A Proof-theoretic Investigation of Natural Language Semantics

Daisuke Bekki

Ochanomizu University, Faculty of Core Research
https://daisukebekki.github.io/

An invited talk at
Mathematical and Computational Linguistics for Proofs (MCLP)
Institut Pascal, Université Paris-Saclay, France
16 September, 2025

Dependent Type Semantics (DTS) (Bekki 2014; Bekki and Mineshima 2017; Bekki 2021)

- A framework of natural language semantics
- Unified approach to (general) inferences and anaphora/presupposition resolution in terms of *type checking* and *proof construction* (cf. Krahmer and Piwek (1999))

Main features:

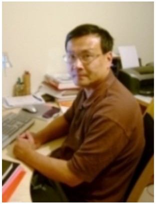
1. **Proof-theoretic semantics**:
   From model theory (denotations and models) to proof theory (proofs and contexts)
2. **Anaphora/Presuppositions**: A proof-theoretic alternative to Dynamic Semantics (DRT, DPL, etc.)
3. **Compositionality**: Syntax-semantics interface via categorial grammars (e.g. CCG, TLG, ACG, etc)
4. **Implementation**: Applications to Natural Language Processing: `https://github.com/DaisukeBekki/lightblue/`

From Dependent Types to Natural Language Semantics

Per Martin-Löf

Martin-Löf (1984) “Intuitionistic type theory”

Natural Language Semantics via Dependent Types

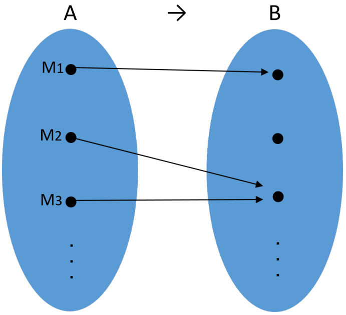
Goran Sundholm	Aarne Ranta	Robin Cooper	Zhaohui Luo
			

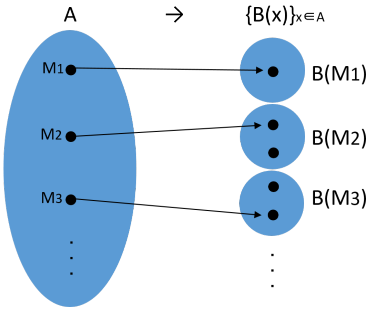
What are Π-types

Π-type is a type of *fibred* functions.

Simple function space

Fibred function space

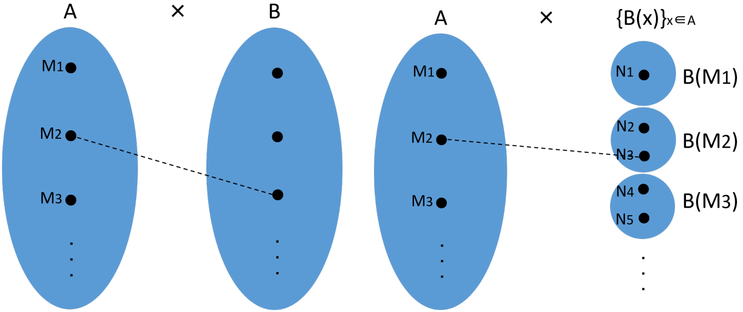




What are Σ-types

Σ-type is a type of *fibred* products.

Simple product space *Fibred* product space



Notations

DTS notation	Standard notation	$x \notin fv(B)$	$x \in fv(B)$
$(x : A) \to B$	$(\Pi x : A)B$	$A \to B$	$(\forall x : A)B$
$(x : A) \times B$ *or* $\begin{bmatrix} x : A \\ B \end{bmatrix}$	$(\Sigma x : A)B$	$A \wedge B$	$(\exists x : A)B$

Scope of the variable in Π-types: $(x : A) \to \boxed{B}$

Scope of the variable in Σ-types: $\begin{bmatrix} x : A \\ \boxed{B} \end{bmatrix}$

Π-type F/I/E rules

$$\dfrac{A : \mathsf{s_1} \quad \begin{array}{c}\overline{x : A}^{\,i} \\ \vdots \\ B : \mathsf{s_2}\end{array}}{(x : A) \to B : \mathsf{s_2}}\,(\Pi F),i \quad where\ (\mathsf{s_1}, \mathsf{s_2}) \in \left\{ \begin{array}{l} (\mathsf{type}, \mathsf{type}), \\ (\mathsf{type}, \mathsf{kind}), \\ (\mathsf{kind}, \mathsf{kind}) \end{array} \right\}.$$

$$\dfrac{A : \mathsf{s} \quad \begin{array}{c}\overline{x : A}^{\,i} \\ \vdots \\ M : B\end{array}}{\lambda x.M : (x : A) \to B}\,(\Pi I),i \quad where\ \mathsf{s} \in \{(\mathsf{type}, \mathsf{kind})\}.$$

$$\dfrac{M : (x : A) \to B \quad N : A}{MN : B[N/x]}\,(\Pi E)$$

Σ-type F/I/E rules

$$\dfrac{A : \text{type} \qquad \begin{matrix} \overline{x : A}^{\,i} \\ \vdots \\ B : \text{type} \end{matrix}}{(x : A) \times B : \text{type}} \; (\Sigma F),i$$

$$\dfrac{M : A \qquad N : B[M/x]}{(M, N) : (x : A) \times B} \; (\Sigma I)$$

$$\dfrac{M : (x : A) \times B}{\pi_1(M) : A} \; (\Sigma E) \qquad \dfrac{M : (x : A) \times B}{\pi_2(M) : B[\pi_1(M)/x]} \; (\Sigma E)$$

Conjunction, Implication, and Negation

Definition

$$\begin{bmatrix} A \\ B \end{bmatrix} \stackrel{def}{\equiv} (x : A) \times B \qquad where\ x \notin fv(B).$$

$$A \to B \stackrel{def}{\equiv} (x : A) \to B \qquad where\ x \notin fv(B).$$

$$\neg A \stackrel{def}{\equiv} (x : A) \to \bot$$

Semantic Representations: Examples

(1) a. Every student is tall.

b. $$\left(u : \begin{bmatrix} x : \text{entity} \\ \textbf{student}(x) \end{bmatrix}\right) \rightarrow \textbf{tall}(\pi_1 u)$$

(2) a. Some student is tall.

b. $$\begin{bmatrix} u : \begin{bmatrix} x : \text{entity} \\ \textbf{student}(x) \end{bmatrix} \\ \textbf{tall}(\pi_1 u) \end{bmatrix}$$

(3) a. No student is tall.

b. $$\neg \begin{bmatrix} u : \begin{bmatrix} x : \text{entity} \\ \textbf{student}(x) \end{bmatrix} \\ \textbf{tall}(\pi_1 u) \end{bmatrix}$$

$where$ $\textbf{student} : \text{entity} \rightarrow \text{type}, \ \textbf{tall} : \text{entity} \rightarrow \text{type}.$

Types adopted in DTS

Types from Martin-Löf Type Theory

- Axioms and Structural rules
- $\mathbf{\Pi}$-type (Dependent function type)
- $\mathbf{\Sigma}$-type (Dependent product type)
- Intensional equality type
- Disjoint union type
- Enumeration type
- Natural number type
- Type universe

New type in DTS

- Underspecified type
  - Anaphora and presupposition triggers (linguistically speaking)
  - Open proofs (logically speaking)

Anaphora in Natural Language

A theory of anaphora

- Anaphora representable by a constant symbol:
  - Deictic use:

    (4) *(Pointing at John)*

    He was born in Detroit.

    $\textbf{bornIn}(j, d)$
  - Coreference:

    (5) John loves a girl who hates him.

    $\exists x(\textbf{girl}(x) \wedge \textbf{love}(j, x) \wedge \textbf{hate}(x, j))$
- Anaphora representable by a variable
  - Bound variable anaphora:

    (6) Every boy loves his father.

    $\forall x(\textbf{boy}(x) \rightarrow \textbf{love}(x, \textbf{fatherOf}(x)))$

A theory of anaphora

- Anaphora not representable by FoL:
  - E-type anaphora:

    (7) A man entered into the park. He whistled.

  - Donkey anaphora:

    (8) Every farmer who owns a donkey beats it.

    (9) If a farmer owns a donkey, he beats it.

- Anaphora not representable by FoL nor dynamic semantics:
  - Syllogistic anaphora:

    (10) Every girl received a present. Some girl opened it.

  - Disjunctive antecedent:

    (11) If Mary sees a horse or a pony, she waves to it.

Donkey anaphora: Geach (1962)

For the donkey sentences (12), a first-order formula (13), whose truth condition is the same as those of (12), is a candidate of its semantic representation (SR). (We only discuss its *strong reading* here. See Tanaka (2021) for its *weak reading*.)

(12) a. Every farmer who owns [a donkey]^{1} beats it$_{1}$.

b. If [a farmer]^{1} owns [a donkey]^{2}, he$_{1}$ beats it$_{2}$.

(13) $\forall x(\mathbf{farmer}(x) \rightarrow \forall y(\mathbf{donkey}(y) \land \mathbf{own}(x,y) \rightarrow \mathbf{beat}(x,y)))$

But the translation from the sentence (12) to (13) is not straightforward since i) the indefinite noun phrase *a donkey* is translated into a universal quantifier in (13) instead of an existential quantifier, and ii) the syntactic structure of (13) does not corresponds to that of (12).

Donkey anaphora: Geach (1962)

(12) a. Every farmer who owns [a donkey]$^{\mathbf{1}}$ beats $\text{it}_{\mathbf{1}}$.

b. If [a farmer]$^{\mathbf{1}}$ owns [a donkey]$^{\mathbf{2}}$, $\text{he}_{\mathbf{1}}$ beats $\text{it}_{\mathbf{2}}$.

The syntactic parallel of (12) is, rather, the SR (14), in which the indefinite noun phrase is translated into an existential quantification.

(14) $\forall x(\mathbf{farmer}(x) \wedge \exists y(\mathbf{donkey}(y) \wedge \mathbf{own}(x, y)) \rightarrow \mathbf{beat}(x, y))$

However, (14) does not represent the truth condition of (12) correctly since the variable y in $\mathbf{beat}(x, y)$ fails to be bound by $\exists$. Therefore, neither (13) nor (14) qualifies as the SR of (12).

Various approaches in discourse semantics

Dynamic Semantics

- Discourse Representation Theory (DRT): Kamp (1981), Kamp and Reyle (1993)
- Dynamic Predicate Logic (DPL): Groenendijk and Stokhof (1991)
- Dynamic Plural Predicate Logic (DPPL): van den Berg (1996), Sudo (2012)

Type-theoretical Semantics

- Analysis of donkey anaphora: Sundholm (1986))
- Type Theoretical Grammar (TTG): Ranta (1994)
- Type Theory with Record (TTR): Cooper (2005)
- MTT-semantics: Luo (1997, 1999, 2010, 2012), Asher and Luo (2012), Chatzikyriakidis (2014)
- Dependent Type Semantics (DTS): Bekki (2014), Bekki and Mineshima (2017), Bekki (2023)

Donkey anaphora: Sundholm (1986)

(12a) Every farmer who owns a donkey beats it .

$$\left(u : \begin{bmatrix} x : \text{entity} \\ \begin{bmatrix} \textbf{farmer}(x) \\ \begin{bmatrix} v : \begin{bmatrix} y : \text{entity} \\ \textbf{donkey}(y) \end{bmatrix} \\ \textbf{own}(x, \pi_1 v) \end{bmatrix} \end{bmatrix} \end{bmatrix} \right) \to \textbf{beat}(\pi_1 u, \pi_1\pi_1\pi_2\pi_2 u)$$

Note: $(x : A) \to B$ is a type for functions from A to $B[x]$.

From TTG to DTS: Compositionality

Q: How could one get to these (dependently-typed) representations from arbitrary sentences?

A: By lexicalization.

Q: But, how could we lexicalize context-dependent words like pronouns?

$$\left(u : \begin{bmatrix} x : \text{entity} \\ \begin{bmatrix} \textbf{farmer}(x) \\ \begin{bmatrix} v : \begin{bmatrix} y : \text{entity} \\ \textbf{donkey}(y) \end{bmatrix} \\ \textbf{own}(x, \pi_1 v) \end{bmatrix} \end{bmatrix} \end{bmatrix} \right) \to \textbf{beat}(\pi_1 u, \pi_1 \pi_1 \pi_2 \pi_2 u)$$

From TTG to DTS: Compositionality

Q: How could one get to these (dependently-typed) representations from arbitrary sentences?

A: By lexicalization.

Q: But, how can we lexicalize context-dependent words like pronouns?

A: By using **underspecified types**.

Q: How can we retrieve a context for an underspecified type?

A: By **type checking**.

Q: How can a pronoun find its antecedent?

A: By **proof construction**.

Dependent Type Semantics (DTS)

Underspecified types

UDTT = DTT + underspecified types

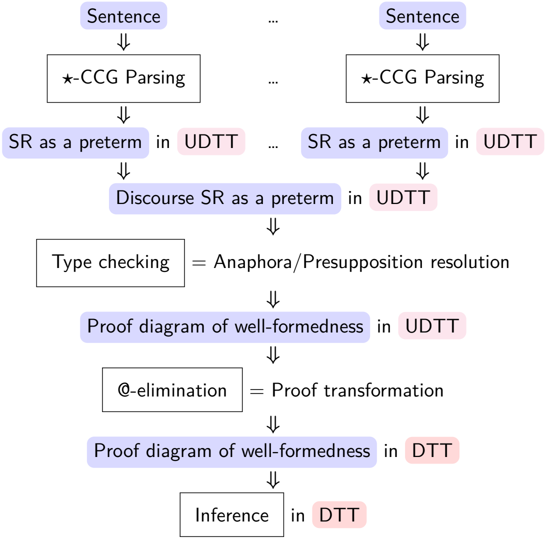
Definition (@-rule for underspecified types)

$$\frac{A : \mathsf{type} \quad M : A' \quad B[M/x] : \mathsf{type}}{\begin{bmatrix} x@A \\ B \end{bmatrix} : \mathsf{type}} (@)$$

- @-rule states that the well-formedness of $\begin{bmatrix} x@A \\ B \end{bmatrix}$ requires:
  - A is a well-formed type
  - the inhabitance of a proof (let it be M) of A, checking of which launches a proof search
  - $B[M/x]$ is a well-formed type

This means that the truth of A is *presupposed* in the underspecified type.

A theory of meaning via ⋆-CCG, UDTT, and DTT



Lexical items in $\star$-CCG

Surface form	syntactic type	Semantic representation
every	$NP^\star/N$	$\lambda n.\left(u:\begin{bmatrix} x:\text{entity}\\ n(x)\end{bmatrix}\right)\to \boxed{\pi_1 u}$
a, some	$NP^\star/N$	$\lambda n.\begin{bmatrix} u:\begin{bmatrix} x:\text{entity}\\ n(x)\end{bmatrix}\\ \boxed{\pi_1 u}\end{bmatrix}$
if	$\begin{cases}(S/S)^\star/S\\ (S\backslash S)^\star/S\end{cases}$	$\lambda p.\,(u:p)\to \boxed{id}$
who/whom	$\begin{cases}N\backslash N/(S\backslash NP)^\star\\ N\backslash N/(S/NP)^\star\end{cases}$	$\lambda p.\lambda n.\lambda x.\begin{bmatrix} nx\\ p(\lambda q.qx)\end{bmatrix}$
farmer	N	**farmer**
donkey	N	**donkey**
owns	$S\backslash NP/NP$	**own**
beats	$S\backslash NP/NP$	**beat**
he/him	$NP^\star$	$\begin{bmatrix} u@\begin{bmatrix} x:\text{entity}\\ \textbf{male}(x)\end{bmatrix}\\ \boxed{\pi_1 u}\end{bmatrix}$
it	$NP^\star$	$\begin{bmatrix} u@\begin{bmatrix} x:\text{entity}\\ \neg\textbf{human}(x)\end{bmatrix}\\ \boxed{\pi_1 u}\end{bmatrix}$
the	$NP^\star/N$	$\lambda n.\begin{bmatrix} u@\begin{bmatrix} x:\text{entity}\\ n(x)\end{bmatrix}\\ \boxed{\pi_1 u}\end{bmatrix}$

Donkey sentence: Syntactic Structure

$$
\dfrac{\dfrac{\dfrac{\text{Every}}{\boldsymbol{NP^\star/N}} \quad \dfrac{\dfrac{\text{farmer}}{\boldsymbol{N}} \quad \dfrac{\dfrac{\text{who}}{\boldsymbol{N\backslash N/(S\backslash NP)^\star}} \quad \dfrac{\dfrac{\dfrac{\text{owns}}{\boldsymbol{S\backslash NP/NP}}}{\boldsymbol{(S\backslash NP/NP)^\star}}\uparrow \quad \dfrac{\dfrac{\text{a}}{\boldsymbol{NP^\star/N}} \quad \dfrac{\text{donkey}}{\boldsymbol{N}}}{\boldsymbol{NP^\star}}>}{\boldsymbol{(S\backslash NP)^\star}}>^\star}{\boldsymbol{N\backslash N}}>}{\boldsymbol{N}}<}{\boldsymbol{NP^\star}}> \quad \dfrac{\dfrac{\dfrac{\text{beat}}{\boldsymbol{S\backslash NP/NP}}}{\boldsymbol{(S\backslash NP/NP)^\star}}\uparrow \quad \dfrac{\text{it}}{\boldsymbol{NP^\star}}}{\boldsymbol{(S\backslash NP)^\star}}>^\star}{\dfrac{\boldsymbol{S^\star}}{\boldsymbol{S}}\checkmark}<^\star
$$

Donkey sentence: Semantic Composition (1/2)

$$\dfrac{\dfrac{\text{a}}{\lambda n.\left[\begin{array}{l} v : \left[\begin{array}{l} y : \text{entity} \\ n(y) \end{array}\right] \\ \boxed{\pi_1 v} \end{array}\right]} \qquad \dfrac{\text{donkey}}{\textbf{donkey}}}{\left[\begin{array}{l} v : \left[\begin{array}{l} y : \text{entity} \\ \textbf{donkey}(y) \end{array}\right] \\ \boxed{\pi_1 v} \end{array}\right]}\,>^{\star}$$

$$\dfrac{\dfrac{\text{owns}}{\boxed{\begin{array}{l} \lambda y.\lambda x. \\ \textbf{own}_{e_1}(x,y) \end{array}}} \qquad \left[\begin{array}{l} v : \left[\begin{array}{l} y : \text{entity} \\ \textbf{donkey}(y) \end{array}\right] \\ \boxed{\pi_1 v} \end{array}\right]}{\left[\begin{array}{l} v : \left[\begin{array}{l} y : \text{entity} \\ \textbf{donkey}(y) \end{array}\right] \\ \boxed{\lambda x.\textbf{own}_{e_1}(x,\pi_1 v)} \end{array}\right]}\,>^{\star}$$

$$\dfrac{\dfrac{\text{who}}{\begin{array}{l} \lambda q.\lambda n.\lambda x. \\ \left[\begin{array}{l} n(x) \\ q(\lambda p.px) \end{array}\right] \end{array}} \qquad \left[\begin{array}{l} v : \left[\begin{array}{l} y : \text{entity} \\ \textbf{donkey}(y) \end{array}\right] \\ \boxed{\lambda x.\textbf{own}_{e_1}(x,\pi_1 v)} \end{array}\right]}{\lambda n.\lambda x.\left[\begin{array}{l} n(x) \\ v : \left[\begin{array}{l} y : \text{entity} \\ \textbf{donkey}(y) \end{array}\right] \\ \textbf{own}_{e_1}(x,\pi_1 v) \end{array}\right]}\,>$$

$$\dfrac{\dfrac{\text{farmer}}{\textbf{farmer}} \qquad \lambda n.\lambda x.\left[\begin{array}{l} n(x) \\ v : \left[\begin{array}{l} y : \text{entity} \\ \textbf{donkey}(y) \end{array}\right] \\ \textbf{own}_{e_1}(x,\pi_1 v) \end{array}\right]}{\lambda x.\left[\begin{array}{l} \textbf{farmer}(x) \\ v : \left[\begin{array}{l} y : \text{entity} \\ \textbf{donkey}(y) \end{array}\right] \\ \textbf{own}_{e_1}(x,\pi_1 v) \end{array}\right]}\,<$$

Donkey sentence: Semantic Composition (2/2)

$$
\dfrac{\dfrac{\text{Every}}{\left(u : \begin{bmatrix} x : \text{entity} \\ n(x) \end{bmatrix}\right) \to \boxed{\pi_1 u}} \qquad \dfrac{\text{farmer who owns a donkey}}{\lambda x. \begin{bmatrix} \textbf{farmer}(x) \\ v : \begin{bmatrix} y : \text{entity} \\ \textbf{donkey}(y) \end{bmatrix} \\ \textbf{own}_{e_1}(x, \pi_1 v) \end{bmatrix}}}{\left(u : \begin{bmatrix} x : \text{entity} \\ \textbf{farmer}(x) \\ v : \begin{bmatrix} y : \text{entity} \\ \textbf{donkey}(y) \end{bmatrix} \\ \textbf{own}_{e_1}(x, \pi_1 v) \end{bmatrix}\right) \to \boxed{\pi_1 u}} <^{\star}
$$

$$
\dfrac{\dfrac{\text{beat}}{\boxed{\begin{matrix} \lambda y.\lambda x. \\ \textbf{beat}_{e_2}(x, y) \end{matrix}}} \qquad \dfrac{\text{it}}{\begin{bmatrix} w @ \begin{bmatrix} z : \text{entity} \\ \neg\textbf{human}(z) \end{bmatrix} \\ \boxed{\pi_1 w} \end{bmatrix}}}{\lambda x. \begin{bmatrix} w @ \begin{bmatrix} z : \text{entity} \\ \neg\textbf{human}(z) \end{bmatrix} \\ \boxed{\lambda x.\textbf{beat}_{e_2}(x, \pi_1 w)} \end{bmatrix}} >^{\star}
$$

$$
\dfrac{\cdots \qquad \cdots}{\left(u : \begin{bmatrix} x : \text{entity} \\ \textbf{farmer}(x) \\ v : \begin{bmatrix} y : \text{entity} \\ \textbf{donkey}(y) \end{bmatrix} \\ \textbf{own}_{e_1}(x, \pi_1 v) \end{bmatrix}\right) \to \begin{bmatrix} w @ \begin{bmatrix} z : \text{entity} \\ \neg\textbf{human}(z) \end{bmatrix} \\ \boxed{\textbf{beat}_{e_2}(\pi_1 u, \pi_1 w)} \end{bmatrix}} <^{\star}
$$

$$
\dfrac{\cdots}{\left(u : \begin{bmatrix} x : \text{entity} \\ \textbf{farmer}(x) \\ v : \begin{bmatrix} y : \text{entity} \\ \textbf{donkey}(y) \end{bmatrix} \\ \textbf{own}_{e_1}(x, \pi_1 v) \end{bmatrix}\right) \to \begin{bmatrix} w @ \begin{bmatrix} z : \text{entity} \\ \neg\textbf{human}(z) \end{bmatrix} \\ \textbf{beat}_{e_2}(\pi_1 u, \pi_1 w) \end{bmatrix}} \checkmark
$$

Donkey sentence: Semantic Felicity Condition (= Type Checking)

$$
\dfrac{
\begin{array}{c}
\mathcal{D}_1\\
\begin{bmatrix}
x:\text{entity}\\
\textbf{farmer}(x)\\
v:\begin{bmatrix} y:\text{entity}\\ \textbf{donkey}(y)\end{bmatrix}\\
\textbf{own}_{e_1}(x,\pi_1 v)
\end{bmatrix}:\text{type}
\end{array}
\quad
\dfrac{
\begin{array}{c}
\mathcal{D}_2\\
\begin{bmatrix} y:\text{entity}\\ \neg\textbf{human}(y)\end{bmatrix}:\text{type}
\end{array}
\quad
\begin{array}{c}
\overline{u:\begin{bmatrix}
x:\text{entity}\\
\textbf{farmer}(x)\\
v:\begin{bmatrix} y:\text{entity}\\ \textbf{donkey}(y)\end{bmatrix}\\
\textbf{own}_{e_1}(x,\pi_1 v)
\end{bmatrix}}^{\,1}\\
\mathcal{D}_3\\
?:\begin{bmatrix} y:\text{entity}\\ \neg\textbf{human}(y)\end{bmatrix}
\end{array}
\quad
\mathcal{D}_4
}{
\begin{bmatrix}
v@\begin{bmatrix} y:\text{entity}\\ \neg\textbf{human}(y)\end{bmatrix}\\
\textbf{beat}_{e_2}(\pi_1 u,\pi_1 v)
\end{bmatrix}:\text{type}
}(@)
}{
\left(u:\begin{bmatrix}
x:\text{entity}\\
\textbf{farmer}(x)\\
v:\begin{bmatrix} y:\text{entity}\\ \textbf{donkey}(y)\end{bmatrix}\\
\textbf{own}_{e_1}(x,\pi_1 v)
\end{bmatrix}\right)\to
\begin{bmatrix}
v@\begin{bmatrix} y:\text{entity}\\ \neg\textbf{human}(y)\end{bmatrix}\\
\textbf{beat}_{e_2}(\pi_1 u,\pi_1 v)
\end{bmatrix}:\text{type}
}(\Pi F),1
$$

Donkey sentence: Anaphora Resolusion (= Proof Search)

$$
\dfrac{
\dfrac{\dfrac{\dfrac{\dfrac{u : \begin{bmatrix} x : \mathsf{entity} \\ \mathsf{farmer}(x) \\ v : \begin{bmatrix} y : \mathsf{entity} \\ \mathsf{donkey}(y) \end{bmatrix} \\ \mathsf{own}(x, \pi_1 v) \end{bmatrix}}{\pi_2 u : \begin{bmatrix} \mathsf{farmer}(x) \\ v : \begin{bmatrix} y : \mathsf{entity} \\ \mathsf{donkey}(y) \end{bmatrix} \\ \mathsf{own}(\pi_1 u, \pi_1 v) \end{bmatrix}}(\Sigma E)}{\pi_2\pi_2 u : \begin{bmatrix} v : \begin{bmatrix} y : \mathsf{entity} \\ \mathsf{donkey}(y) \end{bmatrix} \\ \mathsf{own}(\pi_1 u, \pi_1 v) \end{bmatrix}}(\Sigma E)}{\pi_1\pi_2\pi_2 u : \begin{bmatrix} y : \mathsf{entity} \\ \mathsf{donkey}(y) \end{bmatrix}}(\Sigma E)}{\pi_1\pi_1\pi_2\pi_2 u : \mathsf{entity}}(\Sigma E)
\quad
\dfrac{
\dfrac{\dfrac{\dfrac{u : \begin{bmatrix} x : \mathsf{entity} \\ \mathsf{farmer}(x) \\ v : \begin{bmatrix} y : \mathsf{entity} \\ \mathsf{donkey}(y) \end{bmatrix} \\ \mathsf{own}(x, \pi_1 v) \end{bmatrix}}{\pi_2 u : \begin{bmatrix} \mathsf{farmer}(x) \\ v : \begin{bmatrix} y : \mathsf{entity} \\ \mathsf{donkey}(y) \end{bmatrix} \\ \mathsf{own}(\pi_1 u, \pi_1 v) \end{bmatrix}}(\Sigma E)}{\pi_2\pi_2 u : \begin{bmatrix} v : \begin{bmatrix} y : \mathsf{entity} \\ \mathsf{donkey}(y) \end{bmatrix} \\ \mathsf{own}(\pi_1 u, \pi_1 v) \end{bmatrix}}(\Sigma E)}{\pi_1\pi_2\pi_2 u : \begin{bmatrix} y : \mathsf{entity} \\ \mathsf{donkey}(y) \end{bmatrix}}(\Sigma E)
\quad
\dfrac{}{d : \left(u : \begin{bmatrix} x : \mathsf{entity} \\ \mathsf{donkey}(x) \end{bmatrix} \right) \to \neg\mathsf{human}(\pi_1 u)}(CON)
}{d(\pi_1\pi_2\pi_2 u) : \neg\mathsf{human}(\pi_1\pi_1\pi_2\pi_2 u)}(\Pi E)
}{(\pi_1\pi_1\pi_2\pi_2 u, d(\pi_1\pi_2\pi_2 u)) : \begin{bmatrix} z : \mathsf{entity} \\ \neg\mathsf{human}(z) \end{bmatrix}}(\Sigma I)
$$

Donkey sentence: After @-elimination

$$
\dfrac{
\begin{array}{c}\mathcal{D}_1\\ \begin{bmatrix} x : \text{entity} \\ \textbf{farmer}(x) \\ v : \begin{bmatrix} y : \text{entity} \\ \textbf{donkey}(y) \end{bmatrix} \\ \textbf{own}_{e_1}(x, \pi_1 v) \end{bmatrix} : \text{type}\end{array}
\quad
\begin{array}{l}
\dfrac{}{u : \begin{bmatrix} x : \text{entity} \\ \textbf{farmer}(x) \\ v : \begin{bmatrix} y : \text{entity} \\ \textbf{donkey}(y) \end{bmatrix} \\ \textbf{own}_{e_1}(x, \pi_1 v) \end{bmatrix}}{1} \\
\qquad\qquad\qquad\mathcal{D}_4 \\
\equiv \textbf{beat}_{e_2}(\pi_1 u, \pi_1((\pi_1\pi_1\pi_2\pi_2 u, d(\pi_1\pi_2\pi_2 u)))) : \text{type} \\
\rightarrow_\beta \textbf{beat}_{e_2}(\pi_1 u, \pi_1\pi_1\pi_2\pi_2 u) : \text{type}
\end{array}
}{
\left(u : \begin{bmatrix} x : \text{entity} \\ \textbf{farmer}(x) \\ v : \begin{bmatrix} y : \text{entity} \\ \textbf{donkey}(y) \end{bmatrix} \\ \textbf{own}_{e_1}(x, \pi_1 v) \end{bmatrix} \right) \rightarrow \textbf{beat}_{e_2}(\pi_1 u, \pi_1\pi_1\pi_2\pi_2 u) : \text{type}
}(\Pi F),1
$$

A takeaway message from DTS on Donkey Anaphora

The SR of a pronoun can refer to a donkey in question even from outside of its existential scope, because the donkey can be *constructed from* the variable in the context for the proof search.

In other words, anaphora accessibility is *deducibility*.

Notes on donkey anaphora

- The same analysis applies to the *implicational* donkey sentence:

  (15) If [a farmer]$^{\mathbf{1}}$ owns [a donkey]$^{\mathbf{2}}$, he$_{\mathbf{1}}$ beats it$_{\mathbf{2}}$.

- This analysis only predicts the *strong reading* for donkey sentences. For deriving both the strong readings and the *weak readings*, we need to refine the semantic representations for quantificational expressions: See Tanaka (2021).
- The refined analysis also explains why the anaphoric link to the *parametrized sum individual* (Krifka, 1996) is allowed (Tanaka, 2021).

  (16) [Every farmer]$^{\mathbf{1}}$ who owns [a donkey]$^{\mathbf{2}}$ loves its$_{\mathbf{2}}$ tail. But they$_{\mathbf{1}}$ beat it$_{\mathbf{2}}$.

Anaphoric Potential

Anaphoric Potential and Intuitionism

A central insight of dynamic semantics is that the meaning of a sentence is not exhausted by its truth conditions alone. Rather, a sentence's meaning also includes its capacity to affect the subsequent discourse, a property known as its anaphoric potential (cf. Kamp et al. (2011)). To illustrate this, consider the following two sentences:

(17) a. Some company is not rich.

b. It is not the case that every company is rich.

The semantic representation in FoL of the sentences (17a) and (17b) are (in logic textbooks) (18a) and (18b) respectively.

(18) a. $\exists x(\mathbf{company}(x) \wedge \neg \mathbf{rich}(x))$

b. $\neg \forall x(\mathbf{company}(x) \rightarrow \mathbf{rich}(x))$

Anaphoric Potential and Intuitionism

- In a classical framework, these two sentences are semantically and proof-theoretically equivalent, thus are assigned identical truth conditions.
- From this perspective, they have the *same* meaning.
- However, a closer examination reveals a crucial difference in their linguistic behavior.

Anaphoric Potential and Intuitionism

The two sentences (17a) and (17b) exhibit distinct *anaphoric potentials*, as demonstrated by their ability to license a subsequent pronoun. Consider the following continuation of each sentence:

(19) a. [Some company]^{1} is not rich.
b. It$_{1}$ does not have enough money to replace the system.

(20) a. It is not the case that [every company]^{1} is rich.
b. #It$_{1}$ does not have enough money to replace the system.

The inability of sentence (17b) to provide an antecedent for the pronoun *it* highlights that while it is truth-conditionally equivalent to sentence (17a), it fails to introduce a discourse referent that can be subsequently accessed.

Anaphoric Potential and Intuitionism

- This observation demonstrates that the semantic information conveyed by a sentence is not exhausted by its truth conditions.
- The anaphoric potential of a sentence is a fundamental component of its meaning.
- Consequently, a comprehensive theory of meaning must move beyond a purely truth-conditional approach to account for this dynamic aspect of language.

Anaphoric Potential and Intuitionism

In DTS, the two sentences (17a) and (17b) receive non-equivalent semantic representations, presented as follows, which directly encode their different anaphoric potentials.

(21) a. $$\left[\begin{array}{l} f : \left[\begin{array}{l} u : \left[\begin{array}{l} x : \text{entity} \\ \textbf{company}(x) \end{array}\right] \\ \neg\textbf{rich}_e(\pi_1 u) \end{array}\right] \\ v@\left[\begin{array}{l} y : \text{entity} \\ \neg\textbf{human}(y) \end{array}\right] \\ \neg\textbf{haveMoney}_e(\pi_1 v) \end{array}\right]$$

b. $$\left[\begin{array}{l} f : \neg\left(\left(u : \left[\begin{array}{l} x : \text{entity} \\ \textbf{company} \end{array}\right]\right) \to \textbf{rich}_e(\pi_1 u)\right) \\ v@\left[\begin{array}{l} y : \text{entity} \\ \neg\textbf{human}(y) \end{array}\right] \\ \neg\textbf{haveMoney}_e(\pi_1 v) \end{array}\right]$$

Anaphoric Potential and Intuitionism

(21a) $$\left[\begin{array}{l} \boldsymbol{f} : \boxed{\left[\begin{array}{l} \boldsymbol{u} : \left[\begin{array}{l} \boldsymbol{x} : \text{entity} \\ \textbf{company}(\boldsymbol{x}) \end{array}\right] \\ \neg\textbf{rich}_e(\boldsymbol{\pi_1 u}) \end{array}\right]} \\ \boldsymbol{v}@\left[\begin{array}{l} \boldsymbol{y} : \text{entity} \\ \neg\textbf{human}(\boldsymbol{y}) \end{array}\right] \\ \neg\textbf{haveMoney}_e(\boldsymbol{\pi_1 v}) \end{array}\right]$$

In (21a), the antecedent for a subsequent anaphoric pronoun can be successfully resolved. For a pronoun like *it*, its reference is fixed as $\boldsymbol{\pi_1 v = \pi_1\pi_1(f)}$.

Anaphoric Potential and Intuitionism

(21b) $$\left[\begin{array}{l} f : \neg\left(\left(u : \left[\begin{array}{l} x : \text{entity} \\ \textbf{company} \end{array}\right]\right) \to \textbf{rich}_e(\pi_1 u)\right) \\ v @ \left[\begin{array}{l} y : \text{entity} \\ \neg\textbf{human}(y) \end{array}\right] \\ \neg\textbf{haveMoney}_e(\pi_1 v) \end{array}\right]$$

Conversely, in (21b), the first sentence fails to introduce an antecedent. This is due to the inherent constraints of the system, particularly those related to quantifiers and negation.

Anaphoric Potential and Intuitionism

However, the explanatory power of DTS is fundamentally contingent upon its basis in intuitionistic type theory. This is a crucial point, as the introduction of a classical rule such as Double Negative Elimination (DNE) would lead to an incorrect overgeneration of anaphoric links.

$$\frac{M : \neg\neg A}{\mathsf{dne}(M) : A}\,(DNE)$$

If the following inference had a proof, then the anaphoric link in question would be wrongly predicted to be licenced.

(22) $f : \neg\,(x : \mathsf{entity}) \to \mathbf{L}(x) \to \neg\mathbf{R}(x)$

$$\vdash \begin{bmatrix} x : \mathsf{entity} \\ \begin{bmatrix} \mathbf{L}(x) \\ \mathbf{R}(x) \end{bmatrix} \end{bmatrix} \; true$$

Anaphoric Potential and Intuitionism

(23) Proof diagram of (20)

$$
\dfrac{
  f : \neg\left(\left(u : \begin{bmatrix} x : \text{entity} \\ \textbf{company} \end{bmatrix}\right) \to \textbf{rich}_e(\pi_1 u)\right)
  \qquad
  \dfrac{
    \dfrac{
      \dfrac{
        \dfrac{
          \dfrac{
            \dfrac{}{u : \begin{bmatrix} x : \text{entity} \\ \textbf{company}(x) \end{bmatrix}}{}^{2}
            \quad
            \dfrac{}{r : \neg\textbf{rich}_e(\pi_1 u)}{}^{1}
          }{
            (u, r) : \begin{bmatrix} u : \begin{bmatrix} x : \text{entity} \\ \textbf{company}(x) \end{bmatrix} \\ \neg\textbf{rich}_e(\pi_1 u) \end{bmatrix}
          }(\Sigma I)
          \quad
          \dfrac{}{s : \neg\begin{bmatrix} u : \begin{bmatrix} x : \text{entity} \\ \textbf{company}(x) \end{bmatrix} \\ \neg\textbf{rich}_e(\pi_1 u) \end{bmatrix}}{}^{3}
        }{
          s(u, r) : \bot
        }(\to E)
      }{
        \lambda r.s(u, r) : \neg\neg\textbf{rich}_e(\pi_1 u)
      }(\neg I),1
    }{
      \text{dne}(\lambda r.s(u, r)) : \textbf{rich}_e(\pi_1 u)
    }(DNE)
  }{
    \lambda u.\text{dne}(\lambda r.s(u, r)) : \left(u : \begin{bmatrix} x : \text{entity} \\ \textbf{company}(x) \end{bmatrix}\right) \to \textbf{rich}_e(\pi_1 u)
  }(\to I),2
}{
  \dfrac{
    \dfrac{
      f(\lambda u.\text{dne}(\lambda r.s(u, r))) : \bot
    }{
      \lambda s.f(\lambda u.\text{dne}(\lambda r.s(u, r))) : \neg\neg\begin{bmatrix} u : \begin{bmatrix} x : \text{entity} \\ \textbf{company}(x) \end{bmatrix} \\ \neg\textbf{rich}_e(\pi_1 u) \end{bmatrix}
    }(\neg I),3
  }{
    \text{dne}(\lambda s.f(\lambda u.\text{dne}(\lambda r.s(u, r)))) : \begin{bmatrix} u : \begin{bmatrix} x : \text{entity} \\ \textbf{company}(x) \end{bmatrix} \\ \neg\textbf{rich}_e(\pi_1 u) \end{bmatrix}
  }(DNE)
}(\to E)
$$

Anaphoric Potential and Intuitionism

- Let $\boldsymbol{F(f)}$ be $\text{dne}(\boldsymbol{\lambda s.f(\lambda u.}\text{dne}(\boldsymbol{\lambda r.s(u,r)})))$. Then, the variable $\boldsymbol{v}$ in the underspecified type would be resolved as $(\boldsymbol{\pi_1\pi_1(F(f)),} \textbf{CnH}(\boldsymbol{\pi_1 F(f)}))$.
- This demonstrates that adding (DNE) makes DTS a system of classical logic, thereby causing it to lose its ability to distinguish the different anaphoric potentials.
- Therefore, the capacity of DTS to explain the observed difference in anaphoric potential is inextricably linked to its intuitionistic foundation.

Summary and History

A Unified, Compositional Theory of *Projective* Meaning

- DTS provides a unified analysis for (general) inferences and anaphora resolution mechanisms.
- The background theory for DTS is UDTT (an extention of DTT with underspecified types).
  - Lexical items of anaphoric expressions and presupposition triggers are represented by using underspecified types.
  - Context retrieval in DTS reduces to type checking.
  - Anaphora resolution (and presupposition binding) in DTS reduces to proof search. Consequently, anaphora accessibility reduces to proof constructability.
  - @-elimination transforms a proof diagram of UDTT into a proof diagram of DTT, by which an SR in DTT is obtained with all anaphora/presuppositions resolved.

Natural language semantics via dependent types: The first generation

- Donkey anaphora: Sundholm (1986)
- Translation from DRS to dependent type representations: Ahn and Kolb (1990)
- Summation: Fox (1994a,b)
- Ranta's TTG (Relative and Implicational Donkey Sentences, Branching Quantifiers, Intensionality, Tense): Ranta (1994)
- Translation from Montague Grammar to dependent type representations: Dávila-Pérez (1995)
- Presupposition Binding and Accommodation, Bridging: Krahmer and Piwek (1999), Piwek and Krahmer (2000)

Natural language semantics via dependent types: The second generation

- Type Theory with Record (TTR): Cooper (2005)
- MTT-Semantics: Luo (1997, 1999, 2010, 2012), Asher and Luo (2012), Chatzikyriakidis (2014)
- Semantics with Dependent Types: Grudzinska and Zawadowski (2014; 2017)
- Dynamic Categorial Grammar: Martin and Pollard (2014)
- **Dependent Type Semantics (DTS): Bekki (2014), Bekki and Mineshima (2017), Bekki (2023)**

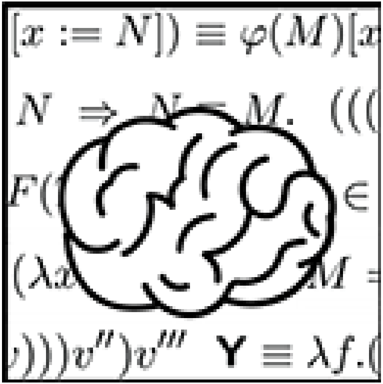
Semantic Analyses by DTS: The third generation

- Generalized Quantifiers: Tanaka (2014)
- Honorification: Watanabe et al. (2014)
- Conventional Implicature: Bekki and McCready (2015), Matsuoka et al. (2023)
- Factive Presuppositions: Tanaka et al. (2015)
- Dependent Plural Anaphora: Tanaka et al. (2017)
- Paycheck sentences: Tanaka et al. (2018)
- Coercion and Metaphor: Kinoshita et al. (2017, 2018)
- Questions: Watanabe et al. (2019), Funakura (2022)
- Comparision with DRT: Yana et al. (2019)
- The proviso problem: Yana et al. (2021)
- Weak Crossover: Bekki (2023), Fukushima et al. (2023)

Computational Aspects of DTS

- Type Checker for (the fragment of) DTS: Bekki and Sato (2015)
- Development of an automated theorem prover (for the fragment of) DTT: Daido and Bekki (2020)
- Integrating Deep Neural Network with DTS: Bekki et al. (2023, 2022)

Thank you!



Reference I

Ahn, R. and H.-P. Kolb. (1990) Discourse Representation meets Constructive Mathematics. Akademiai Kiado.

Asher, N. and Z. Luo. (2012) "Formalisation of coercions in lexical semantics", In the Proceedings of *Sinn und Bedeutung 17*. pp.63–80.

Bekki, D. (2014) "Representing Anaphora with Dependent Types", In the Proceedings of N. Asher and S. V. Soloviev (eds.): *Logical Aspects of Computational Linguistics (8th international conference, LACL2014, Toulouse, France, June 2014 Proceedings), LNCS 8535*. pp.14–29, Springer, Heiderburg.

Bekki, D. (2023) A Proof-theoretic Analysis of Weak Crossover, pp.228–241. Springer.

Reference II

Bekki, D. and E. McCready. (2015) CI via DTS, Vol. LNAI 9067. Springer.

Bekki, D. and K. Mineshima. (2017) Context-Passing and Underspecification in Dependent Type Semantics, pp.11–41, Studies of Linguistics and Philosophy. Springer.

Bekki, D. and M. Sato. (2015) "Calculating Projections via Type Checking", In the Proceedings of *TYpe Theory and LExical Semantics (TYTLES), ESSLLI2015 workshop*.

Bekki, D., R. Tanaka, and Y. Takahashi. (2022) "Learning Knowledge with Neural DTS", In the Proceedings of *the 3rd Natural Logic Meets Machine Learning (NALOMA III)*. pp.17–25, Association of Computational Linguistics.

Reference III

Bekki, D., R. Tanaka, and Y. Takahashi. (2023) Integrating Deep Neural Network with Dependent Type Semantics, Studies in Computational Intelligence 1081. Springer.

Chatzikyriakidis, S. (2014) Adverbs in a Modern Type Theory. Springer.

Cooper, R. (2005) "Records and Record Types in Semantic Theory", *Journal of Logic and Computation* **15**(2), pp.99–112.

Daido, H. and D. Bekki. (2020) "Development of an automated theorem prover for the fragment of DTS", In the Proceedings of *the 17th International Workshop on Logic and Engineering of Natural Language Semantics (LENLS17)*.

Dávila-Pérez, R. (1995) "Semantics and Parsing in Intuitionistic Categorial Grammar", Thesis. Ph.D. thesis.

Reference IV

Fox, C. (1994a) "Discourse Representation, Type Theory and Property Theory", In the Proceedings of H. Bunt, R. Muskens, and G. Rentier (eds.): *the International Workshop on Computational Semantics*. pp.71–80.

Fox, C. (1994b) "Existence Presuppositions and Category Mistakes", *Acta Linguistica Hungarica* **42**(3/4), pp.325–339. Published 1996.

Fukushima, H., D. Plesniak, and D. Bekki. (2023) "Matrix and Relative Weak Crossover on the Level of the Individual: A Proposed Experimental Investigation", In the Proceedings of *the 20th International Workshop on Logic and Engineering of Natural Language Semantics (LENLS20)*. pp.93–98.

Reference V

Funakura, H. (2022) “Answers, Exhaustivity, and Presupposition of wh-questions in Dependent Type Semantics”, In the Proceedings of *Logic and Engineering of Natural Language Semantics 20 (LENLS20)*. pp.72–76.

Geach, P. (1962) *Reference and Generality: An Examination of Some Medieval and Modern Theories*. Ithaca, New York, Cornell University Press.

Groenendijk, J. and M. Stokhof. (1991) “Dynamic Predicate Logic”, *Linguistics and Philosophy* **14**, pp.39–100.

Kamp, H. (1981) A Theory of Truth and Semantic Representation. Amsterdam, Mathematical Centre Tract 135.

Kamp, H. and U. Reyle. (1993) *From Discourse to Logic*. Kluwer Academic Publishers.

Reference VI

Kamp, H., J. van Genabith, and U. Reyle. (2011) Discourse Representation Theory, Vol. 15, pp.125–394. Doredrecht, Springer.

Kinoshita, E., K. Mineshima, and D. Bekki. (2017) An Analysis of Selectional Restrictions with Dependent Type Semantics, pp.19–32. Springer.

Kinoshita, E., K. Mineshima, and D. Bekki. (2018) Coercion as Proof Search in Dependent Type Semantics, pp.1–20.

Krahmer, E. and P. Piwek. (1999) Presupposition Projection as Proof Construction, Studies in Linguistics Philosophy Series. Dordrecht, Kluwer Academic Publishers.

Krifka, M. (1996) "Parametrized Sum Individuals for Plural Anaphora", *Linguistics and Philosophy* **19**, pp.555–598.

Reference VII

Luo, Z. (1997) Coercive subtyping in type theory. Heidelberg, Springer.

Luo, Z. (1999) "Coercive subtyping", *Journal of Logic and Computation* **9**(1), pp.105–130.

Luo, Z. (2010) "Type-theoretical semantics with coercive subtyping", In the Proceedings of *Semantics and Linguistic Theory 20 (SALT 20)*.

Luo, Z. (2012) "Formal Semantics in Modern Type Theories with Coercive Subtyping", *Linguistics and Philosophy* **35**(6).

Martin, S. and C. J. Pollard. (2014) "A dynamic categorial grammar", In the Proceedings of *Formal Grammar 19, LNCS 8612*.

Reference VIII

Matsuoka, D., D. Bekki, and H. Yanaka. (2023) "Appositive Projection as Implicit Context Extension in Dependent Type Semantics", In the Proceedings of *the 20th International Workshop on Logic and Engineering of Natural Language Semantics (LENLS20)*. pp.82–87.

Piwek, P. and E. Krahmer. (2000) Presuppositions in Context: Constructing Bridges, Applied Logic Series. Dordrecht, Kluwer Academic Publishers.

Ranta, A. (1994) *Type-Theoretical Grammar*. Oxford University Press.

Sudo, Y. (2012) "On the Semantics of Phi Features on Pronouns", Thesis. Doctoral dissertation.

Reference IX

Sundholm, G. (1986) Proof theory and meaning, Vol. III, pp.471–506. Reidel, Kluwer.

Tanaka, R. (2014) "A Proof-Theoretic Approach to Generalized Quantifiers in Dependent Type Semantics", In the Proceedings of R. de Haan (ed.): *the ESSLLI 2014 Student Session, 26th European Summer School in Logic, Language and Information*. pp.140–151.

Tanaka, R. (2021) "Natural Language Quantification and Dependent Types", Thesis. Doctoral Dissertation.

Tanaka, R., K. Mineshima, and D. Bekki. (2015) "Factivity and Presupposition in Dependent Type Semantics", In the Proceedings of *TYpe Theory and LExical Semantics (TYTLES), ESSLLI2015 workshop*.

Reference X

Tanaka, R., K. Mineshima, and D. Bekki. (2017) On the Interpretation of Dependent Plural Anaphora in a Dependently-Typed Setting, pp.123–137. Springer.

Tanaka, R., K. Mineshima, and D. Bekki. (2018) "Paychecks, presupposition, and dependent types", In the Proceedings of *the Fifth Workshop on Natural Language and Computer Science (NLCS2018), Preprint no.215*. Oxford University.

van den Berg, M. (1996) "Some aspects of the internal structure of discourse – the dynamics of nominal anaphora –", Thesis. thesis.

Watanabe, K., K. Mineshima, and D. Bekki. (2019) "Questions in Dependent Type Semantics", In the Proceedings of *Proceedings of the Sixth Workshop on Natural Language and Computer Science (NLCS'19)*. pp.23–33.

Reference XI

Watanabe, N., E. McCready, and D. Bekki. (2014) "Japanese Honorification: Compositionality and Expressivity", In the Proceedings of S. Kawahara and M. Igarashi (eds.): *FAJL 7: Formal Approaches to Japanese Linguistics, the MIT Working Papers in Linguistics 73*. pp.265–276.

Yana, Y., D. Bekki, and K. Mineshima. (2019) "Variable Handling and Compositionality: Comparing DRT and DTS", *Journal of Logic, Language and Information* **28**(2), pp.261–285.

Yana, Y., K. Mineshima, and D. Bekki. (2021) "The proviso problem from a proof-theoretic perspective", In the Proceedings of *Logical Aspects of Computational Linguistics (LACL) 2021*. pp.159–176.