Explicit world variables vs. implicit indices

Thomas Ede Zimmermann
Goethe-Universität Frankfurt

when our topic is modality [...] we introduce modal operators to create a special-purpose, nonextensional logic. Why this departure from our custom? Is it a historical accident, or was it forced on us somehow by the very nature of the topic of modaIity? (Lewis 1968: 113)

1 Introduction

(1) John loved Mary.

(2) **P** L(j,m)

(3) $(\exists\, t < t_0)\ \mathrm{L}_t(\mathrm{j},\mathrm{m})$

(4) John knows that it is raining.

(5) $\Box_j$ r

(6) $(\forall w)[w_0 \mathrm{K}_j w\ \rightarrow\ \mathrm{r}_w]$

2 A Montagovian Perspective

Montague (1970)

2.1 Extensions and Intensions

(7) $||snores||$
$=$ that function S such that, for any extension x of a referential subject N:
$S(x) = \vdash\ N\ snores\ \dashv$

- $||snores||(||N||) =\ ||\ N\ snores\ ||$

(8) $||loves\ Mary||$
$=$ that function M such that, for any extension x of a referential subject N:
$M(x) = \vdash\ N\ \ loves\ \ Mary \dashv$

- $||loves\ Mary||(||N||) =\ ||\ N\ loves\ Mary\ ||$

(9) $||loves||$
$=$ that function L such that, for any extension x of a referential object N:
$L(x) =\ ||\ loves\ N\ ||$

- $||loves||(||N||) =\ ||\ loves\ N||$

(10) $||nobody||$
$=$ that function Q such that, for any extension E of a predicate P:
$Q(E) = \vdash$ *Nobody P* $\dashv$

- $||Nobody||(||P||) = ||$ *Nobody P* $||$

(11) *Fregean heuristics* cf. Frege (1891, 1892)

a. $||X||$
$=$ that function f such that, for any [possible] extension x of a possible sister node Y:
$$f(x) = ||X\ Y||.$$

b. $||X||(||Y||) = ||\ X\ Y\ ||$

c. *Prerequisites:*

- The extension of the sister node Y in (11-a) must be determined beforehand – either (i) by previous applications of (11-a), or (ii) by independent assumptions.
- Similarly, (11-a) can only be applied after the extension of the mother node XY has been determined.
- The extension of the mother node XY must depend on the extension of Y in the first place: if Y is replaced by some co-extensional Z, then the resulting expression XZ needs to have the same extension as XY.

(12) $||X\ Y|| = ||X||(||Y||)$

(13) $||believe||$
$=$ that function B such that, for any sense p of a sentence S:
$B(p)\ =\ ||believe\ that\ S||$

(14) $f(x)\ =\ ||X\ Y||^{w^*}\ =\ ||X||^{w^*}(||Y||^{w^*})$

(15) The intension $||X||^\wedge$ of an expression X is that function f such that, for any possible world w:
$f(w) = ||X||^w$

(16) The extension $||X||^{w^*}$ of an expression X (given a world w^*) is the value assigned (to w^*) by its intension:
$||X||^{w^*} = ||X||^\wedge(w^*)$

(17) *Non-Fregean Principles* Carnap (1947)

a. The intension of an expression determines its (actual) extension – *given the facts*.

b. The (possible) extensions of an expression determine its intension.

c. Extensional compositionality implies intensional compositionality.

(18) a. John knows that an equilateral triangle has three equal sides.
b. John knows that an equiangular triangle has three equal sides.

(19) *Fregean Compositionality* cf. Zimmermann (Forthcoming)

$$||X|| = \begin{cases} \text{that function } f \text{ such that, for any possible sister node } Y\text{:} \\ f(||Y||) = ||X\ Y||, \text{ if such a function } f \text{ exists;} \\ \text{that function } g \text{ such that, for any possible sister node } Y\text{:} \\ g(||Y||^{\wedge}) = ||X\ Y|| \text{ otherwise.} \end{cases}$$

(20) If P is a sentence predicate and Q is its quantificational subject, then:
$||Q\ P|| = ||Q||(||P||)$.

(21) If V is a clause-embedding verb and S is its complement, then:
$||V\ S|| = ||V||(||S||^{\wedge})$.

(22) Lucifer loves nobody.

(23) If V is a transitive verb and Q is its quantificational object, then $||V\ Q||$ is that function f such that, for any individual x:
- $f(x) = ||Q||(g)$
  where g is that function such that, for any individual y:
- $g(y) = ||V||(y)(x)$.

- $||V\ Q|| = \lambda x.||Q||(\lambda y.||V||(y)(x))$

2.2 *Ty2*: Explicit Index Variables

(24) *Definition*

a. The set of *two-sorted (functional) types* is the smallest set *2T* such that the following holds:
  - $t \in 2T$; $e \in 2T$; $s \in 2T$;
  - $(a,b) \in 2T$ whenever $a \in 2T$ and $b \in 2T$.

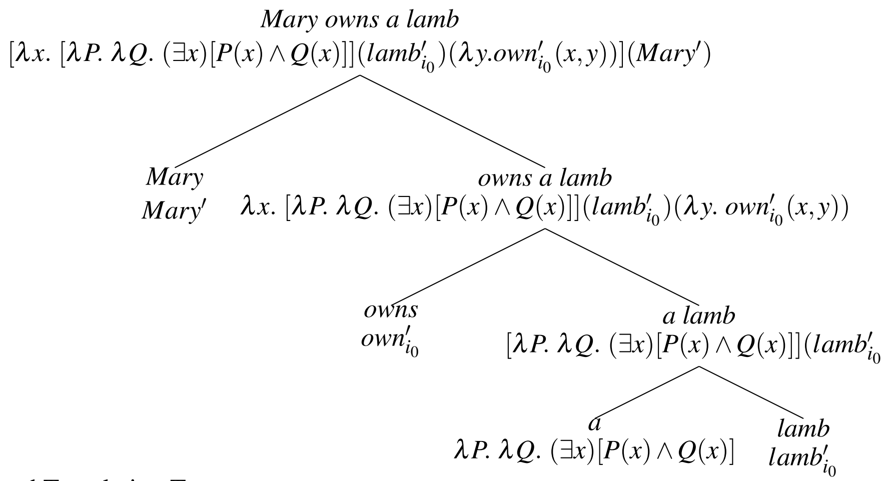
b. For any type $a \in 2T$, the layer D_a is defined according to a's structure:
  - $D_t = \{0,1\}$;
  - D_e is the set of all possible individuals;
  - D_s is the set of all indices;
  - $D_{(a,b)}$ is the set of functions with domain D_a and values in D_b.

Some general features of Ty2

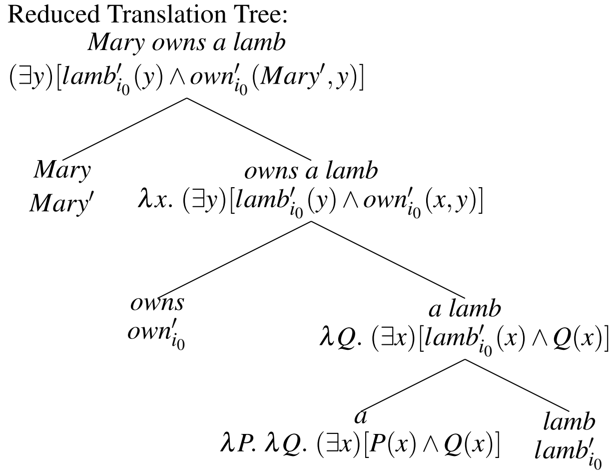
- *Ty2*-formulae are interpreted independently of the translation procedure.
- *Ty2*-formulae may denote objects in any of the layers in the type hierarchy (24).
- *Ty2*-formulae can be combined to match the composition process.

(25) *Montague Trees* Montague (1973)

a. Translation Tree:



b. Reduced Translation Tree:



(26) *λ-Conversion* [aka *β-Conversion*]

- $[\lambda x.\alpha](\beta) \equiv \alpha[^x/_\beta]$,

provided that no variable that is free in β gets bound in $\alpha[^x/_\beta]$.

(27) *Alphabetic Variation* [aka *α-Conversion*]

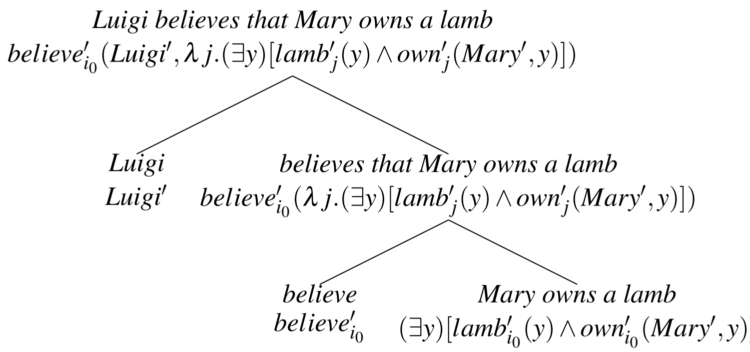
- $[\lambda x.\alpha] \equiv [\lambda y.\alpha[^x/y]]$,

where y is a variable (of the same type as x) that does not occur in α.

(28) *Definition*

If α and β are *Ty2*-formulae of the same type, then:
$\alpha \equiv \beta$ iff for all variable assignments g it holds that $[\![\alpha]\!]^g = [\![\beta]\!]^g$.

(29)



(30) $believe'_{i_0}(\lambda i_0.(\exists y)[lamb'_{i_0}(y) \wedge own'_{i_0}(Mary', y)])$

2.3 *IL*: Implicit Index Parameters

(31) *Definition*

a. The set of *intensional types* is the smallest set IT such that the following holds:
- $t \in IT$; $e \in IT$;
- $(a,b) \in IT$ whenever $a \in IT$ and $b \in IT$;
- $(s,a) \in IT$ whenever $a \in IT$.

b. If $a \in IT$, a formula $\alpha \in Ty2_a$ is *intensional* iff the following holds:
- whenever $\beta \in Ty2_b$ is a sub-formula of α, then: $b \in IT$ or $\beta = i_0$.

c. *Ty2*$^-$ is that fragment of *Ty2* whose basic expressions are either IT-variables or else of the form $c(i_0)$ (where c is a constant of an IT-type (s,a)), and apart from i_0, all variables in λ-prefixes are of IT-types.

(32) From *Ty2*$^-$ to *IL*

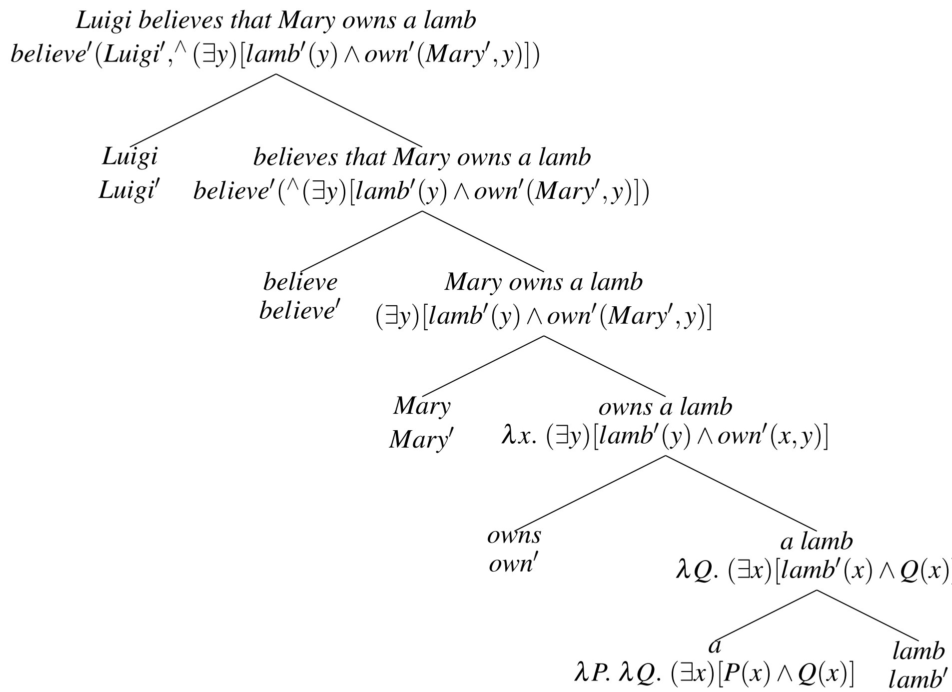
a. *Intensionalising proper names*
- $[\![Luigi']\!]^g$(i) = Luigi.

b. *Restricting and splitting assignments*
- $[\![\alpha]\!]^{g,\mathrm{i}}$

where g is restricted to IT-variables and i$\in D_s$ is the value assigned to (free occurrences of) i_0.

c. *Eliminating* i_0
- Dropping all occurrences of i_0 immediately following IT-constants (in brackets, which are also dropped);
- If i_0 occurs as an argument to some formula $\alpha \in Ty2^-_{(s,a)}$ that is not a constant, it is replaced by the symbol $^\vee$ preceding (the *IL*-version of) α;
- λ-prefixes λi_0. are replaced by the symbol $^\wedge$ [...and the quantifiers $\forall i_0$ and $\exists i_0$ by the symbols $\Box$ and $\Diamond$, respectively].

(33)



3 Evaluating the Montagovian Strategy

(i) *Readability:* The translations of (declarative) sentences should come out as close to ordinary logical notation as possible.

(ii) *Expressivity:* The fragment should only make use of resources needed for indirect interpretation of natural language.

3.1 Logical Observations and Results

Logical Laws

(F) Free occurrences of i_0 in β may be accidentally bound in $\alpha[^x/_\beta]$.

(B) Bound occurrences of i_0 in α cannot be renamed.

(34) a. $believe'_{i_0}(Luigi'_{i_0}, \lambda i_0.(\exists y)[lamb'_{i_0}(y) \wedge own'_{i_0}(Mary'_{i_0}, y)])$

b. $[\lambda P.\ believe'_{i_0}(Luigi', \lambda i_0.(\exists y)[P(y) \wedge own'_{i_0}(Mary', y)])](lamb'_{i_0})$

(35) a. $believe(Luigi', {}^\wedge(\exists y)[lamb'(y) \wedge own'(Mary', y)])$

b. $[\lambda P.\ believe(Luigi'_{i_0}, {}^\wedge(\exists y)[P(y) \wedge own'(Mary'_{i_0}, y)])](lamb')$

(36) *Restricted λ-Conversion*

- $[\lambda x.\alpha](\beta) \equiv \alpha[^{x}/_{\beta}]$ . . . provided that:
- no variable that is free in β gets bound in $\alpha[^{x}/_{\beta}]$, and:
- *either*: no free x in α lies in the scope of an intensional operator, *or*: β is modally closed.

(37) a. $[\lambda i_0.\ \alpha](i_0)$
b. $\alpha[^{i_0}/_{i_0}]$

(38) Eigen-*conversion*
$[\lambda x.\ \alpha](x) \equiv \alpha$

(39) *Down-up-cancellation*
$[^{\vee}[^{\wedge}\alpha]] \equiv \alpha$

(40) $believe'_{i_0}(Luigi'_{i_0}, \lambda j.(\exists y)[lamb'_{i_0}(y) \wedge own'_j(Mary'_j, y)])$

(41) a. $[\lambda x.\ [\lambda y.\ [^{\wedge}\ y] = u(x)](x)](c)$
b. $[\lambda y.\ [^{\wedge}\ y] = u(c)](c)$
c. $[\lambda x.\ [^{\wedge}\ x] = u(x)](c)$
d. $[\lambda x.\ [\lambda y.\ [\lambda i_0.\ y] = u(x)](x)](c_{i_0})$
e. $[\lambda j.\ c_{i_0}] = u(c_{i_0})$

Expressivity

(4) John knows that it is raining.

(5) $\Box_j$ r

(6) $(\forall w)[w_0 \mathrm{K}_j w \ \rightarrow\ \mathrm{r}_w]$

(42) $(\forall j)[K_{i_0}(j)(John'_{i_0}) \ \rightarrow\ it\text{-}is\text{-}raining'_j]$

- (42) does not define the extension of *know* in terms of a constant denoting its intension.
- (42) contains two distinct variables of type s, viz., i_0 and j.

(43) $\lambda p.\ \lambda x.\ (\forall j)[K_{i_0}(j)(x) \ \rightarrow\ p(j)]$

(44) $(\forall j)[K_{i_0}(j)(John'_{i_0}) \rightarrow it\text{-}is\text{-}raining'_j]$
$\equiv$ $[\lambda x.[\lambda F.\ (\forall i_0)[F(i_0)(x) \rightarrow it\text{-}is\text{-}raining'_{i_0}]](K_{i_0})](John'_{i_0})$

(45) $[\lambda x.[\lambda F.\ \Box\ [^{\vee}F(x) \ \rightarrow\ it\text{-}is\text{-}raining']](K)](John')$

(46) a. $[\lambda p.\ [\lambda x.\ [\lambda F.\ (\forall i_0)[F(i_0)(x) \rightarrow p_{i_0}]](K_{i_0})]]$ $\in Ty2^{-}$
b. $[\lambda p.\ [\lambda x.\ [\lambda F.\ \Box\ [[^{\vee}F](x) \rightarrow [^{\vee}p]]](K)]]$ $\in IL$

(47) $(\exists f)(\forall i_0)(\forall p)(\forall q)[if'_{i_0}(p,q) \leftrightarrow (q(f(i_0)(p)))]$

(48) $(\exists R)(\forall i_0)(\forall p)(\forall q)$
$[(\forall j)(\forall k)[[R(i_0)(p)(j) \wedge R(i_0)(p)(k)] \rightarrow j = k] \wedge$
$[if'_{i_0}(p,q) \leftrightarrow (\exists j)[R(i_0)(p)(j) \wedge q(j)]]]$

3.2 Semantic Consequences

> *Montague's Thesis*
> *IL* suffices to express all the extensions and intensions of all expressions as well as all construction meanings of natural language.

Extensions and intensions beyond *IL*: Reference to non-actual worlds

(49) $\lambda i_0.\, \lambda p.\, \lambda q.\, q(sim_{i_0}(p))$ cf. Stalnaker (1968)

(50) $\lambda i_0.\, \lambda q.\, (\exists j)[Sim_{i_0}(p)(j) \wedge q(j)]$

Beyond extension and intension: Generalised *de re* cf. Bäuerle (1983)

(51) Luigi suspects that Mary is afraid that every lamb has been slaughtered.

(52) $suspect'_{i_0}(Luigi_{i_0}, \lambda j.\, afraid'_j(Mary_j,$
$\lambda k.\, (\forall x)[lamb'_j(x) \rightarrow slaughtered'_k(x)]))$

(53) $(\forall x)[lamb'_j(x) \rightarrow slaughtered'_k(x)]$

Capturing (52)

(i) One may attribute to the critical sentences a part-whole structure that deviates from its surface bracketing. Groenendijk & Stokhof (1982)

(ii) One may interpret doubly embedded attitude reports as involving indirect intensions. Zimmermann (Forthcoming)

(54) a.

Mary is afraid that every lamb has been slaughtered
$[\lambda X.\ afraid'_{i_0}(Mary_{i_0}, \lambda i_0.\ (\forall x)[X(x) \rightarrow slaughtered'_{i_0}(x)])](lamb'_{i_0})$

lamb
$lamb'_{i_0}$

Mary is afraid that every X has been slaughtered
$afraid'_{i_0}(Mary_{i_0}, \lambda i_0.\ (\forall x)[X(x) \rightarrow slaughtered'_{i_0}(x)])$

Mary
$Mary'_{i_0}$

is afraid that every X has been slaughtered
$afraid'_{i_0}(\lambda i_0.\ (\forall x)[X(x) \rightarrow slaughtered'_{i_0}(x)])$

is afraid
$afraid'_{i_0}$

every X has been slaughtered
$(\forall x)[X(x) \rightarrow slaughtered'_{i_0}(x)]$

every X
$\lambda Q.\ (\forall x)[X(x) \rightarrow Q(x)]$

has been slaughtered
$slaughtered'_{i_0}$

every
$\lambda P.\ \lambda Q.\ (\forall x)[P(x) \rightarrow Q(x)]$

t_X
X

b.

Luigi suspects that Mary is afraid that every lamb has been slaughtered
$suspect'_{i_0}(Luigi_{i_0}, \lambda i_0.\ [\lambda X.\ afraid'_{i_0}(Mary_{i_0},$
$\lambda i_0.\ (\forall x)[X(x) \rightarrow slaughtered'_{i_0}(x)])](lamb'_{i_0}))$

Luigi
$Luigi'_{i_0}$

suspects that Mary is afraid that every lamb has been slaughtered
$suspect'_{i_0}(\lambda i_0.\ [\lambda X.\ afraid'_{i_0}(Mary_{i_0},$
$\lambda i_0.\ (\forall x)[X(x) \rightarrow slaughtered'_{i_0}(x)])](lamb'_{i_0}))$

suspects
$suspect'_{i_0}$

Mary is afraid that every lamb has been slaughtered
$[\lambda X.\ afraid'_{i_0}(Mary_{i_0}, \lambda i_0.\ (\forall x)[X(x) \rightarrow slaughtered'_{i_0}(x)])](lamb'_{i_0})$

(55) $\lambda j.\ \lambda k.\ (\forall x)[lamb'_j(x) \rightarrow slaughtered'_k(x)]$

Appendix: Some definitions and results

(56) *From Natural Language to Ty2*

a. *Lexical Translations*

- $|Mary| = Mary'_{i_0}$ [where $Mary' \in Con_{(s,e)}$] ...
- $|owns| = own'_{i_0}$ [where $own' \in Con_{(s,(e,(e,t)))}$]; ...
- $|smokes| = smoke'_{i_0}$ [where $smoke' \in Con_{(s,(e,t))}$]; ...
- $|every| = \lambda P.\ \lambda Q.\ (\forall x)[P(x) \to Q(x)]$; ...

...

b. *Compositional Translations*

- If V is a verb phrase and N is its referential (= non-quantificational) subject, then:
  $|N\ V| = |V|(|N|)$;
- If V is a transitive verb and N is its referential object, then:
  $|V\ N| = |V|(|N|)$;
- If V is a transitive verb and Q is its quantificational object, then:
  $|V\ Q| = \lambda x.|Q|(\lambda y.|V|(x,y))$;...

...

(57) *Recursive Definition of Ty2*$^{-}$

- If c is a constant of type $(s,a) \in IT$, $c(i_0) \in Ty2^{-}_{a}$;
- if x is a variable of type $a \in IT$, $x \in Ty2^{-}_{a}$;
- if $\alpha \in Ty2^{-}_{a}$ and $\beta \in Ty2^{-}_{a}$, then $(\alpha = \beta) \in Ty2^{-}_{t}$;
- if $\alpha \in Ty2^{-}_{(a,b)}$ and $\beta \in Ty2^{-}_{a}$, then $\alpha(\beta) \in Ty2^{-}_{b}$;
- if $x \in Var_a$ (where $a \in IT$) and $\alpha \in Ty2^{-}_{b}$, then $[\lambda x.\ \alpha] \in Ty2^{-}_{(a,b)}$;
- if $\alpha \in Ty2^{-}_{(s,a)}$, then $\alpha(i_0) \in Ty2^{-}_{a}$;
- if $\alpha \in Ty2^{-}_{a}$, then $[\lambda i_0.\ \alpha] \in Ty2^{-}_{(s,a)}$.

(58) *Recursive Definition of IL*

- If c is a constant of type $(s,a) \in IT$, $c \in IL_a$;
- if x is a variable of type $a \in IT$, $x \in IL_a$;
- if $\alpha \in IL_a$ and $\beta \in IL_a$, then $\alpha(\beta) \in IL_t$;
- if $\alpha \in IL_{(a,b)}$ and $\beta \in IL_a$, then $\alpha(\beta) \in IL_b$;
- if $x \in Var_a$ (where $a \in IT$) and $\alpha \in IL_b$, then $[\lambda x.\ \alpha] \in IL_{(a,b)}$;
- if $\alpha \in IL_{(s,a)}$, then $[^{\vee}\alpha] \in IL_a$;
- if $\alpha \in IL_a$, then $[^{\wedge}\ \alpha] \in IL_{(s,a)}$.

(59) *From IL to Ty2⁻*

- $[c]^* = c(i_0)$
- $[x]^* = x$
- $[(\alpha = \beta)]^* = (\alpha^* = \beta^*)$
- $[\alpha(\beta)]^* = \alpha^*(\beta^*)$
- $[\lambda x.\ \alpha]^* = [\lambda x.\ \alpha^*]$
- $[{}^\vee\alpha]^* = [\alpha]^*(i_0)$
- $[{}^\wedge\alpha]^* = [\lambda i_0.\ \alpha^*]$

(60) *From Ty2⁻ to IL*

- $[c(i_0)]_* = c$
- $[x]_* = x$
- $[(\alpha = \beta)]_* = (\alpha_* = \beta_*)$
- $[\alpha(\beta)]_* = \alpha_*(\beta_*)$
- $[\lambda x.\ \alpha]_* = [\lambda x.\ \alpha_*]$
- $[\alpha(i_0)]_* = [{}^\vee\alpha_*]$
- $[\lambda i_0.\alpha]_* = [{}^\wedge\alpha_*]$

(61) *Definition* (Gallin 1975: 14)

The class of *modally closed formulae* is the smallest set M of IL-formulae such that the following holds:

- $Var \subseteq M$;
- ${}^\wedge\alpha \in M$ whenever $\alpha \in IL$. [...and $\{\Box\varphi, \Diamond\varphi\} \subseteq M$ whenever $\varphi \in IL_t$]
- $\{\alpha(\beta), [\lambda x.\ \alpha]\} \subseteq M$ whenever $\alpha, \beta \in M \cap IL$, and $x \in Var$;

  [...and $\{(\forall x)\varphi, (\exists x)\varphi\} \subseteq M$ whenever $\varphi \in M \cap IL_t$, and $x \in Var$]

Lemma (Gallin 1975: 14)

For any $a \in IT$, $\alpha \in IL_a$ is modally closed iff i_0 is not free in $[\alpha]^*$.

Lemma (Gallin 1975: 62)

If $\alpha \in IL_a$ (for some $a \in IT$), then $\alpha^* \equiv \alpha$.

Lemma cf. (Zimmermann 1989: 75)

If $\alpha \in Ty2^-_a$ (for some $a \in IT$), then $\alpha_* \equiv \alpha$.

Theorem cf. (Gallin 1975: 105)

If $\varphi \in Ty2_t$ is closed, then there is an $\psi \in IL_t$ such that: $\psi^* \equiv \varphi$.

Theorem (Zimmermann 1989: 75)

If $a \in IT, \alpha \in Ty2_a$, and $b \in IT$, for any $\gamma \in (Var_b \cup Con_b) \setminus \{i_0\}$ occurring freely in α, then there is a $\beta \in IL_a$ such that: $\beta^* \equiv \alpha$.

References

Bäuerle, Rainer. 1983. Pragmatisch-semantische Aspekte der NP-Interpretation. In Manfred Faust, Roland Harweg, Werner Lehfeldt & Götz Wienold (eds.), *Allgemeine Sprachwissenschaft, Sprachtypologie und Textlinguistik*, 121–131. Tübingen: Narr.

Carnap, Rudolf. 1947. *Meaning and Necessity*. Chicago/London: University of Chicago Press.

Frege, Gottlob. 1891. *Function und Begriff*. Jena: Pohle.

Frege, Gottlob. 1892. Über Sinn und Bedeutung. *Zeitschrift für Philosophie und philosophische Kritik* NF 100(1). 25–50.

Gallin, Daniel. 1975. *Intensional and Higher-order Modal Logic*. Amsterdam: North-Holland Pub. Company.

Groenendijk, Jeroen & Martin Stokhof. 1982. Semantic Analysis of *WH*-Complements. *Linguistics and Philosophy* 5(2). 175–233.

Lewis, David K. 1968. Counterpart theory and quantified modal logic. *Journal of Philosophy* 65(5). 113–126.

Montague, Richard. 1970. Universal Grammar. *Theoria* 36(3). 373–398.

Montague, Richard. 1973. The Proper Treatment of Quantification in Ordinary English. In Jaakko Hintikka, Julius Moravcsik & Patrick Suppes (eds.), *Approaches to Natural Language*, 221–242. Dordrecht: Reidel.

Stalnaker, Robert. 1968. A Theory of Conditionals. In Nicholas Rescher (ed.), *Studies in Logical Theory*, 41–55. Oxford: Blackwell.

Zimmermann, Thomas Ede. 1989. Intensional Logic and Two-sorted Type Theory. *Journal of Symbolic Logic* 54(1). 65–77.

Zimmermann, Thomas Ede. Forthcoming. Fregean Compositionality. In Brian Rabern & Derek Ball (eds.), *The Science of Meaning*, Oxford University Press.