt{first}) \wedge \neg P(\texttt{last}) \wedge \forall x(P(x) \leftrightarrow \neg P(\texttt{S}(x))\ \Big].$$

Logic on words: examples

$$\varphi\colon \exists P\Big[\ P(\texttt{first}) \wedge \neg P(\texttt{last}) \wedge \forall x(P(x) \leftrightarrow \neg P(\texttt{S}(x))\ \Big].$$

- *aaaa*

Logic on words: examples

$$\varphi : \exists P \Big[\ P(\texttt{first}) \wedge \neg P(\texttt{last}) \wedge \forall x (P(x) \leftrightarrow \neg P(\texttt{S}(x)))\ \Big].$$

- $aaaa \models \varphi$,

Logic on words: examples

φ: $\exists P \big[\ P(\texttt{first}) \wedge \neg P(\texttt{last}) \wedge \forall x (P(x) \leftrightarrow \neg P(\texttt{S}(x)))\ \big]$.

- $aaaa \models \varphi$, but $aaaaa \not\models \varphi$.
- $W \models \varphi$ iff W has even length.

Logic on words: examples

$\varphi\colon \exists P\big[\ P(\texttt{first}) \wedge \neg P(\texttt{last}) \wedge \forall x(P(x) \leftrightarrow \neg P(\texttt{S}(x))\ \big].$

- $aaaa \models \varphi$, but $aaaaa \not\models \varphi$.
- $W \models \varphi$ iff W has even length.

$\psi\colon \exists P\big[\ \exists x P(x)\ \wedge\ P \subseteq \texttt{a} \wedge \forall y\big(\ (\forall x[P(x) \to x < y])\ \to \texttt{b}(y)\big)\ \big].$

Logic on words: examples

$\varphi\colon \exists P\big[\ P(\texttt{first}) \wedge \neg P(\texttt{last}) \wedge \forall x(P(x) \leftrightarrow \neg P(\texttt{S}(x))\ \big].$

- $aaaa \models \varphi$, but $aaaaa \not\models \varphi$.
- $W \models \varphi$ iff W has even length.

$\psi\colon \exists P\big[\ \exists x P(x)\ \wedge\ P \subseteq \texttt{a} \wedge \forall y\big(\ (\forall x[P(x) \to x < y])\ \to \texttt{b}(y)\big)\ \big].$

- $aacbaccaabbb$

Logic on words: examples

$\varphi\colon \exists P\big[\ P(\texttt{first}) \wedge \neg P(\texttt{last}) \wedge \forall x(P(x) \leftrightarrow \neg P(\texttt{S}(x))\ \big].$

- $aaaa \models \varphi$, but $aaaaa \not\models \varphi$.
- $W \models \varphi$ iff W has even length.

$\psi\colon \exists P\big[\ \exists x P(x)\ \wedge\ P \subseteq \texttt{a} \wedge \forall y\big(\ (\forall x[P(x) \to x < y])\ \to \texttt{b}(y)\big)\ \big].$

- $aacbaccaabbb \models \varphi$,

Logic on words: examples

$\varphi\colon \exists P\big[\ P(\texttt{first}) \wedge \neg P(\texttt{last}) \wedge \forall x(P(x) \leftrightarrow \neg P(\texttt{S}(x))\ \big].$

- $aaaa \models \varphi$, but $aaaaa \not\models \varphi$.
- $W \models \varphi$ iff W has even length.

$\psi\colon \exists P\big[\ \exists x P(x)\ \wedge\ P \subseteq \texttt{a} \wedge \forall y\big(\ (\forall x[P(x) \to x < y])\ \to \texttt{b}(y)\big)\ \big].$

- $aacbaccaabbb \models \varphi$, but $aacbaccaabbc \not\models \varphi$.
- $W \models \varphi$ iff W has a non-empty subset of a-positions after which there are only b-positions.

Logic on words: examples

$\varphi\colon \exists P\big[\ P(\texttt{first}) \wedge \neg P(\texttt{last}) \wedge \forall x(P(x) \leftrightarrow \neg P(\texttt{S}(x))\ \big].$

- $aaaa \models \varphi$, but $aaaaa \not\models \varphi$.
- $W \models \varphi$ iff W has even length.

$\psi\colon \exists P\big[\ \exists x P(x)\ \wedge\ P \subseteq \mathtt{a} \wedge \forall y\big(\ (\forall x[P(x) \to x < y])\ \to \mathtt{b}(y)\big)\ \big].$

- $aacbaccaabbb \models \varphi$, but $aacbaccaabbc \not\models \varphi$.
- $W \models \varphi$ iff W has a non-empty subset of a-positions after which there are only b-positions.

$\psi'\colon \exists x\big[\ \mathtt{a}(x)\ \wedge\ \forall y[x < y \to (\neg \mathtt{a}(y) \wedge \mathtt{b}(y))]\ \big].$

Logic on words: examples

$\varphi\colon \exists P\big[\ P(\texttt{first}) \wedge \neg P(\texttt{last}) \wedge \forall x(P(x) \leftrightarrow \neg P(\texttt{S}(x))\ \big].$

- $aaaa \models \varphi$, but $aaaaa \not\models \varphi$.
- $W \models \varphi$ iff W has even length.

$\psi\colon \exists P\big[\ \exists x P(x)\ \wedge\ P \subseteq \mathtt{a} \wedge \forall y\big(\ (\forall x[P(x) \to x < y])\ \to \mathtt{b}(y)\big)\ \big].$

- $aacbaccaabbb \models \varphi$, but $aacbaccaabbc \not\models \varphi$.
- $W \models \varphi$ iff W has a non-empty subset of a-positions after which there are only b-positions.

$\psi'\colon \exists x\big[\ \mathtt{a}(x)\ \wedge\ \forall y[x < y \to (\neg \mathtt{a}(y) \wedge \mathtt{b}(y))]\ \big].$

"There is a last a-position, with only b-positions after that."

Logic on words: examples

$\varphi\colon \exists P\big[\ P(\texttt{first}) \wedge \neg P(\texttt{last}) \wedge \forall x(P(x) \leftrightarrow \neg P(\texttt{S}(x))\ \big].$

- $aaaa \models \varphi$, but $aaaaa \not\models \varphi$.
- $W \models \varphi$ iff W has even length.

$\psi\colon \exists P\big[\ \exists x P(x)\ \wedge\ P \subseteq \mathtt{a} \wedge \forall y\big(\ (\forall x[P(x) \to x < y])\ \to \mathtt{b}(y)\big)\ \big].$

- $aacbaccaabbb \models \varphi$, but $aacbaccaabbc \not\models \varphi$.
- $W \models \varphi$ iff W has a non-empty subset of a-positions after which there are only b-positions.

$\psi'\colon \exists x\big[\ \mathtt{a}(x)\ \wedge\ \forall y[x < y \to (\neg \mathtt{a}(y) \wedge \mathtt{b}(y))]\ \big].$

"There is a last a-position, with only b-positions after that."

ψ and ψ' are equivalent, and ψ' is first order.

Logic on words: examples

$\varphi\colon \exists P\big[\ P(\texttt{first}) \wedge \neg P(\texttt{last}) \wedge \forall x(P(x) \leftrightarrow \neg P(\texttt{S}(x))\ \big].$

- $aaaa \models \varphi$, but $aaaaa \not\models \varphi$.
- $W \models \varphi$ iff W has even length.

$\psi\colon \exists P\big[\ \exists x P(x)\ \wedge\ P \subseteq \texttt{a} \wedge \forall y\big(\ (\forall x[P(x) \to x < y])\ \to \texttt{b}(y)\big)\ \big].$

- $aacbaccaabbb \models \varphi$, but $aacbaccaabbc \not\models \varphi$.
- $W \models \varphi$ iff W has a non-empty subset of a-positions after which there are only b-positions.

$\psi'\colon \exists x\big[\ \texttt{a}(x)\ \wedge\ \forall y[x < y \to (\neg \texttt{a}(y) \wedge \texttt{b}(y))]\ \big].$

"There is a last a-position, with only b-positions after that."

ψ and ψ' are equivalent, and ψ' is first order.

Question. Does such an equivalent first order formula exist for φ?

Monoids and finite index congruences

- A monoid is a set M equipped with an associative binary operation and a unit.
- The set Σ^* of finite words is a free monoid.
  - multiplication is concatenation;
  - unit is the empty word ϵ;

Monoids and finite index congruences

- A monoid is a set M equipped with an associative binary operation and a unit.
- The set Σ^* of finite words is a free monoid.
  - multiplication is concatenation;
  - unit is the empty word ϵ;
- A congruence on M is an equivalence relation θ which respects multiplication.
  - The quotient M/θ is again a monoid;
  - A congruence θ is called finite index if M/θ is finite.
- A language $L \subseteq \Sigma^*$ is regular iff there exists finite index θ_L under which L is invariant:

$$w \in L \text{ and } w\theta_L w' \text{ implies } w' \in L.$$

Regular languages

Regular languages are subsets $L \subseteq \Sigma^*$ which are ...

- recognizable by a finite automaton;
- invariant under a finite index monoid congruence;
- definable by a monadic second order sentence.

Myhill-Nerode 1958; Büchi 1960

Duality

Key insight. The connection between MSO logic on words and monoids is an instance of Stone-Jónsson-Tarski duality.

Algebra	**Space**
Lindenbaum algebra of a logic	Canonical model
Residuated Boolean algebra of regular languages	(Pro)finite monoid

Gehrke, Grigorieff, Pin 2008

Duality

Key insight. The connection between MSO logic on words and monoids is an instance of Stone-Jónsson-Tarski duality.

Algebra	**Space**
Lindenbaum algebra of a logic	Canonical model
Residuated Boolean algebra of regular languages	(Pro)finite monoid
Equations between languages	Equations between words

Gehrke, Grigorieff, Pin 2008

Overview

Logic on words

Equations between languages

Equations between words

Solving equations

- Solve for $x \in \mathbb{R}$: $x^2 + 1 = 0$.

Solving equations

- Solve for $x \in \mathbb{C}$: $x^2 + 1 = 0$.
- A field F is existentially closed if any existential sentence that becomes true in some extension of F already holds in F.

Solving equations

- Solve for $x \in \mathbb{C}$: $x^2 + 1 = 0$.
- A field F is existentially closed if any existential sentence that becomes true in some extension of F already holds in F.
- This property is first order definable for fields!
  - $\exists \overline{x} p(\overline{x}) = 0$ for every non-constant polynomial p.

Solving equations

- Solve for $x \in \mathbb{C}$: $x^2 + 1 = 0$.
- A field F is existentially closed if any existential sentence that becomes true in some extension of F already holds in F.
- This property is first order definable for fields!
  - $\exists \overline{x} p(\overline{x}) = 0$ for every non-constant polynomial p.
- A structure A is existentially closed* if any existential sentence that becomes true in some extension of A already holds in A.

* If the class of structures does not have amalgamation, a more complicated definition is needed.

Solving equations

- Solve for $x \in \mathbb{C}$: $x^2 + 1 = 0$.
- A field F is existentially closed if any existential sentence that becomes true in some extension of F already holds in F.
- This property is first order definable for fields!
  - $\exists \overline{x} p(\overline{x}) = 0$ for every non-constant polynomial p.
- A structure A is existentially closed* if any existential sentence that becomes true in some extension of A already holds in A.
- This property is often first order definable:
  - Linear orders without endpoints: density;
  - Boolean algebras: atomless;
  - Heyting algebras: mimick fields, use uniform interpolation.

* If the class of structures does not have amalgamation, a more complicated definition is needed.

Model companion

A first order theory T^* which captures the existentially closed models for a universal theory T is called a model companion of T.

Theorem.

The theory T^*, if it exists, is the unique theory such that:

1. T and T^* believe the same universal sentences;

2. T^* believes any sentence to be equivalent to an existential sentence.

Robinson, 1963

Model companion

A first order theory T^* which captures the existentially closed models for a universal theory T is called a model companion of T.

Theorem.

The theory T^*, if it exists, is the unique theory such that:

1. T and T^* believe the same universal sentences;

   T and T^* are co-theories

2. T^* believes any sentence to be equivalent to an existential sentence.

   T^* is model complete

Robinson, 1963

Model companions and languages

Theorem.

The first order theory T^* of an algebra for word languages, $\mathcal{P}(\omega)$,

is the model companion of

a theory T of algebras for a linear temporal logic.

Ghilardi & G. JSL 2017

Model companions and languages

Theorem.

The first order theory T^* of an algebra for tree languages, $\mathcal{P}(2^*)$,

is the model companion of

a theory T of algebras for a fair computation tree logic.

Ghilardi & G. LICS 2016

Proof idea: set-up

Skip

- Enrich the Boolean algebra $\mathcal{P}(\omega)$ with temporal operators:
  - $\mathbf{X}a := \{t \in \omega \mid t+1 \in a\}$,
  - $\mathbf{F}a := \{t \in \omega \mid \exists t' \geq t\colon t' \in a\}$,
  - $\mathbf{I} := \{0\}$.

Proof idea: set-up

Skip

- Enrich the Boolean algebra $\mathcal{P}(\omega)$ with temporal operators:
  - $\mathbf{X}a := \{t \in \omega \mid t+1 \in a\}$,
  - $\mathbf{F}a := \{t \in \omega \mid \exists t' \geq t\colon t' \in a\}$,
  - $\mathbf{I} := \{0\}$.

- Axioms for temporal logic $\rightarrow$ a first order theory T.

Proof idea: set-up

Skip

- Enrich the Boolean algebra $\mathcal{P}(\omega)$ with temporal operators:
  - $\mathbf{X}a := \{t \in \omega \mid t+1 \in a\}$,
  - $\mathbf{F}a := \{t \in \omega \mid \exists t' \geq t\colon t' \in a\}$,
  - $\mathbf{I} := \{0\}$.

- Axioms for temporal logic $\to$ a first order theory T.

Theorem. The theory T^* of $\mathcal{P}(\omega)$ is the model companion of T.

i.e., T^* is model complete and T^* is a co-theory of T.

Proof idea: model completeness

- Any first order formula $\varphi(\overline{p})$ in this temporal algebra translates to an MSO formula $\Phi(\overline{P})$ in logic on words.

Proof idea: model completeness

- Any first order formula $\varphi(\overline{p})$ in this temporal algebra translates to an MSO formula $\Phi(\overline{P})$ in logic on words.
- This MSO formula Φ defines a regular language L_φ.

Proof idea: model completeness

- Any first order formula $\varphi(\overline{p})$ in this temporal algebra translates to an MSO formula $\Phi(\overline{P})$ in logic on words.
- This MSO formula Φ defines a regular language L_φ.
- Build an automaton A for Φ.

Proof idea: model completeness

- Any first order formula $\varphi(\overline{p})$ in this temporal algebra translates to an MSO formula $\Phi(\overline{P})$ in logic on words.
- This MSO formula Φ defines a regular language L_φ.
- Build an automaton A for Φ.
- Describe the automaton A with an existential first order formula φ' in the temporal algebra $\mathcal{P}(\omega)$.

Proof idea: model completeness

- Any first order formula $\varphi(\overline{p})$ in this temporal algebra translates to an MSO formula $\Phi(\overline{P})$ in logic on words.
- This MSO formula Φ defines a regular language L_φ.
- Build an automaton A for Φ.
- Describe the automaton A with an existential first order formula φ' in the temporal algebra $\mathcal{P}(\omega)$.
- **Conclusion.** $\mathcal{P}(\omega)$ believes that any first order formula φ is equivalent to an existential formula φ'.

Overview

Logic on words

Equations between languages

Equations between words

Logic on words: example

$$\varphi\colon \exists P\Big[\ P(\texttt{first}) \wedge \neg P(\texttt{last}) \wedge \forall x(P(x) \leftrightarrow \neg P(\texttt{S}(x))\ \Big].$$

- $aaaa \models \varphi$, but $aaaaa \not\models \varphi$.
- $W \models \varphi$ iff W has even length.

Logic on words: example

$$\varphi\colon \exists P\Big[\ P(\texttt{first}) \wedge \neg P(\texttt{last}) \wedge \forall x(P(x) \leftrightarrow \neg P(\texttt{S}(x))\ \Big].$$

- $aaaa \models \varphi$, but $aaaaa \not\models \varphi$.
- $W \models \varphi$ iff W has even length.

Question. Does an equivalent first order formula exist for φ?

Logic and monoids

A language $L \subseteq \Sigma^*$ is MSO-definable

if, and only if,

L is invariant under a finite index monoid congruence.

Logic and monoids

A language $L \subseteq \Sigma^*$ is FO-definable

if, and only if,

L is invariant under a finite index aperiodic monoid congruence.

Logic and monoids

A language $L \subseteq \Sigma^*$ is FO-definable

if, and only if,

L is invariant under a finite index aperiodic monoid congruence.

A congruence θ on Σ^* is called aperiodic if Σ^*/θ does not have non-trivial subgroups.

Schützenberger 1965; McNaughton, Papert 1971

ω

In a finite monoid, any element x has a unique idempotent, x^ω, in its orbit $\{x, x^2, x^3, \dots\}$.

A finite monoid is aperiodic iff it satisfies the equation

$$x^\omega = x^\omega x.$$

ω

In a profinite monoid, any element x has a unique idempotent, x^ω, in its orbit-closure $\overline{\{x, x^2, x^3, \dots\}}$.

A profinite monoid is aperiodic iff it satisfies the equation

$$x^\omega = x^\omega x.$$

ω

In a profinite monoid, any element x has a unique idempotent, x^ω, in its orbit-closure $\overline{\{x, x^2, x^3, \dots\}}$.

A profinite monoid is aperiodic iff it satisfies the equation

$$x^\omega = x^\omega x.$$

Decision problem. Given two terms in $\cdot$ and $()^\omega$, are they equal in every finite aperiodic monoid?

ω

In a profinite monoid, any element x has a unique idempotent, x^ω, in its orbit-closure $\overline{\{x, x^2, x^3, \dots\}}$.

A profinite monoid is aperiodic iff it satisfies the equation

$$x^\omega = x^\omega x.$$

Decision problem. Given two terms in $\cdot$ and $()^\omega$, are they equal in the free profinite aperiodic monoid?

The free profinite aperiodic monoid

Theorem.

The free profinite aperiodic monoid

=

The topological monoid of ultrafilters of FO-definable languages

=

The topological monoid of $\equiv_{FO}$-classes of pseudo-finite words.

The free profinite aperiodic monoid

Theorem.

The free profinite aperiodic monoid

=

The topological monoid of ultrafilters of FO-definable languages

=

The topological monoid of $\equiv_{FO}$-classes of pseudo-finite words.

Using a concrete interpretation of $\cdot$ and $()^\omega$ via saturated models, the decision problem reduces to checking isomorphism of specific pseudo-finite words.

G. & Steinberg STACS 2017

The future

- Language equations and unification
- Word equations and unification
- From FO to MSO