## From profinite words to profinite $\lambda$ -terms

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

LHC days, ENS Paris-Saclay October 12, 2022

IRIF, Université Paris Cité, Inria Paris

#### Context of the talk

Two different kinds of automata:

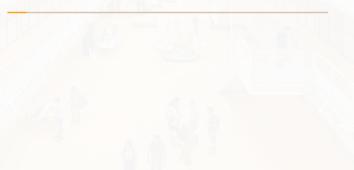
- Deterministic automata (in FinSet)
- Non-deterministic automata (in FinRel)

Profinite methods are well established for words using finite monoids. Contribution: definition of profinite  $\lambda$ -terms in any model and proof that

#### Profinite words are in bijection with deterministic profinite $\lambda$ -terms

using the Church encoding of words and Reynolds parametricity. This leads to a notion of non-deterministic profinite  $\lambda$ -term in **FinRel**.

## Interpreting words as $\lambda$ -terms



#### Simply typed $\lambda$ -terms

 $\lambda$ -terms are defined by the grammar

 $M, N ::= x \mid \lambda x.M \mid MN.$ 

Simple types are generated by the grammar

$$A,B ::= \bullet \mid A \Rightarrow B.$$

For simple types, typing derivations are generated by the following three rules:

$$\frac{\Gamma}{\Gamma, x : A \vdash x : A} \quad \text{Var} \quad \frac{\Gamma \vdash M : A \Rightarrow B}{\Gamma \vdash MN : B} \quad \text{App} \quad \frac{\Gamma, x : A \vdash M : B}{\Gamma \vdash \lambda x . M : A \Rightarrow B} \text{ Abs}$$

#### The Church encoding for words

Any natural number n can be encoded in the simply typed  $\lambda$ -calculus as

$$S: \mathfrak{o} \Rightarrow \mathfrak{o}, Z: \mathfrak{o} \vdash \underbrace{S(\dots(SZ))}_{n \text{ applications}}: \mathfrak{o}$$

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

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

$$a: \mathfrak{o} \Rightarrow \mathfrak{o}, b: \mathfrak{o} \Rightarrow \mathfrak{o}, c: \mathfrak{o} \vdash a(b(b(ac))): \mathfrak{o}$$

is encoded as the closed  $\lambda$ -term

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

## Categorical interpretation

Let **C** be a cartesian closed category.

In order to interpret the simply typed  $\lambda$ -calculus in **C**, we pick an object Q of **C** in order to interpret the base type  $\phi$  and define, for any simple type A, the object

[A]

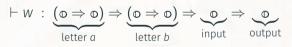
by induction, as follows:

 $\llbracket \mathbf{\Phi} \rrbracket_Q \quad := \quad Q \\ \llbracket A \Rightarrow B \rrbracket_Q \quad := \quad \llbracket A \rrbracket_Q \Rightarrow \llbracket B \rrbracket_Q.$ 

The simply typed  $\lambda$ -terms are then interpreted by structural induction on their type derivation using the cartesian closed structure of **C**.

### The category FinSet

Fact. The category FinSet is cartesian closed: there is a bijection  $FinSet(A \times B, C) \cong FinSet(B, A \Rightarrow C)$ natural in A and C, where  $A \Rightarrow C$  is the set of functions from A to C. In particular, given a finite set Q used to interpret  $\phi$ , every word w over the alphabet  $\Sigma = \{a, b\}$  seen as a  $\lambda$ -term

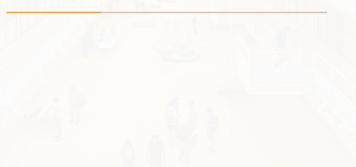


can be interpreted in FinSet as

$$\llbracket w \rrbracket_Q \quad \in \quad (Q \Rightarrow Q) \Rightarrow (Q \Rightarrow Q) \Rightarrow Q \Rightarrow Q$$

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

# Entering the profinite world



#### **Profinite words**

#### Definition. A profinite word is a family of maps

 $u_M$  :  $[\Sigma, M] \longrightarrow M$  where M ranges over all finite monoids such that for every function  $p : \Sigma \rightarrow M$  and homomorphism  $\varphi : M \rightarrow N$ , with Mand N finite monoids, we have  $u_N(\varphi \circ p) = \varphi(u_M(p))$ , i.e. the following diagram commutes:

$$\begin{array}{ccc} [\Sigma, M] & \stackrel{\varphi \circ -}{\longrightarrow} & [\Sigma, N] \\ & & u_M \downarrow & & \downarrow u_N \\ & M & \stackrel{\varphi \circ -}{\longrightarrow} & N \end{array}$$

**Remark.** Any word  $w = a_1 \dots a_n$  induces a profinite word *u* whose components are

 $u_M$  :  $p \mapsto p(a_1) \dots p(a_n)$  where M ranges over all finite monoids.

#### A profinite word which is not a word

In any finite monoid M, all elements  $m \in M$  have a unique power  $m^n$  (for  $n \ge 1$ ) which is idempotent, i.e. such that  $m^n m^n = m^n$ . It is obtained for n = |M|!.

Let a be any letter in  $\Sigma$ . The family of maps

 $u_M$ :  $[\Sigma, M] \longrightarrow M$  $p \longmapsto p(a)^{|M|!}$  where M ranges over all finite monoids

is an profinite word written  $a^{\omega}$  which is not a finite word.

The set of profinite words is endowed with a monoid structure computed pointwise. In that setting,  $a^{\omega}$  is idempotent.

### Key property: parametricity of profinite words

**Definition.** Given *M*, *N* two finite monoids and  $R \subseteq M \times N$ , we say that *R* is a **monoidal relation**  $M \rightarrow N$  if it is a submonoid of  $M \times N$ . This means that

 $(e_M, e_N) \in R$  and for all (m, n) and (m', n') in R, we have  $(mm', nn') \in R$ .

**Proposition.** Let  $u = (u_M)$  be a family of maps. The following are equivalent:

- *u* is profinite
- for every pair of functions  $p: \Sigma \to M$  and  $q: \Sigma \to N$  with M and N finite monoids, and for any monoidal relation  $R: M \to N$ ,

if for all  $a \in \Sigma$  we have  $(p(a), q(a)) \in R$ , then  $(u_M(p), u_N(q)) \in R$ .

## Parametric $\lambda$ -terms

### Definition of logical relations

Recall that for any set Q we have defined the set

by structural induction on the type A.

We extend the construction to set-theoretic relations  $R: P \rightarrow Q$ , giving a relation

[A]\_

$$\llbracket A \rrbracket_R : \llbracket A \rrbracket_P \to \llbracket A \rrbracket_Q .$$

by structural induction on the type A:

$$\begin{split} \llbracket \Phi \rrbracket_R &:= R \\ \llbracket A \Rightarrow B \rrbracket_R &:= \{ (f,g) \in \llbracket A \Rightarrow B \rrbracket_P \times \llbracket A \Rightarrow B \rrbracket_Q \mid \\ & \text{for all } x \in \llbracket A \rrbracket_P \text{ and } y \in \llbracket A \rrbracket_Q , \\ & \text{if } (x,y) \in \llbracket A \rrbracket_R \text{ then } (f(x),g(y)) \in \llbracket B \rrbracket_R \}. \end{split}$$

#### Double categories and main example

A double category is given by the data of objects together with

- $\cdot$  1-cells: vertical (  $\rightarrow$  ) and horizontal (  $\rightarrow$  ) arrows,
- 2-cells: squares ( $\Rightarrow$ ) between pairs of vertical and horizontal arrows which can be composed both horizontally or vertically.

**Example.** the category whose objects are finite sets, vertical arrows are functions, horizontal arrows are relations and whose squares are unique and exist when:

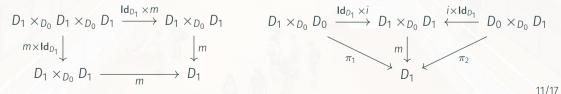
$$\begin{array}{cccc} X & \stackrel{R}{\longrightarrow} & Y \\ f \downarrow & \downarrow g \\ X' & \stackrel{R}{\longrightarrow} & Y' \end{array} & \text{iff} & \forall x \in X, y \in Y, \quad \text{if } (x, y) \in R & \text{then } (f(x), g(y)) \in R' \\ \end{array}$$

#### Double categories as internal categories

The category **Cat** of categories has pullbacks. **Definition.** A double category is a diagram

 $D_1$   $s \left( \begin{array}{c} \uparrow i \\ D_0 \end{array} \right) t$ 

where  $s \circ i = Id_{D_0} = t \circ i$ , together with  $m : D_1 \times_{D_0} D_1 \to D_1$  such that  $s \circ m = s \circ \pi_1$ and  $t \circ m = t \circ \pi_2$  such that the following monoidal identities hold:



#### FinSet as an internal category

**Example.** We can endow **FinSet** with a structure of double category:

- the category  $D_0$  is **FinSet**
- the category  $D_1$  is the category whose objects are relations  $R : X \to Y$  and a morphism  $f : (R : X \to Y) \to (R' : X' \to Y')$  is a pair of functions  $f_1 : X \to X'$  and  $f_2 : Y \to Y'$  such that

if  $(x, y) \in R$  then  $(f_1(x), f_2(y)) \in R'$ .

We take  $s(R : X \rightarrow Y) = X$  and  $t(R : X \rightarrow Y) = Y$ . If  $R : X \rightarrow Y$  and  $R' : Y \rightarrow Z$ , we let

 $m(R, R') = R \circ R' = \{(x, z) \in X \times Z \mid \exists y \in Y, (x, y) \in R, (y, z) \in R'\}.$ 

#### Cartesian double categories

A double category **D** is cartesian if the pairs of squares

is in bijection with the set of squares

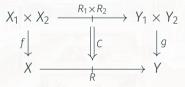
$$\begin{array}{ccc} X & \xrightarrow{R} & Y \\ \langle f_1, f_2 \rangle & & & \downarrow \langle C_1, C_2 \rangle & & \downarrow \langle g_1, g_2 \rangle \\ X_1 \times X_2 & \xrightarrow{S_1 \times S_2} & Y_1 \times Y_2 \end{array}$$

and the horizontal morphism  $Id_1 : 1 \rightarrow 1$  is terminal.

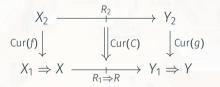
Internally:  $D_0$  and  $D_1$  are cartesian and s and t strictly respect the cartesian structure.

#### Cartesian closed double categories

A cartesian double category **D** is closed if the set of squares



is in bijection with the set of squares



Internally:  $D_0$  and  $D_1$  are CCCs and s and t strictly respect the CCC structure. Fact. The double category of finite sets is cartesian closed.

14/17

#### Parametric $\lambda$ -terms

Let us consider a cartesian closed double category.

**Definition.** Let A be a simple type. A **parametric**  $\lambda$ **-term** of type A is the data

- a family of vertical maps  $\theta_Q : 1 \rightarrow \llbracket A \rrbracket_Q$  where Q ranges over all objects
- a family of squares  $\theta_R : \operatorname{Id}_1 \Rightarrow \llbracket A \rrbracket_R$  where R ranges over all horizontal arrows

such that the horizontal source and target of a square  $\theta_R$  for  $R : P \rightarrow Q$  are the maps  $\theta_P$  and  $\theta_Q$ , which we can represent as

#### Parametric $\lambda$ -terms and profinite words

In the case of **FinSet**, a parametric  $\lambda$ -term of type A amounts to a family  $\theta_Q \in \llbracket A \rrbracket_Q$  where Q ranges over all finite sets, such that, for every binary relation  $R: P \rightarrow Q$ , we have

 $(\theta_P, \theta_Q) \in \llbracket A \rrbracket_R.$ 

**Theorem.** Parametric  $\lambda$ -terms define a cartesian closed category, and the parametric  $\lambda$ -terms of type

$$\begin{array}{lll} \mathsf{Church}_{\Sigma} & := & \underbrace{( \mathbb{O} \Rightarrow \mathbb{O} ) \Rightarrow \ldots \Rightarrow ( \mathbb{O} \Rightarrow \mathbb{O} )}_{|\Sigma| \text{ times}} \Rightarrow ( \mathbb{O} \Rightarrow \mathbb{O} ) \end{array}$$

are in bijection with the profinite words on  $\Sigma$ .

### Conclusion

Current & future work:

- investigate the metric side of the profinite words in the setting of  $\lambda$ -terms;
- determine the parametric  $\lambda$ -terms of type Church<sub> $\Sigma$ </sub> in the model associated to nondeterministic automata;
- investigate a generalization of logic on words with **MSO** to a logic on  $\lambda$ -terms.

### Conclusion

Current & future work:

- investigate the metric side of the profinite words in the setting of  $\lambda$ -terms;
- determine the parametric  $\lambda$ -terms of type Church<sub> $\Sigma$ </sub> in the model associated to nondeterministic automata;
- investigate a generalization of logic on words with **MSO** to a logic on  $\lambda$ -terms.

Thank you for your attention!

Any questions?

### Bibliography

- [Geh13] Mai Gehrke. Stone duality, topological algebra, and recognition. 2013. DOI: 10.48550/ARXIV.1309.2422. URL: https://arxiv.org/abs/1309.2422.
- [GS16] Samuel J. v. Gool and Benjamin Steinberg. Pro-aperiodic monoids via saturated models. 2016. DOI: 10.48550/ARXIV.1609.07736. URL: https://arxiv.org/abs/1609.07736.
- [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.

#### The inverse bijections T and W

**Pro** → **Para.** Every profinite word *u* induces a parafinite term with components  $T(u)_Q : \begin{array}{c} \Sigma \Rightarrow (Q \Rightarrow Q) & \longrightarrow & Q \Rightarrow Q \\ p & \longmapsto & u_{Q \Rightarrow Q}(p) \end{array}$ given the fact that *Q* ⇒ *Q* is a monoid for the function composition.

**Para**  $\rightarrow$  **Pro.** Every parametric term  $\theta$  induces a profinite word with components

$$W(\theta)_{M} : \begin{array}{ccc} \Sigma \Rightarrow M & \longrightarrow & M \\ p & \longmapsto & \theta_{M}(i_{M} \circ p)(e_{M}) \end{array} \begin{array}{c} \Sigma \Rightarrow (M \Rightarrow M) & \xrightarrow{\theta_{M}} & M \Rightarrow M \\ i_{M} \circ -\uparrow & & \downarrow -(e_{M}) \\ \Sigma \Rightarrow M & \xrightarrow{W(\theta)_{M}} & M \end{array}$$

where  $i_M : M \to (M \Rightarrow M)$  is the Cayley embedding.

These are bijections between profinite words and parametric  $\lambda$ -terms.

#### $Pro \rightarrow Para \rightarrow Pro$

Let *u* be a profinite word. Recall that  $u_M : (\Sigma \Rightarrow M) \to M$ . Its associated parametric  $\lambda$ -term T(u) has components

 $T(u)_Q = u_{(Q \Rightarrow Q)}$ 

Its associated profinite word W(T(u)), for  $p: \Sigma \to M$ , is equal to

$$W(T(u))_M(p) = T(u)_M(i_M \circ p)(e_M) = u_{(M \Rightarrow M)}(i_M \circ p)(e_M)$$

In order to show that W(T(u)) is u, we use the parametricity of profinite words. We consider the moinoidal logical relation  $R \subseteq (M \Rightarrow M) \times M$  defined as

$$R := \{(f,m) \in (M \Rightarrow M) \times M \mid \forall n \in M, f(n) = m \cdot n\}$$

#### $\textbf{Pro} \rightarrow \textbf{Para} \rightarrow \textbf{Pro}$

(2/2)

We have that  $(i_M \circ p, p) \in \llbracket \mathfrak{o} \times \cdots \times \mathfrak{o} \rrbracket_R$  because for all  $a \in \Sigma$ , for all  $m \in I$ ,  $(i_M \circ p)(a)(m) = p(a) \cdot m$ .

By parametricity of u applied to R, we have that

 $(u_{(M \Rightarrow M)}(i_M \circ p), u_M(p)) \in \llbracket \mathfrak{o} \Rightarrow \mathfrak{o} \rrbracket_R$ which means, by definition of  $\llbracket \mathfrak{o} \Rightarrow \mathfrak{o} \rrbracket_R$ , that for all  $(f, m) \in R$ , we have  $(u_{(M \Rightarrow M)}(i_M \circ p)(f), u_M(p)(m)) \in R$ 

which gives the desired result:

 $W(T(u)) = u_{(M \Rightarrow M)}(i_M \circ p)(e_M) = u_M(p)(m).$ 

#### $Para \rightarrow Pro \rightarrow Para$

Let  $\theta$  be a parafinite term. Recall that  $\theta_Q \in (\Sigma \Rightarrow (Q \Rightarrow Q)) \Rightarrow (Q \Rightarrow Q)$ . Its associated profinite word  $W(\theta)$  is equal, for  $p : \Sigma \to M$ , to

 $W(\theta)_M(p) = \theta_M(i_M \circ p)(e_M).$ 

Its reassociated parametric  $\lambda$ -term  $T(W(\theta))$  has components

 $T(W(\theta))_Q = W_{(Q \Rightarrow Q)}.$ 

We want to show that, for all  $p: \Sigma \to (Q \Rightarrow Q)$ , we have  $\theta_Q(p) = T(W(\theta))_Q(p)$ , i.e.

for all  $q_0 \in Q$ ,  $\theta_{(Q \Rightarrow Q)}(i_{(Q \Rightarrow Q)} \circ p)(\mathrm{Id}_Q)(q_0) = \theta_Q(p)(q_0)$ 

To show that, we introduce, for any  $q_0 \in Q$ , the logical relation

$$R_{q_0} := \{(f,q) \in (Q \Rightarrow Q) \times Q \mid f(q_0) = q\}.$$

First, we have  $(i_{(Q\Rightarrow Q)} \circ p, p) \in \llbracket ( \mathfrak{o} \Rightarrow \mathfrak{o} ) \times \cdots \times ( \mathfrak{o} \Rightarrow \mathfrak{o} ) \rrbracket_{R_{q_0}}$  because for all  $a \in \Sigma$ ,

for all  $(f,q) \in R$ , we have  $(i_{(Q\Rightarrow Q)} \circ p)(a)(f)(q_0) = p(a)(f(q_0)) = p(a)(q)$ 

By parametricity of  $\theta$ , we obtain that  $(\theta_{(Q \Rightarrow Q)}(i_{(Q \Rightarrow Q)} \circ p), \theta_Q(p)) \in [\![ \mathfrak{o} \Rightarrow \mathfrak{o} ]\!]_{R_{q_0}}$ . Given the fact that  $(\mathsf{Id}_Q, q_0) \in R_{q_0}$  and by definition of  $[\![ \mathfrak{o} \Rightarrow \mathfrak{o} ]\!]_{R_{q_0}}$ , we obtain that

 $\theta_{(Q \Rightarrow Q)}(i_{(Q \Rightarrow Q)} \circ p)(\mathsf{Id}_Q)(q_0) = \theta_Q(p)(q_0)$ 

which concludes the proof.  $\Box$