A Mathematical Formalism for HPSG

Frank Richter

CLAIRE-Workshop,
University of Essex,
12–13 January 2001

I

Pollard and Sag 1994

Methodological Background

Let us begin by making explicit some methodological assumptions. In any mathematical theory about an empirical domain, the phenomena of interest are *modelled* by mathematical structures, certain aspects of which are conventionally understood as corresponding to observables of the domain. The theory itself does not talk directly about the empirical phenomena; instead, it talks about, or is *interpreted by*, the modelling structures. Thus the predictive power of the theory arises from the conventional correspondence between the model and the empirical domain. (Pollard and Sag, 1994, p. 6)

In our view, a linguistic theory should bear exactly the same relation to the empirical domain of natural language, namely, the universe of possible linguistic objects, as a mathematical theory of celestial mechanics bears to the possible motions of n-body systems. Thus we insist on being explicit as to what sorts of constructs are assumed (i.e. what ontological categories of linguistic objects we suppose to populate the empirical domain) and on being mathematically rigorous as to what structures are used to model them. Moreover, we

require that the theory itself actually count as a theory in the technical sense of precisely characterizing those modelling structures that are regarded as admissible or well-formed (i.e. corresponding to those imaginable linguistic objects that are actually predicted to be possible ones). This does not mean that the empirical hypotheses must be rendered in a formal logic as long as their content can be made clear and unambiguous in natural language (the same holds true in mathematical physics), but in principle they must be capable of being so rendered. (Pollard and Sag, 1994, p. 7)

In HPSG, the modelling domain—the analog of the physicist's flows—is a system of *sorted feature structures* . . . , that are intended to stand in a one-to-one relation with the types of natural language expressions and their subparts. The role of linguistic theory is to give a precise specification of which feature structures are to be considered admissible; the types of linguistic entities that correspond to the admissible feature structures constitute the predictions of the theory. (Pollard and Sag, 1994, p. 8)

Since our principle goal in this volume is to propose analyses of linguistic phenomena and make them intelligible to the linguistic community, we eschew extreme formalization here and thereby avoid the many technical design decisions associated with the problem of choosing a feature logic; our rules and principles, in the form of feature structure constraints, will be expressed—clearly and unambiguously, we hope—variously in English or in a quasi-formal description language that we turn to directly. We doubt that the relative informality of analyses such as those presented here is likely to impede understanding; if anything, probably the reverse is true. Nevertheless we consider formalization of the theory an important goal, indeed a necessary one for the proof of computational properties . . . (Pollard and Sag, 1994, p. 9)

In our concern for processing issues . . . we have accepted the conventional wisdom that linguistic theory must account for linguistic knowledge . . . but not necessarily for processes by which that knowledge is brought to bear in the case of individual linguistic tokens. Indeed, we take it to be the central goal of linguistic theory to characterize what it is that every linguistically mature human being knows by virtue of being a linguistic creature, namely, universal grammar. And a theory of a particular language—a grammar—characterizes what linguistic knowledge (beyond universal grammar) is

shared by the community of speakers of that language. Indeed, from the linguist's point of view, that is what language is.

But what does language consist of? One thing that it certainly does not consist of is individual linguistic events or utterance tokens, for knowledge of these is not what is shared among the members of a linguistic community. Instead, what is known in common, that makes communication possible, is the system of linguistic types. For example, the type of the sentence *I'm sleepy* is part of that system, but no individual token of it is. (Pollard and Sag, 1994, p. 14)

Modelling types of conceivable linguistic entities as rooted labelled graphs of a special kind—totally well-typed, sort-resolved feature structures—we formulate universal grammar and grammars of particular languages as systems of constraints on those feature structures. Only those feature structures that satisfy the constraints are taken to model (types of) grammatically well-formed linguistic entities. The distinction between the system of constraints and the collection of linguistic entities that satisfies it can be viewed as corresponding both to Chomsky's (1986a) distinction between *I-language* and *E-language* and to Saussure's ((1916) 1959) distinction

between *langue* and *parole*. Though only the latter is directly observable, only the former can be embodied as a mental computational system shared by the members of a linguistic community. (Pollard and Sag, 1994, pp. 57–58)

Some Properties of the Modelling Structures

- The first formal point to be noted is that the feature structures employed in HPSG are *sorted.* This means simply that each node is labelled with a *sort symbol* that tells what type of object the structure is modelling; that is, there is one sort symbol for each basic type (ontological category) of construct . . . The (finite) set of all sort symbols is assumed to be partially ordered . . . (Pollard and Sag, 1994, pp. 17–18)

- Second, we require feature structures to be *well-typed* . . . This means that what attribute (or, equivalently, feature) labels can appear in a feature structure are [*sic*] determined by its sort; this fact is the reflection within the model of the fact that what attributes (or, equivalently, features or components of structure) an empirical object has depends on its ontological category. (Pollard and Sag, 1994, p. 18)

- Third, feature structures that serve as models of linguistic entities are required to satisfy further criteria of completeness. Roughly what this means is that they are total (not merely partial) models of the objects that they represent. More precisely, they are required to be both *totally well-typed* . . . and *sort-resolved*. A feature structure is *totally* well-typed in case it is well-typed, and, moreover, for each node, every feature that is appropriate for the sort assigned to that node is actually present . . . A feature structure is *sort-resolved* provided every node is assigned a sort label that is maximal (i.e. most specific) in the sort ordering. (Pollard and Sag, 1994, p. 18)

Methodological Assumptions: Summary and Outlook

Principle among these [methodological assumptions] are

- a concern with analyses of particular phenomena that are worked out in detail;
- insistence on mathematical precision (at least in principle) with respect to the grammar formalism itself and its intended interpretation;
- and commitment to the criterion of empirical consequence, that is, the requirement that given a grammar and (a model of) a candidate linguistic expression, it is a determinate matter of fact whether or not the entity satisfies the grammar.

This last criterion is fundamental to the scientific method and at the same time provides a point of departure for meeting the more ambitious criterion of decidability . . . (after Pollard and Sag, 1994, p. 59)

II

Pollard 1999

Methodological Assumptions

Grammars exist in the real world, more specifically the part of the world inside language knower's minds; utterance tokens are events, also in the real world but external (at least in part) to minds. Formal linguists create idealizations of both these things.

- A formal grammar, a certain kind of logical theory, is a mathematical idealization of the mental grammar;
- and utterance tokens which are judged grammatical by native language knowers are idealized as certain mathematical structures, in such a way that distinct utterances which are linguistically indistinguishable, e.g., two utterances of *Poor John ran away*, are idealized as isomorphic structures.

What I think ties all these things together is that, given an appropriate scheme of semantic interpretation for formal grammars, and assuming it is a good grammar, the mathematical structures which idealize the grammatical utterance tokens will be certain kinds of *models* of the grammar. (Pollard, 1999, p. 281)

On the other hand, generative grammarians are accustomed to thinking of a formal grammar as *generating* a set of structures, called the *strong generative capacity* (SGC) of the grammar. This set is usually considered to have the following properties: (1) no two members are structurally isomorphic; and (2) if the grammar is making good predictions, then utterance tokens whose structures are isomorphic to members of the SGC will be judged grammatical. (Pollard, 1999, p. 282)

Now very roughly speaking, the relationship between an HPSG grammar and its SGC is analogous to the relationship between a first-order theory and its models. There are, however, some subtle differences . . . (Pollard, 1999, p. 284)

It might be expected that a model of the grammar . . . could be employed as the SGC of the grammar. Such a notion of the SGC has the virtue of mathematical simplicity, but it suffers from a number of defects. First, there can be many models of a given grammar, and these need not all be isomorphic. Second, some models are 'too small', in the sense that the natural language whose grammar we are trying to write might have some grammatical expressions that do not correspond

to any object of the model. And third, models may differ from each other in ways that make no linguistic difference, e.g., the number of objects in the model which correspond to the sentence *Poor John ran away*. There is a clear intuitive sense in which this should not matter linguistically; we want a notion of SGC which abstracts away from how many 'instances' or 'tokens' of a given expression there are, and instead is sensitive only [to] isomorphism classes. (Pollard, 1999, p. 288)

Important differences to Pollard and Sag 1994:

- No ontological commitment to types; 'type' is replaced with 'isomorphism classes' in the relevant contexts.
- Technically, the conventional feature structures of Pollard and Sag 1994 are replaced with certain canonical models. Although Pollard keeps referring to them as 'feature structures' (as can be justified by certain mathematical correspondences), they are not traditional feature structures.

III

King 1999

Criticism

Against Object Types

Almost all linguistic tokens are partly empirical. Linguistic communication would be almost impossible otherwise. But to what extent are linguistic types empirical? Is it an empirical matter whether a linguistic type is part of a natural language? . . .

The only possible basis for claiming that a linguistic type is part of a natural language is that the linguistic knowledge of an ordinary user of the natural language distinguishes that particular class of tokens. However, it is far from certain that linguistic knowledge *does* distinguish particular classes of linguistic tokens. And even if it does then the extent to which linguistic types are empirical cannot exceed the extent to which linguistic knowledge is empirical. Now, linguistic knowledge is certainly empirical in part . . . Nonetheless, the observation of linguistic knowledge itself is currently very limited and usually indirect. Thus, (Pollard and Sag, 1994) literally characterises truth in such a manner that the veracity of a HPSG grammar of a natural language is immune—possibly in principle, and certainly in current practice—to experimental assessment. HPSGians must either content themselves with writing untestable theories and hoping for the appropriate advances in cognitive technology, or characterise truth anew in a fashion that is susceptible to currently available means of empirical appraisal. (King, 1999, pp. 307–308)

In light of the wholly metempirical nature of linguistic types [the principle of only postulating constructs that correspond to observables in the empirical domain] entails that a characterisation of truth should avoid linguistic types unless to do so compromises the simplicity and elegance of HPSG grammars. Indeed, parsimony suggests that feature structures also be avoided where possible. (King, 1999, p. 310)

Against the Presentation of the Formalism in (Pollard and Sag, 1994)

Secondly, the (Pollard and Sag, 1994) characterisation is poorly defined. Consider feature structures. (Pollard and Sag, 1994) is clear about some aspects of feature structures, but not others. We are unequivocally told that feature structures are both totally well-typed and sort resolved. But whether, to use the terminology of (Moshier, 1988), a feature structure is *concrete*—an individual graph—or *abstract*—an isomorphism class

of graphs—is unclear.. . . Also, (Pollard and Sag, 1994) nowhere states whether feature structures are necessarily finite or possibly infinite. Finite feature structures certainly predominate in the feature-structure literature that (Pollard and Sag, 1994) cites, yet (Carpenter, 1992) clearly explicates the tensions that arise if feature structures are both totally well-typed and finite. . . (King, 1999, pp. 310–311)

Finally, King points out that Pollard and Sag (1994) do not define a formal description language, and that the literature they refer to does not provide the classical description language that the grammatical principles of Pollard and Sag presuppose. (King, 1999, pp. 311–312)

A final Concern

My third—and I believe insuperable—objection leads me to abandon any attempt to repair and extend the (Pollard and Sag, 1994) characterisation. A definition must be not only well-defined but also accurate, in the sense that the definition must faithfully and demonstrably capture the intuitive concept it is intended to express.. . . Presumably, the grammarian has intuitions about linguistic types and constructs their grammar in accord with those intuitions. However, the grammar is formally interpreted via the (Pollard and Sag, 1994) characterisation of truth. Thus, the grammar reflects the intuitions of the grammarian only to the extent that (i) the intuitions of the grammarian are commensurate with those that underly the (Pollard and Sag, 1994) characterisation, and (ii) the (Pollard and Sag, 1994) formal machinery faithfully expresses those underlying intuitions. My gravest concern over the (Pollard and Sag, 1994) characterisation is that (Pollard and Sag, 1994) (and (Pollard, 1999)) barely mentions what those underlying intuitions are, far less demonstrates that the formal machinery faithfully expresses them. Nor can we rely on a consensual intuition shared generally by linguists. The fundamental

ontological notion of the (Pollard and Sag, 1994) characterisation, the linguistic type, is an amorphous, artificial notion that linguists can and do construe in a variety of ways to suit their particular purposes. (King, 1999, p. 313)

King's Proposal

Motivated by his considerations of ontological simplicity, King suggests a construal of a natural language as a system of linguistic tokens. This construal is then used in a new, formal characterization of linguistic truth.

A token can be an object such as my computer, an event such as my typing these words at my computer keyboard, a state such as my being at work, a location such as the space occupied by my computer, or a complex such as the contents of the file system of my computer. . . The intuition here is that a grammar of a natural language should account for the linguistic knowledge of mature users of the natural language, but not for the biological, physical and psychological limitations the universe imposes upon the exercise of that knowledge. I share this intuition, but maintain that a scientifically worthwhile account of linguistic knowledge can presently only be given largely in terms of linguistic behaviour. In order to reconcile this apparent antithesis, I endorse the notion of nonactual tokens—tokens that could exist in some universe but happen not to in ours—and assume that the linguistic knowledge of each mature user of a natural language determines which

tokens—actual and nonactual—are tokens of the natural language. A grammar of a natural language is then true only if the grammar delimits the collection comprising all and only those tokens—actual and nonactual—that the linguistic knowledge of mature users of the natural language would deem tokens of the natural language. (King, 1999, p. 316)

A Logical Language for HPSG

RSRL

The Description Language I

Some examples of traditional HPSG principles:

- The HEAD FEATURE PRINCIPLE
  In a headed phrase, the values of SYNSEM | LOCAL | CATEGORY | HEAD *and* DAUGHTERS | HEAD-DAUGHTER | SYNSEM | LOCAL | CATEGORY | HEAD *are token-identical.*

- The SUBJECT CONDITION (Pollard and Sag, 1994, p. 400)
  If the initial element of the SUBCAT *value of a word is slashed, then so is some other element of that list.*

- The RELATIVE UNIQUENESS PRINCIPLE (Pollard and Sag, 1994, p. 400)
  For any phrase, a member of the set value of SYNSEM | NONLOCAL | INHERITED | REL *may belong to the value of that same path on at most one daughter.*

- Binding Theory, Principle B (Pollard and Sag, 1994, p. 401)
  A personal pronoun must be locally o-free.

A Principle from Abeillé and Godard 1996:

In the original notation:

$$\begin{bmatrix}\text{VFORM} & \textit{part-passé}\\ \text{S-ARG} & \langle \ldots \textit{aff\_r} \ldots \rangle\end{bmatrix} \rightarrow \begin{bmatrix}\text{V-AUX} & \textit{etre}\end{bmatrix}$$

A formalization in RSRL:

$$\left(\begin{array}{l}\exists \boxed{0}\, \exists \boxed{1} \\ \left(\begin{array}{l}\begin{bmatrix}\text{VFORM} & \textit{part-passé}\\ \text{S-ARG} & \boxed{1}\end{bmatrix} \\ \wedge\, \mathtt{member}\left(\boxed{0}\!\begin{bmatrix}\textit{aff\_r}\end{bmatrix}, \boxed{1}\right)\end{array}\right)\end{array}\right) \rightarrow \begin{bmatrix}\text{V-AUX} & \textit{etre}\end{bmatrix}$$

Signatures

Definition 1. Σ *is a* ***signature*** *iff*

Σ is a septuple $\langle \mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F}, \mathcal{R}, \mathcal{AR} \rangle$,

$\langle \mathcal{G}, \sqsubseteq \rangle$ is a partial order,

$$\mathcal{S} = \left\{ \sigma \in \mathcal{G} \,\middle|\, \begin{array}{l}\text{for each } \sigma' \in \mathcal{G},\\ \text{if } \sigma' \sqsubseteq \sigma \text{ then } \sigma = \sigma'\end{array} \right\},$$

$\mathcal{A}$ is a set,

$\mathcal{F}$ is a partial function from the Cartesian product of $\mathcal{G}$ and $\mathcal{A}$ to $\mathcal{G}$,

for each $\sigma_1 \in \mathcal{G}$, for each $\sigma_2 \in \mathcal{G}$ and for each $\alpha \in \mathcal{A}$,

if $\mathcal{F}\langle \sigma_1, \alpha \rangle$ is defined and $\sigma_2 \sqsubseteq \sigma_1$
then $\mathcal{F}\langle \sigma_2, \alpha \rangle$ is defined and $\mathcal{F}\langle \sigma_2, \alpha \rangle \sqsubseteq \mathcal{F}\langle \sigma_1, \alpha \rangle$,

$\mathcal{R}$ is a finite set, and

$\mathcal{AR}$ is a total function from $\mathcal{R}$ to $\mathbb{N}^+$.

Interpretations

Definition 2. *For each signature* $\Sigma = \langle \mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F}, \mathcal{R}, \mathcal{AR} \rangle$, I *is a* Σ ***interpretation*** *iff*

I is a quadruple $\langle \mathsf{U}, \mathsf{S}, \mathsf{A}, \mathsf{R} \rangle$,

U is a set,

S is a total function from U to $\mathcal{S}$,

A is a total function from $\mathcal{A}$ to the set of partial functions from U to U,

for each $\alpha \in \mathcal{A}$ and each $u \in \mathsf{U}$,

if $\mathsf{A}(\alpha)(u)$ is defined
then $\mathcal{F}\langle \mathsf{S}(u), \alpha \rangle$ is defined, and $\mathsf{S}(\mathsf{A}(\alpha)(u)) \sqsubseteq \mathcal{F}\langle \mathsf{S}(u), \alpha \rangle$, and

for each $\alpha \in \mathcal{A}$ and each $u \in \mathsf{U}$,

if $\mathcal{F}\langle \mathsf{S}(u), \alpha \rangle$ is defined then $\mathsf{A}(\alpha)(u)$ is defined,

R is a total function from $\mathcal{R}$ to the power set of $\bigcup\limits_{n \in \mathbb{N}} \overline{\mathsf{U}}^n$, and

for each $\rho \in \mathcal{R}$, $\mathsf{R}(\rho) \subseteq \overline{\mathsf{U}}^{\mathcal{AR}(\rho)}$.

Chains

Definition 3. *For each signature* $\Sigma = \langle \mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F}, \mathcal{R}, \mathcal{AR} \rangle$,

$$\widehat{\mathcal{G}} = \mathcal{G} \cup \{chain, echain, nechain, metatop\},$$

$$\begin{array}{rl}\widehat{\sqsubseteq} = \sqsubseteq \cup & \{\langle echain, chain \rangle, \langle nechain, chain \rangle\} \\ & \cup \left\{ \langle \sigma, \sigma \rangle \,\middle|\, \sigma \in \widehat{\mathcal{G}} \backslash \mathcal{G} \right\} \\ & \cup \left\{ \langle \sigma, metatop \rangle \,\middle|\, \sigma \in \widehat{\mathcal{G}} \right\},\end{array}$$

$\widehat{\mathcal{S}} = \mathcal{S} \cup \{echain, nechain\}$, *and*

$\widehat{\mathcal{A}} = \mathcal{A} \cup \{\dagger, \triangleright\}$.

Definition 4. *For each signature* $\Sigma = \langle \mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F}, \mathcal{R}, \mathcal{AR} \rangle$, *for each* Σ *interpretation* $\mathsf{I} = \langle \mathsf{U}, \mathsf{S}, \mathsf{A}, \mathsf{R} \rangle$,

$\widehat{\mathsf{S}}$ *is the total function from* $\overline{\mathsf{U}}$ *to* $\widehat{\mathcal{S}}$ *such that*

for each $u \in \mathsf{U}$, $\widehat{\mathsf{S}}(u) = \mathsf{S}(u)$,
for each $u_1 \in \mathsf{U}$, ..., *for each* $u_n \in \mathsf{U}$,
$\widehat{\mathsf{S}}(\langle u_1, \ldots, u_n \rangle) = \begin{cases} echain & \text{if } n = 0, \\ nechain & \text{if } n > 0 \end{cases}$, *and*

$\widehat{\mathsf{A}}$ *is the total function from* $\widehat{\mathcal{A}}$ *to the set of partial functions from* $\overline{\mathsf{U}}$ *to* $\overline{\mathsf{U}}$ *such that*

for each $\alpha \in \mathcal{A}$, $\widehat{\mathsf{A}}(\alpha) = \mathsf{A}(\alpha)$,
$\widehat{\mathsf{A}}(\dagger)$ *is the total function from* U^+ *to* U *such that for each* $\langle u_0, \ldots, u_n \rangle \in \mathsf{U}^+$,
$\widehat{\mathsf{A}}(\dagger)(\langle u_0, \ldots, u_n \rangle) = u_0$, *and*
$\widehat{\mathsf{A}}(\triangleright)$ *is the total function from* U^+ *to* U^* *such that for each* $\langle u_0, \ldots, u_n \rangle \in \mathsf{U}^+$,
$\widehat{\mathsf{A}}(\triangleright)(\langle u_0, \ldots, u_n \rangle) = \langle u_1, \ldots, u_n \rangle$.

Formulae

Definition 5. *For each signature* Σ, $\mathcal{T}^\Sigma$ *and* $\mathcal{D}^\Sigma$ *are the smallest sets such that*

$: \in \mathcal{T}^\Sigma$,

for each $v \in \mathcal{VAR}$, $v \in \mathcal{T}^\Sigma$,

for each $\alpha \in \widehat{\mathcal{A}}$ *and each* $\tau \in \mathcal{T}^\Sigma$, $\tau\alpha \in \mathcal{T}^\Sigma$,

for each $\sigma \in \widehat{\mathcal{G}}$, *for each* $\tau \in \mathcal{T}^\Sigma$, $\tau \sim \sigma \in \mathcal{D}^\Sigma$,

for each $\tau_1 \in \mathcal{T}^\Sigma$, *for each* $\tau_2 \in \mathcal{T}^\Sigma$, $\tau_1 \approx \tau_2 \in \mathcal{D}^\Sigma$,

for each $\rho \in \mathcal{R}$, *for each* $x_1 \in \mathcal{VAR}$, ..., *for each* $x_{\mathcal{AR}(\rho)} \in \mathcal{VAR}$,

$\rho(x_1, \ldots, x_{\mathcal{AR}(\rho)}) \in \mathcal{D}^\Sigma$,

for each $x \in \mathcal{VAR}$, *for each* $\delta \in \mathcal{D}^\Sigma$, $\exists x\, \delta \in \mathcal{D}^\Sigma$,

for each $x \in \mathcal{VAR}$, *for each* $\delta \in \mathcal{D}^\Sigma$, $\forall x\, \delta \in \mathcal{D}^\Sigma$,

for each $\delta \in \mathcal{D}^\Sigma$, $\neg\delta \in \mathcal{D}^\Sigma$,

for each $\delta_1 \in \mathcal{D}^\Sigma$, *for each* $\delta_2 \in \mathcal{D}^\Sigma$, $[\delta_1 \wedge \delta_2] \in \mathcal{D}^\Sigma$,

for each $\delta_1 \in \mathcal{D}^\Sigma$, *for each* $\delta_2 \in \mathcal{D}^\Sigma$, $[\delta_1 \vee \delta_2] \in \mathcal{D}^\Sigma$.

The Description Language II

- The TRACE PRINCIPLE (Pollard and Sag, 1994, p. 400)
  The SYNSEM *value of any trace must be a (noninitial) member of the* SUBCAT *list of a substantive word.*

- The CONTROL THEORY (Pollard and Sag, 1994, p. 401)
  (looks for potential controlled *local* objects)
  If the SOA-ARG *value of a control-qfpsoa is token-identical with the* CONTENT *value of a local entity whose* CATEGORY | SUBCAT *value is a list of length one, then the member of that list is (1) reflexive, and (2) coindexed with the* INFLUENCED *(respectively,* COMMITTOR, EXPERIENCER*) value of the control-qfpsoa if the latter is of sort influence (respectively, commitment, orientation).*

- BINDING THEORY
  (looks for potential o-commanders)

- QUANTIFIER BINDING CONDITION (Pollard and Sag, 1994, p. 327)
  Given a quantifier contained within a CONTENT *value, every occurrence within that* CONTENT *value of the quantifier's index must be captured by the quantifier.*

Formula Denotation

Definition 6. *For each signature Σ, for each Σ interpretation $\mathsf{I} = \langle \mathsf{U}, \mathsf{S}, \mathsf{A}, \mathsf{R} \rangle$,*

$Ass_{\mathsf{I}} = \overline{\mathsf{U}}^{\mathcal{VAR}}$ *is the* ***set of variable assignments in*** I.

Definition 7. *For each signature $\Sigma = \langle \mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F}, \mathcal{R}, \mathcal{AR} \rangle$, for each Σ interpretation $\mathsf{I} = \langle \mathsf{U}, \mathsf{S}, \mathsf{A}, \mathsf{R} \rangle$, for each $ass \in Ass_{\mathsf{I}}$, T_{I}^{ass} is the total function from $\mathcal{T}^{\Sigma}$ to the set of partial functions from U to $\overline{\mathsf{U}}$ such that for each $u \in \mathsf{U}$,*

$T_{\mathsf{I}}^{ass}(:)(u)$ is defined and $T_{\mathsf{I}}^{ass}(:)(u) = u$,

for each $v \in \mathcal{VAR}$, $T_{\mathsf{I}}^{ass}(v)(u)$ is defined and $T_{\mathsf{I}}^{ass}(v)(u) = ass(v)$,

for each $\tau \in \mathcal{T}^{\Sigma}$, for each $\alpha \in \widehat{\mathcal{A}}$,

$T_{\mathsf{I}}^{ass}(\tau\alpha)(u)$ is defined
iff $T_{\mathsf{I}}^{ass}(\tau)(u)$ is defined and $\widehat{\mathsf{A}}(\alpha)(T_{\mathsf{I}}^{ass}(\tau)(u))$ is defined, and
if $T_{\mathsf{I}}^{ass}(\tau\alpha)(u)$ is defined then $T_{\mathsf{I}}^{ass}(\tau\alpha)(u) = \widehat{\mathsf{A}}(\alpha)(T_{\mathsf{I}}^{ass}(\tau)(u))$.

Definition 8. *For each signature $\Sigma = \langle \mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F}, \mathcal{R}, \mathcal{AR} \rangle$, for each Σ interpretation $\mathsf{I} = \langle \mathsf{U}, \mathsf{S}, \mathsf{A}, \mathsf{R} \rangle$, and for each $u \in \mathsf{U}$,*

$$\mathsf{Co}_{\mathsf{I}}^{u} = \left\{ u' \in \mathsf{U} \;\middle|\; \begin{array}{l} \text{for some } ass \in Ass_{\mathsf{I}}, \\ \text{for some } \pi \in \mathcal{A}^*, \\ \quad T_{\mathsf{I}}^{ass}(:\pi)(u) \text{ is defined, and} \\ \quad u' = T_{\mathsf{I}}^{ass}(:\pi)(u) \end{array} \right\}.$$

Definition 9. *For each signature Σ, for each Σ interpretation $\mathsf{I} = \langle \mathsf{U}, \mathsf{S}, \mathsf{A}, \mathsf{R} \rangle$, for each $ass \in Ass_{\mathsf{I}}$, for each $v \in \mathcal{VAR}$, for each $w \in \mathcal{VAR}$, for each $u \in \overline{\mathsf{U}}$,*

$$ass\tfrac{u}{v}(w) = \begin{cases} u & \text{if } v = w \\ ass(w) & \text{otherwise.} \end{cases}$$

Definition 10. *For each signature $\Sigma = \langle \mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F}, \mathcal{R}, \mathcal{AR} \rangle$, for each Σ interpretation $\mathsf{I} = \langle \mathsf{U}, \mathsf{S}, \mathsf{A}, \mathsf{R} \rangle$, for each $ass \in Ass_{\mathsf{I}}$, D_{I}^{ass} is the total function from $\mathcal{D}^{\Sigma}$ to the power set of U such that*

for each $\tau \in \mathcal{T}^{\Sigma}$, for each $\sigma \in \widehat{\mathcal{G}}$,

$$D_{\mathsf{I}}^{ass}(\tau \sim \sigma) = \left\{ u \in \mathsf{U} \;\middle|\; \begin{array}{l} T_{\mathsf{I}}^{ass}(\tau)(u) \text{ is defined, and} \\ \widehat{\mathsf{S}}(T_{\mathsf{I}}^{ass}(\tau)(u)) \mathrel{\widehat{\sqsubseteq}} \sigma \end{array} \right\},$$

for each $\tau_1 \in \mathcal{T}^{\Sigma}$, for each $\tau_2 \in \mathcal{T}^{\Sigma}$,

$$D_{\mathsf{I}}^{ass}(\tau_1 \approx \tau_2) = \left\{ u \in \mathsf{U} \;\middle|\; \begin{array}{l} T_{\mathsf{I}}^{ass}(\tau_1)(u) \text{ is defined,} \\ T_{\mathsf{I}}^{ass}(\tau_2)(u) \text{ is defined, and} \\ T_{\mathsf{I}}^{ass}(\tau_1)(u) = T_{\mathsf{I}}^{ass}(\tau_2)(u) \end{array} \right\},$$

for each $\rho \in \mathcal{R}$, for each $x_1 \in \mathcal{VAR}$, ..., for each $x_{\mathcal{AR}(\rho)} \in \mathcal{VAR}$,

$$D_{\mathsf{I}}^{ass}(\rho(x_1, \ldots, x_{\mathcal{AR}(\rho)})) = \left\{ u \in \mathsf{U} \;\middle|\; \left\langle ass(x_1), \ldots, ass(x_{\mathcal{AR}(\rho)}) \right\rangle \in \mathsf{R}(\rho) \right\},$$

for each $v \in \mathcal{VAR}$, for each $\delta \in \mathcal{D}^{\Sigma}$,

$$D_{\mathsf{I}}^{ass}(\exists v\, \delta) = \left\{ u \in \mathsf{U} \;\middle|\; \begin{array}{l} \text{for some } u' \in \overline{\mathsf{Co}_{\mathsf{I}}^{u}}, \\ u \in D_{\mathsf{I}}^{ass\frac{u'}{v}}(\delta) \end{array} \right\},$$

for each $v \in \mathcal{VAR}$, for each $\delta \in \mathcal{D}^{\Sigma}$,

$$D_{\mathsf{I}}^{ass}(\forall v\, \delta) = \left\{ u \in \mathsf{U} \;\middle|\; \begin{array}{l} \text{for each } u' \in \overline{\mathsf{Co}_{\mathsf{I}}^{u}}, \\ u \in D_{\mathsf{I}}^{ass\frac{u'}{v}}(\delta) \end{array} \right\},$$

for each $\delta \in \mathcal{D}^{\Sigma}$,

$$D_{\mathsf{I}}^{ass}(\neg\delta) = \mathsf{U} \backslash D_{\mathsf{I}}^{ass}(\delta),$$

for each $\delta_1 \in \mathcal{D}^{\Sigma}$, for each $\delta_2 \in \mathcal{D}^{\Sigma}$,

$$D_{\mathsf{I}}^{ass}([\delta_1 \wedge \delta_2]) = D_{\mathsf{I}}^{ass}(\delta_1) \cap D_{\mathsf{I}}^{ass}(\delta_2),$$

for each $\delta_1 \in \mathcal{D}^{\Sigma}$, for each $\delta_2 \in \mathcal{D}^{\Sigma}$,

$$D_{\mathsf{I}}^{ass}([\delta_1 \vee \delta_2]) = D_{\mathsf{I}}^{ass}(\delta_1) \cup D_{\mathsf{I}}^{ass}(\delta_2), \text{ and}$$

for each $\delta_1 \in \mathcal{D}^{\Sigma}$, for each $\delta_2 \in \mathcal{D}^{\Sigma}$,

$$D_{\mathsf{I}}^{ass}([\delta_1 \rightarrow \delta_2]) = (\mathsf{U} \backslash D_{\mathsf{I}}^{ass}(\delta)) \cup D_{\mathsf{I}}^{ass}(\delta_2).$$

From Formulae to Descriptions

Definition 11. *For each signature* $\Sigma = \langle \mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F}, \mathcal{R}, \mathcal{AR} \rangle$,

$FV(:) = \emptyset$,

for each $v \in \mathcal{VAR}$, $FV(v) = \{v\}$,

for each $\tau \in \mathcal{T}^{\Sigma}$, *for each* $\alpha \in \widehat{\mathcal{A}}$, $FV(\tau\alpha) = FV(\tau)$,

for each $\tau \in \mathcal{T}^{\Sigma}$, *for each* $\sigma \in \widehat{\mathcal{G}}$, $FV(\tau \sim \sigma) = FV(\tau)$,

for each $\tau_1, \tau_2 \in \mathcal{T}^{\Sigma}$, $FV(\tau_1 \approx \tau_2) = FV(\tau_1) \cup FV(\tau_2)$,

for each $\rho \in \mathcal{R}$, *for each* $x_1, \ldots, x_{\mathcal{AR}(\rho)} \in \mathcal{VAR}$,

$FV(\rho(x_1, \ldots, x_{\mathcal{AR}(\rho)})) = \{x_1, \ldots, x_{\mathcal{AR}(\rho)}\}$,

for each $\delta \in \mathcal{D}^{\Sigma}$, *for each* $v \in \mathcal{VAR}$, $FV(\exists v\ \delta) = FV(\delta)\backslash\{v\}$,

for each $\delta \in \mathcal{D}^{\Sigma}$, *for each* $v \in \mathcal{VAR}$, $FV(\forall v\ \delta) = FV(\delta)\backslash\{v\}$,

for each $\delta \in \mathcal{D}^{\Sigma}$, $FV(\neg\delta) = FV(\delta)$,

for each $\delta_1 \in \mathcal{D}^{\Sigma}$, *for each* $\delta_2 \in \mathcal{D}^{\Sigma}$,

$FV([\delta_1 \wedge \delta_2]) = FV(\delta_1) \cup FV(\delta_2)$,

for each $\delta_1 \in \mathcal{D}^{\Sigma}$, *for each* $\delta_2 \in \mathcal{D}^{\Sigma}$,

$FV([\delta_1 \vee \delta_2]) = FV(\delta_1) \cup FV(\delta_2)$, *and*

for each $\delta_1 \in \mathcal{D}^{\Sigma}$, *for each* $\delta_2 \in \mathcal{D}^{\Sigma}$,

$FV([\delta_1 \rightarrow \delta_2]) = FV(\delta_1) \cup FV(\delta_2)$.

Proposition 1. *For each signature* Σ, *for each* Σ *interpretation* I, *for each* $ass_1 \in Ass_{\mathsf{I}}$, *for each* $ass_2 \in Ass_{\mathsf{I}}$,

for each $\tau \in \mathcal{T}^{\Sigma}$,

if for each $v \in FV(\tau)$, $ass_1(v) = ass_2(v)$
then $T_{\mathsf{I}}^{ass_1}(\tau) = T_{\mathsf{I}}^{ass_2}(\tau)$, *and*

for each $\delta \in \mathcal{D}^{\Sigma}$,

if for each $v \in FV(\delta)$, $ass_1(v) = ass_2(v)$
then $D_{\mathsf{I}}^{ass_1}(\delta) = D_{\mathsf{I}}^{ass_2}(\delta)$.

Definition 12. *For each signature* Σ,

$$\mathcal{D}_0^{\Sigma} = \{\delta \in \mathcal{D}^{\Sigma} \mid FV(\delta) = \emptyset\}.$$

Corollary 1. *For each signature* Σ, *for each* $\delta \in \mathcal{D}_0^{\Sigma}$, *for each* Σ *interpretation* I, *for each* $ass_1 \in Ass_{\mathsf{I}}$, *for each* $ass_2 \in Ass_{\mathsf{I}}$,

$$D_{\mathsf{I}}^{ass_1}(\delta) = D_{\mathsf{I}}^{ass_2}(\delta).$$

Description Denotation and Theory Denotation

Definition 13. *For each signature* Σ, *for each* Σ *interpretation* $\mathsf{I} = \langle \mathsf{U}, \mathsf{S}, \mathsf{A}, \mathsf{R} \rangle$, D_{I} *is the total function from* $\mathcal{D}_0^{\Sigma}$ *to the power set of* U *such that for each* $\delta \in \mathcal{D}_0^{\Sigma}$,

$$D_{\mathsf{I}}(\delta) = \left\{ u \in \mathsf{U} \;\middle|\; \begin{array}{l} \text{for each } ass \in Ass_{\mathsf{I}}, \\ u \in D_{\mathsf{I}}^{ass}(\delta) \end{array} \right\}.$$

Definition 14. *For each signature* Σ, *for each* Σ *interpretation* $\mathsf{I} = \langle \mathsf{U}, \mathsf{S}, \mathsf{A}, \mathsf{R} \rangle$, Θ_{I} *is the total function from the power set of* $\mathcal{D}_0^{\Sigma}$ *to the power set of* U *such that for each* $\theta \subseteq \mathcal{D}_0^{\Sigma}$,

$$\Theta_{\mathsf{I}}(\theta) = \left\{ u \in \mathsf{U} \;\middle|\; \begin{array}{l} \text{for each } \delta \in \theta, \\ u \in D_{\mathsf{I}}(\delta) \end{array} \right\}.$$

Grammars and Models

Definition 15. *Γ is a* ***grammar*** *iff*

Γ is a pair $\langle \Sigma, \theta \rangle$,

Σ is a signature, and

$\theta \subseteq \mathcal{D}_0^{\Sigma}$.

Definition 16. *For each grammar $\Gamma = \langle \Sigma, \theta \rangle$, for each Σ interpretation $\mathsf{I} = \langle \mathsf{U}, \mathsf{S}, \mathsf{A}, \mathsf{R} \rangle$,*

I is a Γ ***model*** *iff $\Theta_{\mathsf{I}}(\theta) = \mathsf{U}$.*

King 1999 in RSRL

An Example

A Signature:

top

car OWNER *person*
DRIVER *person*

vw

bmw

person LIKES-BEST *top*

man

woman

$\mathcal{R} = \{\}$, $\mathcal{AR} = \{\}$

An interpretation of our signature:
Let $\mathsf{I} = \langle \mathsf{U}, \mathsf{S}, \mathsf{A}, \mathsf{R} \rangle$, with:

$$\mathsf{U} = \left\{ \begin{array}{l} \text{Anke, Anke's husband, Detmar, Kordula,} \\ \text{first car, second car} \end{array} \right\}$$

S(Anke) = $woman$,
S(Kordula) = $woman$,
S(Anke's husband) = man,
S(Detmar) = man,
S(first car) = vw, and
S(second car) = vw
A(LIKES-BEST)(Anke) = Anke's husband,
A(LIKES-BEST)(Anke's husband) = Anke,
A(LIKES-BEST)(Detmar) = Kordula,
A(LIKES-BEST)(Kordula) = Detmar,
A(OWNER)(first car) = Kordula,
A(OWNER)(second car) = Anke,
A(DRIVER)(first car) = Detmar,
A(DRIVER)(second car) = Anke,
A(LIKES-BEST) is undefined on the first car and on the second car,
A(OWNER) is undefined on Anke, Anke's husband, Detmar and Kordula, and
A(DRIVER) is likewise undefined on Anke, Anke's husband, Detmar and Kordula, and
R = $\{\}$.

Grammars of I:

Assume $\theta_1 = \left\{ \begin{array}{l} [:\sim bmw \rightarrow \neg :\approx:], \\ [:\sim vw \rightarrow :\text{OWNER} \sim woman] \end{array} \right\}$.

Observation:
I is a model of $\langle \Sigma, \theta_1 \rangle$. However, $\langle \Sigma, \theta_1 \rangle$ overlicenses I. I_2 is a relevant example:

Let $\mathsf{I}_2 = \langle \mathsf{U}, \mathsf{S}, \mathsf{A}, \mathsf{R} \rangle$, with:
U = {Anne, Manfred, Maria, Mike, third car}
S(Anne) = $woman$,
S(Maria) = $woman$,
S(Manfred) = man,
S(Mike) = man,
S(third car) = vw, and
A(LIKES-BEST)(Anne) = Manfred,
A(LIKES-BEST)(Manfred) = Anne,
A(LIKES-BEST)(Maria) = Mike,
A(LIKES-BEST)(Mike) = Maria,
A(OWNER)(third car) = Anne,
A(DRIVER)(third car) = Mike,
A(LIKES-BEST) is undefined on the third car,
A(OWNER) is undefined on Anne, Manfred, Maria and Mike, and

A(DRIVER) is likewise undefined on Anne, Manfred, Maria and Mike, and
R = {}.

Now assume

$$\theta_2 = \left\{ \begin{array}{l} [:\sim woman \leftrightarrow [:\text{LIKES-BEST} \sim man \wedge :\text{LIKES-BEST LIKES-BEST} \approx:]], \\ [:\sim man \leftrightarrow [:\text{LIKES-BEST} \sim woman \wedge :\text{LIKES-BEST LIKES-BEST} \approx:]], \\ [:\sim bmw \rightarrow \neg :\approx:], \\ [:\sim vw \rightarrow [:\text{OWNER} \sim woman \wedge :\text{OWNER} \approx: \text{DRIVER}]] \end{array} \right\}.$$

Problem:
$\langle \Sigma, \theta_2 \rangle$ underlicenses I. I is not even a model of $\langle \Sigma, \theta_2 \rangle$.

Finally, consider

$$\theta_3 = \left\{ \begin{array}{l} [:\sim woman \leftrightarrow [:\text{LIKES-BEST} \sim man \wedge :\text{LIKES-BEST LIKES-BEST} \approx:]], \\ [:\sim man \leftrightarrow [:\text{LIKES-BEST} \sim woman \wedge :\text{LIKES-BEST LIKES-BEST} \approx:]], \\ [:\sim bmw \rightarrow \neg :\approx:], \\ \left[:\sim vw \rightarrow \left[\begin{array}{l} :\text{OWNER} \sim woman \wedge \\ [:\text{OWNER} \approx: \text{DRIVER} \vee :\text{DRIVER} \approx: \text{OWNER LIKES-BEST}] \end{array} \right] \right] \end{array} \right\}.$$

Observation:
I is an exhaustive model of $\langle \Sigma, \theta_3 \rangle$.

Terminology

Definition 17. *For each signature* $\Sigma = \langle \mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F}, \mathcal{R}, \mathcal{AR} \rangle$*, for each* Σ *interpretation* $\mathsf{I} = \langle \mathsf{U}, \mathsf{S}, \mathsf{A}, \mathsf{R} \rangle$*, for each* $u \in \mathsf{I}$*,*

$\mathsf{I}_u = \langle \mathsf{U}_u, \mathsf{S}_u, \mathsf{A}_u, \mathsf{R}_u \rangle$ *is the* Σ ***interpretation under*** u ***in*** I *iff*

$\mathsf{U}_u = \left\{ u' \in \mathsf{U} \,\middle|\, u' \in \mathsf{Co}^u_{\mathsf{I}} \right\}$,
$\mathsf{S}_u = \mathsf{S} \cap (\mathsf{U}_u \times \mathcal{S})$,
$\mathsf{A}_u = \mathsf{A} \cap (\mathcal{A} \times \{\mathsf{U}_u \times \mathsf{U}_u\})$,
$\mathsf{R}_u = \mathsf{R} \cap \left(\mathcal{R} \times Pow \left(\bigcup_{n \in \mathbb{N}} \left(\overline{\mathsf{U}}_u\right)^n \right) \right)$*, and*
$\mathsf{I}_u = \langle \mathsf{U}_u, \mathsf{S}_u, \mathsf{A}_u, \mathsf{R}_u \rangle$.

Definition 18. *For each signature* $\Sigma = \langle \mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F}, \mathcal{R}, \mathcal{AR} \rangle$*, for each* Σ *interpretation* $\mathsf{I} = \langle \mathsf{U}, \mathsf{S}, \mathsf{A}, \mathsf{R} \rangle$*, for each* $u \in \mathsf{I}$*,*

$\langle u, \mathsf{I}_u \rangle$ *is the* ***configuration of entities under*** u ***in*** I *iff*

I_u *is the* Σ *interpretation under* u *in* I.

Definition 19. *For each signature* Σ*, for each* Σ *interpretation* $\mathsf{I}_1 = \langle \mathsf{U}_1, \mathsf{S}_1, \mathsf{A}_1, \mathsf{R}_1 \rangle$*, for each* $u_1 \in \mathsf{U}_1$*, for each* Σ *interpretation* $\mathsf{I}_2 = \langle \mathsf{U}_2, \mathsf{S}_2, \mathsf{A}_2, \mathsf{R}_2 \rangle$*, for each* $u_2 \in \mathsf{U}_2$*,*

f *is a* ***congruence*** *from* $\langle u_1, \mathsf{I}_1 \rangle$ *to* $\langle u_2, \mathsf{I}_2 \rangle$ *in* Σ *iff*

f *is a bijection from* $\overline{\mathsf{Co}^{u_1}_{\mathsf{I}_1}}$ *to* $\overline{\mathsf{Co}^{u_2}_{\mathsf{I}_2}}$*,*
for each $u \in \overline{\mathsf{Co}^{u_1}_{\mathsf{I}_1}}$*,* $\widehat{\mathsf{S}_1}(u) = \widehat{\mathsf{S}_2}(f(u))$*,*
for each $\alpha \in \widehat{\mathcal{A}}$*, for each* $u \in \overline{\mathsf{Co}^{u_1}_{\mathsf{I}_1}}$*,*
$\widehat{\mathsf{A}_1}(\alpha)(u)$ *is defined iff* $\widehat{\mathsf{A}_2}(\alpha)(f(u))$ *is defined, and*
if $\widehat{\mathsf{A}_1}(\alpha)(u)$ *is defined then* $f(\widehat{\mathsf{A}_1}(\alpha)(u)) = \widehat{\mathsf{A}_2}(\alpha)(f(u))$*,*
for each $\rho \in \mathcal{R}$*, for each* $o_1 \in \overline{\mathsf{Co}^{u_1}_{\mathsf{I}_1}}$*, ..., for each* $o_{\mathcal{AR}(\rho)} \in \overline{\mathsf{Co}^{u_1}_{\mathsf{I}_1}}$*,*
$\langle o_1, \ldots, o_{\mathcal{AR}(\rho)} \rangle \in \mathsf{R}_1(\rho)$
iff $\langle f(o_1), \ldots, f(o_{\mathcal{AR}(\rho)}) \rangle \in \mathsf{R}_2(\rho)$*, and*
$f(u_1) = u_2$.

Definition 20. *For each signature Σ, for each Σ interpretation $\mathsf{I}_1 = \langle \mathsf{U}_1, \mathsf{S}_1, \mathsf{A}_1, \mathsf{R}_1 \rangle$, for each $u_1 \in \mathsf{U}_1$, for each Σ interpretation $\mathsf{I}_2 = \langle \mathsf{U}_2, \mathsf{S}_2, \mathsf{A}_2, \mathsf{R}_2 \rangle$, for each $u_2 \in \mathsf{U}_2$,*

*$\langle u_1, \mathsf{I}_1 \rangle$ and $\langle u_2, \mathsf{I}_2 \rangle$ are **congruent in** Σ iff for some f, f is a congruence from $\langle u_1, \mathsf{I}_1 \rangle$ to $\langle u_2, \mathsf{I}_2 \rangle$ in Σ.*

Linguistic Truth in King 1999

The three conditions of King 1999, p. 343, can now straightforwardly be cast in terms of RSRL. The only difference to the original three conditions is that I am claiming that natural languages include relations between components (and chains of components) of tokens. A grammar $\langle \Sigma, \theta \rangle$ is true of a natural language only if

1. the natural language can be construed as an interpretation, I, of the signature Σ, where I is a system of linguistic tokens,
2. each description in θ is true of each entity in the natural language, i.e., the natural language, I, is a $\langle \Sigma, \theta \rangle$ model, and
3. some description in θ is false of some component of each entity, u', in any Σ interpretation, I$'$, for which no entity u in the natural language, I, has isomorphically configured components.

Except for the reference to the extended definitions of all relevant terms, the conditions 1–3 are identical to

King's conditions on SRL grammars. The third condition amounts to saying that the language that is described by a grammar is in the class of exhaustive models of the grammar. Since we know that for each grammar, there exists an exhaustive model, it follows that King's interpretation of the meaning of a grammar is fully compatible with RSRL.

Exhaustive Models

Based on the notion of congruence, we can give an algebraic characterization of exhaustive models:

Definition 21. *For each signature Σ, for each Σ interpretation $\mathsf{I}_1 = \langle \mathsf{U}_1, \mathsf{S}_1, \mathsf{A}_1, \mathsf{R}_1 \rangle$, for each Σ interpretation $\mathsf{I}_2 = \langle \mathsf{U}_2, \mathsf{S}_2, \mathsf{A}_2, \mathsf{R}_2 \rangle$,*

*I_1 **simulates** I_2 **in** Σ iff for each $u_2 \in \mathsf{U}_2$, for some $u_1 \in \mathsf{U}_1$, $\langle u_1, \mathsf{I}_1 \rangle$ and $\langle u_2, \mathsf{I}_2 \rangle$ are congruent in Σ.*

Definition 22. *For each signature Σ, for each $\theta \subseteq \mathcal{D}_0^{\Sigma}$, for each Σ interpretation I,*

*I is an **exhaustive** $\langle \Sigma, \theta \rangle$ **model** iff*

I is a $\langle \Sigma, \theta \rangle$ model, and
for each Σ interpretation I$'$,
if I$'$ is a $\langle \Sigma, \theta \rangle$ model then I simulates I$'$ in Σ.

A simpler characterization of exhaustive models can be based on description denotations alone:

Definition 23. *For each signature Σ, for each Σ interpretation $\mathsf{I}_1 = \langle \mathsf{U}_1, \mathsf{S}_1, \mathsf{A}_1, \mathsf{R}_1 \rangle$, for each $u_1 \in \mathsf{U}_1$, for each Σ interpretation $\mathsf{I}_2 = \langle \mathsf{U}_2, \mathsf{S}_2, \mathsf{A}_2, \mathsf{R}_2 \rangle$, for each $u_2 \in \mathsf{U}_2$,*

$\langle u_1, \mathsf{I}_1 \rangle$ and $\langle u_2, \mathsf{I}_2 \rangle$ are **indiscernible in** Σ
iff for each $\delta \in \mathcal{D}_0^\Sigma$, $u_1 \in D_{\mathsf{I}_1}(\delta)$ iff $u_2 \in D_{\mathsf{I}_2}(\delta)$.

Indiscernibility and congruence turn out to be equivalent:

Proposition 2. *For each signature Σ, for each Σ interpretation $\mathsf{I}_1 = \langle \mathsf{U}_1, \mathsf{S}_1, \mathsf{A}_1, \mathsf{R}_1 \rangle$, for each $o_1 \in \mathsf{U}_1$, for each Σ interpretation $\mathsf{I}_2 = \langle \mathsf{U}_2, \mathsf{S}_2, \mathsf{A}_2, \mathsf{R}_2 \rangle$, for each $o_2 \in \mathsf{U}_2$,*

$\langle o_1, \mathsf{I}_1 \rangle$ and $\langle o_2, \mathsf{I}_2 \rangle$ are congruent in Σ iff $\langle o_1, \mathsf{I}_1 \rangle$ and $\langle o_2, \mathsf{I}_2 \rangle$ are indiscernible in Σ.

Since indiscernibility and congruence are equivalent, the simulation of models of grammars can alternatively be expressed in terms of the denotation of sets of descriptions in interpretations of the same signature.

One Σ model I of a grammar, $\langle \Sigma, \theta \rangle$, simulates another Σ model I' of the grammar in Σ only if each Σ theory that describes something in I' also describes something in I.

Proposition 3. *For each signature Σ, for each $\theta \subseteq \mathcal{D}_0^\Sigma$, for each Σ interpretation I,*

for each Σ interpretation I',

if I' is a $\langle \Sigma, \theta \rangle$ model then I simulates I' in Σ

iff for each $\theta' \subseteq \mathcal{D}_0^\Sigma$, for each Σ interpretation I',

if I' is a $\langle \Sigma, \theta \rangle$ model and $\Theta_{\mathsf{I}'}(\theta') \neq \emptyset$ then $\Theta_{\mathsf{I}}(\theta') \neq \emptyset$.

Corollary 2. *For each grammar $\Gamma = \langle \Sigma, \theta \rangle$, for each Σ interpretation I,*

I is an exhaustive Γ model iff

I is a Γ model, and
for each $\theta' \subseteq \mathcal{D}_0^\Sigma$, for each Σ interpretation I',
if I' is a Γ model and $\Theta_{\mathsf{I}'}(\theta') \neq \emptyset$ then $\Theta_{\mathsf{I}}(\theta') \neq \emptyset$.

The Exhaustive Models Theorem of RSRL

Theorem 1. *For each signature Σ, for each Σ theory θ, there exists a Σ interpretation I such that I is an exhaustive $\langle \Sigma, \theta \rangle$ model.*

As a corollary of THEOREM 1, we get immediately that if an RSRL grammar has a nonempty model, it also has a nonempty exhaustive model. In summation, a non-vacuous grammar has a linguistically sensible meaning in the class of its exhaustive models. King's explanation of the meaning of grammars, *mutatis mutandis*, carries over to RSRL.

A proof of THEOREM 1, due to Richter and King 1997, can be found in Appendix A of Richter 2000.

Pollard 1999 in RSRL

Pollard Feature Structures

On page 47, we introduced the notions of a "Σ interpretation under u in I, I_u" and of a "configuration of entities under u in I, $\langle u, \mathsf{I}_u \rangle$". For each signature Σ, for each Σ interpretation I, and for each $u \in \mathsf{I}$, Pollard 1999 calls the counterpart in SRL of I_u a Σ *interpretation generated by* u *in* I; and, for each signature Σ, for each Σ interpretation I, and for each $u \in \mathsf{I}$, Pollard 1999 calls the SRL counterpart of $\langle u, \mathsf{I}_u \rangle$ a *feature structure* (determined by u in I). Since these mathematical structures differ from standard feature structures in important ways, and since we are concerned with a structural extension of Pollard's entities, I call $\langle u, \mathsf{I}_u \rangle$ *extended Pollard feature structures*.

According to their definition, the extended Pollard feature structures are Pollard's analogue to King's linguistic tokens. To obtain his unique representational idealizations of isomorphic utterance tokens, Pollard must provide a canonical representation of isomorphic utterance tokens. It is the purpose of the following definition of a *node abstraction* to provide the necessary prerequisites:

Definition 24. *For each signature* $\Sigma = \langle \mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F}, \mathcal{R}, \mathcal{AR} \rangle$, ν *is a* **node abstraction in** Σ *iff*

ν *is a pair* $\langle \beta, \varrho \rangle$,

$\beta \subseteq \mathcal{T}^\Sigma$,

$: \in \beta$,

for each $\tau \in \mathcal{T}^\Sigma$, *for each* $\varphi \in \mathcal{A}$, *if* $\tau\varphi \in \beta$ *then* $\tau \in \beta$,

ϱ *is an equivalence relation over* β,

for each $\tau_1 \in \mathcal{T}^\Sigma$, *for each* $\tau_2 \in \mathcal{T}^\Sigma$, *for each* $\varphi \in \mathcal{A}$,

if $\tau_1\varphi \in \beta$ *and* $\langle \tau_1, \tau_2 \rangle \in \varrho$ *then* $\langle \tau_1\varphi, \tau_2\varphi \rangle \in \varrho$.

β is a prefix closed set of terms that contains at least the colon, and ϱ is a right invariant equivalence relation over β.

Extended Pollard abstract feature structures are canonical representations of extended Pollard feature structures. Their nodes are equivalence classes of terms in β, and their root entities are in the equivalence class $|:|_\varrho$:

Definition 25. *For each signature* $\Sigma = \langle \mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F}, \mathcal{R}, \mathcal{AR} \rangle$, *for each* Σ *interpretation* $\mathsf{I} = \langle \mathsf{U}, \mathsf{S}, \mathsf{A}, \mathsf{R} \rangle$, *for each* $u \in \mathsf{U}$,

$\langle u, \mathsf{I} \rangle$ *is an* **extended Pollard abstract feature structure under** Σ *iff*

$\langle u, \mathsf{I} \rangle$ *is an extended Pollard feature structure,*
for some node abstraction in Σ, $\langle \beta, \varrho \rangle$,
$\mathsf{U} = Quo(\langle \beta, \varrho \rangle)$,
$u = |:|_\varrho$,
for each $u_1 \in Quo(\langle \beta, \varrho \rangle)$, *for each* $u_2 \in Quo(\langle \beta, \varrho \rangle)$, *for each* $\varphi \in \mathcal{A}$,
$\mathsf{A}(\varphi)(u_1) = u_2$ *iff*
for some $\tau \in \beta$, $u_1 = |\tau|_\varrho$, *and* $u_2 = |\tau\varphi|_\varrho$.

I write $\mathbb{AFS}^\Sigma_{\mathrm{eP}}$ for the set of extended Pollard abstract feature structures under Σ. We obtain the following crucial result:

Proposition 4. *For each signature* Σ, *for each* Σ *interpretation* $\mathsf{I} = \langle \mathsf{U}, \mathsf{S}, \mathsf{A}, \mathsf{R} \rangle$, *for each* $v \in \mathsf{U}$, *if* $\langle v, \mathsf{I} \rangle$ *is an extended Pollard feature structure then there exists exactly one* $\langle v', \mathsf{I}' \rangle \in \mathbb{AFS}^\Sigma_{\mathrm{eP}}$, *such that for each* $\delta \in \mathcal{D}^\Sigma$,

$v \in D_\mathsf{I}(\delta)$ *iff* $v' \in D_{\mathsf{I}'}(\delta)$.

Strong Generative Capacity

We start with an auxiliary definition:

Definition 26. *For each signature* Σ, *for each* Σ *interpretation* $\mathsf{I} = \langle \mathsf{U}, \mathsf{S}, \mathsf{A}, \mathsf{R} \rangle$, *for each* $v \in \mathsf{U}$,

$\langle v, \mathsf{I}_v \rangle$ *is the* **extended Pollard feature structure determined by** v **in** I *iff*

I_v *is the* Σ *interpretation generated by* v *in* I.

Pollard 1999 presents three equivalent definitions of the strong generative capacity of a grammar. All of them use the familiar relationships between the models of a grammar, the class of exhaustive models of a grammar, and the set of abstract feature structures that can be abstracted from them.

For the first definition of the strong generative capacity of a grammar, Pollard defines an abstraction function, Abst_I. For each entity v in the interpretation I, it contains pairs of v and extended Pollard abstract feature structures whose root entities are indiscernible from v:

Definition 27. *For each signature Σ, for each Σ interpretation* $\mathsf{I} = \langle \mathsf{U}, \mathsf{S}, \mathsf{A}, \mathsf{R} \rangle$,

> $\mathrm{Abst}_{\mathsf{I}}$ *is the binary relation on* $\mathsf{U} \times \mathbb{AFS}^{\Sigma}_{\mathrm{eP}}$ *such that, for each* $v \in \mathsf{U}$, *for each* $\langle v', \mathsf{I}' \rangle \in \mathbb{AFS}^{\Sigma}_{\mathrm{eP}}$, $\langle v, \langle v', \mathsf{I}' \rangle \rangle \in \mathrm{Abst}_{\mathsf{I}}$ *iff*
>
> > *for the extended Pollard feature structure* $\langle v, \mathsf{I}_v \rangle$ *determined by* v *in* I, *for each* $\delta \in \mathcal{D}^{\Sigma}$, $v \in D_{\mathsf{I}_v}(\delta)$ *iff* $v' \in D_{\mathsf{I}'}(\delta)$.

Since each entity, v, in an interpretation, I, determines exactly one extended Pollard feature structure, $\langle v, \mathsf{I}_v \rangle$, in the interpretation I, it follows with PROPOSITION 4 that each $\mathrm{Abst}_{\mathsf{I}}$ is functional:

Proposition 5. *For each signature Σ, for each Σ interpretation* $\mathsf{I} = \langle \mathsf{U}, \mathsf{S}, \mathsf{A}, \mathsf{R} \rangle$, $\mathrm{Abst}_{\mathsf{I}}$ *is a total function from* U *to* $\mathbb{AFS}^{\Sigma}_{\mathrm{eP}}$.

If v is an entity in an interpretation I, then I call $\mathrm{Abst}_{\mathsf{I}}(v)$ its *abstraction*. The strong generative capacity of a grammar is supposed to contain one isomorphic member to each utterance token in the natural language that is judged grammatical:

Definition 28. [SGC 1] *For each signature Σ, SGC is the total function from the set of grammars to* $Pow(\mathbb{AFS}^{\Sigma}_{\mathrm{eP}})$ *such that, for each* $\theta \subseteq \mathcal{D}^{\Sigma}$,

$$\mathsf{SGC}(\langle \Sigma, \theta \rangle) = \left\{ \alpha \in \mathbb{AFS}^{\Sigma}_{\mathrm{eP}} \;\middle|\; \begin{array}{l} \textit{for some } \langle \Sigma, \theta \rangle \textit{ model} \\ \quad \mathsf{I} = \langle \mathsf{U}, \mathsf{S}, \mathsf{A}, \mathsf{R} \rangle, \\ \textit{for some } v \in \mathsf{U}, \\ \alpha = \mathrm{Abst}_{\mathsf{I}}(v) \end{array} \right\}.$$

For each grammar Γ, Pollard calls the set $\mathsf{SGC}(\Gamma)$ the *strong generative capacity of* Γ.

Exhaustive models provide an obvious alternative to obtaining the SGC of a grammar by taking the abstract feature structures that can be abstracted from the entities in all models. It is clearly sufficient to consider only one model if that model is big enough in the sense of containing congruent copies of all possible configurations under entities in all models, and to take the set of abstract feature structures that can be abstracted from the entities in such a model.

Following Pollard 1999, we can exploit the existence of a canonical extended Pollard feature structure for each entity in an interpretation for a new way of characterizing exhaustive models:

Definition 29. *For each signature Σ, for each* $\theta \subseteq \mathcal{D}^{\Sigma}$,

> $\mathsf{I} = \langle \mathsf{U}, \mathsf{S}, \mathsf{A}, \mathsf{R} \rangle$ *is an* **exhaustive** $\langle \Sigma, \theta \rangle$ **model** *iff*
>
> I *is a* $\langle \Sigma, \theta \rangle$ *model, and for each* $\langle \Sigma, \theta \rangle$ *model* $\mathsf{I}' = \langle \mathsf{U}', \mathsf{S}', \mathsf{A}', \mathsf{R}' \rangle$, *for each* $v' \in \mathsf{U}'$, *there exists a* $v \in \mathsf{U}$ *such that* $\mathrm{Abst}_{\mathsf{I}}(v) = \mathrm{Abst}_{\mathsf{I}'}(v')$.

The function that is defined next maps each Σ interpretation I to the smallest set of extended Pollard abstract feature structures under Σ that, for each extended Pollard feature structure determined by some v in I, contains an element that is indiscernible from $\langle v, \mathsf{I}_v \rangle$:

Definition 30. *For each signature Σ, ABST is the large total function from the class of Σ interpretations to* $Pow(\mathbb{AFS}^{\Sigma}_{\mathrm{eP}})$ *such that, for each Σ interpretation* $\mathsf{I} = \langle \mathsf{U}, \mathsf{S}, \mathsf{A}, \mathsf{R} \rangle$,

$$\mathsf{ABST}(\mathsf{I}) = \left\{ \alpha \in \mathbb{AFS}^{\Sigma}_{\mathrm{eP}} \;\middle|\; \begin{array}{l} \textit{for some } v \in \mathsf{U}, \\ \alpha = \mathrm{Abst}_{\mathsf{I}}(v) \end{array} \right\}.$$

For each Σ interpretation I, I call $\mathsf{ABST}(\mathsf{I})$ the *set of extended Pollard abstract feature structures determined by* I. The function ABST leads to the second definition of the strong generative capacity of a grammar:

Definition 31. [SGC 2] *For each signature Σ, SGC is the total function from the set of grammars to* $Pow(\mathbb{AFS}^{\Sigma}_{\mathrm{eP}})$ *such that, for each* $\theta \subseteq \mathcal{D}^{\Sigma}$,

$$\mathsf{SGC}(\langle \Sigma, \theta \rangle) = \left\{ \alpha \in \mathbb{AFS}^{\Sigma}_{\mathrm{eP}} \;\middle|\; \begin{array}{l} \textit{for each (equivalently some)} \\ \textit{exhaustive } \langle \Sigma, \theta \rangle \textit{ model } \mathsf{I}, \\ \alpha \in \mathsf{ABST}(\mathsf{I}) \end{array} \right\}.$$

Finally, the third way of defining the strong generative capacity of a grammar exploits the fact that an extended Pollard abstract feature structure, in contrast to traditional definitions of abstract feature structures, contains an interpretation of a signature. In particular, it contains the interpretation with canonical representations of the elements of the universe that is generated by its root entity. Following Pollard, we say that a grammar Γ *generates* an extended Pollard abstract feature structure exactly if its interpretation is a Γ model:

Definition 32. *For each signature* Σ, *for each* $\theta \subseteq \mathcal{D}^{\Sigma}$, *for each* Σ *interpretation* $\mathsf{I} = \langle \mathsf{U}, \mathsf{S}, \mathsf{A}, \mathsf{R} \rangle$, *for each* $v \in \mathsf{U}$, $\langle \Sigma, \theta \rangle$ **generates** $\langle v, \mathsf{I} \rangle$ *iff*

$\langle v, \mathsf{I} \rangle \in \mathbb{AFS}^{\Sigma}_{\mathrm{eP}}$, *and*

I *is a* $\langle \Sigma, \theta \rangle$ *model.*

This leads to the last characterization of the SGC of a grammar as the set of all extended Pollard abstract feature structures whose interpretation is a model of a grammar Γ:

Definition 33. [SGC 3] *For each signature* Σ, SGC *is the total function from the set of grammars to* $Pow(\mathbb{AFS}^{\Sigma}_{\mathrm{eP}})$ *such that, for each* $\theta \subseteq \mathcal{D}^{\Sigma}$,

$$\mathsf{SGC}(\langle \Sigma, \theta \rangle) = \left\{ \alpha \in \mathbb{AFS}^{\Sigma}_{\mathrm{eP}} \,\middle|\, \langle \Sigma, \theta \rangle \text{ generates } \alpha \right\}.$$

Pollard and Sag 1994 in RSRL

The Definition of an Admission Relation in RSRL

Suppose $\Sigma = \langle \mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F}, \mathcal{R}, \mathcal{AR} \rangle$ is a signature. I call each member of $\mathcal{A}^*$ a Σ path, and write ε for the empty path, the unique path of length zero.

Definition 34. *For each signature* $\Sigma = \langle \mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F}, \mathcal{R}, \mathcal{AR} \rangle$,

μ *is a* **morph in** Σ *iff*

μ *is a quadruple* $\langle \beta, \varrho, \lambda, \xi \rangle$,
$\beta \subseteq \mathcal{A}^*$,
$\varepsilon \in \beta$,
for each $\pi \in \mathcal{A}^*$, *for each* $\alpha \in \mathcal{A}$,
if $\pi\alpha \in \beta$ *then* $\pi \in \beta$,
ϱ *is an equivalence relation over* β,
for each $\pi_1 \in \mathcal{A}^*$, *for each* $\pi_2 \in \mathcal{A}^*$, *for each* $\alpha \in \mathcal{A}$,
if $\pi_1\alpha \in \beta$ *and* $\langle \pi_1, \pi_2 \rangle \in \varrho$ *then* $\langle \pi_1\alpha, \pi_2\alpha \rangle \in \varrho$,
λ *is a total function from* β *to* $\mathcal{S}$,
for each $\pi_1 \in \mathcal{A}^*$, *for each* $\pi_2 \in \mathcal{A}^*$,
if $\langle \pi_1, \pi_2 \rangle \in \varrho$ *then* $\lambda(\pi_1) = \lambda(\pi_2)$,
for each $\pi \in \mathcal{A}^*$, *for each* $\alpha \in \mathcal{A}$,

if $\pi\alpha \in \beta$ *then* $\mathcal{F}\langle \lambda(\pi), \alpha \rangle$ *is defined and* $\lambda(\pi\alpha) \sqsubseteq \mathcal{F}\langle \lambda(\pi), \alpha \rangle$,
for each $\pi \in \mathcal{A}^*$, *for each* $\alpha \in \mathcal{A}$,
if $\pi \in \beta$ *and* $\mathcal{F}\langle \lambda(\pi), \alpha \rangle$ *is defined then* $\pi\alpha \in \beta$,
$\xi \subseteq \mathcal{R} \times \overline{\beta}^*$,
for each $\rho \in \mathcal{R}$, *for each* $\pi_1 \in \overline{\beta}$,..., *for each* $\pi_n \in \overline{\beta}$,
if $\langle \rho, \pi_1, \ldots, \pi_n \rangle \in \xi$ *then* $n = \mathcal{AR}(\rho)$, *and*
for each $\rho \in \mathcal{R}$, *for each* $\pi_1 \in \overline{\beta}$, ..., *for each* $\pi_n \in \overline{\beta}$, *for each* $\pi'_1 \in \overline{\beta}$, ..., *for each* $\pi'_n \in \overline{\beta}$,
if $\langle \rho, \pi_1, \ldots, \pi_n \rangle \in \xi$, *and*
for each $i \in \{1, \ldots, n\}$,
$\pi_i \in \beta$ *and* $\langle \pi_i, \pi'_i \rangle \in \varrho$, *or*
for some $m \in \mathbb{N}$,
$\pi_i \in \beta^*$,
$\pi_i = \langle \pi_{i_1}, \ldots, \pi_{i_m} \rangle$,
$\pi'_i = \langle \pi'_{i_1}, \ldots, \pi'_{i_m} \rangle$, *and*
$\langle \pi_{i_1}, \pi'_{i_1} \rangle \in \varrho, \ldots, \langle \pi_{i_m}, \pi'_{i_m} \rangle \in \varrho$,
then $\langle \rho, \pi'_1, \ldots, \pi'_n \rangle \in \xi$.

Suppose Σ is a signature and $\mu = \langle \beta, \varrho, \lambda, \xi \rangle$ is a Σ morph. I call β the *basis set in* μ, ϱ the *re-entrancy relation in* μ, λ the *label function in* μ, and ξ the *relation extension in* μ. I write $\mathcal{M}_\Sigma$ for the set of Σ morphs.

The Σ morphs are a straightforward extension of the abstract feature structures of Moshier 1988.

Definition 35. *For each signature* $\Sigma = \langle \mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F}, \mathcal{R}, \mathcal{AR} \rangle$, *for each* $\pi \in \mathcal{A}^*$, *for each* $\langle \pi_1, \ldots, \pi_n \rangle \in \mathcal{A}^{**}$,

$\pi\langle \pi_1, \ldots, \pi_n \rangle$ *is an abbreviatory notation for* $\langle \pi\pi_1, \ldots, \pi\pi_n \rangle$.

Definition 36. *For each signature* $\Sigma = \langle \mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F}, \mathcal{R}, \mathcal{AR} \rangle$, *for each* $\mu = \langle \beta, \varrho, \lambda, \xi \rangle \in \mathcal{M}_\Sigma$, *for each* $\pi \in \mathcal{A}^*$,

$\beta/\pi = \{\pi' \in \mathcal{A}^* \mid \pi\pi' \in \beta\}$,

$\varrho/\pi = \{\langle \pi_1, \pi_2 \rangle \in \mathcal{A}^* \times \mathcal{A}^* \mid \langle \pi\pi_1, \pi\pi_2 \rangle \in \varrho\}$,

$\lambda/\pi = \{\langle \pi', \sigma \rangle \in \mathcal{A}^* \times \mathcal{S} \mid \langle \pi\pi', \sigma \rangle \in \lambda\}$,

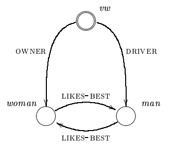
$\xi/\pi = \{\langle \rho, \pi_1, \ldots, \pi_n \rangle \in \mathcal{R} \times \overline{(\beta/\pi)}^* \mid \langle \rho, \pi\pi_1, \ldots, \pi\pi_n \rangle \in \xi\}$, *and*

$\mu/\pi = \langle \beta/\pi, \varrho/\pi, \lambda/\pi, \xi/\pi \rangle$.

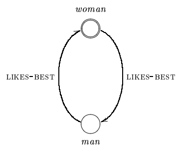
If Σ is a signature, μ is a Σ morph and π is a Σ path then I call μ/π the π *reduct of* μ.

For example, the OWNER reduct of the Σ morph in (1a) is the Σ morph in (1b):

(1)a.



b.



If a Σ path π is in the basis set of the Σ morph μ then the π reduct of μ is also a Σ morph:

Proposition 6. *For each signature* $\Sigma = \langle \mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F}, \mathcal{R}, \mathcal{AR} \rangle$, *for each* $\mu = \langle \beta, \varrho, \lambda, \xi \rangle \in \mathcal{M}_\Sigma$, *for each* $\pi \in \mathcal{A}^*$,

if $\pi \in \beta$ *then* $\mu/\pi \in \mathcal{M}_\Sigma$.

DEFINITION 37 provides a necessary substitute for the variable assignments in interpretations for morph satisfaction. Instead of entities or chains of entities, variables are assigned paths or tuples of paths:

Definition 37. *For each signature* $\Sigma = \langle \mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F}, \mathcal{R}, \mathcal{AR} \rangle$,

I_Σ *is the set of total functions from* $\mathcal{VAR}$ *to* $\overline{\mathcal{A}^*}$.

Let Σ be a signature. Then I call each member of I_Σ a Σ *insertion*. Given a Σ insertion that assigns the Σ path π to the variable v, a Σ morph $\mu = \langle \beta, \varrho, \lambda, \xi \rangle$ satisfies the Σ formula '$v \sim \sigma$' only if $\langle \pi, \sigma \rangle \in \lambda$.

For the definition of morph satisfaction, we finally need the three auxiliary definitions that follow next:

Definition 38. *For each signature* $\Sigma = \langle \mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F}, \mathcal{R}, \mathcal{AR} \rangle$,

T_Σ *is the smallest partial function from* $\overline{\mathcal{A}^*} \times \widehat{\mathcal{A}}$ *to* $\overline{\mathcal{A}^*}$ *such that,*

for each $\pi \in \mathcal{A}^*$, *for each* $\alpha \in \mathcal{A}$,
$\mathrm{T}_\Sigma(\pi, \alpha) = \pi\alpha$,

for each $\langle \pi_0, \ldots, \pi_n \rangle \in \mathcal{A}^{**}$,
$\mathrm{T}_\Sigma(\langle \pi_0, \ldots, \pi_n \rangle, \dagger) = \pi_0$,
$\mathrm{T}_\Sigma(\langle \pi_0, \ldots, \pi_n \rangle, \triangleright) = \langle \pi_1, \ldots, \pi_n \rangle$.

Definition 39. *For each signature* $\Sigma = \langle \mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F}, \mathcal{R}, \mathcal{AR} \rangle$, *for each* $\iota \in \mathrm{I}_\Sigma$,

Π^ι_Σ *is the smallest partial function from* $\mathcal{T}^\Sigma$ *to* $\overline{\mathcal{A}^*}$ *such that*

$\Pi^\iota_\Sigma(:) = \varepsilon$,
for each $v \in \mathcal{VAR}$, $\Pi^\iota_\Sigma(v) = \iota(v)$, *and*
for each $\tau \in \mathcal{T}^\Sigma$, *for each* $\alpha \in \widehat{\mathcal{A}}$, $\Pi^\iota_\Sigma(\tau\alpha) = \mathrm{T}_\Sigma(\Pi^\iota_\Sigma(\tau), \alpha)$.

Suppose Σ is a signature, and ι is a Σ insertion. Then I call each Π^ι_Σ the *path insertion function under* ι *in* Σ.

Definition 40. *For each signature* Σ, *for each* $\langle \beta, \varrho, \lambda, \xi \rangle \in \mathcal{M}_\Sigma$,

$\hat{\varrho}$ *is the smallest subset of* $\overline{\beta} \times \overline{\beta}$ *such that*

$\varrho \subseteq \hat{\varrho}$, *and*
for each $\pi_1 \in \beta, \ldots, \pi_n \in \beta$, *for each* $\pi'_1 \in \beta, \ldots, \pi'_n \in \beta$,

if $\langle \pi_1, \pi_1' \rangle \in \varrho$, *. . . , and* $\langle \pi_n, \pi_n' \rangle \in \varrho$
then $\langle \langle \pi_1, \ldots, \pi_n \rangle, \langle \pi_1', \ldots, \pi_n' \rangle \rangle \in \hat{\varrho}$,

$\hat{\lambda}$ *is the total function from* $\overline{\beta}$ *to* $\widehat{\mathcal{S}}$ *such that*

for each $\pi \in \beta$, $\hat{\lambda}(\pi) = \lambda(\pi)$,
for each $\pi_1 \in \beta$, *. . . , for each* $\pi_n \in \beta$,

$$\hat{\lambda}(\langle \pi_1, \ldots, \pi_n \rangle) = \begin{cases} echain & \textit{if } n = 0, \\ nechain & \textit{if } n > 0. \end{cases}$$

For each signature Σ and for each Σ insertion $\iota \in \mathrm{I}_\Sigma$, the morph satisfaction function under ι in Σ, Δ^ι_Σ is defined inductively for all Σ formulae.

Definition 41. *For each signature* $\Sigma = \langle \mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F}, \mathcal{R}, \mathcal{AR} \rangle$, *for each* $\iota \in \mathrm{I}_\Sigma$, Δ^ι_Σ *is the total function from* $\mathcal{D}^\Sigma$ *to* $Pow(\mathcal{M}_\Sigma)$ *such that*

for each $\tau \in \mathcal{T}^\Sigma$, *for each* $\sigma \in \widehat{\mathcal{G}}$,

$$\Delta^\iota_\Sigma(\tau \sim \sigma) = \left\{ \langle \beta, \varrho, \lambda, \xi \rangle \in \mathcal{M}_\Sigma \;\middle|\; \begin{array}{l} \Pi^\iota_\Sigma(\tau) \textit{ is defined, and,} \\ \textit{for some } \sigma' \in \widehat{\mathcal{S}}, \\ \quad \langle \Pi^\iota_\Sigma(\tau), \sigma' \rangle \in \hat{\lambda} \textit{ and} \\ \quad \sigma' \mathrel{\widehat{\sqsubseteq}} \sigma \end{array} \right\},$$

for each $\tau_1 \in \mathcal{T}^\Sigma$, *for each* $\tau_2 \in \mathcal{T}^\Sigma$,

$$\Delta^\iota_\Sigma(\tau_1 \approx \tau_2) = \left\{ \langle \beta, \varrho, \lambda, \xi \rangle \in \mathcal{M}_\Sigma \;\middle|\; \begin{array}{l} \Pi^\iota_\Sigma(\tau_1) \textit{ is defined,} \\ \Pi^\iota_\Sigma(\tau_2) \textit{ is defined, and} \\ \langle \Pi^\iota_\Sigma(\tau_1), \Pi^\iota_\Sigma(\tau_2) \rangle \in \hat{\varrho} \end{array} \right\},$$

for each $\rho \in \mathcal{R}$, *for each* $v_1 \in \mathcal{VAR}$, *. . . , for each* $v_n \in \mathcal{VAR}$,

$$\Delta^\iota_\Sigma(\rho(v_1, \ldots, v_n)) = \left\{ \langle \beta, \varrho, \lambda, \xi \rangle \in \mathcal{M}_\Sigma \;\middle|\; \langle \rho, \iota(v_1), \ldots, \iota(v_n) \rangle \in \xi \right\},$$

for each $v \in \mathcal{VAR}$, *for each* $\delta \in \mathcal{D}^\Sigma$,

$$\Delta^\iota_\Sigma(\exists v\, \delta) = \left\{ \langle \beta, \varrho, \lambda, \xi \rangle \in \mathcal{M}_\Sigma \;\middle|\; \begin{array}{l} \textit{for some } \pi \in \overline{\beta}, \\ \quad \langle \beta, \varrho, \lambda, \xi \rangle \in \Delta^{\iota[\frac{\pi}{v}]}_\Sigma(\delta) \end{array} \right\},$$

for each $v \in \mathcal{VAR}$, *for each* $\delta \in \mathcal{D}^\Sigma$,

$$\Delta^\iota_\Sigma(\forall v\, \delta) = \left\{ \langle \beta, \varrho, \lambda, \xi \rangle \in \mathcal{M}_\Sigma \;\middle|\; \begin{array}{l} \textit{for each } \pi \in \overline{\beta}, \\ \quad \langle \beta, \varrho, \lambda, \xi \rangle \in \Delta^{\iota[\frac{\pi}{v}]}_\Sigma(\delta) \end{array} \right\},$$

for each $\delta \in \mathcal{D}^\Sigma$,

$$\Delta^\iota_\Sigma(\neg\delta) = \mathcal{M}_\Sigma \backslash \Delta^\iota_\Sigma(\delta),$$

for each $\delta_1 \in \mathcal{D}^\Sigma$, *for each* $\delta_2 \in \mathcal{D}^\Sigma$,

$$\Delta^\iota_\Sigma([\delta_1 \wedge \delta_2]) = \Delta^\iota_\Sigma(\delta_1) \cap \Delta^\iota_\Sigma(\delta_2),$$

for each $\delta_1 \in \mathcal{D}^\Sigma$, *for each* $\delta_2 \in \mathcal{D}^\Sigma$,

$$\Delta^\iota_\Sigma([\delta_1 \vee \delta_2]) = \Delta^\iota_\Sigma(\delta_1) \cup \Delta^\iota_\Sigma(\delta_2), \textit{ and}$$

for each $\delta_1 \in \mathcal{D}^\Sigma$, *for each* $\delta_2 \in \mathcal{D}^\Sigma$,

$$\Delta^\iota_\Sigma([\delta_1 \rightarrow \delta_2]) = (\mathcal{M}_\Sigma \backslash \Delta^\iota_\Sigma(\delta_1)) \cup \Delta^\iota_\Sigma(\delta_2).$$

Let Σ be a signature, μ a Σ morph, ι a Σ insertion and δ a Σ formula. I call Δ^ι_Σ the *morph satisfaction function under* ι *in* Σ, and say μ *satisfies* δ *under* ι *in* Σ if and only if $\mu \in \Delta^\iota_\Sigma(\delta)$.

Definition 42. *For each signature* Σ,

M_Σ *is the total function from* $Pow(\mathcal{D}^\Sigma_0)$ *to* $Pow(\mathcal{M}_\Sigma)$ *such that for each* $\theta \subseteq \mathcal{D}^\Sigma_0$,

$$M_\Sigma(\theta) = \left\{ \langle \beta, \varrho, \lambda, \xi \rangle \in \mathcal{M}_\Sigma \;\middle|\; \begin{array}{l} \textit{for each } \pi \in \beta, \\ \textit{for each } \iota \in \mathrm{I}_\Sigma, \textit{ and} \\ \textit{for each } \delta \in \theta, \\ \quad \langle \beta, \varrho, \lambda, \xi \rangle / \pi \in \Delta^\iota_\Sigma(\delta) \end{array} \right\}.$$

Let Σ be a signature. I call M_Σ the *morph admission function in* Σ. A Σ morph μ is in the set of Σ morphs admitted by a Σ theory θ exactly if μ and every possible π reduct of μ satisfy every description in θ.

Formalization of a few principles of Pollard and Sag 1994

Throughout the following examples, the syntax used in the formalization of principles of grammar is the AVM notation for RSRL defined in Section 3.2 of Richter 2000.

(2) *The* SUBJECT CONDITION *(Pollard and Sag, 1994, p. 400)*

If the initial element of the SUBCAT value of a word is slashed, then so is some other element of that list.

(3) *The* SUBJECT CONDITION *formalized*

$$\forall \boxed{1} \left(\begin{bmatrix} \textit{word} \\ \text{SS LOC CAT SUBCAT } \left\langle \begin{bmatrix}\text{NONLOC INHERITED SLASH } \textit{neset}\end{bmatrix} \middle| \boxed{1} \right\rangle \end{bmatrix} \rightarrow \exists x \ \texttt{member}\left({}^{x}\begin{bmatrix}\text{NONLOC INHERITED SLASH } \textit{neset}\end{bmatrix}, \boxed{1} \right) \right)$$

(4) *The* CONTROL THEORY *(Pollard and Sag, 1994, p. 401)*

If the SOA-ARG value of a *control-qfpsoa* is token-identical with the CONTENT value of a *local* entity whose CATEGORY | SUBCAT value is a list of length one, then the member of that list is (1) reflexive, and (2) coindexed with the INFLUENCED (respectively, COMMITTOR, EXPERIENCER) value of the *control-qfpsoa* if the latter is of sort *influence* (respectively, *commitment*, *orientation*).

(5) *The* CONTROL THEORY *formalized*

$$\forall \boxed{1} \, \forall \boxed{2} \, \forall \boxed{3} \left(\begin{array}{l} \exists \boxed{4} \left(\boxed{1} \begin{bmatrix} \textit{influence} \\ \text{SOA-ARG } \boxed{4} \end{bmatrix} \wedge \boxed{2} \begin{bmatrix} \textit{local} \\ \text{CAT SUBCAT } \langle \boxed{3} \rangle \\ \text{CONT } \boxed{4} \end{bmatrix} \right) \rightarrow \\ \exists \boxed{5} \left(\boxed{3} \begin{bmatrix} \text{LOC CONT } \begin{bmatrix} \textit{reflexive} \\ \text{INDEX } \boxed{5} \end{bmatrix} \end{bmatrix} \wedge \boxed{1} \begin{bmatrix} \text{INFLUENCED } \boxed{5} \end{bmatrix} \right) \end{array} \right) \wedge$$

$$\forall \boxed{1} \, \forall \boxed{2} \, \forall \boxed{3} \left(\begin{array}{l} \exists \boxed{4} \left(\boxed{1} \begin{bmatrix} \textit{commitment} \\ \text{SOA-ARG } \boxed{4} \end{bmatrix} \wedge \boxed{2} \begin{bmatrix} \textit{local} \\ \text{CAT SUBCAT } \langle \boxed{3} \rangle \\ \text{CONT } \boxed{4} \end{bmatrix} \right) \rightarrow \\ \exists \boxed{5} \left(\boxed{3} \begin{bmatrix} \text{LOC CONT } \begin{bmatrix} \textit{reflexive} \\ \text{INDEX } \boxed{5} \end{bmatrix} \end{bmatrix} \wedge \boxed{1} \begin{bmatrix} \text{COMMITTOR } \boxed{5} \end{bmatrix} \right) \end{array} \right) \wedge$$

$$\forall \boxed{1} \, \forall \boxed{2} \, \forall \boxed{3} \left(\begin{array}{l} \exists \boxed{4} \left(\boxed{1} \begin{bmatrix} \textit{orientation} \\ \text{SOA-ARG } \boxed{4} \end{bmatrix} \wedge \boxed{2} \begin{bmatrix} \textit{local} \\ \text{CAT SUBCAT } \langle \boxed{3} \rangle \\ \text{CONT } \boxed{4} \end{bmatrix} \right) \rightarrow \\ \exists \boxed{5} \left(\boxed{3} \begin{bmatrix} \text{LOC CONT } \begin{bmatrix} \textit{reflexive} \\ \text{INDEX } \boxed{5} \end{bmatrix} \end{bmatrix} \wedge \boxed{1} \begin{bmatrix} \text{ORIENTATION } \boxed{5} \end{bmatrix} \right) \end{array} \right)$$

(6) *The* BINDING THEORY *(Pollard and Sag, 1994, p. 401)*

Principle A. A locally o-commanded anaphor must be locally o-bound.
Principle B. A personal pronoun must be locally o-free.
Principle C. A nonpronoun must be o-free.

The BINDING THEORY employs a number of auxiliary relations that we must define first. For constraining the meaning of relation symbols in the grammar, we employ a convenient clause notation.

"One *synsem* object is *more oblique* than another provided it appears to the right of the other on the SUBCAT list of some word." (Pollard and Sag, 1994, p. 401)

(7) $\texttt{more-oblique}(x, y) \overset{\forall}{\Longleftarrow}$
$${}^{w}\begin{bmatrix} \textit{word} \\ \text{SS LOC CAT SUBCAT } \boxed{1} \end{bmatrix} \wedge \texttt{to-the-right}(x, y, \boxed{1})$$

(8) $\texttt{to-the-right}(x, y, z) \overset{\forall}{\Longleftarrow}$
$$\exists \boxed{1} \ (z \, \langle y | \boxed{1} \rangle \wedge \texttt{member}(x, \boxed{1}))$$
$$\vee \exists \boxed{1} \ (z \, \langle \textit{object} | \boxed{1} \rangle \wedge \texttt{to-the-right}(x, y, \boxed{1}))$$

"One referential *synsem* object *locally o-commands* another provided they have distinct LOCAL values and either (1) the second is more oblique than the first, or (2) the second is a member of the SUBCAT list of a *synsem* object that is more oblique than the first."

(9) $\texttt{loc-o-command}(x, y) \stackrel{\forall}{\Longleftarrow}$

$$\left(\begin{array}{l} {}^{x}\!\left[\text{LOC } \boxed{1} \left[\text{CONT INDEX } \textit{ref}\right]\right] \wedge \\ {}^{y}\!\left[\text{LOC } \boxed{2} \left[\text{CONT INDEX } \textit{ref}\right]\right] \wedge \\ \neg \boxed{1} = \boxed{2} \end{array} \right) \wedge \left(\begin{array}{l} \texttt{more-oblique}(y, x) \vee \\ \left(\begin{array}{l} \texttt{more-oblique}\left({}^{s}\!\left[\begin{array}{l} \textit{synsem} \\ \text{LOC CAT SUBCAT } \boxed{3} \end{array}\right], x \right) \wedge \\ \texttt{member}(y, \boxed{3}) \end{array} \right) \end{array} \right)$$

"One referential *synsem* object *o-commands* another provided they have distinct LOCAL values and either (1) the second is more oblique than the first, (2) the second is a member of the SUBCAT list of a *synsem* object that is o-commanded by the first, or (3) the second has the same LOCAL | CATEGORY | HEAD value as a *synsem* object that is o-commanded by the first." (Pollard and Sag, 1994, p. 401)

(10) $\texttt{o-command}(x, y) \stackrel{\forall}{\Longleftarrow}$

$$\left(\begin{array}{l} {}^{x}\!\left[\text{LOC } \boxed{1} \left[\text{CONT INDEX } \textit{ref}\right]\right] \wedge \\ {}^{y}\!\left[\text{LOC } \boxed{2} \left[\begin{array}{ll} \text{CAT HEAD} & \boxed{4} \\ \text{CONT INDEX} & \textit{ref} \end{array}\right]\right] \wedge \\ \neg \boxed{1} = \boxed{2} \end{array} \right) \wedge \left(\begin{array}{l} \texttt{more-oblique}(y, x) \vee \\ \left(\begin{array}{l} \texttt{o-command}\left(x, {}^{s_1}\!\left[\begin{array}{l} \textit{synsem} \\ \text{LOC CAT SUBCAT } \boxed{3} \end{array}\right] \right) \wedge \\ \texttt{member}(y, \boxed{3}) \end{array} \right) \vee \\ \texttt{o-command}\left(x, {}^{s_2}\!\left[\begin{array}{l} \textit{synsem} \\ \text{LOC } \left[\text{CAT HEAD } \boxed{4}\right] \end{array}\right] \right) \end{array} \right)$$

"One referential *synsem* object *(locally) o-binds* another provided it (locally) o-commands and is coindexed with the other. A referential *synsem* object is *(locally) o-free* provided it is not (locally) o-bound. Two *synsem* entities are *coindexed* provided their LOCAL | CONTENT | INDEX values are token-identical." (Pollard and Sag, 1994, p. 401)

(11) $\texttt{loc-o-bind}(x, y) \stackrel{\forall}{\Longleftarrow}$

$$\texttt{loc-o-command}\left({}^{x}\!\left[\text{LOC CONT INDEX } \boxed{1}\right], {}^{y}\!\left[\text{LOC CONT INDEX } \boxed{1}\right] \right)$$

(12) $\texttt{o-bind}(x, y) \stackrel{\forall}{\Longleftarrow}$

$$\texttt{o-command}\left({}^{x}\!\left[\text{LOC CONT INDEX } \boxed{1}\right], {}^{y}\!\left[\text{LOC CONT INDEX } \boxed{1}\right] \right)$$

(13) *The* BINDING THEORY *formalized*

Principle A:

$$\forall x \left(\begin{array}{l} \exists y \; \texttt{loc-o-command}(y, {}^{x}\!\left[\text{LOC CONT } \textit{ana}\right]) \rightarrow \\ \exists z \; \texttt{loc-o-bind}(z, x) \end{array} \right)$$

Principle B:

$$\forall x \; ({}^{x}\!\left[\text{LOC CONT } \textit{ppro}\right] \rightarrow \neg \exists y \; \texttt{loc-o-bind}(y, x))$$

Principle C:

$$\forall x \; ({}^{x}\!\left[\text{LOC CONT } \textit{npro}\right] \rightarrow \neg \exists y \; \texttt{o-bind}(y, x))$$

Some aspects of the case theory of Przepiórkowski 1999

First of all, we must note a few important differences between the signature of Pollard and Sag 1994 and the signature of Przepiórkowski 1999:

(14)a. *argument* ARGUMENT *synsem*
RAISED *boolean*

b. *category* HEAD *head*
VALENCE *valence*

c. *head* ARG-ST *list(argument)*

d. *valence* SUBJECT *list(argument)*
SPECIFIER *list(argument)*
COMPLEMENTS *list(argument)*

(15) *The* RAISED *Principle (after Przepiórkowski 1999, p. 424)*

$$unembedded\text{-}sign \rightarrow$$
$$\forall\boxed{1}\ \forall\boxed{2}\ \forall\boxed{3}\ \forall\boxed{4}\forall\boxed{5}\ \left(\left(\begin{array}{l}\boxed{1}\left[\text{HEAD }\boxed{2}[\text{ARG-ST }\boxed{3}]\right]\wedge\\ \texttt{member}(\boxed{4}[\text{ARG }\boxed{5}],\boxed{3})\end{array}\right)\rightarrow\left(\begin{array}{l}\boxed{4}[\text{RAISED }+]\ \leftrightarrow\\ \exists\boxed{6}\ \exists\boxed{7}\ \exists\boxed{8}\ \exists\boxed{9}\\ \left(\begin{array}{l}\boxed{7}[\text{ARG-ST }\boxed{6}]\wedge\\ \texttt{member}(\boxed{8}[\text{ARG }\boxed{5}],\boxed{6})\wedge\\ \texttt{member}(\boxed{9}[\text{ARG LOC CAT HEAD }\boxed{2}],\boxed{6})\end{array}\right)\end{array}\right)\right)$$

Przepiórkowski 1999 assumes that the HEAD values of two *category* entities are identical if and only if one is the CATEGORY value of a syntactic projection of the other. Supposing that this is the case, (15) says that in each utterance, for each *category* component, $\boxed{1}$, with HEAD value $\boxed{2}$, and for each *argument* entity, $\boxed{4}$, on the ARG-ST list of $\boxed{1}$, $\boxed{4}$ is raised if and only if there is a *head* entity, $\boxed{7}$, in the utterance with an ARG-ST list, $\boxed{6}$, that contains an *argument* entity with the *synsem* entity that is also the ARGUMENT value of $\boxed{4}$ as its ARGUMENT value and another *argument* that has the

same HEAD value as $\boxed{1}$, i.e., a projection of $\boxed{1}$ is selected by means of the ARG-ST list of $\boxed{7}$.

(16) *A case principle of Polish (after Przepiórkowski 1999, p. 427)*

$$\forall\boxed{1}\ \forall\boxed{2}\ \left(\left(\begin{array}{l}\begin{bmatrix}\textit{verbal}\\ \text{NEG} & +\\ \text{ARG-ST} & \langle\ \textit{object}|\ \boxed{2}\ \textit{list}\rangle\end{bmatrix}\\ \wedge\ \texttt{member}(\boxed{1}[\text{ARG LOC CONT INDEX }\textit{ref}],\boxed{2})\\ \wedge\ \texttt{str-cfc}(\boxed{1})\end{array}\right)\rightarrow \boxed{1}[\text{ARG LOC CAT HEAD AGR CASE }\textit{sgen}]\right)$$

Informally (and simplifying a bit), the principle in (16) says that an argument of a negated verb that is not its least oblique argument, has a referential index, and is a "complete functional complex" with structural case receives structural genitive case. A *complete functional complex* is defined as an *argument* entity that is marked as not being raised and whose SUBJ list only contains arguments that are marked as not being raised. The unary relation `str-cfc` holds of exactly those entities in a model that are a complete functional complex with structural case (Przepiórkowski, 1999, p. 425).

Defining the syntax of Ty2 in HPSG

An illustration of the goal:

(17) *The relevant part of the lexical entry of* every

$$\begin{bmatrix}\textit{word} & \\ \text{PHON} & \langle\textit{every}\rangle\\ \text{SS LOC CONT LF} & \lambda P\lambda Q.\forall x[P(w)(x)\rightarrow Q(w)(x)]\end{bmatrix}$$

NP

$\begin{bmatrix}\text{PHON} & \langle \textit{every, unicorn}\rangle \\ \text{SS LOC CONT LF} & \lambda Q.\forall x[\mathsf{unicorn}'(w)(x) \rightarrow Q(w)(x)]\end{bmatrix}$

C: Det

$\begin{bmatrix}\text{PHON} & \langle \textit{every}\rangle \\ \text{SS LOC CONT LF} & \lambda P \lambda Q.\forall x[P(w)(x) \rightarrow Q(w)(x)]\end{bmatrix}$

H: N

$\begin{bmatrix}\text{PHON} & \langle \textit{unicorn}\rangle \\ \text{SS LOC CONT LF} & \mathsf{unicorn}'(w)\end{bmatrix}$

A reminder: The expressions of Ty2

Definition 43. *The **meaningful expressions of Ty2** are the smallest family* $(\mathrm{Ty2}_\tau)_{\tau \in Types}$ *such that*

for each $\tau \in Types$, $Var_\tau \cup Const_\tau \subset \mathrm{Ty2}_\tau$,

for each $\tau \in Types$, *for each* $\tau' \in Types$,

if $\alpha \in \mathrm{Ty2}_{\langle \tau', \tau\rangle}$ *and* $\beta \in \mathrm{Ty2}_{\tau'}$, *then* $\alpha(\beta) \in \mathrm{Ty2}_\tau$,

for each $\tau \in Types$, *for each* $\tau' \in Types$, *for each* $n \in \mathbb{N}$, *for each* $v_{n,\tau'} \in Var_{\tau'}$, *for each* $\alpha \in \mathrm{Ty2}_\tau$,

$(\lambda v_{n,\tau'}.\alpha) \in \mathrm{Ty2}_{\langle \tau', \tau\rangle}$,

for each $\tau \in Types$, *for each* $\alpha \in \mathrm{Ty2}_\tau$, *for each* $\beta \in \mathrm{Ty2}_\tau$,

$(\alpha = \beta) \in \mathrm{Ty2}_t$,

for each $\alpha \in \mathrm{Ty2}_t$,

$\neg\alpha \in \mathrm{Ty2}_t$,

for each $\alpha \in \mathrm{Ty2}_t$, *for each* $\beta \in \mathrm{Ty2}_t$,

$(\alpha \wedge \beta) \in \mathrm{Ty2}_t$, *and* *(analogously for* $\vee, \rightarrow, \leftrightarrow$*)*

for each $\tau \in Types$, *for each* $n \in \mathbb{N}$, *for each* $v_{n,\tau} \in Var_\tau$, *for each* $\alpha \in \mathrm{Ty2}_t$,

$\exists v_{n,\tau}\alpha \in \mathrm{Ty2}_t$. *(and analogously for* $\forall$*)*

We are now ready to define the RSRL grammar of the meaningful expressions of Ty2. The specification of the grammar of Ty2 closely follows the appendix to Richter and Sailer 1999a with the modifications proposed in Sailer 2000. The ontology of Ty2 expressions is determined by a hierarchy of sorts under the supersort $\mathit{meaningful\text{-}expressions}$ (me). The sort me will be defined as appropriate value for the pair $\langle \mathit{content}, \text{LOGICAL-FORM}\rangle$, and thus be anchored in the feature geometry of signs.

(18) The sort hierarchy of the sort $\mathit{meaningful\text{-}expression}$ (me):

me TYPE *type*

- *var(iable)* N-NUM(BER) *n-num(ber)*
- *const(ant)* N-NUM(BER) *n-num(ber)*
- *appl(ication)* FUNC(TOR) *me*, ARG(UMENT) *me*
- *abstr(action)* VAR *variable*, ARG *me*
- *equ(ation)* ARG1 *me*, ARG2 *me*
- *neg(ation)* ARG *me*
- *l(ogical)-const(ant)* AGR1 *me*, ARG2 *me*
  - *dis(junction)*
  - *con(junction)*
  - *imp(lication)*
  - *bi-imp(lication)*
- *quant(ifiers)* VAR *variable*, SCOPE *me*
  - *uni(versal)*
  - *exi(stential)*

The attribute TYPE is defined on every entity of sort *me*. Its value encodes the type of the expression and can be read as a subscript of the expression. Entities of sort *var* and entities of sort *const* have one more subscript. The second subscript is the natural number, n, that is used to number the variables (and constants) of each type. To number the variables and constants of each type, a representation of natural numbers is needed. With existential component quantification, a theory of natural numbers can easily be defined on the basis of a simple signature:

(19) The natural numbers:

n-number
- *zero*
- *non-zero* N-NUMBER *n-number*

The theory of the natural numbers:

$\textit{n-number} \rightarrow \ \exists x \ {}^{x}[\textit{zero}]$

Every meaningful expression has a subscript that encodes its type. A type is either atomic or a pair. There is a species for each atomic type, *entity* for the type e, *truth* for the type t, and *w-index* for the type s. Complex types are modeled by entities of sort *complex-type*. They

have two attributes, IN and OUT, which are again *type*-valued. The entity under IN models the first type in the pair that constitutes a complex type, and the entity under OUT models the second type in the pair.

(20) The sort hierarchy of *type*:

type
- *atomic-type*
  - *entity*
  - *truth*
  - *w-index*
- *complex-type* IN *type*, OUT *type*

(21) Type restrictions on the logical operators:

$$appl \rightarrow \begin{bmatrix} \text{TYPE} & \boxed{2} \\ \text{FUNC TYPE} & \begin{bmatrix} \text{IN} & \boxed{1} \\ \text{OUT} & \boxed{2} \end{bmatrix} \\ \text{ARG TYPE} & \boxed{1} \end{bmatrix},$$

$$abstr \rightarrow \begin{bmatrix} \text{TYPE} & \begin{bmatrix} \text{IN} & \boxed{1} \\ \text{OUT} & \boxed{2} \end{bmatrix} \\ \text{VAR TYPE} & \boxed{1} \\ \text{ARG TYPE} & \boxed{2} \end{bmatrix},$$

$$equ \rightarrow \begin{bmatrix} \text{TYPE} & truth \\ \text{ARG1 TYPE} & \boxed{1} \\ \text{ARG2 TYPE} & \boxed{1} \end{bmatrix},$$

$$neg \rightarrow \begin{bmatrix} \text{TYPE} & truth \\ \text{ARG TYPE} & truth \end{bmatrix},$$

$$l\text{-}const \rightarrow \begin{bmatrix} \text{TYPE} & truth \\ \text{ARG1 TYPE} & truth \\ \text{ARG2 TYPE} & truth \end{bmatrix},$$

$$quant \rightarrow \begin{bmatrix} \text{TYPE} & truth \\ \text{SCOPE TYPE} & truth \end{bmatrix}$$

We must impose two more conditions, and we will add a third one that will ensure that meaningful expressions of Ty2 will have a representation with a unique configuration. Thus we will

- exclude cyclic structures,
- exclude infinite structures, and
- require that in each meaningful expression, isomorphic configurations be identical.

We use a technique introduced by Sailer 2000 for a GENERAL NON-CYCLICITY PRINCIPLE. Sailer uses a binary relation, `component`. The appropriate meaning of `component` can be defined in grammars with signatures whose set of attributes is finite and whose set of relation symbols contains a symbol `component` that is assigned arity two by the arity function. Assume that $\Sigma = \langle \mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F}, \mathcal{R}, \mathcal{AR} \rangle$ is such a signature. The set of clauses that define `component` is the smallest set C such that

$\texttt{component}(x, y) \stackrel{\forall}{\Longleftarrow} x = y \ \in C$, and

for each $\alpha \in \mathcal{A}$,

$\texttt{component}(x, y) \stackrel{\forall}{\Longleftarrow} {}^{y}[\alpha \ \boxed{1}] \land \texttt{component}(x, \boxed{1}) \ \in C$.

In the definition of the meaning of `component`, x is a component of y.

The TY2 NON-CYCLICITY PRINCIPLE is formulated with respect to a signature, $\Sigma = \langle \mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F}, \mathcal{R}, \mathcal{AR} \rangle$, that includes the signature of Ty2 and has a finite set of attributes.

(22) The Ty2 Non-Cyclicity Principle:

$$
\begin{array}{l}
ty2 \rightarrow \\
\quad \forall \boxed{1} \left(\left(\bigvee_{\alpha \in \mathcal{A}} [\alpha\ \boxed{1}] \right) \rightarrow \neg\, \mathtt{component}(:, \boxed{1}) \right)
\end{array}
$$

The consequent of the principle in (22) is Sailer's GNP. The Non-Cyclicity Principle can be paraphrased as saying that for each entity, o, in the grammar of Ty2, if an attribute α is defined on o and $\boxed{1}$ is its value, then o is not a component of $\boxed{1}$. Informally, o is not cyclic.

Similarly, Sailer 2000 shows that it is possible to formulate a General Finiteness Principle (GFP) over signatures with a finite set of attributes. The GFP requires that every entity in every model of the grammar has finitely many components. Again, Sailer's general principle can easily be restricted to some subset of entities in the model of a grammar by turning it into the consequent of a principle with an appropriate antecedent.

(23) The Ty2 Finiteness Principle:

$$
\begin{array}{l}
ty2 \rightarrow \\
\quad \exists \boxed{1}\, \forall \boxed{2} \left(\mathtt{component}(\boxed{2}, :) \rightarrow \mathtt{member}(\boxed{2}, \boxed{1}[chain]) \right)
\end{array}
$$

For the final principle, the Ty2 Identity Principle, we need an auxiliary relation. Following Sailer 2000, we define the meaning of copy in one clause:

(24)
$$
\begin{array}{l}
\mathtt{copy}(x, y) \overset{\forall}{\Longleftarrow} \\
\quad \bigvee_{\sigma \in \mathcal{S}} \left({}^{x}[\sigma] \wedge {}^{y}[\sigma] \right) \wedge \\
\quad \bigwedge_{\alpha \in \mathcal{A}} \left(\forall \boxed{1} \left({}^{x}[\alpha\ \boxed{1}] \rightarrow \exists \boxed{2} \left({}^{y}[\alpha\ \boxed{2}] \wedge \mathtt{copy}(\boxed{1}, \boxed{2}) \right) \right) \right)
\end{array}
$$

y is a copy of x if x and y have the same species (second line), and if an attribute α with value $\boxed{1}$ is defined on x, then α is also defined on y, and $\boxed{1}$ and the value of α on y, $\boxed{2}$, are in the copy relation (third line).

Two entities that stand in the copy relation are of the same species, and they have the same attributes defined on them; and corresponding attribute values stand again in the copy relation. In the Ty2 Identity Principle, we require that all the corresponding entities in the copy relation of representations of Ty2 expressions be identical:

(25) The Ty2 Identity Principle:

$$
\begin{array}{l}
ty2 \rightarrow \\
\quad \forall \boxed{1} \forall \boxed{2} \left(\mathtt{copy}(\boxed{1}, \boxed{2}) \rightarrow \boxed{1} = \boxed{2} \right)
\end{array}
$$

With the Ty2 grammar in place, we can formulate a first approximation to a Semantics Principle (SP).

(26) Semantics Principle (first approximation)

$$
\begin{array}{l}
phrase \rightarrow \\
\left(\begin{bmatrix}
\textsc{synsem} & \begin{bmatrix} \textsc{loc content lf} & \begin{bmatrix} application \\ \textsc{func}\ \boxed{3} \\ \textsc{arg} & \begin{bmatrix} abstraction \\ \textsc{var} & \begin{bmatrix} \textsc{type} & w\text{-}index \\ \textsc{n-num} & zero \end{bmatrix} \\ \textsc{arg}\ \boxed{4} \end{bmatrix} \end{bmatrix} \end{bmatrix} \\
\textsc{dtrs} & \begin{bmatrix} \textsc{head-dtr synsem loc content lf} & \boxed{1} \\ \textsc{nonhead-dtr synsem loc content lf} & \boxed{2} \end{bmatrix}
\end{bmatrix}
\wedge \left(\begin{array}{c} (\boxed{1} = \boxed{3} \quad \wedge \quad \boxed{2} = \boxed{4}) \\ \vee \\ (\boxed{1} = \boxed{4} \quad \wedge \quad \boxed{2} = \boxed{3}) \end{array} \right) \right)
\end{array}
$$

Sailer 2000 proves the crucial properties of the RSRL grammar of Ty2:

Firstly, there is an exhaustive model of the grammar whose universe contains exactly the natural numbers, the set of types, $Types$, and the meaningful expressions of Ty2. This result shows that the grammar indeed specifies what we want to specify with it.

Secondly, for each indiscernibility class of entities of sort me in an exhaustive model of the grammar, there exists a corresponding Ty2 expression that has the same meaning.

Thirdly, for each Ty2 expression there exists an AVM description that picks out exactly the indiscernibility class of entities in models of the RSRL grammar of Ty2 whose me entities correspond to the Ty2 expression. It is this result that allows us to use Ty2 expressions in AVM descriptions in places where the AVM syntax would require a description of the Ty2 expression.

In HPSG, Ty2 has been used in an analysis of negative polarity items in French (Richter and Sailer, 1999a) and in an analysis of negative concord in Polish (Richter and Sailer, 1999b). Richter and Sailer 1999c uses Ty2 as an example of how an underspecified representation language for an arbitrary semantic representation language can be defined in HPSG. Sailer 2000 is the most comprehensive discussion of Ty2 as semantic representation language in HPSG. It explores different ways of formalizing a grammar of Ty2, adds type shifting rules, and uses the system in an analysis of the syntax and semantics of idiomatic expressions in English.

An operational extension

(27) The sort hierarchies of $content$ and of β-$reduction$:

$$\begin{array}{lll} content & \textsc{logical-form} & me \\ & \beta\text{-}\textsc{reduction} & \beta\text{-}reduction \end{array}$$

$$\begin{array}{lll} \beta\text{-}reduction & \textsc{term} & me \\ & \textsc{aux} & me \\ \quad no\text{-}reduction & & \\ \quad one\text{-}reduction & \textsc{rec} & \beta\text{-}reduction \end{array}$$

(28) The theory of β-$reduction$:

$$no\text{-}reduction \rightarrow \begin{bmatrix} \textsc{term} & \boxed{1} \\ \textsc{aux} & \boxed{1} \end{bmatrix}$$

$$\begin{array}{ll} one\text{-}reduction \rightarrow & \begin{bmatrix} \textsc{term} & \boxed{1} \\ \textsc{aux} & \boxed{2} \\ \textsc{rec} & \begin{bmatrix} \textsc{term} & \boxed{1} \\ \textsc{aux} & \boxed{3} \end{bmatrix} \end{bmatrix} \\ & \wedge\ \mathtt{component}\left({}^{y}\begin{bmatrix} appl & \\ \textsc{func} & \begin{bmatrix} abstr & \\ \textsc{var} & \boxed{5} \\ \textsc{arg} & \boxed{4} \end{bmatrix} \\ \textsc{arg} & \boxed{6} \end{bmatrix}, \boxed{2} \right) \\ & \wedge\ \mathtt{replace}(\boxed{4}, \boxed{5}, \boxed{7}, \boxed{6}) \\ & \wedge\ \mathtt{replace}(\boxed{2}, y, \boxed{3}, \boxed{7}) \end{array}$$

(29) The relation `replace`:

The base cases:

$$\mathtt{replace}(x, y, v, w) \stackrel{\forall}{\Longleftarrow} \\ x = y\ \wedge\ v = w$$

$$\mathtt{replace}(x, y, v, w) \stackrel{\forall}{\Longleftarrow} \\ \neg x = y\ \wedge\ {}^{x}[\,var \vee const\,] \wedge x = v$$

An example of a recursive case:

$$\mathtt{replace}(x, y, v, w) \stackrel{\forall}{\Longleftarrow} \\ \neg x = y\ \wedge\ {}^{x}\begin{bmatrix} abstr & \\ \textsc{type} & \boxed{1} \\ \textsc{var} & \boxed{2} \\ \textsc{arg} & \boxed{3} \end{bmatrix} \wedge\ {}^{v}\begin{bmatrix} abstr & \\ \textsc{type} & \boxed{1} \\ \textsc{var} & \boxed{2} \\ \textsc{arg} & \boxed{4} \end{bmatrix} \wedge \\ \mathtt{replace}(\boxed{3}, y, \boxed{4}, w)$$

Given the theory of the sort β-$reduction$, we only need to define one more auxiliary relation, `redex-free`, before we can state a principle that demands that the LOGICAL-FORM value of each $content$ entity be redex free. The definition of `redex-free` in (30) uses the relation `component`, which we defined above.

(30) The relation `redex-free`:

$$\mathtt{redex\text{-}free}(x) \stackrel{\forall}{\Longleftarrow} \\ \neg\,\exists y\ \left({}^{y}\begin{bmatrix} appl & \\ \textsc{func} & abstr \end{bmatrix} \wedge\ \mathtt{component}(y,\ {}^{x}[me]) \right)$$

(31) The β-REDUCTION PRINCIPLE:

$$content \rightarrow \left(\begin{bmatrix} \textsc{logical-form} & \boxed{1} \\ \beta\text{-}\textsc{reduction term} & \boxed{1} \end{bmatrix} \wedge \mathtt{redex\text{-}free}(\boxed{1}) \right)$$

(32) *The* SEMANTICS PRINCIPLE *(final version)*

$$phrase \rightarrow$$

$$\begin{bmatrix} \textsc{ss} & \begin{bmatrix} \textsc{loc cont } \beta\text{-}\textsc{reduct} & \begin{bmatrix} \beta\text{-}reduction & \\ \textsc{aux} & \begin{bmatrix} application & \\ \textsc{func} & \boxed{3} \\ \textsc{arg} & \begin{bmatrix} abstraction & \\ \textsc{var} & \begin{bmatrix} \textsc{type} & w\text{-}index \\ \textsc{n-num} & zero \end{bmatrix} \\ \textsc{arg} & \boxed{4} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix} \\ \textsc{dtrs} & \begin{bmatrix} \textsc{head-dtr ss loc cont lf} & \boxed{1} \\ \textsc{nonhead-dtr ss loc cont lf} & \boxed{2} \end{bmatrix} \end{bmatrix}$$

$$\wedge\ \left((\boxed{1} = \boxed{3} \wedge \boxed{2} = \boxed{4}) \vee (\boxed{1} = \boxed{4} \wedge \boxed{2} = \boxed{3}) \right)$$

References

Abeillé, Anne and Godard, Danièle 1996. La complémentation des auxiliaires français. *Langages*, 122:32–61.

Carpenter, Bob 1992. *The Logic of Typed Feature Structures*. Cambridge University Press. Cambridge, Massachusetts, USA.

King, Paul J. 1999. Towards Truth in Head-driven Phrase Structure Grammar. In Valia Kordoni (ed), *Tübingen Studies in Head-Driven Phrase Structure Grammar*, (= *Arbeitspapiere des SFB 340, Nr. 132, Volume 2*), 301–352. Eberhard-Karls-Universität Tübingen.

Moshier, Michael Andrew 1988. *Extensions to Unification Grammar for the Description of Programming Languages*. PhD thesis, University of Michigan.

Pollard, Carl and Sag, Ivan A. 1994. *Head-Driven Phrase Structure Grammar*. University of Chicago Press.

Pollard, Carl J. 1999. Strong generative capacity in HPSG. In Gert Webelhuth, Jean-Pierre Koenig, and Andreas Kathol (eds), *Lexical and Constructional Aspects of Linguistic Explanation*, 281–297. CSLI Publications.

Przepiórkowski, Adam 1999. *Case Assignment and the Complement-Adjunct Dichotomy: A Non-Configurational Constraint-Based Approach*. PhD thesis, Eberhard-Karls-Universität Tübingen.

Richter, Frank 2000. *A Mathematical Formalism for Linguistic Theories with an Application in Head-Driven Phrase Structure Grammar*. PhD thesis, Eberhard-Karls-Universität Tübingen.

Richter, Frank and King, Paul J. 1997. On the Existence of

Exhaustive Models in a Relational Feature Logic for Head-driven Phrase Structure Grammar. *Note:* Manuscript.

Richter, Frank and Sailer, Manfred 1999a. A lexicalist collocation analysis of sentential negation and negative concord in French. In Valia Kordoni (ed), *Tübingen Studies in Head-Driven Phrase Structure Grammar*, (= *Arbeitspapiere des SFB 340, Nr. 132, Volume 1*), 231–300. Eberhard-Karls-Universität Tübingen.

Richter, Frank and Sailer, Manfred 1999b. LF Conditions on Expressions of Ty2: An HPSG Analysis of Negative Concord in Polish. In Robert D. Borsley and Adam Przepiórkowski (eds), *Slavic in HPSG*, 247–282. CSLI Publications.

Richter, Frank and Sailer, Manfred 1999c. Underspecified Semantics in HPSG. In Harry Bunt and Reinhard Muskens (eds), *Computing Meaning*, (= *Studies in Linguistics and Philosophy*), 95–112. Kluwer academic publishers.

Sailer, Manfred 2000. *Combinatorial Semantics and Idiomatic Expressions in Head-Driven Phrase Structure Grammar. Submitted Version of June 29th, 2000*. PhD thesis, Eberhard-Karls-Universität Tübingen.