

What *Do* We *Do* with *Do*? :
An Analysis of *Do*-periphrasis in the Derivations and Evaluations Model of Syntax

an Honors Project submitted by

Jonathan J. Coffee
1815 High Avenue
Jefferson City, Tennessee 37760
(865) 405-7155
jjcoffee@cn.edu

in partial fulfillment for the degree
Bachelor of Arts in Linguistics and English Literature

April 28, 2010

Project Advisors: Dr. Michael Putnam

Dr. Kip Wheeler

© 2010 Jonathan J. Coffee

Table of Contents

1.0 Abstract.....	2
2.0 Introduction.....	3
2.1 <i>Do</i> -Periphrasis.....	3
2.2 An Overview of Generative Syntax	5
2.2.1 Theoretical Linguistics and Syntax	5
2.2.2 E-Language and I-Language	10
2.2.3 Principles and Parameters	13
3.0 The Models and <i>Do</i> -periphrasis.....	16
3.1 The Minimalist Program	16
3.1.1 Introduction.....	16
3.1.2 A Minimalist Approach to <i>Do</i> -periphrasis.....	17
3.2 Optimality Theory.....	20
3.2.1 Introduction.....	20
3.2.2 An Optimality Theory Perspective on <i>Do</i> -periphrasis.....	25
3.3 The Derivations and Evaluations Model.....	33
3.3.1 Introduction.....	33
3.3.2 <i>Do</i> -periphrasis in the Derivations and Evaluations Model	37
4.0 Analysis and Conclusions.....	41
4.1 Regarding the Constraints.....	41
4.2.0 Focus Constructions.....	42
4.2.1 Focus in Optimality Theory	44
4.2.2 Focus in Derivations and Evaluations.....	46
4.2.3 Analysis of Focus Constructions.....	47
4.3 Conclusion	48
Works Cited.....	50

1.0 Abstract

The English language contains two forms of the verb *do*: a lexical verb and an auxiliary verb. Put very simply, *do*-periphrasis refers to the use of *do* as an auxiliary (helping) verb. There are three functions of *do*-periphrasis in English: i) the formation of negation sentences, ii) the formation of interrogatives, and iii) the formation of focus readings. This paper focuses on three models of syntax and how they deal with *do*-periphrasis: the Minimalist Program, Optimality Theory, and the Derivations and Evaluations model. Section 2.2 provides an overview of the historical background into which the three models were born. In his model, Broekhuis claims to unify aspects of two theoretical models of syntax, The Minimalist Program and Optimality Theory. By doing so, Broekhuis claims to create a model that improves the explanatory and descriptive adequacy of each model: i.e., how each model describes the similarities between languages while explaining the differences between them.

This thesis will examine Broekhuis' model of Derivations and Evaluations through analysis of *do*-periphrasis. By examining the phenomenon in light of the Derivations and Evaluations model, this thesis will show that Broekhuis' model lacks significant discussion of focus readings and fails to significantly distinguish itself from Optimality Theory.

2.0 Introduction

2.1 *Do-Periphrasis*

English uses many different auxiliary verbs to assist with relating meaning through language. Put very simply, *do*-periphrasis refers to the use of *do* as an auxiliary (helping) verb. However, before moving ahead with discussion of *do*-periphrasis, it is important to distinguish the two types of the verb *do* in English. The English language contains two forms of the verb *do*: a lexical verb and an auxiliary verb. The lexical *do* is considered a “true” verb, used in sentences like (a) and (b) below.

- a) I do my homework.
- b) I do windows.

Here, *do* is functioning as a true verb that has both subject and object.

The other form of *do* is an auxiliary. Many languages, such as Japanese, Turkish and Finnish, do not use auxiliary verbs, and instead rely on word inflection to illustrate features like tense and case. For example, to say “he did not have the ability to see” in Japanese requires only two words: the subject, “he,” and the verb “see” with suffixes attached to the verb to communicate “did not have the ability to.” These languages are known as synthetic languages.

In contrast to synthetic languages are analytic languages, “which make extensive use of prepositions and auxiliary verbs and depend upon word order to show other relationships. . .” (Baugh 64). While there are still elements of English that would suggest characteristics of synthetic language, Modern English is much more of an analytic language in that it relies heavily on prepositions and auxiliary verbs. One such example is a function known as *do*-periphrasis, to accomplish the formation of negation and interrogative sentences, as seen in (c) and (d) below.

- c) I did not answer the question.
- d) Did you answer the question?

Periphrasis refers to the “use of a combination of words for the expression of some sort of grammatical relationship instead of inflection or agglutination” (Jager 1)¹. It follows then that *do*-periphrasis is simply periphrasis involving some form of the auxiliary verb *do*. The auxiliary *do* is a verb with little to no distinct definition or meaning. In other words, the auxiliary *do* has no lexical marking and is therefore used as a “dummy verb” in combinations with other verbs for meaning.

In English, we also see the use of *do*-periphrasis in sentences with focus reading such as (e) below.²

- e) I did answer the question.

In a sentence such as (e), the inflected auxiliary *do* is associated with the action of answering. While the auxiliary here takes the tense of the verb *answer* to communicate a past act of “answering the question,” a sentence like this would not be interpretable as only a past tense construction; there is additional information included in the sentence. This information is a focus on the past act of “answering the question.”

As Jager explains, “some languages allow a *do*-auxiliary to be associated with a specific function, usually the marking of tense, aspect, and mood, whereas in other languages grammatical features of the clause condition the occurrence of such an auxiliary” (Jager 2). Jager goes on to point out that English has both. Sentences like (e) illustrate the former, while sentences like (c) and (d) illustrate the latter. In sentence (e), we see the association of the

¹ There are many other types of periphrasis in English and in other more typologically diverse languages that this thesis does not discuss. For an overview of examples of such constructions, see Jager.

² These kinds of focus readings containing *do*-support are not covered in Grimshaw’s or Broekhuis’ work. Section 4.

auxiliary with the function of the verb *answer*, giving us *did answer* versus simply the past tense of the verb, *answered*. As seen in (c) and (d), *do*-periphrasis is required to indicate negation and interrogative forms in sentences. Just to review, there are three functions of *do*-periphrasis in English:

- i. The formation of negation sentences.

I did not answer the question.

- ii. The formation of interrogatives.

Did you answer the question?

- iii. The formation of focus readings.

I *did* answer the question.

The Minimalist Program deals directly with *do*-periphrasis in the 1995 book *The Minimalist Program*, in which Chomsky and Lasnik formally define their model of syntax. Grimshaw's 1997 work with Optimality Theory deals almost exclusively with *do*-periphrasis. Many Optimality Theorists consider Grimshaw's research to be the seminal work of *do*-periphrasis in Optimality Theory. Finally, the Derivations and Evaluations model is such a new theory that linguists have done little research in furthering the model as a whole, much less with respect to *do*-periphrasis. While much research has been written on *do*-periphrasis in other fields of linguistics and with regard to many studies of the English language, this paper will focus only on the work relevant to the three models in question.

2.2 An Overview of Generative Syntax

2.2.1 Theoretical Linguistics and Syntax

The following section will provide an overview of the historical background into which the three models were born. All three of the models draw, in some way, on the historical context outlined in this section. While understanding the various models and what they do with *do* does not require this overview, someone unfamiliar with theories of syntax may find it helpful in establishing a context for these various models. This section, while not pertinent to the discussion, may prove helpful.

Many different areas of study within the field of linguistics include phonetics, the study of acoustics and speech articulation; phonology, the study of sound meaning and interpretation; morphology, study of the structure and meaning of lexical units (i.e., words); and several others (Carnie 3-4). This paper focuses on syntax and its connection with other sub-branches of linguistics. While it is customary to associate linguistics with the humanities, syntax is a cognitive science, belonging to a “group of disciplines that all aim for . . . describing and explaining human being’s ability to think [abstractly]” (Carnie 4). As a science, syntax applies scientific methodology to sentence structure. Syntacticians gather data, generalize that data to arrive at hypotheses, then go back and re-evaluate the hypotheses when appropriate (Carnie 7). This scientific study of syntax is known as Generative Grammar.

Generative Grammar is a principle theory of human language that seeks to understand and identify what occurs at the individual cognitive level in the creation, interpretation, and acquisition of language. Syntax concerns itself with the study of an individual’s “ability to combine words together to form grammatical sentences in his native language and to know which sequences of words form grammatical or ungrammatical sentences in his language” (Radford 3). Essentially, “the basic concern [of Generative Grammar] is to determine and characterize the linguistic capacities of particular individuals. . .” (Lasnik 14).

Generative Grammar perceives grammar, or linguistic competence, as a cognitive process and looks for underlying structures or procedures that generate language (Carnie 26). With regard to syntax, the definition of “grammar” is very different from a traditional understanding of the term. The populace defines “grammar” as a set of formal rules that one must adhere to in order to seem literate. However, these rules are not obligatory. One can still communicate without following all of the formal rules of grammar in a given language. These formal rules are the perceived rules of a language.

In contrast, linguists define grammar as the “unconscious knowledge that makes us speakers of language” (Cowper 1). Prescriptive grammar differs from descriptive grammar. Prescriptive grammar assigns a set of formal, perceived rules, while descriptive grammar attempts to identify the underlying rules that make up language. For example, it is a prescriptive rule that dictates that sentence (a) below is grammatically correct while sentence (b) is poorly formed.³ However, a descriptive rule indicates that either sentence is grammatical.

Prescriptive:

- a) Katie and I are going to the store.
- b) *Me and Katie are going to the store.

Descriptive:

- a) Katie and I are going to the store.
- b) Me and Katie are going to the store.

In descriptive grammar, a person’s ability to interpret the meaning of a construction determines grammaticality. For instance, a native-speaker of English reading sentence (b) would still be able

³ Please note that when providing linguistic data, the asterisk (*) before an example indicates a poorly formed (ungrammatical) linguistic construction. In other fields of linguistics, an asterisk may also indicate a hypothetical construction. For the purposes of this paper, the asterisk only indicates an ungrammatical construction.

to understand the information provided by the sentence. Therefore, sentence (b) is descriptively grammatical and well-formed.

Generative grammar concerns itself primarily with descriptive grammar. When it does touch on issues of prescriptive grammar rules, it is usually focused on the relationship between those formal rules and the descriptive grammar rules. This focus is one of the main goals of generative grammar: descriptive adequacy. As Carnie points out, “a theory that accounts for both corpora and native speaker judgments about well-formedness is called a *descriptively adequate* grammar” (24).

One of the main goals of generative grammar is to produce an *explanatorily adequate* theory of human language: that is, a theory that observes, describes, and explains the grammar enabling humans to acquire language. There appears to be at the root of all languages a common set of biological cognitive processes, referred to as Universal Grammar, which governs language acquisition.⁴ This Universal Grammar, or UG, is common to all human language, “since all (normal) human beings are equally capable of acquiring any language” (Cowper 5).

One can better approach UG by examining the arguments for it. Assume for a moment that observable data is the only basis for the acquisition of human language. That is, assume that humans acquire language by memorization and recollection of previously encountered linguistic constructions. Because in any given language an infinite number of sentences can be constructed, it would not make sense that mastery of a particular language could ever occur to the extent it does. Mastery of any language would mean that a speaker had memorized an infinite set of linguistic constructions.

⁴ It is important to note that some scholars, such as Daniel Everett, do not agree with the theory of Universal Grammar. (For more information on this check the articles by Everett mentioned in the Works Cited. However, because the frameworks in this paper all assume UG as a given, we will avoid extensive arguments against UG.

This problem brings into question what it really means to know a language. For example, what is it that enables us to determine that sentence (a) is grammatical while (b) is not?

- a) I saw the swan's beautiful jacket in the thrift store.
- b) *Seen I swan the coat beautiful thrift store in.

Sentence (a) has probably never been produced in the English language before, yet no native speaker of English could successfully argue that the sentence is ungrammatical. The grammaticality of the sentence, therefore, cannot be based on observable data. That is, native speakers of a language have the ability to make grammaticality judgments on data they have never encountered before. To prove this point, Chomsky's 1957 book introduces the famous sentence "Colorless green ideas sleep furiously." Simply put, sentences can be "grammatical but meaningless" (Cook and Newsom 2). A well-formed sentence may not effectively convey significant meaning.

Since native speakers can produce any number of perfectly grammatical sentences and phrases, knowledge of a language must be more than just a recollection of previously encountered linguistic constructs. Linguists refer to this problem as the Logical Problem of Language Acquisition and use it as one of the main arguments for the idea of Universal Grammar (Carnie 19).

In addition to the Logical Problem of Language Acquisition argument, there is the "Underdetermination of the Data" or the "Poverty of the Stimulus" theory. Consider the following data from the first chapter of Andrew Carnie's book, *Syntax: a Generative Introduction*.

- a) Who do you think that Ciaran will question _____ first?
- b) Who do you think _____ Ciaran will question _____ first?

- c) Who do you think _____ will question Seamus first?
- d) *Who do you think that _____ will question Seamus first? (20)

No native speaker of English would question the grammaticality of sentences (a - c). Based on the construction of the previous sentences, it seems to follow that (d) should be grammatical, yet no native English speaker would consider sentence (d) to be grammatical. Here again, the grammaticality judgment cannot come from any observed data; we have never seen this data before. Moreover, if our knowledge of language came from observed data, (d) would be grammatical by association with sentences (a - c). While the grammaticality of the sentences above is not indeterminate (a grammaticality judgment can be made), the data does not offer enough information, or is too “underdeterminate,” for building a grammaticality judgment from the data alone: hence the name, the “Underdetermination of the Stimulus” theory. The judgment must come from some other source.

One solution to this problem is that humans must be born with some knowledge of language grammaticality, or UG. Generative grammar’s attempts to reconstruct, or generate, this set of processes, known as UG, is an attempt to achieve an explanatorily adequate theory of human language. We can attain an understanding of the grammar that enables us to acquire language by striving for this explanatory adequacy in our theory.

2.2.2 E-Language and I-Language

The concept of UG has been central to the study of generative syntax since Chomsky’s seminal work in the 1950s. Most of the breakthroughs in syntactic theory came about because of the drive for a better understanding of UG. This drive led to the theories of E-language and I-language that would ultimately lead to the Principles and Parameters, or P&P, framework.

The main focus of linguistics is “what we know about language and where this knowledge comes from” (Cook and Newsom 4). Language, as human beings experience it, consists of two main aspects. The first is the actual physical aspect of language, i.e., those interpretable things such as sounds and letters. The second is the abstract knowledge of language (or meaning) that we carry in our minds. The role of language is to link meaning to the physical aspects of human communication.

On the surface, language seems to consist of sounds, letters, symbols, and other sensory cues. These physically-experienced objects of language comprise the external aspects of language. Ask most people what it means to study language and they will most likely refer to the physical differences of language: for instance, being able to speak Spanish and English or being able to read hieroglyphics. Consider the sentence below.

a) The apple is green.

Like all of the sentences in this paper, sentence (a) consists of letters, or written symbols, which constitute the external aspects of language. All of these external aspects of language are collectively referred to as E-language (Cook and Newsom 7).

Consider, though, the nature of words and letters. These physical manifestations have the ability to communicate meaning from one human being to another. Nothing inherent in the nature of these objects endows them with the power of meaning. Rather, something in the minds of all human beings allows us to assign meaning to these external objects. If “external” is used to refer to these physical manifestations, it makes sense that “internal” should be used for the abstract thought that motivates the understanding of meaning.

All normal human beings have the capacity to understand meaning. The human mind is able to assign the external aspects of language to abstract concepts and ideas. This capacity is

common to all human beings no matter what “language” they may speak, be it English, Spanish, Japanese, etc. This internal and individual aspect of language is referred to as “I-language” (Lasnik 15).

While all human beings have the capacity for understanding language by assigning meaning to physical aspects, the way the human mind accomplishes this task is mysterious. The internal reaches the external and the external reaches the internal by some mechanism. Linguistics accounts for this with the idea of the “computational system” (Cook and Newsom 5). As Cook and Newsom explain, “The human mind bridges [the gaps between external and internal] via a computation system that relates sounds to meaning in the other” (5). Inside the human mind, the computational system is the mechanism linking the physical aspects of language to the abstract aspects of thought. Where linguistics is concerned, “the complexity of language resides in features of the computational system, primarily the syntax (Cook and Newsom 6).

Syntacticians want to model this computational system and explain the system itself as well as its connections to the external and the internal aspects of language. Essentially, syntax seeks a theory of the computational system. Part of that theory includes the points of contact between the computational system and the external/internal aspects of human language. Chomsky defined these “points of contact” as interfaces (qtd. Cook and Newsom 6). Interfaces represent where human language cognition intersects with thought and with words.

In representations of language, Phonetic Form (PF) is the interface between the computational system and E-language. PF simply means the component of the model where word order and physical aspects of language are expressed. Logical Form (LF) is the title given to the interface between the computational system and I-language. Logical Form (LF) stands for

the component of the model where the speaker interprets or assigns meaning to language. PF and LF can therefore be summarized as the names of the interfaces where the computational system meets sound and meaning, respectively. Figure 2.0 below represents our understanding of the structure of language up to this point.



Figure 2.0 (Adapted from Cook and Newsom Figure 1.6 [7])

This working idea of E-language, I-language, the computational system, and interfaces lays the ground work for the introduction of the Principles and Parameters framework.

2.2.3 Principles and Parameters

The theory of Principles and Parameters (P&P) centers on the individual capacity for language. The theory, therefore, focuses in large part on the computational system. In the P&P model there are two main components of the computational system: the lexicon and the principles of UG (Cook and Newsom 8). That is, in the mind of every speaker of human language two components are essential to the capacity for language: an internal dictionary and the set of underlying principles that govern all human language.

The lexicon is similar to a mental dictionary and essentially contains all of the vocabulary of a given language. The lexicon also contains information about the properties of each lexical item that includes “a representation of the phonological form of each item, a specification of its syntactic category, and its semantic characteristics” (Lasnik 30). For example, with a word like

push, a speaker of English knows (1) how to say the word, (2) that the word is a verb, and (3) that in a sentence the verb requires both a subject, the *pusher*, and an object, the *pushee*.

The other component of the computational system is the knowledge of the principles of UG. As stated before, Universal Grammar establishes the set of underlying principles for all human language. These principles are the same for every speaker of any human language, regardless of whether they speak English, Arabic, or some other language. Language variation must, therefore, be the result of something besides differences in the principles of a language.

Before the introduction of the Principles and Parameters framework, linguists regarded differences between languages as variation between language-specific rules. After the development of the theory of UG, linguists came to understand this theory of language difference as incomplete. The theory was explanatorily adequate to explain the similarities between languages, but not descriptively adequate to account for differences between languages.

This lack of descriptive adequacy led to the development of the idea of parameters. Parameters are an aspect of UG that vary from language to language: word order, for example. All human languages still hold true to the principles of UG but are allowed some variation in the parameters of UG. As Chomsky stated in his 1995 article “Bare Phrase Structure,” what distinguishes one language from another “is not, then, a system of rules, but a set of specifications for parameters in an invariant system of Principles of Universal Grammar (UG)” (qtd. in Cook and Newsom 9).

Incorporating the theories of P&P into our framework from Figure 2.0, a new model of structure of human language would look something very similar to Figure 2.1 below.

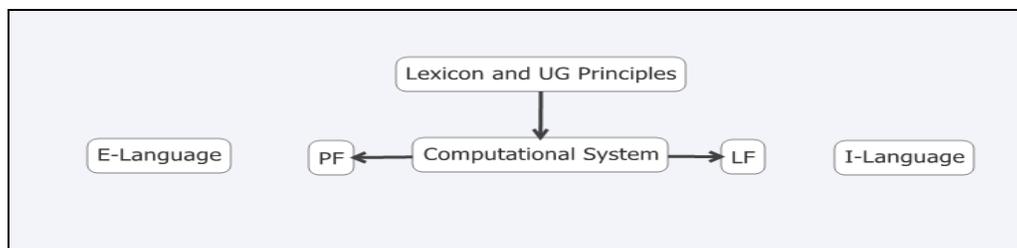


Figure 2.1 (Adapted from Cook and Newsom Figure 1.6 [9])

The lexicon and principles of UG feed the computational system that meets the interfaces between E-language and I-language in PF and LF. This model of the P&P framework will serve as the building block for our understanding of the Minimalist Program.

3.0 The Models and Do-periphrasis

3.1 The Minimalist Program

3.1.1 Introduction

The minimalist program is a theory of Generative Grammar that grew out of the Principles and Parameters movement. Noam Chomsky formally defined this program in 1995 in *The Minimalist Program*. In order to understand the framework proposed by the MP, let us take another look at Figure 2.1.

Concerning the P&P movement, Chomsky points out that “a language consists of two components: a lexicon and a computational system” (168). In the minimalist program, the lexicon feeds into the computational system to create a language-specific set of derivations and structural descriptors (SDs). Derivations are language-specific linguistic forms that allow for creation of new “grammatical” linguistic data while SDs are language-specific linguistic units specifying various aspects of linguistic characteristics and structure. These forms then go into the LF and PF components of the model to produce the external and internal products of language.

Notice that our model has two branches. This bifurcation leads to two outputs: one of internal understanding and representation (LF) and another of external understanding and representation (PF). As Chomsky points out, these two branches account for two different linguistic levels – articulatory-perceptual (A-P) and conceptual-intentional (C-I) (168). A-P corresponds to PF and C-I corresponds to LF. Both of these occur at what Chomsky refers to as “interface levels” (169).

As Broekhuis points out in his summary of the Minimalist Program, the current model of human language in the minimalist tradition looks something approximate to Figure 3.0 below.

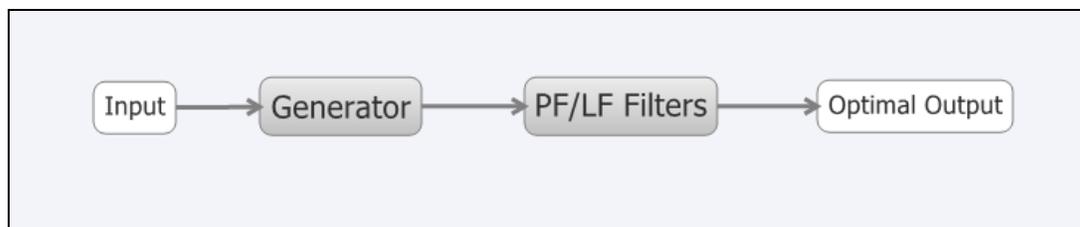


Figure 3.0

In Figure 3.0, the initial input is the lexicon. The lexicon feeds into the computational system (generator) and produces a set of language-specific derivations and SDs that behave as input for the PF and LF filters.

3.1.2 A Minimalist Approach to *Do*-periphrasis

Before moving ahead with the analysis of *do*-periphrasis, we must discuss a typical syntactic representation of a sentence as shown in (1) below.

1) . . . [CP [IP [VP . . .]]] ⁵

For the purposes of this thesis, “sentences” consist of three levels, Clausal Phrase (CP), Inflectional Phrase (IP), and Verb Phrase (VP). These phrases are nested inside of each other, and movement of items between these phrases is responsible for creating variations of utterances.

Important in understanding our discussion of movement between phrase levels are the ideas of head and specifier. The *head* of a phrase is the element that gives the phrase its characteristics. For instance, the head of a verb phrase is a verb. However, in some instances (as in inflectional phrases), the head of a phrase can be phonologically empty or require some other element in the sentence to fill it. The *specifier* is an intermediate position within a phrase level

⁵ This representation of sentence hierarchy and phrase structure is, of course, very simplified. However, for the purposes of this thesis it will be sufficient.

projection: such as *the* in the noun phrase *the cat*. *Cat* is the *head* of the noun phrase while *the* is the *specifier*.

A D-structure representation (before any movement or transformation takes place) for a sentence involving past tense like “John wrote books” looks like (a) below.

a) John (I_{PAST}) write books

In this instance the verb *write* is raised to I to give us *wrote*. Chomsky outlines the basis for this idea of raising in Chapter 2 of *The Minimalist Program* with the assumption that “shorter derivations are always chosen over longer ones” (139).

However, in the case of interrogative and negation clauses, we see instances where raising cannot occur. Consider the following sentence from Chomsky’s *Minimalist Program*.

b) Q John I write books

An S-structure (the final utterance) reading of such a sentence would be “Did John write books?” In this instance we see that here *write* fails to get tense and case from I. Instead, I raises to the Q position and joins with the dummy verb *do* to give us “Did John write books?” (Chomsky 139). Chomsky goes on to explain that in the case where there is no element to be raised to the Q, hereafter referred to as [+wh], English employs *do*-support. In *do*-support a dummy verb, like *do*, is inserted into the modal position. An intermediate step between the D-structure form, as in (b), and the S-structure form “Did John write books?” looks something like (c).

c) [+wh] John did write books.

We see that by the insertion of the dummy verb *do*, I is assigned to the modal position and not the verb *write*. If *write* were able to take I in interrogative forms, we would get the archaic-sounding construction (d) below:

d) *Wrote John books?

Sentences like (c) also answer the question of how MP deals with emphatic reading involving *do*-periphrasis. If sentence (c) were not an interrogative construction, we would get something like (e) below:

e) John did write books.

While this construction is legitimate in modern English, such a construction is used for emphasis. Here, the emphasis is on the I element of the construction. The speaker would be placing emphasis on the fact that John *did* write books. Such a reading could lead to either interpretation that either 1) John wrote books at one time and is no longer writing books, or 2) the fact that John wrote books has been called into question. In this instance, it is easy to see how a *do*-periphrastic construction, as in (e), is used to communicate more information than simple tense and agreement.

To borrow further from Chomsky's discussion of *do*-insertion, the last example of *do*-periphrasis in English involves negation constructions. Sentences (f) and (g) below illustrate both a D-structure and an S-Structure, respectively, of such a construction.

f) John I Neg write books.

g) John did (does) not write books.

h) *John not wrote (writes) books. (140)

Notice that we do not get the construction (h). This is due to a principle in MP known as the Head Movement Constraint (HMC). Lasnik offers the following definition of the HMC.

A general condition . . . is that the target of movement must be the closest possible position, with varying effects depending on the kind of movement

involved. The condition is very strict for head movement which cannot pass over the closest . . . head. (49)

Cowper offers the formal definition of the HMC below:

1. An X^0 may only move into the Y^0 which properly governs it. (138)

Very generally speaking, the HMC stipulates that movement cannot take place in a construction when that movement will skip over other information in the construction. In other words, in a sentence like (h), I cannot move to the verb *write* and leave Neg behind. At the same time, the verb *write* cannot skip over Neg and take I as in sentence (i).

- i) *John wrote not books.⁶

Very generally speaking, with respect to the MP, English uses *do*-periphrasis as a kind of last resort between D- and S-structure. *Do*-periphrasis allows the communication of I in instances where they could not be communicated otherwise, as in interrogative and negation formations. In addition, *do*-insertion allows for emphasis to be placed on I by allowing the inserted auxiliary to take on their features instead of the true verb.

3.2 Optimality Theory

3.2.1 Introduction

In contrast to the MP model of grammar, a separate theory exists within the scope of generative linguistics known as Optimality Theory (hereafter referred to as OT). This theory arose in the early 1990s in response to a “long-standing problem” identified in the field of phonology (McCarthy 1). Phonologists have long known about a phenomenon referred to as a conspiracy. A conspiracy describes a situation in which multiple linguistic rules work together or

⁶ While such constructions may occur in poetry and Early Modern English, the grammaticality judgments here are concerned only with Modern English.

in opposition to each other to describe the same goal or, more simply put, to achieve the same output.

As Zuraw points out, one of the most famous examples of a conspiracy is the one described in 1970 by Charles Kisseberth in which he “identified a ‘conspiracy’ in Yawelmani: rules of vowel insertion and deletion conspire to place every consonant adjacent to a vowel” (Zuraw 4). Essentially, multiple rules interact with each other to produce the same output. According to McCarthy, “when two or more . . . rules are involved in a conspiracy, they directly or indirectly support some constraint on the surface forms” (McCarthy 2). McCarthy further explains that all conspiracies describe some constraint on surface forms, which we can more succinctly refer to as an output constraint (3). OT seeks to provide an adequate theory of these constraints and the model that applies them to human language.

Within OT, there are two types of constraints: faithfulness and markedness constraints. Faithfulness constraints do not allow differences between the input and the output, while markedness constraints change some aspect of the input to show a distinction in the output (McCarthy 13). Essentially, faithfulness constraints require characteristics of the original input in the output; markedness constraints require some kind of identifiable change in the output that shows that it has gone through the system. Examples of each will be provided further into this section.

For the OT model, as we saw with the MP model, the architecture of language begins with an input. The input then has some set of operations that process the input to produce an output (Archangheli 13). As Archangheli points out, “if an operation makes some change in the input, that changed form serves as the input to the next operation” (13). The framework of OT looks very similar to Figure 3.1 below.

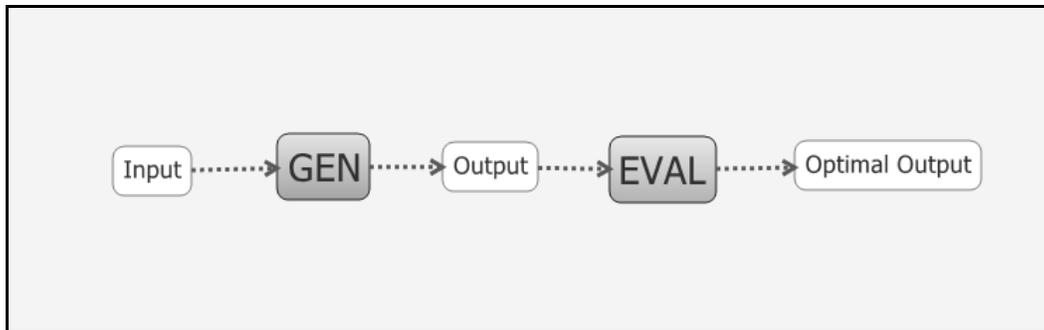


Figure 3.1

The following explanation of the chart above derives from Archangeli’s description of the formal model of OT (14). For any given input, the generator, or GEN, creates a set of possible outputs. As McCarthy explains, “the list of possible outputs supplied by GEN for a given input is called the candidate set for that input” (16). The candidate set is then fed into the evaluation component, or EVAL, in which there is language-specific ranking of the universal set of constraints, or CON. This process produces the most optimal candidate. Further explanation of these components (Input, GEN, EVAL, and CON) will provide a better understanding of the role each plays.

GEN is the first component in this relationship. When an input reaches this component, GEN’s job is to create a set of possible outputs, called a candidate set, for the given input. The candidate set is the output in Figure 3.1 located to the right of the GEN component. This candidate set comprises all possible modifications of a given input. This is not to say, however, that GEN does nothing to the input. The purpose of GEN is to restrict the candidate set to just representations of one given input. Therefore, the candidate set for (Input A) is not going to be the same as the candidate set for (Input B).⁷

⁷ While an example here would provide clarity, research within OT focuses heavily on the EVAL component of the system. A practical example of how GEN restricts candidates is difficult to find. This is one of the issues Broekhuis attempts to resolve by using the GEN component of MP.

Even though GEN does much to restrict a candidate set, the candidate set is still potentially infinite. The candidate set comprises all possible modifications for a given input. This composition is somewhat worrying because the candidate set would potentially contain outputs that would never be optimal in any given language. McCarthy seems to resolve this issue by emphasizing the fact that “the output of GEN isn’t the final output of the grammar” (18). This point is where the functions of EVAL and CON come in.

Different candidates within the candidate set compete for the output. “EVAL’s job is to find the *optimal* candidate [from the candidate list]” through a set of ranked constraints, also known as CON (McCarthy 19). The candidate satisfying the most constraints or more correctly satisfying the highest number of highly ranked constraints is the optimal output. Since “all languages have access to exactly the same set of constraints,” it is in the ranking of these constraints that languages vary (Archangeli 15). Essentially what is created within each language is a hierarchy of constraints in which the higher the constraint is ranked, the less violable it is.

Regarding the computational system there seems to be a kind of paradox at work. The output must show signs of having been through the system, yet must remain close enough to the original input that lexical and semantic interpretation are still possible. Recall again the two types of constraints: faithfulness and markedness constraints. These two types of constraints check that the input remains unchanged and, contrarily, that it undergoes some change so as to provide evidence that it has been through the system. As McCarthy states, markedness constraints affect the output while faithfulness constraints “prohibit differences between input and output” (13).

For an idea of what a markedness constraint could consist of, consider the following examples from Kager:

- a) Vowels must not be nasal.
- b) Syllables must have codas.
- c) Obstruents must not be voiced in coda positions.
- d) Sonorants must be voiced.
- e) Syllables must have onsets.
- f) Obstruents must be voiced after nasals. (9)

Kager points out that markedness constraints are blind to the lexicon and input; they only focus on the output (Kager 9). In other words, markedness constraints are responsible for changing the input to provide evidence that each has been through the system.

Consider now the examples Kager offers for faithfulness constraints.

- a) The output must preserve all segments present in the input.
- b) The output must preserve the linear order of segments in the input.
- c) Output segments must have counterparts in the input.
- d) Output segments and input segments must share values for [voice].

(Kager 10)

As previously stated, faithfulness constraints focus on the relationship between the input and the output. Faithfulness constraints serve two main functions: 1) to preserve lexical contrasts or distinction between various lexical items and 2) to limit the distance between input and output to restrict the amount of change a lexical item can undergo (Kager 10).

It is important here to understand two main ideas concerning constraints. First, constraints are violable, and second, constraints are universal. Recall the discussion regarding the

structure of the computational system of human language. Once an input reaches the EVAL portion of the framework, it passes through a set of universal constraints known as CON (ranked in order of violability). A constraint ranked less violable is said to dominate a constraint that is more violable. The goal of the computational system, then, is to choose the candidate violating the fewest number of less violable constraints, because as McCarthy points out, “constraints are violable in OT, but violation is minimal” (23). The winning candidate is said to be the “optimal” choice.

3.2.2 An Optimality Theory Perspective on *Do*-periphrasis

Now let us look at *do*-periphrasis through the lens of OT. We will begin by examining the same instances of *do*-periphrasis with which the MP dealt. The majority of this discussion will focus on Jane Grimshaw’s 1997 work, “Projection, Heads and Optimality.” The extensive use of this work for this section is mainly due to the fact that many consider this piece to be the seminal work in OT dealing with *do*-periphrasis in English.

Recall that OT focuses mainly on ranking of constraints for its production of grammatical constructions. Grimshaw begins by describing the various constraints involved in an OT analysis of *do*-support. The following chart (adapted from Grimshaw’s) shows the ranking (in English) of the constraints involved in such an analysis:

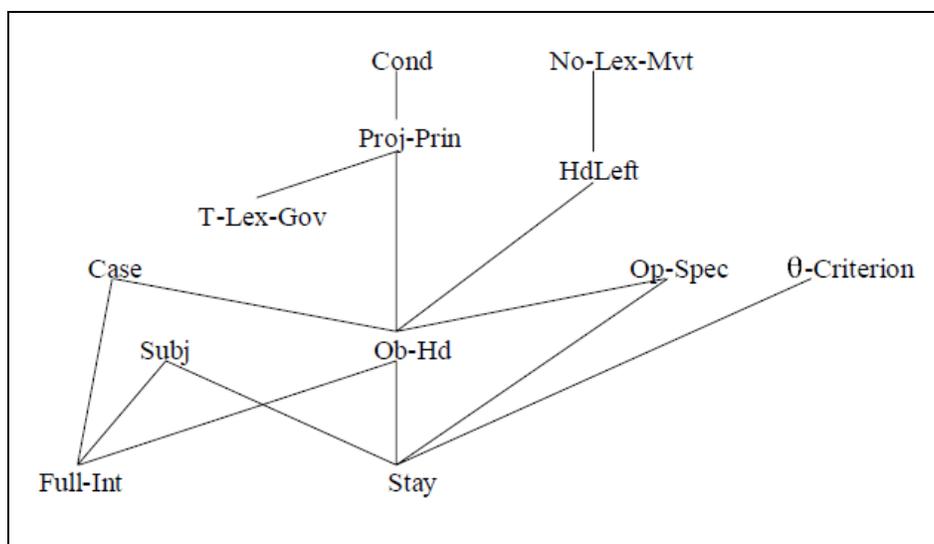


Figure 3.2 Grimshaw Constraint Ranking (Grimshaw 2).

Constraints higher in the chart dominate those lower in the chart. For example *Case* dominates *Ob-Hd* which dominates *Full-Int*. This chart is for reference purposes; constraints will be defined as they are introduced. Grimshaw begins her discussion of *do*-periphrasis by outlining all possible instances of *do*-support in English: “. . . *do* is impossible in (positive) matrix declaratives and subordinate interrogatives, required in matrix interrogatives with no auxiliary, never co-occurs with other auxiliaries and never co-occurs with itself” (Grimshaw 9).

Let us begin by looking at how OT deals with inversion. Inversion is the seeming “switch” of the object and the verb. For example, *what* in sentence (a) below is the object of *eat* yet it is moved to the left of the verb.

- a) What did John eat?
- b) *John ate what?⁸

⁸ This sentence is ungrammatical in the sense of a regular interrogative question: not one involving focus. Note that a grammatical reading involves an emphatic or focus reading.

Grimshaw points out three main constraints that are involved in the process of inversion: STAY, OP-SPEC, and OB-HD (4).

Grimshaw offers the following formal definition of each:

1. ECONOMY OF MOVEMENT (STAY) – trace is not allowed
2. OPERATOR IN SPECIFIER (OP-SPEC) – syntactic operators must be in specifier position
3. OBLIGATORY HEADS (OB-HD) – a projection has a head (1)

STAY is a faithfulness constraint that does not allow movement of any element in the sentence. OP-SPEC, a markedness constraint, requires that operators such as auxiliaries and *wh*-items be moved to a specifier position in interrogative sentences. OB-HD, also a markedness constraint, requires that all projections have heads. Notice in Figure 3.2 that OP-SPEC dominates OB-HEAD while OB-HEAD dominates STAY. OP-SPEC, then, would be the least violable of these three constraints. Sentence (a) violates STAY because OP-SPEC requires movement of *what* to the specifier of CP to satisfy the interrogative (*wh*+) requirement. However, because OP-SPEC dominates STAY, this violation is permissible.

This leads us to our discussion of the distribution of *do* in English. Grimshaw states that *do* contains very little lexical information and differs from other auxiliaries in that *do* lacks semantic content and therefore is not part of the input (10). *Do*, then, is used when no other auxiliary is present yet the constraints require the presence of one.

Let us look again at sentence (a) from above, repeated here as (c) with (d) and (e).

- c) What did John eat? [CP What_i did_j [IP John e_j [VP eat t_i]
- d) *What John ate?. [CP What_i e [IP John [VP ate t_i]
- e) *What John did eat? . . . [CP What_i e [IP John did [VP eat t_i]

Here, Grimshaw introduces the faithfulness constraint: FULL-INT. Grimshaw offers the following formal definition of each:

1. FULL INTERPRETATION (FULL-INT) – lexical conceptual structure is parsed (1)

To paraphrase Kager, FULL-INT does not allow any semantically empty elements in the output (352). For example, *did* in (c) above is semantically empty and therefore violates FULL-INT. Full interpretation requires that all elements of an output be fully interpretable. Because *do* lacks lexical and semantic definition, it is not fully interpretable.

The following Tableau⁹, adapted from Grimshaw (11), illustrates how the various constraints apply to INSTANCES (c) – (e):

	Candidates	OP-SPEC	OB-HD	FULL-INT	STAY
Winner	c) [_{CP} What _i did _j [_{IP} John e _j [_{VP} eat t _i]]			*	**
	d) [_{CP} What _i e [_{IP} John [_{VP} ate t _i]]		*!*		*
	e) [_{CP} What _i e [_{IP} John did [_{VP} eat t _i]]		*!	*	*

Figure 3.3

Because sentence (c) has the least number of violations and has no violations of highly ranked constraints, it is the optimal candidate. *Do* appears in the absence of auxiliaries in interrogatives.

Do is not used in all instances of interrogative questions, however. Take sentences (f) and (g) below. Here again, the data is adapted from Grimshaw (16):

f) Who ate it? [_{VP} Who ate it]

g) *Who did eat it? . . . [_{IP} Who_i did [_{VP} t_i eat it]]¹⁰

⁹ OT refers to constraint tables as tableaux. Asterisks indicate a constraint violation. Exclamation points indicate a violation of a highly ranked constraint; such violations obligatorily rule out candidates.

Notice that *do*-periphrasis is not needed in the example above. Grimshaw solves this by pointing out that OP-SPEC only requires an operator to be in a specifier position. It does not specify any particular specifier position (16). Therefore, in interrogatives where the *wh*-item is in the subject position, none of our constraints regarding *do*-periphrases is violated. This same concept applies to declarative statements, such as (a) and (b) below:

- a) John ate apples. [VP John ate apples.]
 b) *John did eat apples. . . [IP John_i did [VP t_i apples.]

The following Tableau shows how constraints apply to such constructions:

	Candidates	OP-SPEC	OB-HD	FULL-INT	STAY
Winner	c) [VP John ate apples.]				
	d) [IP John _i did [VP t _i apples.]			*!	*

Figure 3.4

Now we will look at the issue of *do*-periphrasis in negation statements. For this discussion, Grimshaw introduces three new constraints: SUBJECT, CASE, and NO-LEX-MVT:

1. SUBJECT (SUBJ) – clauses have subjects
2. CASE-MARKING (CASE) – DPs must be case marked
3. NO MOVEMENT OF A LEXICAL HEAD (NO-LEX-MVT) – a lexical head cannot move (1)

SUBJECT (markedness constraint) requires that a clause must have a subject by requiring that the highest specifier of the clause be occupied (Grimshaw 18). CASE (markedness constraint) requires that a noun-phrase be in a position to receive case (18). NO-LEX-MVT (faithfulness constraint) basically prevents a lexical head from being moved (Kager 352). In the discussion of

¹⁰ Also note that a grammatical reading of (g) requires a focus reading (i.e. an emphatic reading). Grimshaw does not cover issues of focus. Section 4 will shed more light on this issue.

do-periphrasis, this constraint primarily concerns lexical verbs that assign arguments. Recalling again the ranking of our constraints from Figure 3.2, CASE dominates FULL-INT and OB-HD. SUBJECT dominates STAY and FULL-INT. See Figure 3.2 below, shown here as Figure 3.5:

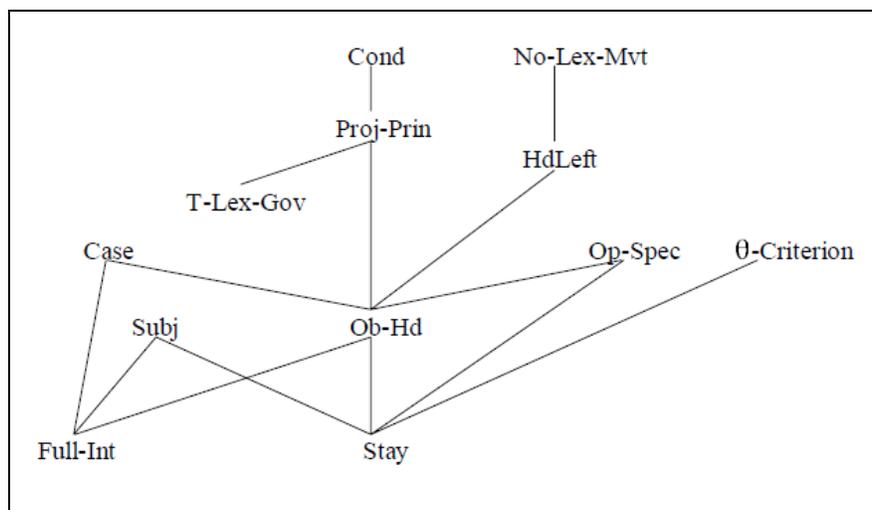


Figure 3.5

Now consider the following examples:

- a) John ate.
- b) John will eat.

In each of these examples, all of our constraints are satisfied. Both (a) and (b) have subjects. Neither sentence shows lexical verb movement. The presence of *not* in a clause, however, causes some problems. Grimshaw holds that *not* is a head with its own projection (18): in a sense we get a “NEG(ative) phrase.” The spec of NEG phrase does not have the ability to assign case, so a subject in the spec position satisfies SUBJECT but fails to satisfy CASE (18-19).

- c) *John not ate. . . . [NEG John_i not [VP t_i ate]]
- d) *Not John ate. . . . [NEG Not [VP John ate]]

Sentence (c) satisfies SUBJECT but not CASE. Sentence (d) satisfies CASE but not SUBJECT. Both constraints must be satisfied since neither one of these constraints dominates the other. SUBJECT and CASE can be satisfied by moving both *John* and *ate* to a higher position but at the expense of violating NO-LEX-MVT (19), as seen in sentence (e) below:

e) *John left not. . . . [John_i ate_j [NEG not[VP t_i t_j]]]

The solution to this problem is the presence of *do* at a higher level projection than the NEG phrase. Sentence (c) above becomes an intermediate step to sentence (f) below:

f) John did not eat. . . [John_i did [NEG t_i not [VP t_i eat]]]

Notice that in this example CASE, SUBJECT, and NO-LEX-MVT are all satisfied. The use of *do*-periphrasis in instances of negations satisfies the constraint ranking. The tableau below, Figure 3.6, shows the OT analysis of *do*-periphrasis with negation in sentences (c) – (f) (yet again adapted from Grimshaw (19)).

Candidate	NO-LEX-MVT	OB-HD	SUBJECT	CASE	FULL-INT	STAY
c) [NEG John _i not [VP t _i ate]				*!		
d) [NEG Not [VP John ate]			*!			
Winner e) [John _i did [NEG t _i not [VP t _i eat]					*	
f) [John _i ate _j [NEG not[VP t _i t _j]	*!					

Figure 3.6

Because sentence (e) has the least number of violations and has no violations of highly ranked constraints, it is the optimal candidate.¹¹

Now consider the construction of yes-no interrogatives such as (a) and (b) below:

a) Will John eat the apple?

¹¹ Note that sentence (f) is not optimal because it has a violation of NO-LEX-MVT, a highly ranked constraint. Since NO-LEX-MVT is more highly ranked than FULL-INT, (e) is the optimal candidate.

b) Did John eat the apple?

Notice that there is no *wh*-subject and no *wh*-object in the sentence that can be moved to a specifier position. Grimshaw solves this problem by adopting the idea that there is “a null operator in the spec of CP” (8). In other words, there is still a *wh*-item in the spec of CP.

However, because there is no *wh*-item to be moved to the specifier of CP, the specifier becomes filled with a null value. Essentially, the *wh*-item is still there, but does not appear in phonological form. Assuming this is the case, the ranking of constraints is still the same and applies in much the same way. Consider the following examples and Tableau:

a) Did John eat apples?

b) *John ate apples?

c) *John did eat apples?

	Candidates	OP-SPEC	OB-HD	FULL-INT	STAY
Winner	a) [_{CP} (wh) Did _j [_{IP} John _i t _j [_{VP} t _i eat apples]]			*	**
	b) [_{CP} (wh) e [_{IP} John _i [_{VP} t _i ate apples]]		*!*		*
	c) [_{CP} (wh) e [_{IP} John _i did [_{VP} t _i eat apples]]		*!	*	*

Figure 3.7

Notice in our Tableau above that OP-SPEC is not violated because of our assumption that a null *wh*-item fills the specifier of CP. In fact, one can argue that OP-SPEC may even require a null *wh*-item. Assuming the specifier of CP is filled, the analysis of *do*-periphrasis in yes-no interrogatives looks very similar to *do*-periphrasis in regular interrogatives. Notice that Figure 3.6 had the exact constraint rankings and violations as Figure 3.3, repeated below as Figure 3.8.

	Candidates	OP-SPEC	OB-HD	FULL-INT	STAY
Winner	c) [_{CP} What _i did _j [_{IP} John e _j [_{VP} eat t _i]]]			*	**
	d) [_{CP} What _i e [_{IP} John [_{VP} ate t _i]]]		*!*		*
	e) [_{CP} What _i e [_{IP} John did [_{VP} eat t _i]]]		*!	*	*

Figure 3.8

The point is that in yes-no questions, OP-SPEC requires a null operator to fill the specifier of CP because of the absence of a *wh*-item in the subject or object positions of the sentence.

3.3 The Derivations and Evaluations Model

3.3.1 Introduction

This discussion now brings us to our introduction of Hans Broekhuis' Derivations and Evaluations model, hereafter referred to as the D&E model. In his introduction to the theory, Broekhuis points out that he developed his theory in keeping with generative grammar's principles of descriptive and explanatory adequacy. In his introductory chapter, Broekhuis states that he developed D&E because it successfully provided explanatory and descriptive adequacy by marrying the generator (GEN) of the Minimalist Program (MP) to the evaluator (EVAL) of Optimality Theory (OT) in what he calls the overarching architecture of syntax (1).

For Broekhuis, both MP and OT agree that accounting for the unchanging and inviolable components of human language (which both attribute to GEN) as well as the variation between various languages (which both attribute to EVAL) is the chief concern. In simplified terms, however, MP seems to focus much more on GEN and OT tends to focus much more on EVAL. In other words, Broekhuis' model argues that MP accounts more for descriptive adequacy, while OT accounts more for explanatory adequacy.

In the D&E model, core grammar (more correctly the computational system of human language (C_{HL}) is used as GEN. With respect to D&E, this accounts for the universal description of human language. The evaluator, on the other hand, comes from the universal set of constraints, CON, of Optimality Theory and is used as the EVAL in the D&E model. CON accounts for the language-specific properties and serves to extend possible grammars to provide descriptive adequacy.

As sections 3.1 and 3.2 point out, both MP and OT have theories of the generator. However, the motivation behind choosing the GEN model of MP is that “the investigation of the generator (C_{HL}) is considered MP’s core business” (Broekhuis 24). The use of core grammar (as MP defines it) accounts for the explanatory adequacy required of a theory because the goal is to achieve a universal model of core grammar by “restricting possible grammars” as much as possible. Of the MP theory of GEN, Broekhuis further explains:

[Within MP,] It is assumed that C_{HL} consists of a small set of operations that are subject to inviolable conditions that are relatively well understood. Perhaps C_{HL} can be reduced to a single merge operation, which has two incarnations, Select (external merge) and Move (internal merge). As a result of this, the output of C_{HL} is also highly restricted. (24)

Broekhuis goes on to explain, as we saw in section 2.3, both MP and OT attribute the inviolable constraints of language to GEN. He then uses this, along with the fact that OT does not offer much in the discourse of GEN, as evidence that MP and OT are complimentary rather than competing theories (25).

For the purposes of D&E, the EVAL of OT provides a much better model because “in [MP] cross-linguistic variation is solely due to differences in lexical specifications, whereas in

[OT] it is rather due to the ranking of the universal constraints” (Broekhuis 28). In the OT model, the EVAL consists of a set of universal constraints, identified as “CON.” The D&E model extends possible grammars by allowing language-specific rankings of the constraints within CON. OT provides the descriptive adequacy of the D&E model because it allows for language-specific phenomena based on rankings of the output of GEN rather than on the properties of lexical items. The result is an optimal output.

By taking what he considers the strong points of the MP and OT models, Broekhuis created the D&E model. The strong suits of the MP model were its consideration of the overarching properties of human language and its ability to produce the set of restrictive output from GEN. OT accounts for the language-specific properties of human language with its system of ranking constraints in EVAL.

Figure 3.9 below provides an overview of the features of the D&E model explained up to this point.

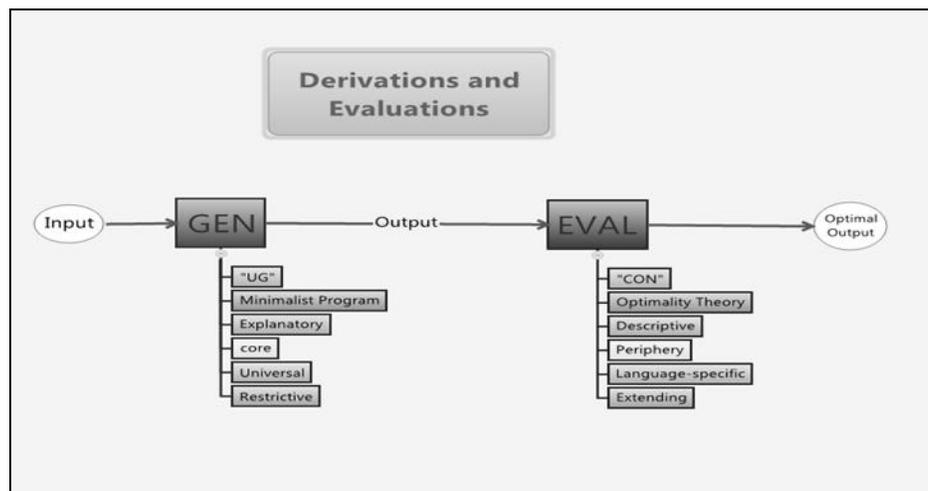


Figure 3.9

In Figure 3.9 the input is the lexicon. The lexicon consists of “all the irregular and memorized parts of language” (Carnie 227). The lexicon feeds into the GEN (taken from the

model in the Minimalist Program). This accounts for the explanatory adequacy required of a theory. GEN produces a highly restricted list of possible grammar constructs. This output then feeds into the EVAL, the set of language-specific ranked constraints that selects the most optimal of the outputs produced by GEN, providing the descriptive adequacy required of a theory. This model produces an optimal output that follows both the core grammar restrictions and the language-specific ranked constraints.

As discussed, the D&E model adopts the Generator (GEN) of the Minimalist Program and the Evaluator (EVAL) of Optimality Theory. There are some slight modifications within each of these that require a closer look before examining the D&E approach to *do*-periphrasis.

Broekhuis suggests that GEN only produces candidates that have applied or not applied principles from UG (32). The set of candidates in the output of GEN consists only of those candidates that have had some combination of UG principles applied to them. Unlike the set of candidates in OT, which is infinite, the number of possible combinations of UG principles is all that makes up the set of candidates in the D&E model (Broekhuis 33).

To explain this more clearly, Broekhuis creates an informal formula for calculating the number of possible candidates:

Consequently, the number of candidates in the candidate set is at most 2^n , where n is the number of operations. . . . C_{HL} thus defines a candidate set that contains a limited number of candidates, and is defined by the optional application of operations Select, Move, and Delete. (32-33)

Broekhuis goes on to explain that using this model for GEN reduces both the number of constraints and the number of constraint rankings (34).

Regarding EVAL, Broekhuis states that there are two types of constraints on the candidate set: C_{HL} constraints and interface constraints (35).¹² C_{HL} constraints further divide into two categories: economy constraints and EPP constraints. Broekhuis defines these two “families” of constraints as follows: “the economy constraints that disfavor the operations of C_{HL} to apply, and EPP constraints that favor them to apply” (58).

3.3.2 *Do*-periphrasis in the Derivations and Evaluations Model

Following the discussion from the preceding sections, the D&E approach to *do*-support is no surprise. Broekhuis acknowledges that *do*-support in interrogatives is directly related to the presence of a CP level projection (252). That is, *do*-support in interrogatives only appears when there is a CP level projection. This fits with both the MP and the OT approach to *do*-support.

The story D&E tells of *do*-support then begins to look a lot like one we have already seen, with the addition of a few constraints. The first is the economy constraint, NO VACUOUS MOVEMENT (NOVACM). Broekhuis offers the following formal definition:

1. NOVACM: Don't apply internal merge when it is invisible in the output. (101)

This constraint “states that movement must have an effect on the linear order of the output” (Broekhuis 36). Simplified, this constraint states that if there is movement, evidence of movement must be visible in the final form. With respect to *do*-support, Broekhuis offers the following observation:

First, note that *do*-support in questions is related to the presence of the CP-level.

If CP is an extended projection of IP . . . *do*-support is required by NOVACM in

¹² Broekhuis' discussion of Interface constraints somewhat unclear. Since our analysis of *do*-periphrasis will not deal directly with these constraints, we will leave them out of our discussion.

order to make I-to-C movement visible as subject-aux inversion [as in (a) below].

. . (252)

(a)[_{CP} What did [you t_I [t_{you} see t_{what}]]]

This is a very similar story to what we observed with inversion within the OT framework. Recall our constraint OB-HD from our OT analysis, which states that all projections have a head. In the sentence above, *wh*- movement creates a CP projection and I moves to C to satisfy the requirement that all projections have heads. Here, it appears that OB-HD and NOVACM are serving the same function. There cannot be movement of I-to-C without something standing in to show movement. Normally, the presence of an auxiliary would accomplish this. However, in the absence of an auxiliary, *do* is used to allow a movement that would otherwise be disallowed.

Broekhuis' next point regarding *do*-periphrasis is an assumption that Grimshaw satisfies with the constraint OP-SPEC:

Secondly, assume that the unvalued feature that triggers *wh*-movement . . . need not be expressed on C but can be expressed on one of the lower heads in the extended projection of the verb. . . . (252)

This looks very similar to the OT OP-SPEC constraint which we recall “only requires an operator to be in a specifier position” (Section 3.2). This allows for constructions like the one in (b) below.

(b)[_{VP} Who ate]

In his last comments on *do*-support, Broekhuis offers the following observation:

Finally, assume that in English the *wh*-feature can be assigned to I. . . . If so, movement of a *wh*-subject into SpecIP will be sufficient both for satisfying the constraint EPP_(PERSON) and for satisfying Full Interpretation . . . (252)

Recall from our discussion of EPP constraints that they require movement to satisfy their goal. Broekhuis offers the following general definition of EPP constraints:

1. $EPP_{(F)}$: Probe F attracts its goal. (39)

For the feature of person then, it follows that $EPP_{(PERSON)}$ is simply movement to satisfy the PERSON requirement of the subject. At this point I will argue that although Broekhuis does not directly define an EPP constraint for *wh*-items, we can assume there is such a constraint.

Namely, (2) below:

2. $EPP_{(WH)}$: Probe WH attracts its goal.

This constraint requires the movement of a *wh*-item to move to specifier position. The D&E model defines Full Interpretation as (3) below:

3. Full Interpretation: all features in the output must receive an interpretation in the articulatory-perceptual or the conceptual-intentional system. (48)

That is, Full Interpretation places a restriction on the output of GEN that requires all features in the output to receive an interpretation in PF or LF, and that no feature present at LF or PF be uninterruptable (Broekhuis 48). $EPP_{(PERSON)}$ and Full Interpretation should look very similar to the OT constraints CASE and FULL-INT. Broekhuis summarizes the interaction of $EPP_{(PERSON)}$ and Full Interpretation as follows:

Consequently, no CP-level need be created and *do*-support will not apply. This results in the IP-structure in (a). However, when we are dealing with a *wh*-object, an additional CP-layer must be created by remerger of I in order to enable *wh*-movement of the object, with the concomitant effect of *do*-support in (b) . . .

- (a) $[_{IP} \text{Who (I) [came } t_{\text{who}}]]$
- (b) $[_{CP} \text{What did } [_{IP} \text{you } t_I [t_{\text{you}} \text{ see } t_{\text{what}}]]]$ (252-53)

Now consider yes-no interrogative constructions in light of the D&E model. Although Broekhuis does not explicitly cover the issue of such constructions, let us assume that he adopts the same assumption that Grimshaw outlines in her analysis. Specifically, such constructions require a null *wh*-item when no *wh*-item is available in the sentence, as in (c) below:

c) Did John eat apples?

Essentially, because a null *wh*-item fills the specifier of CP, our analysis concerning interrogatives is unaffected.

4.0 Analysis and Conclusions

4.1 Regarding the Constraints

In his discussion of the D&E model, Broekhuis argues that his model is significantly different from the OT model. He argues that his adoption of Full Interpretation suggests there are no faithfulness constraints in D&E by requiring “(i) all elements in the input to also be present in the output, and (ii) all elements in the output to also be present in the input” (49). Broekhuis argues that syntax satisfies these requirements by definition. He explains that the D&E framework differs from OT in which “interactions of faithfulness constraints and so-called markedness constraints, which place additional restrictions on the output of the generator, constitute the core of the system” (49). However, a review of the definitions of faithfulness and markedness constraints in OT and economy and EPP constraints in D&E reveal close similarities, as shown in Figure 4.0

Optimality Theory	
Faithfulness Constraints (FAITH)	<i>Two main functions: 1) to preserve lexical contrasts, or distinction between various lexical items, and 2) to limit the distance between input and output to restrict the amount of change a lexical item can undergo (Kager 10).</i>
Markedness Constraints (MARK)	<i>Responsible for changing the input to provide evidence that it has been through the system</i>
Derivations & Evaluation	
Economy Constraints - (ECON)	<i>Disfavor the operations of C_{HL} to apply</i>
EPP Constraints – (EPP)	<i>Favor the operations of C_{HL} to apply</i>

Figure 4.0 OT and D&E Constraint Definitions

Notice the similarities between faithfulness constraints (FAITH) and economy constraints (ECON). By disfavoring the operations of C_{HL} to take place, ECON is essentially preserving

aspects of the input in the output by limiting the amount of change made to the input. ECON, like FAITH, requires the output to maintain some aspect of the input in the output. A similar relationship occurs between markedness constraints (MARK) and EPP constraints (EPP). EPP favor the operations of C_{HL} that change the input. By allowing requiring changes to the input, EPP is in effect doing the job of a markedness constraint. It appears, then, that ECON and EPP are only renamings of FAITH and MARK, respectively. It cannot be said that faithfulness and markedness constraints do not exist in the D&E model, since, by definition, economy and EPP constraints effectively play the same role. In this respect I agree with Sells' conclusions regarding various OT syntax models that "a 'lexicalist view' of faithfulness in syntax should not be dispensed with" (100).

4.2.0 Focus Constructions

Now, let us approach the issue of focus readings regarding *do*-periphrasis. So far the analyses of *do*-periphrasis have skirted the issue of the grammaticality of such readings. Neither Grimshaw nor Broekhuis approach this issue in their respective works. This section will show how the addition of a few new constraints to each of these frameworks (namely OT and D&E) can augment them to account for such constructions.

Consider the examples below:

- a) What did she say?
- b) Did John go?
- c) John did go.

Note that the intended reading for each of these is a focus reading in which *did* is phonologically emphasized. In these instances, such constructions are grammatical. This causes

problems for our models which do not account for such data. The key is sentence (c). Recall that in OT, the following grammaticality judgments occur:

a) *John did eat. . . . [IP John_i did [VP t_i eat]

b) John ate. [VP John ate]

Notice that the construction “John did eat” is ungrammatical in this instance. However, there are instances in which “John did eat” can be grammatical. In these *focus* readings, emphasis is on the fact that *John did* in fact *eat*, instead of merely communicating the fact that *John ate*. This realization gives us the grammaticality judgments in (c) and (d) below:

c) John ate. (normal reading)

d) John did eat. (focus reading)

Notice that both constructions are grammatical yet appear to have the same sentence structure. In his 2004 article, “Locality and Periphery,” Rizzi solves this problem by suggesting that while the two have the same word order, the structure of each is different. Rizzi proposes a phrase level projection between CP and IP that shows up in focus constructions.¹³ That is, between CP and IP there is a Focus Phrase projection or FP as seen in (1) below.

1) [CP [FP [IP . . .

This FP only occurs in instances of focus constructions. Therefore, the analysis from this point on will only deal with such focus constructions. With this new phrase level projection, let us now reexamine both the OT and the D&E set of constraints on *do*-periphrasis constructions. Recall the following constraint rankings for each of the frameworks:

¹³ While Rizzi’s solution to this issue is much more complicated, this thesis will adopt a simplified version for clarity.

OT		D&E	
1)	OP-SPEC	1)	EPP _(WH)
2)	OB-HD	2)	NOVACM
3)	SUBJECT	3)	FULL-INT
4)	FULL-INT		

Figure 4.1

Because of the addition of a phrase level projection and a new consideration for focus constructions, both the OT model and the D&E model require an additional constraint and subsequent ranking. I propose two corresponding constraints FOC-HD and EPP_(FOCUS) for OT and D&E, respectively, with the following formal definitions and rankings:

- 1) Focus Head (FOC-HD) – A Focus Phrase has a head.
- 2) EPP_(FOCUS) – Probe Focus attracts its goal.

OT		D&E	
1)	FOC-HD	1)	EPP _(FOCUS)
2)	OP-SPEC	2)	EPP _(WH)
3)	OB-HD	3)	NOVACM
4)	SUBJECT	4)	FULL-INT

Figure 4.2

With these constraints and rankings in place, let us look at each framework individually to see how these constraints work with focus constructions.

4.2.1 Focus in Optimality Theory

Consider the interrogative constructions and their tableau below:

- a) What did she say? [FP What did [IP she t_{did} [VP t_{she} say t_{what}]]]
- b) What did she say? [FP What e [IP t_{what} did[VP she say t_{what}]]]
- c) Did she say what? [FP e did [IP t_{did} [VP she say what]]]

	Candidate	FOC-HD	OP-SPEC	OB-HD	SUBJECT
Winner	a) [FP What did [IP she t _{did} [VP t _{she} say t _{what}]]]				*
	b) [FP What e [IP t _{what} did[VP she say t _{what}]]]	*!		*	*
	c) [FP e did [IP t _{did} [VP she say what]]]		*!		*

Figure 4.3

FOC-HD requires that FP have a head while OP-SPEC requires that *wh*-items fill the specifier of FP. These requirements explain why (a) is the winner. Also note that in interrogative focus constructions, such as “Did John go,” the same constraints will apply since a null *wh*-item satisfies OP-SPEC.

Now consider the following declarative focus constructions and their tableau:

- a) John did go. [FP John did [IP t t [VP t go]]]
- b) *John did go [FP e [IP John did [VP t go]]]
- c) *Did John go. [FP Did_i [IP t_i [VP John go]]]¹⁴

	Candidate	FOC-HD	OB-HD	SUBJECT
Winner	a) [FP John _i did _j [IP t _i t _j [VP t _i go]]]			
	b) [FP e [IP John did [VP t go]]]	*!		*
	c) [FP Did [IP t [VP John go]]]			*

Figure 4.4

Candidate (a) has no violations of constraints and is, therefore, the optimal candidate.

Note that (b) violates FOC-HD which requires that FP have a head. (c) violates SUBJECT

¹⁴ Note: this is not an interrogative.

because *John* did not move to the specifier of CP (which is empty because this is not an interrogative construction with a *wh*-item in the specifier of CP).

4.2.2 Focus in Derivations and Evaluations

Consider the interrogative constructions and their tableau below:

d) What did she say? [FP What did [IP she t_{did} [VP t_{she} say t_{what}]]]

e) What did she say? [FP What e [IP t_{what} did[VP she say t_{what}]]]

f) Did she say what? [FP e did [IP t_{did} [VP she say what]]]

	Candidate	EPP _(FOCUS)	EPP _(WH)	NOVACM
Winner	d) [FP What did [IP she t _{did} [VP t _{she} say t _{what}]]]			
	e) [FP What e [IP t _{what} did[VP she say t _{what}]]]	*!		*
	f) [FP e did [IP t _{did} [VP she say what]]]		*!	

Figure 4.6

EPP_(FOCUS) requires that FP have a head while EPP_(WH) requires that *wh*-items fill the specifier of FP. These requirements explain why (d) is the winner. Also note that in interrogative focus constructions, such as “Did John go,” our same constraints will apply since a null *wh*-item satisfies EPP_(WH). This explanation looks very similar to the OT data in Section 4.2.1.

Now consider the following declarative focus constructions and their tableau:

d) John did go. [FP John did [IP t t [VP t go]]]

e) *John did go [FP e [IP John did [VP t go]]]

f) *Did John go. [FP Did_i [IP t_i [VP John go]]]¹⁵

¹⁵ Note: this is not an interrogative.

As yet, our model of D&E does not have a constraint that requires that *John* move to the specifier of CP. OT solved this with the constraint SUBJECT. This problem brings us to the introduction of a new constraint $EPP_{(NOM)}$ formally defined below:

- 1) $EPP_{(NOM)}$ – Probe Nom attracts its goal.

Essentially, this constraint requires that noun phrases with nominative case move to the specifier of the highest projection. In focus constructions, that specifier is the specifier of FP. This constraint is ranked lower than $EPP_{(WH)}$ so that nominatives do not move to the specifier of CP in interrogative constructions. With this new constraint, consider the following Tableau:

	Candidate	$EPP_{(FOCUS)}$	$EPP_{(WH)}$	$EPP_{(NOM)}$	NOVACM
Winner	d) $[_{FP} \text{John}_i \text{did}_j [_{IP} t_i t_j [_{VP} t_i \text{go}]]]$				
	e) $[_{FP} e [_{IP} \text{John did} [_{VP} t \text{go}]]]$	*!		*	*
	f) $[_{FP} \text{Did} [_{IP} t [_{VP} \text{John go}]]]$			*	

Figure 4.6

Candidate (d) has no violations of constraints and is, therefore, the optimal candidate. Note that (e) violates $EPP_{(FOCUS)}$ which requires that FP have a head. (f) violates $EPP_{(NOM)}$ because *John* did not move to the specifier of CP (which is empty because this is not an interrogative construction with a *wh*-item in the specifier of CP).

4.2.3 Analysis of Focus Constructions

As evidenced by the analysis of *do*-periphrasis, the production of grammatical constructions relies on the interaction of constraints within each model. The fact that the constraints Broekhuis proposes in his analysis of *do*-support correspond directly to constraints already proposed by Grimshaw does little to support Broekhuis' argument that D&E differs significantly from OT. If there is an advantage to the D&E model, it lies in the explanatory

adequacy of the model: namely, exploring GEN. However, Broekhuis seems to spend little time discussing the significance of the generator and focuses instead on the descriptive aspect of his model. The analysis of *do*-periphrasis in the D&E model suggests that as far as descriptive adequacy is concerned, the D&E model does not offer strong conceptual tools that separate it from a ‘purely’ OT-model.

4.3 Conclusion

This thesis has illustrated how the Derivations and Evaluations model deals with *do*-periphrasis. From the analysis presented in this paper, it cannot be said that faithfulness and markedness constraints do not exist in the D&E model. Therefore, the model does not significantly differ from Optimality Theory in its explanatory adequacy of syntax. Furthermore, this thesis illustrated how, with the addition of a few rules, the Derivations and Evaluations model can be augmented to account for focus constructions using *do*-periphrasis. This conclusion may seem, on the surface, a rather impotent conclusion with little purpose. However, there are moral/ethical and practical motives for the significance of this analysis.

Every model of syntax seeks to better understand human language. From a moral/ethical standpoint, we pursue knowledge for knowledge sake. In other words, we move forward by furthering our understanding of the world around us and improving our ability to share that understanding across academic fields and across time. Regarding language, we should constantly be searching for better explanations and understanding of how human beings communicate with one another.

However, the “knowledge for knowledge’s sake” argument does not appeal to all scholars. There are also practical reasons for furthering our understanding of human language. Syntax is a way of modeling the functions language carries out in our everyday lives. These models often play a key role in the fields of applied linguistics. Instructors of foreign language teach their students new languages by illustrating the similarities between two seemingly different languages. Furthermore, finding better ways of modeling language can prove effective in improving technology. As the world becomes more and more technologically advanced, improving the way humans and technology interact will improve quality of life and the progress of future development. Essentially, modeling human language provides a meta-language (language that describes language) that establishes a common ground for exchanging new concepts and insights regarding human language.

With all of this in mind, research that investigates current models of human language can provide insights for improving our ability to model language. This thesis which analyzes the D&E model underscores the weaknesses of the model and provides an account of the model’s shortcomings. This account guides the way for future research. In this respect, it is not in vain that we strive to reach the goal of a more accurate and adequate understanding of human language. With respect to the D&E model, this is, in fact, what our analysis of *do*-periphrasis “does.”

Works Cited

- Archangheli, Diana. "Optimality Theory: An Introduction to Linguistics in the 1990s." Ed. Diana Archangheli and D. Terence Langendoen. *Optimality Theory: An Overview*. Malden, MA: Blackwell, 1997. 1-32. Print.
- Baugh, Albert C., and Thomas Cable. *A History of the English Language*. 5th ed. Upper Saddle River, NJ: Prentice Hall, 2002. Print.
- Broekhuis, Hans. *Derivations and Evaluations*. New York: Mouton de Gruyter, 2008. Print.
- Carnie, Andrew. *Syntax: a Generative Introduction*. 2nd ed. Malden, MA: Blackwell, 2007. Print.
- Chomsky, Noam. *The Minimalist Program*. Cambridge, MA: MIT Press, 1995. Print.
- Cook, V. J., and Mark Newsom. *Chomsky's Universal Grammar*. 3rd ed. Malden, MA: Blackwell, 2007. Print.
- Cowper, Elizabeth. *A Concise Introduction to Syntactic Theory*. Chicago: U of Chicago P, 1992. Print.
- Grimshaw, Jane. "Projection, Heads, and Optimality." *Linguistic Inquiry* 28. 373-422. 1997. PDF file.
- Jager, Andreas. "Grammaticalization Paths of Periphrastic 'Do'-constructions." Antwerp: U of Antwerp, 2007. PDF file.
- Kager, Rene. *Optimality Theory*. New York: Cambridge UP, 1999. Print.
- Lasnik, Howard. "The Theory of Principles and Parameters." Noam Chomsky. *The Minimalist Program*. Cambridge, MA: MIT, 1995. 13-127. Print.
- McCarthy, John J. *Doing Optimality Theory*. Malden, MA: Blackwell, 2008. Print.

- McCarthy John J., and Alan Prince. "Faithfulness and Reduplicative Identity." *University of Massachusetts Occasional Papers in Linguistics 18: Papers in Optimality Theory*. Amherst, MA: U of Massachusetts P, 1995. 1-126, PDF file.
- Pulleybank, Douglas. "Optimality Theory and Features." Ed. Diana Archangheli and D. Terence Langendoen. *Optimality Theory: An Overview*. Malden, MA: Blackwell, 1997. 59-101. Print.
- Radford, Andrew. *Transformational Syntax*. New York: Cambridge UP, 1981. Print.
- Rizzi, Luigi. "Locality and Periphery." Ed. Adriana Belletti. *Structures and Beyond: The Cartography of Syntactic Structures*. Vol 3. New York: Oxford UP, 2004. Microsoft Word file.
- Sells, Peter. "The Input and Faithfulness in OT Syntax." Ed. Jennifer Spenader, Anders Eriksson, and Oesten Dahl. *Proceedings of the Workshop on Variation within Optimality Theory*. Stockholm: U of Stockholm, 2003. 93-101. PDF file.
- Zuraw, Kie. "Optimality Theory in Linguistics." Ed. Michael Arbib. *Handbook of Brain Theory and Neural Networks*. 2nd ed. Cambridge, MA: MIT Press, 2003. 819-22. PDF file.