Profinite λ-terms and parametricity

Vincent Moreau, joint work with Sam van Gool and Paul-André Melliès

Structure meets Power 2023, Boston

June the 25$^{\text{th}}$, 2023

IRIF, Université Paris Cité, Inria Paris

Context of the talk

Regular languages have a central place in theoretical computer science. Profinite methods are well established for words using finite monoids.

Salvati proposed a notion of regular language of λ-terms using semantic tools.

Contribution: definition of profinite λ-terms using the CCC **FinSet** such that

profinite words are in bijection with profinite λ-terms of Church type

and living in harmony with Stone duality and the principles of Reynolds parametricity.

$\rightarrow$ This is the language side of the story presented at SmP 2022!

Regular languages

Regular languages of words

Let Σ be a finite alphabet, M be a finite monoid and $p : \Sigma \to M$ a set-theoretic function. We write $\bar{p}$ for the associated monoid homomorphism $\Sigma^* \to M$.

For each subset $F \subseteq M$, the set

$$L_F \quad := \quad \{w \in \Sigma^* \mid \bar{p}(w) \in F\}$$

is a regular language. These sets assemble into the Boolean algebra

$$\mathrm{Reg}_M\langle\Sigma\rangle \quad := \quad \{L_F : F \subseteq M\} \ .$$

When M ranges over all finite monoids, we get in this way all regular languages:

$$\mathrm{Reg}\langle\Sigma\rangle \quad = \quad \bigcup_M \mathrm{Reg}_M\langle\Sigma\rangle \ .$$

The Church encoding for words

Any natural number n can be encoded in the simply typed λ-calculus as

$$s : o \Rightarrow o,\ z : o \vdash \underbrace{s\,(\ldots(s\,z))}_{n \text{ applications}} : o\ .$$

A natural number is just a word over a one-letter alphabet.

For example, the word *abba* over the two-letter alphabet $\{a, b\}$

$$a : o \Rightarrow o,\ b : o \Rightarrow o,\ c : o \vdash a\,(b\,(b\,(a\,c))) : o\ .$$

is encoded as the closed λ-term

$$\lambda a.\lambda b.\lambda c.a\,(b\,(b\,(a\,c)))\ :\ \underbrace{(o \Rightarrow o)}_{\text{letter } a} \Rightarrow \underbrace{(o \Rightarrow o)}_{\text{letter } b} \Rightarrow \underbrace{o}_{\text{input}} \Rightarrow \underbrace{o}_{\text{output}}\ .$$

For any alphabet Σ, we define Church_Σ as $\underbrace{(o \Rightarrow o) \Rightarrow \ldots \Rightarrow (o \Rightarrow o)}_{|\Sigma| \text{ times}} \Rightarrow o \Rightarrow o$.

Categorical interpretation

Let **C** be a cartesian closed category and Q be one of its objects.

For any simple type A built from o, we define the object $[\![A]\!]_Q$ by induction as

$$[\![o]\!]_Q \;:=\; Q \qquad \text{and} \qquad [\![A \Rightarrow B]\!]_Q \;:=\; [\![A]\!]_Q \Rightarrow [\![B]\!]_Q \;.$$

Using the cartesian closed structure, one defines an interpretation function

$$[\![-]\!]_Q \quad : \quad \Lambda_{\beta\eta}\langle A\rangle \;\longrightarrow\; \mathbf{C}(1, [\![A]\!]_Q) \;.$$

In **FinSet** which is cartesian closed, given a finite set Q used to interpret o, every word w over the alphabet $\Sigma = \{a, b\}$, seen as a λ-term, is interpreted as a point

$$[\![w]\!]_Q \quad \in \quad (Q \Rightarrow Q) \Rightarrow (Q \Rightarrow Q) \Rightarrow Q \Rightarrow Q$$

which describes how the word will interact with a deterministic automaton.

Regular languages of λ-terms

The notion of regular language of λ-terms has been introduced by Salvati.

For any finite set Q and any subset $F \subseteq [\![A]\!]_Q$, we define the language

$$L_F \quad := \quad \left\{ M \in \Lambda_{\beta\eta}\langle A \rangle \mid [\![M]\!]_Q \in F \right\} \ .$$

All the languages recognized by Q assemble into a Boolean algebra

$$\mathrm{Reg}_Q\langle A \rangle \quad := \quad \left\{ L_F \mid F \subseteq [\![A]\!]_Q \right\} \ .$$

We can then make Q range over all finite sets, and we get the definition

$$\mathrm{Reg}\langle A \rangle \quad := \quad \bigcup_Q \mathrm{Reg}_Q\langle A \rangle \ .$$

Notice that $\mathrm{Reg}\langle A \rangle$ has no reason to be a Boolean algebra for the moment.

A first observation using logical relations

If Q and Q' are two finite sets and $R \subseteq Q \times Q'$, for any simple type A we have

$$[\![A]\!]_R \quad \subseteq \quad [\![A]\!]_Q \times [\![A]\!]_{Q'}$$

In particular, if $f : Q \twoheadrightarrow Q'$ is a partial surjection, then so is $[\![A]\!]_f : [\![A]\!]_Q \twoheadrightarrow [\![A]\!]_{Q'}$.

Using the fundamental lemma of logical relations, one can deduce that

$$\text{if} \quad |Q| \;\geq\; |Q'| \;, \qquad \text{then} \quad \mathrm{Reg}_{Q'}\langle A\rangle \;\subseteq\; \mathrm{Reg}_Q\langle A\rangle \;.$$

This shows that the diagram

$$\Big(\mathrm{Reg}_{Q'}\langle A\rangle \hookrightarrow \mathrm{Reg}_Q\langle A\rangle \Big)_{f : Q \twoheadrightarrow Q'}$$

is directed so we have

$$\mathrm{Reg}\langle A\rangle \quad = \quad \mathrm{colim}_Q\, \mathrm{Reg}_Q\langle A\rangle \;.$$

Profiniteness

The monoid of profinite words

A **profinite word** u is a family (u_p) of elements

$$u_p \in M \qquad \text{where} \qquad \begin{array}{c} M \text{ ranges over all finite monoids} \\ p : \Sigma \to M \text{ ranges over all functions} \end{array}$$

such that for every function $p : \Sigma \to M$ and homomorphism $\varphi : M \to N$, with M and N finite monoids, we have $u_{\varphi \circ p} = \varphi(u_p)$.

The monoid $\widehat{\Sigma^*}$ of profinite words contains Σ^* as a submonoid, since any word $w = w_1 \ldots w_n$, where each $w_i \in \Sigma$, induces a profinite word with components

$$p(w_1) \ldots p(w_n) \qquad \text{for all } p : \Sigma \to M.$$

A profinite word which is not a word

For any finite monoid M there exists $n(M) \geq 1$ such that for all elements m of M, the element $m^{n(M)}$ is the idempotent power of m, which is unique.

Let a be any letter in Σ. The family of elements

$$u_p \quad := \quad p(a)^{n(M)} \qquad \text{for all } p : \Sigma \to M$$

is an idempotent profinite word written a^ω which is not a finite word.

There is a more general construction: if u is a profinite word, then one can build another profinite word u^ω which is idempotent.

Duality: words

Stone spaces, i.e. compact and totally separated spaces, and continuous maps form a category **Stone**. Boolean algebras and their homomorphisms form a category **BA**.

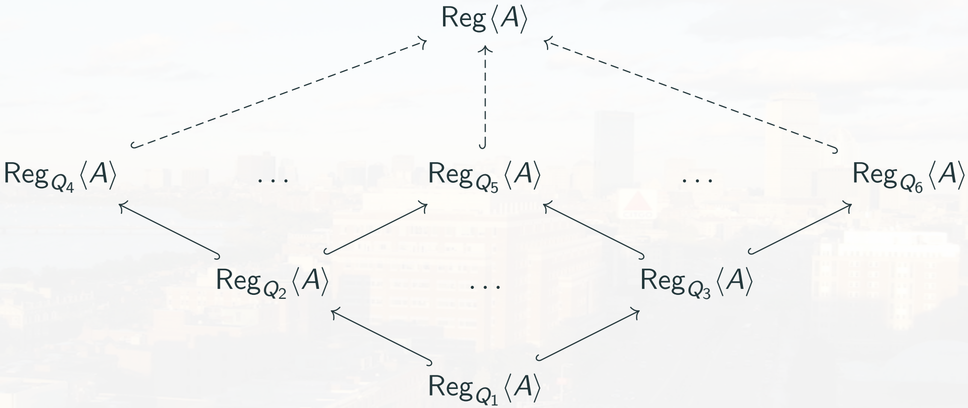
There is an equivalence of categories

$$\mathbf{Stone} \quad \cong \quad \mathbf{BA}^{\mathrm{op}} \ .$$

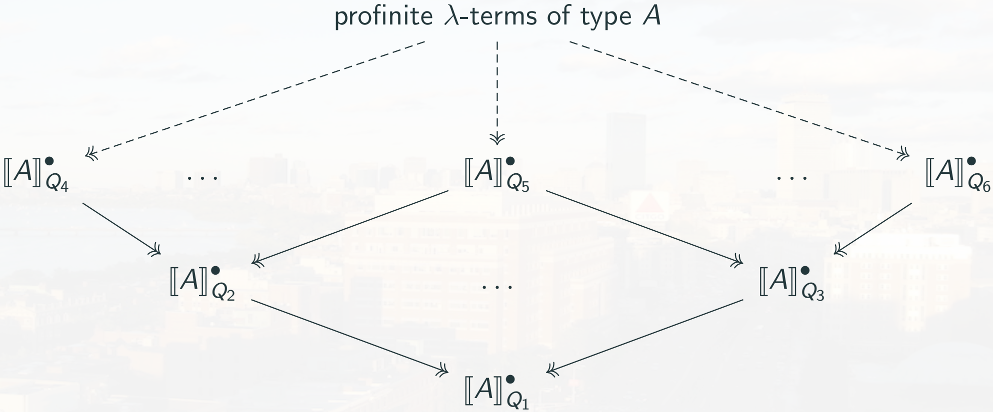
In particular, the monoid of profinite words $\widehat{\Sigma^*}$ has a natural topology such that

$$\widehat{\Sigma^*} \qquad \text{is the Stone dual of} \qquad \mathrm{Reg}\langle\Sigma\rangle \ .$$

Dualizing the diagram



Dualizing the diagram



Definition of profinite λ-terms

The Stone dual of the finite Boolean algebra $\mathrm{Reg}_Q\langle A\rangle$ is the set

$$[\![A]\!]^\bullet_Q \quad := \quad \{[\![M]\!]_Q \mid M \in \Lambda_{\beta\eta}\langle A\rangle\} \;\subseteq\; [\![A]\!]_Q \;.$$

We define $\widehat{\Lambda}_{\beta\eta}\langle A\rangle$ as the limit of the the codirected diagram

$$\left([\![A]\!]^\bullet_f \;:\; [\![A]\!]^\bullet_Q \longrightarrow\!\!\!\!\rightarrow [\![A]\!]^\bullet_{Q'}\right)_{f:Q\twoheadrightarrow Q'}$$

and thus, as expected,

$$\widehat{\Lambda}_{\beta\eta}\langle A\rangle \qquad \text{is the Stone dual of} \qquad \mathrm{Reg}\langle A\rangle \;.$$

Concretely: a **profinite λ-term** θ of type A is a family of elements $\theta_Q \in [\![A]\!]^\bullet_Q$ s.t.

$$[\![A]\!]^\bullet_f(\theta_Q) \;=\; \theta_{Q'} \qquad \text{for every partial surjection } f : Q \twoheadrightarrow Q'.$$

The CCC of profinite λ-terms

Theorem. The profinite λ-terms assemble into a CCC **ProLam** such that

$$\textbf{ProLam}(A, B) \quad := \quad \widehat{\Lambda}_{\beta\eta}\langle A \Rightarrow B\rangle \ .$$

This means that we a compositional notion of profinite λ-calculus.

The interpretation of the simply typed λ-calculus into **ProLam** yields a functor

$$\textbf{Lam} \ \longrightarrow \ \textbf{ProLam}$$

which sends a simply typed λ-term M of type A on the profinite λ-term

$$[\![M]\!]_Q \qquad \text{where } Q \text{ ranges over all finite sets.}$$

This assignment is injective thanks to Statman's finite completeness theorem.

Profinite λ-terms of Church type are profinite words

The Church encoding gives a bijection

$$\Lambda_{\beta\eta}\langle \mathsf{Church}_\Sigma \rangle \quad \cong \quad \Sigma^* \,.$$

This extends to the profinite setting. Indeed, profinite λ-terms of simple type Church_Σ are exactly profinite words as we have a homeomorphism

$$\widehat{\Lambda}_{\beta\eta}\langle \mathsf{Church}_\Sigma \rangle \quad \cong \quad \widehat{\Sigma^*} \,.$$

The profinite λ-term Ω

We consider the profinite λ-term Ω of type $(\obar \Rightarrow \obar) \Rightarrow \obar \Rightarrow \obar$ such that

$$\Omega_Q \quad : \quad f \longmapsto \underbrace{f \circ \cdots \circ f}_{n \text{ times}}$$

where f^n is the idempotent power of the element f of the finite monoid $Q \Rightarrow Q$.

Using Ω, for any Σ of cardinal n, one gets the profinite λ-term

$$\lambda u \lambda a_1 \ldots \lambda a_n . \Omega \, (u \, a_1 \; \ldots \; a_n) \quad : \quad \mathrm{Church}_\Sigma \Rightarrow \mathrm{Church}_\Sigma$$

which is the representation in the profinite λ-calculus of the operator

$$(-)^\omega \quad : \quad \widehat{\Sigma^*} \longrightarrow \widehat{\Sigma^*}$$

on profinite words.

Reynolds parametricity

Parametric families

Let A be a simple type. A **parametric family** θ is a family of elements $\theta_Q \in [\![A]\!]_Q$ s.t.

$$(\theta_Q, \theta_{Q'}) \in [\![A]\!]_R \qquad \text{for all relations } R \subseteq Q \times Q'.$$

Two differences with profinite λ-terms:

- the element θ_Q is not asked to be definable...
- ...but the family is parametric with respect to all relations.

A theorem and its partial converse

We first have a general theorem at every type.

Theorem. Every profinite λ-term is a parametric family.

This theorem admits the following converse at Church types.

Theorem. Every parametric family of type Church_Σ is a profinite λ-term.

The proof of the converse uses the unfolding terms, which generalize the constructors

$$\lambda s \lambda z. z \ : \ \mathsf{Nat} \quad \text{and} \quad \lambda n \lambda s \lambda z. s\,(n\,s\,z) \ : \ \mathsf{Nat} \Rightarrow \mathsf{Nat}$$

of the simple type $\mathsf{Nat} := \mathsf{Church}_1$ to any Church type.

Conclusion

Future work:

- study the situation in other locally finite CCCs (with Tito Nguyen);
- generalize the notion of unfolding term to any simple type;
- investigate the link with the notion of language over monads à la Bojańczyk.

Conclusion

Future work:

- study the situation in other locally finite CCCs (with Tito Nguyen);
- generalize the notion of unfolding term to any simple type;
- investigate the link with the notion of language over monads à la Bojańczyk.

Thank you for your attention!

Any questions?

Bibliography

[Geh16] Mai Gehrke. **"Stone duality, topological algebra, and recognition".** In: *Journal of Pure and Applied Algebra* 220.7 (2016), pp. 2711–2747. ISSN: 0022-4049. DOI: https://doi.org/10.1016/j.jpaa.2015.12.007.

[Mel17] Paul-André Melliès. **"Higher-order parity automata".** In: *Proceedings of the 32nd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2017, Reykjavik, Iceland, 2017*. 2017, pp. 1–12.

[Pin] Jean-Eric Pin. **"Profinite Methods in Automata Theory".** In: *26th International Symposium on Theoretical Aspects of Computer Science STACS 2009*. IBFI Schloss Dagstuhl. URL: https://hal.inria.fr/inria-00359677.

[Sal09] Sylvain Salvati. **"Recognizability in the Simply Typed Lambda-Calculus".** In: *16th Workshop on Logic, Language, Information and Computation*. Vol. 5514. Lecture Notes in Computer Science. Tokyo Japan: Springer, 2009, pp. 48–60.

Salvati generalizes Kleene

The Church encoding induces an isomorphism of Boolean algebras

$$\mathrm{Reg}\langle \mathrm{Church}_\Sigma \rangle \quad \cong \quad \mathrm{Reg}\langle \Sigma \rangle \ .$$

Indeed, every automaton $(Q, \delta, q_0, \mathrm{Acc})$ induces a subset

$$F \quad := \quad \{ q \in [\![\mathrm{Church}_\Sigma]\!]_Q \mid q(\delta, q_0) \in \mathrm{Acc} \}$$

On the other hand, every $q \in [\![\mathrm{Church}_\Sigma]\!]_Q$ induces a finite family of automata

$$(Q, \delta, q_0, \{q(\delta, q_0)\}) \qquad \text{for all } \delta : \Sigma \times Q \to Q \text{ and } q_0 \in Q$$

which determines the behavior of q, and from which one gets finite monoids.

Profinite natural numbers

What does $\widehat{\{a\}^*} \cong \widehat{\mathbb{N}}$, the monoid of profinite natural numbers, look like?

$$0 \;—\; 1 \;—\; 2 \;—\; 3 \;\cdots\cdots \qquad ?$$

Profinite natural numbers

What does $\widehat{\{a\}^*} \cong \widehat{\mathbb{N}}$, the monoid of profinite natural numbers, look like?

$$0 \;\text{———}\; 1 \;\text{———}\; 2 \;\text{———}\; 3 \;\cdots\cdots\cdots \qquad \ldots\omega\ldots$$

Profinite natural numbers

What does $\widehat{\{a\}^*} \cong \widehat{\mathbb{N}}$, the monoid of profinite natural numbers, look like?

$$0 \;\text{———}\; 1 \;\text{———}\; 2 \;\text{———}\; 3 \;\cdots\cdots\cdots \qquad \prod_{p \text{ prime}} \mathbb{Z}_p$$

where $\mathbb{Z}_p$ is the ring of p-adic numbers, i.e. $\mathbb{Z}_p \cong \lim_n \mathbb{Z}/p^n\mathbb{Z}$.