On the Dynamics of Proper Names

Philippe de Groote, Ekaterina Lebedeva

INRIA Nancy - Calligramme

CAuLD Workshop “Logical Methods for Discourse”

2009, December 15th

Goal

Express lexical semantics of a proper name

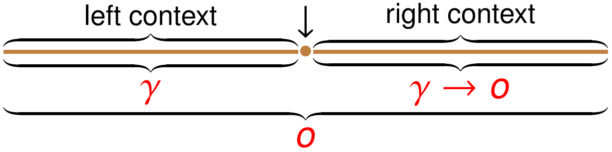
- compositional (Montague's homomorphism requirement)
- in a logical form

- interpretation of a proper name is a constant denoting the corresponding individual
- supposes a preliminary knowledge of all possible individuals
- formal language includes as many constants as there are individuals in the model

Geurts “Good news about the Description theory of names”, 1977

- there are no fundamental semantic differences between names and definite NPs
- names and definite NPs are presuppositional expressions
- names are used to refer to an object that speaker and hearer take to be given
- gives linguistic and philosophical insight
- no technical details

- Montague's system can be extended to account for Gricean conventional implicatures
- Each English phrase is associated with two expressions:
    - the extension expression (assertion, proffered content)
    - the conventional implicature expression (presupposition)
- Binding problem



$$[\![S]\!] = \gamma \rightarrow (\gamma \rightarrow o) \rightarrow o$$

$$[\![D]\!] = \gamma \rightarrow (\gamma \rightarrow o) \rightarrow o$$

$[\![John]\!] = \lambda P.\lambda e\phi.P\mathbf{j}(\mathbf{j} :: e)\phi$

- each time *John* appears in the discourse, **j** will be asserted into the environment
- when embedded under negation, *John* will disappear from the environment

Sentence: $[\![\mathbb{S}]\!] = \Omega \times \Omega$

$\mathbb{S} = \langle \mathbb{S}_1, \mathbb{S}_2 \rangle$

$\mathbb{S}_1$ - proffered content

$\mathbb{S}_2$ - presupposition content

Discourse: $[\![\mathbb{D}]\!] = (\gamma \to o) \to o$

Discourse Update:

$$\mathbb{D}' = \mathbb{D} \wedge \mathbb{S}_2 \wedge \mathbb{S}_1 = \lambda\phi.\, \mathbb{D}\,(\lambda e.\, \mathbb{S}_2\, e\,(\lambda e.\, \mathbb{S}_1\, e\, \phi))$$

$$[\![NP]\!] = (\iota \to [\![S]\!]) \to [\![S]\!] = (\iota \to (\Omega \times \Omega)) \to (\Omega \times \Omega)$$

$$[\![John]\!] = \lambda P.\langle \mathbb{J}_1, \mathbb{J}_2 \rangle$$

$$P : \iota \to (\Omega \times \Omega)$$

$$\mathbb{J}_1 = \lambda e\phi.\pi_1\,(P\,(sel_{\mathrm{John}}\; e))\,e\,\phi$$
$$\mathbb{J}_2 = \lambda e\phi.\exists j.\,(\mathbf{john}\,j) \wedge \pi_2\,(P\,j)\,(j{::}e)\,\phi$$

- problem: introduces a new reference marker for every occurence of the proper name (even if a reference marker already exists in the environment)
- solution: conditional handling of presupposition content

$$[\![NP]\!] = (\iota \rightarrow [\![S]\!]) \rightarrow [\![S]\!] = (\iota \rightarrow (\Omega \times \Omega)) \rightarrow (\Omega \times \Omega)$$

$$[\![John]\!] = \lambda P. \langle \mathbb{J}_1, \mathbb{J}_2 \rangle$$

$$P : \iota \rightarrow (\Omega \times \Omega)$$

$$\mathbb{J}_1 = \lambda e\phi.\, \pi_1\, (P\, (sel_{\mathrm{John}}\, e))\, e\, \phi$$
$$\mathbb{J}_2 = \lambda e\phi.\, \exists j.\, (\mathbf{john}\, j) \wedge \pi_2\, (P\, j)\, (j{::}e)\, \phi$$

- problem: introduces a new reference marker for every occurence of the proper name (even if a reference marker already exists in the environment)
- solution: conditional handling of presupposition content

$$[\![John]\!] = \lambda P.\langle \mathbb{J}_1, \mathbb{J}_2 \rangle$$

$$\mathbb{J}_1 = \lambda e\phi.\, \pi_1\, (P\, (sel_{\text{John}}\, e))\, e\, \phi$$

$$\mathbb{J}_2 = \lambda e\phi.\ \text{if } (\text{sel}_{John} e) = \text{fail then } \mathbb{J}_2' \text{ else } \mathbb{J}_2''$$

$$\mathbb{J}_2' = \lambda e\phi.\, \exists j.\, (\textbf{john}\, j) \wedge \pi_2\, (P\, j)\, (j{::}e)\, \phi$$

$$\mathbb{J}_2'' = \lambda e\phi.\, \pi_2\, (P\, (sel_{\text{John}}\, e))\, e\, \phi$$

$$\mathbb{J}_2 = \lambda e\phi.\, (\mathbb{J}_2''\, e\, \phi\ \texttt{handle}\ \textit{failure}\ \texttt{with}\ \mathbb{J}_2'\, e\, \phi)$$

$$[\![John]\!] = \lambda P.\langle \mathbb{J}_1, \mathbb{J}_2 \rangle$$

$$\mathbb{J}_1 = \lambda e\phi.\pi_1 (P (sel_{\mathrm{John}}\, e))\, e\, \phi$$

$$\mathbb{J}_2 = \lambda e\phi.\ \text{if } (\text{sel}_{John} e) = \text{fail then } \mathbb{J}_2' \text{ else } \mathbb{J}_2''$$

$$\mathbb{J}_2' = \lambda e\phi.\exists j.(\mathbf{john}\, j) \wedge \pi_2 (P\, j)\, (j{::}e)\, \phi$$

$$\mathbb{J}_2'' = \lambda e\phi.\pi_2 (P (sel_{\mathrm{John}}\, e))\, e\, \phi$$

$$\mathbb{J}_2 = \lambda e\phi.(\mathbb{J}_2''\, e\, \phi \ \texttt{handle}\ \textit{failure}\ \texttt{with}\ \mathbb{J}_2'\, e\, \phi)$$

$$[\![John]\!] = \lambda P.\,\langle \mathbb{J}_1, \mathbb{J}_2 \rangle$$

$$\mathbb{J}_1 = \lambda e\phi.\,\pi_1\,(P\,(sel_{\mathrm{John}}\;e))\;e\;\phi$$

$$\mathbb{J}_2 = \lambda e\phi.\;\text{if }(\text{sel}_{John} e) = \text{fail then } \mathbb{J}_2' \text{ else } \mathbb{J}_2''$$

$$\mathbb{J}_2' = \lambda e\phi.\,\exists j.\,(\mathbf{john}\,j) \wedge \pi_2\,(P\,j)\,(j{::}e)\,\phi$$
$$\mathbb{J}_2'' = \lambda e\phi.\,\pi_2\,(P\,(sel_{\mathrm{John}}\;e))\;e\;\phi$$

$$\mathbb{J}_2 = \lambda e\phi.\,(\mathbb{J}_2''\,e\,\phi\;\;\texttt{handle}\;\;\mathit{failure}\;\;\texttt{with}\;\;\mathbb{J}_2'\,e\,\phi)$$

Sentence: $[\![S]\!] = \gamma \to (\gamma \to o) \to o$

Discourse: $[\![\mathbb{D}]\!] = (\gamma \to o) \to o$

Environment:

$$\gamma = \texttt{ list of } (\iota \times o)$$

$$sel : (\iota \to o) \to \gamma \to \iota$$

$$sel\ P\ [\,] = \texttt{ Raise (fail P)}$$
$$sel\ P\ ((x, \Sigma) :: tl) = \texttt{if } (\Sigma \vdash Px) \texttt{ then } x \texttt{ else } (sel\ P\ tl)$$

John loves Mary

$$\lambda e\phi.\,(\mathbf{love}\,(sel\,(\text{named “John”})\,e)\,(sel\,(\text{named “Mary”})\,e))\;\wedge\;(\phi\,e)$$

$$\begin{aligned}\mathbb{D}\curlywedge\mathbb{S} \;=\;& \lambda\phi.\,\mathbb{D}(\lambda e.\mathbb{S}e\phi)\\ & \texttt{handle}\;\;(\mathit{fail}\;P)\;\;\texttt{with}\\ & \lambda\phi.\mathbb{D}(\lambda e.\exists x.(Px)\wedge\phi((x,Px)::e))\curlywedge\mathbb{S}\end{aligned}$$

- $f(t) \rightsquigarrow f(\texttt{Raise E}) \rightsquigarrow (\texttt{Raise E})$
- $f(t) \rightsquigarrow (\texttt{Raise E})(t) \rightsquigarrow (\texttt{Raise E})$
- $\lambda x.(t) \rightsquigarrow \lambda x.(\texttt{Raise E}) \rightsquigarrow \;\; ?$

$$\lambda e\phi.\exists({\color{red}\lambda x}.Px\wedge\phi((x,T)::e))$$

${\color{red}\lambda x}.(\texttt{Raise E}) \rightsquigarrow (\texttt{Raise E})$ provided $x \notin FV(\texttt{E})$
otherwise accommodate in the right place

Conclusion

- Modeling the dynamics of presupposition with the exception handling mechanisms
- The approach can be extended for other presupposition triggers
- A lot to be done!
  (develop the representation of environment, accounts for local/intermediate accommodation, avoid type-raising, . . .)

The End

- Thank you!