Seminar für Sprachwissenschaft
Abteilung Computerlinguistk
Universität Tübingen

SS 2002
June 5th, 2002

Grammar Formalisms and Parsing

Prof. Dr. E. Hinrichs
Frank Richter
Beata Trawiński

In this handout, we will formalize the concepts that we have been using informally. Our goal is to make mathematically precise what a grammar is, and what a grammar means. To achieve that we need to define signatures, the set of descriptions of a signature, feature structures, and the meaning of descriptions and sets of descriptions in a domain of feature structures.

1 Syntax

We begin with defining the essential components of grammars that we extracted from the appendix of [Pollard and Sag, 1994]. As we saw in our work with MoMo, the *Partitions* and the *Feature Declarations* of Pollard and Sag can be interpreted as belonging to what is more traditionally called the *Signature* of a logical language. When we will first define signatures for HPSG grammars, we will not include everything that is needed to formalize a typical HPSG grammar entirely. Instead, we start with a fragment that will allow us to explore constraint-based grammars and their meaning and get a fuller understanding of how they work. Then we will augment our technical machinery in a second pass through the mathematical foundations of HPSG.

To distinguish the partial signatures that we start with from later signatures, we call them *initial* signatures:

Definition 1 Σ *is an* ***initial signature*** *iff*

Σ *is a quintuple* $\langle \mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F} \rangle$,

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

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

$\mathcal{A}$ *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* $\mathcal{F}(\langle \sigma_2, \alpha \rangle) \sqsubseteq \mathcal{F}(\langle \sigma_1, \alpha \rangle)$.

I call each element of $\mathcal{G}$ a *sort*, $\langle \mathcal{G}, \sqsubseteq \rangle$ the *sort hierarchy*, each element of $\mathcal{S}$ a *species*, each element of $\mathcal{A}$ an *attribute*, and $\mathcal{F}$ the *appropriateness function*. If for two sorts σ_1 and σ_2,

$\sigma_2 \sqsubseteq \sigma_1$, then I say that σ_2 is *at least as specific as* σ_1. Alternatively, I say that σ_1 *subsumes* σ_2, or σ_2 is a *subsort* of σ_1. The condition on $\mathcal{F}$ enforces attribute inheritance: If an attribute α is appropriate to some sort σ_1, then it is also appropriate to all subsorts of σ_1, and the value of $\mathcal{F}$ at the subsorts of σ_1 and α is at least as specific as $\mathcal{F}(\langle\sigma_1, \alpha\rangle)$. Note that the definition of an initial signature does not enforce the presence of a top or bottom sort in the sort hierarchy, and leaves the decision to include them or to omit them to the grammar writer.

Based on the definition of initial signatures, we can now define the notion of descriptions relative to a given initial signature:

Definition 2 *For each initial signature* $\Sigma = \langle \mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F} \rangle$, $\mathcal{D}^\Sigma$ *is the smallest set such that*

for each $\sigma \in \mathcal{G}$,

$\sigma \in \mathcal{D}^\Sigma$,

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

$\alpha : \delta \ \in \mathcal{D}^\Sigma$,

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

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

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

for each $\delta_1 \in \mathcal{D}^\Sigma$, *for each* $\delta_2 \in \mathcal{D}^\Sigma$, $(\delta_1 \texttt{*>} \delta_2) \in \mathcal{D}^\Sigma$, *and*

for each $\delta_1 \in \mathcal{D}^\Sigma$, *for each* $\delta_2 \in \mathcal{D}^\Sigma$, $(\delta_1 \texttt{<*>} \delta_2) \in \mathcal{D}^\Sigma$.

I call each element of $\mathcal{D}^\Sigma$ an *initial* Σ *description*. Obviously, our syntax of initial Σ descriptions is not sufficient yet to notate all principles of a typical HPSG grammar of a natural language. For example, there is no syntax to write down the boxed integers called *tags*, and there are no relation symbols. Once we will have understood the meaning of grammars in the given fragment of a syntax for HPSG, we will extend it by missing elements of a complete syntax and semantics. For the moment, we build our definition of *initial grammars* on initial signatures and descriptions. In DEFINITION 3, we use the insight that the principles of an HPSG grammar are a set of descriptions. We use the Greek letter θ to refer to a set of initial Σ descriptions:

Definition 3 Γ *is an* ***initial grammar*** *iff*

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

Σ *is an initial signature, and*

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

An initial grammar is simply an initial signature, Σ, together with a set of initial Σ descriptions.

2 Meaning

2.1 Introduction

When we define the meaning of grammars, we first have to decide what the objects in the denotation of grammars are supposed to stand for. In the evolution of the HPSG formalisms in the tradition of [Pollard and Sag, 1994] different answers were given to this fundamental question.

Pollard and Sag themselves envisioned a formalism in which grammars are about the object types of a language. A grammar admits a collection of object types, which are constructed as a collection of a specific kind of feature structures. [King, 1999] presented a formalism in which a grammar delineates a class of what King calls *exhaustive models*. The exhaustive models of a grammar are mutually indiscernible, and one of them contains exactly all possible tokens of the natural language that the grammar is about. [Pollard, 1999] presented a formalization of HPSG in which the *strong generative capacity* of a grammar is defined as an isomorphism class of structures that are mathematical idealizations of natural language tokens. Neither the formalization of King nor the formalization of Pollard uses feature structures.[1]

The crucial difference between the three approaches is their differing views about how the objects in the denotation of a grammar (or a scientific theory in general) are related to the empirical domain of observable and measurable data. Grossly oversimplifying, one might say that Pollard and Sag believe that knowledge of language is knowledge about the object types of a language, object types are real objects, and a theory of grammar should include mathematical entities that function as models of object types. King denies that, at the current stage of linguistic research, object types are a scientifically useful concept, and claims that a grammar should talk directly about the domain of observable data, which are language tokens. Pollard abondons Pollard and Sag's claim about the ontological reality of object types but maintains the view that grammars should be about mathematical models of language that do not contain structurally isomorphic members. They contain exactly one structurally isomorphic member for each utterance token that will be judged grammatical.

A choice between these three views about the meaning of grammars is first of all a choice between different views of the nature of scientific theories. This choice has nothing to do with technical matters; it is about philosophical arguments. As for the underlying mathematics, [Richter, 2000] shows how all three views can be integrated within one unifying formalism of HPSG, and how the three different modelling domains are mathematically related. In effect, it is possible to switch arbitrarily from any one of the models suggested above to the others. From a mathematical point of view, feature structure models are just a very specialized kind of model; the model theory of King is more general and follows standard ideas of logical models.

When we use feature structures in our formalization of the meaning of grammars, we do not express a preference as to which idea about the nature of scientific theories is correct. Our decision is purely pragmatic in nature: Computational theories of constraint-based grammars traditionally relied on feature structure models of grammars and on processing methods for feature structures. In a computational context, feature structure models are no more than an efficient tool for computational purposes. On a theoretical level we know that feature structure

[1] Pollard still calls the structures in the strong generative capacity of grammars feature structures. However, his feature structures are construed differently from any kind of classical feature structures, and they differ from them in some quite important properties.

models are mathematically closely related to a precise version of each of the other views about the meaning of grammars. Thus, we know that feature structure models of grammars can always be reinterpreted in light of the other two model-theoretic interpretations of grammars.

2.2 Technicalities

Our first understanding of the meaning of descriptions and grammars in MoMo was based on a notion of totally well-typed and sort resolved *feature structures* that we extracted from Pollard and Sag's book. To be able to define the meaning of initial grammars with respect to feature structures, it is now necessary to give an exact definition of feature structures. Guided by the remarks on feature structures of [Pollard and Sag, 1994], we define them as totally-well typed, sort resolved and *abstract* feature structures. They are *abstract* because they are built in such a way that two structurally isomorphic feature structures are necessarily identical, which captures one of the intuitions underlying Pollard and Sag's concept of feature structure models: The members of the set of feature structures admitted by a grammar are mathematical representations of the object types of a language. By contrast, the feature structures pictured in MoMo correspond to so-called *concrete* feature structures: Two structurally identical concrete feature structures may well be different entities.[2]

DEFINITION 4 introduces our abstract feature structures as *simple* abstract feature structures. They are called *simple*, because they will have to be augmented along with the syntax of the description language in order to achieve a complete formalization of a feature structure-based interpretation of HPSG.

Definition 4 *For each initial signature* $\Sigma = \langle \mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F} \rangle$, $\mathbb{A}$ *is a* ***simple abstract feature structure under*** Σ *iff*

$\mathbb{A}$ *is a triple* $\langle \beta, \varrho, \lambda \rangle$,

$\beta \subseteq \mathcal{A}^*$,

$\varepsilon \in \beta$,

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* $\mathcal{S}$, *and*

for each $\pi_1 \in \mathcal{A}^*$, *for each* $\pi_2 \in \mathcal{A}^*$,

if $\langle \pi_1, \pi_2 \rangle \in \varrho$ *then* $\lambda(\pi_1) = \lambda(\pi_2)$.

[2] Concrete feature structures are thus closer to the concept of utterance tokens. There are, however, subtle differences.

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* $\pi\alpha \in \beta$,

If Σ is an initial signature, I write $\mathbb{AFS}^{\Sigma}$ for the *set of simple abstract feature structures under* Σ. If $\mathbb{A} = \langle\beta, \varrho, \lambda\rangle$, I call β the *basis set in* $\mathbb{A}$, ϱ the *re-entrancy relation in* $\mathbb{A}$, and λ the *label function in* $\mathbb{A}$. The basis set in $\mathbb{A}$ is a prefix closed set of paths, the re-entrancy relation in $\mathbb{A}$ is a right invariant equivalence relation on the basis set in $\mathbb{A}$, and the label function in $\mathbb{A}$ is a total function from the basis set in $\mathbb{A}$ to the set of species, $\mathcal{S}$, which respects the re-entrancy relation in $\mathbb{A}$. In other words, if two paths in β are re-entrant then they are assigned the same species.

Definition 5 *For each initial signature* $\Sigma = \langle\mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F}\rangle$, *for each* $\mathbb{A} = \langle\beta, \varrho, \lambda\rangle \in \mathbb{AFS}^{\Sigma}$, *for each* $\pi \in \mathcal{A}^*$,

$$\begin{aligned}
&\beta/\pi = \{\pi' \in \mathcal{A}^* \mid \pi\pi' \in \beta\},\\
&\varrho/\pi = \{\langle\pi_1, \pi_2\rangle \in \mathcal{A}^* \times \mathcal{A}^* \mid \langle\pi\pi_1, \pi\pi_2\rangle \in \varrho\},\\
&\lambda/\pi = \{\langle\pi', \sigma\rangle \in \mathcal{A}^* \times \mathcal{S} \mid \langle\pi\pi', \sigma\rangle \in \lambda\}, \textit{ and}\\
&\mathbb{A}/\pi = \langle\beta/\pi, \varrho/\pi, \lambda/\pi\rangle.
\end{aligned}$$

If a Σ path π is in the basis set of a simple abstract feature structure under Σ, $\mathbb{A}$, then the π reduct of $\mathbb{A}$ is also a simple abstract feature structure under Σ:

Proposition 1 *For each initial signature* $\Sigma = \langle\mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F}\rangle$, *for each* $\mathbb{A} = \langle\beta, \varrho, \lambda\rangle \in \mathbb{AFS}^{\Sigma}$, *for each* $\pi \in \mathcal{A}^*$,

if $\pi \in \beta$ *then* $\mathbb{A}/\pi \in \mathbb{AFS}^{\Sigma}$.

The simple abstract feature structure satisfaction function in Σ, Δ_Σ, is defined inductively over all initial Σ descriptions:

Definition 6 *For each initial signature* $\Sigma = \langle\mathcal{G}, \sqsubseteq, \mathcal{S}, \mathcal{A}, \mathcal{F}\rangle$, Δ_Σ *is the total function from* $\mathcal{D}^{\Sigma}$ *to* $Pow(\mathbb{AFS}^{\Sigma})$ *such that*

for each $\sigma \in \mathcal{G}$,

$$\Delta_\Sigma(\sigma) = \left\{ \langle\beta, \varrho, \lambda\rangle \in \mathbb{AFS}^{\Sigma} \;\middle|\; \begin{array}{l} \textit{for some } \sigma' \in \mathcal{S},\\ \quad \langle\varepsilon, \sigma'\rangle \in \lambda, \textit{ and}\\ \quad \sigma' \sqsubseteq \sigma \end{array} \right\},$$

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

$$\Delta_\Sigma(\alpha : \delta) = \left\{ \langle\beta, \varrho, \lambda\rangle \in \mathbb{AFS}^{\Sigma} \;\middle|\; \begin{array}{l} \alpha \in \beta, \textit{ and}\\ \langle\beta, \varrho, \lambda\rangle/\alpha \in \Delta_\Sigma(\delta) \end{array} \right\},$$

for each $\delta \in \mathcal{D}^{\Sigma}$, $\Delta_{\Sigma}(\sim \delta) = \mathbb{AFS}^{\Sigma} \backslash \Delta_{\Sigma}(\delta)$,
for each $\delta_1 \in \mathcal{D}^{\Sigma}$, *for each* $\delta_2 \in \mathcal{D}^{\Sigma}$, $\Delta_{\Sigma}(\delta_1, \delta_2) = \Delta_{\Sigma}(\delta_1) \cap \Delta_{\Sigma}(\delta_2)$,
for each $\delta_1 \in \mathcal{D}^{\Sigma}$, *for each* $\delta_2 \in \mathcal{D}^{\Sigma}$, $\Delta_{\Sigma}(\delta_1; \delta_2) = \Delta_{\Sigma}(\delta_1) \cup \Delta_{\Sigma}(\delta_2)$,
for each $\delta_1 \in \mathcal{D}^{\Sigma}$, *for each* $\delta_2 \in \mathcal{D}^{\Sigma}$, $\Delta_{\Sigma}(\delta_1 \texttt{*>} \delta_2) = \left(\mathbb{AFS}^{\Sigma} \backslash \Delta_{\Sigma}(\delta_1)\right) \cup \Delta_{\Sigma}(\delta_2)$, *and*
for each $\delta_1 \in \mathcal{D}^{\Sigma}$, *for each* $\delta_2 \in \mathcal{D}^{\Sigma}$,
$\Delta_{\Sigma}(\delta_1 \texttt{<*>} \delta_2) = \left(\left(\mathbb{AFS}^{\Sigma} \backslash \Delta_{\Sigma}(\delta_1)\right) \cap \left(\mathbb{AFS}^{\Sigma} \backslash \Delta_{\Sigma}(\delta_2)\right)\right) \cup \left(\Delta_{\Sigma}(\delta_1) \cap \Delta_{\Sigma}(\delta_2)\right)$.

I call Δ_{Σ} the *simple abstract feature satisfaction function under* Σ, and say $\mathbb{A}$ satisfies δ in Σ if and only if $\mathbb{A} \in \Delta_{\Sigma}(\delta)$.

Definition 7 *For each initial signature* Σ,

M_{Σ} *is the total function from* $Pow(\mathcal{D}^{\Sigma})$ *to* $Pow(\mathbb{AFS}^{\Sigma})$ *such that for each* $\theta \subseteq \mathcal{D}^{\Sigma}$,

$$M_{\Sigma}(\theta) = \left\{ \langle \beta, \varrho, \lambda \rangle \in \mathbb{AFS}^{\Sigma} \;\middle|\; \begin{array}{l} \textit{for each } \pi \in \beta, \textit{ and} \\ \textit{for each } \delta \in \theta, \\ \quad \langle \beta, \varrho, \lambda \rangle / \pi \in \Delta_{\Sigma}(\delta) \end{array} \right\}.$$

For each initial signature Σ, I call M_{Σ} the *simple abstract feature structure admission function in* Σ. A simple abstract feature structure under Σ, $\mathbb{A}$, is in the set of simple abstract feature structures under Σ admitted by a Σ theory θ exactly if $\mathbb{A}$ and every possible π reduct of $\mathbb{A}$ satisfy every description in θ.

Given an initial grammar, $\langle \Sigma, \theta \rangle$, we can regard the set of feature structures admitted by θ, $M_{\Sigma}(\theta)$ as the intended meaning of $\langle \Sigma, \theta \rangle$. Pollard and Sag say that the members of $M_{\Sigma}(\theta)$ are mathematical representations of the object types of the language that the grammar is about.

References

[King, 1999] 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.

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

[Pollard, 1999] 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.

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