λ-definable functions and automata theory

Nguyễn Lê Thành Dũng (a.k.a. Tito) — nltd@nguyentito.eu
LIPN, Université Paris 13

TACL summer school, Porquerolles, June 12th, 2019

The λ-calculus

A naive syntactic theory of functions:

$$\begin{aligned} f\,x &\quad\approx\quad f(x) \\ \lambda x.\ t &\quad\approx\quad x \mapsto t \\ (\lambda x.\ t)\ u =_\beta t\{x := u\} &\quad\approx\quad (x \mapsto x^2 + 1)(42) = 42^2 + 1 \end{aligned}$$

The λ-calculus

A naive syntactic theory of functions:

$$
\begin{aligned}
f\,x &\quad\approx\quad f(x) \\
\lambda x.\ t &\quad\approx\quad x \mapsto t \\
(\lambda x.\ t)\ u =_\beta t\{x := u\} &\quad\approx\quad (x \mapsto x^2 + 1)(42) = 42^2 + 1
\end{aligned}
$$

Church encodings of natural numbers:
morally, $n \in \mathbb{N} \rightsquigarrow \overline{n} : f \mapsto f^n = f \circ \ldots \circ f$

$$\overline{2} = \lambda f.\ (\lambda x.\ f\ (f\ x))$$

The simply typed λ-calculus

Add a *type system*: specifications for λ-terms

$$t : A \to B \quad \approx \quad \text{“}t \text{ is a function from } A \text{ to } B\text{”}$$

Simple types: built from constant o and binary operation $\to$

$$\dfrac{f : o \to o \qquad \dfrac{f : o \to o \qquad x : o}{f\,x : o}}{f\,(f\,x) : o}$$

The simply typed λ-calculus

Add a *type system*: specifications for λ-terms

$$t : A \to B \quad \approx \quad \text{“}t \text{ is a function from } A \text{ to } B\text{”}$$

Simple types: built from constant o and binary operation $\to$

$$\dfrac{f : o \to o \qquad \dfrac{f : o \to o \qquad x : o}{f\,x : o}}{f\,(f\,x) : o}$$

$$\overline{2} = \lambda f.\ (\lambda x.\ f\,(f\,x)) : (o \to o) \to (o \to o)$$

$\mathsf{Nat} = (o \to o) \to (o \to o)$ is the type of natural numbers

λ-definable functions

$f : \mathbb{N} \to \mathbb{N}$ λ-definable iff

$$\exists A \text{ simple type, } t : \mathsf{Nat}\{o := A\} \to \mathsf{Nat} \mid \forall n \in \mathbb{N},\ t\ \overline{n} =_\beta \overline{f(n)}$$

Question: what are the λ-definable functions $\mathbb{N} \to \mathbb{N}$?

λ-definable functions

$f : \mathbb{N} \to \mathbb{N}$ λ-definable iff

$$\exists A \text{ simple type, } t : \mathsf{Nat}\{o := A\} \to \mathsf{Nat} \mid \forall n \in \mathbb{N},\ t\ \overline{n} =_\beta \overline{f(n)}$$

Question: what are the λ-definable functions $\mathbb{N} \to \mathbb{N}$?

Open question! No satisfactory characterization.

$\mathsf{Nat} \to \mathsf{Nat}$ w/o substitution: extended polynomials (Schwichtenberg 1975)

λ-definable functions

$f : \mathbb{N} \to \mathbb{N}$ λ-definable iff

$$\exists A \text{ simple type}, \ t : \mathsf{Nat}\{o := A\} \to \mathsf{Nat} \mid \forall n \in \mathbb{N}, \ t\ \overline{n} =_\beta \overline{f(n)}$$

Question: what are the λ-definable functions $\mathbb{N} \to \mathbb{N}$?

Open question! No satisfactory characterization.

$\mathsf{Nat} \to \mathsf{Nat}$ w/o substitution: extended polynomials (Schwichtenberg 1975)

Theorem (folklore? but not very well-known)

For $X \subseteq \mathbb{N}$, $X = f^{-1}(0)$ for some λ-definable $f : \mathbb{N} \to \mathbb{N}$
iff X is ultimately periodic.

Proof by semantic evaluation

Canonical semantics:

- choose set S, $[\![o]\!] = S$, $[\![A \to B]\!] = [\![B]\!]^{[\![A]\!]}$
- $t : A \rightsquigarrow [\![t]\!] \in [\![A]\!]$, e.g. $[\![f\,x]\!] = [\![f]\!]\,([\![x]\!])$
- soundness: $t =_\beta u \implies [\![t]\!] = [\![u]\!]$

Proof by semantic evaluation

Canonical semantics:

- choose set S, $[\![o]\!] = S$, $[\![A \to B]\!] = [\![B]\!]^{[\![A]\!]}$
- $t : A \rightsquigarrow [\![t]\!] \in [\![A]\!]$, e.g. $[\![f\,x]\!] = [\![f]\!]\,([\![x]\!])$
- soundness: $t =_\beta u \implies [\![t]\!] = [\![u]\!]$

Theorem

For $X \subseteq \mathbb{N}$, $X = f^{-1}(0)$ *for some* λ*-definable* $f : \mathbb{N} \to \mathbb{N}$
iff X *is* ultimately periodic.

Proof sketch of ($\implies$).

- choose S finite: $[\![\mathsf{Nat}]\!]$ is a finite monoid
- $n \mapsto [\![\overline{n}]\!]$ is a monoid morphism from $(\mathbb{N}, +)$ to $[\![\mathsf{Nat}]\!]$
- $[\![\overline{n}]\!]$ determines whether $n \in X$ $\square$

Finally: connections with automata

Generalization to Church-encoded *words* over finite alphabet Σ:

Theorem (Hillebrand & Kanellakis 1995)

For $L \subseteq \Sigma^*$, $L = f^{-1}(\varepsilon)$ *for some* λ*-definable* $f : \Sigma^* \to \Sigma^*$ *iff* X *is a* regular language.

Same proof (characterize reg. lang. by monoids).

λ-definable languages are recognizable by finite automata.

λ-definable *functions* are *regularity-preserving*.

$\longrightarrow$ I'm looking for an automata-theoretic characterization.