March 1992

Ellipsis and Discourse (Dissertation Proposal)

Daniel Hardt
University of Pennsylvania

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-92-21.

Ellipsis and Discourse (Dissertation Proposal)

Abstract

In human discourse there is much that is communicated without being explicitly stated. The grammar of natural language provides a broad array of mechanisms for such implicit communication. One example of this is verb phrase ellipsis, in which a verb phrase is elided, its position marked only by an auxiliary verb. Such elliptical constructions are generally easily and unambiguously understood. In this proposal I will attempt to explain how this is accomplished.

Comments

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-92-21.

Ellipsis and Discourse
(Dissertation Proposal)

MS-CIS-92-21
LINC LAB 216

Daniel Hardt

University of Pennsylvania
School of Engineering and Applied Science
Computer and Information Science Department
Philadelphia, PA 19104-6389

March 1992

Ellipsis and Discourse

Dissertation Proposal

Daniel Hardt

November 8, 1991

Chapter 1

Introduction

In human discourse there is much that is communicated without being explicitly stated. The grammar of natural language provides a broad array of mechanisms for such implicit communication. One example of this is verb phrase ellipsis, in which a verb phrase is elided, its position marked only by an auxiliary verb. Such elliptical constructions are generally easily and unambiguously understood. In this proposal I will attempt to explain how this is accomplished.

To understand an elliptical expression it is necessary to recover the missing material from surrounding context. This can be divided into two subproblems: first, it is necessary to locate an antecedent expression. Second, a method of reconstructing the antecedent expression at the ellipsis site is required. In the case of VP ellipsis, the first problem has been virtually ignored. The second problem has received a great deal of attention in the linguistics literature. It has been generally agreed that the method of reconstruction involves making an identical copy of the antecedent at the ellipsis site. Attention has been focused on the level of representation at which an identity constraint is to be defined. Broadly speaking, the issue is whether reconstruction is defined in terms of the form or the meaning of the antecedent.

Existing accounts of VP ellipsis all define reconstruction by reference to the form of the antecedent; some have claimed surface syntax to be the relevant level, while other accounts rely on syntactic aspects of the meaning representation language. In this proposal, I will pursue the hypothesis that VP ellipsis is reconstructed at the level of meaning. I will develop this hypothesis using the notion of a discourse model. In particular, I will assume that, as part of the process of understanding a discourse, one builds a mental model, including semantic representations of the most salient objects in the discourse. It is widely accepted that pronominal anaphora is to be explained in terms of a discourse model; a pronoun is interpreted by selecting a salient entity in the discourse model as the antecedent or referent of the pronoun. VP ellipsis, I suggest, is to be handled in much the same way; semantic representations of VP's are stored in a discourse model, as potential antecedents for VP ellipsis.

The meaning of a VP in this approach is defined as a three place relation on a property, an input discourse model, and an output discourse model. This means that a VP expresses a certain property only relative to a particular discourse context. It has long been recognized that semantic representation involves this sort of reference to context. In Montague's Universal Grammar [29], the meanings of sentences and all subexpressions have a specific context parameter. For example, VP-meanings are defined there as functions from contexts to properties. Since Montague, the elaboration of utterance context and its interaction with interpretation has been an important topic in logic and linguistics. The development of Discourse Representation Theory (DRT) can be seen as a sustained argument for treating meanings as relations on contexts. This is most clear in compositional approaches to DRT, such as Dynamic Predicate Logic. The Incremental Interpretation System is a computational implementation incorporating this relational approach to semantic representations. I will show that my approach to VP ellipsis can be computationally implemented by means of some simple extensions to this system.

In my dissertation, I propose to explore the consequences of the hypothesis that VP ellipsis is resolved at the level of meaning. Given that meanings are relations on contexts, one consequence is the following: any context-dependent elements within the antecedent VP will be evaluated independently in the antecedent and the target.[1] This prediction differs sharply from standard accounts of VP Ellipsis, and I will give evidence that it is correct. More generally, there is evidence that VP ellipsis must be resolved at the meaning level, because reconstruction appears to require inference and other semantic relations.

In what follows, I begin with a survey of some alternative accounts of VP ellipsis. All of these accounts define reconstruction in terms of the form, rather the the meaning, of the antecedent. Each successive account further complicates the syntactic representation of the antecedent or the mechanisms for manipulating them. I show that there is a body of evidence that remains unaccounted for, and I suggest that a meaning level account can better explain the data, and allows one to remove complexities imposed in the syntax. In Chapter 3, I describe a computational model of a meaning level account. This model is based on the Incremental Interpretation System of Pereira and Pollack [32]. Chapter 4 is devoted to proposed work: a computer implementation of the model is envisioned, as well as plans for testing the model on several hundred naturally occurring examples.

[1]The "target" is the reconstructed elliptical VP.

Chapter 2

Background

2.1 Form and Meaning

VP ellipsis predicates the same thing of two subjects. At least, this is assumed in most accounts, in which the fundamental constraint is an identity relation between antecedent and target. The major issue has been to determine the level of representation at which this identity relation is to be defined. All existing proposals have defined this relation in terms of various aspects of the form of the antecedent. I will suggest it is meaning rather than form that is relevant for VP ellipsis.

A fundamental problem for any identity condition is the phenomenon of "sloppy identity": a pronoun can under certain circumstances switch its referent from antecedent to target. In the accounts of Sag and Williams this phenomenon is assimilated to the bound variable/free variable distinction; sloppy identity is possible only with respect to bound variable pronouns. The possibility of sloppy identity is thus derived from an independently motivated fact about pronouns in the antecedent. Problems arise concerning exactly what sort of fact this is; in particular, attempts to define a bound variable [35] in terms of syntactic configurations such as c-command remain controversial.

A more serious problem arises with examples, such as "cascaded ellipsis", which would require a single pronoun to be both a bound and free variable to allow the desired reading. In response to this, Dalrymple, Shieber and Pereira argue that a backward-looking "matching" mechanism is required. Here, the bound/free variable distinction is still fundamental, but it is no longer a fact about the antecedent, but rather, arises in the backward-looking matching process. Another approach, due to Fiengo and May, attempts to retain the bound/free distinction as a syntactic fact, adding an additional distinction, dependent/independent. Another account, that of Prüst et al, is motivated by examples involving quantifier bindings, and introduces a notion of "structural parallelism" that is explicitly defined in terms of syntactic facts.

In all the above accounts, it is the form of the antecedent which is essential to reconstruction. In response to specific counterexamples, the proposal has been either to complicate the syntactic representations of these forms or complicate the mechanisms manipulating them. Thus Sag and Williams appeal to the bound/free variable distinction, Fiengo and May introduce an additional dependent/independent distinction, Prüst et al introduce a notion of "structural parallelism", and Dalrymple et al introduce a backward-looking "matching" mechanism. There are still a wide range of examples that not accounted for by any of these additional mechanisms and distinctions. I will argue that a meaning level approach can handle these examples, making it possible to dispense with all of the above distinctions.

2.2 Reconstruction Based on Form

2.2.1 Sag and Williams

The "Identity of Logical Form" theory was proposed independently by Sag [38] and Williams [42]. A basic principle in this account is the Derived Verb Phrase rule [30], which allows a VP to be represented at Logical Form (LF) as a lambda expression in which the subject is lambda-abstracted. Given this representation, an appropriate identity condition follows from the lambda calculus itself: this is the notion of an ***alphabetic variant***. Two lambda expressions are alphabetic variants if they differ at most in the naming of bound variables. Applied to VP ellipsis, this condition requires that the antecedent and target VP's must match exactly in the names of any free variables.

With respect to pronouns, the LF theory derives the following rules from the alphabetic variance condition:

1. If a pronoun is coreferential with the subject in the antecedent, it can retain the same referent in the target, or its referent can switch to the new subject.

2. Pronouns that do not corefer with the subject in the antecedent cannot switch reference in the target.

3. If a pronoun is bound by some operator O outside the VP in the antecedent, it must be bound by O in the target as well.

This LF theory successfully handles examples like the following, involving referential pronouns.

(1) John$_i$ saw him$_j$. Bill$_k$ did too.

(2) John$_i$ thinks he$_i$ is smart. Bill$_j$ does too.

In (2), the only possible reading is that Bill saw him$_j$. This follows from Rule 2 above, since him$_j$ is not coreferential with the subject. In (3), exactly two readings are possible: Bill might think he$_i$ or he$_j$ is smart. Again this follows from the theory (Rule 1), since he$_i$ is coreferential with the subject. The possibility of "sloppy identity" is explained in terms of the bound/free variable distinction; only pronouns that can be construed as bound variables within the VP (i.e., subject-bound) permit sloppy identity.

2.2.2 Dalrymple, Shieber, and Pereira

The Sag/Williams account relies on the bound/free variable distinction to characterize the class of possible readings. Dalrymple, Shieber, and Pereira [8] show that, regardless of how one defines variable binding, it is does not provide an explanation for VP ellipsis, if one regards the bound/free distinction as a fact about the antecedent expression. Consider the following example:

(3) i. John realizes that he is a fool, but
ii. Bill does not, even though
iii. even though his wife does.

The relevant reading is that in which John realizes John is a fool, Bill does not realizes Bill is a fool, and Bill's wife does realize Bill is a fool. It is not possible to have three copies of the same VP "realizes he's a fool" to permit this reading, regardless of whether "he" is a bound or a free pronoun. If it is bound, we get the reading in (iii.) that Bill's wife realizes that Bill's wife is a fool. If it is free, we get the reading in (ii.) that Bill doesn't realize that John is a fool.

Dalrymple, Shieber, and Pereira solve this problem by proposing a backwards-looking mechanism, in which the antecedent is determined only when an elliptical VP is reached. This is done by solving an equation, in which the antecedent is equated to a expression in which a second order variable is applied to the subject.

To resolve the ellipsis in (ii.) the equation is

P(John) = realize(John, fool(John))

Intuitively, the question is, what property could have been applied to John, to produce the proposition in (i). The equation is solved by a process of second order matching, which is defined in terms of lambda calculus derivations. Two possible solutions are:

P = λx. realize(x,fool(x))

Q = λx. realize(x, fool($John$))

Solution P gives the desired reading for (ii), i.e., P applied to Bill gives the reading

realize(Bill, fool(Bill))

Next, to resolve the ellipsis in (iii.), we solve the equation

P(Bill) = realize(Bill, fool(Bill))

Again there are two possible solutions:

P′ = λx. realize(x,fool(x))

Q′ = λx. realize(x, fool($Bill$))

Here, we select Q′, which applied to Bill's mother, produces:

realize(Bill's mother, fool(Bill))

Use of the matching mechanisms allows a solution to this example because the bound/free variable distinction is not a fact about (ii.), but a fact about particular solutions produced by the matching mechanism.

2.2.3 Fiengo and May

Unlike Dalrymple, Shieber, and Pereira, Fiengo and May ([9], [10]) retain the standard bound/free distinction as a fact about the antecedent. They introduce another distinction into the syntax: that of dependent/independent. Dependent variables they term α occurrences, and dependent variables are β occurrences. Ellipsis is governed by an LF identity condition similar to that imposed by Sag and Williams; the main modification is simply that syntactic representations are now more complicated, representing dependencies as well as binding.

The Fiengo and May approach does not accept DSP's use of a backward-looking mechanism. Thus, they have problems dealing with the "cascaded ellipsis" examples, which require a pronoun to be both bound and free. They claim that in examples such as:

(4) i. John thinks he's a fool.
ii. Bill does too, although
iii. his mother doesn't.

the sloppy reading is understood as a β occurrence reconstructed in (iii.) bound to the pronoun "his". But that would rule out examples like the following:

(5) i. John thinks he's ready to play.
ii. Bill does too, although
iii. the coach doesn't.

on the reading that the coach doesn't think Bill is ready to play.

2.2.4 Prüst et al

Prüst et al([33],[34]) have also pointed out cases where a pronoun is bound to different binders in antecedent and target. They propose that a pronoun can be bound by two different quantifiers in antecedent and target, as long as the two quantifiers are in "structurally parallel" positions. However, example (7) indicates that the two quantifiers need not be in structurally parallel positions. Indeed, example (8) shows that there is no requirement for a corresponding quantifier at all.

(6) Almost every boy_i in the class hope Mary will **ask him_i out**, but I know there are a few boys_j who hope that she won't. [ask him_j out]

(7) Every boy_i in Mrs. Smith's class hoped she would **pass him_i**. In John's_j case, I think she will. [pass him_j]

2.3 Reconstruction Based on Meaning

2.3.1 Dispensability of Logical Form

It is argued by Partee and Bach [31] that the identity condition imposed by Sag and William violates Montague's requirement that logical form must be "dispensable". They show that it involves essential reference to the syntax of logical form expressions, distinguishing, for example, between two token occurrences of the same quantifier. In this sense, the Sag/Williams account defines reconstruction in terms of the form, rather the the meaning, of the antecedent. The other accounts mentioned above also involve essential reference to syntactic aspects of the meaning representation language.

Partee and Bach argue that such a restriction is necessary to rule out examples such as

(8) No man_0 believes that Mary loves him_0. *But she does. [love him_0]

However, Lappin [25] argues that the alphabetic variance condition is too strong. He gives the following example:

(9) Every boy in Bill's $class_1$ wants Mary to kiss him_1, but none of $them_2$ believes that she will [kiss him_2].

This violates alphabetic variance because the antecedent verb contains a bound variable (him_1) that is bound by different operator in the target. Lappin suggests a modified condition, in which such a variable can be bound by two different operators, as long as they "can be naturally understood as having the same intended range of possible values". (p. 278) This principle, Lappin argues, makes no reference to the syntax of logical form expressions. In addition, it allows sentences like (10) which were ruled out by the LF identity theory. However, there remain clear counterexamples to Lappin's rule as well. For example:

(10) Every boy in Bill's $class_1$ wanted Mary to kiss him_1, but a boy in John's $class_2$ hopes that she won't. [kiss him_2].

Lappin's weakened identity condition must be further weakened. A meaning level account accomplishes this in a natural way.

2.3.2 Evidence for a Meaning Level Account

I represent a VP meaning as a three place relation on a property and an input and output discourse models. According to this approach, a pronoun in a VP is evaluated independently in the antecedent and target contexts. This differs sharply from the accounts discussed above. The following examples violate the constraints imposed in this accounts. The examples are quite naturally explained by appealing to change in discourse context from antecedent to target.

In examples (12) - (14), a pronoun in the antecedent switches referent to a newly salient entity in the target, although this cannot be explained by the variable-binding mechanisms of any of the above accounts.

(11) i. I never expected $John_i$ to fail his_i exam.
ii. $Bill_j$, I did.[expect to fail his_j exam]

(12) i. If Tom_i was was having trouble in school, I wouldn't help him_i.
ii. If $Harry_j$ was having trouble, I guess I would. [help him_j]

(13) i. If women_i are often frustrated because men_j do not respond to their_i troubles by offering matching troubles,

ii. men_j are often frustrated because women_i do. [respond to their_j troubles by offering matching troubles]

That's Not What I Meant, Tannen 1990.

In the following examples pronouns switch from one binder to another, or from bound to free, or free to bound.

(14) Every boy_i thinks the teacher will like his_i work, but in Bill_j's case, I think she will. [like his_j work]

(15) Every boy_i in the class wants Mary to kiss him_i.
Except John_j – he doesn't want her to.[kiss him_j].
In fact he can't stand Mary.

(16) John_i doesn't want Mary to kiss him_i.
That's funny – I thought any boy_j would want her to.[kiss him_j]

(17) Every boy_i in Bill's class wanted mary to [kiss him_i],
but three boys_j in John's class actually asked her to.

There is also more general evidence that VP ellipsis is to be handled on the level of meanings. For example, deictic cases of VP ellipsis are possible, such as the following example[3], which one person might say to another if they are contemplating jumping into a cold stream:

(18) I will if you do.

The possibility of VP ellipsis with no linguistic antecedent would seem to be rule out any account which relies on a syntactic level of representation. There are also other cases that indicate that VP ellipsis is to be resolved at the level of meanings: these cases show that inference is sometimes required to resolve VP ellipsis, and the desired antecedent is sometimes formed by a logical combination of salient predicates.

Webber[41] argues that examples such as the following require inference:

(19) Martha and Irv wanted to **dance together**, but Martha couldn't, because her husband was there.

The following example requires the disjunction of two salient predicates ("move forward in time or move backward in time"):

(20) After the symmetry between left-handed particles and right-handed anti-particles was broken by the kaons in the 1960s, a new symmetry was introduced which everybody swears is unbreakable. This is between left-handed particles **moving forwards in time**, and right-handed anti-particles **moving backwards in time** (none do, in any practical sense, but that does not worry theorists too much).
(From *The Economist*, 4 August 1990, p.69. Bonnie Webber, p.c.)

It is also possible to have an apparently unbounded amount of intervening material between antecedent and target. Since the memory of syntactic structure is known to be much more short-lived than memory of meanings, this is also evidence for a meaning level approach.

(21) i. Do you think that has **influenced your approach to filmmaking**? For example, "Stranger Than Paradise" seems influenced by Godard.

ii. Yes, I'm sure it has.
(Terry Gross interviewing Jim Jarmush on "Fresh Air" 12/3/90)

In the following section, I describe a meaning level approach.

Chapter 3

Defining the Model

3.1 Introduction

In this section I will describe a computational model in which VP ellipsis is interpreted at the level of meaning. Interpretation will be defined with respect to a discourse model, as follows:

- VP-meanings are stored in the discourse model.
- VP-meanings have context parameters.

This allows the antecedent and target VP's to be interpreted independently in their respective discourse contexts.

The model I will describe is based on the "Incremental Interpretation" system described by Pereira and Pollack in [32]. This system gives the desired semantic representation for VP's, since VP's, and indeed all expressions, are represented semantically with parameters for input and output discourse models. In addition, an account is provided in this system for pronominal anaphora, including rules for introducing entities into the discourse model and for selecting entities among those represented in the discourse model. The rules I give for VP ellipsis will be analogous to those for pronominal anaphora.

The semantic representation of a VP is a three-tuple $< \mathrm{DM}_{in}, \mathrm{P}, \mathrm{DM}_{out} >$, consisting of a property P and input and output discourse models. Two types of rules will be defined. First, a rule to introduce new VP-meanings to the discourse model. Such a rule will be associated with "complete VP's", i.e., a verb with all its arguments filled except for the subject. Second, a rule will be defined which resolves VP ellipsis by selecting among the VP-meanings stored in the input discourse model.

3.2 The Incremental Interpretation System

3.2.1 Overview

A semantic representation in the Incremental Interpretation (henceforth II) System is called a "Conditional Interpretation", which is defined as an assumption-sense pair, A:s, where A is a set of assumptions, and s is the sense. The sense can be thought of as the ordinary truth-conditional semantic representation. The assumption set consists of assumptions that have been introduced during the derivation, and must be discharged before the derivation is complete. The assumption set "represents constraints on how the sense may be further connected to its context." [32]

The incremental interpreter is defined by a set of ***structural rules*** and a set of ***discharge rules.*** Structural rules build the conditional interpretation of a phrase compositionally, from the conditional interpretation of its parts. Discharge rules remove assumptions. In principle all rules have an input and output discourse model, but only the discharge rules actually interact with the discourse model.

The form of a structural rule is

$P \sim A{:}s$ **if** $P_1 \sim A_1{:}s_1$ **and** ...**and** $P_k \sim A_k{:}s_k$

The $\sim$ denotes the interpretation relation between a node of an analysis tree (produced by the parser) and a node of a semantic derivation tree (produced by the incremental interpreter). P denotes a syntactic node, where its immediate constituents are denoted by variables P_1 through P_k. The rule schema is to be understood as stating a constraint that P should have the interpretation A:s if it has constituents P_1 through P_k, and these constituents have the interpretations indicated.

The form of a discharge rule is

$P \sim A'{:}s'$ **if** $P \sim A{:}s$

Here, $A' = A - \{R\}$, where R is the discharged assumption. The discharge of R, together with the current state of the discourse model, determines some modifications to s, resulting in s'.

The assumption storage mechanism is based on Cooper storage [6], which was applied to quantifier phenomena. In the II system, this mechanism is applied to several additional phenomena. Below, I will describe the rules for quantifiers and for pronominal anaphora.

3.2.2 Rules for Pronominal Anaphora

The treatment of pronominal anaphora in the II system is similar to the approach in Discourse Representation Theory([23], [17]): indefinite NP's introduce new elements in the discourse model. Pronouns and definite descriptions find their referent among elements in the discourse model.

Four types of referential NP's are defined: pronouns, definite descriptions, indefinites, and names. They are represented as follows:

bind(x,pronoun,number/gender): x

bind(x,def,sort): x

bind(x,indef,sort): x

bind(x,name,N): x

In each case, the sense is represented by a parameter x, and a binding assumption expresses constraints on the way x will be replaced by an entity in the discourse model. This is achieved by discharging the bind assumption. The discharge rules are:

A, bind(x,pronoun,num/gen): S $\Rightarrow$ A: S[x/e]

A, bind(x,def,sort):S $\Rightarrow$ A: S[x/e]

A, bind(x,indef,sort):S $\Rightarrow$ A: S[x/e]

A, bind(x,name,N): S $\Rightarrow$ A: S[x/e]

In the case of pronouns and definite descriptions, the element e must be a salient element in the input discourse model, satisfying the constraints expressed in the binding assumption. An indefinite assumption causes a new element e to be added to the output discourse model. In each case, e is substituted for each occurrence of x in the sense S. At least for pronouns, there is a second possibility: instead of selecting e from the discourse model, some other, undischarged parameter can be selected. This allows a pronoun to be bound by a quantifier, as described below.

3.2.3 Rules for Quantifiers

The treatment of quantifiers in the II system essentially duplicates that of Cooper[6]. A quantified NP is represented by storing a quantifier assumption, together with a parameter representing the sense. At some later stage in the derivation, the quantifier assumption is discharged, determining the scope of the quantifier. There are two general rules for quantifiers, governing the introduction and discharge of quantifier assumptions. A quantified NP is represented as:

bind(x,q,n): x

where x is a parameter, q is the quantifier, and n is the common noun. For example, "every jet" is represented

bind(x,every,jet): x

Simplifying slightly, the discharge of quantifier assumptions can be represented as follows:

bind(x,q,s): $p_t \Rightarrow$ (q s x) p

As an example,

bind(x,every,jet): fly(x) $\Rightarrow$ (every jet x) fly(x)

As mentioned above, when a pronoun assumption is discharged, its parameter is replaced either by an entity in the discourse model, or by some, yet undischarged parameter. A pronoun becomes "bound" by a quantifier if the quantifier parameter replaces the pronoun parameter in this way.

3.3 The Account of VP Ellipsis

I now describe a semantic account of VP ellipsis in terms of some simple extensions to the II system. The approach parallels the above approach to pronominal anaphora. I define a rule to add VP-meanings in the discourse model, and a rule for recovering those VP-meanings to resolve an elliptical VP. Thus full VP's are analogous to indefinite NP's, in that they both typically introduce semantic objects into the discourse model, and elliptical VP's are analogous to pronouns, in that their interpretation requires the selection of an appropriate object from the discourse model. The discourse model has two sets: SE, the set of salient entities, and SP, the set of salient predicates.

To add VP-meanings to the discourse model, I allow all lexical verbs to introduce an assumption which adds the VP-meaning to the discourse model. I call this binding assumption type "pred". It is discharged as follows:

A, bind(pred):S $\Rightarrow$ A: S

where

DM_{out} (SP) = DM_{in} (SP) $\cup$ {A:S}

That is, the discharge results in the semantic representation of the VP (i.e.,the assumption-sense pair A:S) being added to the SP set of the output discourse model.

I add the requirement that all arguments except the subject must be filled before the assumption is discharged. That is, the discharge of this assumption is permitted only if the sense is of the form

P(SUBJ, a_1, ..., a_n)

where SUBJ represents an unfilled subject argument position, with the remaining arguments a_1 through a_n filled.

The assumption for recovering a VP-meaning is introduced by a lexical auxiliary verb; this assumption is termed epred, for elliptical predicate.

The discharge rule is:

bind(epred): AUX $\Rightarrow$ A:S

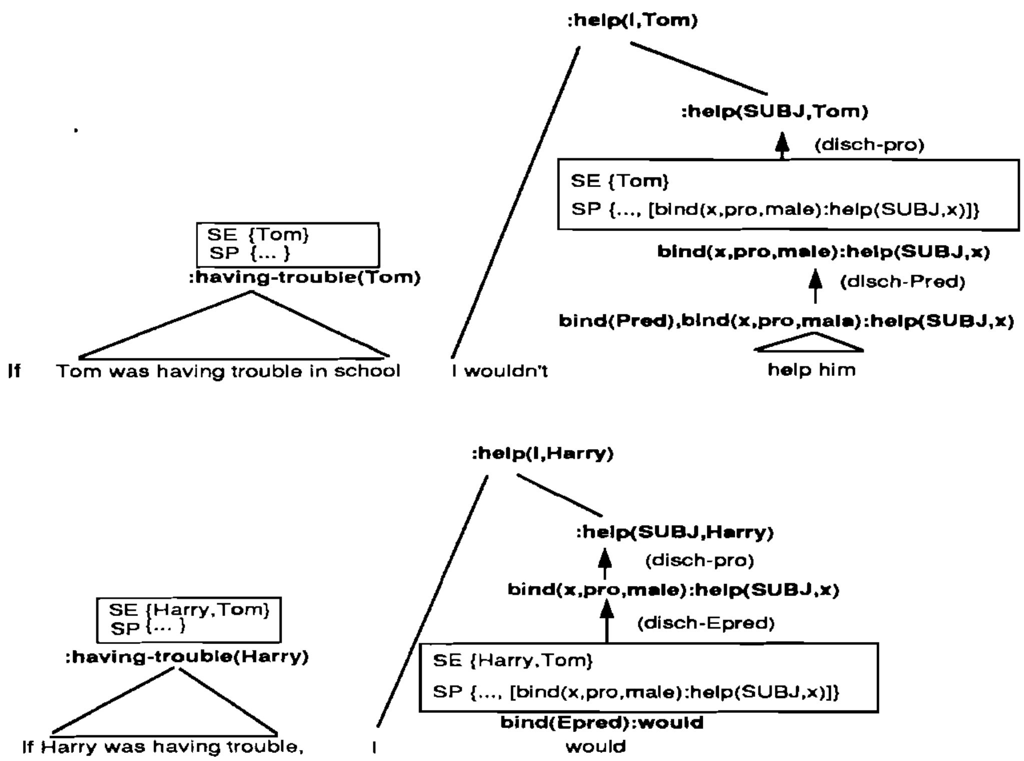

Figure 3.1: Derivation of Example (13)

where A:S is some element of the SP set in DM_{in}. That is, upon discharge of the epred assumption, an auxiliary verb is replaced by some VP-meaning in the input discourse model[1].

The crucial point in these rules is that the antecedent VP is represented as an assumption-sense pair, since it is the assumptions that represent dependencies on context. For example, the representation of the VP "help him" might be

bind(x,pronoun,male): help(SUBJ,x).

This expresses the constraint that the object position must be filled by some entity in the discourse model according to constraints of pronominal reference. Two copies of this VP, as antecedent and target in VP ellipsis, could allow the pronoun to refer to different entities, depending on the state of the current discourse model.

3.4 Sample Derivations

A simplified derivation of Example (13) is displayed in Figure 3.1. Each node of the derivation tree is labelled with the derivation rule, the current state of the discourse model, and the semantic representation, represented as an assumption:sense pair. The nodes of interest involve

[1]Note: The contribution of the auxiliary verb, e.g., tense and polarity, is ignored for the sake of simplicity.

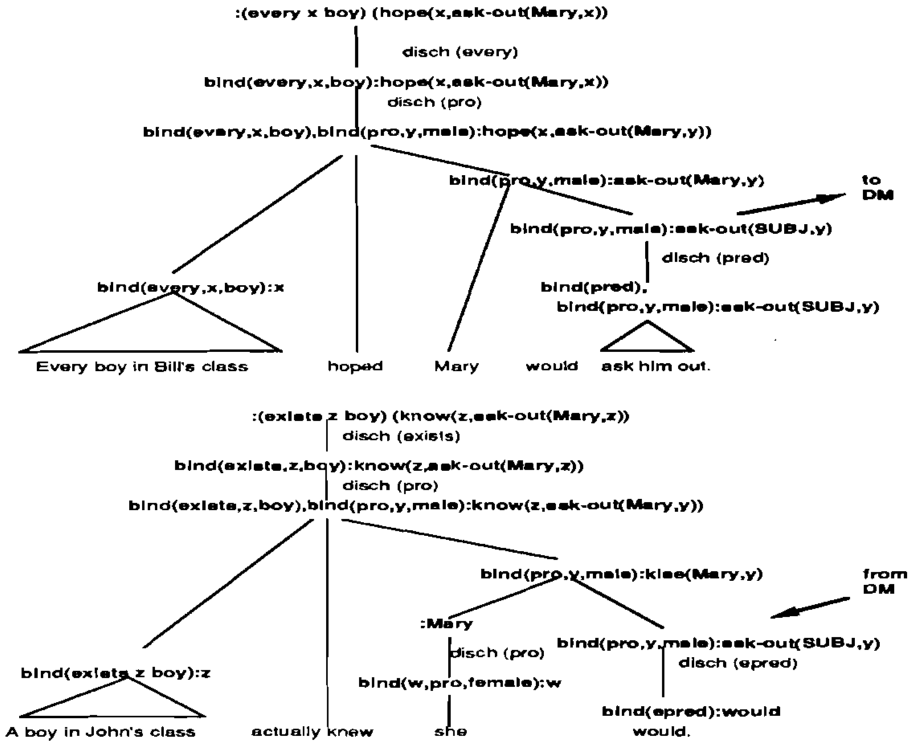

Figure 3.2: Derivation of Example (23)

assumption discharge, showing the resolution of pronominal anaphora and VP ellipsis. The lexical verb "help" introduces the binding assumption pred, which is discharged after the object position is filled. The discharge assumption results in the semantic representation of the VP being added to the set SP in the discourse model. Next the binding assumption for the pronoun "him" is discharged, selecting the entity *Tom* from the discourse model. In the elliptical sentence, the auxiliary verb "would" introduces the assumption epred, which is immediately discharged. This results in an element of SP being selected from the discourse model; in this case, **bind(x,pro,male): help(SUBJ,x)**. This becomes the representation of the VP. The pronoun binding assumption is now discharged, and the entity *Harry* is selected from the discourse model, resulting in the desired reading.

Next, I describe the derivation of the following example.

(22) Every boy$_i$ in Bill's class hoped Mary would **ask him$_i$ out**, but a boy$_j$ in John's class actually knew that she would. [ask him$_j$ out]

The derivation is displayed in Figure 3.2.

The antecedent VP "ask him out" is represented as

bind(pred),

bind(y,pronoun,male): ask-out(SUBJ,x).

The discharge of the **pred** assumption results in

bind(y,pronoun,male): ask-out(SUBJ,x)

being added to the discourse model. Later, the binding assumption for the pronoun is discharged, allowing it to be bound by the quantifier every boy. In the interpretation of the elliptical VP, the auxiliary "would" is represented

bind(epred):would

The discharge of the **epred** assumption results in the selection of a VP-meaning from the current discourse model: in this case,

bind(y,pronoun,male):ask-out(SUBJ,x)

is selected. Later, the binding assumption for the pronoun is discharged, allowing the pronoun to be bound to "a boy".

3.5 Constraints on Reference Resolution

In this section I will describe the way in which entities and predicates are located in the discourse model. The approach to entity location will draw on existing syntactic and computational accounts. Little work has been done on the analogous problem for VP ellipsis, the location of predicates. I will argue that the location mechanism for predicates has strong similarities to the entity locater, suggesting that there may in fact be a single location mechanism with a small set of parameter settings for objects of different types.

3.6 Locating Entities

The rules given above on pronominal anaphora merely state that a salient entity e is selected from the discourse model, replacing the pronoun parameter. Nothing has been said about how that entity is selected, when there have been several different entities mentioned in the discourse. This problem, which I am calling the entity location problem, has been addressed in various ways: in theoretical linguistics frameworks such as GB, the problem of configurational non-coreference constraints has received much attention. In Computational Linguistics, it has been recognized that recency and relative salience of various potential antecedents must be taken into account. The centering framework is one attempt to implement such factors computationally. While it is limited in important ways, has performed relatively well in empirical tests [40], and has been validated to some extent in psycholinguistic experiments [20]. Also, it has long been recognized that "selectional restrictions" rule out certain possible antecedents as semantically inappropriate.

Recent work [7] suggests that it may be possible to automatically derive approximations of selectional restrictions from large corpora.

The entity locater function described below incorporates aspects of all the above perspectives, by applying three filters in sequence to the set of possible antecedents: a syntactic filter, a salience filter, and a selectional restrictions filter. The entity locater function ϕ takes an NP and discourse model as arguments, and returns an entity, selected from the SE set, as the referent of the NP. It is defined as follows:

Entity Locater Function:
ϕ(NP,dm)

```
let rset = syn-filter(NP,SE(dm))
  if |rset| = 1 done
  else let rset = salience-filter(rset)
    if |rset| = 1 done
    else let rset = selectional-restr-filter(rset)
      return(rset)
```

The filters applied by The Entity Locater Function are described below.

3.6.1 Syntactic Filter

The syntactic filter eliminates entities that violate syntactic coreference constraints, as discussed in the "Binding Theory" [5]. While the precise formulation of these constraints is controversial, a standard version is in terms of three principles: Principle A, requiring that reflexives and reciprocals are locally bound, Principle B, requiring that pronominals are locally free, and Principle C, requiring that Names are globally free. In addition to enforcing these configurational constraints, the syntactic filter matches features such as number and gender.

3.6.2 Salience Filter

The salience filter relies on the fact that elements of SE are partially ordered in terms of salience. The specific definition of this salience ordering will be a topic for further investigation. In the Centering model [12] there are two factors which are taken into account: syntactic prominence, and the pronoun/full NP distinction. In addition, there is a preference for retaining the same "center", if possible. Various implementations and extensions to this approach exist. I will attempt to incorporate the most plausible features of these approaches, and subject them to some empirical testing.

3.6.3 Selectional Restrictions Filter

Finally, entities which violate selectional restrictions are eliminated. This filter could be implemented along the lines of [7]. In this work, statistical evidence on cooccurrence patterns is collected, and this data functions as a sort of approximation of selectional restrictions. For example, in

(23) The company's statement said it was "studying its options".

there are two possible antecedents for "it": "company" and "statement". But "statement" is not an appropriate subject for the verb "studying". This would presumably be reflected in the fact that "statement" and "studying" cooccur infrequently. Thus "statement" is ruled out as a possible antecedent.

3.7 Locating Predicates

The problem of locating predicates, i.e., antecedents for VP ellipsis has received little attention in the literature. As an initial hypothesis, I will assume that the predicate locater has the same structure as the entity locater, defining it in terms of the successive application of three filters, as follows:

Predicate Locater Function:
ψ(EVP,dm)

```
let rset = syn-filter(EVP,SP(dm))
  if |rset| = 1 done
  else let rset = salience-filter(rset)
    if |rset| = 1 done
    else let rset = selectional-restr-filter(rset)
```

In what follows, I will briefly examine each of these filters, suggesting ways in which they might be defined.

3.7.1 Syntactic Filter

The syntactic filter eliminates potential antecedents based on configurational restrictions. Such configurational "coreference constraints" have been much studied in the case of pronominal anaphora, but have not been systematically addressed concerning VP ellipsis. For present purposes, I will mention two proscribed configurations that have been suggested:

1. An antecedent that is preceded and c-commanded by its target. [22]
2. An antecedent A, such that the target T is contained in a relative clause modifying a noun N, where N is an argument of a finite verb A′ that is dominated by A. [26]

Restriction 1 rules out examples such as

(24) Charlie will, if his mother-in-law doesn't leave town.

in which the VPE precedes and c-commands the antecedent.

Restriction 2 deals with cases of antecedent-contained VP ellipsis. It requires that the containing antecedent be the "nearest" containing VP; it is not possible for a VPE V to have a containing antecedent verb A if there is some finite verb A' contained within A such that A' also contains V. For example

(25) John knows Harry read the book that Bill did.

Here the only possible antecedent is the VP headed by "read"; the VP "knows Harry..." is ruled out by Restriction 2.

3.7.2 Salience Levels for VP's

As an initial hypothesis, I will define three levels of salience for the "Salient Predicate" set SP: level 1,the lowest level, is Predicates evoked by VP's dominated by NP's; level 3, the highest level, is VP's evoked as questions. The salience of VP's in questions has been supported in the psycholinguistic literature[27], in which it has been argued that VP's in questions are more likely as antecedents than other VP's. Level 1 Predicates are those that are not being asserted, but rather, being used for some other purpose, such as to help determine the referent of an NP. These are generally less salient, as pointed out in [14] to account for examples such as the following:

(26) A: The policeman paid no attention to the girl who was driving the car.

(27) *B: Was she really?

3.7.3 Selectional Restriction Filter

Finally, if there is still ambiguity, selectional restrictions are applied. An implementation is planned along the lines of the filter in the "entity locater".

Chapter 4

Proposed Work

A central aspect of the work proposed here is a computer implementation of the model I have described. There are two general purposes for this implementation. First, it provides a context for the examination of many details in the model that have not been fully addressed here. Second, the computer implementation will allow testing of the model on a large number of naturally-occuring examples. The results of these tests will provide guidance for further modifications in the model. Below, I describe the plans for implementing and testing the model. In addition, I sketch ways of extending the model.

4.1 Computer Implementation

In Chapter 3, I described a system based on some extensions to the existing Incremental Interpretation system. As currently implemented, this system takes a parse tree and discourse model as input, the output being a semantic representation and modified discourse model. I will add two types of rules to this system for VP ellipsis: rules for adding VP-meanings to the discourse model, and rules for recovering them. In addition, it will be necessary to extend the discourse model representation to include VP meanings.

4.1.1 Adding VP-meanings to the Discourse Model

In the II system, all interaction with the discourse model involves the discharge of an assumption. So adding an object to the discourse model requires two steps: first, a rule for the introduction of an assumption during the interpretation process, and second, a rule for the discharge of that assumption at some later point in the interpretation process. The discharge rule will specify the way in which the object is added to the discourse model.

Introducing the "pred" Assumption

The semantic interpretation of a syntactic parse tree is performed by recursive calls to the predicate **interpret1**. Below is the clause for the interpretation of a verb:

```
interpret1(verb,Fs,'Int,Assms,_Context,TContext,TContext) :-
access(wordstem,Fs,Word),
v_sense(Word,VerbSense),
VerbSense =.. [Pred|ArgTypes0],
create_vars(ArgTypes0,StoredVars,GenVars),
sub_all(ArgTypes0,StoredVars,GenVars,ArgTypes),
ubind_assms(ArgTypes,ArgVars,Assms),
Int =.. [Pred|ArgVars].
```

The parameters of this clause are the syntactic category of the expression to be interpreted, the syntactic representation of the expression, the resulting conditional interpretation, the assumptions made during the interpretation of the expression, the overall context, and the input and output (temporary) contexts.

The **Assms** parameter contains a list of all the assumptions contributed by the verb. These assumptions are used to express selectional restrictions upon the argument requirements of the particular verb. It is necessary to add to this list an assumption of type "pred". This is effected by the followed, modified version of this clause:

```
interpret1(verb,Fs,'Int,Assms,_Context,TContext,TContext) :-
access(wordstem,Fs,Word),
v_sense(Word,VerbSense),
VerbSense =.. [Pred|ArgTypes0],
create_vars(ArgTypes0,StoredVars,GenVars),
sub_all(ArgTypes0,StoredVars,GenVars,ArgTypes),
ubind_assms(ArgTypes,ArgVars,Assms1),
        Assms = [bind(pred) | Assms1],
Int =.. [Pred|ArgVars].
```

Here, the list of output assumptions, **Assms**, includes the assumption **bind(pred)** in addition to the other assumptions.

Discharging the pred Assumption

Assumptions are discharged by the predicate **eliminate**. The following clause is added to the definition of this predicate. The parameters of interest are: **pred**, the type of assumption to be discharged, **Int**, the current interpretation, **Assms0**, the input assumptions, and **Tcontext0**, the input temporary context. The eighth parameter is the output temporary context, which later is added to the overall discourse model.

```
eliminate(pred,Int,Assms0,TContext0,_Context,Int,Assms,
    [spec(pred,Pn,[Vsense|Assms0])|TContext0],Current) :-
            getsense(Current,Vsense),
```

This definition allows the discharge of a **pred** assumption, causing a verb phrase meaning to be added to the output temporary context, which is later added the the overall discourse model. The VP meaning is represented as a **spec** predicate. This predicate takes four parameters: the first is the type: "pred" or "entity". The second is a variable name, and the third is the VP-meaning, represented as a list whose first element is the verb sense, and the rest of the list is the current assumptions.

4.1.2 Recovering VP-meanings from the Discourse Model

To recover a VP-meaning from the discourse model and thereby resolve an occurrence of VP ellipsis, I define a new assumption type: **epred** ("elliptical predicate"). It is introduced by the occurrence of a bare auxiliary verb, and it is discharged at some unspecified point during a derivation.

Introducing the "epred" Assumption

The introduction of an epred assumption is effected by adding a clause to the interpret1 predicate, as follows:

```
interpret1(bare-aux,Fs,'Int,Assms,_Context,TContext,TContext) :-
           .
           .
           .
     Assms = [bind(epred) | Assms1].
```

This adds an epred assumption to the output list of assumptions, in the interpretation of a bare auxiliary verb.

Discharging the "epred" Assumption

The epred assumption is discharged by adding a clause to the definition of eliminate:

```
eliminate(epred,Int,Assms0,TContext0,Context,Int0,PredAssm,
          TContext0,Current) :-
    resolve_evp(Context,PredAssm,Int0).
```

The predicate **resolve_evp** ("resolve elliptical VP") selects from the discourse model a VP-meaning, which becomes the value of **Int0**. This is the output interpretation. The predicate **resolve_evp** will be defined to reflect the filters for predicate location described in chapter 3. The definition of these filters depends on the structure of the discourse model, which is discussed in the next section.

4.1.3 Representation of the Discourse Model

The discourse model as defined in [32] consists of the set of entities evoked in prior discourse. I will call this set SE ("Salient Entities"). These entities are ordered in terms of recency, and facts about their syntactic position are also recorded. These facts are taken into account by the **resolve_pn** predicate, which implements a simplified version of GB "binding theory" constraints on coreference, and also displays a general preference for more recent antecedents. I plan to explore some ways of improving the constraints on pronominal anaphora to reflect the filters on reference discussed in Chapter 3.

I will add the set SP ("Salient Predicates") to the discourse model, and a predicate **resolve_evp** will be defined with respect to this set. This predicate will select an element of SP, based on recency and other constraints. I will implement the filters described in Chapter 3.

4.2 Testing the Model

I plan to test the model on a large number of naturally occurring examples of VP ellipsis. I have collected several hundred examples of VP ellipsis from the Brown Corpus, as described in [16]. The model will be tested on these examples in two ways:

1. **Location of Antecedent:** this is a test of the "Predicate Locater" algorithm. The results will be compared with those of other algorithms, such as [26].

2. **Interpretation:** once a given antecedent is selected, the model will determine what readings are possible. These results can be compared with LF and S-structure approaches.

4.3 Extending the Model

In Chapter 2 several examples were given as general evidence for a meaning level approach: these examples are repeated here, with some suggestions about how the proposed model might be extended to handle them.

4.3.1 Inference

It has been pointed out by Webber [41] that there are cases of VP ellipsis in which the antecedent is a predicate that was not syntactically evoked, but is inferrable in context. One example Webber gives is:

(28) Martha and Irv wanted to **dance together**, but Martha couldn't, because her husband was there.

The simplest account of this would be to allow the set SP to be closed under an inference relation. That is, if, for some A in SP, if A' follows from A, then A' is also added to SP. Clearly, such a rule overgenerates,however, as Webber points out. It is necessary to determine the constraints that govern the inference relation. In doing this, it is worth noting that pronouns and definite descriptions can also refer to inferrable entities. In future work, I intend to define some constrained inference relations that can be applied to both the SE and SP sets.

4.3.2 Combined Predicates

It is sometimes possible for two salient predicates to be combined as the antecedent for VP ellipsis. For example:

(29) After the symmetry between left-handed particles and right-handed anti-particles was broken by the kaons in the 1960s, a new symmetry was introduced which everybody swears is unbreakable. This is between left-handed particles **moving forwards in time**, and right-handed anti-particles **moving backwards in time** (none do, in any practical sense, but that does not worry theorists too much).
(From *The Economist*, 4 August 1990, p.69. Bonnie Webber, p.c.)

Clearly, arbitrary conjunctions of salient predicates are not permitted. Similarly, salient entities can be combined to produce plural entities. For example,

(30) Tom and Sally were married last week.
(31) They had a big wedding.

Combined salient propositions can also be antecedents. In future work, I will investigate the constraints governing such "combining" operations. Perhaps it will be possible to formulate a single constraint governing a polymorphic "combining" operation.

4.3.3 Non-syntactically Parallel Antecedents

The antecedent for VP ellipsis is not always a syntactic VP. Similarly, pronouns such as "it" can have non-NP antecedents. Such cases are not treated in the grammar as currently formulated. One solution would be to to define rules allowing non-VP's to evoke predicates. Another possibility would be to allow a process of "accommodation" to handle such cases.

Bibliography

[1] Jon Barwise. "Noun Phrases, Generalized Quantifiers, and Anaphora". In Gardenfors, P. (ed.) *Generalized Quantifiers*, Dordrecht: Riedel, 1-29. 1987.

[2] Susan E. Brennan, Marilyn Walker Friedman, and Carl J. Pollard. "A Centering Approach to Pronouns", *Proceedings of the 25th Annual Meeting of the ACL*, 1987.

[3] Wynn Chao. *On Ellipsis.* Ph.D. thesis, University of Massachusetts-Amherst. 1987.

[4] Noam Chomsky. *Aspects of the Theory of Syntax.* Cambridge, MA. MIT Press, 1965.

[5] Noam Chomsky. *Lectures on Government and Binding.* Foris. 1981.

[6] Robert Cooper. *Quantification and Syntactic Theory.* D. Riedel, Dordrecht, Netherlands. 1983.

[7] Ido Dagan and Alon Itai. "A Statistical Filter for Resolving Pronoun References" in *7th Israeli Symposium on Artificial Intelligence and Computer Vision*, 1990.

[8] Mary Dalrymple, Stuart Shieber and Fernando Pereira. Ellipsis and Higher-Order Unification. *Linguistics and Philosophy.* Vol. 14, no. 4, August 1991.

[9] Robert Fiengo and Robert May. "Ellipsis and Anaphora". Paper presented at GLOW 1990, Cambridge University, Cambridge, England.

[10] Robert Fiengo and Robert May. Indices and Identity. ms. 1991.

[11] Jeroen Groenendijk and Martin Stokhof. "Dynamic Predicate Logic". ms. 1989.

[12] Barbara Grosz, Aravind Joshi, and Scott Weinstein. "Providing a Unified Account of Definite Noun Phrases in Discourse." In *Proceedings, 21st Annual Meeting of the ACL*, pp. 44-50, Cambridge, MA, 1983.

[13] Barbara Grosz, Aravind Joshi, and Scott Weinstein. "Towards a Computational Theory of Discourse Interpretation". ms. 1986.

[14] M.A.K. Halliday and R.H. Hasan. *Cohesion in English.* London: Longman, 1976.

[15] Jorge Hankamer and Ivan Sag. "Towards a Theory of Anaphoric Processing." in *Linguistics and Philosophy* 7, pp 325-345, 1984.

[16] Daniel Hardt. "A Corpus-based Survey of VP Ellipsis", ms. University of Pennsylvania, 1990.

[17] Irene Heim. *The Semantics of Definite and Indefinite Noun Phrases.* Ph.D. thesis, University of Massachusetts-Amherst. 1981.

[18] Jerry Hobbs. "Resolving Pronoun References", *Lingua* 44, pp. 311-338. 1978.

[19] Arild Hestvik. "Strict/sloppy identity, anaphors, and reconstruction." Manuscript, University of Stuttgart. 1991.

[20] Hudson and Tanenhaus. (Forthcoming)

[21] Gerard Huet and Bernard Lang. "Proving and Applying Program Transformations Expressed with Second Order Patterns". *Acta Informatica*, 11(1):31-55. 1978.

[22] R.S. Jackendoff. *Semantic Interpretation in Generative Grammar.* Cambridge, Mass. MIT Press. 1972.

[23] Hans Kamp. "A Theory of Truth and Semantic Representation". In Groenendijk, J, Janssen, T.M.V. and Stokhof, M. (eds.) *Formal Methods in the Study of Language*, Volume 136, pp. 277-322. 1980.

[24] Yoshihisa Kitagawa. Deriving and Copying Predication 1990.

[25] Shalom Lappin. VP Anaphora, Quantifier Scope, and Logical Form. *Linguistic Analysis.* Volume 13, No. 4, pp. 273-315. 1984.

[26] Shalom Lappin and Michael McCord. "Anaphora Resolution in Slot Grammar", in *Computational Linguistics*, vol 16, no 4. 1990.

[27] Barbara Malt. The Role of Discourse Structure in Understanding Anaphora. *Journal of Memory and Language, 24*, 271-289. 1984.

[28] Michael McCord. "SLOT GRAMMAR: A system for simpler construction of practical natural language grammars." In *Natural Language and Logic, International Scientific Symposium*, edited by R. Studer, Lecture Notes in Computer Science, Springer Verlag, 118-145. 1990.

[29] Richard Montague. *Formal Philosophy.* 1974.

[30] Barbara Partee. "Montague Grammar and Transformational Grammar" in *Linguistic Inquiry*,

[31] Barbara Partee, Emmon Bach. "Quantification, Pronouns, and VP Anaphora", in *Truth, Interpretation and Information*, Dordrecht 1984.

[32] Fernando Pereira and Martha Pollack. Incremental Interpretation. *Artificial Intelligence.* Vol. 50. no. 1, pp. 37-82. June 1991.

[33] Hub Prüst and Remko Scha. A Discourse Approach to Verb Phrase Anaphora. Proc. ECAI, 1990.

[34] Hub Prüst, Remko Scha, and Martin van den Berg. A Formal Discourse Grammar tackling Verb Phrase Anaphora. ms. 1991.

[35] Tanya Reinhart. *Anaphora and Semantic Interpretation.* Univ. of Chicago Press. 1983.

[36] Mats Rooth. A Comparison of Three Theories of VP Ellipsis. In W. Chao and D. Wheeler, eds., *University of Massachusetts Occasional Papers in Linguistics 7*, Amherst, Ma. 1981.

[37] Mats Rooth. "Noun Phrase Interpretation in Montague Grammar, File Change Semantics, and Situation Semantics". In Gardenfors, P. (ed.) *Generalized Quantifiers*, Dordrecht: Riedel, 237-269. 1987.

[38] Ivan A. Sag. *Deletion and Logical Form.* Ph.D. thesis, University of Massachusetts-Amherst. 1976.

[39] Paul Schacter. "Constraints on Coordination". *Language* 53, 86-103. 1977.

[40] Marilyn Walker. Evaluating discourse processing algorithms. In *Proceedings, 27th Annual Meeting of the ACL*, Vancouver, Canada. 1989.

[41] Bonnie Lynn Webber. *A Formal Approach to Discourse Anaphora.* Ph.D. thesis, Harvard University. 1978.

[42] Edwin Williams. *Discourse and Logical Form. Linguistic Inquiry*, 8(1):101-139. 1977.