Chapter 26

Presuppositions

(*Joint work with Koji Mineshima*)

When considering the meaning of a sentence, we can often distinguish between its foreground and background contents. The sentence in (682) serves as a clear illustration.

(682) It is John who broke my iPhone.

The background content of (682) is *someone broke my iPhone*, which the speaker assumes is mutually accepted as true by the hearer. The foreground content, or assertion, is *John was the one who did it*, which conveys the primary communicative intent.

The background content poses a unique challenge in semantic theory. Unlike entailments, the background information is inferred from both the positive and negative forms of this sentence, as demonstrated in (683) and (684).

(683) It is John who broke my iPhone. $\Longrightarrow$ Someone broke my iPhone.

(684) It is not John who broke my iPhone. $\Longrightarrow$ Someone broke my iPhone.

The inference relation $\Longrightarrow$ presented here cannot be analyzed as *entailment*: for it were, the two inferences would take the following form.

(685) $A \vdash B, \ \neg A \vdash B$

In a classical setting where $\vdash A + \neg A$ holds, this inference leads to an asurd conclusion, since $\vdash B$ is derivable from (685), namely, B is true under any context. This means that *someone broke my iPhone* is a logical theorem, which is clearly not the case. Therefore, the inference relation $\Longrightarrow$ observed in (683) and (684) must be a distinct mode of inference from entailment, which we call *presupposition*.

When discussing this phenomenon, we can describe the background content as *projecting out of*, or *surviving in* the negation.

26.1 Presupposition Projection

The background contents not only project out of negation, but also out of other embedded contexts. (686) is inferred from every sentence in (687).

(686) Someone broke my iPhone.

(687) a. It wasn't John who broke my iPhone. **negation**
b. Maybe it was John who broke my iPhone. **modal**
c. If it was John who broke my iPhone, then he has to pay for it. **the antecedent of a conditional**
d. Was it John who broke my iPhone? **question**
e. Suppose that it was John who broke my iPhone. **hypothetical assumption**

These examples stand in stark contrast to the case of entailment as exemplified below.

(688) John is an American pianist.

(689) John is American.

The sentence (688) entails (689), but this entailment does not survive in the environments (690b,c), in contrast to the case of presupposition.

(690) a. John is not an American pianist. **negation**
b. Maybe John is an American pianist. **modal**
c. If John is an American pianist, he is skillful. **the antecedent of a conditional**
d. Is John an American pianist? **question**
e. Suppose that John is an American pianist. **hypothetical assumption**

Standard (FoL) semantics correctly predicts these patterns:

(688) John is an American pianist.
$\mathbf{american}(\mathbf{john}) \wedge \mathbf{pianist}(\mathbf{john})$

(690) a. John is not an American pianist.
$\neg(\mathbf{american}(\mathbf{john}) \wedge \mathbf{pianist}(\mathbf{john}))$
b. Maybe John is an American pianist.
$\Diamond(\mathbf{american}(\mathbf{john}) \wedge \mathbf{pianist}(\mathbf{john}))$
c. If John is an American pianist, he is skillful. $(\mathbf{american}(\mathbf{john}) \wedge \mathbf{pianist}(\mathbf{john})) \rightarrow \mathbf{skillful}(\mathbf{john})$

Therefore, we obtain:

$$(688) \vdash \mathbf{american}(\mathbf{john})$$
$$(690a) \nvdash \mathbf{american}(\mathbf{john})$$
$$(690b) \nvdash \mathbf{american}(\mathbf{john})$$
$$(690c) \nvdash \mathbf{american}(\mathbf{john})$$

The presupposition shows a different behaviour.

(682) It was John who broke my iPhone. SR_1

(687) a. It wasn't John who broke my iPhone. $\neg SR_1$

b. Maybe it was John who broke my iPhone. $\Diamond SR_1$

c. If it was John who broke my iPhone, he has to fix it. $SR_1 \rightarrow \cdots$

Now the question is what SR accounts for the following inference patterns.

$$\begin{aligned}
SR_1 &\Vdash \exists x(\mathbf{broke}(x, \mathbf{myiPhone})) \\
\neg SR_1 &\Vdash \exists x(\mathbf{broke}(x, \mathbf{myiPhone})) \\
\Diamond SR_1 &\Vdash \exists x(\mathbf{broke}(x, \mathbf{myiPhone})) \\
SR_1 \rightarrow A &\Vdash \exists x(\mathbf{broke}(x, \mathbf{myiPhone}))
\end{aligned}$$

?

26.2 Presupposition Filteration

The background content presents an additional puzzle: the phenomenon of *presupposition filteration* (or *presupposition filtering*). Consider, for example, the sentence (691), which presupposes that someone broke the window.

(691) It was John who broke the window. $\Longrightarrow$ Someone broke the window

When the first conjunct of a conjunction entails the presupposed content, the presupposition is filtered, meaning it is not inherited by the entire conjunction.

(692) The window was broken, and it was John who broke it.
$\not\Longrightarrow$ Someone broke the window.

The same property is observed in implication where the premise entails the presupposed content.

(693) If the window was broken, it was John who broke it.
$\not\Longrightarrow$ Someone broke the window.

In (695) and (696) below, filtering is at play as well.

(694) The king of France is wise.
$\Longrightarrow$ France has a king.

(695) France has a king, and the king of France is wise.
$\not\Longrightarrow$ France has a king.

revise .

(696) If France has a king, the king of France is wise.
$\not\Longrightarrow$ France has a king.

The examples presented here raise the question of how to account for the observation that a straightforward sentence such as (694) presupposes the existence of a king of France, whereas sentences (695) and (696) do not?

26.3 Presupposition triggers

Beyond clefts and the definite description, other constructions also *tringgers* presuppositions. Below is a conprehensive, though not exhaustive, list of these triggers.

(697) a. **The elevator** in this building is clean. **Description**
b. There is an elevator in this building.

(698) a. **John's sister** is happy. **Possessive**
b. John has a sister.

(699) a. Bill **regrets** that he lied to Mary. **Factive**
b. Bill lied to Mary.

(700) a. John has **stopped** beating his wife. **Aspectual**
b. John has beaten his wife.

(701) a. Harry **managed** to find the book. **Implicative**
b. Finding the book required some effort.

(702) a. Sam broke the window **again** today. **Iterative**
b. Sam broke the window before.

(703) a. **It was** Sam **who** broke the window. **Cleft**
b. Someone broke the window.

(704) a. **What** John broke **was** his typewriter. **Pseudo-cleft**
b. John broke something.

(705) a. $[\text{Pat}]_F$ is leaving, **too**. (Focus on *Pat*) **Additive**
b. Someone other than Pat is leaving.

For classical examples of presupposition triggers, see Levinson (1983), Soames (1989), Geurts (1999), and Beaver (2001), among others.

26.4 "Presupposition Is Anaphora" hypothesis

There are striking parallels between anaphoric expressions and presupposition triggers (van der Sandt, 1992; Geurts, 1999; Kripke, 2009).

Presupposition filtering

(706) a. John has children and **John's children** are wise.
b. If John has children, **John's children** are wise.

(707) a. The window was broken and **it was** John **who** broke it.
b. If the window was broken, **it was** John **who** broke it.

Compare (706) and (707) with the paragraph examples of anaphora.

Anaphora binding

(708) a. John owns a donkey and he beats **it**.
b. If John owns a donkey, he beats **it**.

26.5 DTS on Presupposition

26.5.1 On Filtering

The present account can explain the filtering of presupposition without further stipulation.

(709) If France has a king, **the king of France** is wise.

To begin with, the SR of the sentence (709) is compositionally obtained via the following derivation. We assume that the lexical entry for the determiner *the* is specified as follows.

(710) $$[\![\text{the} \vdash NP^\star/N]\!] \stackrel{def}{\equiv} \lambda n.\begin{bmatrix} u@\begin{bmatrix} x:\mathsf{e} \\ n(x) \end{bmatrix} \\ \boxed{\pi_1 u} \end{bmatrix}$$

By using the above, the syntactic structure and semantic composition of (709) are derived as follows.

(711) Syntactic structure of (709)

$$\dfrac{\dfrac{\dfrac{\text{If}}{(S/S)^\star/S}\quad\dfrac{\text{France has a king}}{S}}{(S/S)^\star}{>}\qquad \dfrac{\dfrac{\dfrac{\text{the}}{NP^\star/N}\quad\dfrac{\dfrac{\text{king}}{N/PP_{of}}\quad\dfrac{\dfrac{\text{of}}{PP_{of}/NP}\quad\dfrac{\text{France}}{NP}}{PP_{of}}{>}}{N}{>}}{NP^\star}{>}\quad\dfrac{\text{is wise}}{(S\backslash NP)^\star}}{S^\star}{<^\star}}{S^\star}{>^\star}$$

(712) Semantic composition of (709)

$$\dfrac{\dfrac{\dfrac{\text{If}}{\lambda p.(u:p)\to\boxed{id}}\quad\dfrac{\text{France has a king}}{\begin{bmatrix} x:\mathsf{e} \\ \mathbf{KingOf}(x,fr)\end{bmatrix}}}{\left(u:\begin{bmatrix} x:\mathsf{e} \\ \mathbf{KingOf}(x,fr)\end{bmatrix}\right)\to\boxed{id}}{>}\qquad \dfrac{\dfrac{\dfrac{\text{the}}{\lambda n.\begin{bmatrix} u@\begin{bmatrix} x:\mathsf{e} \\ n(x)\end{bmatrix} \\ \boxed{\pi_1 u}\end{bmatrix}}\quad\dfrac{\dfrac{\text{king}}{\lambda z.\lambda x.\,\mathbf{kingOf}(x,z)}\quad\dfrac{\dfrac{\text{of}}{id}\quad\dfrac{\text{France}}{fr}}{fr}{>}}{\lambda x.\mathbf{kingOf}(x,fr)}{>}}{\begin{bmatrix} u@\begin{bmatrix} x:\mathsf{e} \\ \mathbf{kingOf}(x,fr)\end{bmatrix} \\ \boxed{\pi_1 u}\end{bmatrix}}{>}\quad\dfrac{\text{is wise}}{\boxed{\lambda x.\mathbf{wise}^*_s(x)}}}{\begin{bmatrix} v@\begin{bmatrix} y:\mathsf{e} \\ \mathbf{KingOf}(y,fr)\end{bmatrix} \\ \boxed{\mathbf{wise}^*_s(\pi_1 v)}\end{bmatrix}}{<^\star}}{\left(u:\begin{bmatrix} x:\mathsf{e} \\ \mathbf{KingOf}(x,fr)\end{bmatrix}\right)\to\begin{bmatrix} v@\begin{bmatrix} y:\mathsf{e} \\ \mathbf{KingOf}(y,fr)\end{bmatrix} \\ \boxed{\mathbf{wise}^*_s(\pi_1 v)}\end{bmatrix}}{>^\star}$$

(713) Type check diagram of (709)

$$
\dfrac{\begin{array}{c}\vdots\\ \begin{bmatrix} x:\mathsf{e}\\ \mathbf{kingOf}(x,fr)\end{bmatrix}:\mathsf{type}\end{array} \qquad \dfrac{\begin{array}{c}\vdots\\ \begin{bmatrix} y:\mathsf{e}\\ \mathbf{kingOf}(y,fr)\end{bmatrix}:\mathsf{type}\end{array} \quad \begin{array}{c}\dfrac{}{u:\begin{bmatrix} x:\mathsf{e}\\ \mathbf{kingOf}(x,fr)\end{bmatrix}}{}^{1}\\ \vdots\\ ?:\begin{bmatrix} y:\mathsf{e}\\ \mathbf{kingOf}(y,fr)\end{bmatrix}\end{array} \quad \begin{array}{c}\vdots\\ \mathbf{wise}(\pi_1 v)[?/v]\end{array}}{\begin{bmatrix} v@\begin{bmatrix} y:\mathsf{e}\\ \mathbf{kingOf}(y,fr)\end{bmatrix}\\ \mathbf{wise}(\pi_1 v)\end{bmatrix} : \mathsf{type}}(@)}{\left(u:\begin{bmatrix} x:\mathsf{e}\\ \mathbf{kingOf}(x,fr)\end{bmatrix}\right)\to\begin{bmatrix} v@\begin{bmatrix} y:\mathsf{e}\\ \mathbf{kingOf}(y,fr)\end{bmatrix}\\ \mathbf{wise}(\pi_1 v)\end{bmatrix}:\mathsf{type}}(\Pi F),1
$$

(714) @-elimination of (709)

$$
\dfrac{\begin{array}{c}\vdots\\ \begin{bmatrix} x:\mathsf{e}\\ \mathbf{kingOf}(x,fr)\end{bmatrix}:\mathsf{type}\end{array} \qquad \dfrac{\dfrac{}{\mathbf{wise}:\mathsf{e}\to\mathsf{type}}(CON) \quad \dfrac{\dfrac{}{u:\begin{bmatrix} x:\mathsf{e}\\ \mathbf{kingOf}(x,fr)\end{bmatrix}}{}^{1}}{\pi_1 u:\mathsf{e}}(\Sigma E)}{\mathbf{wise}(\pi_1 u):\mathsf{type}}(\Pi E)}{\left(u:\begin{bmatrix} x:\mathsf{e}\\ \mathbf{kingOf}(x,fr)\end{bmatrix}\right)\to\mathbf{wise}(\pi_1 u):\mathsf{type}}(\Pi F),1
$$

Type checking algorithm returns a fully-specified semantic representation. Presupposition filtering is performed via exactly the same process as anaphora resolution.

Proposition 464 (A takeaway message from DTS on Presupposition Filtering)**.** A presupposition is filtered in the same way as an anaphoric expression is resolved.

26.5.2 On Projection

The projection of presupposition is naturally accounted for using DTS. Consider how to derive the presupposition projected out of negation. Recall that negation is defined to be an implication of the form $\neg A \equiv A \to \bot$.where $\bot$ is the absurdity type, i.e., the type that has no inhabitants. Given the formation rule for the bottom type shown on the left below, the formation rule for negation can be derived as on the right:

$$
\dfrac{}{\bot:\mathsf{type}}(\bot F) \qquad\qquad \dfrac{A:\mathsf{type}}{\neg A:\mathsf{type}}(\neg F)
$$

The underspecified term $@_1$ is a function that takes a local context c as argument. A term of the form $@_i\Lambda$ is called *underspecified term* and specifies that the term $@_i$ has type Λ. In the case of (**??**), the term $@_i$ is annotated with a Σ-type $(x:\mathsf{e})\times nx$. This means that the underspecified term $@_i$ is a term having the Σ-type. In this case, such a term is a pair of an entity x and a proof that x satisfies the condition n. Then its first projection, i.e., an entity x, is applied to a given predicate p. The annotated Σ-type here means that the underspecified term $@_i$ introduced by the definite article *the* requires a pair of an entity x and a proof that x satisfies the condition n provided by the restrict of the category N, given a local context c. the type $(x:\mathbf{E})\times\mathbf{door}(x)$ from a given local context c. Such a pair consists of some entity x and a proof that x is a door, Then its first projection, i.e., an entity x, is applied to a given predicate. This means that for the entire sentence containing an expression of the form *the* N to be well-typed, one needs to construct a proof of the existence of such an N. Intuitively, this captures the existence presupposition triggered by a definite description.

According to the formation rule (ΠF), the proposition A and its negation $\neg A$ have the same presupposition.

(715) It is not the case that the king of France is bald.

(716) $$\neg \left[\begin{array}{l} @\left[\begin{array}{l} x : \mathsf{e} \\ \mathbf{kingOf}(x, fr) \end{array}\right] \\ \mathbf{bold}(\pi_1 u) \end{array}\right]$$

The anaphora/presupposition resolution for the semantic representation (SR) A is triggered by the judgement $\vdash A : \mathbf{type}$. This means that the presupposition resolution is amount to proving that the SR in question is well-typed given the signature.

Assuming that $\mathbf{wise} : \mathsf{e} \to \mathbf{type}$ is in the signature, the proof that the SR yielded by the sentence *The king of France is not wise* is well-typed runs as follows.[*1]

(717) Syntactic structure of (715)

$$\dfrac{\dfrac{\text{It is not the case that}}{S^\star/S} \qquad \dfrac{\dfrac{\text{the king of France}}{NP^\star} \qquad \dfrac{\dfrac{\text{is}}{(S\backslash NP/(S\backslash NP))^\star} \qquad \dfrac{\text{bald}}{(S\backslash NP)^\star}}{(S\backslash NP)^\star}\,{>^\star}}{\dfrac{S^\star}{S}\,\checkmark}\,{<^\star}}{\dfrac{S^\star}{S}\,\checkmark}\,{>}$$

(718) Semantic composition of (715)

$$\dfrac{\dfrac{\dfrac{\text{It is not the case that}}{\neg\boxed{id}} \qquad \dfrac{\dfrac{\text{the king of France}}{\left[\begin{array}{l} u@\left[\begin{array}{l} x : \mathsf{e} \\ \mathbf{kingOf}(x, fr) \end{array}\right] \\ \boxed{\pi_1 u} \end{array}\right]} \qquad \dfrac{\dfrac{\text{is}}{\boxed{id}} \quad \dfrac{\text{bald}}{\boxed{\lambda x.\mathbf{bald}^*_s(x)}}}{\boxed{\lambda x.\mathbf{bald}^*_s(x)}}\,{>}}{\left[\begin{array}{l} u@\left[\begin{array}{l} x : \mathsf{e} \\ \mathbf{kingOf}(x, fr) \end{array}\right] \\ \boxed{\mathbf{bald}^*_s(\pi_1 u)} \end{array}\right]}\,{>}}{\neg\left[\begin{array}{l} u@\left[\begin{array}{l} x : \mathsf{e} \\ \mathbf{kingOf}(x, fr) \end{array}\right] \\ \boxed{\mathbf{bald}^*_s(\pi_1 u)} \end{array}\right]}\,{>}}{\neg\left[\begin{array}{l} u@\left[\begin{array}{l} x : \mathsf{e} \\ \mathbf{kingOf}(x, fr) \end{array}\right] \\ \mathbf{bald}^*_s(\pi_1 u) \end{array}\right]}\,\checkmark$$

[*1] To be explicit, we the sequent-style of natural deduction, where the context on which each step in a derivation depends is indicated on the left-hand side of $\vdash$.

(719) Type check diagram of (715)

$$\dfrac{\dfrac{\begin{matrix}\vdots\\ \begin{bmatrix} x : \mathsf{e} \\ \mathbf{kingOf}(x, fr) \end{bmatrix} : \mathsf{type}\end{matrix} \quad \begin{matrix}\vdots\\ ? : \begin{bmatrix} x : \mathsf{e} \\ \mathbf{kingOf}(x, fr) \end{bmatrix}\end{matrix} \quad \begin{matrix}\vdots\\ \mathbf{bald}^*_s(\pi_1 u)[?/u] : \mathsf{type}\end{matrix}}{\begin{bmatrix} u@\begin{bmatrix} x : \mathsf{e} \\ \mathbf{kingOf}(x, fr) \end{bmatrix} \\ \mathbf{bald}^*_s(\pi_1 u) \end{bmatrix} : \mathsf{type}}(@) \quad \dfrac{}{\bot : \mathsf{type}}(\bot F)}{\neg \begin{bmatrix} u@\begin{bmatrix} x : \mathsf{e} \\ \mathbf{kingOf}(x, fr) \end{bmatrix} \\ \mathbf{bald}^*_s(\pi_1 u) \end{bmatrix} : \mathsf{type}}(\Pi F)$$

In order for the sentence "The king of France is bald" to be well-formed, the context Γ must be such that the following type inhabits a proof (namely, there exists a king of France).

(720) $\Gamma \vdash ? : \begin{bmatrix} x : \mathsf{e} \\ \mathbf{kingOf}(x, fr) \end{bmatrix}$

The same inference is triggered for the antecedent of a conditional sentence like (721).

(721) If the king of France is wise, people will be happy.

(722) Type check diagram of (721)

$$\dfrac{\begin{matrix}\vdots\\ \begin{bmatrix} u@\begin{bmatrix} x : \mathsf{e} \\ \mathbf{kingOf}(x, fr) \end{bmatrix} \\ \mathbf{wise}(\pi_1 u) \end{bmatrix} : \mathsf{type}\end{matrix} \quad \begin{matrix}\vdots\\ \mathbf{happy}(\mathit{people}) : \mathsf{type}\end{matrix}}{\begin{bmatrix} u@\begin{bmatrix} x : \mathsf{e} \\ \mathbf{kingOf}(x, fr) \end{bmatrix} \\ \mathbf{wise}(\pi_1 u) \end{bmatrix} \to \mathbf{happy}(\mathit{people}) : \mathsf{type}}(\Pi F)$$

What is presupposed by the original sentence in (**??**b) can be read off from the open branch ending with the judgment having the underspecified term $@_1$. The context on which this judgement depends can be made explicit in the following way. For the given SR to be well-typed, (which means that the utterance of (**??**b) is felicitous), one has to find a term by which (723) holds.

(723) $\begin{bmatrix} x : \mathsf{e} \\ \mathbf{kof}(x) \end{bmatrix}$ *true*

That is to say, one has to find a proof term for the proposition that there is a king of France. In this way, we can derive the existence presupposition for the negated sentence (**??**b), as well as for the positive counterpart (**??**a). As is easily seen by the definition $\neg A \equiv A \to \bot$, the same inference is triggered for the antecedent of a conditional sentence in (**??**c). Thus we can also account for presupposition projection for conditionals as exemplified in (**??**c).

Proposition 465 (A takeaway message from DTS on Presupposition Projection)**.** A presupposition projects because it's truth is a requirement for a sentence containing it to be *semantically well-formed*, not to be *true*.

Corollary 466. Existence of an antecedent is a requirement for a sentence containing anaphora to be semantically well-formed, not to be true.

26.6 Accommodation

Consider (724), taken from Heim (1983).[*2]

(724) No nation cherishes its king.

The sentence (724) is interpreted in two distinct ways regarding its presupposition, as described in Heim (1983, p. 403) as follows:

> What about quantifiers other than universal? Concerning "no," we find conflicting factual claims in the literature. According to Cooper (1983), (24) [=our (724)] should presuppose that every nation (in the relevant domain of discourse) has a king; for **?**, it presuppose that some nation does.

One reading of (724) that implies `every nation has a king` is obtained via the *global accommodation*, whereas another reading of (724) implying `Some nation has a king` is obtained via the *local accommodation*.

We assume that the lexical item of `its` is defined as (725) in almost the same way as `his` in (**??**) except for its presuppositional content:

(725) $$[\![\text{its} \vdash NP^\star/N]\!] \stackrel{def}{\equiv} \lambda n. \begin{bmatrix} u@\begin{bmatrix} x : \mathsf{e} \\ \neg\mathbf{human}(x) \end{bmatrix} \\ v@\begin{bmatrix} y : \mathsf{e} \\ n(y) \\ \mathbf{of}(y, \pi_1 u) \end{bmatrix} \\ \boxed{\pi_1 v} \end{bmatrix}$$

Since the syntactic structure of (724) is almost the same as (**??**), we obtain its semantic representation via the homomorphic semantic composition, which is (726) corresponding to the BVA reading for (724).

(726) $$(w : (x : \mathsf{e}) \times \mathbf{nation}(x)) \to \neg \begin{bmatrix} u@(x : \mathsf{e}) \times \neg\mathbf{human}(x) \\ v@(y : \mathsf{e}) \times \mathbf{king}(y) \times \mathbf{of}(y, \pi_1 u) \\ \mathbf{cherish}(\pi_1 v, \pi_1 w) \end{bmatrix}$$

The type checking for (726) factors through (727).

(727) $$\Gamma,\ w : (x : \mathsf{e}) \times \mathbf{nation}(x) \vdash \begin{bmatrix} u@(x : \mathsf{e}) \times \neg\mathbf{human}(x) \\ v@(y : \mathsf{e}) \times \mathbf{king}(y) \times \mathbf{of}(y, \pi_1 u) \\ \mathbf{cherish}(\pi_1 v, \pi_1 w) \end{bmatrix} : \mathsf{type}$$

which will invoke the @-rule (**??**) and call for the type checking (728a), which obviously succeeds, and then the proof search (728b).

(728) a. $\Gamma,\ w : (x : \mathsf{e}) \times \mathbf{nation}(x) \vdash (x : \mathsf{e}) \times \neg\mathbf{human}(x) : \mathsf{type}$

b. $\Gamma,\ w : (x : \mathsf{e}) \times \mathbf{nation}(x) \vdash ? : (x : \mathsf{e}) \times \neg\mathbf{human}(x)$

[*2] Sentence (24) on p. 403.

If we assume world knowledge that every nation is non-human (i.e., the constant term **nnh**: $(x : \mathrm{e}) \to \mathbf{nation}(x) \to \neg\mathbf{human}(x))$, the proof search for (728b) will find the term $(\pi_1 w, \mathbf{nnh}(\pi_1 w)(\pi_2 w))$ that corresponds to the BVA reading of (724). The next step is the type checking (729), which breaks down to the type checking (730a) and the proof search (730b).

(729) $\Gamma,\ w : (x : \mathrm{e}) \times \mathbf{nation}(x) \vdash \left[\begin{array}{l} v@(y : \mathrm{e}) \times \mathbf{king}(y) \times \mathbf{of}(y, \pi_1 w) \\ \mathbf{cherish}(\pi_1 v, \pi_1 w) \end{array}\right] : \mathsf{type}$

(730) a. $\Gamma,\ w : (x : \mathrm{e}) \times \mathbf{nation}(x) \vdash (y : \mathrm{e}) \times \mathbf{king}(y) \times \mathbf{of}(y, \pi_1 w) : \mathsf{type}$

b. $\Gamma,\ w : (x : \mathrm{e}) \times \mathbf{nation}(x) \vdash ? : (y : \mathrm{e}) \times \mathbf{king}(y) \times \mathbf{of}(y, \pi_1 w)$

Again, (730a) is a routine, and the proof search for (730b) is a search for the king of y. Remember that, in the case of (**??**) and (**??**), we assumed that there exists a world knowledge **fatherOf**, that is, every boy has a father. The question here is what happens if we do not have a knowledge that every nation has a king, and thus the proof searching for (730b) fails.

The first strategy for our type checker is global accommodation, defined in Bekki (2014), which can be restated as the following instruction.

Definition 467 (Global accommodation)**.** When the proof search

$$[\![\Gamma \vdash ? : A]\!]$$

fails for the type A such that $fv(A) = x_1, \ldots, x_n$, the type checker may add the constant term of type $(x_1 : A_1) \to \cdots \to (x_n : A_n) \to A$ (where $x_1 : A_1, \ldots, x_n : A_n \subseteq \Gamma$) to the signature and re-run the type checking.

Applying Definition 467 to the case of (730b), the constant term added to the signature is $(w : (x : \mathrm{e}) \times \mathbf{nation}(x)) \to (y : \mathrm{e}) \times \mathbf{king}(y) \times \mathbf{of}(y, \pi_1 w)$, which is exactly the knowledge that every nation has a king.

The second strategy for our type checker is local accommodation, which, under the definition of underspecified types, is given the following definition.

Definition 468 (Local accommodation)**.** When the proof search

$$[\![\Gamma \vdash ? : A']\!]$$

required for the type checking $[\![\Gamma \vdash (x@A) \times B : \mathsf{type}]\!]$ fails, the type checker may replace the result of $[\![\Gamma \vdash (x@A) \times B : \mathsf{type}]\!]$ with the result of type checking $[\![\Gamma \vdash (x : A) \times B : \mathsf{type}]\!]$.[*3]

[*3] This operation should not be applied to all types of anaphora and presupposition: for example, it is known that pronouns in general do not undergo local accommodation when their antecedents are missing. For this purpose, we may want to add a binary feature to underspecified types that tells us whether local accommodation is applicable to them, an approach pursued by Yana et al. (2024). Alternatively, we may argue that the prohibition of local accommodation for pronouns is based rather on pragmatic factors. This is a controversial issue and I would like to leave it open.

One benefit of employing the underspecified types is that it allows us to define the operation of local accommodation in a straightforward way. This is due to their definition that takes a closer form to that of the Σ-formation rule: when the proof search $[\![\Gamma \vdash ?: A]\!]$ fails in the type checking $[\![\Gamma \vdash (x@A) \times B : \mathbf{type}]\!]$, the semantic system may optionally replace $(x@A) \times B$ with $(x : A) \times B$ and re-run the type checking. Note that this transformation is safe owing to the similarity between the verification conditions of the @-rule and Σ-formation rule.

In other words, the local accommodation in DTS is understood as accommodating the presupposed content of an underspecified type by existentially quantifying it. Applying Definition 468 to (730b), replacing the type checking for (729) with the following:

(731) $\Gamma,\ w : (x : \mathrm{e}) \times \mathbf{nation}(x) \vdash \begin{bmatrix} v : (y : \mathrm{e}) \times \mathbf{king}(y) \times \mathbf{of}(y, \pi_1 w) \\ \mathbf{cherish}(\pi_1 v, \pi_1 w) \end{bmatrix} : \mathbf{type}$

thus the whole semantic representation for (724) turns into (732).

(732) $\Gamma \ \vdash \ (w : (x : \mathrm{e}) \times \mathbf{nation}(x)) \ \rightarrow \ \neg \begin{bmatrix} v : (y : \mathrm{e}) \times \mathbf{king}(y) \times \mathbf{of}(y, \pi_1 w) \\ \mathbf{cherish}(\pi_1 v, \pi_1 w) \end{bmatrix} : \mathbf{type}$

In words, (732) claims that `No nation has a king and cherishes it`, which exactly corresponds to the locally accommodated reading of (724).

26.7 Discussion

History and Further Readings