ORACLES AND CHOICE SEQUENCES FOR TYPE-THEORETIC PRAGMATICS

Jon Sterling

October 8, 2015

joint work with Darryl McAdams

INTRODUCTION

NATURAL LANGUAGE SEMANTICS AND ANAPHORA

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

$\triangledown$

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

$$\triangledown$$

$$(\Sigma p \in Woman)$$

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

$$\triangledown$$

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

$[\![$*She sat down*$]\!]$

$\triangledown$

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

$$\triangledown$$

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

$[\![$*A woman walked in. She sat down*$]\!]$

$\bigtriangledown$

$$[\![\textit{A woman walked in. She sat down}]\!]$$

$$\triangledown$$

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

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

$$[\![\textit{A woman walked in. She sat down}]\!]$$

$$\triangledown$$

$$(\Sigma x \in (\Sigma p \in Woman)\, WalkedIn(p))\, SatDown(\pi_1(x))$$

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

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

$$\triangledown$$

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

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

$$\triangledown$$

$$(\Pi p \in (\Sigma x \in \mathit{Farmer})\,(\Sigma y \in \mathit{Donkey})\,\mathit{Owns}(x; y))\,\mathit{Beats}(\pi_1(p); \pi_1(\pi_2(p)))$$

TWO THINGS TO DEAL WITH

- terms for presuppositions

TWO THINGS TO DEAL WITH

- terms for presuppositions
- resolution of presuppositions

TWO THINGS TO DEAL WITH

- 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

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

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

require — FORMAL RULES

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

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

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

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

$$[\![\textit{A woman walked in. She sat down}]\!]$$
$$\triangledown$$
$$(\Sigma x \in (\Sigma p \in \mathit{Woman})\,\mathit{WalkedIn}(p))\ \mathbf{require}\ y : \mathit{Woman}\ \mathbf{in}\ \mathit{SatDown}(y)$$

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:

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

$$\rightsquigarrow$$

$$(\Sigma x \in (\Sigma p \in Woman)\, WalkedIn(p))\, \pi_1(x)$$

~~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:

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

$$\sim$$

$$(\Sigma x \in (\Sigma p \in Woman)\, WalkedIn(p))\, \pi_1(x)$$

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

EVERY GRAMMATICAL SENTENCE HAS A MEANING

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

require—NAÏVE DYNAMICS

$$\frac{M \in A \quad [M/x]\, N \Downarrow N'}{\textbf{require}\ x : A\ \textbf{in}\ N \Downarrow N'} \qquad (??)$$

require—NAÏVE DYNAMICS

$$\frac{M \in A \quad [M/x]\,N \Downarrow N'}{\textbf{require}\; x : A \;\textbf{in}\; N \Downarrow N'} \quad \text{(??)}$$

Can the above be made precise?

require—NAÏVE DYNAMICS

$$\frac{M \in A \quad [M/x]\,N \Downarrow N'}{\textbf{require}\ x : A\ \textbf{in}\ N \Downarrow N'} \qquad (??)$$

Can the above be made precise? There are two problems:

require—NAÏVE DYNAMICS

$$\frac{M \in A \quad [M/x]\,N \Downarrow N'}{\textbf{require}\; x : A \;\textbf{in}\; N \Downarrow N'} \qquad (??)$$

Can the above be made precise? There are two problems:

1. impredicativity

$$\frac{M \in A \quad [M/x]\,N \Downarrow N'}{\textbf{require}\ x : A\ \textbf{in}\ N \Downarrow N'} \tag{??}$$

Can the above be made precise? There are two problems:

1. impredicativity
2. non-determinism

(HOLD THAT THOUGHT)

A POSITIVE EXAMPLE

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

$$\triangledown$$

A POSITIVE EXAMPLE

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

$$\triangledown$$

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

A POSITIVE EXAMPLE

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

A POSITIVE EXAMPLE

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

$$\Downarrow$$

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

A NEGATIVE EXAMPLE

A NEGATIVE EXAMPLE

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

$\triangledown$

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

$$\triangledown$$

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

A NEGATIVE EXAMPLE

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

$$\triangledown$$

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

(not a proposition)

Is **require** COMPUTATIONALLY EFFECTIVE?

Yes, but we need two things:

Yes, but we need two things:

1. judgments shall be local / sensitive to knowledge

Yes, but we need two things:

1. judgments shall be local / sensitive to knowledge
2. non-determinism must be eliminated

Yes, but we need two things:

1. judgments shall be local / sensitive to knowledge (via forcing)
2. non-determinism must be eliminated

Yes, but we need two things:

1. judgments shall be local / sensitive to knowledge (via forcing)
2. non-determinism must be eliminated

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

Yes, but we need two things:

1. judgments shall be local / sensitive to knowledge (via forcing)
2. non-determinism must be eliminated (via choice sequences)

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

Yes, but we need two things:

1. judgments shall be local / sensitive to knowledge (via forcing)
2. non-determinism must be eliminated (via choice sequences)

$$\boxed{\alpha \Vdash_w \mathscr{J}}$$

THE CREATING SUBJECT

All mathematics is a mental construction performed by an idealized subject, subject to the following observations about knowledge:

All mathematics is a mental construction performed by an idealized subject, subject to the following observations about knowledge:

All mathematics is a mental construction performed by an idealized subject, subject to the following observations about knowledge:

1. experiences are never forgotten (monotonicity, functoriality)

All mathematics is a mental construction performed by an idealized subject, subject to the following observations about knowledge:

1. experiences are never forgotten (monotonicity, functoriality)
2. at a point in time, the subject knows whether or not it has experienced a judgment (decidability)

All mathematics is a mental construction performed by an idealized subject, subject to the following observations about knowledge:

1. experiences are never forgotten (monotonicity, functoriality)
2. at a point in time, the subject knows whether or not it has experienced a judgment (decidability)

Corollary
The meaning of a judgment $\mathscr{J}$ *must be explained in terms of its forcing condition,* $\boxed{w \Vdash \mathscr{J}}$*, for any stage/world* w.

...

2. at a point in time, the subject knows whether or not it has experienced a judgment (decidability)

...

2. at a point in time, the subject knows whether or not it has experienced a judgment (decidability)

Remark

Contra Dummett, I by no means take the above as requiring that the following shall be true in a constructive metatheory, divorced from time:

$$\forall w. \forall \mathscr{J}.\ [\![w \Vdash \mathscr{J}]\!] \lor \neg [\![w \Vdash \mathscr{J}]\!] \qquad \text{(Dummett's infelicity)}$$

The above is impossible in a Beth model.

MEANING THEORY WITHOUT UNIFORMITY CONSTRAINTS

logical consequence $\Rightarrow$ semantic consequence

Brouwer?, Martin-Löf, Sundholm $\Rightarrow$ *Brouwer?, Heyting, Allen, Zeilberger*

logical consequence $\Rightarrow$ **semantic consequence**

Brouwer?, Martin-Löf, Sundholm $\Rightarrow$ *Brouwer?, Heyting, Allen, Zeilberger*

proof conditions $\Rightarrow$ **assertion conditions**

Martin-Löf, Sundholm $\Rightarrow$ *Heyting, Van Atten*

logical consequence $\Rightarrow$ **semantic consequence**

Brouwer?, Martin-Löf, Sundholm $\Rightarrow$ *Brouwer?, Heyting, Allen, Zeilberger*

proof conditions $\Rightarrow$ **assertion conditions**

Martin-Löf, Sundholm $\Rightarrow$ *Heyting, Van Atten*

global meaning explanation $\Rightarrow$ **local meaning explanation**

Husserl, Dummett, Martin-Löf $\Rightarrow$ *Brouwer?, Beth, Kripke, Grothendieck, Lawvere, Joyal*

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)

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

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

assertion acts (judgments) are intensional (local)

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

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

$$M \Downarrow N \qquad \text{(evaluation)}$$

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)
$A\ \mathit{type}$	(typehood)

assertion acts (judgments) are intensional (local)

$\vert_x \mathscr{J}(x)$	(general judgment)
$\mathscr{J}_2\ (\mathscr{J}_1)$	(hypothetical judgment)
$M \Downarrow N$	(evaluation)
$A\ \mathit{type}$	(typehood)
$A\ \mathit{verif}$	(verification)

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)
$A\ \mathit{type}$	(typehood)
$A\ \mathit{verif}$	(verification)
$A\ \mathit{true}$	(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)
$A\ \mathit{type}$	(typehood)
$A\ \mathit{verif}$	(verification)
$A\ \mathit{true}$	(truth)
$M = N \in A$	(membership)

BETH-KRIPKE SEMANTICS FOR ASSERTIONS

assertion acts (judgments) are intensional (local)

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

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

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

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

SEMANTICS OF HIGHER-ORDER ASSERTIONS

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

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

The meaning of a proposition/type is an intensional (world-indexed) specification of verification acts, i.e. a local meaning explanation for $\boxed{w \Vdash P\ \mathit{verif}}$ (and its synthesis).

The meaning of a proposition/type is an intensional (world-indexed) specification of verification acts, i.e. a local meaning explanation for $\boxed{w \Vdash P\ \mathit{verif}}$ (and its synthesis).

For a type A, implicit in the explanation of $w \Vdash P\ \mathit{verif}$ is a PER $\mathscr{V}[\![A]\!]_w \subseteq \mathscr{D}_w \times \mathscr{D}_w$ whose members reflect the computational content of verification acts.

Truth (justification) consists in recognizing the effectiveness of a procedure for verification.

Truth (justification) consists in recognizing the effectiveness of a procedure for verification.

In the model, this corresponds to the inevitability of verification (i.e. a bar, in which verification occurs at all nodes):

$$w \Vdash A \ \mathit{true} \iff \exists \mathfrak{B} \ \textbf{bars} \ w. \forall u \in \mathfrak{B}.\ u \Vdash A \ \mathit{verif} \qquad \text{(due to Dummett)}$$

The analytic judgments of type theory are reflections on mathematical activity.

The analytic judgments of type theory are reflections on mathematical activity.

1. **Canonical membership** reflects **verification**

$$\boxed{\mathscr{V}\,[\![A]\!]_{w}(M,N) \quad \rtimes\!\!\!\ltimes \quad w \Vdash A\ \mathit{verif}}$$

The analytic judgments of type theory are reflections on mathematical activity.

1. **Canonical membership** reflects **verification**
2. **Membership** reflects **justification**

$$\boxed{w \Vdash M = N \in A \quad \bowtie \quad w \Vdash A \ \mathit{true}}$$

The analytic judgments of type theory are reflections on mathematical activity.

1. **Canonical membership** reflects **verification**
2. **Membership** reflects **justification**
3. **Computation** reflects **the recognition of a bar**

$$\boxed{w \Vdash M = N \in A \quad \rtimes\!\!\ltimes \quad w \Vdash A\ \mathit{true}}$$

The analytic judgments of type theory are reflections on mathematical activity.

1. **Canonical membership** reflects **verification**
2. **Membership** reflects **justification**
3. **Computation** reflects **the recognition of a bar**

$$\wedge \left\{ \begin{array}{l} w \Vdash M \Downarrow M' \\ w \Vdash N \Downarrow N' \\ w \Vdash \mathscr{V}[\![A]\!]_w (M', N') \end{array} \right\} \quad \rtimes\!\!\!\ltimes \quad \exists \mathfrak{B} \ \textbf{bars}\ w. \forall u \in \mathfrak{B}.\ u \Vdash A\ \mathit{verif}$$

THE **require** ORACLE: DYNAMICS

require — DYNAMICS

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

(spread law)

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

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

Questions?