

Solving Partee’s Temperature Puzzle in an EFL-Ontology

Kristina Liefke*

Munich Center for Mathematical Philosophy
Ludwig-Maximilians-Universität München
`Kristina.Liefke@lrz.uni-muenchen.de`

Abstract. According to the received view of type-logical semantics (suggested by Montague [18] (cf. [19, 21]) and adopted by many of his successors (cf. [1, 6–9, 28])), the number of a semantics’ basic types depends proportionally on the syntactic and lexical diversity of the modeled natural language fragment. This paper provides a counterexample to this principle. In particular, it shows that Partee’s temperature puzzle from [21] – whose solution is commonly taken to require a basic type for indices (for the formation of individual concepts) or for individual concepts – can be interpreted in the poorer type system from [19], which only assumes basic individuals and propositions. This result contributes to the general project of identifying the minimal semantic requirements on models for certain linguistic fragments.

Keywords: Individual concepts, Intensionalization, PTQ-fragment, Simple Theory of Types, Temperature puzzle.

1 Introduction

It is a commonplace in ontology engineering that the modeling of *complex* target systems requires *different kinds* of objects. For example, to build a semantic ontology for English, we assume the existence of individuals, propositions, properties of individuals, relations between individuals, situations, events (or eventualities), degrees, times, and other kinds of objects. These objects serve the interpretation of proper names, declarative sentences or complement phrases, common nouns or intransitive verbs, transitive verbs, neutral perception verbs, adverbial and degree modifiers, temporal adjectives, etc.

Traditional type-logical semantics (cf. [19–21]) tames this zoo of objects by assuming only a small set of primitive objects, and constructing all other types of objects from these primitives via a number of type-forming rules. In this way, Montague reduces the referents of the fragment of English from [19] to *two* basic types of objects: individuals (type ι) and propositions (analyzed as functions

* Thanks to Ede Zimmermann, whose comments on my talk at *Sinn und Bedeutung 18* inspired this paper.

from indices to truth-values, type o). From these objects, properties of individuals and binary relations between individuals are constructed as functions from individuals to propositions (type $\iota \rightarrow o$), respectively as functions from individuals to functions from individuals to propositions (type $\iota \rightarrow (\iota \rightarrow o)$). Because of its reductive effect on the number of basic types, this view of formal semantics will hereafter be called the *reduction view* of formal semantics.

In the last forty years, revisions and extensions of Montague’s formal semantics have caused many semanticists to depart from the reduction view. This departure is witnessed by the introduction of a fair number of new basic types (including types for primitive propositions [3, 28], situations [1, 16], events [7, 8], degrees [6], and times [9]). The introduction of these new types is motivated by the coarse grain of set-theoretic functions (s.t. Montague semantics sometimes generates wrong predictions about natural language entailment; cf. [2, 5, 17]), and by the need to find semantic values for ‘new’ kinds of expressions, which are not included in Montague’s small linguistic fragments (e.g. for neutral perception verbs, and adverbial and degree modifiers). Since many of these values are either treated as semantic primitives or are identified with constructions out of the new primitives, we will hereafter refer to this view of formal semantics as the *ontology engineering-view*.¹

The ‘ontology engineering-view’ of formal semantics is supported by Montague’s semantics from [19] and [21]: The latter interpret the fragments from [19] (hereafter, the *EFL-fragment*) and [21] (the *PTQ-fragment*) in models with primitive individuals and propositions, resp. with primitive individuals, indices, and truth-values. Since the PTQ-fragment extends the EFL-lexicon via intensional common nouns (e.g. *temperature*, *price*) and intensional intransitive verbs (e.g. *rise*, *change*), since PTQ-models extend EFL-models via (constructions out of) individual concepts, and since PTQ-models interpret intensional common nouns and intransitive verbs as functions over individual concepts, it is commonly assumed that any empirically adequate model for the PTQ-fragment requires a basic type for indices, or for individual concepts.

The availability of a larger number of basic types, which is advocated by proponents of the ‘ontology engineering view’, facilitates work for the empirical linguist: In a rich type system, fewer syntactic expressions are interpreted in a non-basic type. As a result, the compositional translations of many syntactic structures will be *simpler*, and will involve *less* lambda conversions than their ‘reductive’ counterparts.

However, the proliferation of basic types is not an altogether positive development. Specifically, the adoption of *additional* basic types *reduces* the number of representational relations between different types of objects (cf. [14, 23]) and obfuscates the semantic requirements on different linguistic fragments. For example, since Montague does not discuss the relation between his model-theoretic

¹ Notably, the ‘ontology engineering-view’ still understands formal semantics as a compositional enterprise, which identifies each expression’s semantic value with its contribution to the truth-conditions of the sentence in which the expression occurs.

objects from [19] and [21], it remains unclear whether a primitive type for indices is *required* for the interpretation of the PTQ-fragment. But an identification of these requirements is vital to our understanding of the semantic type system of natural language.

This paper aids the identification of the above requirements. In particular, it shows that Partee’s temperature puzzle from [21, pp. 267–268] – which is commonly taken to prove the impossibility (or undesirability) of interpreting the PTQ-fragment in an EFL-semantics – can be interpreted in the poorer semantic models from [19]. The paper is organized as follows: Sections 2 and 3 introduce Partee’s temperature puzzle for extensional semantics, and present Montague’s solution from [21]. Section 4 identifies a strategy for the puzzle’s solution in an EFL-ontology, and applies this strategy. The paper closes with a summary of our results and with pointers to future work.

2 Partee’s Temperature Puzzle for Extensional Semantics

Partee’s temperature puzzle [21, pp. 267–268] identifies a problem with extensional semantics for natural language, which regards their validation of the counterintuitive inference from ($\star$):

$$\frac{\begin{array}{l} \text{The temperature is ninety.} \\ \text{The temperature rises.} \end{array}}{\text{Ninety rises.}} \quad (\star)$$

The origin of this problem lies in the different readings of the phrase **the temperature** in the two premises of ($\star$), and in the inability of extensional semantics to accommodate one of these readings: In the second premise, the occurrence of the phrase **the temperature** is intuitively interpreted as a *function* (type $\sigma \rightarrow \iota$) from worlds and times (or ‘indices’, type σ) to the temperature at those worlds at the given times.² In the first premise, the occurrence of the phrase **the temperature** is interpreted as the *value* (type ι) of the above function at the actual world at the current time. Since extensional semantics (like the models of the Simple Theory of Types from [4], and Montague’s semantics from [19]) do not have a type for indices – s.t. they also lack a type for index-to-value functions –, they are unable to capture the reading of the phrase **the temperature** from the second premise.

The inference from the conjunction of the logical translations of the two premises of ($\star$) to the translation of the conclusion of ($\star$) in classical extensional logic (cf. [4]) is given in Figure 1 (next page). In the figure, the variables x and y range over individuals (type ι). The expression *ninety* is an individual constant (type ι). The expressions *temp* and *rise* are constants for functions from individuals to propositions (i.e. for first-order properties of individuals, type $\iota \rightarrow o$).

² As a result, this reading is sometimes called the *function reading* [15, 25]. The reading of the phrase **the temperature** from the first premise is called the *value reading*.

$$\frac{\begin{array}{l} \exists x \forall y. (\text{temp}(y) \leftrightarrow x = y) \wedge x = \text{ninety} \\ \exists x \forall y. (\text{temp}(y) \leftrightarrow x = y) \wedge \text{rise}(x) \end{array}}{\text{rise}(\text{ninety})}$$

Fig. 1. The valid counterintuitive inference in extensional semantics.

The asserted identity of the temperature x with the value *ninety* in the formula in the first premise of Figure 1 justifies the (counterintuitive) substitution of the translation of the phrase **the temperature** from the second premise by the translation of the name *ninety*.

3 Montague’s Intensional Solution to the Puzzle

Montague’s semantics from [21] blocks the counterintuitive inference from Figure 1 by interpreting intensional common nouns (e.g. **temperature**) and intransitive verbs (e.g. **rise**) as properties of *individual concepts* (type $(\sigma \rightarrow \iota) \rightarrow o$), and by restricting the interpretation of the copula **is** to a relation between the *extensions* of two individual concepts at the actual world @ at the current time (i.e. to a relation between *individuals*, type $\iota \rightarrow (\iota \rightarrow o)$). Since the first premise of $(\star)$ thus only asserts the identity of the individual ‘the temperature at @’ and the value *ninety*, it blocks the substitution of the ‘individual concept’-denoting phrase **the temperature** in the second premise of $(\star)$ by the name *ninety*.

The invalidity of the inference from the conjunction of the two premises of $(\star)$ to the conclusion of $(\star)$ in (a streamlined version of) Montague’s Intensional Logic (cf. [12]) is captured in Figure 2. In the figure, c and c_1 are variables over individual concepts (type $\sigma \rightarrow \iota$). The expressions **TEMP** and **RISE** are constants for functions from individual concepts to propositions (i.e. for first-order properties of individual concepts, type $(\sigma \rightarrow \iota) \rightarrow o$). **NINETY** is a constant (type $\sigma \rightarrow i$) for the function from indices to the denotation of the constant *ninety*.³ The remaining expressions are typed as above.

$$\frac{\begin{array}{l} \exists c \forall c_1. (\text{TEMP}(c_1) \leftrightarrow c = c_1) \wedge x = \text{ninety} \\ \exists c \forall c_1. (\text{TEMP}(c_1) \leftrightarrow c = c_1) \wedge \text{RISE}(c) \end{array}}{\text{RISE}(\text{NINETY})}$$

Fig. 2. The invalid counterintuitive inference in PTQ-semantics.

Since Montague’s models from [19] only assume basic types for individuals and propositions (s.t. they do not allow the construction of individual concepts via the available type-forming rules), it is commonly assumed that these models are unable to block the counterintuitive inference from $(\star)$. We show below that this assumption is mistaken.

³ Thus, **NINETY** satisfies the axiom $\forall i_\sigma. \text{NINETY}(i) = \text{ninety}$.

4 A New, Extensional, Solution to the Puzzle

To demonstrate that Montague’s models from [19] enable a solution to Partee’s temperature puzzle, we first identify a strategy for the type- $\{\iota, o\}$ representation of indices, which allows us to code all linguistically relevant objects from [21] as objects from [19] (in Sect. 4). We then show that type-incompatibilities, which result from this coding procedure, are easily resolved through (EFL-correlates of) the familiar type-shifting operations from [23] and [13] (in Sect. 4.2).

4.1 Coding PTQ- as EFL-Objects

To block the counterintuitive inference from $(\star)$ in the class of models from [19], we code indices as type- o propositions. This coding is made possible by the interpretation of o as the type for functions from indices to truth-values (s.t. there is an injective map, $\lambda i_\sigma. i = w$, from indices w to propositions), and is supported by a corollary of Stone’s Theorem [27].

The existence of an injective relation between indices and propositions justifies the replacement of each occurrence of σ in the type for individual concepts by the type o .⁴ The traditional designators of individual concepts will then be interpreted in the type $o \rightarrow \iota$. Intensional common nouns and intransitive verbs receive an interpretation in the type $(o \rightarrow \iota) \rightarrow o$. Expressions from all other lexical classes of the EFL-fragment retain their original type-assignment. In particular, extensional common nouns (e.g. *man*) and intransitive verbs (e.g. *walk*) are interpreted in the type $\iota \rightarrow o$.

The coding relation between objects in the models from [19] and [21] is captured in Definitions 3 and 4. In the definitions, PTQ-types and EFL-types are defined as follows:

Definition 1 (PTQ-types). The set 2Type of PTQ-*types* is the smallest set of strings such that $\iota, \sigma, t \in 2\text{Type}$ and, for all $\alpha, \beta \in 2\text{Type}$, $(\alpha \rightarrow \beta) \in 2\text{Type}$.

Definition 2 (EFL-types). The set 1Type of EFL-*types* is the smallest set of strings such that $\iota, o \in 1\text{Type}$ and, for all $\alpha, \beta \in 1\text{Type}$, $(\alpha \rightarrow \beta) \in 1\text{Type}$.

Our definition of PTQ-types follows [12] in treating the index-type σ as a fully-fledged type. Because of the correspondence between unary and n -ary function types (cf. [22]), our definition of EFL-types neglects the existence of types for n -ary functions.

We will hereafter call a frame (or ‘ontology’) which only contains objects of an EFL-type an EFL-*frame* (or -*ontology*), and call a logical language which only contains expressions of an EFL-type an EFL-*language*. Models with an EFL-frame are called EFL-*models*. A formal semantics which interprets natural language

⁴ Since the truth-value type t only relevantly occurs in the construction $\sigma \rightarrow t$, and since objects of the type $\sigma \rightarrow t$ correspond to objects of the type for propositions, o , we need not specify a separate coding for type- t objects.

age into EFL-models called an EFL-*semantics*. We adopt an analogous terminology for PTQ-types.

The relation between PTQ-types and EFL-types is captured below:

Definition 3 (Type-conversion). The function ξ sends PTQ-types to EFL-types by the following recursion:

- I. i. $\xi(\iota) = \iota$; $\xi(\sigma \rightarrow t) = o$;
- ii. $\xi(\sigma \rightarrow \iota) = (o \rightarrow \iota)$;
- II. $\xi(\alpha \rightarrow \beta) = (\xi(\alpha) \rightarrow \xi(\beta))$, where α and β are PTQ-types.

In virtue of the above, the function ξ sends the type for properties of individual concepts, $(\sigma \rightarrow \iota) \rightarrow (\sigma \rightarrow t)$, to the type for properties of functions from propositions to individuals, $(o \rightarrow \iota) \rightarrow o$.

Notably, the function ξ does not identify an EFL-correlate for *every* PTQ-type. In particular, ξ is undefined for some truth-value function types (e.g. for the type $\iota \rightarrow t$), and for types for functions from indices to other functions (e.g. for the type $\sigma \rightarrow (\sigma \rightarrow t)$). Our exclusion of these types is motivated by the complications that would arise from coding type- σ and - t objects separately. (For example, this coding might yield the representation $\xi(\sigma \rightarrow t) = (\xi(\sigma) \rightarrow \xi(t)) = (o \rightarrow o)$, instead of the desired $\xi(\sigma \rightarrow t) = o$). However, since no expression from the PTQ-fragment receives an interpretation in these types, the partial definition of ξ is unproblematic.

We next define the coding function on PTQ-objects:

Definition 4 (EFL-encoding). Let $\{D_\alpha \mid \alpha \in 2\text{Type}\}$ and $\{D'_\alpha \mid \alpha \in 1\text{Type}\}$ be a general PTQ- and a general EFL-frame, respectively, for which it holds that $D_\iota = D'_\iota$ and $D_{\sigma \rightarrow t} = D'_o$. For each PTQ-type α for which ξ is defined, we define a function $\Xi_\alpha : D_\alpha \rightarrow D'_{\xi(\alpha)}$ by the following recursion:

- I. i. $\Xi_\iota(d) = d$, if $d \in D_\iota$; $\Xi_{\sigma \rightarrow t}(d) = d$, if $d \in D_{\sigma \rightarrow t}$;
- ii. If $f \in D_{\sigma \rightarrow \iota}$, then $\Xi_{\sigma \rightarrow \iota}(f)$ is the function $g \in D'_{o \rightarrow \iota}$ s.t., for all $w \in D_\sigma$, $f(w) = g(\lambda i_\sigma. i = w)$;
- II. If $h \in D_{\alpha \rightarrow \beta}$, then $\Xi_{\alpha \rightarrow \beta}(h)$ is the function $h' \in D'_{\xi(\alpha) \rightarrow \xi(\beta)}$.

It is easy to prove that all functions Ξ_α are injections.

4.2 Modeling the Puzzle in EFL-Frames

The previous section has identified a procedure for the coding of PTQ-objects. The present section uses this coding to solve Partee's temperature puzzle in an EFL-ontology.

To facilitate our proof that the temperature puzzle can be solved in an $\{\iota, o\}$ -based semantics, we invert Montague's strategy of "generalizing to the worst case" [24, p. 34] (cf. [14]): Rather than interpreting all expressions from the PTQ-fragment as functions over (the EFL-correlates of) individual concepts and ob-

taining their extensional equivalents through the use of meaning postulates, we interpret these expressions as functions over individuals. Only intensional common nouns and intensional intransitive verbs retain their interpretation as functions over (the EFL-correlates of) individual concepts. The application of the logical translations of intensional expressions to the translations of other PTQ-expressions is then handled through type-shifting.

In particular, to enable the application⁵ of the EFL-translations of determiners (type $(\iota \rightarrow o) \rightarrow ((\iota \rightarrow o) \rightarrow o)$) to the translations of intensional common nouns (type $(\underline{o \rightarrow \iota}) \rightarrow o$) in an EFL-typed logic, we introduce the *extensionalization* operator *ext*. This operator sends properties of the EFL-correlates of *individual concepts* (type $(\underline{o \rightarrow \iota}) \rightarrow o$) to properties of *individuals* (type $\iota \rightarrow o$). Below, the type-*o* constant **w** denotes the EFL-correlate of the current index, @.⁶ The variables **c** (type $\underline{o \rightarrow \iota}$) and **T** (type $(o \rightarrow \iota) \rightarrow o$) are the EFL-proxies for variables over individual concepts, resp. over properties of individual concepts.

Definition 5. The function $ext := \lambda T \lambda x \exists c. T(c) \wedge x = c(w)$ sends terms of the type $(o \rightarrow \iota) \rightarrow o$ to terms of the type $\iota \rightarrow o$.

The operator *ext* enables the extensionalization of the type- $((o \rightarrow \iota) \rightarrow o)$ translation, **temp**, of the noun **temperature** to the term $\lambda x \exists c. temp(c) \wedge x = c(w)$. This term denotes the property of being identical to the result of applying some type- $(o \rightarrow \iota)$ witness, **c**, of the property denoted by **temp** to the type-*o* correlate, **w**, of the current index. As a result, the term $\lambda x \exists c. temp(c) \wedge x = c(w)$ denotes the property of being the temperature at the current index.

The possibility of interpreting intensional nouns in the type $\iota \rightarrow o$ enables the EFL-translation of the first premise from ($\star$). This translation is given in (1). In the translation, $lift := \lambda x \lambda P_{\iota \rightarrow o}. P(x)$ is a variant of the type-lifting operator from [23]. The variables *P* and *P*₁ range over objects of the type $\iota \rightarrow o$. The variable *Q* ranges over objects of the type $(\iota \rightarrow o) \rightarrow o$:

1. $[_{NP} \text{ninety}] \rightsquigarrow ninety$ (1)
2. $[_{TV} \text{is}] \rightsquigarrow \lambda Q \lambda x. Q(\lambda y. x = y)$
3. $[_{VP} [_{TV} \text{is}] [_{NP} \text{ninety}]] \rightsquigarrow \lambda Q \lambda x. Q(\lambda y. x = y) [lift(ninety)]$
 $= \lambda Q \lambda x. Q(\lambda y. x = y) [\lambda P. P(ninety)]$
 $= \lambda x. [\lambda P. P(ninety)](\lambda y. x = y)$
 $= \lambda x. (x = ninety)$
4. $[_{N} \text{temperature}] \rightsquigarrow temp = \lambda c. temp(c)$
5. $[_{DET} \text{the}] \rightsquigarrow \lambda P_1 \lambda P \exists x \forall y. (P_1(y) \leftrightarrow x = y) \wedge P(x)$
6. $[_{NP} [_{DET} \text{the}] [_{N} \text{temperature}]] \rightsquigarrow \lambda P_1 \lambda P \exists x \forall y. (P_1(y) \leftrightarrow x = y) \wedge P(x) [ext(temp)]$
 $= \lambda P_1 \lambda P \exists x \forall y. (P_1(y) \leftrightarrow x = y) \wedge P(x) [\lambda z \exists c. temp(c) \wedge z = c(w)]$
 $= \lambda P \exists x \forall y. ((\exists c. temp(c) \wedge y = c(w)) \leftrightarrow x = y) \wedge P(x)$

⁵ Here, the argument-type is underlined.

⁶ Thus, **w** := $(\lambda i_{\sigma}. i = @)$.

$$\begin{aligned}
7. \quad & [_S[_{NP}[_{DET}the] [_Ntemperature]] [_{VP}[_{TV}is] [_{NP}ninety]]] \\
& \rightsquigarrow \lambda P \exists x \forall y. ((\exists c. \mathbf{temp}(c) \wedge y = c(\mathbf{w})) \leftrightarrow x = y) \wedge P(x) [\lambda z. (z = \mathbf{ninety})]) \\
& = \exists x \forall y. ((\exists c. \mathbf{temp}(c) \wedge y = c(\mathbf{w})) \leftrightarrow x = y) \wedge x = \mathbf{ninety}
\end{aligned}$$

Notably, the term from (1.7) does not result from the term in the first premise of Figure 2 by replacing ‘ c ’ and ‘ c_1 ’ by ‘ c ’ and ‘ c_1 ’, and by replacing ‘TEMP’ and ‘RISE’ by ‘**temp**’ and ‘**rise**’, respectively. In particular, while the term in the first premise of Figure 2 states the existence of a unique witness of the type- $((\mathbf{o} \rightarrow \iota) \rightarrow o)$ property of being a temperature (i.e. the existence of a sequence of ordered index/number-pairs), the term from (1.7) only states the existence of a unique witness of the type- $(\iota \rightarrow o)$ property of being the temperature *at the current index* (i.e. the existence of the number ninety). However, since the occurrence of the phrase **the temperature** in the first premise of $(\star)$ receives an extensional interpretation (type ι), this weakening is unproblematic. We will see at the end of this section that our weaker EFL-term still blocks Partee’s temperature puzzle.

To enable the application of determiners (traditionally, type $(\iota \rightarrow o) \rightarrow ((\iota \rightarrow o) \rightarrow o)$) to intensional common nouns (type $(\mathbf{o} \rightarrow \iota) \rightarrow o$) – and, thus, to enable the application of noun phrases to intensional intransitive verbs –, we introduce the *intensionalization* operator *int*. This operator sends individuals to the EFL-correlates of individual concepts, and sends functions from properties of *individuals* to generalized quantifiers over *individuals* (type $(\iota \rightarrow o) \rightarrow ((\iota \rightarrow o) \rightarrow o)$) to functions from the EFL-correlates of properties of *individual concepts* to the EFL-correlates of generalized quantifiers over *individual concepts* (type $((\mathbf{o} \rightarrow \iota) \rightarrow o) \rightarrow (((\mathbf{o} \rightarrow \iota) \rightarrow o) \rightarrow o)$).

In Definition 6, **ninety** is the EFL-correlate of the constant NINETY.⁷

Definition 6. The operator *int* then works as follows:

$$\begin{aligned}
\mathit{int}(\mathbf{ninety}) &:= \mathbf{ninety} \\
\mathit{int}(\lambda P_1 \lambda P \exists x. P_1(x) \wedge P(x)) &:= \lambda T_1 \lambda T \exists c. T_1(c) \wedge T(c) \\
\mathit{int}(\lambda P_1 \lambda P \forall x. P_1(x) \rightarrow P(x)) &:= \lambda T_1 \lambda T \forall c. T_1(c) \rightarrow T(c) \\
\mathit{int}(\lambda P_1 \lambda P \exists x \forall y. (P_1(y) \leftrightarrow x = y) \wedge P(x)) \\
&:= \lambda T_1 \lambda T \exists c \forall c_1. (T_1(c_1) \leftrightarrow c = c_1) \wedge T(c)
\end{aligned}$$

The operator *int* is the EFL-correlate of an ‘ ι -to- $(\sigma \rightarrow \iota)$ ’-restricted variant of the intensionalization operator for extensional TY_2 terms from [13] (cf. [10, Ch. 8.4]). This operator systematically replaces each occurrence of ι in the type of a linguistic expression by the type $\sigma \rightarrow \iota$ (our type $\mathbf{o} \rightarrow \iota$). As a result, the type for generalized quantifiers over individuals, $(\iota \rightarrow o) \rightarrow o$, will be replaced by the type $((\sigma \rightarrow \iota) \rightarrow o) \rightarrow o$ (our type $((\mathbf{o} \rightarrow \iota) \rightarrow o) \rightarrow o$).

⁷ Thus, **ninety** satisfies the metatheoretical axiom $\forall p_o. \mathbf{ninety}(p) = \mathbf{ninety}$.

The interpretation of intensional noun phrases in the type $((o \rightarrow \iota) \rightarrow o) \rightarrow o$ enables the translation of the second premise from $(\star)$. In the translation, the type- $(o \rightarrow \iota) \rightarrow o$ term *rise* is the EFL-correlate of the constant RISE.

1. $[_{IV} \text{rises}] \rightsquigarrow \text{rise}$ (2)
2. $[_N \text{temperature}] \rightsquigarrow \text{temp}$
3. $[_{DET} \text{the}] \rightsquigarrow \lambda P_1 \lambda P \exists x \forall y. (P_1(y) \leftrightarrow x = y) \wedge P(x)$
4. $[_{NP}[_{DET} \text{the}][_N \text{temperature}]] \rightsquigarrow \text{int}(\lambda P_1 \lambda P \exists x \forall y. (P_1(y) \leftrightarrow x = y) \wedge P(x))[_{\text{temp}}]$
 $= \lambda T \exists c \forall c_1. (\text{temp}(c_1) \leftrightarrow c = c_1) \wedge T(c)$
5. $[_S[_{NP}[_{DET} \text{the}][_N \text{temperature}]][_{IV} \text{rises}]] \rightsquigarrow \exists c \forall c_1. (\text{temp}(c_1) \leftrightarrow c = c_1) \wedge \text{rise}(c)$

The interpretation of proper names in the type $o \rightarrow \iota$ enables us to translate the conclusion from $(\star)$ as follows:

1. $[_{NP} \text{ninety}] \rightsquigarrow \text{ninety}$ (3)
2. $[_{IV} \text{rises}] \rightsquigarrow \text{rise}$
3. $[_S[_{NP} \text{ninety}][_{IV} \text{rises}]] \rightsquigarrow \text{rise}[\text{int}(\text{ninety})] = \text{rise}(\text{ninety})$

This completes our translation of the ‘ingredient sentences’ for Partee’s temperature puzzle. The invalid inference from the conjunction of (1.7) and (2.5) to (3.3) in our EFL-semantics is captured in Figure 3:

$$\frac{\exists x \forall y. ((\exists c. \text{temp}(c) \wedge y = c(w)) \leftrightarrow x = y) \wedge x = \text{ninety} \quad \exists c \forall c_1. (\text{temp}(c_1) \leftrightarrow c = c_1) \wedge \text{rise}(c)}{rise(\text{ninety})}$$

Fig. 3. The invalid inference in EFL-semantics.

In particular, while the formula in the second premise attributes the property ‘rise’ to the type- $(o \rightarrow \iota)$ object which has the property of being a temperature, the formula in the first premise attributes the property ‘is ninety’ only to the result (type ι) of applying a temperature-object to the EFL-correlate of @. In virtue of this fact – and the resulting invalidity of substituting *ninety* for *c* in the second premise of Figure 3 –, the formula in the conclusion does not follow from the conjunction of the two premise-formulas by the rules of classical logic.

5 Conclusion

This paper has demonstrated that Partee’s temperature puzzle can be solved in an EFL-semantics, which only commands basic types for individuals and propositions. We have obtained this result by coding indices in the type for propositions, and by defining shiftability relations between functions over individuals and (the EFL-correlates of) individual concepts. Since the temperature puzzle is a general litmus test for empirically adequate PTQ-models, the result shows that ontologically minimal models for the PTQ-fragment need not contain indices or individual concepts.

The identification of the minimal semantic requirements on PTQ-models is a contribution to the general program of *reverse formal semantics*. This program – named in analogy with the corresponding program, *reverse mathematics*, in the foundations of mathematics [11, 26] – attempts to identify the *minimal* formal semantics (with ‘minimality’ defined w.r.t. the number and complexity of basic types) which interpret certain fragments of natural language. Questions in reverse formal semantics include the following:

- (1) What *minimal number* of types is required for the interpretation of certain fragments of natural language (e.g. the EFL- or PTQ-fragment)?
- (2) What is the *identity* of the types in the smallest sets of types from (1)? What are the objects in the types’ domains?
- (3) Which equinumerous type-sets are *equivalent* (up to coding) to the sets from (1) and (2) (s.t. they preserve the explanatory and modeling power of a semantics with basic types from the original set)?
- (4) By (1) to (3): Which existing formal semantics (for which fragments?) can be *reduced* to which other semantics (for which other fragments)?
- (5) Which types (if any) resist coding through the familiar types (s.t. they require the *jump* from one to another, non-equivalent, set of types)?

The present paper has provided a partial answer for particular cases of questions (1), (3), and (4). In particular, it has shown that the set of basic types from [19] is equivalent to the basic-type set from [21] (cf. (3)), such that we can reduce Montague’s formal semantics for the PTQ-fragment to (a variant of) the semantics from [19] (cf. (4)).

We hope that the answers to other cases of the above questions, and the answers to the remaining questions will identify new relations between existing formal semantic models, and that they will further our insight into the type system of natural language.

References

- [1] J. Barwise, *Scenes and other situations*, The Journal of Philosophy **78/7** (1981), 369–397.
- [2] J. Barwise and J. Perry, *Situations and Attitudes*, MIT Press, Cambridge, 1983.
- [3] G. Chierchia and R. Turner, *Semantics and property theory*, Linguistics and Philosophy **11** (1988), 261–302.
- [4] A. Church, *A formulation of the Simple Theory of Types*, Journal of Symbolic Logic **5/2** (1940), 56–68.
- [5] M. J. Cresswell, *Logics and Languages*, Methuen Young Books, London, 1973.
- [6] ———, *The semantics of degree*, Montague Grammar (B. Partee, ed.), Academic Press, New York, 1976.
- [7] D. Davidson, *The logical form of action sentences*, Essays on Actions and Events: Philosophical essays of Donald Davidson, Clarendon Press, Oxford, 2001.

- [8] D. R. Dowty, *Word Meaning and Montague Grammar: The semantics of verbs and times in generative semantics and in Montague's PTQ*, Synthèse Language Library, vol. 7, D. Reidel Publishing Company, Dordrecht, 1979.
- [9] M. Duží, B. Jespersen, and P. Materna, *Foundations of semantic analysis*, Procedural Semantics for Hyperintensional Logic: Foundations and applications of Transparent Intensional Logic, Springer, Dordrecht etc., 2010, 2010.
- [10] J. van Eijck and Ch. Unger, *Computational Semantics with Functional Programming*, Cambridge University Press, Cambridge, 2010.
- [11] H. Friedman, *Some systems of second order arithmetic and their use*, Proceedings of the International Congress of Mathematicians, Vol. 1, Canadian Mathematical Congress, Montréal, 1974, 1975.
- [12] D. Gallin, *Intensional and Higher-Order Modal Logic with Applications to Montague Semantics*, North Holland, Amsterdam, 1975.
- [13] P. de Groote and M. Kanazawa, *A note on intensionalization*, Journal of Logic, Language and Information **22/2** (2013), 173–194.
- [14] H. Hendriks, *Flexible Montague Grammar*, ITLI Prepublication Series for Logic, Semantics and Philosophy of Language **08** (1990).
- [15] T.M.V. Janssen, *Individual concepts are useful*, Varieties of Formal Semantics: Proceedings of the 4th Amsterdam Colloquium (F. Landman and F. Veltman, eds.), Groningen-Amsterdam Studies in Semantics, vol. 3, 1984, 1984.
- [16] A. Kratzer, *An investigation into the lumps of thought*, Linguistics and Philosophy **12** (1989), 607–653.
- [17] D. Lewis, *General semantics*, Synthèse **22/1–2** (1970), 18–67.
- [18] R. Montague, *On the nature of certain philosophical entities*, Formal Philosophy: Selected papers of Richard Montague (R. H. Thomason, ed.), Yale University Press, New Haven, 1976.
- [19] ———, *English as a formal language*, Formal Philosophy, 1976.
- [20] ———, *Universal grammar*, Formal Philosophy, 1976.
- [21] ———, *The proper treatment of quantification in ordinary English*, Formal Philosophy, 1976.
- [22] R. Muskens, *Meaning and Partiality*, CSLI Lecture Notes, CSLI Publications, Stanford, 1995.
- [23] B. Partee, *Noun phrase interpretation and type-shifting principles*, Studies in Discourse Representation Theory and the Theory of Generalized Quantifiers (J. Groenendijk, D. de Jong, and M. Stokhof, eds.), Foris Publications, Dordrecht, 1987.
- [24] ———, *The development of formal semantics*, The Handbook of Contemporary Semantic Theory (S. Lappin, ed.), Blackwell, Oxford, 1996.
- [25] M. Schwager, *Bodyguards under cover: The status of individual concepts*, Proceedings of SALT XVII (T. Friedman and M. Gibson, eds.), Cornell University, Ithaca, NY, 2007.
- [26] S. G. Simpson, *Subsystems of Second Order Arithmetic*, Perspectives in Logic, vol. 2, Cambridge University Press, Cambridge, 2009.
- [27] M. H. Stone, *The theory of representation for Boolean algebras*, Transactions of the American Mathematical Society **40/1** (1936), 37–111.
- [28] R. H. Thomason, *A model theory for the propositional attitudes*, Linguistics and Philosophy **4** (1980), 47–70.