Singleton Theorem Using Models

Srivathsan B, Igor Walukiewicz

LaBRI

Paris, March 2010

Introduction

Singleton Theorem [Statman'82]

For every lambda term M, there exists a finite standard model $\mathcal{D}$ and a variable assignment v such that M is uniquely determined in $\mathcal{D}$ and v.

Motivation: Standard models are strong enough to identify single terms (up to β,η-reductions).

Method: Construction of $\mathcal{D}$ for M by induction on the Böhm tree of M.

Simply typed λ terms

Types τ

$$\tau ::= 0 \mid \tau \to \tau$$

Terms

- Variables: $x^\alpha, y^\alpha, \ldots$
- λ-abstraction: $\lambda x^\alpha . M^\beta$
- Application: $MN : \beta$; if $M : \alpha \to \beta$ and $N : \alpha$

Remarks

- We can have more than one basic type.
- Constants can be added without any problems.

Standard Models

Standard Finite Model $\mathcal{D} = (D_\alpha)_{\alpha \in \tau}$

- D_0: a finite set of elements of the basic type.
- $D_{\alpha \to \beta}$: the set of functions from D_α to D_β.

Variable assignment

A variable assignment is a function v associating to a variable of type α an element of D_α.

Notation: $v[d/x^\alpha]$.

Interpretation

Interpretation

Interpretation of a term M of type α in a model $\mathcal{D}$ and variable assignment v
$[\![M]\!]^v_{\mathcal{D}} \in D_\alpha$:

- $[\![x^\alpha]\!]^v_{\mathcal{D}} = v(x^\alpha)$
- $[\![MN]\!]^v_{\mathcal{D}} = [\![M]\!]^v_{\mathcal{D}}[\![N]\!]^v_{\mathcal{D}}$
- $[\![\lambda x^\alpha.M]\!]^v_{\mathcal{D}}$ is a function mapping an element $d \in D_\alpha$ to $[\![M]\!]^{v[d/x^\alpha]}_{\mathcal{D}}$.

- β-reduction $(\lambda x.M)N \to_\beta M[N/x]$.
- η-reduction $\lambda x.Mx \to_\eta M$, provided x is not free in M.

η-long form

Using λ to make the functions explicit:

$$\lambda x^\alpha.z^{\alpha\to\beta}x \quad \text{instead of} \quad z^{\alpha\to\beta}$$

Böhm Trees

Observe that a term in a β-normal, and η-long form is of a shape:

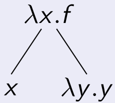
$$\lambda \overrightarrow{x}.zM_1 \ldots M_k,$$

where z is a variable, $zM_1 \ldots M_k : 0$, and the sequence $\lambda \overrightarrow{x}$ may be empty.

Böhm Trees

If $M = \lambda \overrightarrow{x}.zM_1 \ldots M_k$, then the root of $BT(M)$ is labeled $\lambda \overrightarrow{x}.z$ and has $BT(M_1), \ldots, BT(M_k)$ as its children.

Example: $\lambda x.(f\ x\ (\lambda y.y))$



Remark

$BT(M)$ is a particular way of representing terms in a normal form as a tree.

Statement of the Theorem

Uniquely determined

M is said to be *uniquely determined* in a model $\mathcal{D}$ with a variable assignment v if for all lambda terms N, $[\![N]\!]^v_{\mathcal{D}} = [\![M]\!]^v_{\mathcal{D}}$ iff $N =_{\beta\eta} M$.

Singleton Theorem [Statman'82]

For every lambda term M, there exists a standard finite model $\mathcal{D}$ and a variable assignment v such that M is uniquely determined in $\mathcal{D}$ and v.

Basic Idea

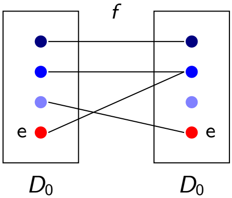
- We consider a lambda term M in η-long normal form.
- We assume that we have a model $\mathcal{D}$ and an interpretation in which all subterms of M are uniquely determined.
- We add "an element" to $\mathcal{D}$, and alter the interpretation to make M uniquely determined too.

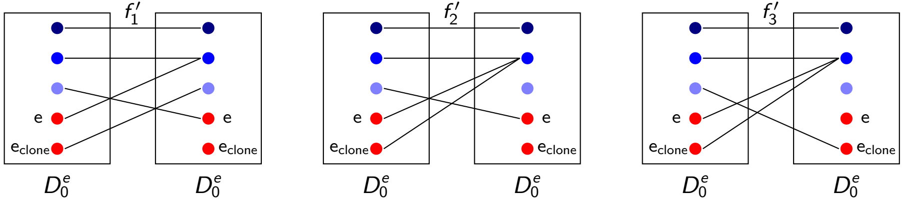
The Extended Model

Model $\mathcal{D}^e$

Given a model $\mathcal{D} = (D_\alpha)_{\alpha\in\tau}$ and an element $e \in D_0$ the extended model $\mathcal{D}^e = (D^e_\alpha)_{\alpha\in\tau}$ is determined by:

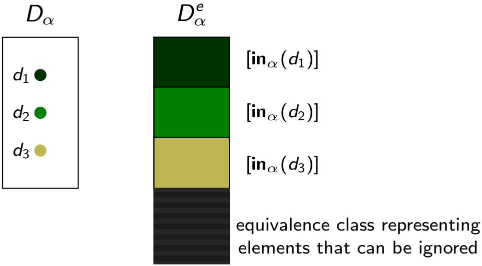
$$D^e_0 = D_0 \uplus \{e_{clone}\}$$





Visualizing a set D^e_α

In general, we would like to visualize each set D^e_α as follows

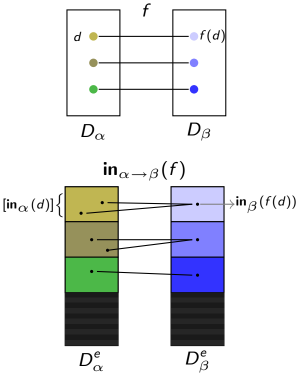


- $\mathbf{in}_\alpha$ represents the injection function, and
- $[d']$ denotes the equivalence class of $d' \in D^e_\alpha$.

A null element h_0 is any arbitrary element of D_0^e different from e_{clone}. For a type $\alpha \rightarrow \beta$, element $h_{\alpha\rightarrow\beta}$ is the constant function mapping every element to h_β.

Definition $\mathbf{in}_0$ and $\leftrightarrow_0$

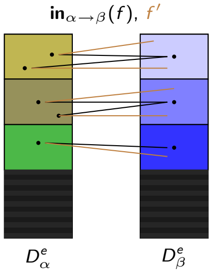
- $\mathbf{in}_0 : D_0 \rightarrow D_0^e$ is the identity.
- $\leftrightarrow_0$ is the smallest equivalence containing $e \leftrightarrow_0 e_{clone}$.



Definition $\mathbf{in}_{\alpha\rightarrow\beta}$

- If $f \in D_{\alpha\rightarrow\beta}$ then $\mathbf{in}_{\alpha\rightarrow\beta}(f)$ is $f' \in D^e_{\alpha\rightarrow\beta}$ such that:

$$f'(d') = \begin{cases} \mathbf{in}_\beta(f(d)) & \text{if } d' \in [\mathbf{in}_\alpha(d)] \\ h_\beta & \text{otherwise} \end{cases}$$



Equivalence relation

- We say that $f' \in D^e_{\alpha\to\beta}$ simulates $f \in D_{\alpha\to\beta}$ ($sim(f', f)$) if for all $d \in D_\alpha$, for all $d' \in [\mathbf{in}_\alpha(d)]$: $f'(d') \leftrightarrow_\beta \mathbf{in}_\beta(f(d))$
- For $f', g' \in D^e_{\alpha\to\beta}$, we have

$$f' \leftrightarrow_{\alpha\to\beta} g' \quad \text{if for all } h \in D_{\alpha\to\beta},\ sim(f', h) \Leftrightarrow sim(g', h).$$

Observation

For every $d_1, d_2 \in D_\alpha$, if $d_1 \neq d_2$, then $\mathbf{in}_\alpha(d_1) \not\leftrightarrow_\alpha \mathbf{in}_\alpha(d_2)$.

Definition

A variable assignment v' on $\mathcal{D}^e$ *simulates* a variable assignment v on $\mathcal{D}$ if for all variables x: $sim(v'(x), v(x))$.

Lemma

If v' *simulates* v then for every lambda term M:

$$sim(\llbracket M \rrbracket^{v'}_{\mathcal{D}^e}, \llbracket M \rrbracket^{v}_{\mathcal{D}})$$

where α is the type of M.

Corollary

Every term uniquely determined in $(\mathcal{D}, v)$ is uniquely determined in $(\mathcal{D}^e, v')$.

Proof of the Singleton Theorem

Consider a lambda term $\lambda\overrightarrow{x}.yM_1 \dots M_k$, with $yM_1 \dots M_k$ of type 0.

Assume

- $M_1, \dots, M_k$ are uniquely determined in a model $\mathcal{D}$ and a variable assigment v,
- $[\![yM_1 \dots M_k]\!]^v_{\mathcal{D}} = e$.

Construct the model $\mathcal{D}^e$ by adding e_{clone}.

Variable assignment v^e

1. $v^e(x) = \mathbf{in}_{\tau(x)}(v(x))$, if $x \neq y$.
2. For the variable y,

$$v^e(y)(d'_1, \dots, d'_k) = \begin{cases} e_{clone} & \text{if } d'_i \in [\mathbf{in}_{\beta_i}([\![M_i]\!]^v_{\mathcal{D}})], \\ & \text{for } i \in \{1, \dots, k\} \\ \mathbf{in}_{\tau(y)}(v(y))(d'_1, \dots, d'_k) & \text{otherwise} \end{cases}$$

As v^e *simulates* v we have:

- For all lambda terms N, $sim([\![N]\!]_{\mathcal{D}^e}^{v^e}, [\![N]\!]_{\mathcal{D}}^{v})$, that is, $[\![N]\!]_{\mathcal{D}^e}^{v^e} \leftrightarrow_{\beta_i} \mathbf{in}_{\beta_i}([\![N]\!]_{\mathcal{D}}^{v})$
- So $M_1, \dots, M_k$ are uniquely determined in $(\mathcal{D}^e, v^e)$
- Moreover, $[\![yM_1 \dots M_k]\!]_{\mathcal{D}^e}^{v^e} = e_{clone}$.

Uniqueness

Let $[\![wN_1 \dots N_p]\!]_{\mathcal{D}^e}^{v^e} = e_{clone}$.

- $w \neq y$ is not possible.
- when $w = y$ we get:

$$
\begin{aligned}
& [\![N_i]\!]_{\mathcal{D}^e}^{v^e} \in [\mathbf{in}_{\beta_i}([\![M_i]\!]_{\mathcal{D}}^{v})] \\
& \Rightarrow \mathbf{in}_{\beta_i}([\![N_i]\!]_{\mathcal{D}}^{v}) \leftrightarrow_{\beta_i} \mathbf{in}_{\beta_i}([\![M_i]\!]_{\mathcal{D}}^{v}) \\
& \Rightarrow [\![N_i]\!]_{\mathcal{D}}^{v} = [\![M_i]\!]_{\mathcal{D}}^{v} \\
& \Rightarrow N_i = M_i
\end{aligned}
$$

$yM_1 \dots M_k$ uniquely determined implies $\lambda\overrightarrow{x}.yM_1 \dots M_k$ is uniquely determined.

Base Case

Leaf is a variable z of type 0.

- Start: trivial model with only one element $\{\perp\}$ in its atomic set, trivial variable assignment.
- Add an extra element $\{\perp_{clone}\}$ to type 0.
- New variable assignment assigns z to $\perp_{clone}$ and the rest is kept same.

Conclusions

- In our approach we
  - define an operation of model extension, and
  - explain the relation between elements of the initial and extended model.
- We work mostly with semantics, the only syntactic tool is η-long forms (and Böhm trees).

Related Work:

- [Statman'82] Finite Completeness Theorem
- [Statman & Dowek'92]
- [Salvati'07] Using intersection types