Pointlike sets for varieties determined by groups

Sam van Gool

DIAMANT Symposium

Veenendaal, November 2018

Supported by EU Marie Curie grant no. 655941

Regular languages and profinite semigroups

Separation problems and pointlike sets

New result

Proof techniques

Regular languages and profinite semigroups

Separation problems and pointlike sets

New result

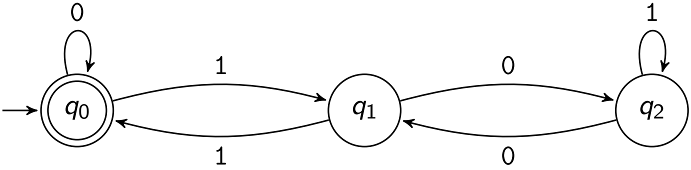
Proof techniques

Regular languages: example

- A programming problem: given a natural number in binary, $w \in \{0, 1\}^+$, determine whether or not w is divisible by 3.

Regular languages: example

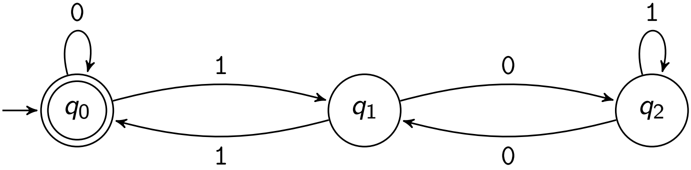
- A programming problem: given a natural number in binary, $w \in \{0, 1\}^+$, determine whether or not w is divisible by 3.
- Solution 1: a (deterministic) automaton A:



Answer yes iff A accepts w.

Regular languages: example

- A programming problem: given a natural number in binary, $w \in \{0,1\}^+$, determine whether or not w is divisible by 3.
- Solution 1: a (deterministic) automaton A:



Answer yes iff A accepts w.

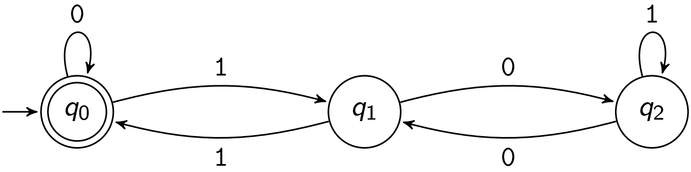
- Solution 2: a regular expression

$$R := 0^*(1(01^*0)^*1)^*0^*$$

Answer yes iff w matches the expression R.

Regular languages: example

- A programming problem: given a natural number in binary, $w \in \{0,1\}^+$, determine whether or not w is divisible by 3.
- Solution 1: a (deterministic) automaton A:



Answer yes iff A accepts w.

- Solution 3: a homomorphism $\phi\colon \{0,1\}^+ \to S_3$

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

Answer yes iff the permutation $\phi(w)$ fixes 0.

Regular languages

Let Σ be a finite alphabet.

Theorem (Kleene, 1950s)

For any language $L \subseteq \Sigma^+$, the following are equivalent:

Regular languages

Let Σ be a finite alphabet.

Theorem (Kleene, 1950s)

For any language $L \subseteq \Sigma^+$, the following are equivalent:

1. *the language L is accepted by a (deterministic) finite automaton;*

Regular languages

Let Σ be a finite alphabet.

Theorem (Kleene, 1950s)

For any language $L \subseteq \Sigma^+$, the following are equivalent:

1. *the language L is accepted by a (deterministic) finite automaton;*
2. *the language L is described by a regular expression, built from Σ, $()^*$, $\cdot$, $\cup$ (and complementation);*

Regular languages

Let Σ be a finite alphabet.

Theorem (Kleene, 1950s)

For any language $L \subseteq \Sigma^+$, the following are equivalent:

1. *the language L is accepted by a (deterministic) finite automaton;*
2. *the language L is described by a regular expression, built from Σ, $()^*$, $\cdot$, $\cup$ (and complementation);*
3. *the language L is recognized by a homomorphism to a finite semigroup;*

Regular languages

Let Σ be a finite alphabet.

Theorem (Kleene, 1950s)

For any language $L \subseteq \Sigma^+$, the following are equivalent:

1. *the language L is accepted by a (deterministic) finite automaton;*
2. *the language L is described by a regular expression, built from Σ, $()^*$, $\cdot$, $\cup$ (and complementation);*
3. *the language L is recognized by a homomorphism to a finite semigroup;*
4. *the language L is definable in monadic second order logic.*

Regular languages

Let Σ be a finite alphabet.

Theorem (Kleene, 1950s)

For any language $L \subseteq \Sigma^+$, the following are equivalent:

1. *the language L is accepted by a (deterministic) finite automaton;*
2. *the language L is described by a regular expression, built from Σ, $()^*$, $\cdot$, $\cup$ (and complementation);*
3. *the language L is recognized by a homomorphism to a finite semigroup;*
4. *the language L is definable in monadic second order logic.*

Moreover, there is a computable 'minimal' such semigroup, called the syntactic semigroup of L.

Star-free languages

- A meta-problem: given a regular expression R, determine whether or not there is a star-free regular expression R' which describes the same language: $L(R) = L(R')$.

Star-free languages

- A meta-problem: given a regular expression R, determine whether or not there is a star-free regular expression R' which describes the same language: $L(R) = L(R')$.

Theorem (Schützenberger, 1960s)

The following are equivalent, for any language L:

1. *the language L is accepted by a counter-free finite automaton;*

Star-free languages

- A meta-problem: given a regular expression R, determine whether or not there is a star-free regular expression R' which describes the same language: $L(R) = L(R')$.

Theorem (Schützenberger, 1960s)

The following are equivalent, for any language L:

1. *the language L is accepted by a counter-free finite automaton;*
2. *the language L is described by a star-free expression;*

Star-free languages

- A meta-problem: given a regular expression R, determine whether or not there is a star-free regular expression R' which describes the same language: $L(R) = L(R')$.

Theorem (Schützenberger, 1960s)

The following are equivalent, for any language L:

1. *the language L is accepted by a counter-free finite automaton;*
2. *the language L is described by a star-free expression;*
3. *the language L is recognized by a homomorphism to a finite semigroup any subgroup of which is trivial;*

Star-free languages

- A meta-problem: given a regular expression R, determine whether or not there is a star-free regular expression R' which describes the same language: $L(R) = L(R')$.

Theorem (Schützenberger, 1960s)

The following are equivalent, for any language L:

1. *the language L is accepted by a counter-free finite automaton;*
2. *the language L is described by a star-free expression;*
3. *the language L is recognized by a homomorphism to a finite semigroup any subgroup of which is trivial;*
4. *the language L is definable in first order logic.*

Star-free languages

- A meta-problem: given a regular expression R, determine whether or not there is a star-free regular expression R' which describes the same language: $L(R) = L(R')$.

Theorem (Schützenberger, 1960s)

The following are equivalent, for any language L:

1. *the language L is accepted by a counter-free finite automaton;*
2. *the language L is described by a star-free expression;*
3. *the language L is recognized by a homomorphism to a finite semigroup any subgroup of which is trivial;*
4. *the language L is definable in first order logic.*

- Schützenberger's Theorem solves the membership problem for the class of star-free languages.

Star-free languages

- A meta-problem: given a regular expression R, determine whether or not there is a star-free regular expression R' which describes the same language: $L(R) = L(R')$.

Theorem (Schützenberger, 1960s)

The following are equivalent, for any language L:

1. *the language L is accepted by a counter-free finite automaton;*
2. *the language L is described by a star-free expression;*
3. *the language L is recognized by a homomorphism to a finite semigroup any subgroup of which is trivial;*
4. *the language L is definable in first order logic.*

- Schützenberger's Theorem solves the membership problem for the class of star-free languages.
- A semigroup as in (3) is called aperiodic.

Varieties

- A class of finite semigroups $\mathbf{V}$ is a variety if it is closed under finite products, homomorphic images, and subsemigroups.
- If $\mathbf{V}$ is a variety and Σ is a finite alphabet, $\mathcal{V}(\Sigma)$ denotes the set of languages $L \subseteq \Sigma^+$ with syntactic semigroup in $\mathbf{V}$.
- The indexed collection $\mathcal{V} \colon \Sigma \mapsto \mathcal{V}(\Sigma)$ is a presheaf of Boolean algebras.

Varieties

- A class of finite semigroups $\mathbf{V}$ is a variety if it is closed under finite products, homomorphic images, and subsemigroups.
- If $\mathbf{V}$ is a variety and Σ is a finite alphabet, $\mathcal{V}(\Sigma)$ denotes the set of languages $L \subseteq \Sigma^+$ with syntactic semigroup in $\mathbf{V}$.
- The indexed collection $\mathcal{V} \colon \Sigma \mapsto \mathcal{V}(\Sigma)$ is a presheaf of Boolean algebras.

Theorem (Gehrke, Grigorieff, Pin 2010)

The enriched Stone dual space of ultrafilters of $\mathcal{V}(\Sigma)$ coincides with the free pro-$\mathbf{V}$ semigroup generated by Σ.

Varieties

- A class of finite semigroups $\mathbf{V}$ is a variety if it is closed under finite products, homomorphic images, and subsemigroups.
- If $\mathbf{V}$ is a variety and Σ is a finite alphabet, $\mathcal{V}(\Sigma)$ denotes the set of languages $L \subseteq \Sigma^+$ with syntactic semigroup in $\mathbf{V}$.
- The indexed collection $\mathcal{V} \colon \Sigma \mapsto \mathcal{V}(\Sigma)$ is a presheaf of Boolean algebras.

Theorem (Gehrke, Grigorieff, Pin 2010)

The enriched Stone dual space of ultrafilters of $\mathcal{V}(\Sigma)$ coincides with the free pro-$\mathbf{V}$ semigroup generated by Σ.

- The free profinite semigroup, $\widehat{\Sigma^+}$, maps onto the free pro-$\mathbf{V}$ semigroup with a map $\pi_{\mathbf{V}} \colon \widehat{\Sigma^+} \twoheadrightarrow \widehat{F}_{\mathbf{V}}(\Sigma)$.

Generalizing the membership problem

- An interpolation problem: given regular expressions R_1, R_2, determine whether there exists a star-free expression R such that $L(R_1) \subseteq L(R) \subseteq L(R_2)$.

Generalizing the membership problem

- An interpolation problem: given regular expressions R_1, R_2, determine whether there exists a star-free expression R such that $L(R_1) \subseteq L(R) \subseteq L(R_2)$.
- Equivalently, a separation problem: determine whether there exists a star-free expression R such that $L(R_1) \subseteq L(R)$ and $L(R_2) \cap L(R) = \emptyset$.

Generalizing the membership problem

- An interpolation problem: given regular expressions R_1, R_2, determine whether there exists a star-free expression R such that $L(R_1) \subseteq L(R) \subseteq L(R_2)$.
- Equivalently, a separation problem: determine whether there exists a star-free expression R such that $L(R_1) \subseteq L(R)$ and $L(R_2) \cap L(R) = \emptyset$.

Theorem

The separation problem for star-free languages is decidable.

Generalizing the membership problem

- An interpolation problem: given regular expressions R_1, R_2, determine whether there exists a star-free expression R such that $L(R_1) \subseteq L(R) \subseteq L(R_2)$.
- Equivalently, a separation problem: determine whether there exists a star-free expression R such that $L(R_1) \subseteq L(R)$ and $L(R_2) \cap L(R) = \emptyset$.

Theorem

The separation problem for star-free languages is decidable.

- The proof (Henckell 1988) translates the problem to a combinatorial question about a finite semigroup, namely, to compute its aperiodic-pointlike sets.

Pointlike sets

Proposition

Let $\mathbf{V}$ *be a variety of finite semigroups, and let* S *be a finite semigroup. For any subset* X *of* S*, the following are equivalent:*

1. *there exist profinite words* $x_1, \ldots, x_n \in \widehat{S^+}$ *such that* $X = \{[x_1]_S, \ldots, [x_n]_S\}$ *and* $\pi_{\mathbf{V}}(x_1) = \cdots = \pi_{\mathbf{V}}(x_n)$*;*

Pointlike sets

Proposition

Let $\mathbf{V}$ *be a variety of finite semigroups, and let* S *be a finite semigroup. For any subset* X *of* S*, the following are equivalent:*

1. *there exist profinite words* $x_1, \ldots, x_n \in \widehat{S^+}$ *such that* $X = \{[x_1]_S, \ldots, [x_n]_S\}$ *and* $\pi_{\mathbf{V}}(x_1) = \cdots = \pi_{\mathbf{V}}(x_n)$*;*
2. *for every relational morphism* $\rho \colon S \to T$ *with* $T \in \mathbf{V}$*, there exists* $t \in T$ *such that* $x \rho t$ *for all* $x \in X$*.*

A relational morphism is a subsemigroup $\rho \subseteq S \times T$ with $s\rho \neq \emptyset$ for all $s \in S$.

Pointlike sets

Proposition

Let $\mathbf{V}$ *be a variety of finite semigroups, and let* S *be a finite semigroup. For any subset* X *of* S*, the following are equivalent:*

1. *there exist profinite words* $x_1, \ldots, x_n \in \widehat{S^+}$ *such that* $X = \{[x_1]_S, \ldots, [x_n]_S\}$ *and* $\pi_{\mathbf{V}}(x_1) = \cdots = \pi_{\mathbf{V}}(x_n)$*;*
2. *for every relational morphism* $\rho \colon S \to T$ *with* $T \in \mathbf{V}$*, there exists* $t \in T$ *such that* $x \rho t$ *for all* $x \in X$*.*

A relational morphism is a subsemigroup $\rho \subseteq S \times T$ with $s\rho \neq \emptyset$ for all $s \in S$.

The set X is called a $\mathbf{V}$-pointlike subset of S if the conditions in the proposition are satisfied.

Pointlike sets

Proposition

Let $\mathbf{V}$ *be a variety of finite semigroups, and let* S *be a finite semigroup. For any subset* X *of* S*, the following are equivalent:*

1. *there exist profinite words* $x_1, \ldots, x_n \in \widehat{S^+}$ *such that* $X = \{[x_1]_S, \ldots, [x_n]_S\}$ *and* $\pi_{\mathbf{V}}(x_1) = \cdots = \pi_{\mathbf{V}}(x_n)$*;*
2. *for every relational morphism* $\rho\colon S \to T$ *with* $T \in \mathbf{V}$*, there exists* $t \in T$ *such that* $x\rho t$ *for all* $x \in X$*.*

A relational morphism is a subsemigroup $\rho \subseteq S \times T$ with $s\rho \neq \emptyset$ for all $s \in S$.

The set X is called a $\mathbf{V}$-pointlike subset of S if the conditions in the proposition are satisfied.

Example. Any subgroup G of a finite semigroup S is $\mathbf{A}$-pointlike.

Separation and pointlike sets

Proposition

Let $\mathbf{V}$ *be a variety. For any regular languages* $L_1, L_2 \subseteq \Sigma^+$*, the following are equivalent:*

1. L_1 *is not separable from* L_2 *by a language in* $\mathcal{V}(\Sigma)$*;*

Separation and pointlike sets

Proposition

Let $\mathbf{V}$ *be a variety. For any regular languages* $L_1, L_2 \subseteq \Sigma^+$*, the following are equivalent:*

1. L_1 *is* *not* *separable from* L_2 *by a language in* $\mathcal{V}(\Sigma)$*;*
2. *there exist profinite words* $w_1, w_2 \in \widehat{\Sigma^+}$*, with* $w_i \in \mathrm{cl}(L_i)$*, such that* $\pi_{\mathbf{V}}(w_1) = \pi_{\mathbf{V}}(w_2)$*;*

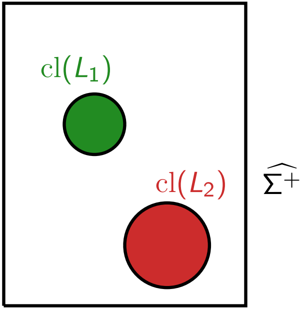
Separation and pointlike sets

Proposition

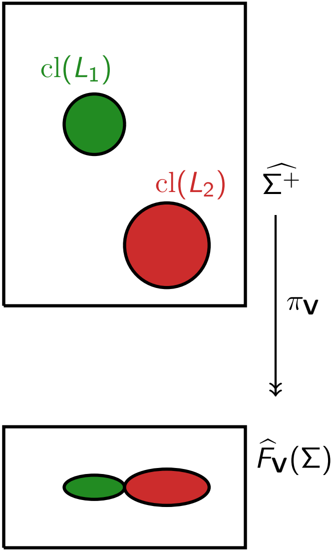
Let $\mathbf{V}$ be a variety. For any regular languages $L_1, L_2 \subseteq \Sigma^+$, the following are equivalent:

1. *L_1 is not separable from L_2 by a language in $\mathcal{V}(\Sigma)$;*
2. *there exist profinite words $w_1, w_2 \in \widehat{\Sigma^+}$, with $w_i \in \mathrm{cl}(L_i)$, such that $\pi_{\mathbf{V}}(w_1) = \pi_{\mathbf{V}}(w_2)$;*
3. *for any homomorphism ϕ from Σ^+ to a finite semigroup which recognizes L_1 and L_2, there exist $s_i \in \phi(L_i)$ such that the set $\{s_1, s_2\}$ is $\mathbf{V}$-pointlike.*

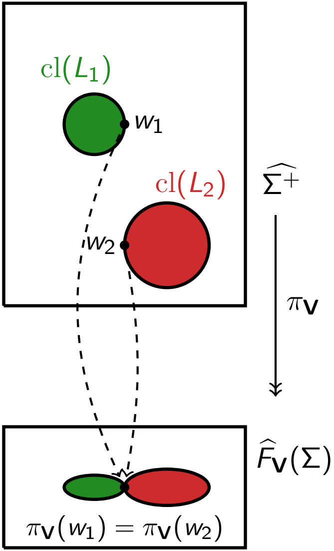
Separation and pointlike sets



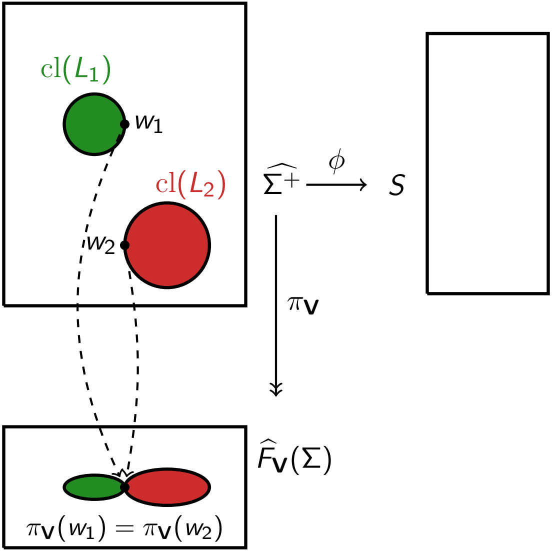
Separation and pointlike sets



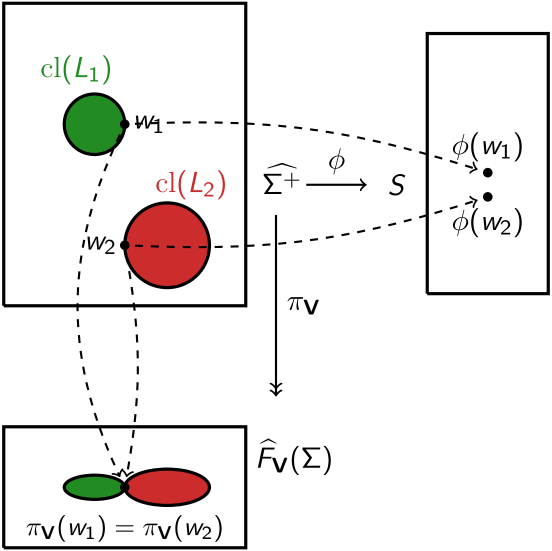
Separation and pointlike sets



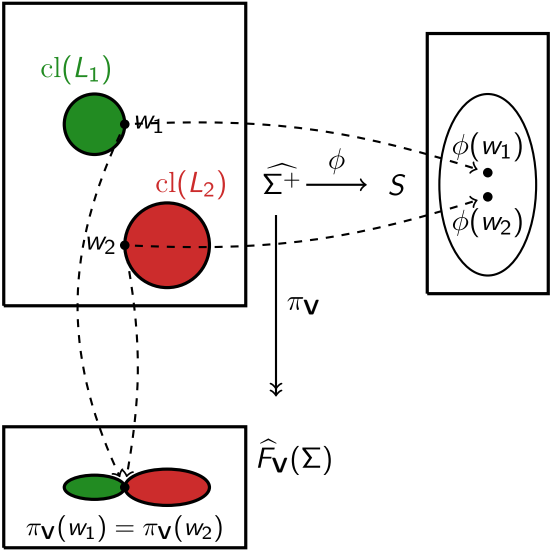
Separation and pointlike sets



Separation and pointlike sets



Separation and pointlike sets



Separation and pointlike sets

Proposition

Let $\mathbf{V}$ *be a variety. For any regular languages* $L_1, L_2 \subseteq \Sigma^+$*, the following are equivalent:*

1. L_1 *is not separable from* L_2 *by a language in* $\mathcal{V}(\Sigma)$*;*
2. *there exist profinite words* $w_1, w_2 \in \widehat{\Sigma^+}$*, with* $w_i \in \mathrm{cl}(L_i)$*, such that* $\pi_{\mathbf{V}}(w_1) = \pi_{\mathbf{V}}(w_2)$*;*
3. *for any homomorphism* ϕ *from* Σ^+ *to a finite semigroup which recognizes* L_1 *and* L_2*, there exist* $s_i \in \phi(L_i)$ *such that the set* $\{s_1, s_2\}$ *is* $\mathbf{V}$*-pointlike.*

Theorem (Henckell, 1988)

The **A***-pointlike sets of a finite semigroup are computable, where* **A** *is the variety of aperiodic semigroups.*

Theorem (Henckell, 1988)

The **A***-pointlike sets of a finite semigroup are computable, where* **A** *is the variety of aperiodic semigroups.*

- Thus, the separation problem for star-free languages is decidable.

Benjamin Steinberg

January 16 ·

Hard to believe I'm still working on pointlike sets after 20 years but...

ARXIV.ORG

[1801.04638] Pointlike sets for varieties determined by groups

Like Comment Share

Joint work with Benjamin Steinberg (CCNY).

Generalizing aperiodic semigroups

- Aperiodic semigroup = all subgroups trivial = iterated semi-direct product of semilattices.

Generalizing aperiodic semigroups

- Aperiodic semigroup = all subgroups trivial = iterated semi-direct product of semilattices.

Theorem (Krohn-Rhodes Decomposition)

Any finite semigroup divides an iterated semi-direct product of finite semilattices and finite simple groups.

Generalizing aperiodic semigroups

- Aperiodic semigroup = all subgroups trivial = iterated semi-direct product of semilattices.

Theorem (Krohn-Rhodes Decomposition)

Any finite semigroup divides an iterated semi-direct product of finite semilattices and finite simple groups.

- For a variety of finite groups $\mathbf{H}$, define

$$\overline{\mathbf{H}} := \{S \text{ finite semigroup} : \text{all subgroups of } S \text{ are in } \mathbf{H}\}.$$

Theorem (G. & Steinberg 2018)

For any variety of finite groups $\mathbf{H}$ *with decidable membership, the* $\overline{\mathbf{H}}$*-pointlike sets are computable, and thus, in particular, the separation problem for* $\overline{\mathbf{H}}$*-languages is decidable.*

Corollaries

- $\mathbf{H} = \mathbf{1}$, the variety containing only the trivial group.
  $\Rightarrow$ Aperiodic-pointlikes computable (Henckell 1988)

Corollaries

- $\mathbf{H} = \mathbf{1}$, the variety containing only the trivial group.
  $\Rightarrow$ Aperiodic-pointlikes computable (Henckell 1988)
- $\mathbf{H} = \mathbf{G}_\pi$, the variety of groups whose order is divisible only by primes in a computable set π.
  $\Rightarrow$ $\overline{\mathbf{G}_\pi}$-pointlikes computable (Henckell, Rhodes, Steinberg 2010)

Corollaries

- $\mathbf{H} = \mathbf{1}$, the variety containing only the trivial group.
  $\Rightarrow$ Aperiodic-pointlikes computable (Henckell 1988)
- $\mathbf{H} = \mathbf{G}_\pi$, the variety of groups whose order is divisible only by primes in a computable set π.
  $\Rightarrow$ $\overline{\mathbf{G}_\pi}$-pointlikes computable (Henckell, Rhodes, Steinberg 2010)
- $\mathbf{H} = \mathbf{G}_{\mathrm{sol}}$, the variety of solvable groups. Semigroups in $\overline{\mathbf{G}_{\mathrm{sol}}}$ are called solvable semigroups.
  $\Rightarrow$ $\overline{\mathbf{G}_{\mathrm{sol}}}$-pointlikes computable (G. & Steinberg 2018)

Regular languages and profinite semigroups

Separation problems and pointlike sets

New result

Proof techniques

Computing pointlikes

Proposition

Let $\mathbf{V}$ *be a variety of finite semigroups.*

The set of $\mathbf{V}$*-pointlikes,* $\mathrm{PL}_{\mathbf{V}}(S)$*, of a finite semigroup* S*, is:*

- *a subsemigroup of* $\mathcal{P}(S)$*:* $X \cdot Y$ *is* $\mathbf{V}$*-pointlike if* X *and* Y *are,*
- *a down-set: if* X *is* $\mathbf{V}$*-pointlike then so is any non-empty subset of* X*.*

Computing pointlikes

Proposition

Let $\mathbf{V}$ *be a variety of finite semigroups.*

The set of $\mathbf{V}$*-pointlikes,* $\mathrm{PL}_{\mathbf{V}}(S)$*, of a finite semigroup* S*, is:*

- *a subsemigroup of* $\mathcal{P}(S)$*:* $X \cdot Y$ *is* $\mathbf{V}$*-pointlike if* X *and* Y *are,*
- *a down-set: if* X *is* $\mathbf{V}$*-pointlike then so is any non-empty subset of* X*.*

Moreover, $\mathrm{PL}_{\mathbf{V}}$ *is a submonad of* $\mathcal{P}$*:*

- *singletons are* $\mathbf{V}$*-pointlike,*
- *the union* $\bigcup \mathcal{X}$ *of any* $\mathbf{V}$*-pointlike subset* $\mathcal{X}$ *of the semigroup* $\mathrm{PL}_{\mathbf{V}}(S)$ *is* $\mathbf{V}$*-pointlike.*

Computing $\overline{\mathbf{H}}$-pointlikes

Definition

The **H**-kernel of a group G, $K_{\mathbf{H}}(G)$, is the smallest normal subgroup N of G such that G/N belongs to $\mathbf{H}$.

Computing $\overline{\mathbf{H}}$-pointlikes

Definition

The $\mathbf{H}$-kernel of a group G, $K_{\mathbf{H}}(G)$, is the smallest normal subgroup N of G such that G/N belongs to $\mathbf{H}$.

Definition

Let T be a finite semigroup. The $\overline{\mathbf{H}}$-saturation of T is the smallest downward closed subsemigroup S of $\mathcal{P}(T)$ containing all singletons $\{t\}$ $(t \in T)$, such that $\bigcup K_{\mathbf{H}}(G) \in S$ for any subgroup G of S.

Computing $\overline{\mathbf{H}}$-pointlikes

Definition

The $\mathbf{H}$-kernel of a group G, $K_{\mathbf{H}}(G)$, is the smallest normal subgroup N of G such that G/N belongs to $\mathbf{H}$.

Definition

Let T be a finite semigroup. The $\overline{\mathbf{H}}$-saturation of T is the smallest downward closed subsemigroup S of $\mathcal{P}(T)$ containing all singletons $\{t\}$ $(t \in T)$, such that $\bigcup K_{\mathbf{H}}(G) \in S$ for any subgroup G of S.

Note. $\overline{\mathbf{H}}$-saturations are computable if $\mathbf{H}$ is decidable.

Computing $\overline{\mathbf{H}}$-pointlikes

Definition

The $\mathbf{H}$-kernel of a group G, $K_{\mathbf{H}}(G)$, is the smallest normal subgroup N of G such that G/N belongs to $\mathbf{H}$.

Definition

Let T be a finite semigroup. The $\overline{\mathbf{H}}$-saturation of T is the smallest downward closed subsemigroup S of $\mathcal{P}(T)$ containing all singletons $\{t\}$ $(t \in T)$, such that $\bigcup K_{\mathbf{H}}(G) \in S$ for any subgroup G of S.

Note. $\overline{\mathbf{H}}$-saturations are computable if $\mathbf{H}$ is decidable.

Theorem (Computation of $\overline{\mathbf{H}}$-pointlikes)

Let T be a finite semigroup. A subset X of T is $\overline{\mathbf{H}}$-pointlike if, and only if, X belongs to the $\overline{\mathbf{H}}$-saturation of T.

Computing $\overline{\mathbf{H}}$-pointlikes

Definition

The $\mathbf{H}$-kernel of a group G, $K_{\mathbf{H}}(G)$, is the smallest normal subgroup N of G such that G/N belongs to $\mathbf{H}$.

Definition

Let T be a finite semigroup. The $\overline{\mathbf{H}}$-saturation of T is the smallest downward closed subsemigroup S of $\mathcal{P}(T)$ containing all singletons $\{t\}$ $(t \in T)$, such that $\bigcup K_{\mathbf{H}}(G) \in S$ for any subgroup G of S.

Note. $\overline{\mathbf{H}}$-saturations are computable if $\mathbf{H}$ is decidable.

Theorem (Computation of $\overline{\mathbf{H}}$-pointlikes)

Let T be a finite semigroup. A subset X of T is $\overline{\mathbf{H}}$-pointlike if, and only if, X belongs to the $\overline{\mathbf{H}}$-saturation of T.

Difficult direction: every pointlike set is in the saturation.

Completeness of the algorithm

- Write S for the $\overline{\mathbf{H}}$-saturation of T.

Completeness of the algorithm

- Write S for the $\overline{\mathbf{H}}$-saturation of T.
- To show that every pointlike set of T lies in S, we need to expand the semigroup S to a semigroup $S^{\mathbf{H}}$ such that:

Completeness of the algorithm

- Write S for the $\overline{\mathbf{H}}$-saturation of T.
- To show that every pointlike set of T lies in S, we need to expand the semigroup S to a semigroup $S^{\mathbf{H}}$ such that:
  - $S^{\mathbf{H}}$ lies in $\overline{\mathbf{H}}$, and

Completeness of the algorithm

- Write S for the $\overline{\mathbf{H}}$-saturation of T.
- To show that every pointlike set of T lies in S, we need to expand the semigroup S to a semigroup $S^{\mathbf{H}}$ such that:
  - $S^{\mathbf{H}}$ lies in $\overline{\mathbf{H}}$, and
  - there is a relational morphism $\rho\colon T \to S^{\mathbf{H}}$ such that $\rho^{-1}(\sigma) \in S$ for every $\sigma \in S^{\mathbf{H}}$.

Completeness of the algorithm

- Write S for the $\overline{\mathbf{H}}$-saturation of T.
- To show that every pointlike set of T lies in S, we need to expand the semigroup S to a semigroup $S^{\mathbf{H}}$ such that:
  - $S^{\mathbf{H}}$ lies in $\overline{\mathbf{H}}$, and
  - there is a relational morphism $\rho\colon T \to S^{\mathbf{H}}$ such that $\rho^{-1}(\sigma) \in S$ for every $\sigma \in S^{\mathbf{H}}$.
- Ingredients for building $S^{\mathbf{H}}$ and ρ:

Completeness of the algorithm

- Write S for the $\overline{\mathbf{H}}$-saturation of T.
- To show that every pointlike set of T lies in S, we need to expand the semigroup S to a semigroup $S^{\mathbf{H}}$ such that:
  - $S^{\mathbf{H}}$ lies in $\overline{\mathbf{H}}$, and
  - there is a relational morphism $\rho\colon T \to S^{\mathbf{H}}$ such that $\rho^{-1}(\sigma) \in S$ for every $\sigma \in S^{\mathbf{H}}$.
- Ingredients for building $S^{\mathbf{H}}$ and ρ:
  - a blow-up operator on S;
  - an action on strict L-chains of $\mathbf{H}$-elements in S;

Completeness of the algorithm

- Write S for the $\overline{\mathbf{H}}$-saturation of T.
- To show that every pointlike set of T lies in S, we need to expand the semigroup S to a semigroup $S^{\mathbf{H}}$ such that:
  - $S^{\mathbf{H}}$ lies in $\overline{\mathbf{H}}$, and
  - there is a relational morphism $\rho\colon T \to S^{\mathbf{H}}$ such that $\rho^{-1}(\sigma) \in S$ for every $\sigma \in S^{\mathbf{H}}$.
- Ingredients for building $S^{\mathbf{H}}$ and ρ:
  - a blow-up operator on S;
  - an action on strict L-chains of $\mathbf{H}$-elements in S;
- Hardest part: showing that $S^{\mathbf{H}}$ lies in $\overline{\mathbf{H}}$.

L-chains of $\mathbf{H}$-elements

- For elements s, t of a semigroup S, we write $s \leq_L t$ when $s = \alpha t$ for some $\alpha \in S \cup \{1\}$, i.e., 't is a suffix of s'.

L-chains of **H**-elements

- For elements s, t of a semigroup S, we write $s \leq_L t$ when $s = \alpha t$ for some $\alpha \in S \cup \{1\}$, i.e., 't is a suffix of s'.
- The induced equivalence relation is called L and L_u denotes the L-equivalence class of u.

L-chains of **H**-elements

- For elements s, t of a semigroup S, we write $s \leq_L t$ when $s = \alpha t$ for some $\alpha \in S \cup \{1\}$, i.e., 't is a suffix of s'.
- The induced equivalence relation is called L and L_u denotes the L-equivalence class of u.
- Every element $s \in S$ has a group Γ_s associated to it, its Schützenberger group, which is the faithful quotient of the right permutation action on L_s by its stabilizer.

L-chains of $\mathbf{H}$-elements

- For elements s, t of a semigroup S, we write $s \leq_L t$ when $s = \alpha t$ for some $\alpha \in S \cup \{1\}$, i.e., 't is a suffix of s'.
- The induced equivalence relation is called L and L_u denotes the L-equivalence class of u.
- Every element $s \in S$ has a group Γ_s associated to it, its Schützenberger group, which is the faithful quotient of the right permutation action on L_s by its stabilizer.
- Call $s \in S$ an $\mathbf{H}$-element if Γ_s lies in $\mathbf{H}$.

A blow-up operator on S

Proposition

There exists an idempotent operation $b\colon S \to S$ which fixes exactly the $\mathbf{H}$-elements, is $\leq_L$-contracting and $\subseteq$-expanding, i.e., for all $s \in S$, $b(s) \leq_L s$ and $s \subseteq b(s)$.

A blow-up operator on S

Proposition

There exists an idempotent operation $b\colon S \to S$ which fixes exactly the **H**-*elements, is $\leq_L$-contracting and $\subseteq$-expanding, i.e., for all $s \in S$, $b(s) \leq_L s$ and $s \subseteq b(s)$.*

Proof.

- For every $s \in S$, there is a subgroup G_s of S with quotient Γ_s.

A blow-up operator on S

Proposition

There exists an idempotent operation $b \colon S \to S$ which fixes exactly the $\mathbf{H}$*-elements, is* $\leq_L$*-contracting and* $\subseteq$*-expanding, i.e., for all* $s \in S$, $b(s) \leq_L s$ *and* $s \subseteq b(s)$.

Proof.

- For every $s \in S$, there is a subgroup G_s of S with quotient Γ_s.
- Define $b_0(s) := (\bigcup K_{\mathbf{H}}(G_s)) \cdot s$.

A blow-up operator on S

Proposition

There exists an idempotent operation $b\colon S \to S$ which fixes exactly the $\mathbf{H}$-elements, is $\leq_L$-contracting and $\subseteq$-expanding, i.e., for all $s \in S$, $b(s) \leq_L s$ and $s \subseteq b(s)$.

Proof.

- For every $s \in S$, there is a subgroup G_s of S with quotient Γ_s.
- Define $b_0(s) := (\bigcup K_{\mathbf{H}}(G_s)) \cdot s$.
- Composing b_0 sufficiently often with itself yields an idempotent operation b. $\square$

The semigroup $S^{\mathbf{H}}$

- The original semigroup T acts on the finite set of strict L-chains of $\mathbf{H}$-elements of S.

The semigroup $S^{\mathbf{H}}$

- The original semigroup T acts on the finite set of strict L-chains of **H**-elements of S.
- For any $t \in T$ and $\mathbf{q}$ a strict L-chain of **H**-elements:
  1. multiply every item in the chain $\mathbf{q}$ by t and add $\{t\}$ in front,
  2. recursively apply the blow-up operator b to the chain,
  3. 'pop' $\mathcal{L}$-equivalent elements to obtain a strict chain $\sigma_t(\mathbf{q})$.

The semigroup $S^{\mathbf{H}}$

- The original semigroup T acts on the finite set of strict L-chains of **H**-elements of S.
- For any $t \in T$ and $\mathbf{q}$ a strict L-chain of **H**-elements:
  1. multiply every item in the chain $\mathbf{q}$ by t and add $\{t\}$ in front,
  2. recursively apply the blow-up operator b to the chain,
  3. 'pop' $\mathcal{L}$-equivalent elements to obtain a strict chain $\sigma_t(\mathbf{q})$.
- Let $S^{\mathbf{H}}$ be the semigroup generated by the functions σ_t, $t \in T$.

The semigroup $S^{\mathbf{H}}$

- The original semigroup T acts on the finite set of strict L-chains of $\mathbf{H}$-elements of S.
- For any $t \in T$ and $\mathbf{q}$ a strict L-chain of $\mathbf{H}$-elements:
  1. multiply every item in the chain $\mathbf{q}$ by t and add $\{t\}$ in front,
  2. recursively apply the blow-up operator b to the chain,
  3. 'pop' $\mathcal{L}$-equivalent elements to obtain a strict chain $\sigma_t(\mathbf{q})$.
- Let $S^{\mathbf{H}}$ be the semigroup generated by the functions σ_t, $t \in T$.

Theorem

The semigroup $S^{\mathbf{H}}$ is a quotient of a subsemigroup of an infinite wreath product acting on S^*, which lies in $\overline{\mathbf{H}}$.

Final remarks

- We showed that pointlikes are computable and separation decidable for $\overline{\mathbf{H}}$ whenever $\mathbf{H}$ is a decidable variety of groups.

Final remarks

- We showed that pointlikes are computable and separation decidable for $\overline{\mathbf{H}}$ whenever $\mathbf{H}$ is a decidable variety of groups.
- Computability of $\overline{\mathbf{H}}$-pointlikes implies decidability of membership for more involved varieties built from $\overline{\mathbf{H}}$; one may studies these algorithms for specific choices of $\mathbf{H}$.

Final remarks

- We showed that pointlikes are computable and separation decidable for $\overline{\mathbf{H}}$ whenever $\mathbf{H}$ is a decidable variety of groups.
- Computability of $\overline{\mathbf{H}}$-pointlikes implies decidability of membership for more involved varieties built from $\overline{\mathbf{H}}$; one may studies these algorithms for specific choices of $\mathbf{H}$.
- If the variety $\mathbf{H}$ is defined by profinite identities (e.g., trivial, abelian, solvable), these can be used to obtain a faster algorithm for computing $\overline{\mathbf{H}}$-pointlikes than the 'generic' saturation algorithm via $\mathbf{H}$-kernels.

Final remarks

- We showed that pointlikes are computable and separation decidable for $\overline{\mathbf{H}}$ whenever $\mathbf{H}$ is a decidable variety of groups.
- Computability of $\overline{\mathbf{H}}$-pointlikes implies decidability of membership for more involved varieties built from $\overline{\mathbf{H}}$; one may studies these algorithms for specific choices of $\mathbf{H}$.
- If the variety $\mathbf{H}$ is defined by profinite identities (e.g., trivial, abelian, solvable), these can be used to obtain a faster algorithm for computing $\overline{\mathbf{H}}$-pointlikes than the 'generic' saturation algorithm via $\mathbf{H}$-kernels.
- A detailed study of the complexity and possible implementations for concrete $\mathbf{H}$ are future work.

Pointlike sets for varieties determined by groups

Sam van Gool

DIAMANT Symposium

Veenendaal, November 2018

Supported by EU Marie Curie grant no. 655941

Regular expressions for solvable semigroups

Theorem

A language is recognizable by a solvable semigroup iff it can be described by an AC-regular expression, i.e., an expression built up from Σ^*, Boolean operations, and, for any AC-regular expressions R, S, $a \in \Sigma$, prime p and $0 \le q < p$, the expressions RaS, and $(RaS)^{q \bmod p}$, which describes the language:

$$\{w \in \Sigma^* : \#\{(u,v) \in R \times S : w = uav\} = q \bmod p\}.$$

Example

Let $\Sigma = \{0, 1\}$ and $R = \Sigma^* \setminus (\Sigma^* 1 \Sigma^*)$, so that $L(R) = 0^*$.
The AC-expression $(R1R)^{q \bmod p}$ describes the language of words containing $q \mod p$ occurrences of 1.