MODULARITY AND SOFT CONSTRAINTS
A STUDY OF CONFLICT RESOLUTION IN GRAMMAR

by

Mohammad Haji-Abdolhosseini

A thesis submitted in conformity with the requirements for the degree of
Doctor of Philosophy
Graduate Department of Linguistics
University of Toronto

© Copyright by Mohammad Haji-Abdolhosseini, 2005

Modularity and Soft Constraints

A study of conflict resolution in grammar

Doctor of Philosophy 2005
Mohammad Haji-Abdolhosseini
Department of Linguistics
University of Toronto

This thesis advocates a modular and parallel grammar architecture with declarative constraints on the syntactic, semantic, prosodic, and pragmatic structures which are derived in parallel while mutually constraining one another as proposed by Jackendoff (1997, 2002). The main claim of this thesis is that because of the many conflicting requirements among modules, the interfaces cannot employ crisp constraints. Instead, a soft-constraint satisfaction approach is required. We also argue that simply violable constraints are insufficient to account for certain linguistic phenomena; there is need for graded constraints that allow for degrees of violation.

The dissertation first provides a review of different conceptions of gradience in linguistics followed by a review of the concept of modularity in cognitive science and linguistics. The problem of conflicting requirements in the field of Constraint Logic Programming (CLP) has led to various soft constraint satisfaction approaches. The dissertation then presents a generalized theory of soft constraint satisfaction (Bistarelli, 2001) from the CLP literature.

The dissertation then presents a case study of graded constraints showing that such constraints exist at interfaces and that they can exhibit degrees of violation. Another case study shows that the modular parallel architecture allows for simpler modules and is able to capture generalizations better. We then conclude by showing how the generalized theory of soft-constraint satisfaction can be incorporated within grammar without disrupting the existing explanatory power of constraint-based theories such as LOT (Keller, 2000) and HPSG (Pollard and Sag, 1994).

To Shadi and Nima

Acknowledgments

I spent much of the last four years of my life alone at RL-14221, a windowless room on the 14th floor of Robarts Library, which I fondly call the *Linguistics Superfix* because it is one of a few rooms that the Linguistics Department has on the 14th floor, the rest of it being on the sixth where all the fun is. But I did not or could not write this thesis alone by any means. If it weren't for the strong support of almost everyone I have interacted with while at UofT, none of this would have been possible. First and foremost, I'd like to thank my co-supervisors Elizabeth Cowper and Gerald Penn. Elizabeth was one of my first points of contact with formal linguistics even before I came to UofT (I studied her Government and Binding textbook in Iran). She was there when I started learning linguistics; she was there when I started tearing it apart and turning it upside down. She has always welcomed me with her pleasant smile (which you don't really expect from a Karate black-belt) even during the busiest and most difficult times of her life; she has always listened to whatever I had to say about anything from linguistics, to teaching, to family life, and to doubting pretty much everything about myself. And she's always had something supportive, insightful, useful and funny to say. Gerald dared to trust that shy guy from Linguistics who attended the meetings of the Computational Linguistics group in the Department of Computer Science, and hired him as his assistant. What I learned about computational linguistics working with Gerald, I couldn't have learned in a million courses... okay, maybe not exactly a million, but I did learn quite a bit. I enjoyed every word of our discussions whether they took place in his office or in the Second Cup across the street on College Street, or in some restaurant that served vegetarian food. Both Elizabeth and Gerald are true scholars, great teachers and fantastic friends. When your supervisors confide in you, invite you to their homes, invite you to their wedding, serve as your personal chauffeur, play with you in a band and share your weird sense of humour, you know they are more than just your supervisors; they're friends, and I cherish that

more than anything.

In addition, I must thank Elan Dresher for his informative comments on this thesis and for letting me hack his Prolog code during my first summer at UofT. Dave McKercher also deserves mention here for serving as the reader of my second generals paper, Chapter 6 of this dissertation is based upon. I'm also grateful to Frank Keller and Jean-Pierre Koenig for their insightful comments and astute observations on this work. This dissertation owes a lot to these people, but that does not mean that they all agree with everything I say here. I take responsibility for all the shortcomings of this work.

In addition, I'd like to thank Diane Massam with whom I took numerous courses and for whom I worked on a Niuean project. Thanks, Diane. I learned a lot and truly enjoyed working with you. Also I must thank Peter Reich, my teaching mentor, for giving me a lot of opportunities to learn about teaching and cognitive science. He was also a great neighbour on the 14th floor. I should also thank Keren Rice for giving me the opportunity to work for her on several occasions and for just being such a kind and delightful person. I should also thank Jack Chambers, my newest neighbour in the Linguistics Superfix, for his support and encouragement. Thanks, Jack. You're an inspiration.

My fellow graduate students deserve many thanks as well. Thank you, Kenji Oda for being such a good friend and office mate. Also, thanks to Arsalan Kahnemuyipour for his support with all my concerns during my years at UofT. I should also thank Carlos Ramirez for being a great friend. Carlos, Kenji, and I shared many happy hours in the department, in the neighbourhood bars, on Wasaga Beach and, believe it or not, on more than one occasion, at Hart House Gym. Other past and present graduate students whom I'm proud to call my friends include Abdel-Khalig Ali, Susana Béjar, Alex D'Arcy, Chiara Frigeni, Alexei Kochetov, Manami Hirayama, and Jack Panster just to name a few.

Of course I wouldn't even be here if it weren't for my mother Fatemeh Haj-Agha-Mohammad and my father Akbar Haji-Abdolhosseini. I owe them for all their care, love and support. I also should thank my son, Nima, for bringing so much love, joy and meaning to my life. And most of all, I should thank my loving wife Shadi Kiaee, who put most of her life dreams on hold to support me every which way she could in the past six years. I'm grateful to her and proud of all the achievements she has made during this time.

This research has been partially funded by Social Sciences and Humanities Research Council of Canada doctoral fellowships and Ontario Graduate Scholarships.

Mohammad Haji-Abdolhosseini
June, 2005

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Words should be weighed not counted.
—CHINESE FORTUNE COOKIE

1.1 Motivation

Inquiry in theoretical linguistics, cognitive science, and AI has led many researchers to believe that constraint-based approaches in modelling human behaviour capture our understanding of the phenomena in question better than procedural approaches. The advantage of these approaches is that expressing what we know about the data in the form of constraints can capture generalizations more accurately and intuitively. A procedural formalization has the disadvantage of mixing *knowledge* with the *processing* of that knowledge. By keeping the two separate, we give ourselves the opportunity to improve each separately. Newell (1982) calls this way of representing knowledge *the knowledge level accounting of skill* in the context of human problem solving. In the field of artificial intelligence, problems have been found to be characterized best if one thinks of them as objects that have certain constraints enforced on their interactions. In linguistics, Bird (1990, 1995), and Bird and Klein (1994) argue that a constraint-based approach to phonology can capture linguistic generalizations better while procedural approaches sometimes overlook such generalizations because of their focus on rule ordering and symbol manipulation. Constraint-based linguistic theories have been making headway in better understanding of language. Two noteworthy examples are Head-Driven Phrase Structure Grammar (HPSG, Pollard and Sag, 1987, 1994), and Lexical Functional Grammar (LFG, Bresnan, 1982, 2001). Another remarkable example is Optimality Theory (OT, Prince and Smolensky, 1993). The underlying assumption in

this theory is that constraints are not absolute (or as some put it *crisp*), they are violable. Constraints are thus ranked according to their importance, and the form that violates the fewest high-ranking constraints is considered the optimal form. Constraint-based systems are also widely used to solve real-life problems in computer science. Network management, scheduling, and transportation problems are most easily solved in a constraint-based approach.

As mentioned above, in cognitive science and AI, problems are envisioned as constituting discrete objects with constraints imposed on their interactions. A keyword in the preceding statement is *objects* as objects are more or less independent entities with certain properties and they perform a set of predefined functions. Several theories in cognitive science have found modular approaches beneficial. For example, Newell's (1990) unified theory of cognition paints a modular picture of the mind in which each cognitive faculty takes the form of a discrete entity that communicates with other modules. Oatley and Johnson-Laird (1987) and Oatley (1992) develop a theory of emotions within a larger context of cognitive science. This theory also relies on the fundamental assumption that human cognitive processes are modular and need to communicate with one another. On more familiar turf, Jackendoff (1992) also argues that human cognitive faculties form modules that need to communicate with one another. He also believes that each module (e.g., language, vision, or musical perception) is itself made up of its own sub-modules which in turn communicate with one another. Zooming in on language, Jackendoff (1997, 2002) argues for a tripartite architecture of grammar where phonological, morpho-syntactic and semantic components work in parallel and communicate at interface levels (see section 3.4 and Figure 3.1).

The modular view of cognition and the constraint-based account of knowledge are more or less widely accepted in cognitive sciences. What is still a matter of debate is the nature of the constraints, and the mechanisms involved in the communication among modules. Disagreement among modules in any given intelligent system is a fact of life. Having crisp constraints, therefore, is not considered a desirable feature because as we gradually grow out of toy models and move towards approximating real-life problems, a system with crisp constraints quickly turns into what is known as an *overconstrained* system; i.e., one that yields no solution; or it becomes so complicated that it takes the system an inordinate amount of time to find an answer. In the AI community, several approaches have been proposed to remedy such problems. *Partial constraint satisfaction* (Freuder and Wallace, 1992), *constraint hierarchies* (Borning et al., 1992), *probabilistic soft constraint satisfaction* (Fargier and Lang, 1993), *valued constraint satisfaction* (Schiex et al., 1995), and *fuzzy soft constraint satisfaction* (Rosenfeld et al., 1976; Dubois et al., 1993; Ruttkay,

1994) are most notable.[1]

In computational linguistics, probabilistic approaches are dominant, and have led to some theoretical contributions (Abney, 1996, 1997; Bod, 1998; Bod, Hay, and Jannedy, 2003; Bod, Scha, and Sima'an, 2003; Foth, Menzel, and Schröder, 2005; Schröder, 2002, among others). The only approach within linguistics proper that relies on non-crisp constraints is OT. As I will argue later, however, the OT approach does not sufficiently capture linguistic phenomena and we need a more sophisticated constraint system.

Jackendoff (1997; 2002), who is a proponent of a modular approach, introduces *correspondence rules* that apply at the interfaces. These rules are phrased as non-crisp constraints. For example, his *phonological-syntactic correspondence rule* is given in (1.1):

(1.1) a. *General form for phonological-syntactic correspondence rules (PS-SS rules)* (p. 28)
Syntactic structure X {must/may/preferably does} correspond to phonological structure Y.

b. i. A syntactic X^0 constituent preferably corresponds to a phonological word.

ii. If syntactic constituent X_1 corresponds to phonological constituent Y_1,
and syntactic constituent X_2 corresponds to phonological constituent Y_2,
then the linear order of X_1 and X_2 preferably corresponds to the linear order of Y_1 and Y_2.

The problem here is that there is no mention of how strong these preferences are. As will become evident later, we not only need to specify exactly the degree of these preferences; we also have to find ways of evaluating the degree to which a constraint is complied with as this plays a role in determining the actual prosody and word order as well.

Based on a case study of pragmatic and prosodic constraints on word order, this dissertation argues for a modular grammar architecture that implements a theory of soft-constraint satisfaction at interface levels. It is argued that besides approximating linguistic phenomena more closely, this approach paints a picture of the language faculty more in tune with what is known about other cognitive faculties. The data considered in this dissertation involve sentences with or without canonical word order partly due to pragmatic and/or prosodic constraints.

[1]Some useful literature reviews can be found in Bistarelli (2001) and at `http://kti.ms.mff.cuni.cz/~bartak/constraints/`.

From a practical point of view, Haji-Abdolhosseini (2003a,b) argues that there are also other reasons why it is important to do research in grammatical interfaces in constraint-based and multi-partite frameworks: A modular theory is easier for the researcher to work with. A grammar written in this approach is certainly more readable and more convenient to maintain. Furthermore, with the emergence of large-scale grammars a modular approach becomes even more significant to promote code readability and reuse. This latter point is of course an engineering desideratum rather than a scientific criterion; however, we take the convergent approaches from linguistic theory and grammar engineering as a welcome state of affairs.

1.2 Thesis

This dissertation relies on the following working assumptions:

- A theory of the language faculty is modular.
- The modules operate independently of and in parallel with one another.
- The modules constrain one another's operation.

The thesis of this dissertation is as follows:

- Soft constraints mediate the communications among modules.
- Soft constraints help capture intuitions about markedness as part of linguistic knowledge.

1.3 Structure of the Dissertation

This dissertation comprises eight chapters. Chapter 2 reviews the various conceptions of gradient linguistic phenomena in scholarly literature today. It helps delineate the scope of this research, and pinpoints its place in the larger picture. Chapter 3 reviews the notion of modularity in cognitive science and linguistics. It helps distinguish different interpretations of this concept, and defines what we mean by that in this dissertation. Chapter 4 presents an overview of a generalized

theory of *Soft Constraint Satisfaction*. This theory plays an important role in the model proposed here. Chapter 5 presents a case study of soft constraints showing that the traditional constraint-based frameworks cannot handle the phenomenon discussed in that chapter. It also builds and evaluates a statistical model of the phenomenon discussed. Chapter 6 takes a deeper look into the problem of conflicts within the grammar modules. It demonstrates that a parallel modular grammar architecture allows for simpler modules and it helps capture linguistic generalizations better. Chapter 7 demonstrates an implementation of soft intermodular constraints. It also shows how the treatment of soft constraints as discussed in Chapter 5 can be thought of as a natural extension of the present theories without any disruption of their existing mechanisms. A discussion of the contributions of the dissertation and some directions for future research are then presented in Chapter 8.

Chapter 2

Types of Gradience in Grammar

2.1 Introduction

Interestingly, the word *gradience* is not even listed in the Oxford English Dictionary, but it has often been used by linguists (e.g., Aarts, 2004a,b; Aarts et al., 2004; Bolinger, 1961; Keller, 2000) to refer to any gradient phenomenon in language. In general, *gradience* is a cover term for a range of linguistic phenomena that defy discrete categorization. This chapter reviews what different researchers have called gradient phenomena in order to classify these concepts, and then state what kind of gradience it is that the present dissertation focuses on.

2.2 Historical Background

This subsection is a summary of Aarts (2004a) which contains a broad historical account of the conceptions of gradience in most major linguistic camps. For more detailed information, the reader is referred to Aarts (2004a,b) and Aarts et al. (2004) as well as the references cited therein. Aarts says that gradience in grammar is usually characterized "as the phenomenon of blurred boundaries between two categories of form classes α and β, with a third group of elements belonging to the middle ground between the two categories" (p. 344). In set-theoretic terms, Aarts presents a working definition of gradience quoted below as (2.1) (ibid.).

(2.1) [Gradience] occurs if there exists between α and β an intersection $\alpha \cap \beta$, containing elements that possess α-like features as well as β-like features.

Aarts (2004a) starts his account of gradience with Aristotle whose system of categorization was "rigidly all-or-none." He then turns to the notion of *vagueness*

in philosophy and states that *vagueness* can especially be useful in capturing the indeterminacy that lies in the Sorites Paradox (Paradox of the Heap). This paradox comes about when it is not clear where p becomes $\neg p$ in a chain of elements $\alpha_1, \ldots, \alpha_n$. The best-known example of this paradox comes in the form of the question, "when does a collection of grains of sand become a heap?"

Classical grammarians, in Aarts' words, were "inveterate categorizers" (The term *category* in this context refers solely to parts of speech). Later, from the time of the Renaissance, however, scholars began to doubt categories and some even went so far as to reject them altogether, while some devised part-of-speech systems that deviated from the classical Graeco-Roman tradition. In the eighteenth century, we see the early emergence of the idea of prototypes.

Within the structuralist tradition, Sapir and Bloomfield allowed for non-discrete category membership, but post-Bloomfieldian structuralists, like Joos and Hockett, adamantly excluded gradience from the study of language because they thought that continuity was just not part of language design. Joos (1950) (reprinted in Aarts et al. (2004)) denies the existence of gradation or continuity "in either form or meaning." Following this period, from the late 1950s, the structuralist conviction about the non-existence of gradience in linguistic categorization was challenged. One example of the work in this period is Bolinger (1961).

In his section on transformational grammar, Aarts likens early generativists to structuralists in accepting the existence of gradient phenomena in *language use* and not in the *language system*. Early generativists, like structuralists, did not deny the existence of gradient phenomena but they postponed their study until such time as we had a better understanding of the language system, an approach which Aarts analogizes to "abstract[ing] away from the wood in order to see the trees" (p. 352). Having mentioned Chomsky's emphasis on idealization, Aarts argues, "[d]espite dismissing performance phenomena as being outside language in the narrow sense, gradience *has* played a role in Chomskyan linguistics, specially in early discussion of the notion of 'degrees of grammaticalness'" (p. 353, emphasis in the original) (see Chomsky, 1955, 1961, 1965; Chomsky and Miller, 1963). Chomsky (1955, p. 129) states that

> a partition of utterances into just two classes, grammatical and non-grammatical, will not be sufficient to permit the construction of adequate grammars in terms of what we have broadly described as distributional analysis.

Graded grammaticality continues to play a role in later stages of generative linguistics (see Erteschik-Shir and Lappin, 1979 on picture NPs as well as Belletti and Rizzi, 1988; and Andrews, 1990 on different levels of grammaticality). As

some more recent treatments of gradience in generative linguistics, Aarts mentions Chomsky (1995) for his use of [±strong] features—as strength is a scalable attribute—as well as Pinker (1999) (on irregular verbs), Borsley and Kornfilt (2000) (on mixed extended projections), together with van Riemsdijk (1998, 1999) and Corver and van Riemsdijk (2001) (on semi-lexical categories).

Among the many other researches that Aarts surveys in his paper, Ross' (1969a; 1969b; 1972; 1973a; 1973b; 1974; 1987; 2000) and Radford's (1976) work on category conflation ("squishes"), as well as Lakoff's (1973a; 1973b; 1987a; 1987b) and McCawley's (1977; 1982; 1998) work on fuzzy syntactic categories within the framework of generative semantics are also noteworthy.

In Lexical Functional Grammar (LFG), Bresnan (1997) treats mixed categories as head-sharing constructions such that at c-structure (constituent structure) the items in question occupy different head positions but share the same head at f-structure (functional structure). In Head-Driven Phrase Structure Grammar (HPSG), Malouf (2000) and Hudson (2003) treat *gerunds* as multiply inheriting their properties from both *noun* and *verbal* types (see section 2.6 for a discussion).

Relying on corpus data, Manning (2003) advocates a probabilistic approach towards categorization and strength of constraints in a cross-linguistic perspective; whereas, Keller (2000), working within the framework of Optimality Theory (OT), uses relative constraint weights and the number of violations of constraints in order to approximate human subjects' graded grammaticality judgements obtained through psycholinguistic tests.

2.3 Categorizing Gradience

The definition in (2.1) is deliberately stated very vaguely in order to accommodate all the different conceptions of gradience discussed by Aarts (2004a). Aarts (2004b), on the other hand, attempts to classify different types of gradience and provide a formal definition for each class. Aarts distinguishes two types of gradience: *intersective* and *subsective*. *Intersective Gradience* (IG) refers to "an inter-categorial phenomenon which is characterized by two form classes 'converging' on each other" (p. 1). *Subsective Gradience* (SG) is an intra-categorial phenomenon which allows "for members of a class to display the properties of that class to varying degrees" (ibid.). Aarts argues that true IG is rare (if at all existent) in language; whereas, SG is quite common. The formal definitions of IG and SG are given in (2.2) and (2.3).[1]

[1] In (2.2), it seems that the term "grammatical formative" refers to lexical categories. Aarts' switch in terminology here is not clear to the author.

(2.2) **Intersective Gradience (adapted from Aarts, 2004b, p. 31)**

If	α, β are form classes characterized by syntactic properties $\{\mathrm{a_1}, \ldots, \mathrm{a_m}\}$ and $\{\mathrm{b_1} \ldots \mathrm{b_n}\}$, respectively;
and	$\exists\Delta$, Δ a grammatical formative, which conforms to a set of syntactic properties $\{\mathrm{c_1}, \ldots, \mathrm{c_p}\}$;
such that	$\{\mathrm{c_1}, \ldots, \mathrm{c_x}\} \subset \{\mathrm{a_1}, \ldots, \mathrm{a_m}\}$ and $\{\mathrm{c_{x+1}}, \ldots, \mathrm{c_p}\} \subset \{\mathrm{b_1}, \ldots, \mathrm{b_n}\}$
then	α and β are in an intersective gradient relationship with respect to Δ, and its projection ΔP.

(2.3) **Subsective Gradience (adapted from Aarts, 2004b, p. 30)**

If	$\alpha, \beta \in \Gamma$, where Γ is a form class characterized by syntactic properties $\{\mathrm{p_1}, \ldots, \mathrm{p_n}\}$;
and	α is characterized by $\{\mathrm{p_1}, \ldots, \mathrm{p_x}\}$, such that $0 < x \leq n$;
and	β is characterized by $\{\mathrm{p_1}, \ldots, \mathrm{p_y}\}$, such that $0 < y < x$;
then	α and β are in a subsective gradient relationship, such that α is a more prototypical member of Γ than β.

Note that Aarts defines gradience in grammatical categories in terms of their *morpho-syntactic* properties. As for IG, Aarts states that in some cases certain categories become more like other categories due to some *semantic* resemblance. He calls this *weak convergence*. For example, it has been argued that the word *utter*, an adjective, also has adverbial properties because it behaves like an intensifier (e.g., *very*). According to Aarts, *utter* indisputably belongs to the adjective class on distributional grounds. On the contrary, *strong convergence* occurs when an element also displays some morpho-syntactic properties of another class. Based on the above definitions, IG is manifested only through strong convergence, which Aarts argues does not happen in language. There may be cases where a lexical item exhibits properties of more than one lexical category *in different contexts*, but in any given context only the properties of one category are observed. There are no cases, Aarts argues, where an item exhibits properties of more than one category simultaneously (i.e., in the same context).

Aarts stresses that SG has to do with *prototypicality* and not degrees of class membership as is done in fuzzy logic. For Aarts, *thin* is a more prototypical adjective than *utter* because the former can occur in both attributive and predicative positions; it also has the comparative and superlative forms; whereas, the latter only occurs attributively. Nonetheless, they are both adjectives.

Finally, Aarts points out that there is an implicational relationship between SG and IG. All cases of IG, with the exception of true hybridity ($\mathrm{x} = \mathrm{p} - \mathrm{x}$ with reference to (2.2), i.e., when a category inherits the exact same number of properties from two categories), are necessarily SG. This is because all the members of α that

are intersectively gradient with β are away from the core of α. Aarts concludes that IG, therefore, is SG plus strong convergence.

2.4 What about graded grammaticality?

Aarts' formal definitions of gradience ((2.2) and (2.3)) do not say anything about graded grammaticality judgements. Even though Aarts (2004a,b) provides a review of graded grammaticality judgements in the literature, he does not include the topic in his analysis. One can, however, think of graded grammaticality as a case of SG, meaning that if we assume that a perfectly grammatical sentence is a prototypical example of good sentences in a language defined by all the constraints that it abides by, then a marked sentence deviates from the prototypical sentence(s) by the number of constraints that it violates. This position is in line with the approach taken by Keller (2000). But note that the violation of just any constraint does not lead to graded grammaticality. Keller (2000) divides grammatical constraints into hard and soft. The (non-)violation of hard constraints leads to absolute decisions about grammaticality, while the (non-)violation of soft constraints leads to graded grammaticality judgements. Keller measures subjects' grammaticality judgements using the *Magnitude Estimation* (ME) method of Bard, Robertson, and Sorace (1996), which is based on Stevens' (1975) techniques for measuring subjects' judgements of sensory stimuli (for other approaches to measuring grammaticality judgements, see also Cowart, 1997).

Working within the OT framework, Keller expands standard OT so that it accounts for graded grammaticality based on the weights of individual constraints as well as the number of violations of each constraint. Keller's model, which he calls *Linear Optimality Theory* (LOT) also accounts for the "ganging-up" effect of constraints in which the violation of several low-ranking constraints can take over the violation of a single high-ranking one. Keller ranks the constraints based on subjects' gradient judgements; that is, between two constraints A and B, if the violation of A results in a lower acceptability than the violation of B, then A is ranked higher than B.

LOT is so far the only explicitly worked-out theoretical framework that can handle soft constraints in order to capture speakers' intuitions about graded grammaticality. The following section, which is based on pages 251–254 of Keller (2000), presents an overview of the model.

2.5 Linear Optimality Theory in Brief

Keller's extension to OT only affects *HEval*, the function that evaluates the harmony of a candidate, and *Rank*, the ranking component. It does not affect any of the assumptions about the input and the generation function *Gen*, the two components of an OT grammar that determine which structures compete with each other. Also the constraint component *Con*, the formal apparatus for representing constraints and candidates, is unaffected.

The new versions of *HEval* and *Rank* include changes in the way the optimal candidate is computed. They also require a new type of ranking argumentation (a method for establishing constraint ranks from a set of linguistic examples). Keller argues that this type of ranking argumentation is considerably simpler than the one classically assumed in OT. He also shows that well-understood algorithms exist for automating this type of ranking argumentation.

The model of graded grammaticality that Keller develops relies on constraint cumulativity and constraint ranking. He adopts two hypotheses to formalize his findings. The first hypothesis deals with constraint ranking ((2.4)=Keller's (6.1)).

(2.4) **Ranking Hypothesis**
The ranking of linguistic constraints can be implemented by annotating each constraint with a numerical weight representing the reduction in acceptability caused by a violation of this constraint.

The above definition allows us to model speakers' intuitions about the absolute impact of a constraint violation and not just its ranking relative to other constraints. The following quote from Keller explains this best (p. 252, emphasis in the original):

> [T]his notion of constraint ranks as numerical weights is more general than the notion of ranks standardly assumed in Optimality Theory. Standard OT formulates constraint ranks as binary ordering statements of the form $C_1 \gg C_2$, meaning that constraint C_1 is ranked higher than the constraint C_2. Such statements do not make any assumptions regarding *how much* higher the ranking of C_1 is compared to the ranking of C_2... Such information is only available once we adopt a numerical concept of constraint ranking.

Keller computes the overall acceptability of a construction by a simple summation of the weights of the constraints that the structure violates. This will account straightforwardly for the cumulativity of constraint violations. To account for the cumulativity of constraint weights, Keller formulates the Linearity Hypothesis in (2.5) (Keller's (6.2)). He calls this "the core of Linear Optimality Theory."

(2.5) **Linearity Hypothesis**
The cumulativity of constraint violations can be implemented by assuming that the grammaticality of a structure is proportional to the weighted sum of the constraint violations it incurs, where the weights correspond to constraint ranks.

(2.6)–(2.9) formulate a numerical model that makes explicit the hypotheses made in (2.4) and (2.5). This model relates constraint ranks with degrees of grammaticality. (2.6) (Keller's (6.3)) defines the grammar signature. A grammar signature specifies the constraint set and the associated weights for a grammar. Based on a grammar signature, a given candidate structure has a *constraint violation profile* as defined in (2.7) (Keller's (6.4)). The profile specifies which constraints are violated by the structure and how many times they have been violated. Based on definitions (2.6) and (2.7), the harmony of a structure using a simple linear model is defined in (2.8) (Keller's (6.5)) and (2.9) (Keller's (6.6)).

(2.6) **Grammar Signature**
A grammar signature is a tuple $\langle \mathbf{C}, w \rangle$ where $\mathbf{C} = \{C_1, C_2, \ldots C_n\}$ is the constraint set, and $w(C_i)$ is a function that maps a constraint $C_i \in \mathbf{C}$ to its constraint weight w_i.

(2.7) **Violation Profile**
Given a constraint set $\mathbf{C} = \{C_1, C_2, \ldots C_n\}$, the violation profile of a candidate structure S is the function $v(S, C_i)$ that maps S to the number of violations of the constraint $C_i \in \mathbf{C}$ incurred by S.

(2.8) **Harmony**
Let $\langle \mathbf{C}, w_i \rangle$ be a grammar signature. Then the harmony $H(S)$ of a candidate structure S with a violation profile $v(S, C_i)$ is given in (2.9).

(2.9) $H(S) = -\sum_i w(C_i)v(S, C_i)$

> [(2.9)] states that the harmony of a structure is the negation of the weighted sum of the constraint violations that the structure incurs. Intuitively, the harmony of a structure describes its degree of well-formedness relative to a given set of constraints.

Only constraint violations influence the harmony of a structure because constraint weights are assumed to be positive. Constraint satisfactions will not change the harmony of the structure.

Let us now see how harmony relates to grammaticality in such a way that it implements the Linearity Hypothesis (see (2.5)).

Grammaticality in Linear Optimality Theory is defined in terms of the relative harmony of two candidates in the same candidate set as in (2.10) (=Keller's (6.7)).

(2.10) **Grammaticality**
Let S_1 and S_2 be candidate structures in the candidate set $\mathbf{R}$. Then S_1 is more grammatical than S_2 if $H(S_1) > H(S_2)$. This can be abbreviated as $S_1 > S_2$.

Note that based on (2.8) and (2.10), harmony is an absolute notion that describes the *overall well-formedness* of a structure while grammaticality describes the *relative ill-formedness* of a structure compared with another. Grammaticality in this model is only well-defined for two structures that belong to the same candidate set. Definitions (2.10) and (2.11) provide a relative notion of well-formedness in line with the optimality theoretic tradition. The optimal structure in a candidate set is then defined as the one with the highest relative grammaticality (see (2.11)). Constraint rank in LOT is based on the relative weights of two constraints (see (2.12)).

(2.11) **Optimality**
A structure S_{opt} is optimal in a candidate set $\mathbf{R}$ if $S_{opt} > S$ for every $S \in R$.

(2.12) **Constraint Rank**
A constraint C_1 outranks a constraint C_2 if $w(C_1) > w(C_2)$. This can be abbreviated as $C_1 \gg C_2$.

2.6 Discussion

If Aarts is on the right track about categorizing gradience, then a lot of what scholars have called gradient would turn out to be discretely categorizable; the perception of gradience, then, originates from the fact that the more prototypical items in a class appear in more distinctive morphosyntactic environments than the less prototypical ones. But, as Aarts argues, there seems to be little fading of categories from one to the other or instances where an item simultaneously exhibits properties of two categories *in the same context*. There is still the issue of mixed categories discussed by Bresnan (1997), Malouf (2000), and Hudson (2003) (see section 2.2 above). These categories, as the authors have analyzed them, are not evidence of gradience. They do exhibit multiple inheritance of their properties from other categories (what Aarts calls IG), but still the defining features are discrete. A gerund is a noun with a few verbal properties, but since the nominal properties are more than the verbal ones, we still call it a noun. A whale is a mammal with some fish-like properties; it is just an atypical mammal. A catfish is still a fish but also not very typical. Categories are then discrete; prototypicality is not. Categories are

useful for our understanding of our world; yet, we can still divide each category into two or more based on our observations. For example, instead of having fish and mammal, we could have fish, fammal and mammal, and put catfish in the fammal category. But where does this subdivision game stop? It could go on until we have one category per item. The answer depends on what we want to do with our categories; that is, we stop the subdivision game when we think that with any more subdivision our categories will stop making useful or interesting distinctions for us.

Is grammaticality gradient? The answer is the same. It depends on what you want to do with it. There was a time when a simple grammatical/ungrammatical distinction sufficed, but not anymore. We now have a range of grammaticality judgements, and some think that we still need to show how much a sentence marked with "??" is worse than one marked with "?".

All these aside, there are still truly gradient concepts that play very significant roles in language. Prosodic weight, givenness, level of formality or register, time, and scalar predicates are the best examples of these concepts. In Chapter 5, we will look at the role of prosodic weight in determining the order of discourse units in sentences.

2.7 Summary

This chapter presented an overview of the different conceptions of gradience in the linguistics literature and their categorization by Aarts (2004a; 2004b). It also summarized Keller's (2000) LOT framework which handles hard and soft constraints modelling speakers' graded grammaticality judgements.

The concern of the present dissertation is cross-modular soft constraints similar to the ones discussed by Keller (2000). It will be shown in Chapter 5, however, that there is a class of soft constraints, graded constraints, that cannot be handled by Keller's model.

Chapter 3

Modularity

3.1 Introduction

This chapter briefly reviews the concept of modularity in cognitive science and linguistics. Section 3.2 presents a very general philosophical view of modularity arguing that modular systems are more stable, can develop faster and are more easily understood. Section 3.3 discusses the modularity of mind hypothesis put forward by Jerry Fodor (1983). It goes over the main properties of a module in a cognitive system as seen by Fodor, and then outlines some of the objections raised against this view. Section 3.4 introduces Jackendoff's view of representational modularity and his proposed architecture of the language faculty. In conclusion, I will argue that employing hard constraints inside modules and soft constraints among them is a desirable approach leading to more informationally encapsulated modules.

3.2 Simon's Theory of Complex Systems

Simon (1996)[1] lays down the foundations of the sciences of the artificial, be it machines, economies, social structures, theories, or artificial models of cognition. Chapter 8 of that work, "The Architecture of Complexity: Hierarchic Systems," talks about modularity (or as Simon himself calls it, hierarchy) and its significance in any complex system, either natural or artificial. A module, according to Simon's definition, is "a system that is composed of interrelated subsystems, each of the latter being in turn hierarchic in structure until we reach some lowest level of elementary subsystem" (p. 184). The decision of where this lowest level of elementary

[1] The first edition of *The Sciences of the Artificial* was published in 1969.

subsystem lies depends on the goals of the scientist.

Modular systems are "nearly decomposable;" that is, one is able to separate the system into more or less independent and stable subsystems. This property, Simon argues, is vital in the evolution of any complex system including biological or social. If a system is not modular, any change might jeopardize its stability as a whole, while in a modular system, a change may only destabilize a single module. This means that a modular system is more likely to survive. This explains "the observed predominance of hierarchies among the complex systems nature presents to us" (p. 197).

An important property of modular systems, as observed by Simon, is that "[i]ntracomponent linkages are generally stronger than intercomponent linkages" (p. 204) and that "[s]ubparts belonging to different parts only interact in an aggregative fashion—the details of their interaction can be ignored" (p. 207).

> The fact then that many complex systems have a nearly decomposable, hierarchic structure is a major facilitating factor enabling us to understand, describe, and even "see" such systems and their parts. Or perhaps the proposition should be put the other way round. If there are important systems in the world that are complex without being hierarchic, they may to a considerable extent escape our observation and understanding. Analysis of their behavior would involve such detailed knowledge and calculation of the interactions of their elementary parts that it would be beyond our capacities of memory or computation. (ibid.)

3.3 Fodorian Modularity

Fodor (1983, pp. 36–37) defines a *cognitive module* as a domain-specific, innately specified, hardwired, autonomous, informationally encapsulated, and not assembled[2] system. However, he cautions that "the notion of modularity ought to admit of degrees." He says, "[w]hen I speak of a cognitive system as modular, I shall therefore always mean 'to some interesting extent.' "

Fodor divides human cognitive faculties into *input systems* and *central systems*. Input systems are modular; that is, they exhibit domain specificity and they are innate, hardwired, autonomous, informationally encapsulated, and not assembled. He also mentions that these systems are fast and their operation is involuntary. Input systems are those that take information from the outside world, interpret it,

[2]By not assembled, he means not "having been put together from some stock of more elementary subprocesses."

and form a mental representation of it. Vision and linguistic perception are input systems. According to Fodor, the speed of these systems is crucial to the organism's survival and their modular properties ensure this speed. Central systems, on the other hand, are not modular and are slow. Reasoning, scientific theorizing, and belief fixation are such systems. They are not informationally encapsulated since they get information from a variety of sources (i.e., visual, auditory, sensory, etc.), and they are slow. It might take minutes or even hours for someone to solve a problem, for example.

Fodor discusses eight properties of input systems, which we will go over briefly below. The first four properties are essential to the modularity hypothesis and the next four are not. Let us consider the essential properties first.

Domain-specificity means that

> there are highly specialized computational mechanisms in the business of generating hypotheses about the distal sources of proximal stimulations. The specialization of these mechanisms consists in constraints either on the range of information they can access in the course of projecting such hypotheses, or in the range of distal properties they can project such hypotheses about, or, most usually, on both. (p. 47)

Such modules might include, in vision, mechanisms for the perception of colour and shapes, or for the analysis of three-dimensional spatial relations. At a higher level, there might be mechanisms concerned with the visual guidance of bodily motions or with the recognition of faces. In audition, there might be "computational systems that assign grammatical descriptions to token utterances; or ones that detect the melodic or rhythmic structure of acoustic arrays;" (ibid.) or mechanisms for the recognition of voices.

The second property of input systems is that their operation is mandatory. One cannot choose not to understand an utterance in one's native language, and one cannot help interpret the visual input one receives. "You can't hear speech as noise *even if you would prefer to*" (p. 53, original emphasis).

The third property of the input systems is that the mental representations that they compute are accessible to central systems only to a limited degree. This means that subjects do not have (easy) access to intermediate levels of representation that our theories predict. They are only aware of the output of the system.

Speed is the fourth property of input systems according for Fodor. He states that "[i]dentifying sentences and visual arrays are among the fastest of our psychological processes" (p. 61). By fast, Fodor means taking about a quarter of a second. Interestingly, Fodor also mentions that there is evidence that people also understand what they hear quite rapidly. He cites Marslen-Wilson (1973) as providing evidence that "fast shadowers" (people who repeat what they hear as they

are hearing it) understand what they repeat. This point will prove crucial in the next section.

Information encapsulation is closely related to access of the central systems to intermediate representations in the input systems. By information encapsulation, Fodor means lack of feedback from "high-level expectations or beliefs" to certain perceptual mechanisms. There must be some perceptual mechanisms "that compute the structure of a percept largely, perhaps solely, in isolation from background information" (p. 66). Some evidence for this claim comes from the persistence of many perceptual illusions even after the subject finds out that the percept is illusory. Information encapsulation is the essence of modularity, and as Fodor argues, it is key to the fast performance of input systems, which is crucial to the organism's survival. You do not want to impede a reflex by feeding it too much information.

The next four properties of input systems are less essential. The first is that input systems have "shallow outputs." Fodor makes the "highly speculative" suggestion that the fast performance of input systems requires that their input not incorporate a lot of information. For language, Fodor assumes that this output is a syntactic parse of the utterance with no information about the meanings of the lexical items in that utterance. The second less essential property of input systems is that they are associated with fixed neural architecture. The fact is that there are neurological structures associated with the perceptual systems and with language, but this does not necessarily mean that these systems are modular; nor does it mean that a system that is not neurologically localized is non-modular in terms of information encapsulation and speed. The third one is that "[i]nput systems exhibit characteristic and specific breakdown patterns" (p. 99); that is, specific neural circuitry exists for perceptual systems. And the last of these properties, which Fodor hardly gets into, is that "[t]he ontogeny of input systems exhibits a characteristic pace and sequencing" (p. 100). As this hypothesis goes, the modularity theory is completely compatible with the observation that "a great deal of the developmental course of the input system is endogenously determined."

The linguistic module for Fodor (1983) has a Chomskyan form. It only contains syntactic processes. Fodor places semantic and pragmatic aspects of language within the central systems. To him, the linguistic module (on the input side) takes phonetic information and creates a syntactic representation of it, which is then interpreted by the central systems. Linguistic production is supposedly the reverse of this process.

Since the publication of Fodor's monograph in 1983, many scholars have presented evidence for or against the modularity hypothesis. Most of the arguments deal with language and, as Jackendoff (2000) observes, all of those against the modularity hypothesis try to falsify modularity by showing that semantics does influence syntactic parsing. For a representative set of examples, see Altmann

(1987), Clifton and Ferreira (1987), Crain and Steedman (1985), Frazier (1987), and Marslen-Wilson and Tyler (1987). Jackendoff's (2000) approach, however, is to keep the modularity hypothesis but rethink what fits inside the language module.

3.4 Jackendoff's Representational Modularity

The basic argument in Jackendoff (2000) is that a shallow syntactic representation is not the right kind of input to the central systems; meaning *is* part of the linguistic input system, and interpreting utterances is fast and mandatory. Recall Fodor's example of "fast shadowers" above. Apparently, Fodor was aware of the speed of interpretation as were Jackendoff (1987) and Marslen-Wilson and Tyler (1987), but he thought that it was a form of belief fixation since to interpret a sentence involves determining its truth value. Jackendoff (2000) argues, "*in order to be fixed* (i.e., for its truth value to be determined), *a belief has to be formulated in terms of propositional structure*. A syntactic representation is simply the wrong vehicle for belief fixation" (p. 8, original emphasis). As another example, Jackendoff mentions that antecedent resolution of PRO relies on not just syntactic information but "is a complex function of the semantics of the main and subordinate verbs" (p. 10). For example in (3.1) (Jackendoff's (4)), the interpretation of PRO changes depending on the verb used.

(3.1) a. Bill asked Harry to be examined by the doctor.
(Bill asked Harry$_i$ [PRO$_i$ to be examined by the doctor])

b. * Bill asked Harry to be forced to leave.
(*Bill$_i$ asked Harry$_j$ [PRO$_{i/j}$ to be forced PRO$_{i/j}$ to leave])

In addition, the interpretation of aspect as in (3.2) (Jackendoff's (5)) is not based on syntax alone; it is part of the propositional content, and it is fast and mandatory. Notice that (3.2) means that the light flashed repeatedly, but *the light flashed* could mean that it only flashed once. Based on arguments like these, Jackendoff (2000) concludes that "the output of language perception is not a syntactic structure but an expression in the Language of Thought/conceptual structure/narrow content" (p. 11).

(3.2) a. The light flashed until dawn.

Jackendoff's idea of modularity, which he calls "representational modularity," is that information encapsulation and domain specificity of a mental process "has to do precisely with what representations it accesses and derives" (p. 13). He divides mental processes into three categories (pp. 12–13):

1. **Integrative processes:** These are processes that make a complete representation in a given form from a collection of fragmentary structures in that same form. An example of this is the syntactic parser that makes a syntactic tree from a list of lexical categories.

2. **Interface processes:** These are processes that convert one form of mental representation into another. As examples of this, one can mention the conversion of a syntactic parse into a specification of semantic roles, or the conversion of continuous speech signal into a discretely segmented phonetic representation.

3. **Inferential processes:** These are processes that take complete representations in some format and relate them to or construct new representations in the same format. Classic examples of this sort of process are inference rules, which derive new propositions from existing ones.

The language module in Jackendoff's model consists of integrative and interface sub-modules. Syntax, semantics, and phonology process different kinds of information, yet they are related in some aspects. The relations among these modules are made possible by the interface modules. The overall picture of the architecture of the language faculty as Jackendoff sees it is presented in Figure 3.1 on the facing page. Thinking in terms of spreading activation, Jackendoff points out that when one receives auditory linguistic input, the modules get activated from left to right and when one speaks, the activation spreads from right to left. It is not necessary for a level of representation to be completed by its integrative processor before the interface processors start passing information. "Any fragment of representation at one level is sufficient to call into action (or activate) any modules that can make use of this information" (p. 16). This model allows for "opportunistic" or "incremental" processing. The model also allows for modular feedback during processing. This architecture also fits nicely with constraint-based grammars.

> One can see the constraints specific to a particular level as principles applied by the integrative processor for that level; likewise, the linking constraints are the business of the interface processors. The constraints within one level are nondirectional, so that one can use them to build (or activate) structures from the bottom up or from the top down. The interface constraints are likewise nondirectional, so that one can, for instance, use them to build (or activate) partial syntactic structures based on phonological input—or equally vice versa. Moreover, the interface constraints can be used directly to generate feedback, so

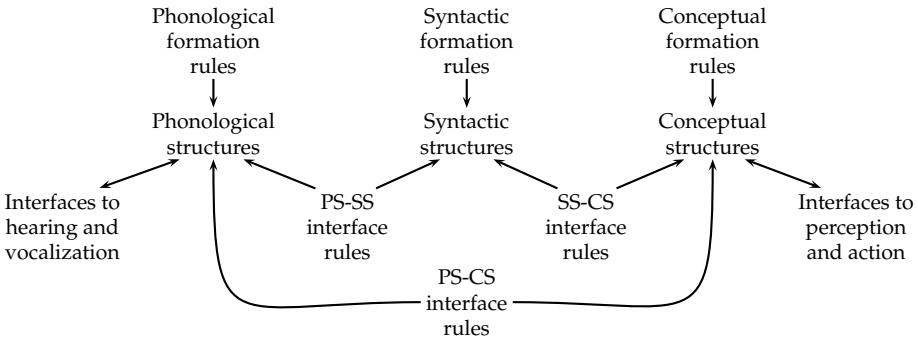


Figure 3.1: The tripartite parallel architecture of language faculty (Jackendoff, 2002, p. 125)

> that for instance, in perception, semantic-to-syntactic and syntactic-to-phonological constraints can together serve to constrain phonological analysis... (p. 24)

3.5 Discussion

The term *module*, as we have used it here, has two related senses. First, one can think of a linguistic module as a neuropsychological entity; that is, a part of the brain/mind with a distinct function. Second, a module can be thought of as an abstract theoretical entity that helps us make sense of our observations. The first view of module is a *realistic* view. The researcher that subscribes to this view tries to find out how the division of labour in the human mind actually works. The second view, on the other hand, is a *pragmatic* one. The adherents to this approach try to categorize the phenomena they observe into simple manageable pieces of knowledge that together make a coherent whole. These pieces of knowledge may or may not correspond to any module in the realistic sense. Psycho- and neurolinguists belong to the realist camp, while theoretical and computational linguists largely belong to the pragmatic camp.

A pragmatist sees a module as a coherent piece of knowledge with little or no internal conflicts. It seems that theoretical linguists have always agreed with this, at least implicitly. Whenever conflicting requirements have been observed somewhere, that piece of knowledge has been broken apart at that point and the resulting pieces have been called separate modules/components. The traditional division of linguistic knowledge into the lexicon and grammar, and within grammar, into syntax, semantics and phonology is a most obvious example. More recently, some researchers have suggested that syntax should also be broken down into two

distinct submodules exactly because of conflicts observed therein. Penn (1999a,b), and Penn and Haji-Abdolhosseini (2003), for instance, suggest that, based on certain word-order phenomena in German and Serbo-Croatian, syntax should be broken up into two components: one that accounts for immediate dominance relations, and one that accounts for word-order. Similar ideas have also been suggested by Kathol (1995, 2000).

Now given that modules process different kinds of information, they may not agree with one another at all times, and the connections among them are weaker than the connections inside modules. I argue that in a constraint-based framework one can model the intermodular connections with soft constraints and the intramodular connections with hard constraints. As mentioned in section 2.4, Keller (2000) divides grammatical constraints into hard and soft; the violation of hard constraints results in absolute ungrammaticality and the violation of soft constraints results in graded grammaticality. For example, in the case of extraction from *picture NPs*, he shows that sentences with verbs that presuppose the existence of their object (e.g., *tear up*) are more resistent to extraction than those that do not (see (3.3)). Extraction from indefinite NPs is also more acceptable than from definite ones (see (3.4)). In addition, referential NPs have been found to be more extractable than non-referential NPs (see (3.5)). However, extraction in sentences that violate subject-auxiliary inversion results in categorically unacceptable sentences (see (3.6)), and so does the use of resumptive pronouns in *wh*-extraction (see (3.7)).

(3.3) a. Which friend has Thomas painted a picture of?
b. ? Which friend has Thomas torn up a picture of?

(3.4) a. Which friend has Thomas painted a picture of?
b. ? Which friend has Thomas painted the picture of?

(3.5) a. Which friend has Thomas painted a picture of?
b. ? How many friends has Thomas painted a picture of?

(3.6) a. Which friend has Sarah painted a picture of?
b. * Which friend Sarah has painted a picture of?

(3.7) a. Which friend has Sarah painted a picture of?
b. * Which friend has Sarah painted a picture of her?

Keller also shows that soft constraints are subject to context effects; that is, the acceptability of sentences violating such constraints improves in the presence of a felicitous context. But this is not the case with hard constraints such as agreement, subject-auxiliary inversion, and resumptive pronoun licensing. Other studies on the interaction of prosodic, syntactic, and information-structural constraints

(see, among others, Büring, 2001; Keller and Alexopoulou, 2001; Choi, 2001) have shown that such constraints are violable (i.e., do not cause categorical acceptability judgements), and thus have been used to argue for Optimality Theoretic accounts of syntax. Work on the correspondence between syntactic and prosodic structures has also shown that intonation phrases are only loosely related to syntactic structures, and this relation can best be expressed in terms of preferences rather than crisp constraints (see Chapter 6 for more detail).

Keller's (2000) soft constraints all tie syntax with semantics. The notions of presupposition, referentiality, and definiteness are all semantic. Information structure is from discourse and its effect on word order and prosody is gradient. Heavy-NP shift, which is an interface constraint, is also a soft constraint, meaning that its violation does not result in categorical acceptability judgements. Hard constraints, on the other hand, are found inside modules. For instance, binding, subject-auxiliary inversion, and resumptive pronoun licensing are all purely syntactic; constraints on meter and syllable structure are purely phonological; and selectional restrictions are purely semantic. The idea that soft constraints seem to operate at interfaces has recently been suggested by Sorace and Keller (2005), as well, but the authors make no strong claims about modularity.

There is a certain kind of soft constraints that are not simply violable. They have degrees of violation. I will talk about degrees of violation in chapters 5 and 7. In the case of heavy-NP shift, for example (for other examples, see also Arnold et al., 2000; Wasow, 2002), we see that the heavier the NP, the stronger the tendency for it to come last (see (3.8)).

(3.8) a. John bought a computer yesterday.
b. John bought several pieces of hardware yesterday.
c. ? John bought several pieces of hardware that he'd been dreaming about yesterday.
d. ?? John bought several pieces of hardware that he'd been dreaming about for months yesterday.
e. *? John bought several pieces of very expensive hardware that he'd been dreaming about since he saw my fancy computer yesterday.

Through corpus analysis and experimental work, Arnold et al. (2000) show that heavy-NP shift and dative alternation are influenced by the length of the constituents involved as well as their information status. They show that the corpus results and the experimental results are compatible and discuss where the differences in the results may have originated. They also discuss the fact that when the difference between the lengths of the constituents is larger, heavy-NP shift and dative alternation are more likely to happen, a result compatible with what is pro-

posed in Chapter 5. They, however, do not attempt to formalize their findings in order to incorporate them within the linguistic theory. No claims about modularity and its relationship to soft constraints is made in that work, either.

Keller and Alexopoulou (2001) and Sorace and Keller (2005) discuss competition among modules and talk about soft versus hard constraints. These works advocate the use of the magnitude estimation method as a rigourous experimental method for eliciting acceptability judgements from language users. Keller and Alexopoulou (2001) investigate the interaction of phonological and syntactic constraints on the realization of information structure in Greek. They also extend the standard Optimality Theory in order to account for not only the optimal structure but the markedness of the suboptimal structure also. Sorace and Keller (2005) advocate linear optimality theory and in the conclusion of the paper suggest that soft constraints seem to be applying at interfaces, a claim that this thesis is set to pursue. They make no strong claims about modularity, nor do they discuss the notion of graded constraints.

3.5.1 Soft, Violable, and Graded Constraints

One point of clarification is in order here. I shall use "soft constraint" as a cover term for constraints that can be violated. "Violable constraints" are those whose violation is discrete and one can count the number of violation of these constraints by a structure. "Graded constraints" are those soft constraints whose violation cannot be counted. The violation of such constraints is a matter of degree. This type of constraint is not modelled in any major linguistic theory. I propose a *weighted soft-constraint satisfaction* approach (see Chapter 7) for modelling interface constraints (the parameters of this model can be set using experimental or corpus data), and a crisp intra-modular constraint system. The modules operate independently and in parallel communicating only through interface constraints as proposed by Jackendoff (2002).

This division of constraint systems allows for more coherent and informationally encapsulated modules, while still accounting for gradient effects of soft constraints as well as multiple violations and "ganging-up" effects as discussed by Keller (2000). This model also provides us with metatheoretical diagnostics regarding the nature of observed phenomena; that is, whenever graded grammaticality is observed, we would expect to find some form of inter-modular interaction, rather than posit an arbitrary constraint in a single module.

3.6 Summary

This chapter presented an overview of the different views on modularity in linguistics and cognitive science. We started with Simon's discussion of hierarchic complex systems, saying that firstly, hierarchic (modular) systems are more stable and thus more likely to survive, and secondly, we are more tuned to seeing and understanding such systems better. The second part of the previous statement is not a claim about our subject of study but a claim about ourselves; yet, its point is that a modular artificial system (as in a theory), even if not real, is more easily understood than a non-modular one. We then discussed Fodorian modularity and its properties; informational encapsulation and speed being the most important ones. Turning to the Jackendovian view of modularity, we went over representational modularity. According to this view, syntactic representation is the wrong vehicle for belief fixation. Jackendoff resolves several of the objections raised against the modularity view by positing a parallel architecture for the language faculty where each module processes its own data type and the whole system is constrained through interface modules/rules. In the last section, we argued that it is desirable to implement graded constraints at interfaces.

Chapter 4

A Generalized Theory of Soft Constraint Satisfaction

4.1 Introduction

Constraint programming has been a very exciting area of research in artificial intelligence in the past decade. The holy grail of constraint programming is to find ways of describing a problem in terms of constraints without having to worry about how those constraints are processed in finding a solution. This will allow one to concentrate on the problem as opposed to the details of algorithms and processing (for an excellent introduction, see Marriott and Stuckey, 1998). This constraint-based view of characterizing problems has also found its way into linguistics. HPSG, LFG, and OT are all constraint-based theories of language, and their claim is that by expressing linguistic generalizations in terms of constraints, we are better able to see the phenomena that are involved without getting entangled in procedural details.

Researchers in constraint programming have found that many real-life problems cannot be expressed in terms of *crisp* (non-violable) constraints because as the problem gets more complicated, we reach a point where it is either impossible to find a solution or it takes a very long time to do so. This has led researchers to seek ways of relaxing or weighting constraints so that the less important ones can be violated in favour of the more important ones. This is also a route that OT has taken (see section 2.5 on page 12).

In the following section, we present a generalized theory of soft constraint satisfaction introduced by Bistarelli (2001). This theory is based on a certain algebraic structure called the *semiring*. Based on a solid mathematical foundation, Bistarelli's theory of Semiring-based Constraint Satisfaction Problems (SCSP) illustrates that

several of the previous models of soft constraint satisfaction are instances of SCSP. The next section provides a formal introduction of constraint satisfaction problems in general, and section 4.3 introduces Bistarelli's semiring-based account. In Chapter 7, we will show that linear optimality theory discussed in Chapter 2 can be seen as an instance of SCSP and how it can be extended to account for the kind of graded constraints discussed in Chapter 5. We will also outline an SCSP-based extension to HPSG type antecedent constraints in section 7.4.

4.2 Constraint Satisfaction Problems

4.2.1 Formal Definition

This subsection is based on section 1.1 of Bistarelli (2001).

DEFINITION 4.2.1 ***(Constraint Satisfaction Problem)*** *A Constraint Satisfaction Problem is a sextuple* $\langle V, D, C, con, def, a\rangle$ *where*

- V *is a finite set of variables, i.e.,* $V = \{v_1, \ldots, v_n\}$*;*
- D *is a set of values, called* the domain*;*
- C *is a finite set of constraints, i.e.,* $C = \{c_1, \ldots, c_m\}$*.* C *is ranked, i.e.,* $C = \bigcup_k C_k$ *such that* $c \in C_k$ *if* c *involves* k *variables;*
- con *is called the* connection function *and it is such that*

$$con : \bigcup_k (C_k \rightarrow V^k),$$

  where $con(c) = \langle v_1, \ldots, v_k\rangle$ *is the tuple of variables involved in* $c \in C_k$*;*

- def *is called the* definition function *and it is such that*

$$def : \bigcup_k (C_k \rightarrow \wp(D^k)),$$

  where $\wp(D^k)$ *is the power set of* D^k*, that is, all the possible subsets of* k*-tuples in* D^k*;*

- $a \subseteq V$*, and represent the* distinguished *variables of the problem.*

con describes which variables are involved in which constraint; def specifies which are the domain tuples permitted by the constraint. The set a is used to point out the variables of interest in the given Constraint Satisfaction Problem (CSP),

i.e., the variables for which we want to know the possible assignments, compatible with all the constraints. This set is equal to V if all the variables are of interest. This does not have to be the case however. In fact, it is reasonable to think that the CSP representation of a problem contains many details (in terms of constraints and/or variables) which are needed for a correct specification of the problem but are not important as far as the solution of the problem is concerned.

The solution $Sol(P)$ of a CSP $P = \langle V, D, C, con, def, a\rangle$ is defined as the set of all instantiations of the variables in a which can be extended to instantiations of all the variables which are consistent with all the constraints in C.

DEFINITION 4.2.2 ***(Tuple Projection and CSP Solution)*** *Given a tuple of domain values $\langle v_1, \ldots, v_n\rangle$, consider a tuple of variables $\langle x_{i1}, \ldots, x_{im}\rangle$ such that for all $j = 1, \ldots, m$, there exists a $k_j \in \{1, \ldots, n\}$ such that $x_{ij} = x_{kj}$. Then the projection of $\langle v_1, \ldots, v_n\rangle$ over $\langle x_{i1}, \ldots, x_{im}\rangle$, written $\langle v_1, \ldots, v_n\rangle_{|\langle x_{i1}, \ldots, x_{im}\rangle}$, is the tuple of values $\langle v_{i1}, \ldots, v_{im}\rangle$. The solution $Sol(P)$ of a CSP $P = \langle V, D, C, con, def, a\rangle$ is defined as*

$$\left\{ \langle v_1, \ldots, v_n\rangle_{|a} \text{ such that } \left\{ \begin{array}{l} v_i \in D \text{ for all } i; \\ \text{for all } c \in C, \langle v_1, \ldots, v_n\rangle_{|con(c)} \in def(c). \end{array} \right\} \right.$$

The solution to a CSP is therefore an assignment of a value from its domain to every variable, in such a way that every constraint is satisfied. We may want to find just one solution, with no preferene as to which one, or all solutions.

To give a graphical representation of a CSP problem, we use a labelled hypergraph which is usually called a *constraint graph* (Dechter and Pearl, 1988).

DEFINITION 4.2.3 **Labelled Hypergraph** *Given a set of labels L, a hypergraph labelled over L is a quadruple $\langle N, H, c, l\rangle$, where N is a set of nodes, H is a set of hyperarcs, $c : \bigcup_k (H_k \to N^k)$, and $l : \bigcup_k (H_k \to \wp(L^k))$. That is, c gives the tuple of nodes connected by each hyperarc, and l gives the label of each hyperarc.*

DEFINITION 4.2.4 **From CSPs to Labelled Hypergraphs** *Consider a CSP $P = \langle V, D, C, con, def, a\rangle$. Then the labelled hypergraph corresponding to P, written $G(P)$, is defined as the hypergraph $G(P) = \langle V, C, con, def\rangle$ labelled over D.*

In the hypergraphs corresponding to a CSP, the nodes represent the variables of the problem, and the hyperarcs represent the constraints. In particular, each constraint c can be represented as a hyperarc connecting the nodes representing the variables in $con(c)$. Constraint definitions are instead represented as labels of hyperarcs. More precisely, the label of the hyperarc representing constraint c will be $def(c)$.

4.2.2 Example

Let us now illustrate the definitions presented in the previous subsection in the form of an example. A well-known constraint satisfaction problem is presented below:[1]

Consider the problem of choosing matching clothes (shirt, shoes and pants). This problem can easily be modelled using three finite domain variables with a number of binary constraints between them. In this case, $P = \langle V, D, C, con, def, a \rangle$ is defined as follows:

- $V = \{s, f, p\}$; s for shirt, f for footwear, and p for pants.
- $D = \{r, w, c, s, b, d, g\}$; r for red, w for white, c for cordovans, s for sneakers, b for blue, d for denim, and g for grey.
- $C = \{sp, fp, sf\}$; sp says what shirt goes with what pants; fp says what footwear goes with what pants; and sf says what shirt goes with what footwear. In other words:
  - $con(sp) = \langle s, p \rangle$
  - $con(fp) = \langle f, p \rangle$
  - $con(sf) = \langle s, f \rangle$
- The def function is defined as follows:
  - $def(sp) = \{\langle r, g \rangle, \langle w, b \rangle, \langle w, d \rangle\}$
  - $def(fp) = \{\langle s, d \rangle, \langle c, g \rangle\}$
  - $def(sf) = \{\langle w, c \rangle\}$

The above CSP can be explained in words as follows. We have a red and a white shirt, a pair of cordovans, a pair of sneakers, a pair of blue pants, a pair of denim pants and a pair of grey pants. The red shirt goes with the grey pants. The white shirt goes with the blue pants; it also goes with the denim pants. The sneakers go with the denim pants while the cordovans go with the grey pants. Finally, the white shirt goes with the cordovans. This CSP can be represented graphically as in Figure 4.1 on the facing page.

In Figure 4.1, the nodes represent the variables, V. The values over the nodes represent the members of the domain, D, i.e., the values that each variable can

[1] Adapted from Roman Barták's on-line guide to constraint programming at `http://kti.ms.mff.cuni.cz/~bartak/constraints/`.

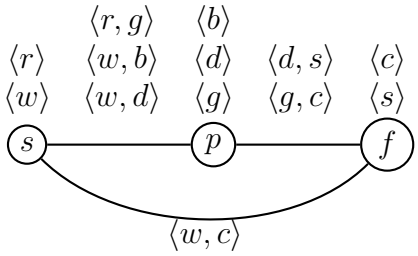


Figure 4.1: The Dressing Problem

take, and the tuples over the arcs connecting the nodes represent the constraints that apply to pairs of variables in this constraint system, C. Of course, it is easy to see that this is an over-constrained system with no solutions. There is only one option for matching shirts and shoes, which states that a white shirt should be worn with cordovan shoes, while cordovan shoes only go with grey pants, which in turn only go with the red shirt.

4.3 A Semiring-Based Theory of Constraint Satisfaction Problems

Clearly the constraint system depicted in Figure 4.1 does not have a solution, which makes it an instance of an over-constrained CSP. To solve the problem of over-constrained CSPs, researchers have proposed several alternative approaches which enable one to relax some constraints in order to find a solution to the problem. As discussed earlier, Bistarelli (2001) shows that some of these approaches (e.g., probabilistic, fuzzy, and weighted CSPs) can be thought of as special instances of a more general soft-constraint satisfaction framework, which he calls the Semiring-based Constraint Satisfaction Problems (SCSP). The present and the following sections, which are based on Chapter 2 of Bistarelli (2001), briefly introduce this theory.

Bistarelli's main idea is that

> ...a semiring (that is, a domain plus two operations satisfying certain properties) is all that is needed to describe many constraint satisfaction schemes. In fact, the domain of the semiring provides the levels of consistency (which can be interpreted as cost, or degree of preference, or probabilities, or others), and the two operations define a way to combine constraints together. More precisely, we define the notion of constraint solving over any semiring. Specific choices of the semiring

will then give rise to different instances of the framework, which may correspond to known or new constraint solving schemes.

DEFINITION 4.3.1 **Semiring** *A semiring is a quintuple* $\langle A, sum, \times, 0, 1\rangle$ *such that*

- A *is a set and* $0, 1 \in A$*;*
- sum*, called the* additive operator*, is a commutative (i.e.,* $sum(a, b) = sum(b, a)$*) and associative (i.e.,* $sum(a, sum(b, c)) = sum(sum(a, b), c)$*) operation with* 0 *as its unit element (i.e.,* $sum(a, 0) = a = sum(0, a)$*);*
- $\times$*, called the* multiplicative operator*, is an associative operation such that* 1 *is its* unit element *and* 0 *is its* absorbing element *(i.e.,* $a \times 0 = 0 = 0 \times a$*);*
- $\times$*, distributes over* sum *(i.e., for any* $a, b, c \in A, a \times sum(b, c) = sum((a \times b), (a \times c))$*).*

The reader may have noted that the set of real numbers between 0 and 1 (inclusive) together with arithmetic $+$ and $\times$ form a semiring, for example.

Bistarelli introduces semirings with additional properties for the two operations. He calls this algebra a *c-semiring* (c for "constraint"), and defines it as follows:

DEFINITION 4.3.2 **C-Semiring** *A c-semiring is a quintuple* $\langle A, +, \times, 0, 1\rangle$ *such that*

- A *is a set and* $0, 1 \in A$*;*
- $+$ *is defined over (possibly infinite) sets of elements of* A *as follows:*[2]
  - *for all* $a \in A, \sum(\{a\}) = a$*;*
  - $\sum(\emptyset) = 0$ *and* $\sum(A) = 1$*;*
  - $\sum(\bigcup A_i, i \in I) = \sum(\{\sum(A_i), i \in I\})$ *for all sets of indices* I *(flattening property);*
- $\times$ *is a binary associative and commutative operation such that* 1 *is its unit element and* 0 *is its absorbing element;*
- $\times$ *disributes over* $+$ *(i.e., for any* $a \in A$ *and* $b \subseteq A, a \times \sum(B) = \sum(\{a \times b, b \in B\})$*).*

[2]We use $+$ in infix notation for a two-element set, and the symbol $\sum$ in prefix notation for more elements.

The fact that $+$ is defined over *sets* of elements, and not *pairs* or *tuples*, automatically makes such an operation commutative, associative, and idempotent. It is also possible to show that 0 is the unit element of $+$. By using the flattening property, we get $\sum(\{a, 0\}) = \sum(\{a\}\emptyset) = \sum(\{a\}) = a$. This means that a c-semiring is a semiring (where the sum operation is $+$) with some additional properties. It is also possible to prove that 1 is the absorbing element of $+$. By flattening and by the fact that we set $\sum(A) = 1$, we get $\sum(\{a, 1\}) = \sum(\{a\} \cup A) = \sum(A) = 1$.

According to Bistarelli, the advantage of using c-semirings instead of semirings are as follows: The idempotency of the $+$ operation is needed in order to define a partial ordering $\leq_s$ over the set A, which will enable us to compare different elements of the semiring. Such a partial order is defined as: $a \leq_s b$ iff $a + b = b$. Intuitively, $a \leq_s b$ means that b is "better" than a, or, from another point of view, that between a and b, the $+$ operation chooses b. This ordering is used to choose the "best" solution in constraint problems.

Given any c-semiring $S = \langle A, +, \times, 0, 1\rangle$, consider the relation $\leq_s$ over A such that $a \leq_s b$ iff $a + b = b$. Then Bistarelli proves that $\leq_s$ is a partial order. He also proves that $+$ and $\times$ are monotones over $\leq_s$. That is, given any c-semiring $S = \langle A, +, \times, 0, 1\rangle$, consider the relation $\leq_s$ over A. Then $+$ and $\times$ are monotones over $\leq_s$ means that $a \leq_s a'$ implies $a + b \leq_s a' + b$ and $a \times b \leq_s a' \times b$.

Since 1 is also the absorbing element of the additive operation, then $a \leq_s 1$ for all a. Thus 1 is the maximum element of the partial ordering. This implies that the $\times$ operation is *intensive*, that is, $a \times b \leq_s a$. This is important since it means that combining more constraints leads to a "worse" result in terms of the $\leq_s$ ordering.

Sometimes we need the $\times$ operation to be closed on a certain finite subset of the c-semiring.

DEFINITION 4.3.3 **AD-closed** *Given any c-semiring $S = \langle A, +, \times, 0, 1\rangle$, consider a finite set $AD \subseteq A$. Then $\times$ is AD-closed if for any $a, b \in AD, (a \times b) \in AD$.*

It is shown that c-semirings can be assimilated to complete lattices. We also sometimes need to consider c-semirings where $\times$ is idempotent, which makes the c-semiring equivalent to distributive lattices.[3]

DEFINITION 4.3.4 **LUB, GLB, (Complete Lattice)** *Consider a partially ordered set S and any subset I of S. Then we define the following:*

- *an* upper bound *(resp.* lower bound*) of I is any element x such that for all $y \in I, y \leq x$ (resp., $x \leq y$);*

[3]For an introduction to lattices and ordered sets, see Davey and Priestley (1990).

- *the* least upper bound (LUB) *(resp.* greatest lower bound (GLB) *of I is an upper bound (resp. lower bound) x of I such that for any other upper bound (resp. lower bound) x' of I, we have that $x \leq x'$ (resp., $x' \leq x$).*

A lattice *is a partially ordered set where every subset of two elements has a LUB and a GLB. A* complete lattice *is a partially ordered set where every subset has a LUB and GLB.*

Bistarelli proves that $\langle A, \leq_s \rangle$ is a complete lattice, which entails $\sum(I) = LUB(I)$ for any set $I \subseteq A$. Thus every subset I of A has a least upper bound (which coincides with $\sum(I)$). This means that $\langle A, \leq_s \rangle$ is a LUB-complete partial order. Note that the $+$ operator coincides with the LUB of the lattice $\langle A, \leq_s \rangle$.

Bistarelli also proves that given a c-semiring $S = \langle A, +, \times, 0, 1 \rangle$ and a corresponding complete lattice $\langle A, \leq_s \rangle$, $\times$ is also idempotent. Furthermore, in the particular case in which $\times$ is idempotent and $\leq_s$ is total, we have that $a+b = max(a, b)$ and $a \times b = min(a, b)$.

4.4 Constraint Systems and Problems

The notions of constraint system, constraint, and constraint problem in this theory are parametric with respect to the notion of c-semiring discussed in the previous section. Intuitively, a constraint system specifies the c-semiring $\langle A, +, \times, 0, 1 \rangle$ to be used along with the set of all variables and their domain D.

DEFINITION 4.4.1 **Constraint System** *A constraint system is defined as a triple $CS = \langle S, D, V \rangle$, where S is a c-semiring, D is a finite set, and V is an ordered set of variables.*

A constraint over a given constraint system specifies the involved variables and the "allowed" values for them. More precisely, for each tuple of values (of D) for the involved variables, a corresponding element of A is given. This element can be interpreted as the tuple's weight, or cost, or level of confidence, etc.

DEFINITION 4.4.2 **Constraint** *Given a constraint system $CS = \langle S, D, V \rangle$, where $S = \langle A, +, \times, 0, 1 \rangle$, a constraint over CS is a pair $\langle def, con \rangle$, where*

- *$con \subseteq V$, it is called the* type *of the constraint;*
- *$def : D^k \rightarrow A$ (where k is the cardinality of con) is called the* value *of the constraint.*

A constraint problem is then just a set of constraints over a given constraint system, plus a selected set of variables (thus a *type*). These are the variables of interest in the problem, i.e., the variables of which we want to know the possible assignments' compatibly with all the constraints.

4.4.1 Instances of the SCSP Framework

Having laid out the c-semiring based theory of constraint satisfaction, Bistarelli shows that some of the previous constraint satisfaction approaches can be seen as instances of this theory differing only in the choice of the semiring. Below I list the different CSPs and the semirings used in them as discussed by Bistarelli.

- **Classical CSPs:** A classical CSP is just a set of variables and constraints, where each constraint specifies the tuples that are allowed for the involved variables (see Figure 4.1, for example). Since the constraints in a CSP are crisp, they can be modelled with a semiring containing only 0 and 1 in A. Also we can model constraint combination with logical *and*, and the projection over some of the variables (to obtain the value of the tuples of the variables in the type of the problem), with logical *or*. Thus, a CSP can be seen as just an SCSP with the following semiring:

$$\mathrm{S}_{\mathrm{CSP}} = \langle \{0, 1\}, \vee, \wedge, 0, 1 \rangle$$

- **Fuzzy CSPs:** Fuzzy CSPs allow for non-crisp constraints, which associate a preference level with each tuple of values. This level of preference is always between 0 and 1. The solution to a fuzzy CSP is defined as the set of tuples of values for all the variables which have the maximal value. Fuzzy CSPs can be modelled in the SCSP framework by choosing the following semiring:

$$\mathrm{S}_{\mathrm{FCSP}} = \langle \{x | x \in [0, 1]\}, max, min, 0, 1 \rangle$$

- **Probabilistic CSPs:** In probabilistic CSPs, each constraint c has an associated probability $p(c)$. Saying that c has probability p, means that the situation corresponding to c has probability p of occurring in the real-life problem. The semiring corresponding to the probabilistic CSPs is as follows:

$$\mathrm{S}_{\mathrm{prob}} = \langle \{x | x \in [0, 1]\}, max, \times, 0, 1 \rangle$$

- **Weighted CSPs:** Contrary to fuzzy CSPs whose constraints come with preferences, in weighted CSPs, constraints have associated costs. The solution to a problem in such models is the one with minimum cost (e.g., time, space, number of resources, etc.). Therefore, the associated semiring for a weighted CSP is the following:

$$\mathrm{S}_{\mathrm{WCSP}} = \langle \mathbb{R}^*, min, +, +\infty, 0 \rangle$$

- **Set-Based CSPs:** The SCSP framework gives rise to an interesting class of its instances that are based on set operations such as union and intersection. The corresponding semiring for this class of CSPs is this:

$$\mathrm{S}_{\mathrm{set}} = \langle \wp(A), \bigcup, \bigcap, \emptyset, A \rangle$$

4.5 Summary

This chapter presented a brief overview of Bistarelli's c-semiring based generalized theory of soft constraint satisfaction systems. As Bistarelli shows, many previous CLP approaches to soft constraints are in fact instances of this generalized framework, which is parametric with respect to the semiring used. In Chapter 7 we will show that an instance of this theory, the *weighted soft constraint satisfaction* approach is suitable for modelling linguistic constraints.

Chapter 5

A Case Study of Graded Constraints

5.1 Introduction

As the title suggests, this chapter presents a case study of some graded constraints at work. The data for this study come from the Carlson et al. (2002) corpus (see section 5.2.1 below). The aim of the study was to investigate the interactions (or lack thereof) among three major linguistic modules: prosody, syntax, and discourse structure. More precisely, we would like to find out whether and how the prosodic weight of a discourse unit influences sentential discourse structure and what happens if such an influence results in a more syntactically marked sentence. We will show that prosodic weight influences the order of the discourse units when certain discourse relations are involved and that this influence is not constant and depends on the difference in the weights of the discourse units. Such a constraint cannot be modelled by a constraint system that simply counts the number of violation of constraints. Section 5.2 presents the data for this study, and discusses how syntactic markedness and prosodic weight were measured. Section 5.3 provides a detailed explanation of the studies conducted on the data, as well as an interpretation of the results. Section 5.4 shows how these results can be modelled statistically. An evaluation of the model is also presented in the same section.

5.2 Data

5.2.1 Corpus

The Carlson et al. (2002) corpus consists of 385 Wall Street Journal Articles from the Penn Treebank (Marcus et al., 1993). These articles have been annotated with dis-

course structure in the RST framework (for a brief introduction to RST, see Appendix A). In addition, the corpus includes human-generated extracts and abstracts associated with the original documents. Since I was interested in the internal functional structure of the sentences as opposed to the structure of the texts, I used Soricut and Marcu's (2003) sentence-level discourse parser (SPADE). The program uses syntactic and lexical information in a probabilistic model in order to produce sentence-level discourse structures for input sentences at near-human levels of performance. Figure 5.1 shows an example of SPADE output. The top of the discourse tree, the *root*, is at depth 0 of the tree. The ELABORATION relation in Figure 5.1 occurs at depth 1 of the tree and the ATTRIBUTION relation occurs at depth 2.

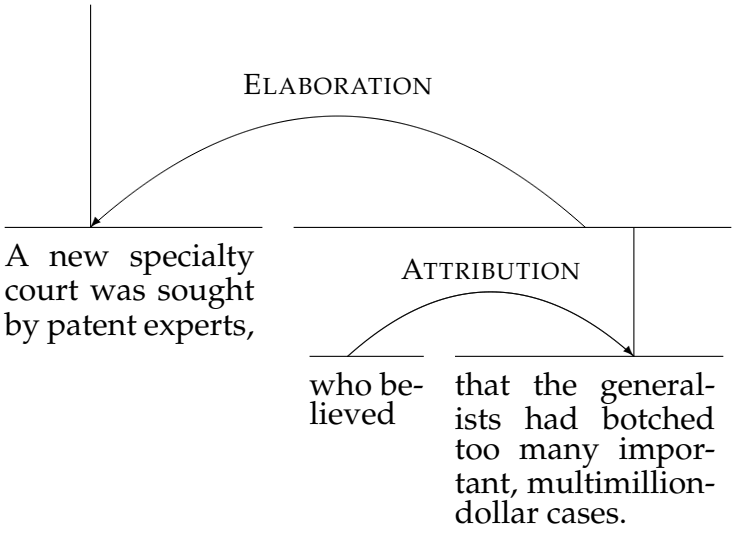


Figure 5.1: An example of SPADE output

I ran SPADE over the 178 articles in the training set of the corpus. Then, using Rohde's (2002) TGrep2 program (a program used for searching for patterns in trees), I extracted sentences that had Nucleus-Satellite (NS) or Satellite-Nucleus (SN) sequences at depth 1 of their discourse trees—immediately dominated by the ROOT. Finally, I randomly selected 1,500 sentences as my sample data. This included 843 sentences with NS order, and 657 sentences with SN order. Some examples sentences are given in (5.1) and (5.2) below.

(5.1) a. [USI Far East will hold a 60% stake in Luzon Petrochemical,] [according to papers signed with the Philippine government's Board of Investments.] (Attribution)

b. [The Court of Appeals for the Federal Circuit was created in 1982] [to serve, among other things, as the court of last resort for most patent

disputes.] (Enablement)

(5.2) a. [Dr. Lourie says] [the Justice Department interviewed him last July.] (Attribution)

b. [To answer the brokerage question,] [Kidder, in typical fashion, completed a task-force study.] (Enablement)

These relations are in fact coherence relations. A nucleus is said to be essentially more important to the author/speaker's point than a satellite, which is provided to support the nucleus or facilitate its understanding or perform other functions depending on the nature of the relation.

5.2.2 Prosodic Weight

Many definitions of weight exist in the literature from a simple length of the string in words to elaborate structure-based accounts. Wasow (1997) compares several of the proposed definitions through corpus analysis and concludes that they are all good predictors of weight.[1]

Since the data came in orthographic form, I had to measure the prosodic weight of each discourse unit relying on written language only. There were three ways to do this: (1) simply count the number of words and count that as a measure of prosodic weight; (2) somehow syllabify the written form, and count the number of syllables; or (3) feed the written form to a pronunciation dictionary, get a phonetic form, and then count the number of syllables in the phonetic transcription. The first approach is a poor approximation, although in a pilot study I conducted, the results were still promising. The third approach involves a lot of complications and it is not clear how much better results one would get compared to the second approach. This chapter presents results based on the second approach, i.e., using syllabified text to arrive at the length of discourse units (DUs) in syllables.

To do this I used TEX's hyphenation algorithm. Note that hyphenation is largely based on syllabification and TEX does this with very high accuracy. A major difference is that TEX does not insert a hyphen after the initial vowel in a /CVC(C)…/ pattern, as in *alone*. I solved the problem by adding this feature to the algorithm. The only problem that remained was abbreviations like *GE* or *OPEC*, as well as numbers like 1984 and symbols like $ (which are ignored as punctuation marks). Abbreviations and acronyms have very unpredictable pronunciations, for example, *GE* is pronounced letter by letter (/ʤi.ʔi/) while *OPEC* is pronounced as a word (/o.pɛk/). It is reasonable to assume that the distribution of these items is independent of the discourse and syntactic structures of the sentence and treating

[1] For an structural approach to complexity, see Hawkins (1994).

them as monosyllabic words will not affect the statistical results. This will only mean that the numbers we get as a measure of DU lengths in syllables will be a close approximation.

5.2.3 Markedness

Bard et al. (1996) and Keller (2000) use the magnitude estimation (ME) method to measure speakers' acceptability judgements of linguistic input (for other methods for assessing judgements, see Cowart, 1997). According to Bard et al. (1996), *acceptability* is a way of getting at *grammaticality*. The former is speakers' judgements of how good a sentence is, and the latter is a theoretical notion. Keller (2003), based on Bard, Frenck-Mestre, Kelly, Killborn, and Sorace's (1999) study, claims that ME data also correlate with data from self-paced reading and eye-tracking experiments. Therefore, he claims that the less acceptable a sentence is the more difficult it is for a human judge to process it.[2] Keller (2003) also shows that a lexicalized probabilistic context-free grammar (PCFG) to some extent reflects speakers' acceptability judgements.

As stated above, one question in the present study was whether the syntactic markedness of a sentence had any influence on the order of its discourse units (at depth 1). In other words, is it the case that the more/less marked a sentence is syntactically, the more/less marked its discourse structure is?

To measure the markedness of sentences, I extracted a lexicalized PCFG from the training set of the corpus (the same set used for discourse parsing). A PCFG consists of a set of context free rules in the form of $LHS \rightarrow RHS$ each annotated with a probability $P(RHS|LHS)$. This probability represents the likelihood of LHS being expanded to RHS. To ensure mathematical soundness, the probability measures of all the rules with the same LHS must add up to one (see below for a more formal definition). Figure 5.2 on the next page shows a simple (non-lexicalized) PCFG. The probability of a syntactic tree is the product of the probabilities of its subtrees (see Figure 5.3 on the facing page, as an example).

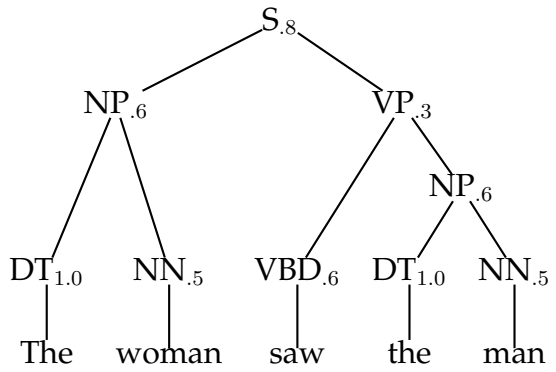
However, it is now widely accepted that bare phrase structure grammars (even probabilistic ones, as the one exemplified in Figure 5.2 on the next page) are too impoverished as models of human grammar. A simple approach to improve the situation is to incorporate some lexical information in the rules. This is a common approach in computational linguistics (for example, see Carroll and Rooth, 1998; Keller, 2003).

[2]Note that although less grammatical sentences take more time for people to process, it does not necessarily follow that any sentence that takes long to process is grammatically ill-formed.

S	→	NP VP	.8	DT	→	the	1.0
S	→	NP Aux VP	.2	NN	→	woman	.5
NP	→	DT NN	.6	NN	→	man	.5
NP	→	PN	.4	PN	→	John	.4
VP	→	VBD NP	.3	PN	→	Mary	.6
VP	→	VBZ NP	.7	VBD	→	saw	.6
				VBZ	→	sees	.4

Aux: auxiliary verb, DT: determiner, NN: common noun, NP: noun phrase, S: sentence, VBD: past verb, VBZ: 3rd person singular verb, VP: verb phrase

Figure 5.2: A fragment of a non-lexicalized PCFG



$$p(t) = .8 \times .6 \times .3 \times .6 \times 1.0 \times .5 \times .6 \times 1.0 \times .5 = .01296$$

Figure 5.3: Example of a tree generated by the (non-lexicalized) PCFG in Figure 5.2

Formally, a lexicalized context-free grammar (CFG) is a septuple $\langle N, P, T, R, H, \mathcal{H}, s \rangle$, where:[3]

- N is a set of non-terminals (phrasal categories, e.g., S, NP, VP);
- P is a set of pre-terminals (parts of speech, e.g., DT, NN, VB);
- T is a set of terminals (lexical items, e.g., *the*, *man*, *saw*);
- R is a set of rules in the form:
  - $N^i \rightarrow \zeta^j$ where ζ^j is a sequence of items from $N \cup P$, or
  - $P^k \rightarrow T^l$;
- $s \in N$ is the designated start symbol;
- $H \subseteq N \cup P$ is a set of categories whose lexical heads project to higher categories;
- $\mathcal{H}$ is a partial function from $N \cup P$ to $N \cup P \cup T$, such that in a rule $m \rightarrow \alpha n \beta$, where $m \in N \cup P$, α and β are sequences from $N \cup P \cup T \cup \{\epsilon\}$ (ϵ being the empty string), we have:
  - $\mathcal{H}(m) = n$ iff $n \in P \cap H$,
  - $\mathcal{H}(m) = \mathcal{H}(n)$ iff $m \in N \cap H$.

This means that the head of a pre-terminal belonging to H is the item on the right-hand-side of the rule (i.e., the terminal); the head of a non-terminal is the head of a designated daughter on the right-hand-side of the rule; and a category not belonging to H does not have a defined head.

A lexicalized PCFG G is, therefore, a lexicalized CFG where each rule in R has a corresponding probability such that for all i:

$$\sum_j p(N^i \rightarrow \zeta^j) = 1$$

For the present study H consisted of only nominal and verbal categories. These were NN, NNS, NNP, NNPS, NP, FW, VB, VBD, VBG, VBP, VBZ, VBN, VP, S, SBAR, S1, SBARQ, SINV, and SQ. The portion of the corpus used for grammar extraction contained 3,378 sentences, which resulted in a lexicalized PCFG with 37,145 rules and 12,725 lexical entries.

[3]For more information on PCFGs, see Carroll and Rooth (1998), Charniak (1993), and Manning and Schütze (2000).

S[saw]	→	NP[man] VP[saw]	.2	DT	→	the	1.0
S[saw]	→	NP[woman] VP[saw]	.3	NN	→	man	.5
S[sees]	→	NP[man] VP[sees]	.3	NN	→	woman	.5
S[sees]	→	NP[woman] VP[sees]	.2	VBD	→	saw	1.0
NP[man]	→	DT NN[man]	.4	VBZ	→	sees	1.0
NP[woman]	→	DT NN[woman]	.6				
VP[saw]	→	VBD[saw] NP[man]	.7				
VP[saw]	→	VBD[saw] NP[woman]	.3				
VP[sees]	→	VBZ[sees] NP[man]	.4				
VP[sees]	→	VBZ[sees] NP[woman]	.6				

Figure 5.4: A fragment of a lexicalized PCFG

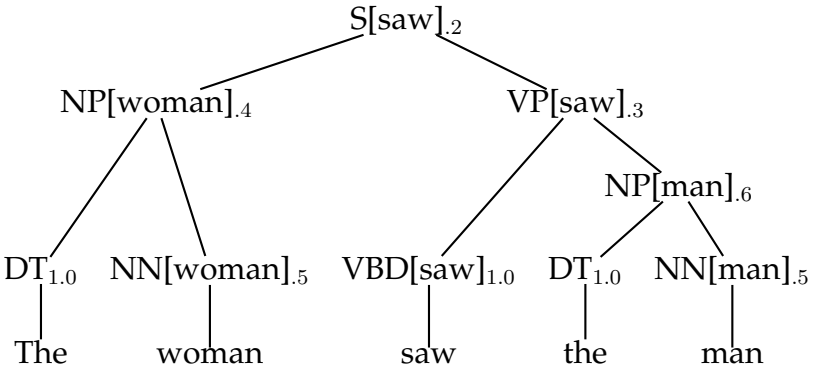
These lexicalized rules can be seen as approximating some of the linguistic information that is not explicit in the corpus or a simple treebank. The presence of lexical items in rules approximates such information as collocations, selectional restrictions, morphological data and linguistic functions of constituents.

If the probability of a sentence, calculated based on a lexicalized PCFG, reflects speakers' acceptability judgements, then we can use it as a measure of markedness. However, since the longer the sentence is, the lower its probability is (because more rules are used), we need to normalize the probabilities over the length of the sentences.[4] We shall call this measure of markedness (for the lack of a better term) *Normalized Markedness* or *N-markedness* for short. The N-markedness of a tree t is, therefore, calculated as follows:

$$\text{N-markedness}(t) = \frac{-\ln(p(t))}{length(t)}$$

Hence, more marked sentences get higher N-markedness scores than less N-marked ones. Figures 5.4 and 5.5 are provided as an example of a lexicalized PCFG and a tree generated by that grammar.

[4]Note that we measure length in syllables.



$$p(t) = .2 \times .4 \times .3 \times .6 \times 1.0 \times .5 \times 1.0 \times 1.0 \times .5 = .0036$$
$$\text{N-markedness}(t) = \frac{-\ln(.0036)}{length(t)} \approx .93$$

Figure 5.5: Example of a tree generated by the lexicalized PCFG in Figure 5.4 on the previous page

5.3 Data Analysis

5.3.1 Overall Syntactic Markedness and DU Order

5.3.1.1 Question

As mentioned in section 5.2.3, the goal of this experiment was to ascertain whether syntactic markedness (as measured by the negative logarithm of the probability of the sentence's parse tree) had any influence on the order of discourse units in sentences with NS and SN patterns at depth 1 of their discourse parse. The null hypothesis, then, would be that there is no relationship between syntactic markedness and DU order.

5.3.1.2 Analysis

Figure 5.6 on the facing page shows that there is a significant difference between the markedness measures of the sentences in the NS and SN groups in the two groups as a whole. The breakdown of the sentences based on the rhetorical relations shows that this significant difference arises only from the ATTRIBUTION group, i.e., when the satellite comes in the ATTRIBUTION relation with the nucleus (see Figure 5.7 on page 48). This result is also verified by an independent samples t-test ($t = 2.692$, $\mathrm{df} = 180.986$, $p < .05$). Figure 5.7 shows that in the ATTRIBUTION group, more syntactically N-marked sentences tend to come in the NS order

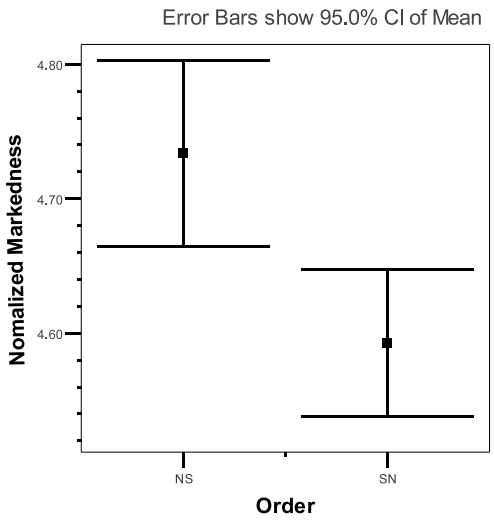


Figure 5.6: Comparison of the N-markedness measures of the sentences in the NS and SN groups

(see also Table 5.1 on page 49). A Pearson Correlation test between the percentage of the ATTRIBUTION sentences coming in the NS or SN order and their markedness measures revealed that this correlation is significant albeit not very strong ($r = .125$, $p < .01$). Therefore, we reject the null hypothesis. At least when it comes to ATTRIBUTION, there is a relation between syntactic N-markedness and DU order; however, in section 5.4.1 we see that this measure of markedness is too gross and simply taking into account the length of the sentence leads to a better prediction.

We are now going to focus on the relationship between syntactic markedness and DU order in the ATTRIBUTION group. Table 5.1 on page 49 shows that 81.0% of sentences with low N-markedness measures come in the SN order. But as the markedness of the sentences increases the likelihood of their taking the SN order decreases to 72.4%. The question here is why the NS order becomes more probable as the the N-markedness measure increases. The answer lies in the interaction among syntactic, discourse and prosodic structures.

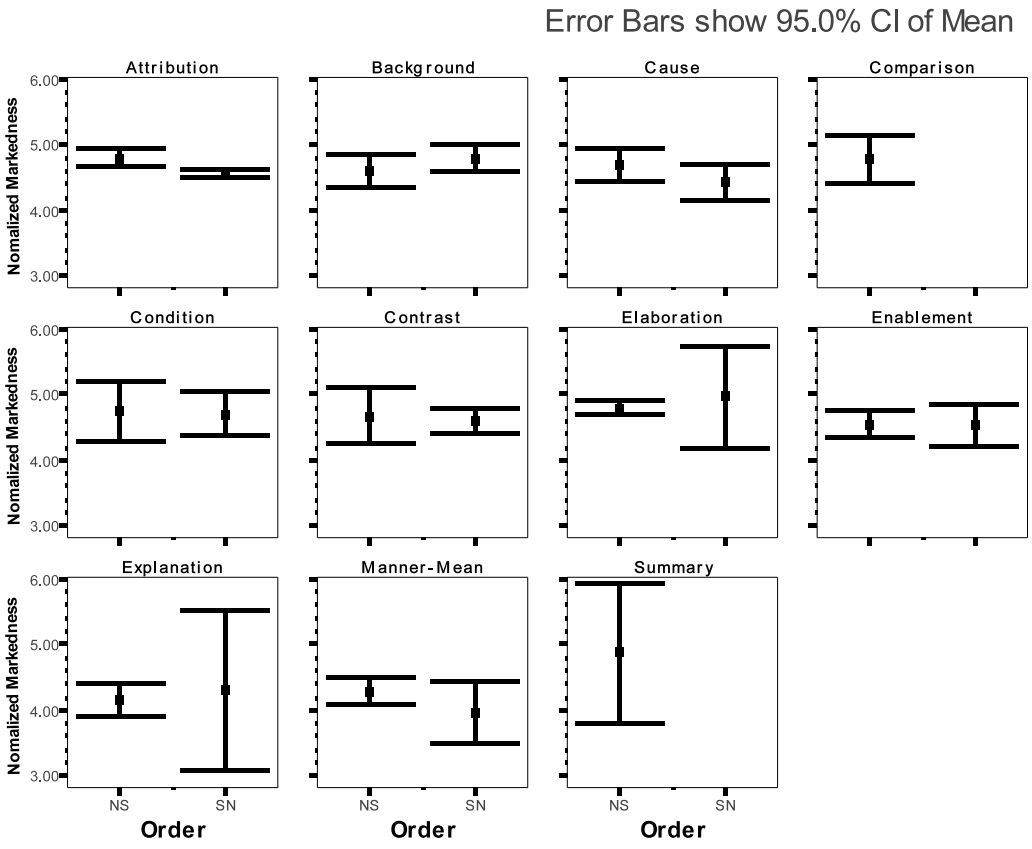


Figure 5.7: Comparison of the N-markedness measures of the sentences in the NS and SN groups divided by relation

N-Markedness (ν)		Order		Overall
		NS	SN	
Low	Percent	19.0%	81.0%	20.9%
$4 \leq \nu < 6$	Count	24	102	126
Medium	Percent	21.6%	78.4%	74.3%
$6 \leq \nu < 8$	Count	97	352	449
High	Percent	27.6%	72.4%	4.8%
$8 \leq \nu < 10$	Count	8	21	29
Overall	Percent	21.4%	78.6%	100.0%
	Count	129	475	604

Note: Pearson Correlation $= .125$, $p < .01$

Table 5.1: Comparison of the distribution of N-markedness measures of the sentences in the NS and SN groups (Rel=ATTRIBUTION)

5.3.1.3 Remarks

Mann and Thompson (1988b, p. 256), based on corpus data, provide a table of canonical DU orders for various relations (see section 5.3.3 and Table 5.7 on page 58). ATTRIBUTION, however, does not appear in that list. Figures 5.8 and 5.9 are examples of NS and SN sentences where the satellite forms the ATTRIBUTION relation with the nucleus. Note that in the ATTRIBUTION relation, the satellite corresponds to the part of the sentence that contains the subject and the verb while the nucleus corresponds to the object of the sentence. We know that the canonical word order for English is SVO, which means syntax prefers the example given in Figure 5.9. In addition, in section 5.3.3, I argue that SN is the canonical DU order for ATTRIBUTION, which means that discourse would also prefer SVO sentences since the subject and the verb are part of the satellite in ATTRIBUTION sentences (e.g., see the example presented in figures 5.8 and 5.9). This, of course, could be the result of the rigid word order requirements of English. Now the reason that more N-marked sentences are more likely to come in the NS order than less N-marked ones is the grammar's utilitarian approach to conflict resolution. The terms *utilitarian* and *egalitarian* are used by Moulin (1988) to refer to two different approaches in decision making. The terms have also been used by Bistarelli (2001) to refer to two kinds of constraint satisfaction. In utilitarian constraint satisfaction one would like to minimize the cost of the system as a whole even if it means incurring high costs on certain individual constraints. In the egalitarian approach, however, one would like to minimize the cost of individual constraints. Weighted soft constraint satisfaction systems (see Chapter 7) adopt the utilitarian approach, while fuzzy

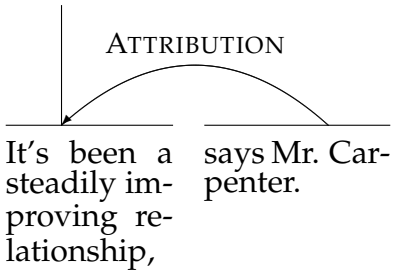


Figure 5.8: An example of an NS sentence in the ATTRIBUTION subgroup

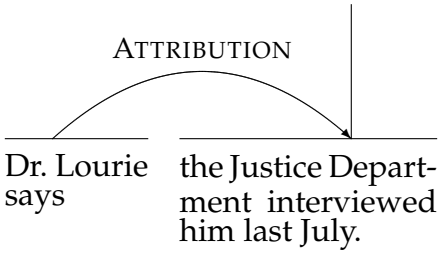


Figure 5.9: An example of an SN sentence in the ATTRIBUTION subgroup

soft constraint satisfaction systems adopt the egalitarian approach. Therefore, the grammar wants to minimize the total cost of the sentences that it generates. When a sentence with a marked and complicated structure comes along the grammar wants to avoid adding to its markedness by giving it the OSV or OVS order. At the same time, prosody prefers to have prosodically heavy units to appear last in the sentence, which means that some constraints may have to be violated in favour of better overall acceptability. This sort of conflict resolution is usually modelled using ranked violable constraints as in the OT approach. Keller (2000) shows that multiple violation of a constraint in a sentence makes it more marked than when the constraint is violated only once. He also shows that when several low-ranking constraints are violated they can gang up against a high-ranking constraint. In the present study we are looking at a different kind of constraint. Here markedness and prosodic weight are continuous measures. We cannot count how many times something is violated; rather, we are looking at the degree of markedness or the prosodic weight of a DU. In the next section we examine the role prosodic weight of DUs plays in this picture.

	N	Min	Max	Mean	SD
NS Order	843	−59	85	1.34	19.03
SN Order	657	−53	85	15.39	17.45
Overall	1,500	−59	85	7.50	19.63

Table 5.2: Summary statistics for δ values

5.3.2 Prosodic Weight and DU Order

5.3.2.1 Question

As in the previous study (section 5.3.1), the question in this study is whether the prosodic weight of a discourse unit has any influence on its placement. The sample used in this study is the same as the one used in section 5.3.1, which means we are looking at the order of DUs in NS and SN sentences. As mentioned in section 5.2.2, prosodic weight was calculated based on the lengths of DUs in syllables (e.g., see (5.3)). The null hypothesis in this study is that there is no relationship between prosodic weight and DU placement.

(5.3) [$_{\text{N}}$ "It's been a steadily improving relationship,"] [$_{\text{S}}$ says Mr. Carpenter.]
Length of N: 13 syllables
Length of S: 5 syllables

5.3.2.2 Analysis

In order to investigate the effect of prosodic weight of DUs on their order, the length differences between nuclei and satellites, $\delta = length(\text{N}) - length(\text{S})$, were calculated. Table 5.2 presents the summary statistics of the δ values for the whole sample sentences. Note that when the satellite is prosodically heavier than the nucleus, we get a negative number, and when it is lighter than the nucleus, we get a positive number. According to the last row of Table 5.2, nuclei on average are 7.50 syllables longer than satellites. But when we split the sentences into NS and SN groups a remarkable difference manifests itself. The nuclei and satellites in the NS group are roughly the same size while in the SN group nuclei are on average 15.39 syllables longer than satellites. Note that satellites, in general, are shorter than nuclei simply by virtue of being satellites (i.e., auxiliary to the point the speaker/author is trying to make). The average length of nuclei in our sample is 26.20 syllables; whereas, the average length of satellites is 10.47. But the fact that sentences with smaller satellites and longer nuclei tend to come in SN order is interesting (see Figure 5.10 on the following page).

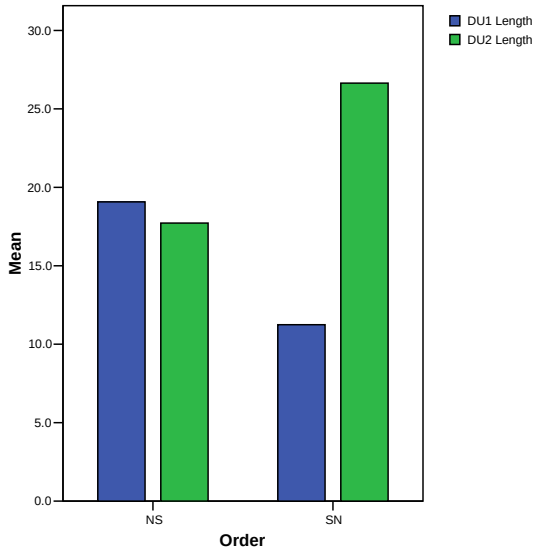


Figure 5.10: Mean DU lengths in NS and SN groups

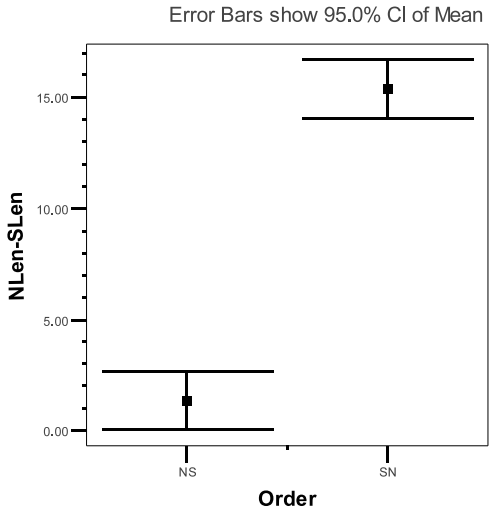


Figure 5.11: Comparison of δ values in NS and SN groups

Are these differences statistically significant? An independent samples t-test on the DU length differences for the NS and SN groups showed that the differences observed in the distribution of these measures is statistically significant ($t = -14.866$, $\mathrm{df} = 1458.959$, $p < .05$, see also Figure 5.11 on the preceding page). The breakdown of δ value distribution into different relational subgroups shows that this overall difference arises from ATTRIBUTION, BACKGROUND, ENABLEMENT and EXPLANATION ($p < .05$, see Figure 5.12 on the following page; for some examples see (5.4)–(5.7) below.). The distribution of the δ values for each group is provided in Tables 5.3 to 5.6 on pages 55–56.

(5.4) a. **NS:** [USI Far East will hold a 60% stake in Luzon Petrochemical,] [according to papers signed with the Philippine government's Board of Investments.]

b. **SN:** [Dr. Lourie says] [the Justice Department interviewed him last July.]

(5.5) a. **NS:** [But even that niche is under attack,] [as several Wall Street firms pulled back from program trading last week under pressure from big investors.]

b. **SN:** [As previously reported,] [a member of the Philippines' House of Representatives has sued to stop the plant.]

(5.6) a. **NS:** [The Court of Appeals for the Federal Circuit was created in 1982] [to serve, among other things, as the court of last resort for most patent disputes.]

b. **SN:** To answer the brokerage question,] [Kidder, in typical fashion, completed a task-force study.]

(5.7) a. **NS:** [In 1988, Kidder eked out a $46 million profit,] [mainly because of severe cost cutting.]

b. **SN:** [Because we refuse to face the tough answers,] [the questions continue as fodder for the commissions and committees, for the media and politicians.]

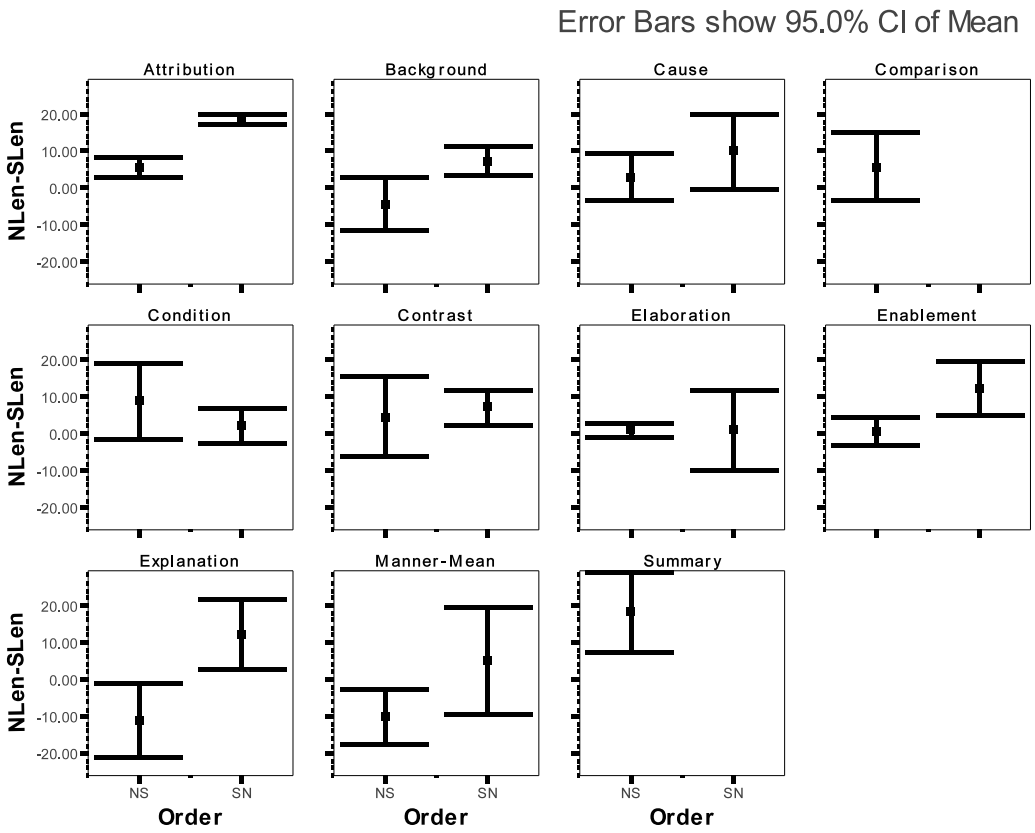


Figure 5.12: Comparison of δ values in NS and SN groups divided by relation

DU Length Difference (δ)		**Order**	
		NS	SN
$-59 \leq \delta < -23$	Percent	53.8%	46.2%
	Count	7	6
$-23 \leq \delta < 13$	Percent	31.4%	68.6%
	Count	81	177
$13 \leq \delta < 49$	Percent	13.2%	86.8%
	Count	41	270
$49 \leq \delta < 85$	Percent	0.0%	100.0%
	Count	0	22
Overall	Percent	21.4%	78.6%
	Count	129	475

Table 5.3: Comparison of the distribution of δ values for the sentences in the NS and SN groups, (Rel=ATTRIBUTION)

DU Length Difference (δ)		**Order**	
		NS	SN
$-59 \leq \delta < -23$	Percent	85.7%	14.3%
	Count	6	1
$-23 \leq \delta < 13$	Percent	31.3%	68.7%
	Count	21	46
$13 \leq \delta < 49$	Percent	30.4%	69.6%
	Count	7	16
$49 \leq \delta < 85$	Percent	0%	100.0%
	Count	0	1
Overall	Percent	34.7%	65.3%
	Count	34	64

Table 5.4: Comparison of the distribution of δ values for the sentences in the NS and SN groups, (Rel=BACKGROUND)

DU Length Difference (δ)		**Order**	
		NS	SN
$-59 \leq \delta < -23$	Percent	100.0%	0.0%
	Count	4	0
$-23 \leq \delta < 13$	Percent	87.9%	12.1%
	Count	51	7
$13 \leq \delta < 49$	Percent	78.6%	21.4%
	Count	11	3
$49 \leq \delta < 85$	Percent		
	Count		
Overall	Percent	86.8%	13.2%
	Count	66	10

Table 5.5: Comparison of the distribution of δ values for the sentences in the NS and SN groups, (Rel=ENABLEMENT)

DU Length Difference (δ)		**Order**	
		NS	SN
$-59 \leq \delta < -23$	Percent	100.0%	0.0%
	Count	7	0
$-23 \leq \delta < 13$	Percent	84.6%	15.4%
	Count	11	2
$13 \leq \delta < 49$	Percent	50.0%	50.0%
	Count	3	3
$49 \leq \delta < 85$	Percent		
	Count		
Overall	Percent	80.8%	19.2%
	Count	21	5

Table 5.6: Comparison of the distribution of δ values for the sentences in the NS and SN groups, (Rel=EXPLANATION)

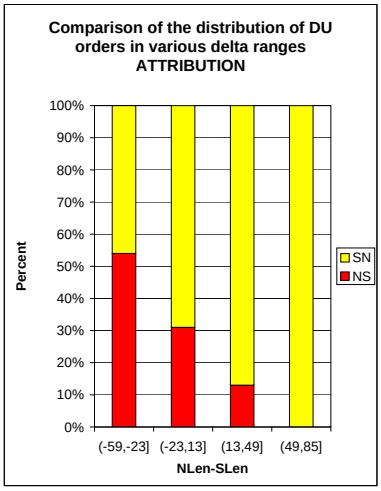


Figure 5.13: Visualization of Table 5.3

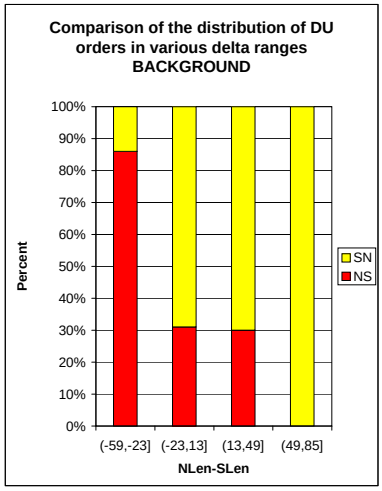


Figure 5.14: Visualization of Table 5.4

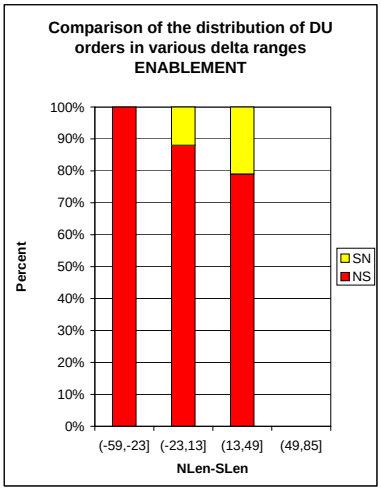


Figure 5.15: Visualization of Table 5.5

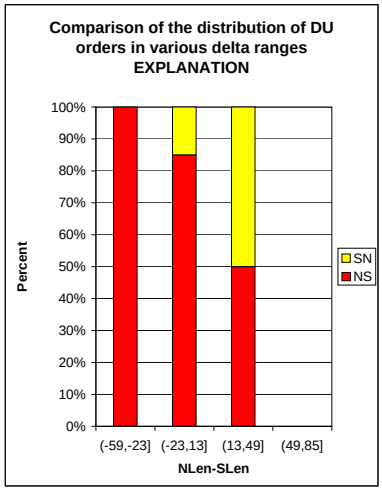


Figure 5.16: Visualization of Table 5.6

5.3.3 What DU order is unmarked?

Mann and Thompson (1988b, p. 256) provide a list of canonical DU orders in text based on the relations involved. This list is given in Table 5.7. Note that this classification is based on the structure of text which includes suprasentential as well as the subsentential units we have used in this study. The statistics that we have gathered here can help us verify if Mann and Thompson's claims also hold for subsentential units alone. Also as noted in section 5.3.1.3, Mann and Thompson's list does not say anything about ATTRIBUTION or EXPLANATION. The insight we gain in this section also helps us find the canonical DU order for these two relations.

Unmarked DU Order	
NS	SN
ELABORATION	ANTITHESIS
ENABLEMENT	BACKGROUND
EVIDENCE	CONCESSIVE
PURPOSE	CONDITION
RESTATEMENT	JUSTIFY
	SOLUTION

Table 5.7: Canonical DU order in text adapted from Mann and Thompson (1988b, Table 2)

According to Table 5.3 on page 55, the canonical DU order for BACKGROUND is SN. This certainly makes sense. It is more logical to set the background for what one is trying to say before one actually says it. Yet as Table 5.4 on page 55 demonstrates, this is not always the case. Clearly, among other things, the prosodic weight of the DUs influences their order. When the satellite is heavier than the nucleus, the sentence is more likely to come in the NS order than when the nucleus is heavier than the satellite. Therefore, the best way to find the canonical DU order is not to look at the overall distribution, which could be misleading, but to consider the cases that are least influenced by external factors (in this case prosodic weight, for instance). Thus looking at the two middle rows of Table 5.4, we see that when nuclei and satellites are more or less the same weight, we observe the SN order almost 70% of the time, which suggests that the canonical DU order for BACKGROUND is SN. This conforms with Mann and Thompson's observation. Using the same technique, we can verify that the canonical DU order for ENABLEMENT is NS because where prosodic weight has the least influence, the NS order is observed about 80% of the time.

Turning back to ATTRIBUTION, we can now claim that the canonical DU order for this relation is SN because when prosodic weight has the least influence this

order is observed about 70–80% of the time (see Table 5.3 on page 55). The same conclusion can also be drawn by looking at Table 5.1 on page 49 where we see that when the sentences are least N-marked, they are likely to come in the SN order about 80.0% of the time (see Table 5.5 on page 56). In addition, when we select only the cases that are least influenced by prosodic weight (i.e., $-10 \leq \delta \leq 10$) *and* are least marked (i.e., $\nu \leq 4$), we observe the SN order 81.25% of the time, which strongly suggests that the unmarked order for the ATTRIBUTION group is SN. As for EXPLANATION, we can conclude that the canonical DU order is NS (see Table 5.6 on page 56).

5.3.3.1 Constraint Resolution

Recall that in section 5.3.1.3 we said that syntax prefers the SVO order in English. And in the previous section we concluded that discourse prefers the SN order for ATTRIBUTION, a constraint which, if satisfied, would result in the syntactically preferred word order (SVO). We also found out that prosody prefers heavy discourse units to come after light ones (call it LH). It is then not surprising to see most ATTRIBUTION sentences in the SN rather than the NS order (78.6% vs. 21.4%) given that two constraints from syntax (SVO) and discourse (SN) prefer this order. However, looking back to Table 5.3 on page 55, under ATTRIBUTION, we notice that when the satellite is considerably heavier than the nucleus (the first row of the table), more sentences appear in the NS order than in the SN order (53.8% vs. 46.2%). This means that in these cases the prosodic constraint LH has gained so much strength that it has overridden the two strong syntactic and discourse constraints. But as this difference in prosodic weight gets smaller, the prosodic constraint also loses its strength (see the second row of the table). And when the nucleus gets heavier than the satellite (the third and fourth rows of the table), LH gangs up with SN and SVO, which results in the appearance of 80–100% of the sentences in the SN order. This gaining and losing of strength by the constraint LH is reminiscent of constraint re-ranking in OT; yet this "re-ranking" happens on a continuous scale, which cannot be implemented in OT or LOT.

5.4 Modelling the Data

In this section, we build a statistical model of the constraints discussed above. The statistical model of our choice is the logistic regression model, which is used to predict a binary result (in this case, NS or SN order) from a series of continuous (and possibly non-continuous) independent variables (e.g., prosodic weight, markedness etc.).

The use of logistic regression for building statistical models is not uncommon in linguistics. For example, Leech et al. (1994) use logistic regression to model the use of genitive and partitive constructions in English based on semantic and other factors. Similarly, Riezler et al. (2000) use a log-linear model, which is in the same family of statistical models as logistic regression (i.e., generalized linear models or GLMs for short), to estimate the parameters of a constraint-based grammar using corpus data. Sankoff (1988) explains how logistic regression is used in sociolinguistic analyses.

Our goal here is to build a parsimonious statistical model to approximate the distribution of NS and SN order in the corpus sentences. And one of the advantages of logistic regression is that it allows for adding and deleting predictor variables in order to test their effects and/or interactions with other variables.

Logistic regression usually makes use of log odds ratio (logit value), $\ln\left(\frac{\pi(x)}{1-\pi(x)}\right)$, as the dependent variable (where $\pi(x)$ is the probability of the event in question happening given x, i.e., $p(\mathrm{NS}|x)$ in our case). The logit model solves the following problem:

$$\ln\left(\frac{\pi(x)}{1-\pi(x)}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_k x_k \tag{5.8}$$

which is equivalent to

$$\frac{\pi(x)}{1-\pi(x)} = \exp\beta_0 \times \exp\beta_1 x_1 \times \exp\beta_2 x_2 \times \ldots \times \exp\beta_k x_k \tag{5.8′}$$

where:

- $x_1 \ldots x_k$ are the independent variables;
- β_0 is a constant called the *intercept*;
- $\beta_1 \ldots \beta_k$ are the variable coefficients.

Given this we can calculate the probability of the event in question happening given a set of values for the independent variables by the following equation:[5]

$$\pi(x_1, x_2, \ldots, x_k) = \frac{1}{1+\exp(-\beta_0 - \beta_1 x_1 - \beta_2 x_2 - \ldots - \beta_k x_k)} \tag{5.9}$$

[5]For more information on logistic regression, see Hosmer and Lemeshow (1989); and for information on the use of logistic regression in language studies, see chapter 9 of Rietveld and van Hout (1993) and chapter 1 of Oakes (1998) as well as the references therein.

Group	Intercept		Variables			
			NLen−SLen		N-markedness	
	Value	Std. Err.	Coefficient	Std. Err.	Coefficient	Std. Err.
ATTRIBUTION	−1.80	0.67	−0.05	0.01	0.23	0.14
BACKGROUND	−0.56	0.22	−0.04	0.01		
ENABLEMENT	2.22	0.43	−0.05	0.02		
EXPLANATION	1.48	0.59	−0.06	0.03		

Table 5.8: Parameters of the GLM-1 model

The parameters of the resulting models for ATTRIBUTION, BACKGROUND, ENABLEMENT and EXPLANATION (the groups shown to be significantly influenced by the prosodic weights of DUs) are represented in Table 5.8.[6] According to these results, the probability of an ATTRIBUTION sentence appearing in the NS order is calculated as in (5.10).

$$p(\mathrm{NS}|\mathrm{Attribution}, \delta, \nu) = \frac{1}{1 + \exp(1.80 + 0.05\delta - 0.23\nu)} \tag{5.10}$$

where:

- δ is $length(\mathrm{N}) - length(\mathrm{S})$ (in syllables), and
- ν is the N-markedness measure of the sentence.

Similarly, the probability of a sentence in the other groups to appear in the NS order will be calculated using the equations in (5.11) to (5.13). We shall label this model GLM-1.

$$p(\mathrm{NS}|\mathrm{Background}, \delta) = \frac{1}{1 + \exp(0.56 + 0.04\delta)} \tag{5.11}$$

$$p(\mathrm{NS}|\mathrm{Enablement}, \delta) = \frac{1}{1 + \exp(-2.22 + 0.05\delta)} \tag{5.12}$$

$$p(\mathrm{NS}|\mathrm{Explanation}, \delta) = \frac{1}{1 + \exp(-1.48 + 0.06\delta)} \tag{5.13}$$

[6] I used the statistical package R to perform the parameter estimation.

5.4.1 Interaction between N-markedness and Sentence Length

This study also revealed an interesting interaction between sentence length and N-markedness, which is visualized in Figure 5.17 on the facing page. The N-markedness measures of shorter sentences tend to be quite dispersed and as the sentences get longer their N-markedness measures move toward four, which is in the lower middle range in the N-markedness scale. This is not surprising as very low- or high-frequency rules or lexical items have a stronger impact on the markedness of shorter sentences than longer ones. In addition, as shown in Figure 5.18, the NS sentences in the ATTRIBUTION group have a higher variance in their N-markedness measures than their SN counterparts. This variation is such that the bulk of the distribution in the NS group almost completely includes the bulk of the distribution of the SN group even though the difference in the mean of the two groups is statistically significant. On the other hand, Figure 5.19 shows that the bulk of the distributions of the total lengths of the NS and SN sentences are largely disjoint, which makes total length of sentences potentially a better predictor of DU order than N-markedness for the ATTRIBUTION group. Based on these observations, I also built a logistic regression model for this group containing the total length of the sentence as an independent variable. This model predicts the probability of an ATTRIBUTION sentence to appear in the NS order with the equation given in (5.14). We shall call this new model for the ATTRIBUTION group GLM-2. The parameters for this model are summarized in Table 5.9. It should be noted that in a model containing both sentence length and N-markedness, the latter did not improve the accuracy of the model significantly. This is due to the interaction mentioned above.

$$p(\mathrm{NS}|\mathrm{Attribution}, \delta, \lambda) = \frac{1}{1 + \exp(-0.38 + 0.05\delta + 0.04\lambda)} \tag{5.14}$$

where:

- δ is $length(\mathrm{N}) - length(\mathrm{S})$, and
- λ is the total length of the sentence (in syllables).

5.4.2 Evaluation of the models

In this section we compare the results we gained by applying the two GLM-1 and GLM-2 models to both seen and unseen data with a baseline and a naïve model. The baseline simply assigns NS order to sentences with a 50% chance. The naïve

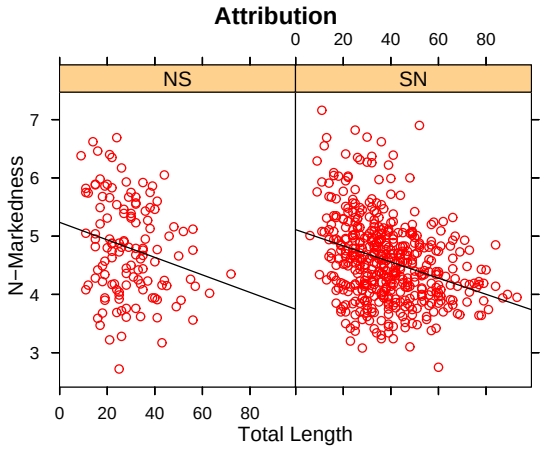


Figure 5.17: Relation between total length and N-markedness in the NS and SN sentences of the ATTRIBUTION group

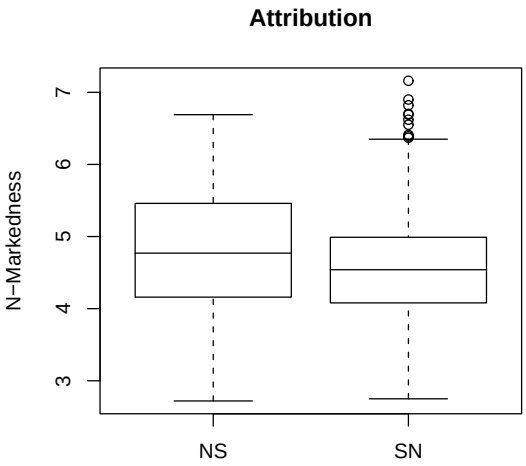


Figure 5.18: Comparison of the distribution of N-markedness measures in the NS and SN subgroups of the ATTRIBUTION group

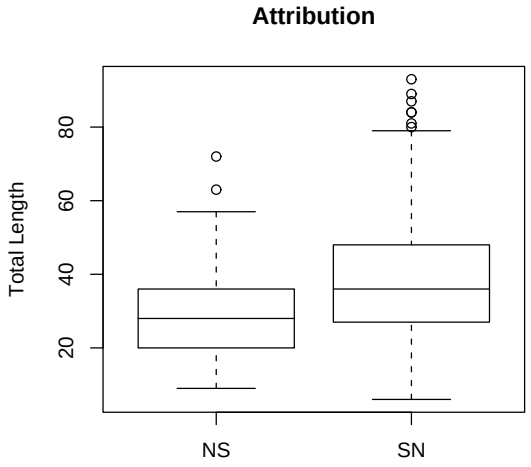


Figure 5.19: Comparison of the distribution of Total Length measures in the NS and SN subgroups of the ATTRIBUTION group

Group	**Intercept**		**Variables**			
			NLen−SLen		Total Length	
	Value	Std. Err.	Coefficient	Std. Err.	Coefficient	Std. Err.
ATTRIBUTION	0.38	0.28	−0.05	0.01	0.03	0.01

Table 5.9: Parameters of the GLM-2 model for the ATTRIBUTION group

		Baseline	Naïve	Binary LH	GLM-1		GLM-2	
	N	$\overline{e}rr$	$\overline{e}rr$	$\overline{e}rr$	$\overline{e}rr$	$\hat{E}rr$	$\overline{e}rr$	$\hat{E}rr$
ATTRIBUTION	604	0.501	0.228	0.339	0.213	0.238	0.200	0.200
BACKGROUND	98	0.501	0.457	0.306	0.265	0.275		
ENABLEMENT	76	0.504	0.238	0.473	0.144	0.145		
EXPLANATION	26	0.520	0.314	0.230	0.269	0.278		

Table 5.10: Comparison of the results of the three models

model, randomly assigns NS order to sentences according to the maximum likelihood estimation of the sentence coming in the NS order, calculated from the corpus data as follows:

$$p(\mathrm{NS}|g) = \frac{count(\mathrm{NS}|g)}{count(\mathrm{NS}|g) + count(\mathrm{SN}|g)} \tag{5.15}$$

where g represents the relation groups ATTRIBUTION, BACKGROUND, ENABLEMENT, and EXPLANATION. The Binary LH model makes a binary decision by assigning NS order if $weight(\mathrm{S}) > weight(\mathrm{N})$ and SN otherwise.

In order to see how well the logistic regression model fits unseen data, a leave-one-out cross-validation experiment was performed on the logistic regression model and its true error rate was estimated.[7] Table 5.10 shows a summary of the results. Note that $\overline{e}rr$ denotes the apparent error rate (i.e., the error rate of the model based on seen data only), while $\hat{E}rr$ is the estimated true error rate of the population (i.e., the error rate of the model based on seen *and* unseen data). According to these results, our logistic regression model (GLM, after generalized linear model) performs on seen and unseen data better than the other models do on only seen data.

5.5 Summary

This chapter was devoted to a descriptive analysis and statistical modelling of the influence of syntactic markedness and prosodic weight on the order of discourse units in sentences containing one nucleus and one satellite. We noticed collaborative/conflicting constraints from prosody and discourse in ATTRIBUTION, BACKGROUND, ENABLEMENT and EXPLANATION groups. We also noticed that the constraints involved showed a graded strength depending on prosodic weight difference between nuclei and satellites as well as the overall markedness measure of a

[7] This was also performed using the statistical package R.

sentence. The measure of syntactic markedness that we employed in this chapter seemed too gross to allow us to make strong predictions. Further investigation on the role of syntax on discourse order using more sophisticated models is required. The following chapters will be devoted to developing a model of grammar that is capable of handling such constraints but at the same time is not at odds with existing theories that attempt to model gradience.

Chapter 6

Conflicts and Modularity

6.1 Introduction

This chapter focuses on the interface between discourse and phonology and provides support for our working assumption that a parallel modular architecture of grammar is possible, resulting in more straightforward accounts of our observations by allowing us to formulate more general constraints. In section 6.2, we look at the well-known problem of syntactic vs. prosodic constituency mismatches and show that deriving the prosodic structure independently with little reference to syntactic structure is promising. In section 6.3 we argue that this modular architecture needs soft constraints to resolve the conflicting demands from different modules.

6.2 Prosodic vs. Syntactic Constituency

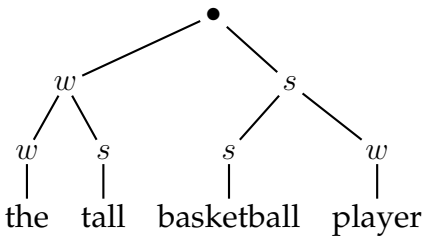
6.2.1 Background

Further motivation for adopting a modular architecture comes from the many mismatches between syntactic and prosodic structures. As Zwicky (1982) puts it,

> [t]he divergence between the syntactic and phonological organizations of the same material has long been recognized as a problem in analysis and a challenge to theorizing, finding recognition in works as diverse as Kahane and Beym (1948); Pulgram (1970); Bing (1979); Cooper and Paccia-Cooper (1980) and the writing of the 'metrical phonologists', in particular [Selkirk (1981a)].

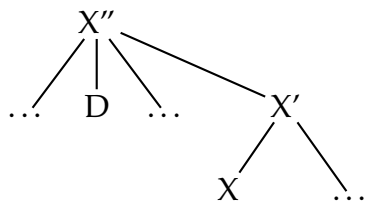
Initially, mainstream linguistics assumed that the prosodic structure mirrors syntactic structure unless otherwise specified in order to satisfy certain phonological constraints. These constraints, however, render virtually every prosodic structure different from the syntactic structure of the same sentence. This section argues for the treatment of prosodic and syntactic constituency as two distinct entities. Let us start with what has come to be known as the *Prosodic Isomorphism Hypothesis* (PIH). Liberman and Prince (1977) in their seminal work on metrical phonology assumed that prosodic structure was isomorphic to syntactic structure. They assumed that each node in the syntactic tree is labelled as strong (s) or weak (w). This was considered to capture the relative strength of each constituent in its local structural context. Pitch accents were assigned to the terminal elements that were dominated by the largest number of s nodes. For example in (6.1), *tall* is relatively stronger than *the*, and *basketball* is in turn stronger than *player*. At the next higher level we see that *the tall* is marked relatively weaker than *basketball player*. *Basketball* is then said to take the pitch accent because it has the largest number of s nodes above it. By default, it is assumed that the second daughter in a binary branching subtree is stronger in English unless otherwise specified, as in the above example where the compound stress rule has applied.

(6.1)

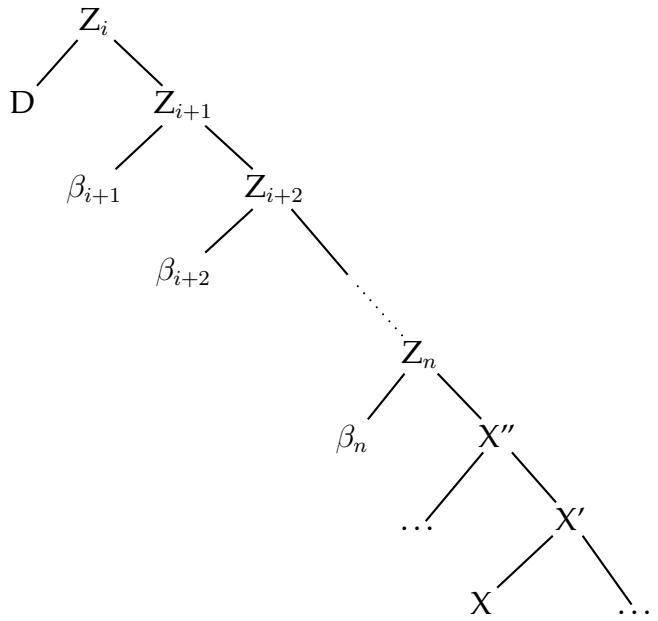


Later on Selkirk (1981b) showed that monosyllabic words get destressed and (prosodically) associated with the next word if they are non-lexical (i.e., functional) and syntactically dependent on another category. She defines syntactic dependence as basically the structural relation that exists between a node in the specifier position (or in an adjunct attached to the specifier) in a subtree and the head of that subtree. Therefore in (6.2a, b), the D node is considered a syntactic dependent of X. The β nodes are the adjuncts that are attached to X".

(6.2) a.



b.



This specification was motivated based on PIH and data such as (6.3)–(6.5). What was observed in these data was that frequently small function words, such as prepositions and determiners, are cliticized, resulting in a prosodic structure that is not isomorphic to syntactic structure.

(6.3) a. thě wóman
b. ĭts impórtance
c. sŏ sóundly

(6.4) a. ĭn thě róom
b. ăt hĕr requést
c. fŏr thě tìme béing

(6.5) a. They wĕre ĭn a cŏlléctive.
b. Lou wăs ùnder thě wéather.

Soon after this observation, Selkirk (1981a) argued,

> there is not an isomorphism between prosodic structure and syntactic structure, rather…prosodic structure is an entity distinct, and…a mapping of a non-trivial sort must be defined between it and syntactic structure.

She then defines the prosodic categories *prosodic words*, *prosodic phrases*, and *intonation phrases* as follows:

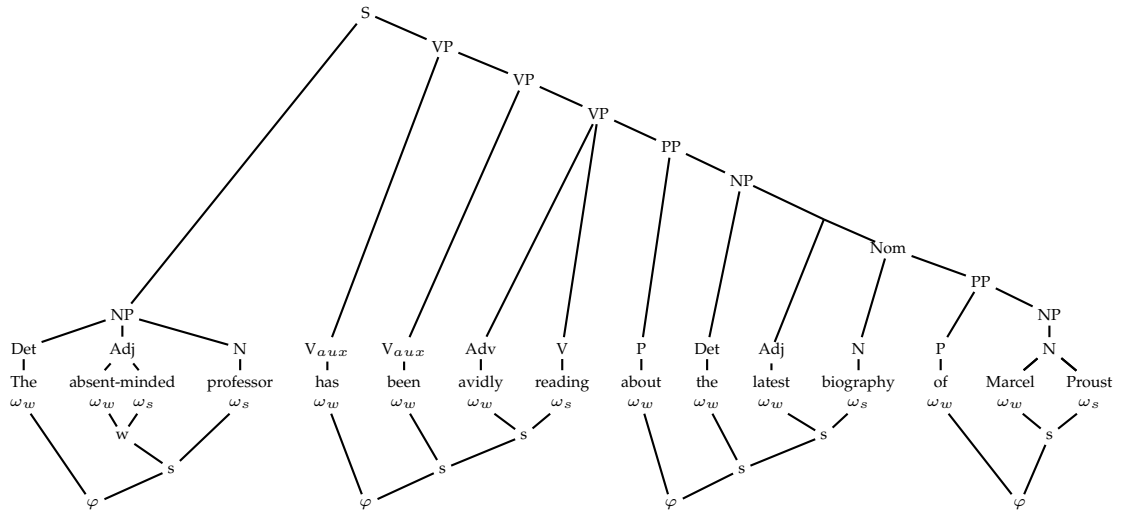


Figure 6.1: Mismatch between prosodic and syntactic constituency (Selkirk, 1981a)

(6.6) *The Intonation Phrase* (Selkirk, 1981a, p. 134):

a. *Constituency:* The I [Intonational Phrase] is composed of φ [Phonological Phrase] joined in a right branching structure.

b. *Prominence:* In I, the nodes N_1, N_2 are in the relation w/s.

c. *Syntactic Domain:*

i. Parentheticals, preposed adverbials, non-restrictive relative clauses, etc. are Is.

ii. Otherwise, the choice is free. [i.e., other syntactic categories may or may not form their own Is.]

(6.7) *The Phonological Phrase Constituency Rule* (p. 126):

a. An item which is the specifier of a syntactic phrase joins with the head of the phrase.

b. An item belonging to a "non-lexical" category, such as Det, Prep, Comp, $Verb_{aux}$, Conjunction, joins with its sister constituent.

Obviously at this point Selkirk does not believe in PIH anymore; yet, she still believes that phonological structure should be derived from syntactic structure (in a "non-trivial" manner). An example of the mismatch between the two constituencies is given in Figure 6.1.

Later, in her treatment of focus and its phonological realization, Selkirk (1984), posits a new level of representation that she calls *intonation structure* upon which certain other rules apply. First, she assumes that *focus* is a syntactic feature and

hence it is assigned to syntactic constituents. Given the architecture of the Government and Binding (GB) theory at the time, this seemed to be the only option as syntax was the only generative component of the theory. *Focus* is then projected up the tree just in case one of the daughters that is the head or an argument of the head bears the *focus* feature. Finally, a pitch accent is assigned to a constituent that bears the *focus* feature. These ideas are shown in (6.8) and (6.9) below.

(6.8) *Basic Focus Rule:* A constituent to which a pitch accent is assigned is a focus.

(6.9) *Phrasal Focus Rule:* A constituent may be a focus if (a) or (b) (or both) is/are true:
   a. The constituent that is its [syntactic] *head* is a focus.
   b. A constituent contained within it, that is an *argument* of the head, is a focus.

In order to account for the mismatches between syntactic structure and intonation (prosodic) structure, Selkirk then introduces the notions of *sense unit* and *sense unit condition* as follows:

(6.10) *Sense Unit:* Two constituents C_i, C_j form a sense unit if (a) or (b) is true of the semantic interpretation of the sentence:
   a. C_i modifies C_j (a [syntactic] head)
   b. C_i is an argument of C_j (a [syntactic] head) or contains the argument up to the head of C_i [meaning that the postmodifiers of C_i may be excluded].

(6.11) *Sense Unit Condition:* Daughters of an intonation phrase must form a sense unit.

The *Sense Unit Condition* takes care of cases where a syntactic constituent is divided up into two or more prosodic constituents. For example, (6.12a) is acceptable because *John* and *the book* are both arguments of the head *gave* and they are all inside the same intonation phrase; thus, the *Sense Unit Condition* is observed. On the other hand, (6.12b) is unacceptable because in *the book to Mary* neither noun phrase is an argument or a modifier of the other, and the fact that they both appear in the same constituent without the verb *gave* violates the *Sense Unit Condition*. Note that (6.12c) does not violate the *Sense Unit Condition* as both noun phrases are arguments of *gave*.

(6.12) a. [Jane gave the book] [to Mary]
   b. * [Jane gave] [the book to Mary]

c. [Jane] [gave the book to Mary]

The introduction of a new level of representation called *intonation structure* led to a complicated theoretical architecture with the intonation structure generated based on PF and LF. This architecture is an indication that perhaps syntactic structure is not suitable as a basis for phonological structure. This complexity, I will argue, is the result of adherence to a syntactocentric grammar architecture which leads to confusion as to the nature of such linguistic phenomena as focus. Focus is inherently a semantic/discourse phenomenon with syntactic and prosodic correlates.

Another argument against PIH and other such approaches that derive prosodic structure from syntactic structure was given by Nespor and Vogel (1986), who argue that syntactic constituents are inappropriate as the domains of application of phonological rules. They present three arguments for their claim. These arguments are summarized below. As a solution, they propose a derivational model within the generative phonology framework.

(6.13) Nespor and Vogel's (1986) arguments against treating syntactic structure as a domain of phonological rules:
   a. Direct reference to syntactic constituents does not make the correct predictions about the domains of phonological rules.
   b. Whereas syntactic constituency is determined uniquely in terms of structural factors, a non-structural factor, the length of a given string, is relevant to phonology.
   c. There exist phonological rules that apply in larger domains than sentences.

However, Steedman (1991, 2000a,b) argues that the distinction between prosodic constituency and syntactic constituency is spurious. He claims that treating the two types of constituency differently unnecessarily complicates the theory, and provides a series of type raising operations to derive a structure that matches prosodic constituency but at the same time gives the correct logical form all in a syntactocentric approach.

6.2.2 A Parallel Architecture Approach

Here we propose a parallel architecture of grammar which averts the mismatch problem discussed in the preceding subsection.[1] The model is designed to address information structure and prosody correspondence in the constraint-based

[1]Earlier versions of this analysis appear in Haji-Abdolhosseini (2003a,b).

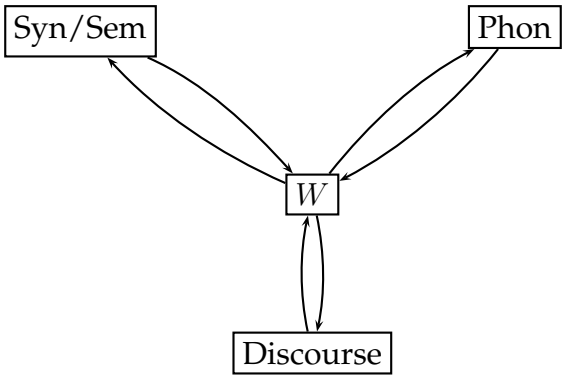


Figure 6.2: Proposed modular architecture

theory of HPSG. This architecture is depicted in Figure 6.2. According to that model, the syntactic/semantic, prosodic and information structures are all constructed from a single list of lexical items, W. The arrows pointing from W to various structures represent well-formedness constraints on those structures. The arrows pointing back to W represent constraints on the features of the members of W imposed by those structures. We shows that since a modular grammar architecture is not syntax-driven, prosodic structure can be defined in parallel with syntactic structure over a list of words commonly accessed from syntax, phonology, and discourse. As that work was done within the HPSG framework, the structural constraints were the ones found in standard HPSG literature such as rule schemata and the like. Haji-Abdolhosseini (2003a) argued that this architecture captures several generalizations with a few simple and very general constraints and avoids the need for certain *ad hoc* constraints that previous approaches called for.

The modular model proposed in Haji-Abdolhosseini (2003a,b) straightforwardly accounts for the phenomena that Butt and King (1998) call "prosodic promotion", and "prosodic flattening" (discussed in the following subsection) without having to manipulate syntactic structures. In addition, information structure-prosody correspondence is handled elegantly in a parallel modular fashion without recourse to unnecessary and *ad hoc* operations and/or levels of representation. What follows is a recapitulation of the proposals made in Haji-Abdolhosseini (2003a).

As stated above, Haji-Abdolhosseini (2003a) lays the groundwork for a unification-based model of prosody sensitive to the syntax and information structure of the sentence. The approach adopted is modular, and the theory developed derives syntactic and prosodic structures at different layers interacting only at the interfaces. That paper was a response to the claim made by the proponents of Combinatory Categorial Grammar (CCG, Steedman, 1991, 2000b; Prevost

and Steedman, 1994; Prevost, 1995) that modular theories are overly complicated and unconstrained. Haji-Abdolhosseini (2003a) claims that by making use of sufficient constraints on each module, we *can* have a theory with very simple subcomponents that are more readable, extensible, and maintainable. The analysis presented in that paper builds on ideas proposed in Klein (2000), but departs from the syntactocentric approach adopted in that work.

The next subsection reviews some background information necessary for understanding what follows. Then we will go over the data that Haji-Abdolhosseini (2003a) accounts for. Subsection 6.2.3 presents a formal account of the data.

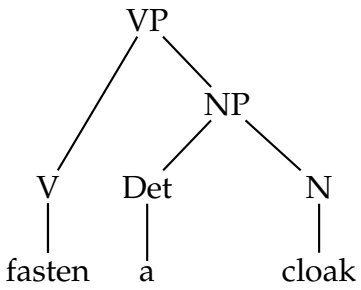
6.2.2.1 Preliminaries

Klein (2000) lays out a framework for analyzing prosodic constituency in HPSG. He takes Liberman and Prince's (1977) PIH as his starting point. According to Liberman and Prince's approach, syntactic constituents are assigned relative prosodic weight based on the *Nuclear Stress Rule (NSR)*.

(6.14) *NSR*: In a configuration [$_C$ A B], if C is a phrasal category, B is strong.

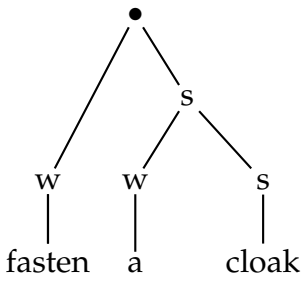
Consider the following phrase for example:

(6.15)



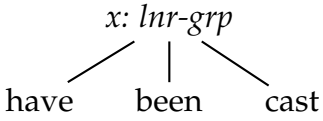
According to *NSR*, the constituent in (6.15) will have the following prosodic structure. (*W* stands for 'weak' and *s* for 'strong'. The top node is left unlabelled.)

(6.16)



Klein adopts Zwicky's (1982) term, *leaner*, to refer to a class of words that "form a rhythmic unit with the neighbouring material, are normally unstressed with respect to this material, and do not bear the intonational peak of the unit" (Zwicky, 1982, p. 5).[2] In Zwicky's words, "English articles, coordinating conjunctions, complementizers, relative markers, and subject and object pronouns are all leaners in this sense" (ibid.). Klein refers to a group of leaners followed by a single prosodic word as a *leaner group*. Such a group forms one node in the metrical tree and is represented as *x: lnr-grp* where *x* stands for a weak or strong accent. Look at (6.17) for example:

(6.17)



Klein's (2000) HPSG analysis accounts for two types of mismatch between syntactic and prosodic constituency: **Prosodic Flattening** and **Prosodic Promotion**. *Prosodic Flattening* is the term used to indicate that prosodic structure tends to be flatter than syntactic structure, e.g., (6.17). Klein (2000, p. 177) schematically shows this as in (6.18). *Prosodic Promotion* refers to a similar type of mismatch, which occurs when "the complement or postmodifier of a syntactic head is 'promoted' to the level of the sister of the constituent in which the head occurs" (Klein, 2000, p. 179) (e.g., (6.24)). This is shown schematically in (6.19). In (6.19b), ϕSpec and ϕX represent the prosodic contribution of Spec and X, respectively.

6.2.2.2 Prosodic Constituency in HPSG

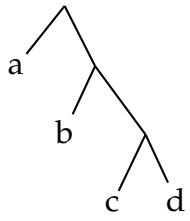
Klein (2000) extends HPSG's standard unstructured phonology to account for the phonology-syntax mismatches mentioned above as follows: He first represents prosodic structure using feature structures. Then, he defines a relation between phonology values and metrical trees, and finally he incorporates the relation into prosodic constraints within a constructional hierarchy.

The prosodic hierarchy that Klein proposes is an extension of Zwicky's dichotomy between *leaners* and *prosodic words*. According to this type hierarchy (represented in Figure 6.3 on page 77), prosodic types are either *leaners* or *full*. Type *full* immediately subsumes prosodic words (*p-wrd*) and metrical trees (*mtr(τ)*). The latter, being a parametric type,[3] subsumes metrical trees that include either *leaners*

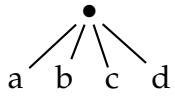
[2] The term *clitic* is also commonly used for the same notion.

[3] A parametric type is a polymorphic type that changes shape according to some parameter. For example, *mtr($\tau \lessdot pros$)* is a metrical tree of either leaners or full objects depending on the parameter τ.

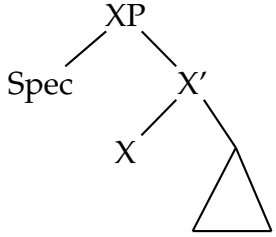
(6.18) a. **Syntactic Configuration:**



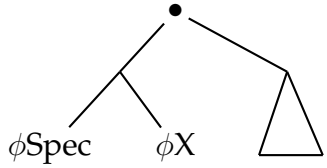
b. **Prosodic Configuration:**



(6.19) a. **Syntactic Configuration:**



b. **Prosodic Configuration:**



or objects of type *full*. Type *mtr(τ)* has two appropriate features: DOM is the domain feature introduced in Kathol (1995) and Reape (1994) that makes use of the domain union relation ◯.[4] The value of DOM has been formulated so that it holds a list of prosodic objects. One of these objects is labelled as the *Designated Terminal Element*—by structure-sharing with the value of DTE, which is constrained to be of type *full*. A metrical tree of type *lnr*, *mtr(lnr)*, places the more stringent constraint on the value of DTE that it be of type *p-wrd*. Objects of type *mtr(full)* have a list of other *full* objects as the value of their DOM feature.

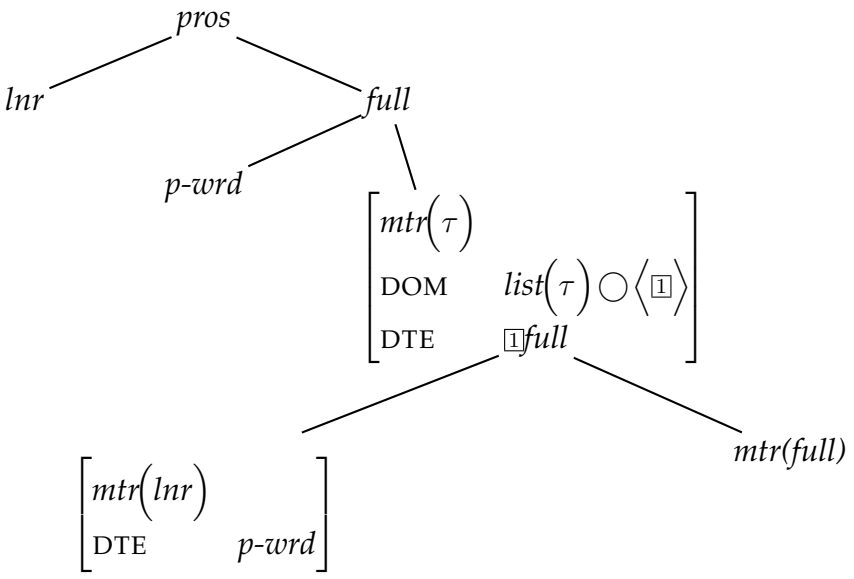


Figure 6.3: Prosodic Type Hierarchy (Klein, 2000)

The mkMtr function provides a mapping from lists of prosodic objects to larger prosodic structures. The value of τ is restricted to the types covered (i.e., immediately subsumed) by *pros*. Klein uses $\lessdot$ to show this. The assumption implicit in the definition of the mkMtr function is that more specific rules take precedence over more general ones. (6.20a-i) accounts for cases where mkMtr^{τ} has a singleton list of one prosodic element in its argument. (6.20a-ii) builds a standard metrical tree with the last element labelled DTE as a default for English which may be overridden by narrow focus. (6.20b) analyses the longest metrically analyzable prefix of its input list as a *mtr(lnr)* and takes the resulting subtree together with the re-

[4]The domain union relation allows the combination of two lists in a manner similar to shuffling a deck of cards. The relative orders of the members of the original lists are maintained in the resulting list, but adjacent members may be separated by members from the other list. In this context, the relation is simply used to non-deterministically pick out a member of a list.

maining elements to make a *mtr(full)*. This accounts for cases like (6.21)[5] where the input to mkMtr^{τ} cannot be exhaustively analyzed as either *mtr(lnr)* or *mtr(full)*. The application of mkMtr^{lnr} and mkMtr^{full} to (6.21c) is shown in (6.22).

(6.20) **The** mkMtr **Function (adapted from Klein (2000))**

a. $\mathsf{mkMtr}^{\tau \lessdot pros} : list(pros) \mapsto mtr(\tau)$

i. $\mathsf{mkMtr}^{\tau}(\langle \boxed{1} pros \rangle) = \boxed{1}$

ii. $\mathsf{mkMtr}^{\tau}(\langle \boxed{1}, \boxed{2}, \ldots, \boxed{n} \rangle) = \begin{bmatrix} mtr\Big(\tau\Big) & \\ \text{DOM} & \Big\langle \boxed{1}, \boxed{2}, \ldots, \boxed{n} \Big\rangle \\ \text{DTE} & \boxed{n} \end{bmatrix}$

b. $\mathsf{mkMtr} : list(pros) \mapsto mtr(pros)$
$\mathsf{mkMtr}(\boxed{1} \oplus \boxed{2}) = \mathsf{mkMtr}^{full}(\mathsf{mkMtr}^{lnr}(\boxed{1}) \oplus \boxed{2})$

c. $\mathsf{mkMtr}^{lnr} : list(pros) \mapsto mtr(pros)$
$\mathsf{mkMtr}^{lnr}(\langle \boxed{1} lnr, \Big[\text{DOM} \quad \boxed{2}\Big] \rangle) = \mathsf{mkMtr}(\langle \boxed{1} \rangle \oplus \boxed{2})$

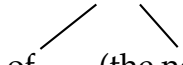
d. $\mathsf{mkMtr}_{\text{LA}} : list(pros) \mapsto mtr(pros)$
$\mathsf{mkMtr}_{\text{LA}}(\boxed{1} \oplus \boxed{2}) = \mathsf{mkMtr}^{full}(\mathsf{mkMtr}^{\tau}(\boxed{1}) \oplus \boxed{2})$

(6.21) a. * (shŏuld hăve béen thóroughly revísed)
b. * [shŏuld hăve béen thóroughly revísed]
c. [(shŏuld hăve béen) thóroughly revísed]

(6.22) $\mathsf{mkMtr}(\langle$should, have, been$\rangle \oplus \langle$ thoroughly revised$\rangle$) =
$\mathsf{mkMtr}^{full}(\mathsf{mkMtr}^{lnr}$(should, have, been), thoroughly revised) =
[(should have been) thoroughly revised]

Because the input to mkMtr^{lnr} is determined by the append operator $\oplus$, mkMtr reduces to mkMtr^{full} when no leaners are prefixed to its input, and to mkMtr^{lnr} when there is only one prosodic word and that prosodic word occurs at the end of the input list. (6.20c) accounts for cases in which an unaccented preposition combines with a complement NP as in (6.23). The definition in (6.20c) allows for the combination of a leaner with a metrical tree.

(6.23) a. (of the necklace)



[5] As in Klein (2000), we use parentheses () in this section to succinctly represent leaner groups, $mtr(lnr)$, and square brackets [] to represent full metrical trees, $mtr(full)$.

b. [(of this) necklace]

of [this necklace]

The mkMtr definitions discussed above account for cases of Prosodic Flattening while the last one (6.20d) accounts for Prosodic Promotion that occurs in head-final constructions and involves grouping of a prehead element with the head as in (6.24).

(6.24) [[this possession](of the samurai)]

this [possession (of the samurai)]

What mkMtr_{LA}[6] does is first make a prosodic tree with a prefix of its input list and then use the resulting tree together with the remainder of the list to create a *full* metrical tree.

Constructions make use of the mkMtr relation to account for proper mappings between syntactic and prosodic structures. This is shown in (6.25). The type *base-pr* makes use of mkMtr; therefore, any type that inherits from *base-pr* also uses mkMtr. According to the type hierarchy in Figure 6.4, this would be the head-complement constructions (*hd-comp-cx*), which are head-initial. On the other hand, mkMtr_{LA} is used by *ext-pr*, which in turn passes down this property to head-specifier constructions (*hd-spr-cx*), which are head-final.

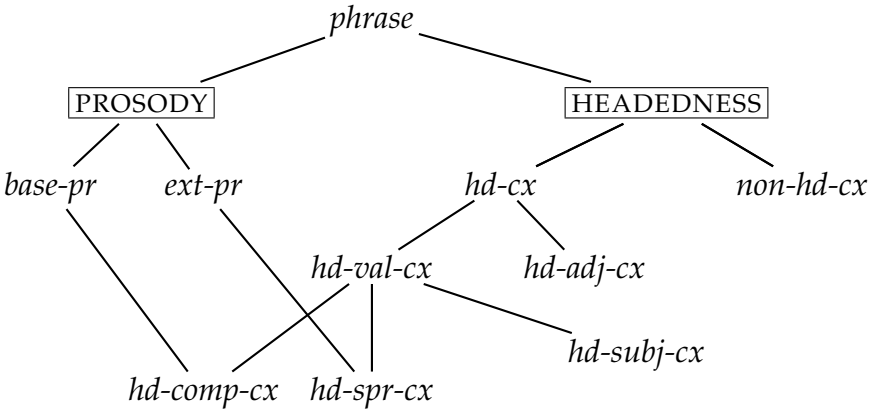


Figure 6.4: Prosody and Headedness (Klein, 2000, p. 191)

[6]LA stands for Left Associating.

(6.25) a. **Head-Initial Constructions**

$$\textit{base-pr} \Rightarrow \begin{bmatrix} \text{MOTHER} \mid \text{PHON} & \mathsf{mkMtr}\left(\boxed{1},\ldots,\boxed{n}\right) \\ \text{DTRS} & \left\langle \begin{bmatrix}\text{PHON} & \boxed{1}\end{bmatrix},\ldots,\begin{bmatrix}\text{PHON} & \boxed{n}\end{bmatrix} \right\rangle \end{bmatrix}$$

b. **Head-Final Constructions**

$$\textit{ext-pr} \Rightarrow \begin{bmatrix} \text{MOTHER} \mid \text{PHON} & \begin{bmatrix} \textit{mtr}\left(\textit{pros}\right) & \\ \text{DOM} & \left\langle \mathsf{mkMtr}_{\text{LA}}\left(\left\langle \boxed{1} \right\rangle \oplus \boxed{2}\right) \right\rangle \\ \text{DTE} & \boxed{3} \end{bmatrix} \\ \text{DTRS} & \left\langle \begin{bmatrix}\text{PHON} & \boxed{1}\end{bmatrix}, \begin{bmatrix}\text{PHON} & \begin{bmatrix}\text{DOM} & \boxed{2} \\ \text{DTE} & \boxed{3}\end{bmatrix}\end{bmatrix} \right\rangle \end{bmatrix}$$

As Klein (2000) points out, $\mathsf{mkMtr}_{\text{LA}}$ overgenerates, allowing for prosodic trees such as the one represented in (6.26). Klein corrects this indirectly by positing a global *Lexical Head Association Constraint* that

> prohibits headed constructions from building a PHON value for the mother where (i) the head daughter has undergone pre-head modification, and (ii) a proper subpart (i.e., [1] [in (6.27) below]) of the head's prosodic structure is left-associated with some material which precedes the head and its modifiers. In other words, if some element left-associates with material preceding a head, then it must include the head itself in the grouping (p. 195).

In other words, what (6.27) does is to disallow the making of a full prosodic tree from a part of a pre-modified head adjunct phrase (*hd-pre-adj-ph*). Klein calls this constraint a partial implementation of Selkirk's (1986; 1996) notion of end-based mappings between syntax and phonology because it effectively aligns the right edge of noun phrases with the right edge of a corresponding prosodic phrase.

(6.26) * [[this treasured] possession (of the samurai)]

this [treasured possession (of the samurai)]

(6.27) **Lexical Head Association Constraint:**

$$\textit{hd-cx} \Rightarrow \neg \left[\begin{array}{ll} \text{MOTHER} \mid \text{PHON} \mid \text{DOM} & \left\langle \begin{bmatrix} \textit{mtr}\left(\textit{full}\right) & \\ \text{DOM} & \textit{ne-list} \oplus \boxed{1} \end{bmatrix} \right\rangle \oplus \boxed{2} \\ \text{HD-DTR} & \begin{bmatrix} \textit{hd-pre-adj-ph} & \\ \text{PHON} \mid \text{DOM} & \boxed{1} \oplus \boxed{2}\ \textit{ne-list} \end{bmatrix} \end{array}\right]$$

6.2.2.3 Data

Let us go over some examples to illustrate the empirical coverage of Klein's interface model. Starting with (6.28), we can see how the application of mkMtr results in a correct derivation of a prosodic tree.

(6.28) I want to begin to try to write a play.

Stepping into the derivation bottom-up and right-to-left, we can trace the working of mkMtr. For example, *a play* is a *hd-spr-cx* and thus also of type *ext-pr*, which employs mkMtr_{LA} according to Klein (2000). As shown in (6.29), the application of mkMtr_{LA} to *a play* results in a metrical tree of type *mtr(lnr)*.

(6.29) $$\text{mkMtr}_{\text{LA}}(\langle \text{a}, \text{play} \rangle) = \text{mkMtr}^{full}(\text{mkMtr}^{lnr}(\langle \text{a}, \text{play} \rangle) \oplus \langle \quad \rangle) =$$
$$\text{mkMtr}^{full}\left(\left\langle \begin{bmatrix} \textit{mtr}\left(\textit{lnr}\right) & \\ \text{DOM} & \left\langle \text{a}, \boxed{1}\ \text{play} \right\rangle \\ \text{DTE} & \boxed{1} \end{bmatrix} \right\rangle\right) = \begin{bmatrix} \textit{mtr}\left(\textit{lnr}\right) & \\ \text{DOM} & \left\langle \text{a}, \boxed{1}\ \text{play} \right\rangle \\ \text{DTE} & \boxed{1} \end{bmatrix}$$

Going through the derivation procedurally in the same manner yields the result shown in (6.30). The following example is frequently mentioned by Steedman (e.g., Steedman, 2000b, 94) as one that needs to be accounted for by any theory that deals with syntax-phonology mismatches.

(6.30) [(I want) [(to begin) [(to try) [(to write) (a play)]]]]

(6.31) * [[(I want) [(to begin) to]][try [(to write) (a play)]]]

In this example a pause has been placed between a leaner and the prosodic word that it leans on. Clearly, a pause should not be allowed to intervene within leaner groups and we should make provisions in our theory to reject such ill-formed structures.

Klein's account incorrectly marks (6.32) ungrammatical as *I*, being a personal pronoun is considered a leaner in that model.

(6.32) [[I] [want [(to begin) [(to try) [(to write) (a play)]]]]]].

The sentences in (6.32) and (6.33) appear in Steedman (2000b, p. 93). He suggests a model of syntax whose surface structures correspond directly to intonational contours. Thus, in these examples, all of the observed intonational contours correspond to possible surface structures for the sentence in a CCG framework.

(6.33) a. [[(I want)][(to begin) [(to try) ([to write) (a play)]]]].
b. [[(I want) (to begin)][(to try) [(to write) (a play)]]].
c. [(I want) [(to begin) (to try)]][(to write) (a play)]].
d. [[(I want) [(to begin) [(to try) (to write)]]][(a play)]].

We would like to develop a model that not only is able to account for these alternate intonational contours and their corresponding semantics, but also maintains the modularity of its component theories as much as possible. Another example that Steedman (2000b), *inter alia*, discusses is (6.34).

(6.34) *[[Three mathematicians] [(in ten) prefer margarine]].

Selkirk (1984) attributes the ungrammaticality of (6.34) to the violation of the Sense Unit Condition, meaning that the prepositional phrase *in ten* and the verb phrase *prefer margarine* fail to form a sense unit as neither is a complement or modifier of the other. Steedman's CCG model accounts for this. Again, approaching the problem from our standpoint, we would like a multi-partite account for this fact. Another type of data that we want to account for here is:

(6.35) a. [Jane [gave [(the book) (to Mary)]]]
b. [[Jane] [gave [(the book) (to Mary)]]]
c. [[Jane [gave (the book)]] [(to Mary)]]
d. [[Jane gave] [(the book)] [(to Mary)]]
e. * [[Jane] [gave] [(the book) (to Mary)]]
f. * [[Jane gave] [(the book) (to Mary)]]
g. [[Jane] [[gave (the book)] [(to Mary)]]]
h. [[Jane] [gave] [(the book)] [(to Mary)]]

These data have been discussed in Selkirk (1984), and similar examples have been talked about in Steedman (2000a). Selkirk (1984) also attributes the ungrammaticality of (6.35e, f) to the violation of the Sense Unit Condition: The phrases *the book* and *to Mary* do not combine to form a sense unit because neither is a complement or modifier of the other.

6.2.3 Analysis

6.2.3.1 Information Status and Intonation

Like Steedman, who adopts a Hallidayan tradition, Haji-Abdolhosseini (2003a) uses the term *theme* to refer to given information and *rheme* to new information.[7] Steedman (2000b, p. 101), following Pierrehumbert (1980), attributes L+H* LH% intonation contour to theme and H*LL% to rheme.[8] L+H* LH% and H*LL% are in Pierrehumbert's notation (Pierrehumbert, 1980), and respectively correspond to *rise-fall-rise* and *fall* intonation in British style (Ladd, 1996, p. 82). Going back to our example about writing a play (extended here as (6.36)), we can discuss some of the interaction between information structure and prosody. Hereafter, $_\theta$ stands for *theme* and $_\rho$ for *rheme*.

(6.36) a. $[\text{I}]_\theta$ $[\text{want } [(\text{to begin}) [(\text{to try}) [(\text{to write}) (\text{a play})]]]]_\rho$
L+H* LH% H*LL%
As an answer to: "And you, what's up with you?"

b. $[(\text{I want})]_\theta$ $[(\text{to begin}) [(\text{to try}) [(\text{to write}) (\text{a play})]]]_\rho$
L+H* LH% H*LL%
As an answer to: "What do you want to do?"

c. $[(\text{I want}) (\text{to begin})]_\theta$ $[(\text{to try}) [(\text{to write}) (\text{a play})]]_\rho$
L+H* LH% H*LL%
As an answer to: "What do you want to begin doing?"

d. $[(\text{I want}) (\text{to begin}) (\text{to try})]_\theta$ $[(\text{to write}) (\text{a play})]_\rho$
L+H* LH% H*LL%
As an answer to: "What do you want to begin to try?"

e. $[(\text{I want}) (\text{to begin}) (\text{to try}) (\text{to write})]_\theta$ $[(\text{a play})]_\rho$
L+H* LH% H*LL%

[7] Other terms used in the partitioning of information include *(back)ground/focus*, and *topic/comment* among others. For the purposes of this section, we assume that all of these correspond to *given/new* information. Steedman (2000b) makes a distinction between *background/focus* and *theme/rheme*. For him, *theme* or *rheme* can be partitioned into *background* and *focus*. In this account, the DTE can be thought of Steedman's *focus* and whatever that is not a DTE can be considered as *background*. For a survey of literature on information packaging, see Vallduví and Engdahl (1996).

[8] L+H* is called the "scooped accent," a low tone target on the accented syllable which is immediately followed by a relatively sharp rise to a peak in the upper part of the speaker's pitch range. LH% is a low phrase accent closing the last intermediate phrase, followed by a high boundary tone. H* is a high pitch accent, and LL% is a low phrase accent ending its final intermediate phrase and a low boundary tone falling to a point low in the speaker's range. For more information on this notation, see "The ToBI Annotation Conventions" by Mary Beckman and Julia Hirschberg available online at `www.ling.ohio-state.edu/~tobi/ame_tobi/annotation_conventions.html`.

As an answer to: "What do you want to begin to try to write?"

f. [(I want) [(to begin) [(to try) [(to write) (a play)]]]]
Unmarked with respect to information structure

In (6.36a–e), each sentence is marked with respect to its information structure; whereas (6.36f) is unmarked. Assuming that the correlation between information structure and intonation holds and ignoring the possibility of foregrounding items other than the last in an intonational phrase, we conclude that in (6.36a–e) the last prosodic word (i.e., the default DTE) in theme bears a L+H* LH% (rise-fall-rise) intonation and the last prosodic word in rheme bears a H*LL% (fall) intonation (This, of course, is specific to English).

6.2.3.2 The Type Hierarchy and Constraints

Klein's model does not have provisions for relating the information status of the constituents in the sentence to prosody. It is clear, however, that in order for it to be able to return the correct intonational phrasing, such a correspondence is necessary. We need to make sure that themes and rhemes (when marked) bear the right intonation and do not occupy the same intonation phrase. Sensitivity to contextual information by the prosodic component entails modification to the feature appropriateness conditions in the prosodic type hierarchy as well as new constraints. Pollard and Sag (1994) assume a CONTEXT feature for SIGN|SYNSEM|LOCAL. It seems only natural to place information structure in context. However as Engdahl and Vallduví (1994) propose, placing information structure in *local* objects is problematic for a trace-based account of unbounded dependencies. It is exactly for this reason that De Kuthy (2002), in her theory of information structure, assumes that information structure is a feature appropriate to *sign* on a par with PHON, and SYNSEM. This is another step towards a tripartite architecture of grammar and we are will adopt it in this work as well. But unlike De Kuthy, we do not assume that the scope of information status is represented as a symbolic language with a model-theoretic interpretation. There are two reasons for this: Firstly, taking De Kuthy's approach requires adherence to one particular semantic theory. In this work, we would like to remain theory-neutral as much as possible when it comes to the internal structures of phonology and semantics. Secondly, linking semantics directly to information structure and in turn to phonology adds to the monolithic structure of the theory. In addition to Jackendoff (2002), a considerable body of work suggests that semantics, syntax, and phonology should be allowed to work separately while making sure that they constrain one another. For more information see Penn (1999a,b); Penn and Haji-Abdolhosseini (2003). What is assumed here is that phonology, syntax and information structure all operate as indepen-

dently as possible while working on one common list of domain objects that we assume to be lexical items here for convenience. Thus, *sign* will have (at least) the feature appropriateness constraint presented in (6.37) defined over it. In this ap-

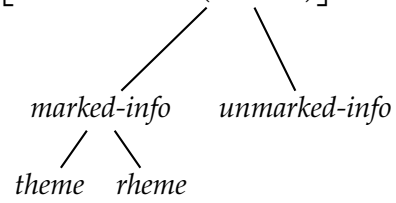
(6.37) **Appropriateness Constraint on *sign***

$$\begin{bmatrix} \textit{sign} & \\ \text{PHON} & \textit{pros} \\ \text{SYNSEM} & \textit{synsem} \\ \text{DOM} & \textit{list}\big(\textit{dom-obj}\big) \\ \text{INFO} & \textit{list}\big(\textit{info}\big) \end{bmatrix}$$

proach, phonology, syntax/semantics and information structure operate independently on a shared list of domain objects (DOM). Here I am treating this list simply as a list of lexical items, while keeping open the question of the actual nature of that list. It is possible that the value of what I have called the DOM feature turns out to best be modelled using a recursive structure for linear order of items such as phenogrammatical structure (see Penn and Haji-Abdolhosseini, 2003). Type *info* has two subtypes: *marked-info* and *unmarked-info*. The type *marked-info* itself subsumes *theme* and *rheme*. In the prosody partition, we need a place to record the tonal information. Therefore, we add the feature TONE to *mtr(τ)*. The feature TONE takes as its value a list of *tone* objects, which have the following subtypes: *marked-tone* and *unmarked-tone*. The type *marked-tone* (at least) subsumes *rfr*, which stands for rise-fall-rise (L+H* LH%) intonation, and *fall*, which stands for falling (H*LL%) intonation (see (6.39)). Our revised prosodic type hierarchy takes the form shown in Figure 6.5 on the next page.

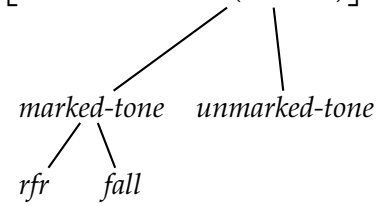
(6.38) **Informational Types:**

$$\begin{bmatrix} \textit{info} & \\ \text{INF-DOM} & \textit{list}\big(\textit{dom-obj}\big) \end{bmatrix}$$



(6.39) **Tonal Types:**

$$\begin{bmatrix} \textit{tone} & \\ \text{TONE-DOM} & \textit{list}\big(\textit{dom-obj}\big) \end{bmatrix}$$



Another point to discuss here is Klein's type hierarchy of phrases that cross-classify prosodic phrases under syntactic phrases. What that hierarchy assumes is

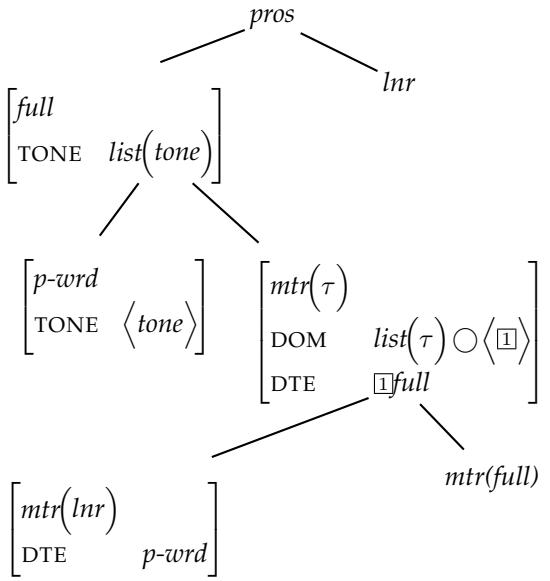


Figure 6.5: Prosodic Type Hierarchy

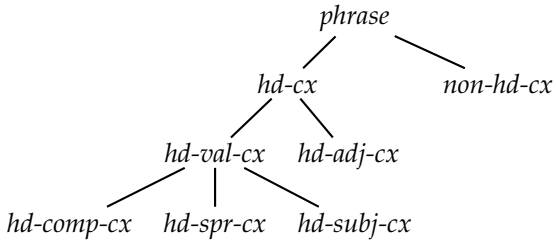


Figure 6.6: Type hierarchy of phrasal constructions

that all head-complement and head-specifier phrases match some prosodic phrase in their yield unless some non-trivial mapping is also assumed. While this is a logical starting point since syntactic trees and prosodic trees often look very similar, even isomorphic in some cases, they clearly are not the same as we observe in the data above and in the literature. Sometimes prosodic phrases do not correspond to any syntactic constituent and vice versa. In our move towards a tripartite architecture, we should therefore treat these two types of constituency differently. Klein's approach is heavily syntax-driven and involves making prosodic trees by manipulating syntactic trees. What we need to do instead is to modify mkMtr such that it declaratively defines prosodic trees without referring to full syntactic trees. This will also simplify mkMtr as we shall see shortly. What this means for the type hierarchy of *phrase* types is that phrases are no longer cross-classified with respect to the dimensions of headedness and prosody. Prosodic structure is defined over the list of domain objects as opposed to a list of partial prosodic structures. Figure 6.6 on the facing page presents the type hierarchy of phrases that we assume in this paper.

A constraint is now required to associate the tones introduced in (6.39) with the information that they convey. This constraint has to be declared for any object of type *word*. This can be regarded as an interface point between conceptual structure and phonological structure in Jackendoff's terms. The constraint, which is called the *Information-Tone Association Constraint (ITAC)*, is formulated in Figure 6.7 on the next page. The first disjunct in Figure 6.7 relates theme with the *rise-fall-rise* (L+H* LH%) intonation. The second disjunct relates rheme with *falling* (H*LL%) intonation, and the third one is the default situation where lexical items are left unmarked with regard to their information status and tone. The last disjunct states that some *word* objects are prosodically leaners.

6.2.3.3 The mkMtr Function Revisited

We now need to revise the mkMtr function to handle the new formalism. Before we do that, however, let us go over the type of change that needs to be made. Take the examples in (6.40).

(6.40) a. [Jane [drank milk]]
b. [[Jane drank] milk]

In (6.40a), *Jane* is the theme and *drank milk* the rheme; whereas, in (6.40b), *Jane drank* is the theme and *milk* the rheme. (6.40a) is compatible with the Prosodic Isomorphism Hypothesis (PIH) but (6.40b) is not. *Jane* and *drank* form their own prosodic constituent because they both correspond to the theme of the sentence and *milk* belongs to a different prosodic constituent because its informational status

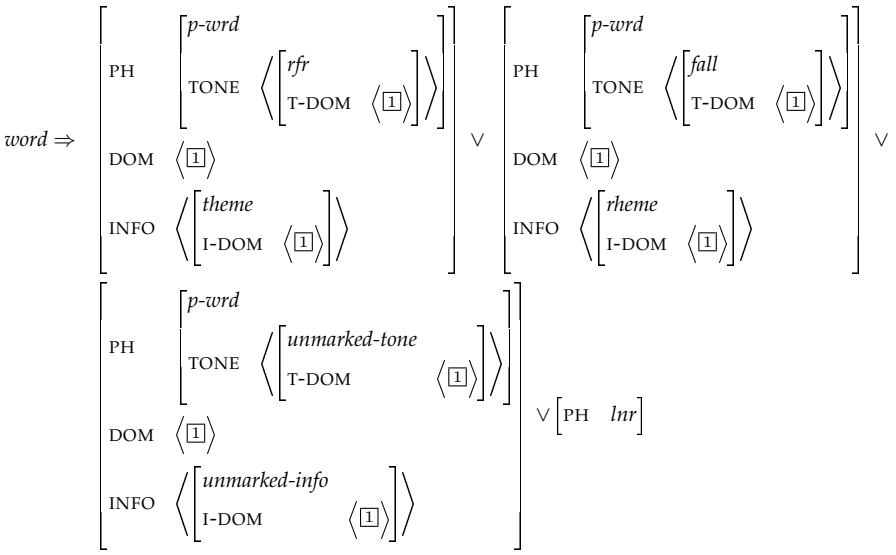


Figure 6.7: Information-Tone Association Constraint (ITAC)

is different. Therefore, what we want mkMtr to do is to relate prosodic structure and information structure. What this amounts to theoretically is that a weak form of PIH in this model holds for prosody and information structure as opposed to syntactic structure.

(6.41) **The** mkMtr **Function (Revised)**

a. $\mathsf{mkMtr} : list(pros) \mapsto mtr(pros)$
$\mathsf{mkMtr}(\boxed{1}) = \mathsf{mkMtr}^{full}(mkAllLnrs(\boxed{1}))$

A metrical tree consists of prosodic objects of type *full* some of which may be leaner groups.

b. $\mathsf{mkMtr}^{\tau \lessdot pros} : list(pros) \mapsto mtr(\tau)$
$\mathsf{mkMtr}^{\tau}\left(\left\langle\left[\text{PHON} \quad \boxed{1}pros\right]\right\rangle\right) = \boxed{1}$

E.g. $\mathsf{mkMtr}^{full}(\langle Jane \rangle) = Jane$

c. $\mathsf{mkMtr}^{lnr} : list(pros) \mapsto mtr(pros)$

$$\mathsf{mkMtr}^{lnr}\left(\left\langle \boxed{2}\mathit{lnr},\ldots,\boxed{n}\mathit{lnr},\boxed{m}\begin{bmatrix}\textit{p-wrd} & \\ \text{TONE} & \boxed{3}\end{bmatrix}\right\rangle\right)=\begin{bmatrix}\mathit{mtr}\left(\mathit{lnr}\right) \\ \text{DOM}\left\langle \boxed{2},\ldots,\boxed{n},\boxed{m}\right\rangle \\ \text{DTE}\ \boxed{m} \\ \text{TONE}\left\langle \boxed{3}\right\rangle\end{bmatrix}$$

A leaner group consists of one or more leaners followed by a prosodic word that is also the DTE.

E.g. $\mathsf{mkMtr}^{lnr}(\langle to, the, store\rangle) =$ *(tŏ thĕ stóre)*

d. $\mathsf{mkMtr}^{full} : list(pros) \mapsto mtr(full)$

i. $\mathsf{mkMtr}^{full}\left(\left\langle \boxed{1}\left[\text{TONE}\left\langle \boxed{3}\right\rangle\right],\boxed{2}\left[\text{TONE}\left\langle \boxed{3}\right\rangle\right],\ldots,\boxed{n}\left[\text{TONE}\left\langle \boxed{3}\right\rangle\right]\right\rangle\right)=$

$$\begin{bmatrix}\mathit{mtr}\left(full\right) & \\ \text{DOM} & \left\langle \boxed{1},\boxed{4}\mathsf{mkMtr}^{full}\left(\boxed{2},\ldots,\boxed{n}\right)\right\rangle \\ \text{DTE} & \boxed{4} \\ \text{TONE} & \left\langle \boxed{3}\right\rangle\end{bmatrix}$$

A list of prosodic objects with coindexed TONE values make up a full (right-branching) prosodic tree.

E.g. $\mathsf{mkMtr}^{full}(\langle \frac{drank}{\boxed{1}fall}, \frac{milk}{\boxed{1}}\rangle) =$ [*drank milk*]

ii. $\mathsf{mkMtr}^{full}(\boxed{1}\oplus\boxed{2}\oplus\cdots\oplus\boxed{n}) = \begin{bmatrix}\mathit{mtr}\left(full\right) & \\ \text{DOM} & \left\langle \boxed{5},\boxed{7}\mathsf{mkMtr}^{full}\left(\boxed{6},\ldots,\boxed{o}\right)\right\rangle \\ \text{DTE} & \boxed{7} \\ \text{TONE} & \left\langle \boxed{3},\boxed{4},\ldots,\boxed{m}\right\rangle\end{bmatrix}\wedge$

$$\boxed{1}=\left\langle\left[\text{TONE}\quad\boxed{3}\right],\ldots,\left[\text{TONE}\quad\boxed{3}\right]\right\rangle\wedge$$
$$\boxed{2}=\left\langle\left[\text{TONE}\quad\boxed{4}\right],\ldots,\left[\text{TONE}\quad\boxed{4}\right]\right\rangle\wedge\cdots\wedge$$
$$\boxed{n}=\left\langle\left[\text{TONE}\quad\boxed{m}\right],\ldots,\left[\text{TONE}\quad\boxed{m}\right]\right\rangle\wedge$$
$$\boxed{3}\neq\boxed{4}\neq\cdots\neq\boxed{m}\wedge$$
$$\mathsf{mkMtr}^{full}(\boxed{1})=\boxed{5}\wedge\mathsf{mkMtr}^{full}(\boxed{2})=\boxed{6}\wedge\cdots\wedge\mathsf{mkMtr}^{full}(\boxed{n})=\boxed{o}$$

A full metrical tree consists of smaller full metrical trees each of which belongs to a different intonation phrase.

E.g. $mkmtr^{full}(\langle \overset{Jane}{\boxed{1}}, \overset{drank}{\boxed{1}_{rfr}}, \overset{milk}{\boxed{2}_{fall}} \rangle)$ = [[*Jane drank*]*milk*]

This version of mkMtr function makes use of another function called *mkAllLnrs* to form the leaner groups based on which larger prosodic structures are formed. These larger prosodic structures are created as they were in Klein's model with the exception that only items that belong to the same information package (i.e., theme, rheme or unmarked) are placed in the same prosodic phrase. The revised mkMtr function is used in a constraint on *sign* objects as formalized in (6.43). The function *collect-phon* that is defined below in (6.42) and used in (6.43) takes a list of domain objects and returns a list of the PHON values of those objects. Theoretically, relations like *collect-phon* not only ensure the correct input type to other relations or modules of the grammar, they are also ideal in restricting access. In this case, *collect-phon* allows phonology to see only the phonological data inside DOM. Except for the interface constraints (such as ITAC, and ISPC introduced in Figure 6.8 on the facing page), nothing from phonology can access the data in the syntactic/semantic, or information-structural modules.

We no longer make use of *base-pr* and *ext-pr*; rather, we let what has been described as prosodic flattening and prosodic promotion follow naturally from general constraints on prosody and information structure.

(6.42) *collect-phon*: *list(dom-obj)* $\mapsto$ *list(pros)*

a. *collect-phon*($\langle\rangle$)= $\langle\rangle$

b. *collect-phon*($\langle \boxed{1} \mid \boxed{2} \rangle$) = $\langle [\text{PHON } \boxed{1}\,] \mid$*collect-phon*($\boxed{2}$)$\rangle$

(6.43) $sign \Rightarrow \begin{bmatrix} \text{PHON} & \mathsf{mkMtr}\Big(\textit{collect-phon}\Big(\boxed{1}\Big)\Big) \\ \text{DOM} & \boxed{1} \end{bmatrix}$

(6.44) $mkAllLnrs : list(pros) \mapsto list(pros)$

a. $mkAllLnrs(\boxed{1} \oplus \boxed{2} \oplus \boxed{3}) = mkAllLnrs(\boxed{1} \oplus \langle \mathsf{mkMtr}^{lnr}(\boxed{2}) \rangle \oplus \boxed{3})$

b. $mkAllLnrs(\boxed{1}) = \boxed{1}$

(6.41a) is the top-level function called by *sign* objects. It uses the *mkAllLnrs* function defined in (6.44) to generate all the possible leaner groups in the list of domain objects, and passes the resulting mixed list of leaner groups and prosodic words to mkMtr^{full} to generate a complete prosodic structure for the original list of domain objects.

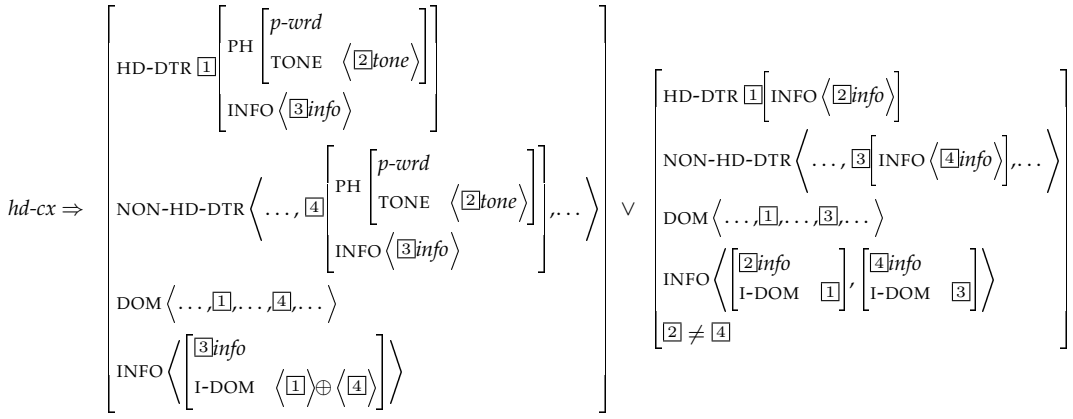


Figure 6.8: Information Status Projection Constraint (ISPC)

(6.41b) is essentially the same as before. It simply returns a singleton argument intact because a metrical tree requires at least two daughters. (6.41c), similar to the original formulation of mkMtr, defines metrical trees as consisting of a group of leaners attached to a final prosodic word with the latter being the DTE. The leaner group has the value of its TONE feature structure-shared with that of the prosodic word of the leaner group. (6.41d-i) is the first of the two definitions for mkMtr^{full}. It requires that all the members of its argument list share the same tone value, which means they should all belong to the same intonational phrase (IP). In that case, it makes a metrical tree in the usual manner and structure-shares its tone value with that of the daughters. (6.41d-ii) places metrical objects in the same prosodic constituent just in case those objects bear the same tone specification (i.e., the value of their TONE feature is structure-shared). Then it makes a metrical tree out of the result with the remainder of the list of prosodic objects passed to it. Notice that mkMtr_{LA} has been omitted because we are no longer making prosodic structures based on syntactic ones.

6.2.3.4 Scope of *Theme*/*Rheme* Status

The issue of the scope of *theme* and *rheme*, also known as "the projection problem" is approached in this subsection (see also Lambrecht and Michaelis, 1998). We define this concept in the form of the *Information Status Projection Constraint (ISPC)* as a type constraint on *hd-cx*. ISPC is formalised in Figure 6.8.

According to ISPC the arguments of the head daughter in a headed construction by default inherit the information status of that predicate through structure sharing (this is specified in the first disjunct in Figure 6.8). When an argument is overtly marked for information structure, it will not inherit the information status

(and tone) of the head (this is specified in the second disjunct in Figure 6.8). Thus in (6.36c), repeated here as (6.45), for example, *begin* inherits theme status from *want*, and *write* and *play* inherit *rheme* from *try*.

(6.45) [(I want) (to begin)]$_\theta$ [(to try) [(to write) (a play)]]$_\rho$
L+H* LH% H*LL%

Multiple theme and rheme markings are also possible and they can be distinguished by the fact that multiple themes/rhemes are listed separately in the INFO feature. We do not consider the projection problem in non-headed constructions in this work. Since we assume that the rule schemata allow for the union of the domain objects of their daughters as well as the lists of informational objects, we always have access to the information status of any given prosodic word.

6.2.3.5 Accounting for the Data

Let us now go over the derivation of the examples in (6.40). These derivations are straightforward. In the following two derivations, we use the AVM notation for better exposition. Subsequent examples are represented in Klein's more succinct notation.

Figure 6.9 on the facing page shows the derivation of (6.40a) in terms of its syntactic and information structures. Initially, *milk* is not marked for information status. It inherits the *rheme* status because of ISPC due to being an argument of the verb. This is shown in the VP construction. The subject does not fall under the scope of *rheme* because it is already marked as *theme*. The application of the ITAC throughout the derivation provides the list of domain objects shown in (6.46) for the resulting S construction.

(6.46)
$$\left[\text{DOM}\ \left\langle \boxed{1}\left[\text{PH}\ \left[\begin{array}{ll}\textit{Jane} & \\ \text{TONE} & \left\langle\left[\boxed{4}\textit{rfr}\right]\right\rangle\end{array}\right]\right], \boxed{2}\left[\text{PH}\ \left[\begin{array}{ll}\textit{drank} & \\ \text{TONE} & \left\langle\left[\boxed{5}\textit{fall}\right]\right\rangle\end{array}\right]\right], \boxed{3}\left[\text{PH}\ \left[\begin{array}{ll}\textit{milk} & \\ \text{TONE} & \left\langle\left[\boxed{5}\textit{fall}\right]\right\rangle\end{array}\right]\right]\right\rangle\right]$$

The application of mkMtr to the list of domain objects shown in (6.46) is represented in (6.47). The second example, (6.40b) is derived analogously.

(6.47)
$$\textsf{mkMtr}\left(\left\langle \boxed{1}\left[\begin{array}{ll}\textit{Jane} & \\ \text{TONE} & \left\langle\left[\boxed{4}\textit{rfr}\right]\right\rangle\end{array}\right], \boxed{2}\left[\begin{array}{ll}\textit{drank} & \\ \text{TONE} & \left\langle\left[\boxed{5}\textit{fall}\right]\right\rangle\end{array}\right], \boxed{3}\left[\begin{array}{ll}\textit{milk} & \\ \text{TONE} & \left\langle\left[\boxed{5}\right]\right\rangle\end{array}\right]\right\rangle\right) =$$

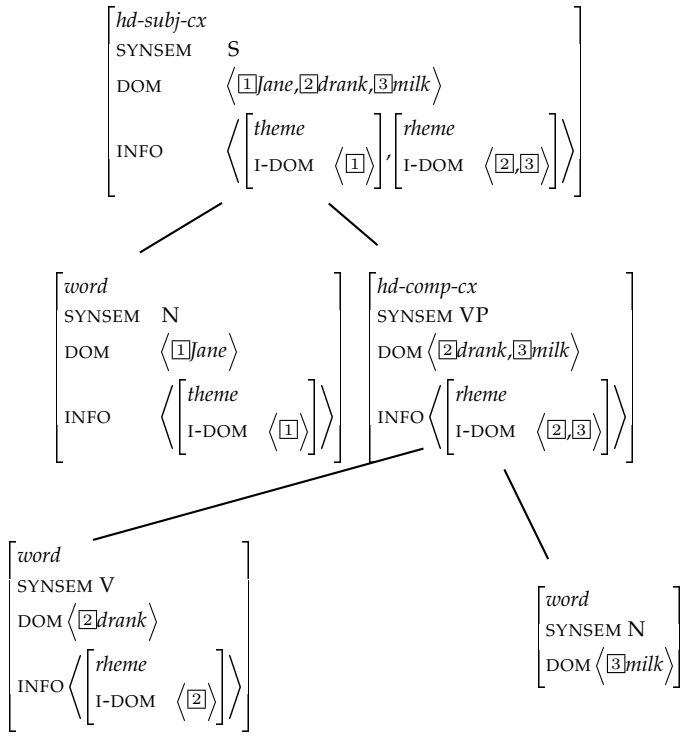


Figure 6.9: Syntactic/information-structural derivation of (6.40a)

$$\mathsf{mkMtr}^{full}\Big(mkAllLnrs\Big(\langle \boxed{1},\boxed{2},\boxed{3}\rangle\Big)\Big)= \mathsf{mkMtr}^{full}\Big(\langle \boxed{1},\boxed{2},\boxed{3}\rangle\Big)=$$

$$\mathsf{mkMtr}^{full}\Big(\Big\langle \mathsf{mkMtr}^{full}\Big(\langle \boxed{1}\rangle\Big),\mathsf{mkMtr}^{full}\Big(\langle \boxed{2},\boxed{3}\rangle\Big)\Big\rangle\Big)=$$

$$\begin{bmatrix} mtr\Big(full\Big) & \\ \text{DOM} & \Big\langle \boxed{1},\boxed{4}\begin{bmatrix} mtr\Big(full\Big) & \\ \text{DOM} & \langle \boxed{2},\boxed{3}milk\rangle \\ \text{DTE } \boxed{3} & \end{bmatrix}\Big\rangle \\ \text{DTE} & \boxed{4} \end{bmatrix}$$

We can again consider the play-writing examples, which are shown in (6.48). Let us assume that these sentences roughly correspond to the semantic specifications represented in Figure 6.10 on the next page. In fact, we present the semantic specifications that correspond to (6.48b). The difference between Figure 6.10 on the facing page and the semantic specifications of (6.48a, c, d) is merely in the scope of theme/rheme (see below). (6.48e) is not marked for theme/rheme and gets the default prosodic constituency. (6.48b), therefore, receives the prosodic structure shown in (6.49). The cases of (6.48a,c, d) are similar.

(6.48)
a. [[(I want)]$_\theta$[(to begin) [(to try) [(to write) (a play)]]]]$_\rho$.
b. [[(I want) (to begin)]$_\theta$[(to try) [(to write) (a play)]]]$_\rho$.
c. [[(I want) [(to begin) (to try)]]$_\theta$[(to write) (a play)]]$_\rho$.
d. [[(I want) [(to begin) [(to try) (to write)]]$_\theta$[(a play)]]$_\rho$.
e. [(I want) [(to begin) [(to try) [(to write) (a play)]]]].

(6.49)
$$\mathsf{mkMtr}\Big(\boxed{1}\langle I, want, to, begin, to, try, to, write, a, play\rangle\Big)=$$

$$\mathsf{mkMtr}^{full}\Big(mkAllLnrs\Big(\boxed{1}\Big)\Big)=$$

$$\mathsf{mkMtr}^{full}\Big(\Big\langle\Big(I\ want\Big),\Big(to\ begin\Big),\Big(to\ try\Big),\Big(to\ write\Big),\Big(a\ play\Big)\Big\rangle\Big)=$$

$$\Big[\Big[\Big(I\ want\Big)\Big(to\ begin\Big)\Big]^{rfr}\Big[\Big(to\ try\Big)\Big[\Big(to\ write\Big)\Big(a\ play\Big)\Big]\Big]^{fall}\Big]$$

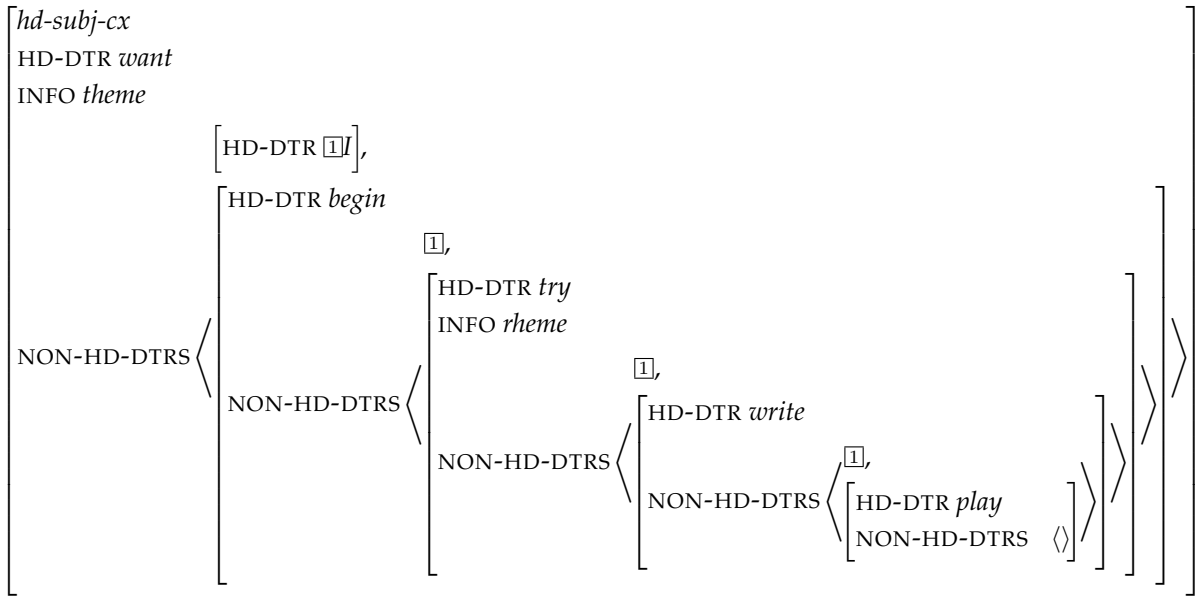


Figure 6.10: Basic semantics and information structure of (6.48b)

(6.50) $\Big[\Big(I\ want\Big)\Big[\Big(to\ begin\Big)\Big[\Big(to\ try\Big)\Big[\Big(to\ write\Big)\Big(a\ play\Big)\Big]\Big]\Big]\Big]$

Notice that because the lexical items are unmarked in (6.48e) with respect to their information status, the prosodic structure that emerges is flat as shown in (6.50). Here we see that what is generally known as prosodic flattening follows naturally from this account and no special theoretical devices are required to derive that structure from a highly structured syntactic tree.

The case of (6.36a) is somewhat different from the others. In this example, the pronoun *I*, a leaner, forms its own prosodic phrase bearing the L+H* LH% intonation that corresponds to theme. According to our model, however, the feature TONE is not appropriate to *lnr* because leaners by definition need a prosodic word to attach to. This can be solved by introducing a lexical rule that type-shifts leaners when their INFO feature is marked. This is formulated as (6.51) below.

(6.51) ***lnr* Type-Shifting Rule**

$$\begin{bmatrix} \text{PHON} & \text{lnr} \\ \text{INFO} & \textit{marked-info} \end{bmatrix} \Rightarrow \begin{bmatrix} \text{PHON} & \text{p-wrd} \end{bmatrix}$$

Let us now discuss example (6.34) repeated below as (6.52).

(6.52) *[Three mathematicians] [in ten prefer margarine]

In Klein's model, this constituency simply does not arise because of PIH. In this model, we do not get the unacceptable constituency in (6.52) either because the informational status of one argument does not affect the other(s); i.e. if *prefer* is marked as theme and *margarine* as rheme, we still get the correct prosodic structure because the subject, *three mathematicians in ten*, inherits the theme status from *prefer*. However, one can think of a very implausible case that could give rise to (6.52) in our information-based analysis, and that is when *mathematicians* alone is marked as theme and *in ten* and *prefer* are marked as multiple rhemes. This information structure may not be felicitous in any context, but if it ever is, (6.52) will still be unacceptable because two different rhemes in (6.52) occur in the same IP. The correct prosodic structure that complies with the new definition of mkMtr is (6.53).

(6.53) [[Three mathematicians]$_\theta$ (in ten)$_\rho$ [prefer margarine]$_\rho$]

The above example brings us to our next set of data presented earlier in (6.35), and repeated below as (6.54).

(6.54)
a. [Jane [gave [(the book) (to Mary)]]]
b. [[Jane] [gave [(the book) (to Mary)]]]
c. [Jane [gave (the book)] [(to Mary)]]
d. [[Jane gave] [(the book)] [(to Mary)]]
e. * [[Jane] [gave] [(the book) (to Mary)]]
f. * [[Jane gave] [(the book) (to Mary)]]
g. [[Jane] [gave (the book)] [(to Mary)]]
h. [[Jane] [gave] [(the book)] [(to Mary)]]

According to our analysis, (6.54a) is considered the unmarked case. In (6.54b), *Jane* has been marked as theme and *gave* as rheme, which passes down this status to its arguments *book* and *Mary*. In (6.54c), *gave* has been marked as theme and *Mary* as rheme. As mentioned earlier, Selkirk (1984) attributes the ungrammaticality of (6.54e, f) to the violation of the Sense Unit Condition since *the book* and *to Mary* do not form a sense unit. We achieve the same effect in this approach by ISPC and assuming that no more than one information unit (i.e., theme/rheme) can be present in one IP; this follows from the formulation of ISPC and ITAC. In other words, each intonation phrase corresponds to only one information unit. This is in line with our version of PIH. Such an analysis entails that in (6.54d, g, h), there are multiple themes or rhemes and those multiple themes or rhemes are reflected

as separate IPs in phonology. (6.54e, f) are ungrammatical because *the book* and *to Mary* have different informational markings, i.e., theme/rheme, $\text{rheme}_1/\text{rheme}_2$ or the like. This condition also prevents (6.52) because the only way that *in ten* can be separated from *three mathematicians* is to have a different informational marking, which by ISPC could not be structure-shared with the informational marking of *prefer margarine*. Not only does ISPC ensure that each information unit reflects the right intonation in phonology; together with the mkMtr function, it also provides an implementation of Selkirk's (1984) *Sense Unit Condition* without resorting to another level of representation and unnecessary complication of the theory.

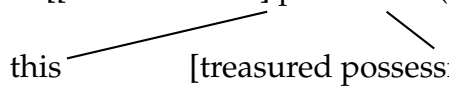
As an example, let us look at the sentences in (6.54) again. (6.54d, g, h) have multiple themes or rhemes. The indexed *info* and its corresponding *tone* value ensure that multiple themes or rhemes are not mistakenly grouped together. (6.54c) receives the following prosodic and information structure if we assume that *give* and *book* are marked as multiple themes.

(6.55) $[[\text{Jane gave}]^{rfr1}_{\theta 1}$ $(\text{the book})^{rfr2}_{\theta 2}$ $(\text{to Mary})^{foll1}_{\rho 1}]$

Examples (6.54e, f) are automatically rejected because the two arguments of *give* are sisters of one another; therefore, they cannot bear the same information status by ISPC, and thus, cannot be in the same IP.

Another interesting consequence of the information-based account of prosody in a tripartite grammar architecture is the fact that an ill-formed prosodic structure like (6.56) never arises because of the way mkMtr has been defined and this relieves us from positing Klein's *Lexical Head Association Constraint*, which according to him is a partial implementation of Selkirk's end-based mapping.

(6.56) * [[this treasured] possession (of the samurai)]



The derivation of the prosodic structure of *this treasured possession of the samurai* is shown below:

(6.57) mkMtr(⟨this, treasured, possession, of, the, samurai⟩ =
mkMtr^{full}(*mkAllLnrs*(⟨this, treasured, possession, of, the, samurai⟩)) =
mkMtr^{full}(⟨this, treasured, possession, (of the samurai)⟩) =
[this treasured possession (of the samurai)]

6.2.4 Universality of the Claims

The analysis provided above raises the question of universality. In other words, "to what extent do we think that this analysis is universal?" While all the data that

we have been considering come from English, there are parts of this analysis that I claim will hold universally. First is the appropriateness constraint on *sign* presented in (6.37). This is the heart of the analysis claiming that grammar in general should have a parallel modular architecture. The information types presented in (6.38) is also assumed to be universal (albeit a simplification). The prosodic type hierarchy in Figure 6.5 is also assumed to be universal with the exception of one detail. The DTE in objects of type $mtr(\tau)$ is assumed to be the last element in the list of domain objects; this is a language-specific fact. The hierarchy of phrasal constructions in Figure 6.6 and the tonal types in (6.39) have been based on English. The idea of the Information-Tone Association Constraint (ITAC) is universal, but the details of associating theme with L+H* LH% and rheme with H*LL% intonation is language-specific. And so is mkMtr. The basic idea of mkMtr presented in (6.41) is assumed to be universal but the details of it (e.g., characterizing prosodic structures as right-branching trees) are language-specific. We also assume the Information Status Projection Constraint (ISPC) presented in Figure 6.8 to be universal.

6.3 Discussion

The study presented in this chapter starts off with Klein's (2000) analysis of prosodic constituency in HPSG and extends it to account for some prosodic variation phenomena that are dependent upon the information structure of the sentence. Because a constraint-based approach to prosodic phenomena is employed here, we can capture some interesting linguistic generalizations without recourse to *ad hoc* operational rules. In addition, the modular design of the theory allows for better readability and maintainability—a welcome outcome of the architecture of the theory for grammar engineering. The departure from a syntactocentric theory towards a tripartite one in terms of Jackendoff (2002) proves to be a promising approach as it captures in simpler terms many phenomena previously discussed in the literature.

However, many issues have yet to be resolved. Although Haji-Abdolhosseini's (2003a) account can be extended to account for prosodic- and/or discourse-driven word order variations such as heavy-NP shift, clefting, and pseudo-clefting, because of its use of crisp constraints, it is still very rigid in what it generates. In addition, in cases where there is more than one option available for producing a grammatical sentence (e.g., the choice of using or not using a cleft construction), this grammar, like any other non-probabilistic grammar, simply provides non-determinism as a solution. And this is because it lacks a mechanism for showing preferences, and degrees of (non-)compliance with certain constraints.

Gradable constraints are easy to find in language if one knows where to look. Take the sentences in (6.58) and (6.59) for example. The sentences in (6.58) are all in canonical word order, and the ones in (6.59) have undergone "heavy-NP shift", so to speak.[9] It is clear that in (6.58) the sentences become increasingly marked as the object of the sentence gets prosodically heavier; whereas, in (6.59) the situation is reversed. The sentences become less marked as the object NP gets heavier. No present theory is actually able to handle cases like this where constraints gain or lose strength on a case by case basis. In this case, for example, a prosodic constraint (put heavy NPs last) overrides a syntactic constraint (NPs appear in the order of obliqueness) when the object NP gets heavy enough. When the NP is too light, the prosodic constraint cannot apply. When the NP is rather heavy, the constraint may or may not apply. But when the NP is very heavy, the constraint has to apply. For a review of such phenomena, see Arnold et al. (2000) and Wasow (2002).

(6.58) a. He wanted to demonstrate it to us.
b. He wanted to demonstrate that life to us.
c. He wanted to demonstrate the consequences to us.
d. ? He wanted to demonstrate the consequences of such an unholy life to us.
e. ?? He wanted to demonstrate the consequences of such a horribly filthy and unholy life to us.

(6.59) a. * He wanted to demonstrate to us it.
b. ?? He wanted to demonstrate to us that life.
c. He wanted to demonstrate to us the consequences of such an unholy life.

Another example of the effect of graded constraints can be observed at the syntax-discourse interface. Birner (1992, 1994) argues that givenness and newness of information in determining word order is not absolute. Information that has been presented more recently in discourse is considered newer than the information that has been presented earlier. The claim that givenness is a gradient notion has also been made by Ariel (1990), Arnold (1998), Arnold et al. (2000), Chafe (1976), and Givón (1983) among others. Therefore it is expected that, when speaking of two pieces of information (X and Y) that have both been previously mentioned, one tends to evaluate the one that has been mentioned more recently as given (X) and the one that has been mentioned earlier as new (Y), which means the sentence will tend to have an XY order corresponding to the given-new order

[9] We use this terminology for convenience only. In a constraint-based framework, a structure is not defined in terms of derivational processes such as movement.

of information. However, if the two pieces of information have both been mentioned at roughly the same time in discourse, then the order of the information is not that strongly set. Here we see that the degree of the givenness or newness of the information influences the choice of word order in syntax in varying degrees.

6.4 Summary

Through a detailed analysis of the issue of prosodic versus syntactic constituency and the correspondence between information structure and prosody in English, this chapter provided evidence for the advantages of a parallel modular architecture for grammar. We also argued that the resolution of conflicting requirements from different modules requires the use of soft constraints. The following chapter provides the formal definition of soft linguistic constraints based on the c-semiring-based theory of soft constraint satisfaction introduced in Chapter 4. We will then show how a c-semiring-based approach to type-antecedent constraints will allow us to handle the data discussed in the previous section as well as provide us with the necessary theoretical machinery to account for graded grammaticality and ganging up effects in a constraint-based theory of language such as HPSG.

Chapter 7

Soft Intermodular Constraints

7.1 Introduction

This chapter outlines a theory of soft intermodular constraint satisfaction based on the SCSP framework (the c-semiring-based theory of Soft Constraint Satisfaction Problems). Given the theoretical apparatus discussed in Chapter 4, we are now in a position to implement the type of graded constraints that we discussed in chapters 5 and 6. Section 7.2 casts Keller's Linear Optimality Theory into the SCSP framework. It shows how LOT can be thought of as an instance of SCSP once some incompatibilities between LOT and SCSP have been resolved. Section 7.3 provides a formal definition of an SCSP-based grammar and goes through some illustrative examples. Section 7.4 briefly talks about how a c-semiring-based approach might be incorporated into a unification-based theory of grammar.

An interesting philosophical outcome of the approach presented in this chapter is that linguistic constraint satisfaction is shown to be an instance of general human constraint satisfaction (as it is based on the general theory of soft constraint satisfaction), which situates linguistics along side other human cognitive faculties. What makes language special is not its constraint satisfaction mechanism, but its constraints.

7.2 Linear Optimality Theory as SCSP

This section demonstrates how Linear Optimality Theory (Keller, 2000) can be viewed as an instance of the SCSP framework. Before that, however, we need to clarify some discrepancies between the way constraint satisfaction is viewed in the OT literature and the way it is viewed in the AI literature. There are some overlaps

between the two views; yet, the definitions are not totally interchangeable. What we will do in this section is to paint a coherent picture that brings the two views together.

7.2.1 Valuation vs. Violation Profile

Candidate linguistic structures in OT are evaluated directly against a set of constraints, each of which returns a violation profile for that structure. In the constraint satisfaction literature, however, the problem space must first be formally defined and then it may be mapped into some other space of representations by an embedding function such as a vector space (see definition 7.2.6 on page 105). This (other) representation is then evaluated with respect to the constraints imposed on values inside that representation. The embedding takes place when we want to cast a complicated problem into a form that is easier to solve. We shall call the result of this method of evaluating this easier form against constraints a *valuation* (see definition 7.2.10 on page 106). As Figure 7.1 shows, in the constraint satisfaction framework, an instance of the problem (here a candidate structure) $can \in Can$ is mapped into a vector of features D^k (where k is the dimensionality of D^k) by the embedding function e, then the valuation function $\bar{c}_i$ (where $1 \leq i \leq m$ for some $m \leq k$) returns a value in C_i, which represents the degree of compliance of can with the ith constraint being considered. In LOT, however, a candidate structure can is directly assigned a value in C_i, namely the violation profile for the constraint being considered, by the function represented as c_i in the figure. This is shown with the dashed arrow in the figure. Therefore, evaluation in LOT can be thought of as the composition of $\bar{c}_i$ and e; that is, $c_i = \bar{c}_i \circ e$.

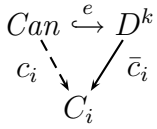


Figure 7.1: Valuation vs. violation profile

7.2.2 Harmony vs. Global Valuation

As mentioned in section 2.5, the harmony of a candidate structure is calculated as in (2.9), repeated below as (7.1) (the variable names have been changed to conform with the ones used in this chapter).

$$H(can) = -\sum_i w(c_i)v(can, c_i) \quad (7.1)$$

can is a candidate linguistic structure; $c_i \in$ C is an OT constraint; $w(c_i)$ is the weight of the constraint c_i; and $v(can, c_i)$ is a function that returns the number of violations of constraint c_i by the candidate structure can, its violation profile.

The harmony of a candidate structure in LOT equals the negated weighted sum of its violation profile for each constraint. The optimal structure is the one with the maximum harmony (i.e., closest to zero). The candidate structures with lower values for their violation profiles (i.e., larger negative numbers) are considered worse than those with values closer to zero. Keller introduced the negation in the harmony calculation in (7.1) to stay close to the conventions in standard OT. In order to make things a little simpler here, we will eliminate this negation. Let us define $\mathcal{V}(can)$ as the negation of $H(can)$. $\mathcal{V}(can)$ is then the *global valuation* of can with respect to all constraints (see definition 7.2.11 on page 106).

$$\mathcal{V}(can) = -H(can) = \sum_i w(c_i)v(can, c_i) \quad (7.2)$$

Let us now refer to the definitions in section 4.4 repeated below as definitions 7.2.1 and 7.2.8).

DEFINITION 7.2.1 **Constraint System** *A constraint system is defined as a triple $CS = \langle S, D, V\rangle$, where S is a c-semiring, D is a finite set, and V is an ordered set of variables.*

DEFINITION 7.2.2 **Constraint** *Given a constraint system $CS = \langle S, D, V\rangle$, where $S = \langle A, \oplus, \otimes, 0, 1\rangle$, a constraint over CS is a pair $\langle def, con\rangle$, where*

- *$con \subseteq V$ is called the* type *of the constraint;*
- *$def : D^k \rightarrow A$ (where k is the cardinality of con) is called the* value *of the constraint.*

In other words, we can think of an LOT grammar in SCSP terms as a constraint system, $CS = \langle S, D, V\rangle$, where $S = \langle A, \oplus, \otimes, 0, 1\rangle$ is a semiring, V is a set of variables characterizing the candidate. D is a finite set of values that the variables in V can take. Therefore, a constraint over this CS is a tuple $\langle def, con\rangle$ such that $con \subseteq V$ and $def : D^k \rightarrow A$. Values in the carrier set A correspond to the overall compliance of a candidate structure, can, with the whole constraint system.

The function def in SCSP, takes the vector representation D^k of the candidate and maps it to a global valuation, whereas LOT takes individual violation profiles for this purpose. We can reconcile this mismatch by filling in some detail as shown in Figure 7.2, which shows how Can is mapped to A. Embedding applies to $can \in Can$ returning a vector of features, which is passed to $\bar{c}_i$, for $1 \leq i \leq m$ perhaps,

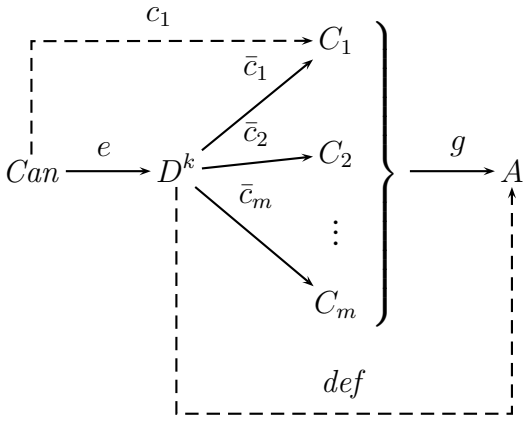


Figure 7.2: Global valuation calculation of candidate structures

which returns a vector of valuations. These valuations are then combined (in this case, weighted and summed up) by the global valuation function g returning a value in A; that is, the global valuation function, $\mathcal{V}$, in LOT and $g \circ e$ in SCSP both return a value in A. The function c_1, for example, is an instance of an LOT-style valuation function, shown here for comparison. The function def is SCSP's def function. As can be seen, the two sets of terminology in LOT and SCSP do not match one to one, but they are globally consistent.

7.2.3 The Semiring and LOT Constraint System

If we consider $\mathcal{V}(can)$ instead of $H(can)$ in our optimization problem, the optimal candidate will be the one that has the smallest value for its global valuation; that is, the candidate with a global valuation closest to zero is optimal and the ones with larger values are increasingly suboptimal (i.e., a cost). In this context, the semiring used will be the one shown below in (7.3), where $\mathbb{Z}^*$ is the set of non-negative integers, min chooses the solution, and $+$ combines the values in the carrier set A, (which in this semiring is $\mathbb{Z}^*$). Absolute consistency is denoted by 0 and inconsistency by $+\infty$.

$$S_{\mathrm{LOT}(\mathcal{V})} = \langle \mathbb{Z}^*, min, +, +\infty, 0 \rangle \tag{7.3}$$

This is also known as the *tropical semiring*. $S_{\mathrm{LOT}(\mathcal{V})}$ is a c-semiring since the additive operation, min, is idempotent (i.e., $min(a, a) = a$ for all a), and the multiplicative operation, $+$, is commutative. Also, $+\infty$ is the identity element of the semiring,

min (i.e., $min(a, +\infty) = +\infty$ for all a), and the absorbing element of the semiring, (i.e., $a + \infty = +\infty$ for all a given that $\mathbb{Z}^*$ is the set of non-negative integers (including 0)). The associated ordering $\leq_s$ corresponds to $\geq$ over non-negative integers, which means that smaller numbers correspond to better candidates.

An LOT constraint system is then defined as follows:

DEFINITION 7.2.3 *A linguistic constraint system based on SCSP,* $CS = \langle S_{\mathrm{LOT}(\mathcal{V})}, D, V \rangle$*, will then have the following components:*

- **C-Semiring:** $S_{\mathrm{LOT}(\mathcal{V})} = \langle \mathbb{Z}^*, min, +, +\infty, 0 \rangle$.
- **Variables:** *An ordered set* V *representing the candidate structure.*
- **Domain:** D *a finite set of values that members of* V *can take.*

DEFINITION 7.2.4 **Connection:** $con \subseteq V$*, is called the type of the constraint.*

The set con tells us which variables are involved in each constraint.

DEFINITION 7.2.5 **Domain Function:** $dom : V \rightarrow \mathsf{D}$*, where* $\mathsf{D} \subseteq D$.

The domain function dom tells us which members of D can be assigned to each member of V.

DEFINITION 7.2.6 **Embedding:** $e : Can \hookrightarrow D^k$ *is a bijective function that maps the set of linguistic structures onto vector representations; in particular, for all* $\vec{d} \in D^k$, $d_i \in dom(v_i)$*, where* $1 \leq i \leq k$.

Embedding ensures a representation of linguistic structures that is suitable for the constraint solver. The embedding function must be bijective because once the solver returns a vector as the solution, we want to be able to identify a candidate with that vector.

Bistarelli (2001) also defines a function def as follows:

DEFINITION 7.2.7 **Definition:** $def_k : D^k \rightarrow \mathbb{Z}^*$

This function is actually the composition of $\bar{c}_i$ with g (see Figure 7.2 above and definition 7.2.11 below), that is, $def = g \circ \bar{c}_i$. Based on this definition, a constraint in SCSP is defined as follows:

DEFINITION 7.2.8 **Constraint:** $\langle \bar{c}_i, con^n \rangle$*, for some* $1 \leq n \leq k$ *where* n *is the cardinality of* con.

DEFINITION 7.2.9 **Constraint Weight:** *w_i is a numerical weight associated with each constraint $\langle \bar{c}_i, con_i^n \rangle$. In OT literature, this is known as the* rank *of the constraint.*

DEFINITION 7.2.10 **Valuation:** *$\bar{c}_i : D^k \rightarrow C_i$, a function that evaluates the value d of each variable v. The number returned by this function, in the case of LOT, is the number of violations of the constraint $\langle \bar{c}_i, con_i^n \rangle$. (According to this definition, valuation is equivalent to violation profile.)*

DEFINITION 7.2.11 **Global Valuation:** *$g : C_1 \times C_2 \times \ldots \times C_m \rightarrow \mathbb{Z}^*$ is the combination function that calculates the global valuation of $\vec{d}$ based on a vector of valuations returned by all of the $\bar{c}_i$. In this model, $g(c_1, c_2, \ldots, c_m) = \Sigma_{i=1}^m w_i c_i$ for all $c_i \in C_i$.*

7.2.4 A (Very) Simple Example

Let us see, for instance, how LOT would evaluate the sentences in (7.4).

(7.4) a. The chief executive officer of the largest trading firm in the United States said, "No."

b. "No," said the chief executive officer of the largest trading firm in the United States.

Let us assume that the conflicting constraints determining the order of the discourse units are SN (a discourse constraint on ATTRIBUTION) and LH (Light before Heavy, a prosodic constraint). Let us also assume that SN ranks higher than LH (i.e., $\mathrm{SN} \gg \mathrm{LH}$), with $w(\mathrm{LH}) = 1$ and $w(\mathrm{SN}) = 2$. Thus we have $\mathcal{V}$(7.4a)$' = 1$ (because (7.4a) violates LH but not SN), and $\mathcal{V}$(7.4b)$' = 2$ (because (7.4b) violates SN but not LH). This means that the winning candidate will be (7.4a), where (7.4a)' and (7.4b)' are the grammatical representations of (7.4a) and (7.4b) respectively.

Let us now see how we can represent the LOT example discussed above in the SCSP framework. For simplicity, I ignore the problem of mapping candidate structures onto vector representations (this problem is discussed further in section 7.2.5). Also more detail on the exact definitions of the functions involved is presented in the following section. For now, let us think of Dis and Pros, corresponding to discourse and prosodic constraints, as a shorthand for two instances of $\langle \bar{c}_i, con_i^n \rangle$ (see definition 7.2.8). D will then contain NS, SN, LH, and HL. To visually represent this constraint system, I will be using labelled graph notation. Figure 7.3 depicts this constraint system. The numbers next to the potential values for each variable represent the valuation of the constraint for choosing that value for that variable. Assuming that discourse outranks prosody, we give more weight to discourse constraints, hence a higher cost for NS.

Global valuation is calculated in LOT by summing up the valuations associated with each variable assignment. This is shown in Figure 7.4. Thus the numbers next to the tuples on the arc represent the violation profile of that tuple. It is clear that this constraint system prefers the SN order with the LH constraint satisfied, and it strongly disfavours the NS order with the HL prosodic structure.

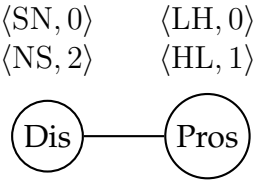


Figure 7.3: An LOT constraint system represented in the labelled graph notation

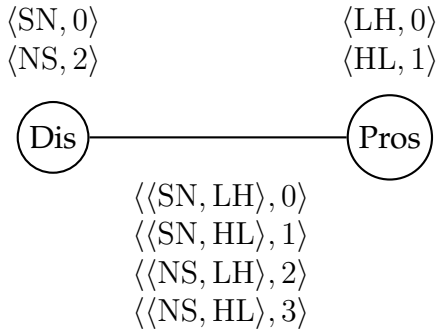


Figure 7.4: An LOT constraint system represented in the labelled graph notation with the valuations calculated

7.2.5 Representing Candidates

As briefly noted in sections 7.2.1 and 7.2.3, the embedding function can map a candidate structure can into a feature vector V^k. When V^k is taken from $\mathbb{Z}^k$ or $\mathbb{R}^k$, this allows us to solve our optimization problem using well-studied mathematical techniques such as integer programming or gradient descent. We also mentioned that embedding must be bijective because once a solution has been found, we want to be able to map the solution, i.e., the winning vector representation, to a candidate structure. The problem of mapping discrete structures such as trees onto such vectors has been studied by Haussler (1999), Collins and Duffy (2001, 2002) and others, and is beyond the scope of this dissertation. What we shall do here instead is to create a very simple feature vector characterizing our candidates for the sake of simplicity of exposition.

SCSP is a theory of constraint satisfaction over finite domains. This means that D has to be finite. Yet, as soon as we admit sentence length into V as a measure of prosodic weight, we are allowing for potential unboundedness because of the recursive nature of language. All sentences that have been or will ever be produced in the world are of finite length but there is no bound on their size. So, in order to meet the finiteness condition on D, we must set an upper bound on sentence length, such as that of the longest sentence that can actually be produced by a single person in his or her lifetime.[1] The lower bound would trivially be 1.

Now if we want to optimize over, say, word order, discourse structure and information structure given the discourse relation and prosodic weights of DUs, and the information status of the sentence components, then according to the parallel modular approach that we have been advocating, a candidate structure will look like the representation in Table 7.1. At this point, we assume that Can is defined only for candidate structures with the same lexical items. In other words, the candidates in a candidate set all represent different realizations of the same sentence.

	The chief executive officer of the largest trading firm in the United States	said,	"No."
Word Order:	Subj	Verb	Obj
Discourse Str.:	Satellite (Attribution)		Nuc
Information Str.:	Theme	Rheme	

Table 7.1: A multilayered representation of a candidate structure

Note that in this simple example syntactic structure reduces to the order of subject, object and verb, and sentential discourse structure only refers to the rhetorical relations at the top-level DUs.

Given this specification of candidate structures, V could contain the following variables: $\langle DRel, NR, SR, SbjR, ObjR, VR, ThR, RhR, DOrd, IOrd, WOrd\rangle$. $DRel$ is the discourse relation involved; NR, SR, $SbjR$, $ObjR$, VR, ThR, and RhR are the spans (see (7.5) below) of the nucleus, satellite, subject, object, verbal group, theme and rheme respectively. Of course, since we have been using syllable counts as a measure of length (prosodic weight), these ranges will also be based on syllable number. $DOrd$ stands for discourse order, $IOrd$ for information structure, and $WOrd$ for word order. The domain function for these variables is defined as follows:

[1]The longest sentence in literary history is said to be Jonathan Coe's novel, *The Rotters' Club*, which contains a sentence of $13,955$ words (Source: `www.bbc.co.uk/radio4/today/reports/archive/arts/sentence.shtml`, accessed April 22, 2005).

(7.5)
a. $dom(DRel) = \{x | x \text{ is an RST relation}\}$
b. $dom(NR) = \{\langle l, r\rangle | 1 \leq l \leq r \leq max(S)\}$, where l and r are the syllable numbers on the left and right edges of the nucleus, and $max(S)$ is the maximum allowable sentence length.
c. $dom(SR) = \{\langle l, r\rangle | 1 \leq l \leq r \leq max(S)\}$, where l and r are the syllable numbers on the left and right edges of the satellite.
d. $dom(SubjR) = \{\langle l, r\rangle | 1 \leq l \leq r \leq max(S)\}$, where l and r are the syllable numbers on the left and right edges of the subject.
e. $dom(ObjR) = \{\langle l, r\rangle | 1 \leq l \leq r \leq max(S)\}$, where l and r are the syllable numbers on the left and right edges of the object.
f. $dom(VR) = \{\langle l, r\rangle | 1 \leq l \leq r \leq max(S)\}$, where l and r are the syllable numbers on the left and right edges of the verbal group.
g. $dom(ThR) = \{\langle l, r\rangle | 1 \leq l \leq r \leq max(S)\}$, where l and r are the syllable numbers on the left and right edges of the theme.
h. $dom(RhR) = \{\langle l, r\rangle | 1 \leq l \leq r \leq max(S)\}$, where l and r are the syllable numbers on the left and right edges of the rheme.
i. $dom(DOrd) = \{\text{NS}, \text{SN}\}$
j. $dom(IOrd) = \{\text{ThRh}, \text{RhTh}\}$
k. $dom(WOrd) = \{\text{SVO}, \text{OSV}, \text{OVS}, \text{VSO}, \text{SOV}\}$

Given a candidate structure, embedding will assign the appropriate values to the variables in V; thus, the representation of the candidate structure shown in Table 7.1 will have the variable assignments shown in Table 7.2. Note that the sentence exemplified in Table 7.1 has 24 syllables, with the subject ranging from syllable 1 to 22, verb from syllable 23 to 23, and object from syllable 24 to 24. Embedding assigns the other values accordingly. Embedding is bijective up to the

Variable	Value	Variable	Value
$DRel$	Attribution	ThR	$\langle 1, 22\rangle$
NR	$\langle 24, 24\rangle$	RhR	$\langle 23, 24\rangle$
SR	$\langle 1, 23\rangle$	$DOrd$	SN
$SbjR$	$\langle 1, 22\rangle$	$IOrd$	ThRh
VR	$\langle 23, 23\rangle$	$WOrd$	SVO
$ObjR$	$\langle 24, 24\rangle$		

Table 7.2: Variable assignments in V for the candidate structure shown in Table 7.1

choice of lexical items. Embedding maps a candidate set onto a set of vector representations. If two candidates can_1 and can_2 map onto a single vector representation

$\vec{d}$, then the only way that can_1 and can_2 can be different will be for them to have different lexical items that together cover identical ranges. In other words, the two candidates can only be distinct sentences that happen to have identical structures and constituents of identical lengths. Since a candidate set consists of different arrangements of the same lexical items, then can_1 and can_2 must be the same.[2]

Valuations are calculated according to the cost of assigning each value to a subset of the variables (see definition 7.2.10). Global valuation in LOT is calculated by weighting and summing up the valuation assignments. Thus, the three constraints that we are considering in our running example along with their associated variables are as follows:

(7.6) a. **Word Order**: $\langle \bar{c}_{WOrd}, con^1_{WOrd} = \{WOrd\}\rangle$
b. **Discourse Structure:** $\langle \bar{c}_{DOrd}, con^1_{DOrd} = \{DOrd\}\rangle$
c. **Information Structure:** $\langle \bar{c}_{IOrd}, con^1_{IOrd} = \{IOrd\}\rangle$

I shall define $\bar{c}_{IOrd}$ below. The other valuation functions, $\bar{c}_{WOrd}$ and $\bar{c}_{DOrd}$ are defined analogously in the next section.

$$\bar{c}_{IOrd} \rightarrow \begin{cases} 0 & \text{iff} \quad IOrd = \text{ThRh} \\ 1 & \text{iff} \quad IOrd = \text{RhTh} \end{cases}$$

This formulation returns 0 if the value of $IOrd$ is ThRh (i.e., theme before rheme) and 1 if the value of $IOrd$ is RhTh (i.e., rheme before theme). Note that since we are using the tropical semiring here, 0 means no violation and 1 means slightly violated.

We should also define a number of hard constraints in this system to ensure the soundness of the solution we get. In particular, we want to make sure that subjects, verbal groups, and objects do not overlap, and neither do nuclei with satellites nor do themes with rhemes. We also want to make sure that the three constraints in (7.6) all point to the same ordering; that is, if the best word order is SVO, and the best discourse order is SN, we want to make sure that the subject of the sentence is inside the satellite. These soundness constraints are defined below:

(7.7) **Non-Overlapping:**

[2]Given the upper and lower bounds on sentence length, other sentence features will also have to be bounded. For example, if we have a feature that represents binding or C-command, we can use tuples of the ranges of the yields of the structures in question. For instance, if we need to make available binding information at interfaces, we can represent it as follows:

- For all pairs of syntactic categories $\langle \alpha, \beta \rangle$ in a syntactic tree t, and their yields $y(\alpha)$ and $y(\beta)$, if α binds β, then $\langle y(\alpha), y(\beta)\rangle \in dom(B)$, where $B \in V$.

I shall leave the details of these representations for future research.

a. $\langle \bar{c}_{syn}, con^3_{syn} = \{SbjR,\ VR,\ ObjR\}\rangle$
b. $\langle \bar{c}_{dis}, con^2_{dis} = \{NR, SR\}\rangle$
c. $\langle \bar{c}_{inf}, con^2_{inf} = \{\ ThR, RhR\}\rangle$

$$left(x) \stackrel{\text{def}}{=} y \quad \text{iff} \quad x = \langle y, z\rangle$$

$$right(x) \stackrel{\text{def}}{=} z \quad \text{iff} \quad x = \langle y, z\rangle$$

$$range(x) \stackrel{\text{def}}{=} \{y | left(x) \leq y \leq right(x)\}$$

$$\bar{c}_{syn} \rightarrow \begin{cases} 0 & \text{iff} \quad range(SbjR) \cap range(\ VR) \cap range(\ ObjR) = \emptyset \\ +\infty & \text{otherwise} \end{cases}$$

And analogously for $\bar{c}_{dis}$ and $\bar{c}_{inf}$. The non-overlapping constraints in (7.7) return 0 if the ranges do not overlap and $+\infty$ if they do.

7.3 Graded Linguistic Constraints

The last section showed how LOT can be cast into the SCSP framework. One limitation of LOT (and consequently its SCSP variant) is that it does not handle graded constraints, and as discussed in section 5.3.3.1, this kind of constraint is needed to handle the prosodic effect observed in sentential discourse structure (see also the discussion in section 3.5).

If we were to incorporate the statistical model that we built in Chapter 5 into LOT, we would need to allow the valuation of constraints to take real number values in addition to simple integers. Now since global valuation is the weighted sum of the individual valuations $\bar{C}_i \in \mathbb{R}^*$, this means that we are promoting the c-semiring $\mathrm{S_{LOT}}$ to the c-semiring used in weighted constraint satisfaction problems, $\mathrm{S_{WCSP}} = \langle \mathbb{R}^*, min, +, +\infty, 0\rangle$, where $\mathbb{R}^*$ is the set of non-negative real numbers.

The valuation functions $\bar{c}_{WOrd}$, $\bar{c}_{IOrd}$, and $\bar{c}_{DOrd}$ for the present constraint system are presented in Table 7.3. The word order possibilities along with their valuations have been selected subjectively based on what is possible with verbs of reporting as this is the kind of verb that we will be considering in our examples in this chapter. A more elaborate constraint system would need to take the type of verb into account.

The weight of the word-order, discourse-order, and information-structure constraints shown in (7.6) are 1, 2 and 1 respectively and have been assigned through trial and error. Of course, training algorithms exist that can estimate these parameters automatically.

According to this model, the optimal word order for the sentence represented in Table 7.1 is (7.4b) repeated below as (7.8b) with a global valuation of 1.52.

Variable	Value	Valuation	Weight
$WOrd$	SVO	0.00	1
	OSV	0.25	
	OVS	0.50	
	VSO	0.75	
	SOV	1.00	
$DOrd$	NS	$1 - p(\text{NS}\|DRel, \delta, \lambda)$	2
	SN	$p(\text{NS}\|DRel, \delta, \lambda)$	
$IOrd$	ThRh	0.00	1
	RhTh	1.00	

Table 7.3: Values returned by the valuations functions $\bar{c}_{WOrd}$, $\bar{c}_{DOrd}$, and $\bar{c}_{IOrd}$. The probabilities are calculated according to the formula presented in (5.11)–(5.14).

(7.8) a.

The chief executive officer of the largest trading firm in the United States	said,	"No."
Sbj	V	Obj
S		N
Theme	Rheme	

Global Valuation: 1.97

b.

"No,"	said	the chief executive officer of the largest trading firm in the United States.
Obj.	V	Sbj
N	S	
Rheme	Theme	

Global Valuation: 1.52

Note that the valuation of constraint violation is not just 0 or 1 (or only the *number* of violations) anymore. More importantly, as is modelled in the Discourse Structure constraint, valuation is now graded and sensitive to the probability of the sentence appearing in a certain order or another. Note that the SN order is associated with $p(\text{NS}|DRel, \delta, \lambda)$, and the NS order is associated with $1 - p(\text{NS}|DRel, \delta, \lambda)$. This is because we want the constraint to return a cost. It should also be noted that these measures are not simply defining a probability distribution over a discrete event. The $DOrd$ constraint, the way it has been formulated here, in fact takes into account the interaction of two constraints: one from discourse that prefers the SN order ATTRIBUTION and BACKGROUND sentences and NS for ENABLEMENT and

EXPLANATION sentences, and one prosodic constraint that prefers heavy discourse units last. The influence of this prosodic constraint on the discourse constraint has been measured as the probability of NS conditioned upon the discourse relation, the length difference between the discourse units and, in the case of ATTRIBUTION sentences the total length of the sentence.

Let us also consider (7.9) and (7.10). Between (7.9a) and (7.9b), the model selects (7.9a) with a global valuation of 1.36 as opposed to 1.88 for (7.9b). Between (7.10a) and (7.10b), however, it picks (7.10b) with a global valuation of 0.50 over (7.10a), which gets a global valuation of 1.99.

(7.9) a. The spokesman warned, "It will be very expensive."
Global Valuation: 1.36

b. "It will be very expensive," the spokesman warned.
Global Valuation: 1.88

(7.10) a. Charles C. Mihalek, a Lexington attorney and former Kentucky state securities commissioner warns, "It's a big-risk business."
Global Valuation: 1.99

b. "It's a big-risk business," warns Charles C. Mihalek, a Lexington attorney and former Kentucky state securities commissioner.
Global Valuation: 0.50

Of course, we can still count the number of violations of the violable constraints; for graded constraints also, we can count the number of the applications of the constraint and sum up the result of each application in order to account for ganging up effects.

The source code for a simple Prolog implementation of this SCSP-based grammar is provided in Appendix B.

7.4 Toward Graded Unification-Based Grammars

In Chapter 6, we argued that a parallel modular grammar architecture leads to simpler modules and captures generalizations better. One important advantage of implementing such an architecture in a unification-based framework is that unification naturally allows for the modules to constrain one another. Since all modules of grammar build their respective structures with reference to a common list of lexical items (see Figure 6.2 on page 73), as soon as the value of a feature is bound in one module, all other features whose values are structure-shared with that feature have their values bound. This means that even though the modules may not care about nor see the details inside other modules, they cannot generate structures

that are unacceptable to other modules. It would then be natural to try to implement the proposed soft-constraint satisfaction system in a unification-based framework such as HPSG. In order to do this, we need to change how type antecedent constraints are enforced without modifying the unification mechanism. Standard HPSG type antecedent constraints are crisp; their violation causes the generated structure to be rejected. Multiple constraints on a type are explicitly connected by logical AND, which also means the violation of any one constraint results in the rejection of the generated structure. This constraint system roughly corresponds to Bistarelli's $\mathrm{S}_{\mathrm{CSP}}$ mentioned in section 4.4.1 on page 37.

Malouf (2003) argues that the fact that an analysis naturally falls out of OT's notion of ranked violable constraints does not necessarily mean that it has to be analyzed that way. He states that OT suffers from a "procedural metaphor;" that is, the theory relies on some cognitively unreal and intractable operations to account for acceptable structures. The most notable part of this metaphor is the generate-and-test procedure of the theory where a partial representation (such as a logical form) is fed to a component called *Gen* that generates an infinite number of candidate output structures to be evaluated against a set of constraints by *HEval*. This is a common concern. The solution that Malouf suggests is to discard the procedural metaphor along with the violability of the constraints, and account for his data using HPSG type hierarchy. His analysis, although elegant, does not leave any room for graded grammaticality judgements, accounting for multiple violations of the same constraint, and ganging up effects, not to mention the graded constraints that we have been discussing in this dissertation. In this section, I show that we can incorporate soft constraints within constraint-based grammars such as HPSG without resorting to any procedural metaphors.

In order to account for violable constraints as well as degrees of constraint violation, multiple constraint violations, and ganging up effects discussed in Keller (2000), we can use the weighted CSP ($\mathrm{S}_{\mathrm{WCSP}} = \langle \mathbb{R}^*, min, +, +\infty, 0 \rangle$) paradigm defined in the previous section.

The following subsection goes over some illustrative examples. It should be mentioned that the goal of the following examples is not to derive the "correct" analysis but to show how a system of type antecedent constraints based on the tropical semiring would calculate costs for different analyses.

7.4.1 Examples

In the case of sentences (6.58) and (6.59) repeated below as (7.11) and (7.12) respectively, we can formulate a constraint as in Figure 7.5. For simplicity of exposition, this constraint only employs two non-head daughters. The extension of the constraint to accommodate more daughters is straightforward.

(7.11) a. He wanted to demonstrate it to us.
b. He wanted to demonstrate that life to us.
c. He wanted to demonstrate the consequences to us.
d. ? He wanted to demonstrate the consequences of such an unholy life to us.
e. ?? He wanted to demonstrate the consequences of such a horribly filthy and unholy life to us.

(7.12) a. * He wanted to demonstrate to us it.
b. ?? He wanted to demonstrate to us that life.
c. He wanted to demonstrate to us the consequences of such an unholy life.

$$\textit{hd-comp-ph} \Rightarrow \begin{bmatrix} \text{PHON} & \langle \boxed{1}, \boxed{3} \rangle \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \text{PHON} & \boxed{1} \end{bmatrix}, \begin{bmatrix} \text{PHON} & \boxed{2} \end{bmatrix} \right\rangle \end{bmatrix} \wedge length\Big(\boxed{1}\Big) \leq length\Big(\boxed{2}\Big)$$

Figure 7.5: An HPSG formulation of the LH constraint on verb complements

The valuation function for the LH constraint as formulated above can be calculated according to the following function:

(7.13) **Valuation Function for LH:**

$$\text{Given the description } \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \boxed{3} \oplus \langle \boxed{1}, \boxed{2} \rangle \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \text{PHON} & \boxed{1} \end{bmatrix}, \begin{bmatrix} \text{PHON} & \boxed{2} \end{bmatrix} \right\rangle \end{bmatrix},$$

$$val(\text{LH}) = \left(\frac{length(\boxed{1}) - length(\boxed{2})}{length(\boxed{1}) + length(\boxed{2})} + 1\right) \times .5$$

Let us assume, for now, that $length(x)$ is the number of syllables in x, and that the weight of the constraint LH is 1. The formula in (7.13) returns a number between 0 and 1. If the two complements are of equal size, the number returned will be 0.5; the number approaches 1 as the first complement gets longer than the second, and it approaches 0 as the second complement gets longer than the first. Of course, one can think of many ways to formulate LH. The definition presented here is just one of them. What function models the exact behaviour of LH remains to be seen.

We can now see how the sentences in (7.11) are evaluated in terms of LH. The valuations of LH calculated for (7.11a–e) are shown in (7.14) below.

(7.14) a. For (7.11a): $val(\text{LH}) = (\frac{1-2}{1+2} + 1) \times .5 \approx .33$
b. For (7.11b): $val(\text{LH}) = (\frac{2-2}{2+2} + 1) \times .5 = .5$
c. For (7.11c): $val(\text{LH}) = (\frac{5-2}{5+2} + 1) \times .5 \approx .71$
d. For (7.11d): $val(\text{LH}) = (\frac{7-2}{7+2} + 1) \times .5 \approx .78$
e. For (7.11e): $val(\text{LH}) = (\frac{10-2}{10+2} + 1) \times .5 \approx .83$

As can be seen, this accounts for the declining acceptability of the examples in (7.11).

The examples in (7.12) demonstrate the interaction of two constraints: (i) LH, and (ii) the constraint that requires verbal complements to appear in descending order of obliqueness (call it CompOrd). If obliqueness is a total order defined over verbal complements represented with $>_o$, then we can formulate CompOrd as in Figure 7.6. The valuation function for the CompOrd is defined in (7.15).

$$\textit{hd-comp-ph} \Rightarrow \left[\text{NON-HD-DTRS} \quad \left\langle \boxed{1},\boxed{2} \right\rangle\right] \wedge \boxed{1} >_o \boxed{2}$$

Figure 7.6: An HPSG formulation of CompOrd

(7.15) **Valuation Function for** CompOrd**:**

Given the description $\begin{bmatrix} \textit{hd-comp-ph} \\ \text{NON-HD-DTRS} \quad \left\langle \boxed{1},\boxed{2} \right\rangle \end{bmatrix}$,

$$val(\text{CompOrd}) = \begin{cases} 0 & \text{iff } \boxed{1} >_o \boxed{2} \\ 1 & \text{otherwise} \end{cases}$$

$val(\text{CompOrd})$ is defined as a *characteristic* or *selector* function returning either 0 or 1, but notice that since we are using the tropical semiring these values do not have their traditional *true* or *false* meaning. In this constraint system, 0 corresponds to no violation and 1 corresponds a violation of degree 1. Also notice, since we are adding costs, multiple instances of such constraint violations will incrementally increase global evaluation as it does in LOT. Let us assume that the two constraints LH and CompOrd have equal weights. Then the valuation of the sentences in (7.12a–c) with respect to LH and CompOrd is calculated as in (7.16).

(7.16) a. For (7.12a): $val(\text{LH}) + val(\text{ComOrd}) = .66 + 1 = 1.66$
b. For (7.12b): $val(\text{LH}) + val(\text{ComOrd}) = .50 + 1 = 1.50$

c. For (7.12c): $val(\text{LH}) + val(\text{COMORD}) = .14 + 1 = 1.14$

This analysis quantitatively captures the increasing acceptability of the sentences in (7.12) as the sentence-final direct object gets longer than the indirect object.

An interesting outcome of this analysis is that it naturally captures speaker's intuitions about the relative acceptability of forms like the ones in (7.17) without the need to posit an arbitrary constraint prohibiting ending a dative construction with a pronoun (which would completely rule out (7.17b), incorrectly). According to this analysis, we not only capture the graded grammaticality of each example, we can also show how much each example is worse than the other.[3]

(7.17) a. Give it to me.
$val(\text{LH}) + val(\text{COMPORD}) \approx .33 + 0 = .33$

b. ?? Give me it.
$val(\text{LH}) + val(\text{COMPORD}) = .5 + 1 = 1.5$

c. * Give to me it.
$val(\text{LH}) + val(\text{COMPORD}) \approx .66 + 1 = 1.66$

Let us now consider how information structure can be integrated into this model. Let THRH stand for the violable constraint that requires the theme to appear before the rheme. A version of this constraint for just one theme and one rheme is shown in Figure 7.7; an extension to the constraint for multiple themes and rhemes is also straightforward. The valuation function for THRH is formu-

$$\textit{clause} \Rightarrow \begin{bmatrix} \text{DOM} & \boxed{1} \oplus \boxed{2} \\ \text{INFO} & \left\langle \begin{bmatrix} \textit{theme} & \\ \text{I-DOM} & \boxed{1} \end{bmatrix}, \begin{bmatrix} \textit{rheme} & \\ \text{I-DOM} & \boxed{2} \end{bmatrix} \right\rangle \end{bmatrix}$$

Figure 7.7: An HPSG formulation of THRH

lated in (7.18).

(7.18) **Valuation Function for** THRH**:**

Given the description $\begin{bmatrix} \textit{clause} & \\ \text{INFO} & \left\langle \boxed{1}, \boxed{2} \right\rangle \end{bmatrix}$,

$$val(\text{THRH}) = \begin{cases} 0 & \text{iff } type(\boxed{1}) = \textit{theme} \wedge type(\boxed{2}) = \textit{rheme} \\ 1 & \text{otherwise} \end{cases}$$

[3]Note that we are assuming equal weights for these constraints. Estimating the exact weights of the constraints requires having access to training data obtained through corpus analysis or experimental work.

Let us assume that the preferred response to the question "What did John give to the man?" is (7.19a) as opposed to (7.19b).[4]

(7.19) a. [He gave]$^{\theta_1}$ [money]$^{\rho}$ [to the man]$^{\theta_2}$.
b. [He gave the man]$^{\theta}$ [money]$^{\rho}$.

Again assuming equal weights for LH, COMPORD, and THRH, we can calculate the global valuations of these sentences with respect to these three constraints as in (7.20). It can be seen that (7.19a) gets a lower global valuation (i.e., is preferred by the model).

(7.20) a. For (7.19a): $val(\text{LH}) + val(\text{COMPORD}) + val(\text{THRH}) = .4 + 0 + 1 = 1.4$
b. For (7.19b): $val(\text{LH}) + val(\text{COMPORD}) + val(\text{THRH}) = .5 + 1 + 0 = 1.5$

In this example, THRH has been violated in favour of COMPORD and LH. Note that since the difference in the lengths of the two verb complements is small the two sentences show a small difference in their global valuation (0.1).

Let us look at another example in which the difference in the lengths of the verb complements is larger.

(7.21) a. [He gave]$^{\theta_1}$ [a lot of his hard earned money]$^{\rho}$ [to the man]$^{\theta_2}$.
b. [He gave the man]$^{\theta}$ [a lot of his hard earned money]$^{\rho}$.

The global valuations of these sentences are given below:

(7.22) a. For (7.21a): $val(\text{LH}) + val(\text{COMPORD}) + val(\text{THRH}) = .95 + 0 + 1 = 1.95$
b. For (7.21b): $val(\text{LH}) + val(\text{COMPORD}) + val(\text{THRH}) = .04 + 1 + 0 = 1.04$

Here we see that 7.21b is preferred. We also see that the difference between the global valuations of (7.21a) and (7.21b) is larger than before ($.91$), which means that in this example the alternative is costlier than in the previous example. In other words, the gradient characterization of LH captures the fact that larger differences in the lengths of the complements result in higher degrees of constraint violation if the heavier constituent appears before the lighter one, an observation that we made in Chapter 5 and was also made by Arnold et al. (2000). In addition, the c-semiring-based implementation of type antecedent constraints in HPSG allows for capturing the ganging up effects of constraint violation as well as multiple violations of the same constraint (since valuations are summed up).

[4] Again note that we are not making any strong claims as to what sentence is actually the preferred response. This should be determined through separate studies. The point here is to illustrate how valuation calculations work.

7.4.2 Feature Structure Cost Calculation

At this point, some remarks are in order about how costs are to be assigned to feature structures. Note that in the examples presented in the previous subsection, two of the constraints that we discussed were written for objects of type *hd-comp-ph* and the third was written for *clause*. We were implicitly assuming that the cost of a feature structure equals the sum of the costs of its substructures. In this subsection, we discuss this issue in more detail and talk about some other conditions that we assume to hold for cost assignment to feature structures.

7.4.2.1 The cost of a feature structure is the sum of the costs of its substructures

The cost of a feature structure, f, of type τ is the weighted sum of the valuations of all the constraints imposed on τ with respect to f plus the sum of the costs of all the feature values of f. This is formalized in (7.23).

(7.23) $cost(f_\tau) = \sum_i w_i \cdot val(c_i^\tau) + \sum_j cost(g_j)$,
where f_τ is the feature structure of type τ to which we want to assign a cost; c_i^τ is a constraint on the type τ; $val(c_i^\tau)$ is the valuation of c_i^τ, and g_j is a feature value of f.

The formula in (7.23) means that the cost of a feature structure is never less than the sum of the costs of its substructures (provided that there are no negative weights, which is what we have been assuming). If there are any cases where the same description gets different valuations in different contexts (i.e., in different feature structures), then we can replace our original constraint with other more specific constraints.

The reason for this desideratum is twofold: (i) We want to make sure that every part of the feature structure is contributing information about constraint violations in substructures; and (ii) We want the constraints to be local; that is, every constraint has to sensitive only to the description on its consequent and should not be affected by the context in which it is applied. For instance, consider the following feature structures:

(7.24) a. $\begin{bmatrix} t & \\ \text{F} & a \end{bmatrix}$

b. $\begin{bmatrix} u & \\ \text{G} & a \end{bmatrix}$

If the cost of a feature structure f of type a depends on whether f is the value of F or G, then we cannot formulate a single constraint on type a because it would make that constraint on a non-local as it would have to have information about

what feature the feature structure f (of type a) is a value of. Instead, we must formulate two constraints on t and u making reference to the value of the F and G features, respectively.

7.4.2.2 Overlapping constraints should not conflate global valuation

If two constraints overlap in their denotation, then the parameter estimation algorithm should ensure that the overlapping parts do not conflate global valuation. This means that we want our theory to make the same predictions with or without redundancies in constraint definitions.

For example, assume that we have $t \sqsubseteq u$ (i.e., if t subsumes u). Then the first conjuncts of the following two constraints overlap.

(7.25) a. $t \Rightarrow \text{F}{:}d \wedge \text{G}{:}c$
b. $u \Rightarrow \text{F}{:}d \wedge \text{H}{:}e$

As another example, again assume that we have $t \sqsubseteq u$, and $a \sqcup b = d$. Then the following two constraints also overlap in the first conjunct of their consequent.

(7.26) a. $t \Rightarrow \text{F}{:}(a \wedge b) \wedge \text{G}{:}c$
b. $u \Rightarrow \text{F}{:}d \wedge \text{H}{:}e$

Thus, our parameter estimation algorithm should be able to recognize such overlaps and adjust the weights of the constraints so that the predictions of the grammar are the same as that of a grammar without these redundancies.

In general, consider types t, u, and v, where $t \sqcup u = v$. If we have,

(7.27) a. $t \Rightarrow \phi$
b. $u \Rightarrow \psi$

then we have $val(f_v) = val(g_t) + val(h_u)$ when it comes to the constraints in (7.27) above.

7.4.2.3 Overriding default unification must not be penalized

Default unification (Lascarides et al., 1996) is sometimes used in the HPSG literature to account for regularities in the lexicon or grammar; these regularities can be overridden by constraints on more specific types. Considering this purpose of default unification, we should not allow the acceptability of an irregular form to drop due to overriding a default value. For instance, *went* is neither a better nor a worse past tense verb than *walked*; therefore, overriding the default value that assumes the addition of *-ed* to the verb stem results in a past tense should not affect the valuation of an irregular verb form. As another example, consider Sag's (1997)

Valence Principle that is formulated in terms of a constraint on headed phrases (*hd-ph*).

(7.28) **Sag's (1997) Valence Principle:**

$$\textit{hd-ph} \Rightarrow \begin{bmatrix} \text{SUBJ} & /\boxed{1} & \\ \text{SPR} & /\boxed{2} & \\ \text{COMPS} & /\boxed{3} & \\ \text{HD-DTR} & \begin{bmatrix} \text{SUBJ} & /\boxed{1} \\ \text{SPR} & /\boxed{2} \\ \text{COMPS} & /\boxed{3} \end{bmatrix} \end{bmatrix}$$

This constraint states that the value for a valence feature of a phrase is identical to that of the head daughter of that phrase, unless it is an instance of a more specific type than *hd-ph* that says otherwise. And an example of such a subtype is *hd-subj-ph*.

(7.29) $$\textit{hd-subj-ph} \Rightarrow \begin{bmatrix} \text{SUBJ} & \langle\rangle & \\ \text{HD-DTR} & \begin{bmatrix} \text{SUBJ} & \left\langle \boxed{1} \right\rangle \\ \text{SPR} & \langle\rangle \end{bmatrix} \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \text{SYNSEM} & \boxed{1} \end{bmatrix} \right\rangle \end{bmatrix}$$

Obviously, a head-subject phrase that has filled up its subject position is a desirable thing and should not be penalized for that.

Defaults, however, are sometimes used as an alternative to probabilities or numerical weights. But since our constraints now come with weights, we do not need such uses of defaults. In our view here, defaults are merely a convention for theory description or elucidation and therefore should come with no penalty. This is consistent with Sag's view of defaults.

7.4.2.4 Structure-shared values are evaluated as many times as they occur in the feature structure

When it comes to structure-shared feature values, we have two options for calculating valuations. We can either evaluate only one instance of the structure-shared feature value or we can evaluate all instances (effectively evaluate the greatest rational approximant of the feature structure). This should not matter as long as we are consistent. However, one issue arises if we do not take into account the number of occurrences of the structure-shared value. Consider the following example:

(7.30) $\begin{bmatrix} t & \\ \text{F} & a \\ \text{G} & a \end{bmatrix}$

This description is consistent with the following two feature structures:

(7.31) a. $\begin{bmatrix} t & \\ \text{F} & \boxed{1}a \\ \text{G} & \boxed{2}a \\ \boxed{1} \neq \boxed{2} & \end{bmatrix}$

b. $\begin{bmatrix} t & \\ \text{F} & \boxed{1}a \\ \text{G} & \boxed{1} \end{bmatrix}$

That is, (7.30) induces the signature shown in Figure 7.8. This signature is isomorphic to the one shown in Figure 7.9 that includes only totally well-typed feature structures.[5] The predictions of the theory should remain the same regardless of

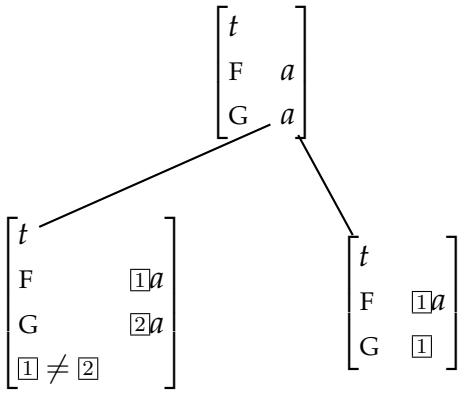


Figure 7.8: The feature structure hierarchy induced by (7.30)

whether we use the signature in Figure 7.8 or the one in Figure 7.9. According to Figure 7.9, the cost of a feature structure of type *be* or *bi* is equal to the valuations of the constraints assigned to *b* plus those assigned to *be* or *bi*, respectively. This means that a feature structure of type *b* will always have a cost lower than that of its subtypes. We want the same to hold for Figure 7.8 as well. But if we ignore structure-sharing and calculate the cost of (7.31b) based on only one instance of *a*, then we are effectively allowing a feature structure that is a subtype of *b* to have

[5]For the proof of such isomorphisms, see Penn (2000).

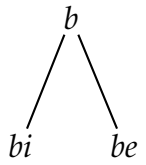


Figure 7.9: A totally well-typed signature isomorphic to the one shown in Figure 7.8

a cost that is lower than the cost of a feature structure that is of type *b* itself. To avoid this situation, we need to assign costs to every instance of structure-shared feature values. This also means that we are viewing feature structures as information states; the more specific a type is, the more information we have and thus are able to assign higher costs given the constraints on that type and all its supertypes.

7.5 Summary

This chapter showed how soft constraints can be implemented within a parallel modular approach to linguistic theorizing. It was also shown that graded constraints, whose violation cannot be counted, can also be incorporated in this framework without the loss of explanatory power. One advantage of formulating linguistic constraints within the SCSP framework is that advances made in the c-semiring based constraint satisfaction theory will have direct positive impact on the linguistic theories that are based on it. We also mentioned that a philosophical outcome of this approach is that linguistic constraint satisfaction is shown to be an instance of general human constraint satisfaction. This situates linguistics alongside other human cognitive faculties.

We also hinted at how unification-based grammars can incorporate soft constraints in an SCSP-based framework. It is of course much too early to make any conclusions in this respect, but if such an enterprize proves fruitful, we will have brought together the underlying constraint satisfaction mechanism in distinct theories such as OT and HPSG.

Chapter 8

Discussion and Directions for Future Research

This thesis incorporates a generalized c-semiring based theory of soft constraint satisfaction within a parallel modular grammar architecture. Since conflicts are expected to occur more frequently at interfaces than within modules, this work argued for implementing soft constraints at the interfaces while maintaining the traditional hard-constraint approach within modules. It was shown that this integration is possible without losing the expressive power of the existing models.

One interesting outcome of this work is that the incorporation of the SCSP framework shows that unification-based theories and optimality theory can be brought closer together and viewed as fundamentally the same as far as the underlying constraint satisfaction engine is concerned. What is even more interesting is that the SCSP framework has been designed in a domain completely distinct from linguistics. A theory like that attempts to model human decision-making, and its successful incorporation in linguistic theory implies that linguistic constraint solving is not any different from other human constraint-solving processes. What makes language special is the constraints that are involved and not the method of constraint satisfaction. This outcome brings language closer to other human cognitive faculties than traditional views of grammar have done.

Another advantage of the approach advocated in this thesis is that we now have a metatheoretical diagnostic tool to pinpoint soft constraints; that is, whenever we see gradience in grammar, we can expect it to be the result of intermodular conflict resolution, which in turn can lead us to find the source of the conflict rather than posit arbitrary rules or constraints.

This work can be pursued in many different directions. One obvious way to follow up this work would be to try to work out the formal details of incorporating

SCSP in a theory like HPSG. Another way to pursue would be to try to implement an SCSP-based constraint solver in a logic-programming language like ALE (Carpenter and Penn, 1999). One can also investigate different learning algorithms for an SCSP-based grammar. This architecture is also useful in studies of language learner error as proposed by Menzel and Schröder (1999). As another very important step to take next, one can also mention investigating other possible sources of graded constraints such as the ones mentioned earlier in this thesis.

Appendix A

Rhetorical Structure Theory

Rhetorical Structure Theory (RST, Mann, 1984; Mann and Thompson, 1986, 1988a,b, among others) is a descriptive theory of text organization that has had a variety of applications from natural language generation, (Mann, 1984; Mann and Thompson, 1986) and automatic text summarization (Marcu, 1997, 2000), to the linguistic analysis of various types of text, and teaching writing skills.[1] The theory views text as a hierarchical structure with functional relations among its constituent parts. Similar ideas have been expressed in other discourse analytic works such as Polanyi and Scha (1984) and Polanyi et al. (2003).

RST stands on four major components: *relations*, *schemas*, *schema applications*, and *structures*. Relations define the particular relationships that can exist between two parts of a text. Schemas define structural patterns in which a particular span of text can be realized based on the relations. Schema applications define the ways that a schema can be instantiated. The composition of schema applications result in a structure for the whole text. These are discussed in more detail in the following sections.

A.1 Relations

In RST, two non-overlapping discourse constituents (or *discourse units* to use the RST terminology) can stand in certain hierarchical relations: a subordinating relation, or a coordinating relation. A text span involved in a relationship is labelled as either a *nucleus* or a *satellite* (N or S). A relationship is then defined over a piece of text spanning over nuclei and satellites. Essentially, a nucleus is considered to

[1] The interested reader is referred to the RST Web site which contains several bibliographies of RST-related work (`http://www.sil.org/~mannb/rst/`).

- Circumstance
- Solutionhood
- Elaboration
- Background
- Enablement and Motivation
  - Enablement
  - Motivation
- Evidence and Justify
  - Evidence
  - Justify
- Relations of Cause
  - Volitional Cause
  - Non-Volitional Cause
  - Volitional Result
  - Non-Volitional Result
  - Purpose
- Antithesis and Consession
  - Antithesis
  - Consession
- Condition and Otherwise
  - Condition
  - Otherwise
- Interpretation and Evaluation
  - Interpretation
  - Evaluation
- Restatement and Summary
  - Restatement
  - Summary
- Other Relations
  - Sequence
  - Contrast

Figure A.1: Hierarchical List of RST Relations

be more essential to the author's purpose than the satellite. The definition of a relationship contains five fields. If a field is not present in the definition, it means that there are no constraints enforced by that field.

(A.1) **Information fields in the definitions of relations:**

a. Constraints on the nucleus
b. Constraints on the satellite
c. Constraints on the combination of nucleus and satellite
d. The effect
e. The locus of the effect

Figure A.1 lists the list of the RST relations as presented in Mann and Thompson (1988b).

The definitions of some of the relations in Figure A.1 on the preceding page appear below (adapted from Mann and Thompson, 1988b). In these definitions, N stands for nucleus, S for satellite, R for reader, and W for writer.

Antithesis

- **Constraints on N:** W has positive regard for the situation presented in N.
- **Constraints on N+S combination:** The situation presented in N and S are in contrast (cf. CONTRAST). Because of an incompatibility that arises from the contrast, one cannot have positive regard for both the situations presented in n and S. Comprehending S and the incompatibility between the situations presented in N and S increases R's positive regard for the situation presented in N.
- **The effect:** R's positive regard for N is increased.
- **Locus of the effect:** N

Background

- **Constraints on S:** R will not comprehend N sufficiently before reading text of S.
- **Constraints on N+S combination:** S increases the ability of R to comprehend an element in N.
- **The effect:** R's ability to comprehend N increases.
- **Locus of the effect:** N

Circumstance

- **Constraints on S:** S presents a situation (not unrealized)
- **Constraints on N+S combination:** S sets a framework in the subject matter within which R is intended to interpret the situation presented in N.
- **The effect:** R recognizes that the situation in S provides the framework for interpreting N.
- **Locus of the effect:** N and S

Concession

- **Constraints on N:** W has positive regard for the situation presented in N.
- **Constraints on S:** W is not claiming that the situation presented in S does not hold.
- **Constraints on N+S combination:** W acknowledges a potential or apparent incompatibility between the situations presented in N and S; W regards the situations presented in N and S as compatible; recognizing that the compatibility between the situations presented in N and S increases R's positive regard for the situation presented in N.
- **The effect:** R's positive regard for the situation presented in N is increased.
- **Locus of the effect:** N and S

Enablement

- **Constraints on N:** It presents R an action (including an offer), unrealized with respect to the context of N.
- **Constraints on N+S combination:** R comprehending S increases R's potential ability to perform the action presented in N.
- **The effect:** R's potential ability to perform the action presented in N increases.
- **Locus of the effect:** N

Evidence

- **Constraints on N:** R might not believe N to a degree satisfactory to W.
- **Constraints on S:** The reader believes S or will find it credible.
- **Constraints on N+S combination:** R's comprehending S increases R's belief of N.
- **The effect:** R's belief of N is increased.
- **Locus of the effect:** N

Justify

- **Constraints on N+S combination:** R's comprehending S increases R's readiness to accept W's right to present N.
- **The effect:** R's readiness to accept W's right to present N is increased.
- **Locus of the effect:** N

Motivation

- **Constraints on N:** It presents an action in which R is the actor (including an offer), unrealized with respect to the context of N.
- **Constraints on N+S combination:** Comprehending S increases R's desire to perform action presented in N.
- **The effect:** R's desire to perform action presented in N is increased.
- **Locus of the effect:** N

Sequence

- **Constraints on N:** multinuclear
- **Constraints on combination of nuclei:** A succession relationship between the situations is presented in the nuclei.
- **The effect:** R recognizes the succession relationships among the nuclei.
- **Locus of the effect:** multiple nuclei

Contrast

- **Constraints on N:** multinuclear
- **Constraints on combination of nuclei:** no more than two nuclei; the situation presented in these two nuclei are (a) comprehended as the same in many respects, (b) comprehended as differing in a few respects, and (c) compared with respect to one or more of these differences
- **The effect:** R recognizes the succession relationship among the nuclei.
- **Locus of the effect:** multiple nuclei

Volitional Result

- **Constraints on S:** It presents a volitional action or a situation that could have arisen from a volitional action.
- **Constraints on the N+S combination:** N presents a situation that could have caused the situation presented in S. The situation presented in N is more central to W's purposes than is that presented in S.
- **The effect:** R recognizes that the situation in N could be a cause for the action or situation presented in S.
- **Locus of the effect:** N and S

Joint

The schema called JOINT has no corresponding relation. The schema is multinuclear, and no relation is claimed to hold between the nuclei.

A.2 Schemas

Similar to rule schemas in HPSG, RST schemas define the structural organization of constituents, in this case, discourse constituents. As Mann and Thompson (1988b) put it:

> They are abstract patterns consisting of a small number of constituent text spans, a specification of the relations between them, and a specification of how certain spans (nuclei) are related to the whole collection.

RST recognizes four rule schemas exemplified in Figure A.2.

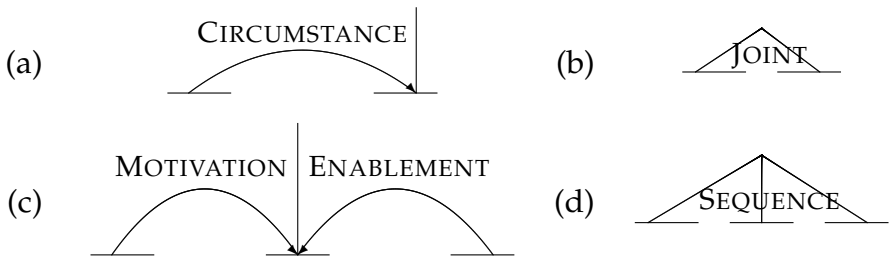


Figure A.2: Examples of the four RST schemas

Schemas (c) and (d) in Figure A.2 on the preceding page only represent the relations shown. Schema (b) represents JOINT and CONTRAST relations; whereas, schema (a) represents all the other relations that involve a nucleus and a satellite. The multi-nuclear schemas obviously represent organizational patterns of text that involve more than one nucleus. CONTRAST always has exactly two nuclei. SEQUENCE and JOINT, in principle, can have infinitely many nuclei. Note that in these schemas order is irrelevant. The next section presents all the relations that Mann and Thompson (1988b) present in their paper.

A.3 Examples

This section contains some sample texts with their RST analyses.

(A.2) a. The next music day is scheduled for July 21 (Saturday), noon-midnight.

b. I'll post more details later,

c. but this is a good time to reserve the place on your calendar.

In (A.2), units (A.2b) and (A.2c) are in a JUSTIFY relation with unit (A.2a). They inform the reader why the writer believes that he is uttering (A.2a) without any details about the location of the event. The desired effect of CONCESSION (and ANTITHESIS) "is to cause the reader to have a positive regard for the nucleus" (Mann and Thompson, 1988b, p. 253). The RST diagram for (A.2) is given in Figure A.3.

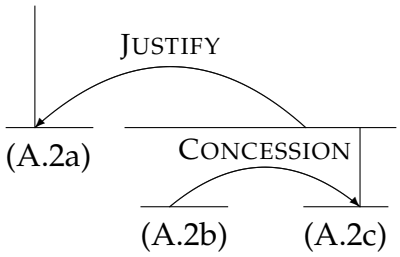


Figure A.3: The RST diagram of text (A.2)

The following example illustrate the use of ANTITHESIS. The RST diagram for (A.3) is given in Figure A.4 on the next page.

(A.3) a. Farmington police had to help control traffic recently

b. when hundreds of people lined up to be among the first applying for jobs at the yet-to-open Marriott Hotel.

c. The hotel's help-wanted announcement—for 300 openings—was a rare opportunity for many unemployed.
d. The people waiting in line carried a message, a refutation, of claims that the jobless could be employed if only they showed enough moxie.

e. Every rule has exceptions,
f. but the tragic and too-common tableaux of hundreds or even thousands of people snake-lining up for any task with a paycheck illustrates a lack of jobs,
g. not laziness.

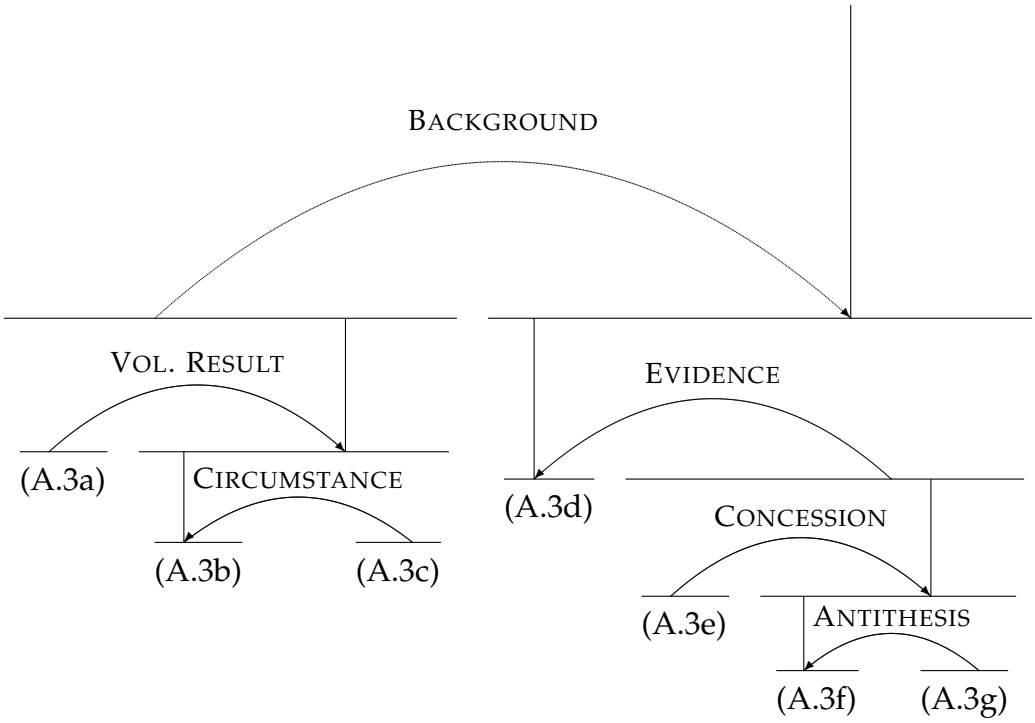


Figure A.4: The RST diagram of text (A.3)

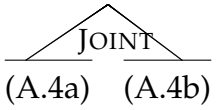


Figure A.5: The RST diagram of text (A.4)

Text (A.4) is an example of the JOINT schema in which several ideas are presented with no relations among them.

(A.4) a. Employees are urged to complete new beneficiary designation forms for retirement or life insurance whenever there is a change in marital or family status.

b. Employees who are not sure who is listed as their beneficiary should complete new forms since the retirement system and the insurance carrier use the most current form to disburse benefits.

Appendix B

A simple c-semiring based linguistic constraint solver

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                       %
%                  Simple C-semiring based                              %
%                linguistic constraint solver                           %
%                                                                       %
% Written by: Mohammad Haji-Abdolhosseini                               %
% For       : SWI-Prolog Version 5.4.7                                  %
% Date:     : April 23, 2005                                            %
%                                                                       %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% There are three constraints:
%   Word Order
%   Discourse Structure
%   Information Structure

% Simplifications:
%   (a) NLen and SLen are passed to the solver.
%   (b) Actual lexical items are passed instead of ranges or
%       bit vectors.

:- dynamic bestSol/2.

go:-
```

```
    \+ (data(NLen,SLen,Rel,Nuc,Sat,Subj,Obj,Verb,Theme,Rheme),
        sol(NLen,SLen,Rel,Nuc,Sat,Subj,Obj,Verb,Theme,Rheme,
                Sol,Valuation),
        write(Sol),nl,
        write(Valuation),nl,nl,
        fail).

sol(NLen,SLen,Rel,Nuc,Sat,Subj,Obj,Verb,Theme,Rheme,Sol,
    Valuation):-
    retractall(bestSol(_,_)),
    pns(Rel,NLen,SLen,P_NS),
    sort_cons([W1-C1,W2-C2,W3-C3]),
    solve(n-Nuc,s-Sat,s-Subj,o-Obj,v-Verb,t-Theme,r-Rheme,
          P_NS,W1,W2,W3,C1,C2,C3,Sol,Valuation).

solve(n-Nuc,s-Sat,s-Subj,o-Obj,v-Verb,t-Theme,r-Rheme,P_NS,
      W1,W2,W3,C1,C2,C3,Sol,BestValuation):-
    \+ ( dom(C1,[P_NS],Val1,Cost1),
         cons(C1,Val1,n-Nuc,s-Sat,s-Subj,o-Obj,v-Verb,
            t-Theme,r-Rheme,Sol1),
         dom(C2,[P_NS],Val2,Cost2),
         cons(C2,Val2,n-Nuc,s-Sat,s-Subj,o-Obj,v-Verb,
            t-Theme,r-Rheme,Sol1),
         dom(C3,[P_NS],Val3,Cost3),
         cons(C3,Val3,n-Nuc,s-Sat,s-Subj,o-Obj,v-Verb,
            t-Theme,r-Rheme,Sol1),
         Valuation is (W1*Cost1)+(W2*Cost2)+(W3*Cost3),
         update(Sol1,Valuation),
         fail
       ),
    bestSol(Sol,BestValuation).

update(NSol,NV):-
    bestSol(Sol,V) ->
        ( NV < V ->
            (retract(bestSol(Sol,V)),
             assert(bestSol(NSol,NV))
        );
          true
        );
```

```
        assert(bestSol(NSol,NV)).

cons(wordOrd,Val,_,_,s-Subj,o-Obj,v-Verb,_,_,Sol):-
    nth1(NA,Val,s),
    nth1(NB,Val,v),
    nth1(NC,Val,o),
    Sol1=[_,_,_],
    nth1(NA,Sol1,Subj),
    nth1(NB,Sol1,Verb),
    nth1(NC,Sol1,Obj),
    flatten(Sol1,Sol).

cons(disOrd,Val,n-Nuc,s-Sat,_,_,_,_,_,Sol):-
    nth1(NA,Val,n),
    nth1(NB,Val,s),
    Sol1=[_,_],
    permutation(Nuc,Nuc1),
    permutation(Sat,Sat1),
    nth1(NA,Sol1,Nuc1),
    nth1(NB,Sol1,Sat1),
    flatten(Sol1,Sol).

cons(infOrd,Val,_,_,_,_,_,t-Theme,r-Rheme,Sol):-
    nth1(NA,Val,t),
    nth1(NB,Val,r),
    Sol1=[_,_],
    permutation(Theme,Theme1),
    permutation(Rheme,Rheme1),
    nth1(NA,Sol1,Theme1),
    nth1(NB,Sol1,Rheme1),
    flatten(Sol1,Sol).

sort_cons([W1-C1,W2-C2,W3-C3]):-
    weight(wordOrd, WordOrdW),
    weight(disOrd, DisOrdW),
    weight(infOrd, InfOrdW),
    sort([WordOrdW-wordOrd,DisOrdW-disOrd,InfOrdW-infOrd],
         [W3-C3,W2-C2,W1-C1]).

% Constraint Weights
```

```
weight(wordOrd,1).%
weight(disOrd,2).%
weight(infOrd,1).

% Probability of NS

pns(attribution,NLen,SLen,P_NS):-
    Mu is NLen + SLen,
    Delta is NLen - SLen,
    P_NS is 1/(1+(exp(1.8+0.05*Delta-0.23*Mu))).
pns(background,NLen,SLen,P_NS):-
    Delta is NLen - SLen,
    P_NS is 1/(1+exp(0.56+0.04*Delta)).
pns(enablement,NLen,SLen,P_NS):-
    Delta is NLen - SLen,
    P_NS is 1/(1+exp(-2.22+0.05*Delta)).
pns(explanation,NLen,SLen,P_NS):-
    Delta is NLen - SLen,
    P_NS is 1/(1+exp(-1.48+0.06*Delta)).

% Variable domains
% dom(Variable,Conditions,Value,Cost)

dom(wordOrd, _,          [s,v,o], 0).%
dom(wordOrd, _,          [o,s,v], 0.25).%
dom(wordOrd, _,          [o,v,s], 0.5).%
dom(wordOrd, _,          [v,s,o], 0.75).%
dom(wordOrd, _,          [s,o,v], 1).

dom(disOrd,  [P_NS],[s,n],  Cost):-
    Cost is P_NS.
dom(disOrd,  [P_NS],[n,s],  Cost):-
    Cost is 1 - P_NS.

dom(infOrd,  _,          [t,r],  0).%
dom(infOrd,  _,          [r,t],  1).

data(NLen,SLen,Rel,Nuc,Sat,Subj,Obj,Verb,Theme,Rheme):-
  A = 'the chief executive officer of the largest trading
```

```
       firm in the united states',
  B = 'said',
  C = 'no',
  Nuc = [C],
  Sat = [A,B],
  Subj = [A],
  Obj = [C],
  Verb = [B],
  Theme = [A],
  Rheme = [B,C],
  Rel = attribution,
  NLen = 1,
  SLen = 21.

data(NLen,SLen,Rel,Nuc,Sat,Subj,Obj,Verb,Theme,Rheme):-
  A = 'the chief executive officer of the largest trading
       firm in the united states',
  B = 'said',
  C = 'no',
  Nuc = [C],
  Sat = [A,B],
  Subj = [A],
  Obj = [C],
  Verb = [B],
  Theme = [B,C],
  Rheme = [A],
  Rel = attribution,
  NLen = 1,
  SLen = 21.

data(NLen,SLen,Rel,Nuc,Sat,Subj,Obj,Verb,Theme,Rheme):-
  A = 'the spokesman',
  B = 'warned',
  C = 'it will be very expensive',
  Nuc = [C],
  Sat = [A,B],
  Subj = [A],
  Obj = [C],
  Verb = [B],
  Theme = [A, B],
```

```
  Rheme = [C],
  Rel = attribution,
  NLen = 8,
  SLen = 4.

data(NLen,SLen,Rel,Nuc,Sat,Subj,Obj,Verb,Theme,Rheme):-
  A = 'charles c. mihalek, a lexington attorney and former
       kentuky state securities commissioner',
  B = 'warns',
  C = 'it\'s a big-risk business',
  Nuc = [C],
  Sat = [A,B],
  Subj = [A],
  Obj = [C],
  Verb = [B],
  Theme = [],
  Rheme = [A,B,C],
  Rel = attribution,
  NLen = 6,
  SLen = 27.
```

Bibliography

Aarts, B. (2004a). Conceptions of gradience in the history of linguistics. *Language Sciences 26*(4), 343–389.

Aarts, B. (2004b). Modelling linguistic gradience. *Studies in Language 28*(1), 1–49.

Aarts, B., D. Denison, E. Keizer, and G. Popova (Eds.) (2004). *Fuzzy Grammar: A reader*. Oxford: Oxford University Press.

Abney, S. (1996). Statistical methods and linguistics. In J. Klavans and P. Resnik (Eds.), *The Balancing Act: Combining Symbolic and Statistical Approaches to Language*. Cambridge, MA: The MIT Press.

Abney, S. (1997). Stochastic attribute-value grammars. *Computational Linguistics 23*(4), 597–618.

Altmann, G. (1987). Modularity and interaction in sentence processing. See Garfield (1987).

Andrews, A. (1990). Case structure and control in modern icelandic. In J. Maling and A. Zaenen (Eds.), *Modern Icelandic Syntax*, Volume 24 of *Syntax and Semantics*, pp. 187–234. New York: Academic Press.

Ariel, M. (1990). *Accessing Noun-Phrase Antecedents*. Cambridge: Routledge.

Arnold, J. E. (1998). *Reference Form and Discourse Patterns*. Ph.D. thesis, Stanford University.

Arnold, J. E., T. Wasow, A. Losongco, and R. Ginstrom (2000). Heaviness vs. newness: The effects of structural complexity and discourse status on constituent ordering. *Language 76*(1), 28–55.

Bard, E. G., C. Frenck-Mestre, L. Kelly, K. Killborn, and A. Sorace (1999). Judgement and perception of gradable linguistic anomaly. Unpublished ms., Human Communication Research Centre, University of Edinburgh.

Bard, E. G., D. Robertson, and A. Sorace (1996). Magnitude estimation of linguistic acceptability. *Language 72*(1), 32–68.

Belletti, A. and L. Rizzi (1988). Psych verbs and theta theory. *Natural Language and Linguistic Theory 2*(1), 65–86.

Bing, J. (1979). *Aspects of English Prosody*. Ph.D. thesis, University of Massachusetts.

Bird, S. (1990). *Constraint-Based Phonology*. Ph.D. thesis, University of Edinburgh.

Bird, S. (1995). *Computational Phonology: A Constraint-Based Approach*. Studies in Natural Language Processing. Cambridge: Cambridge.

Bird, S. and E. Klein (1994). Phonological analysis in typed feature systems. *Computational Linguistics 20*, 455–91.

Birner, B. J. (1992). *The Discourse Function of Inversion in English*. Ph.D. thesis, Northwestern University, Evanston, IL.

Birner, B. J. (1994). Information status and word order: An analysis of English inversion. *Language 70*(2), 233–259.

Bistarelli, S. (2001). *Soft Constraint Solving and Programming: A General Framework*. Ph.D. thesis, Università di Pisa.

Bod, R. (1998). *Beyond Grammar: An Experience-Based Theory of Language*. CSLI, Cambridge University Press.

Bod, R., J. Hay, and S. Jannedy (Eds.) (2003). *Probabilistic Linguistics*, Cambridge, MA. MIT Press.

Bod, R., R. Scha, and K. Sima'an (Eds.) (2003). *Data-Oriented Parsing*. CSLI, University of Chicago Press.

Bolinger, D. L. (1961). *Generality, gradience and all-or-none*. The Hague: Mouton.

Borning, A., B. Freeman-Benson, and M. Wilson (1992). Constraint hierarchies. *Lisp and Symbolic Computation 5*(3), 223–270.

Borsley, R. and J. Kornfilt (2000). Mixed extended projections. In R. Borsley (Ed.), *The Nature and Function of Syntactic Categories*, Volume 32 of *Syntax and Semantics*, pp. 101–131. New York: Academic Press.

Bresnan, J. (Ed.) (1982). *The Mental Representation of Grammatical Relations*. Cambridge, MA: The MIT Press.

Bresnan, J. (1997). Mixed categories as head sharing constructions. In M. Butt and T. Holloway (Eds.), *Proceedings of the LFG97 Converende*, Stanford, CA, pp. 1–17. University of California, San Diego: CSLI Publications. Online: `http://csli-publications.stanford.edu/LFG2/lfg97-toc.html`.

Bresnan, J. (2001). *Lexical-Functional Syntax*. Blackwell Textbooks in Linguistics. Malden, MA: Blackwell.

Büring, D. (2001). Let's phrase it! — focus, word order, and prosodic phrasing in German double object constructions. In G. Müller and W. Sternefeld (Eds.), *Competition in Syntax*, Number 49 in Studies in Generative Grammar, pp. 101–137. Berlin & New York: de Gruyter.

Butt, M. and T. H. King (1998). Interfacing phonology with LFG. In M. Butt and T. H. King (Eds.), *Proceedings of the LFG98 Conference*, Stanford, CA. CSLI. http://csli-publications.stanford.edu/LFG/3/butt-king/butt-king.html.

Carlson, L., D. Marcu, and M. E. Okurowski (2002). RST Discourse Treebank. FTP File `http://www.ldc.upenn.edu/`. LDC2002T07.

Carpenter, B. and G. Penn (1999). ALE the attribute logic engine: User's guide. available online at http://www.cs.toronto.edu/~gpenn/ale/files/aleguide.ps.gz.

Carroll, G. and M. Rooth (1998). Valence induction with a head-lexicalized PCFG. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Granada, pp. 36–45.

Chafe, W. (1976). Givenness, contrastiveness, definiteness, subjects, topics, and point of view. In C. N. Li (Ed.), *Subject and Topic*. New York: Academic Press.

Charniak, E. (1993). *Statistical Language Learning*. Cambridge, MA: The MIT Press.

Choi, H. (2001). Phrase structure, information structure, and resolution of mismatch. In P. Sells (Ed.), *Formal and Empirical Issues in Optimality Theoretic Syntax*, Studies in Constraint-Based Lexicalism, pp. 17–62. Stanford, CA: CSLI.

Chomsky, N. (1955). *The Logical Structure of Linguistic Theory*. New York: Plenum Press.

Chomsky, N. (1961). Some methodological remarks on generative grammar. *Word 17*, 219–239.

Chomsky, N. (1965). *Aspects of the Theory of Syntax*. Cambridge, MA: The MIT Press.

Chomsky, N. (1995). *The Minimalist Program*. Cambridge, MA: The MIT Press.

Chomsky, N. and G. Miller (1963). Formal properties of grammars. In R. Luce, R. Bush, and E. Galanter (Eds.), *Handbook of Mathematical Psychology*, Volume II, pp. 323–428. New York: Wiley.

Clifton, C. J. and F. Ferreira (1987). Modularity in sentence comprehension. See Garfield (1987).

Collins, M. and N. Duffy (2001). Parsing with a single neuron: Convolution kernels for natural language problems. Technical Report UCSC-CRL-01-01, University of California at Santa Cruz.

Collins, M. and N. Duffy (2002). Convolution kernels for natural language. In T. G. Dietterich, S. Becker, and Z. Ghahramani (Eds.), *Advances in Neural Information Processing Systems 14*, Cambridge, MA. MIT Press.

Cooper, W. and J. Paccia-Cooper (1980). *Syntax and Speech*. Cambridge, Mass.: Harvard University Press.

Corver, N. and H. van Riemsdijk (2001). Semi-lexical categories. In N. Corver and N. van Riemsdijk (Eds.), *Semi-Lexical Categories: The Function of Content Words and the Content of Function Words*, pp. 1–19. Berlin: Mouton de Gruyterm.

Cowart, W. (1997). *Experimental Syntax: Applying Objective Methods to Sentence Judgements*. Thousand Oaks, CA: Sage Publications.

Crain, S. and M. Steedman (1985). On not being led up the garden path. In D. Dowty, L. Kartunnen, and A. M. Zwicky (Eds.), *Natural language parsing: Psycholinguistic, computational, and theoretical perspectives*. Cambridge: Cambridge University Press.

Davey, B. and H. Priestley (1990). *Introduction to Lattices and Order*. Cambridge Mathematical Textbooks. Cambridge: Cambridge University Press.

De Kuthy, K. (2002). The information structure of discontinuous NPs in German. In L. H. van Eynde, Frank and D. Beermann (Eds.), *Proceedings of the 8th International HPSG Conference*, Stanford, pp. 148–161. Norwegian University of Science and Technology: CSLI.

Dechter, R. and J. Pearl (1988). Network-based heuristics for constraint-satisfaction problems. In Kanal and Kumar (Eds.), *Search in Artificial Intelligence*, pp. 370–425. Springer-Verlag.

Dubois, D., H. Fargier, and H. Prade (1993). The calculus of fuzzy restrictions as a basis for flexible constraint satisfaction. In *Proceedings of IEEE International Conference on Fuzzy Systems*, pp. 1131–1136. IEEE.

Engdahl, E. and E. Vallduví (1994). *Information Packaging and Grammar Architecture: A Constraint-Based Approach*, Volume 1.3.B of DYANNA-2 Report, pp. 41–78. Amsterdam: ILLC.

Erteschik-Shir, N. and S. Lappin (1979). Dominance and the functional explanation of island phenomena. *Theoretical Linguistics 6*, 41–86.

Fargier, H. and J. Lang (1993). Uncertainty in constraint satisfaction problems: A probabilistic approach. In *Proceedings of the European Conference on Symbolic and Qualitative Approaches to Reasoning and Uncertainty (ECSQARU), number 747 in LNCS*, pp. 97–104. Springer-Verlag.

Fodor, J. A. (1983). *The Modularity of Mind: An Essay on Faculty Psychology*. Cambridge, MA: The MIT Press.

Foth, K., W. Menzel, and I. Schröder (2005). Robust parsing with weighted constraints. *Natural Language Engineering 11*(1), 1–25.

Frazier, L. (1987). Theories of sentence processing. See Garfield (1987).

Freuder, E. and R. J. Wallace (1992). Partial constraint satisfaction. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence, IJCAI-89*, Number 58, Detroit, MI.

Garfield, J. L. (Ed.) (1987). *Modularity in Knowledge Representation and Natural-Language Understanding*, Cambridge, MA. The MIT Press.

Givón, T. (1983). *Topic continuity in discourse*. Quantitative cross-language studies, TSL #3. Amsterdam: Benjamins.

Haji-Abdolhosseini, M. (2003a). A constraint-based approach to information structure and prosody correspondence. In S. Müller (Ed.), *Proceedings of The HPSG-2003 Conference*, pp. 143–162. `http://cslipublications.stanford.edu/HPSG/4/`.

Haji-Abdolhosseini, M. (2003b). Information-prosody correspondence in HPSG. In M. Barrie, M. Haji-Abdolhosseini, and J. Herd (Eds.), *Proceedings of the Fourth Annual Meeting of the Niagara Linguistic Society*, Volume 21 of *Toronto Working Papers in Linguistics*, Toronto, pp. 43–60. Dept. of Linguistics, University of Toronto.

Haussler, D. (1999). Convolution kernels on discrete structures. Technical Report UCS-CRL-99-10, UC Santa Cruz.

Hawkins, J. A. (1994). *A Performance Theory of Order and Constituency*. Cambridge Studies in Linguistics 73. Cambridge: Cambridge University Press.

Hosmer, D. W. and S. Lemeshow (1989). *Applied Logistic Regression*. New York: John Wiley & Sons.

Hudson, R. (2003). Gerunds without phrase structure. *Natural Language and Linguistic Theory 21*(3), 579–615.

Jackendoff, R. (1987). *Consciousness and the Computational Mind*. Cambridge, MA: The MIT Press.

Jackendoff, R. (1992). *Languages of the Mind: Essays on Mental Representation*. The MIT Press.

Jackendoff, R. (1997). *The Architecture of the Language Faculty*. Linguistic Inquiry: Monograph Twenty-Eight. Cambridge, Mass.: The MIT Press.

Jackendoff, R. (2000). Fodorian modularity and representational modularity. In Y. Grodzinsky, L. P. Shapiro, and D. Swinney (Eds.), *Language and the Brain: Representation and Processing*. San Diego: Academic Press.

Jackendoff, R. (2002). *Foundations of language: Brain, Meaning, Grammar, Evolution*. New York, NY: Oxford.

Joos, M. (1950). Description of language design. *Journal of the Acoustical Society of America 22*, 701–708.

Kahane, H. and R. Beym (1948). Syntactical juncture in colloquial Mexican Spanish. *Language 24*, 388–396.

Kathol, A. (1995). *Linearization-Based German Syntax*. Ph.D. thesis, Ohio State University.

Kathol, A. (2000). *Linear Syntax*. Oxford: Oxford University Press.

Keller, F. (2000). *Gradience in Grammar: Experimental and Computational Aspects of Degrees of Grammaticality*. Ph.D. thesis, University of Edinburgh.

Keller, F. (2003). A probabilistic parser as a model of global processing difficulty. In R. Alterman and D. Kirsh (Eds.), *Proceedings of the 25th Annual Conference of the Cognitive Science Society*, Boston, pp. 646–651.

Keller, F. and T. Alexopoulou (2001). Phonology competes with syntax: Experimental evidence for the interaction of word order and accent placement in the realization of information structure. *Cognition 79*(3), 301–372.

Klein, E. (2000). Prosodic constituency in HPSG. In R. Cann, C. Grover, and P. Miller (Eds.), *Grammatical Interfaces in HPSG*, Studies in Constraint-Based Lexicalism, pp. 169–200. Stanford: CSLI.

Ladd, D. R. (1996). *Intonational Phonology*. Cambridge Studies in Linguistics. New York: Cambridge.

Lakoff, G. (1973a). Fuzzy grammar and the performance/competence terminology game. In C. T. Corum, C. Smith-Stark, and A. Weiser (Eds.), *Papers from the Ninth Regional Meeting of the Chicago Linguistic Society*, Chicago, IL, pp. 271–291.

Lakoff, G. (1973b). Hedges: a study in the meaning criteria and the logic of fuzzy concepts. *Journal of Philosophical Logic 2*, 458–508.

Lakoff, G. (1987a). Cognitive models and prototype theory. In C. T. Corum, C. Smith-Stark, and A. Weiser (Eds.), *Concepts and Conceptual Development: Ecological and Intelligence Factors in Categorization*, pp. 63–100. Cambridge: Cambridge University Press.

Lakoff, G. (1987b). *Women, Fire and Dangerous Things: What Categories Reveal about the Mind*. Chicago, IL: University of Chicago Press.

Lambrecht, K. and L. Michaelis (1998). Sentence accent in information questions: Default and projection. *Linguistics and Philosophy 21*(5), 477–544.

Lascarides, A., T. Briscoe, N. Asher, and A. Copestake (1996). Order independent and persistent typed default unification. *Linguistics and Philosophy 19*(1), 1–89.

Leech, G., B. Francis, and X. Xu (1994). The use of computer corpora in the textual demonstrability of gradience in linguistic categories. In C. Fuchs and B. Victorri (Eds.), *Continuity in Linguistic Semantics*, Volume 19 of *Studies in French and General Linguistics*, pp. 57–76. Philadelphia: John Benjamins.

Liberman, M. and A. Prince (1977). On stress and linguistic rhythm. *Linguistic Inquiry 8*, 249–336.

Malouf, R. P. (2000). *Mixed Categories in the Hierarchical Lexicon*. Stanford, CA: CSLI Publications.

Malouf, R. P. (2003). Cooperating constructions. In E. Francis and L. Michaelis (Eds.), *Mismatch: Form-function Incongruity and the Architecture of Grammar*, pp. 403–424. Stanford: CSLI Publications.

Mann, W. C. (1984). Discourse structure for text generation. In *Proceedings of the 22nd Annual Meeting*. Association for Computational Linguistics.

Mann, W. C. and S. A. Thompson (1986). Rhetorical structure theory: description and construction of text structures. In G. Kempen (Ed.), *Natural Language Generation: New Results in Artificial Intelligence, Psychology and Linguistics*, Proceedings of the Third International Workshop on Text Generation, Nijmegen, The Netherlands.

Mann, W. C. and S. A. Thompson (1988a). Rhetorical structure theory: a theory of text organization. In L. Polanyi (Ed.), *The Structure of Discourse*, Number 3. Norwood, N.J.: Ablex.

Mann, W. C. and S. A. Thompson (1988b). Rhetorical structure theory: Toward a functional theory of text organization. *Text 8*(3), 243–281.

Manning, C. (2003). Probabilistic approaches to syntax. See Bod et al. (2003).

Manning, C. and H. Schütze (2000). *Foundations of Statistical Natural Language Processing*. Cambridge, MA: The MIT Press.

Marcu, D. (1997). *The Rhetorical Parsing, Summarization, and Generation of Natural Language Texts*. Ph.D. thesis, Dept. of Computer Science, University of Toronto.

Marcu, D. (2000). *The Theory and Practice of Discourse Parsing and Summarization*. Cambridge, MA: The MIT Press.

Marcus, M. P., B. Santorini, and M. A. Marcinkiewicz (1993). Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics 19*(2), 313–330.

Marriott, K. and P. Stuckey (1998). *Programming with Constraints: An Introduction*. Cambridge, MA: The MIT Press.

Marslen-Wilson, W. (1973). *Speech Shadowing and Speech Perception*. Ph.D. thesis, MIT.

Marslen-Wilson, W. and L. K. Tyler (1987). Against modularity. See Garfield (1987).

McCawley, J. (1977). The nonexistence of syntactic categories. In *Second Annual Metatheory Conference Proceedings*, East Lansing, MI. Michigan State University.

McCawley, J. (1982). *Thirty Million Theories of Grammar*. Chicago, IL: University of Chicago Press.

McCawley, J. (1998). *The Syntactic Phenomena of English* (Second Ed. ed.). Chicago, IL: University of Chicago Press.

Menzel, W. and I. Schröder (1999). Error diagnosis for language learning systems. *ReCALL* (Special Edition, May 1999), 20–30.

Moulin, H. (1988). *Axioms of Cooperative Decision Making*. Cambridge: Cambridge University Press.

Nespor, M. and I. Vogel (1986). *Prosodic Phonology*, Volume 28 of *Studies in generative grammar*. Riverton, NJ: Foris.

Newell, A. (1982). The knowledge level. *Artificial Intelligence 18*(1), 87–127.

Newell, A. (1990). *Unified Theories of Cognition*. Cambridge, MA: Harvard University Press.

Oakes, M. P. (1998). *Statistics for Corpus Linguistics*. Edinburgh Textbooks in Empirical Linguistics. Edinburgh: Edinburgh University Press.

Oatley, K. (1992). *Best Laid Schemes: The Psychology of Emotions*. Cambridge: Cambridge.

Oatley, K. and P. Johnson-Laird (1987). Towards a cognitive theory of emotions. *Cognition and Emotions 1*, 29–50.

Penn, G. (1999a). A generalized-domain-based approach to Serbo-Croatian second position clitic placement. In G. Bouma, E. Hinrichs, G. Kruijff, and R. Oehrle (Eds.), *Constraints and Resources in Natural Language Syntax and Semantics*, Studies in Constraint-Based Lexicalism, pp. 119–136. Stanford: CSLI.

Penn, G. (1999b). Linearization and WH-extraction in HPSG: Evidence from Serbo-Croatian. In *Slavic in Head-Driven Phrase Structure Grammar*, Studies in Constraint-Based Lexicalism, pp. 149–182. Stanford: CSLI.

Penn, G. (2000). *The Algebraic Structure of Attributed Type Signatures*. Ph. D. thesis, School of Computer Science, Language Technologies Institute, Carnegie Mellon University.

Penn, G. and M. Haji-Abdolhosseini (2003). Topological parsing. In *Proceedings of the 10th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 283–290.

Pierrehumbert, J. (1980). *The phonology and phonetics of English intonation*. Ph.D. thesis, MIT. published 1988 by IULC.

Pinker, S. (1999). *Words and Rules: The Ingredients of Language*. London: Weidenfeld and Nicolson.

Polanyi, L. and R. Scha (1984). A syntactic approach to discourse semantics. In *Proceedings COLING10*, Stanford, CA, pp. 413–419. CSLI Publications.

Polanyi, L., M. van den Berg, and D. Ahn (2003). Discourse structure and sentential information structure. *Journal of Logic, Language and Information 12*, 337–350.

Pollard, C. and I. Sag (1987). *Information-Based Syntax and Semantics, Volume I: Fundamentals*. Number 13 in CSLI Lecture Notes. Stanford: CSLI.

Pollard, C. and I. Sag (1994). *Head-Driven Phrase Structure Grammar*. Studies in Contemporary Linguistics. Chicago: CSLI.

Prevost, S. (1995). *A semantics of contrast and information structure specifying intonation in spoken language generation*. Ph.D. thesis, University of Pennsylvania.

Prevost, S. and M. Steedman (1994). Specifying intonation from context for speech synthesis. *Speech Communication 15*, 139–153.

Prince, A. and P. Smolensky (1993). Optimality theory: Constraint interaction in generative grammar. Technical Report 2, Rutgers University Center for Cognitive Science, Piscataway, NJ.

Pulgram, E. (1970). *Syllable, Word, Nexus, Cursus*. The Hague: Mouton.

Radford, A. (1976). On the non-discrete nature of the verb-auxiliary distinction in english. *Nottingham Linguistic Circle 5*(2), 8–19.

Reape, M. (1994). Domain union and word order variation in German. In J. Nerbonne, K. Netter, and C. J. Pollard (Eds.), *German in Head-Driven Phrase Structure Grammar*, Number 46 in CSLI Lecture Notes, pp. 151–197. CSLI Publications.

Rietveld, T. and R. van Hout (1993). *Statistical Techniques for the Study of Language and Language Behaviour*. Berlin: Mouton de Gruyter.

Riezler, S., D. Prescher, J. Kuhn, and M. Johnson (2000). Lexicalized stochastic modeling of constraint-based grammars using log-linear measures and EM training. In *Proceedings of the 38th Annual Meeting of the ACL*.

Rohde, D. L. T. (2002). TGrep2 User Manual: version 1.06. `http://tedlab.mit.edu/~dr/Tgrep2/`.

Rosenfeld, A., R. Hummel, and S. Zucker (1976). Scene labelling by relaxation operations. *IEEE Transactions on Systems, Man, and Cybernetics 6*(6), 420–433.

Ross, J. R. (1969a). Adjectives as noun phrases. In D. A. Reibel and S. A. Shane (Eds.), *Modern studies in English*, pp. 352–360. Englewood Cliffs, NJ: Prentice Hall.

Ross, J. R. (1969b). Auxiliaries as main verbs. In W. Todd (Ed.), *Studies in Philosophical Linguistics. Series I*, pp. 77–102. Evanstown, IL: Great Expectations Press.

Ross, J. R. (1972). The category squish: endstation hauptwort. In P. M. Peranteau, J. N. Levi, and G. C. Phares (Eds.), *Papers from the Eighth Regional Meeting of the Chicago Linguistic Society*, Chicago, IL, pp. 316–328.

Ross, J. R. (1973a). A fake NP squish. In C. J. Bailey and R. W. Shuy (Eds.), *New Ways of Analyzing Variation in English*, pp. 96–140. Washington D.C.: Georgetown University Press.

Ross, J. R. (1973b). Nouniness. In O. Fujimura (Ed.), *Three Dimensions of Linguistic Research*, pp. 137–257. Tokyo: TEC Company Ltd.

Ross, J. R. (1974). Three batons for cognitive psychology. In W. Weimer and D. Palermo (Eds.), *Cognition and Symbolic Processes*, pp. 63–124. Hillsdale, NJ: Lawrance Erlbaum Associates.

Ross, J. R. (1987). Islands and syntactic prototypes. In B. Need, E. Schiller, and A. Bosch (Eds.), *Papers from the Twenty-Third Regional Meeting of the Chicago Linguistic Society*, Chicago, IL, pp. 309–320. Lawrance Erlbaum Associates.

Ross, J. R. (2000). The frozenness of pseudoclefts: towards an inequlity-based syntax. In A. Okrent and J. P. Boyle (Eds.), *Papers from the Thirty-Sixth Regional Meeting of the Chicago Linguistic Society*, Chicago, IL, pp. 385–426. Lawrance Erlbaum Associates.

Ruttkay, Z. (1994). Fuzzy constraint satisfaction. In *Proceedings of the 3rd IEEE International Conference on Fuzzy Systems*, pp. 1263–1268.

Sag, I. (1997). English relative clause constructions. *Journal of Linguistics 33*(2), 431–484.

Sankoff, D. (1988). Variable rules. In U. Ammon, N. Dittmar, and K. Mattheier (Eds.), *Sociolinguistics. An International Handbook of the Science of Language and Society*, pp. 984–997. Berlin: Walter de Gruyter.

Schiex, T., H. Fargier, and G. Verfaille (1995). Valued constraint satisfaction problems: Hard and easy problems. In *Proceedings of IJCAI95*, pp. 631–637. Morgan Kaufman.

Schröder, I. (2002). *Natural Language Parsing with Graded Constraints*. Ph. D. thesis, Fachbereich Informatik der Universität Hamburg.

Selkirk, E. O. (1981a). On prosodic structure and its relation to syntactic structure. In T. Fretheim (Ed.), *Nordic Prosody II: Papers from a Symposium*, pp. 111–140. Trondheim: Tapir.

Selkirk, E. O. (1981b). *The phrase phonology of English and French*. Bloomington: Indiana University Linguistics Club. Originally presented as the author's thesis, Massachusetts Institute of Technology, 1972.

Selkirk, E. O. (1984). *Phonology and Syntax: The Relation between Sound and Structure*. Cambridge, MA: The MIT Press.

Selkirk, E. O. (1986). On derived domains in sentence phonology. *Phonology Yearbook 3*, 371–405.

Selkirk, E. O. (1996). The prosodic structure of function words. In J. L. Morgan and K. Demuth (Eds.), *Signal to Syntax: Bootstrapping from Syntax to Grammar in Early Acquisition*, pp. 187–213. Mahwah, NJ: Lawrence Elbaum Associates.

Simon, H. A. (1996). *The Sciences of the Artificial* (3rd ed.). Cambridge, MA: The MIT Press.

Sorace, A. and F. Keller (2005). Gradience in linguistic data. *Lingua 115*(11), 1497–1524.

Soricut, R. and D. Marcu (2003). Sentence level discourse parsing using syntactic and lexical information. In *Proceedings of the Human Language Technology and North American Association for Computational Linguistics Conference (HLT/NAACL)*, Edmonton, Canada.

Steedman, M. (1991). Structure and intonation. *Language 25*(2), 193–247.

Steedman, M. (2000a). Information structure and syntax-phonology interface. *Linguistic Inquiry 31*(4), 649–689.

Steedman, M. (2000b). *The syntactic process*. Cambridge, MA: The MIT Press.

Stevens, S. S. (1975). *Psychophysics: Introduction to its perceptual, neural, and social prospects*. New York: John Wiley.

Vallduví, E. and E. Engdahl (1996). The linguistic realization of information packaging. *Linguistics 34*, 459–519.

van Riemsdijk, H. (1998). Categorial feature magnetism: the endocentricity and distribution of projections. *Journal of Comparative Germanic Linguistics 2*, 1–48.

van Riemsdijk, H. (1999). A far from simple matter: syntactic reflexes of syntax-pragmatics misalignments. Masters thesis, University of Tilburg.

Wasow, T. (1997). Remarks on grammatical weight. *Language Variation and Change 9*, 81–105.

Wasow, T. (2002). *Postverbal Behavior*. Chicago: CSLI Publications.

Zwicky, A. M. (1982). Stranded *to* and phonological phrasing. *Linguistics 20*, 3–57.