ORACLES AND CHOICE SEQUENCES FOR TYPE-THEORETIC PRAGMATICS

Jon Sterling

October 8, 2015

joint work with Darryl McAdams

INTRODUCTION

$[\![$*A man walked in.*$]\!]$

$\triangledown$

$$[\![\textit{A man walked in.}]\!]$$

$$\triangledown$$

$$(\Sigma p \in Man)$$

$$[\![\textit{A man walked in.}]\!]$$

$$\triangledown$$

$$(\Sigma p \in Man)\, WalkedIn(p)$$

$$[\![\textit{He sat down}]\!]$$
$$\triangledown$$

$$[\![\textit{He sat down}]\!]$$

$$\triangledown$$

$$\mathit{SatDown}(???)$$

$$[\![\textit{A man walked in. He sat down}]\!]$$
$$\bigtriangledown$$

$$[\![\textit{A man walked in. He sat down}]\!]$$

$$\triangledown$$

$$(\Sigma x \in (\Sigma p \in Man)\, WalkedIn(p))$$

$$[\![\textit{A man walked in. He sat down}]\!]$$
$$\triangledown$$
$$(\Sigma x \in (\Sigma p \in \mathit{Man})\, \mathit{WalkedIn}(p))\, \mathit{SatDown}(???)$$

$$[\![\textit{A man walked in. He sat down}]\!]$$
$$\triangledown$$
$$(\Sigma x \in (\Sigma p \in Man)\, WalkedIn(p))\, SatDown(\pi_1(x))$$

$$[\![\textit{Every farmer who owns a donkey beats it.}]\!]$$

$$\triangledown$$

$$(\Pi p \in (\Sigma x \in \mathit{Farmer})\,(\Sigma y \in \mathit{Donkey})\,\mathit{Owns}(x; y))$$

$$\llbracket \textit{Every farmer who owns a donkey beats it.} \rrbracket$$

$$\triangledown$$

$$(\Pi p \in (\Sigma x \in \mathit{Farmer})\,(\Sigma y \in \mathit{Donkey})\,\mathit{Owns}(x; y))\,\mathit{Beats}(???; ???)$$

$$[\![\textit{Every farmer who owns a donkey beats it.}]\!]$$
$$\triangledown$$
$$(\Pi p \in (\Sigma x \in \mathit{Farmer})\,(\Sigma y \in \mathit{Donkey})\,\mathit{Owns}(x; y))\,\mathit{Beats}(\pi_1(p); \pi_2(p))$$

TWO THINGS TO DEAL WITH

TWO THINGS TO DEAL WITH

- terms for presuppositions

TWO THINGS TO DEAL WITH

- terms for presuppositions
- resolution of presuppositions

- terms for presuppositions
- resolution of presuppositions

TWO THINGS TO DEAL WITH

- terms for presuppositions (this talk)
- resolution of presuppositions

THE **require** ORACLE: STATICS

require — FORMAL RULES

require — FORMAL RULES

$$\textbf{require} : (0;1) \qquad \text{(operator)}$$

$$\textbf{require}\ x : A\ \textbf{in}\ N \stackrel{\text{def}}{=} \textbf{require}(A; x.N) \qquad \text{(notation)}$$

require — FORMAL RULES

$$\textbf{require} : (0;1) \qquad \text{(operator)}$$

$$\textbf{require}\ x : A\ \textbf{in}\ N \overset{\text{def}}{=} \textbf{require}(A; x.N) \qquad \text{(notation)}$$

$$\frac{\Gamma \vdash A\ \mathit{type} \quad \Gamma, x : A \vdash N \in B}{\Gamma \vdash \textbf{require}\ x : A\ \textbf{in}\ N \in B} \qquad (\mathit{require})$$

$$[\![\textit{A man walked in. He sat down}]\!]$$

$$\triangledown$$

$$(\Sigma x \in (\Sigma p \in Man)\, WalkedIn(p))\, SatDown(???)$$

$$[\![\textit{A man walked in. He sat down}]\!]$$

$$\triangledown$$

$$(\Sigma x \in (\Sigma p \in Man)\, WalkedIn(p))\ \mathbf{require}\ y : Man\ \mathbf{in}\ SatDown(y)$$

The meaning of a sentence is a logical proposition.

~~The meaning of a sentence is a logical proposition.~~

~~The meaning of a sentence is a logical proposition.~~

The meaning of a sentence is a type-theoretic expression which may evaluate to a canonical proposition.

~~The meaning of a sentence is a logical proposition.~~

The meaning of a sentence is a type-theoretic expression which may evaluate to a canonical proposition.

~~The meaning of a sentence is a logical proposition.~~

The meaning of a sentence is a type-theoretic expression which may evaluate to a canonical proposition.

What we want:

$$(\Sigma x \in (\Sigma p \in \mathit{Man})\, \mathit{WalkedIn}(p))\, \mathbf{require}\ y : \mathit{Man}\ \mathbf{in}\ \mathit{SatDown}(y)$$

$$\sim$$

$$(\Sigma x \in (\Sigma p \in \mathit{Man})\, \mathit{WalkedIn}(p))\, \pi_1(x)$$

~~The meaning of a sentence is a logical proposition.~~

The meaning of a sentence is a type-theoretic expression which may evaluate to a canonical proposition.

What we want:

$$(\Sigma x \in (\Sigma p \in \mathit{Man})\, \mathit{WalkedIn}(p))\, \mathbf{require}\ y : \mathit{Man}\ \mathbf{in}\ \mathit{SatDown}(y)$$

$$\sim$$

$$(\Sigma x \in (\Sigma p \in \mathit{Man})\, \mathit{WalkedIn}(p))\, \pi_1(x)$$

$$\boxed{\text{where } M \sim N \stackrel{\text{def}}{=} (M \preceq N) \wedge (N \preceq M)}$$

EVERY GRAMMATICAL SENTENCE HAS A MEANING

...but only some of them denote propositions (types)!

A NEGATIVE EXAMPLE

A NEGATIVE EXAMPLE

$[\![$ *The unicorn ran a marathon* $]\!]$

$\bigtriangledown$

A NEGATIVE EXAMPLE

$$[\![\textit{The unicorn ran a marathon}]\!]$$

$$\triangledown$$

$$\textbf{require}\ x : \mathit{Unicorn}\ \textbf{in}\ (\Sigma y \in \mathit{Marathon})\, \mathit{Ran}(x; y)$$

$$[\![\textit{The unicorn ran a marathon}]\!]$$

$$\triangledown$$

$$\textbf{require } x : \mathit{Unicorn} \textbf{ in } (\Sigma y \in \mathit{Marathon})\, \mathit{Ran}(x; y)$$

(not a proposition)

A POSITIVE EXAMPLE

A POSITIVE EXAMPLE

$[\![$ *The President ran a marathon* $]\!]$

$\triangledown$

$$\llbracket \textit{ The President ran a marathon } \rrbracket$$

$$\triangledown$$

$$\mathbf{require}\ x : \mathit{President}\ \mathbf{in}\ (\Sigma y \in \mathit{Marathon})\, \mathit{Ran}(x; y)$$

$$\textbf{require}\ x : \mathit{President}\ \textbf{in}\ (\Sigma y \in \mathit{Marathon})\, \mathit{Ran}(x; y)$$

A POSITIVE EXAMPLE

$$\textbf{require}\ x : \mathit{President}\ \textbf{in}\ (\Sigma y \in \mathit{Marathon})\, \mathit{Ran}(x; y)$$
$$\Downarrow$$
$$(\Sigma y \in \mathit{Marathon})\, \mathit{Ran}(\mathit{Obama}; y)$$

$$\textbf{require}\ x : \mathit{President}\ \textbf{in}\ (\Sigma y \in \mathit{Marathon})\, \mathit{Ran}(x; y)$$

$$\Downarrow$$

$$(\Sigma y \in \mathit{Marathon})\, \mathit{Ran}(\mathit{Obama}; y)$$

Evaluation is now non-deterministic

Is **require** COMPUTATIONALLY EFFECTIVE?

Yes, but we need two things:

Yes, but we need two things:

1. knowledge-sensitive judgments

Yes, but we need two things:

1. knowledge-sensitive judgments
2. non-deterministic computation

Yes, but we need two things:

1. knowledge-sensitive judgments (forcing)
2. non-deterministic computation

Yes, but we need two things:

1. knowledge-sensitive judgments (forcing)
2. non-deterministic computation

$$\boxed{w \Vdash \mathscr{J}}$$

Yes, but we need two things:

1. knowledge-sensitive judgments (forcing)
2. deterministic computation (use choice sequences)

$$\boxed{\alpha \Vdash_{w} \mathscr{J}}$$

ASSERTION ACTS IN TIME

assertion acts (judgments) are intensional (local)

assertion acts (judgments) are intensional (local)

$$|_x \mathscr{J}(x) \qquad \text{(general judgment)}$$

assertion acts (judgments) are intensional (local)

$$|_x \mathscr{J}(x) \qquad \text{(general judgment)}$$

$$\mathscr{J}_2 \ (\mathscr{J}_1) \qquad \text{(hypothetical judgment)}$$

BETH-KRIPKE SEMANTICS FOR ASSERTIONS

assertion acts (judgments) are intensional (local)

$\mid_x \mathscr{J}(x)$	(general judgment)
$\mathscr{J}_2 \ (\mathscr{J}_1)$	(hypothetical judgment)
$M \Downarrow N$	(evaluation)

assertion acts (judgments) are intensional (local)

$\mid_x \mathscr{J}(x)$	(general judgment)
$\mathscr{J}_2\ (\mathscr{J}_1)$	(hypothetical judgment)
$M \Downarrow N$	(evaluation)
$A = B\ \mathit{type}$	(typehood)

assertion acts (judgments) are intensional (local)

$\mid_x \mathscr{J}(x)$	(general judgment)
$\mathscr{J}_2\ (\mathscr{J}_1)$	(hypothetical judgment)
$M \Downarrow N$	(evaluation)
$A = B\ \mathit{type}$	(typehood)
$A\ \mathit{true}$	(truth)

assertion acts (judgments) are intensional (local)

$\mid_x \mathscr{J}(x)$	(general judgment)
$\mathscr{J}_2\ (\mathscr{J}_1)$	(hypothetical judgment)
$M \Downarrow N$	(evaluation)
$A = B\ \mathit{type}$	(typehood)
$A\ \mathit{true}$	(truth)
$M = N \in A$	(membership)

assertion acts (judgments) are intensional (local)

$$w \Vdash |_x\, \mathscr{J}(x) \qquad \text{(general judgment)}$$
$$w \Vdash \mathscr{J}_2\ (\mathscr{J}_1) \qquad \text{(hypothetical judgment)}$$
$$w \Vdash M \Downarrow N \qquad \text{(evaluation)}$$
$$w \Vdash A = B\ \mathit{type} \qquad \text{(typehood)}$$
$$w \Vdash A\ \mathit{true} \qquad \text{(truth)}$$
$$w \Vdash M = N \in A \qquad \text{(membership)}$$

$$w \Vdash |_x \mathscr{J}(x) \qquad \text{(general judgment)}$$

$$w \Vdash \mathscr{J}_2 \ (\mathscr{J}_1) \qquad \text{(hypothetical judgment)}$$

SEMANTICS OF HIGHER-ORDER ASSERTIONS

$$w \Vdash |_x \mathscr{J}(x) \iff \cdots$$

$$w \Vdash \mathscr{J}_2 (\mathscr{J}_1) \iff \cdots$$

SEMANTICS OF HIGHER-ORDER ASSERTIONS

$$w \Vdash \rceil_x \mathscr{J}(x) \iff \forall u \geq w. \forall x \in \mathscr{M}_u.\ u \Vdash \mathscr{J}(x)$$

$$w \Vdash \mathscr{J}_2\ (\mathscr{J}_1) \iff \forall u \geq w.\ u \Vdash \mathscr{J}_1 \Rightarrow u \Vdash \mathscr{J}_2$$

SEMANTICS OF HIGHER-ORDER ASSERTIONS

$$
\begin{aligned}
w \Vdash |_x\, \mathscr{J}(x) &\iff \forall u \geq w. \forall x \in \mathscr{M}_u.\ u \Vdash \mathscr{J}(x) \\
w \Vdash \mathscr{J}_2\ (\mathscr{J}_1) &\iff \forall u \geq w.\ u \Vdash \mathscr{J}_1 \Rightarrow u \Vdash \mathscr{J}_2
\end{aligned}
$$

where $\mathscr{M}_w$ is the species of constructions that have been effected by stage w

$$w \Vdash A \ \mathit{true} \iff$$

$$w \Vdash A\ \mathit{true} \iff \exists m \in \mathscr{M}_w.$$

$$w \Vdash A\ \mathit{true} \iff \exists m \in \mathscr{M}_w.\ w \Vdash m = m \in A$$

$$w \Vdash A \; \mathit{true} \iff$$

$$w \Vdash A\ \mathit{true} \iff \exists \mathfrak{B}\ \textbf{bars}\ w.$$

$$w \Vdash A\ \mathit{true} \iff \exists \mathfrak{B}\ \textbf{bars}\ w. \forall u \in \mathfrak{B}.$$

$$w \Vdash A\ \mathit{true} \iff \exists \mathfrak{B}\ \textbf{bars}\ w. \forall u \in \mathfrak{B}. \exists m \in \mathscr{M}_u.$$

$$w \Vdash A\ \mathit{true} \iff \exists \mathfrak{B}\ \textbf{bars}\ w. \forall u \in \mathfrak{B}. \exists m \in \mathscr{M}_u.\ u \Vdash m = m \in A$$

THE **require** ORACLE: DYNAMICS

require — DYNAMICS

$$\frac{}{\mathfrak{S}(\langle\rangle)} \qquad \frac{\mathfrak{S}(\vec{u}) \quad |_n\, \rho(n) < n \ \ (n \in \mathbb{N}^+)}{\mathfrak{S}(\vec{u} \frown \rho)} \qquad \text{(spread law)}$$

$$\frac{\alpha \models_t A \Downarrow A' \quad |\varkappa_{A'}(t)| = \ell \quad \mathsf{hd}(\alpha)(\ell) = j \quad \mathsf{tl}(\alpha) \models_t \left[\varkappa_{A'}(j)/x\right] N \Downarrow N'}{\alpha \models_t \textbf{require}\ x : A\ \textbf{in}\ N \Downarrow N'}$$

(for $\alpha \in \mathfrak{S}$)

Questions?