

DEPARTMENT OF COMPUTER SCIENCES  
CITY COLLEGE  
CITY UNIVERSITY OF NEW YORK

Research Report #76-1

August 1976

PRONOUN RESOLUTION

Jerry R. Hobbs

*Present Address:*

*Artificial Intelligence Center  
SRI International  
333 Ravenswood Ave  
Menlo Park, California 94025*

## PRONOUN RESOLUTION

Jerry R. Hobbs

Department of Computer Sciences

City College, CUNY

### ABSTRACT

Two approaches to the problem of pronoun resolution are presented. The first is a naive algorithm that works by traversing the surface parse trees of the sentences of the text in a particular order looking for noun phrases of the correct gender and number. The algorithm is shown to incorporate many, though not all, of the constraints on co-referentiality between a non-reflexive pronoun and a possible antecedent, which have been discovered recently by linguists. The algorithm clearly does not work in all cases, but the results of an examination of several hundred examples from published texts show that it performs remarkably well.

In the second approach, it is shown how pronoun resolution is handled in a comprehensive system for semantic analysis of English texts. The system consists of four basic semantic operations which work by accessing a data base of "world knowledge" inferences, which are drawn selectively and in a context-dependent way in response to the operations. The first two operations seek to satisfy the demands made by predicates on the nature of their arguments and to discover the relations between sentences. The third operation--knitting--recognizes and merges redundant expressions. These three operations frequently result in a pronoun reference being resolved as a by-product.

The fourth operation seeks to resolve those pronouns not resolved by the first three. It involves a bidirectional search of the text and "world knowledge" for an appropriate chain of inference and utilizes the efficiency of the naive algorithm.

Four examples, including the classic examples of Winograd and Charniak, are presented that demonstrate pronoun resolution within the semantic approach.

## PRONOUN RESOLUTION

Jerry R. Hobbs

Department of Computer Sciences  
City College, CUNY

### 1. Introduction

1.1. The importance of having the right algorithm for pronoun resolution, or finding the antecedent of a pronoun, can be seen in any episode of the George Burns and Gracie Allen re-runs, for much of Gracie's humor depends on her having the wrong algorithm.<sup>1</sup> One episode is built around her misunderstanding of a fire inspector's warning:

There's a pile of inflammable trash next to your car.

You'll have to get rid of it.

In another, she is shocked when a shoe salesman suggests

If these shoes don't fit your feet, you can exchange them.

In a different vein, imagine a computer program of the future which translates English descriptions of algorithms into the corresponding code in a higher-level programming language. In the text

Decrease N by DELTA. If it is 0, reset it to MAX.

the program must identify "it" with N and not with DELTA in order to produce the correct code.

In Jespersen (1954:143) the problem of pronoun resolution rates brief mention:

An ambiguity (not very serious) may sometimes arise

when there are two antecedents to which it may refer:

If the baby does not thrive on raw milk, boil it.<sup>2</sup>

What counts as serious depends on one's point of view.

In this paper, two approaches to pronoun resolution are considered. The first is a simple, efficient, but naive algorithm working on the surface parse trees of the sentences in the text. Examination of several hundred examples from a variety of published texts shows that in spite of its obvious flaws, the algorithm works remarkably well.

In the second approach, it is shown how pronoun resolution happens in a total system for semantic analysis. And the word "happens" is appropriate here. Charniak (1972) demonstrated rather convincingly that in order to do pronoun resolution, one had to be able to do everything else. In the latter part of this paper it is argued that once everything else is done, pronoun resolution comes free--it happens automatically. The system is centered around certain semantic operations which draw appropriate inferences from a data base of "world knowledge". There is one operation--"knitting"-- which recognizes redundancies among the inferences that are made. It is shown by analysis of several examples that these operations, and knitting in particular, locate the antecedents of most pronouns as a by-product. For those pronouns not resolved in this way, an efficient search method is described for locating the antecedents.

The problems studied in this paper are addressed from a computational, analytic point of view, that is, within a clearly specified framework for analyzing texts rather than generating them. In some generative approaches, a noun phrase in deep structure is transformed into a pronoun by a pronominalization rule, under the condition of "identity of reference" with another noun phrase. Little more is said about this condition; presumably the one generating the sentence is

somehow aware of this identity of reference. However, from the computational, analytic point of view, we must be very precise about how a listener could become aware of this identity of reference. When he hears or reads a pronoun, how, out of all the possible structures which could serve as an antecedent, might he be able to pick the correct one. Our emphasis is directed toward developing algorithms for doing this. Even if it is the generation of texts one is interested in, the analytic point of view may be the correct one to take, for as Olson (1970) argues, the speaker elaborates his description of an entity just to the extent that will allow his listener to identify it.

The reader will find many numbers in the exposition that follows, especially in Part 2. He should not thereby be fooled into believing the orientation of this work is strictly practical. Part 2 addresses a very important theoretical question: To what extent is a purely syntactic solution to the problem of finding antecedents possible? It is equally clear that syntax does not provide a complete solution and that syntax exerts a significant influence. Therefore, studies such as that presented in Part 2 are important for determining just what shape this influence takes. In Part 3 a principled way of incorporating this syntactic influence into a semantic solution is presented.

1.2. Review of the Natural Language Processing Literature. Most work by linguists on the problems of pronoun resolution has been concerned with elucidating syntactic constraints on the coreferentiality and non-coreferentiality of two entities occurring in the same sentence. Some of this work will be reviewed in Part 2. This section is a brief review of the most important work in natural language processing, in which the problem of locating antecedents has been addressed more directly.

Winograd (1972) was the first to write procedures for locating antecedents, in his system for manipulating and carrying on dialogs about a blocks microworld. He first applies a heuristic which is discussed in Sections 2.5 and 2.6. If this fails, he collects all possible referents and rates their plausibility on the basis of syntactic position, apparently in an order similar to that which the naive algorithm of Part 2 defines. A subject is favored over an object and both are favored over the object of a preposition. In addition, "focus" elements are favored, where focus is determined from, among other things, the answers to wh-questions and from indefinite noun phrases in yes-no questions.

In his system for general natural language inferencing, Rieger (1974) finds the antecedent of any definite entity by creating and narrowing down a candidate set of possible antecedents, based on the known properties of the entity. The difficulty with this approach for pronouns is that normally the only explicit property we have is new and not very useful for identification. If implied properties are included, then one runs into an overall problem with Rieger's system (shared by Charniak's). There are no controls on inferencing; everything that can be inferred is. When more than one candidate remains, the most recently referenced is chosen, where recency is determined by a "system clock", i.e. determined by a complex and largely unspecified order of processing.

The chief value of Charniak's work has been to show just how difficult the pronoun resolution problem is. In particular, he showed how spurious the recency principle is. His thesis (Charniak 1972) contains a wealth of difficult cases in the guise of children's stories, which show that arbitrarily detailed world knowledge can be required to decide upon an antecedent. He points out that the knowledge required for

pronoun resolution is just that which might be required by a conversational system which is asked questions about the stories. He internalizes these questions in the form of "demons", which are axioms whose antecedents have been matched and which are looking for their consequents. However, he offers no solution to the problem of what questions are the appropriate ones to ask. Section 3.2 of this paper may be seen as an attempt to solve this problem.

Wilks (1974) has given a very nice partial solution to the problem of deciding among competing plausible antecedents, based on the use of selectional information to maximize the redundancy. In addition, he uses a bidirectional search through a data base of world knowledge to resolve pronouns. His approach, in general, is similar to that of operation 4 in Section 3.2, although he lacks a notion of salience.



## 2. The Syntactic Approach: The Naive Algorithm

In this section a naive algorithm for finding antecedents of pronouns is described and related to previous linguistic research. Results are presented of a statistical study of the algorithm's effectiveness on 300 examples of pronoun occurrences from three very different texts.

2.1. Pronouns without Explicit Antecedents. By far the majority of pronouns occurring in English texts have explicit antecedents. There are some exceptions, however.

First, "it" occurs in sentences like

It surprised him that he was elected chairman  
which is transformationally related to

That he was elected chairman surprised him.

"It" may be introduced as a transformational constant, or it may refer to the "that" clause. In either case, it can be handled on the basis of purely syntactic criteria. In what follows, it will be assumed that such examples have been recognized in the analysis of the surface structure of the sentence and that no antecedent is sought.

Second, "it" occurs without explicit antecedent in weather and time constructions like

It's the weekend and it's raining again.

These may arise because the predicates are zero-argument predicates and English has no provision for subject-less sentences. On the other hand, Bollinger (1973) has presented rather strong arguments that "it" refers to the general environment. He points out the parallel between sentences like (1) and (2):

It's raining in Boston. (1)

It's intellectually stimulating in Boston (2)

The latter does not quite mean the same as

Boston is intellectually stimulating

and "be intellectually stimulating" is certainly not a zero-argument predicate.

In either case, there is no explicit antecedent in the text. Below it will be assumed that the ambient "it" in time and weather constructions can be so recognized. This does not seem possible for sentences like (2) however, for in the similar (3), "it" has the non-ambient interpretation:

Is the annual ACL conference usually interesting?

It was intellectually stimulating in Boston, but the (3)  
one in Amherst was a dud.

It happens that no sentences like (2) with the ambient "it" occurred in the sample studied.

Third, "it" is sometimes used without an explicit antecedent to refer to the implicit product of something. For example, in

Multiply the  $j$ th row by  $a$  and add it to the  $i$ th row

"it" does not refer to the  $j$ th row, for in this elementary row operation of linear algebra, the  $j$ th row remains unchanged. "It" refers to the result of multiplying the  $j$ th row by  $a$ . Similarly, in the dialog

Where does the Bible talk about pollution?

It's in the Old Testament.

"It" refers not to the Bible or pollution but to the result of the "talking".

Fourth, there is a well-studied class of examples in which the antecedent is buried in a lexically related word.

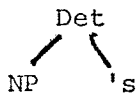
Reagan-ites think he would make a good President.

(A veritable jungle of such examples can be found in Watt 1973, 1975.)

In the last two cases the naive algorithm will simply fail. Fortunately, they are rare and neither occurred in the 300 examples studied.

2.2 The Algorithm. In what follows reference will be made to the "surface parse tree". By this is meant the tree that exhibits the grammatical structure of the sentence--its division into subject, verb, objects, adverbials, etc.--without permuting or omitting any of the words in the original sentence. That is, the terminal nodes of the tree taken in left-to-right order form the English sentence. It will be assumed however that certain syntactically recoverable omitted elements are available as antecedents, as is described in the next section.

Several assumptions will be made about the structure of the noun phrase. It will be assumed that a pronoun is immediately dominated by an NP node, and that a possessive noun or pronoun has the structure



These assumptions are a matter of convenience and anyone objecting to them will have no difficulty adjusting the algorithm to his own views of surface structure. A more crucial assumption is that the NP node has an  $\bar{N}$  node below it, as proposed by Chomsky (1970), to which a prepositional phrase containing an argument of the head noun may be attached. Truly adjunctive prepositional phrases are attached to the NP node. This assumption, or something equivalent to it, is necessary to distinguish between the following two sentences:

Mr. Smith saw a driver in his truck. (4)

Mr. Smith saw a driver of his truck. (5)

In (4) "his" may refer to the driver, but in (5) it may not. The structures we are assuming for the relevant noun phrases in (4) and (5) are shown in Figures 1a and 1b, respectively.

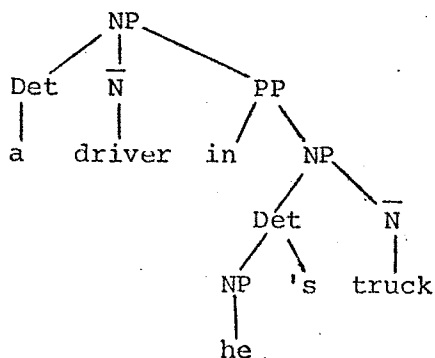


Figure 1a

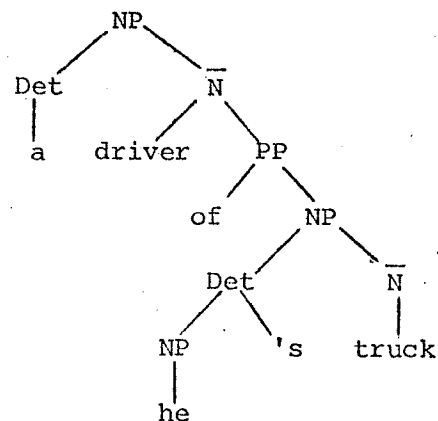


Figure 1b

The naive algorithm traverses the surface parse tree in a particular order looking for a noun phrase of the correct gender and number. The traversal order is as follows:

1. Begin at the NP node immediately dominating the pronoun.
2. Go up the tree to the first NP or S node encountered. Call this node X, and call the path used to reach it p.
3. Traverse all branches below node X to the left of path p in a left-to-right, breadth-first fashion. Propose as the antecedent any NP node that is encountered which has an NP or S node between it and X.
4. If node X is the highest S node in the sentence, traverse the surface parse trees of previous sentences in the text in order of recency, the most recent first; each tree is traversed in a left-to-right, breadth-first manner, and when an NP node is encountered, it is proposed as antecedent. If X is not the highest S node in the sentence, continue to step 5.

5. From node X, go up the tree to the first NP or S node encountered. Call this new node X, and call the path traversed to reach it p.
6. If X is an NP node and if the path p to X did not pass through the  $\bar{N}$  node that X immediately dominates, propose X as the antecedent.
7. Traverse all branches below node X to the left of path p in a left-to-right, breadth-first manner. Propose any NP node encountered as the antecedent.<sup>2</sup>
8. If X is an S node, traverse all branches of node X to the right of path p in a left-to-right, breadth-first manner, but do not go below any NP or S node encountered. Propose any NP node encountered as the antecedent.
9. Go to step 4.

A breadth-first search of a tree is one in which every node of depth n is visited before any node of depth n+1. Steps 2 and 3 of the algorithm take care of the level in the tree where a reflexive pronoun would be used. Steps 5-9 cycle up the tree through S and NP nodes. Step 4 searches the previous sentences in the text.

For the sake of concreteness, suppose we have the following context-free grammar for generating the surface structures of a fragment of English:

S → NP VP

NP →  $\left\{ \begin{array}{l} (\text{Det}) \bar{N} ( \left\{ \begin{array}{l} \text{PP} \\ \text{Rel} \end{array} \right\} )^* \\ \text{pronoun} \end{array} \right\}$

Det →  $\left\{ \begin{array}{l} \text{article} \\ \text{NP's} \end{array} \right\}$

$\bar{N}$  → noun ( PP ) \*  
 PP → preposition NP  
 Rel → wh-word S  
 VP → verb NP ( PP ) \*

Words in lower case letters mean any word of that category; parentheses, ( . . . ), indicate optional elements; the asterisk means 0 or more copies of the element that precedes it; braces, { . . . }, contain alternatives.

Figure 2 illustrates the algorithm working on the sentence

The castle in Camelot remained the residence of the king  
until 536 when he moved it to London.

Beginning from node  $NP_1$ , step 2 rises to node  $S_1$ . Step 3 searches the left portion of  $S_1$ 's tree but finds no eligible NP node. Step 4 does not apply. Step 5 rises to  $NP_2$  which step 6 proposes as antecedent. Thus, "536" is recommended as antecedent of "it".

The algorithm can be improved somewhat by applying simple selectional constraints, such as

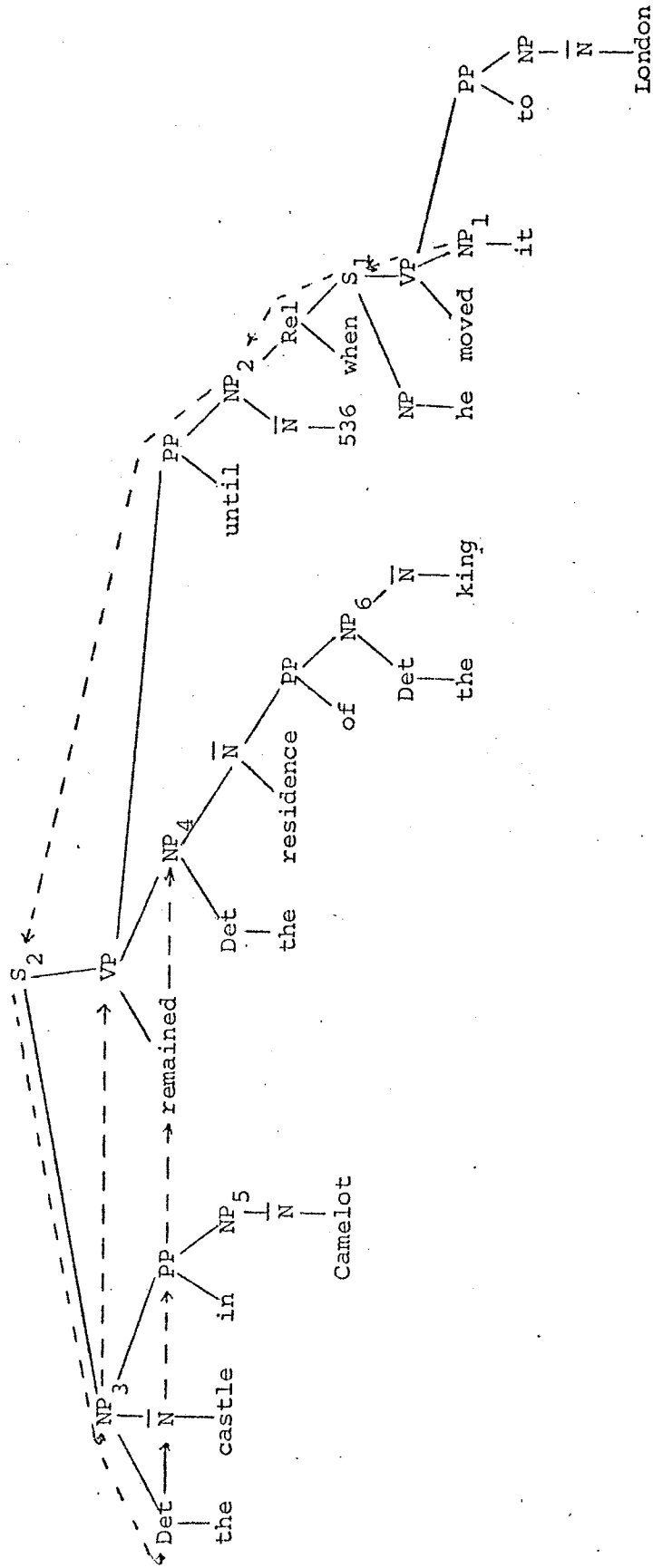
Dates can't move;

Places can't move;

Large fixed objects can't move.

The utility of these constraints is limited. They never help with the pronoun "he", since what one male human can do another can too. Even with "it" the utility is limited since most English words can occur in such a wide variety of contexts. However, in the present example, they help.

After  $NP_2$  is rejected, steps 7 and 8 turn up nothing, and control is returned to step 4 which does not apply. Step 5 rises to  $S_2$ , where step 6 does not apply. In step 7, the breadth-first search first suggests  $NP_3$  (the castle), which selectional constraints reject. It then continues



(Indices are for reference by the text.)

Figure 2

to NP<sub>4</sub> where it correctly settles upon "the residence" as antecedent of "it".

If we were searching for the antecedent of "he", the algorithm would continue, first rejecting NP<sub>5</sub> because of gender and finally lighting upon NP<sub>6</sub>, the king.

2.3. Justifications and The Relation of the Algorithm to Results from the Generative Transformational Tradition. The relative positions of pronouns and noun phrases in a surface parse tree tell something of coreferentiality. In general, from reflexive pronouns we can determine coreference and from other pronouns and nouns we can determine non-coreference. Consider

John shaved himself. (6)

John shaved him. (7)

John shaved John. (8)

We know that in (6) the subject and object are coreferential, and that in (7) and (8) they are not.

The criteria for coreferentiality and non-coreferentiality are quite complex. A great deal of work has been done in transformational grammar in recent years on stating precisely the conditions under which a noun or non-reflexive pronoun may not be coreferential with another element in the sentence. That is, the goal has been to state constraints of the form

A and B are necessarily non-coreferential if . . . .

This is equivalent to restricting the possible antecedents to a (usually still very large) set of entities. The problem of this paper is how do we determine out of this remaining set what in fact really is the antecedent. It is important to be clear about the distinction between these



two problems. The problems are of course related, but they are not the same. In a sense, a solution to the former problem defines the boundaries within which a mechanism solving the latter problem must operate.

The easiest way to take these constraints into account would be simply to assume that there is a mechanism which applies them. Then any entity which the naive algorithm proposes is checked by the mechanism, which thus acts as a filter. This in fact is what will be assumed in Part 3. Here it is more interesting, however, to see to what extent the constraints are incorporated in the algorithm itself. We will see that the principal constraints are reflected in the algorithm, but that there are classes of examples which are not and could not easily be captured by the algorithm. Generally, for these, we will make the decidedly non-theoretical remark that no such examples occurred in the sample studied. This of course is not a justification for ignoring such examples when devising the naive algorithm. But in view of the fact that the algorithm fails on much more natural and common examples than these, there seems to be little point in greatly complicating the algorithm to handle them.

This section is not meant to be an exhaustive review of the literature on pronouns, and we especially do not wish to comment on formalisms that have been devised for generating pronouns within a base or transformational component. The references cited are intended to be representative, and are used principally for the interesting examples that they present.

Steps 2 and 3 of the algorithm deal with the level of the tree in which the pronoun occurs. A non-reflexive pronoun and its antecedent cannot occur on the same level below an NP or S node. In (7) and in

John's portrait of him

"him" and "John" cannot be coreferential. However, if an NP node is on

a lower level than the pronoun and precedes the pronoun, it is a possible antecedent (step 3). In

John's father's portrait of him (9)

and

After John robbed the bank, the police apprehended him (10)

"him" and "John" can be coreferential. Thus, the algorithm captures the constraint proposed by Lees and Klima (1963) and by Langacker (1969), and restated in an interpretive framework by Jackendoff (1972),<sup>3</sup> that a non-reflexive pronoun and its antecedent may not occur in the same simplex sentence.

Jackendoff has given a number of examples of reflexivization which occur beyond simplex sentences. Of course, the algorithm presented here is not responsible for dealing with reflexives. But it should avoid returning as antecedent an entity that could be rejected on the basis of an obligatory reflexivization rule. The algorithm fails on the class of picture noun examples discussed by Jackendoff, such as

John saw a picture of him (11)

John saw a picture of himself. (12)

In (11) the algorithm would interpret "him" as "John"; yet, it cannot be, for if they were to be coreferential, "himself" would have been used.

Jackendoff has given an analysis of how reflexives are to be interpreted beyond the scope of simplex sentences. Unfortunately, the corresponding rule for how non-reflexives are not to be interpreted is incorrect. For there are cases where either the reflexive or non-reflexive pronoun may be used. Consider

\* John<sub>i</sub> saw him<sub>i</sub>.

?\* John<sub>i</sub> saw a picture of him<sub>i</sub>.

?? John<sub>i</sub> saw a picture of him<sub>i</sub> hanging in the post office.

? John<sub>i</sub> saw that a picture of him<sub>i</sub> was hanging in the post office.

John<sub>i</sub> claimed that the picture of him<sub>i</sub> hanging in the post office was a fraud.

"Himself" is perfectly acceptable in place of "him" in all five sentences. But apparently the more deeply the pronoun is embedded and the more elaborate the construction it occurs in, the more acceptable the non-reflexive becomes. Yet there is no precise boundary between where it is acceptable and where it is not.

Rather than complicate the algorithm excessively, we will simply let it fail on cases like (11). In the statistical studies reported below, no such examples occurred.

There is a related class of examples where the algorithm can and should be improved. Consider

John said his mother would pay the man who shaved him.

The algorithm, working dumbly, would pick "the man" as the antecedent. Yet the coreferentiality of "the man" and the omitted subject of "shaved" and the non-coreferentiality of that subject and "him" are syntactically derivable. Hence, "the man" cannot be the antecedent. In what follows it will be assumed that such derived non-coreference relations are available to the algorithm. This is a reasonable assumption if pronoun resolution is viewed as taking place within a larger interpretation process which, working in a left-to-right manner, also recovers syntactically recoverable omitted material and records coreference and non-coreference relations.

This larger interpretation process allows the algorithm to deal with some of the examples of "missing antecedents" discovered by Grinder

and Postal (1971). For example, in

My uncle doesn't have a spouse but your aunt does and he  
is lying on the floor.

the interpretation process would first expand the second clause into

. . . but your aunt does have a spouse . . . .

The algorithm would then propose the aunt's spouse as the antecedent of "he". The interpretation process apparently should work in a left-to-right manner, for backward pronominalization cannot occur with a missing antecedent.

John didn't yell at his wife, but Bill did, especially  
after she wrecked their car. (13)

John didn't yell at his wife, but especially after she  
wrecked their car, Bill did. (14)

In (13) "she" can refer to Bill's wife, but in (14) it cannot.

Langacker (1969) proposed the rule that the antecedent of a pronoun must precede or command the pronoun, and Jackendoff (1972) recast the constraint for an interpretive framework. Steps 3 and 7 handle the case of antecedents preceding the pronoun in the same sentence. (9) and (10) are examples. However there are cases where these steps will yield incorrect antecedents. Consider

\* That John<sub>i</sub> was the best boxer in the world was (15)  
claimed by him<sub>i</sub>. (Kuno 1972)

\* In Mary<sub>i</sub>'s apartment, she<sub>i</sub> smokes pot. (Lakoff 1968) (16)

In (15) the algorithm would choose "John" as antecedent for "him", in (16) "Mary" for "she". Hinds (1975) has given a rather interesting explanation for (15) in terms of the functions of passivization and pronominalization. Sentences, he argues, tend to progress from material

which is more predictable from the context to material which is less predictable. The earlier parts of a sentence set the stage and tie things together with the previous context, while the later parts communicate the new or important information. The function of passivization is to move less predictable material--the identity of the agent--to the end of the sentence. Pronominalization applies because the identity of the entity is predictable. These two functions are contradictory in (15), making it bad. It is possible that a similar explanation will work for (16). The locative prepositional phrase has been fronted to provide a link with the previous context. In the supposedly more predictable first half of the sentence, Mary was identified explicitly. Yet in the less predictable second half of the sentence, her identity was predictable enough for pronominalization to apply.

It is difficult to see how the algorithm could be extended to handle such cases as these. This is not serious, for especially sentence (15) would not be very good even if "John" and "him" were not coreferential. We can expect few such sentences to occur, and in fact none did in the sample studied.

Ostensibly similar to (15) and (16) is the example given by Ross (1967):

\*Realizing that Oscar<sub>1</sub> was unpopular didn't disturb him<sub>1</sub>.

It would seem that the algorithm would pick "Oscar" as antecedent for "him". But if the left-to-right interpretation process is working, the omitted subject of "realizing" would already have been recovered. This subject's coreferentiality with "him" and its non-coreferentiality with "Oscar" are syntactically derivable.

Step 8 of the algorithm, which searches the tree to the right of the pronoun, handles cases in which the antecedent commands but does not precede the pronoun.<sup>4</sup> A node  $NP_1$  is said to command Node  $NP_2$  if neither  $NP_1$  nor  $NP_2$  dominates the other and if the S node which most immediately dominates  $NP_1$  also dominates  $NP_2$  (but not immediately). The command relation was proposed by Langacker (1969) to take care of backward pronominalization examples:

After he robbed the bank, John left town.

That he was elected chairman surprised John.

In step 8 there is a search downward in order to include examples like

That he<sub>i</sub> had done something terrible was disturbing to John<sub>i</sub>.

But the algorithm does not search below S and NP nodes because of the apparent unacceptability of

? That he<sub>i</sub> had done something terrible disturbed John<sub>i</sub>'s teacher.

\* That he<sub>i</sub> had done something terrible disturbed the teacher who punished John<sub>i</sub>.

This constraint will cause the algorithm to fail on several examples which have been discussed in the literature:

Mary sacked out in his<sub>i</sub> apartment before Sam<sub>i</sub> could kick her out. (Lakoff 1968, Culicover 1976)

Girls who he<sub>i</sub> has dated say that Sam<sub>i</sub> is charming. (Ross 1967)

However, this constraint never caused the algorithm to fail in the sample studied. If it were lifted, the performance of the algorithm would be degraded drastically.

The algorithm also handles the pronoun antecedent problem of the Bach-Peters paradox:

The boy<sub>i</sub> who deceived her<sub>j</sub> kissed the girl<sub>j</sub> who loved him<sub>i</sub>.  
 Of course it does not address the real problem of this sentence--the reference of the definite noun phrase. (See Bach (1970), Karttunen (1971), Dik (1973), and for an especially enlightening discussion, Hintikka and Saarinen (1975).)

In steps 3 and 7, the branches of the tree are searched in a left-to-right manner. Thus, in

John told Bill that he had been lucky.

"John" would be chosen over "Bill". This seems reasonable, because it is more common for subjects to be subsequently pronominalized than for objects. This choice was borne out in the sample studied.

What is at issue in steps 3, 4, 7, and 8 when we specify a "breadth-first" as opposed to "depth-first" search downward is the interpretation of texts like

John's mother told Bill she was angry. She wanted to  
 know where he had been.

A breadth-first strategy would pick "Bill", a depth-first "John", as antecedent of "he". Very few cases arose in the sample studied where this choice would make a difference, but in almost all of those few cases, the breadth-first strategy was the appropriate one.

Finally there is the problem of sentence pronominalization. In

Ford was in trouble, and he knew it  
 "it" refers to the fact that Ford was in trouble rather than any noun phrase. The algorithm as it stands does not handle such cases. It might be suggested that the algorithm be modified to accept an S node as well as an NP node dominating a sentential noun as the antecedent of a pronoun occurring in certain contexts. Cushing (1972) has examined some of these

contexts, viz. when the pronoun occurs as the object of a "stance" verb like "believe", "know", "suggest", or "doubt". However, the problem of avoiding spurious antecedents when S nodes are allowed would be quite severe. In

The newspaper reported that Ford had claimed the economy was improving, but I didn't believe it

the algorithm allowing both S and NP nodes would recommend the following as plausible antecedents, in the given order:

The newspaper reported that Ford had claimed the economy was improving

the newspaper

Ford claimed the economy was improving

the economy was improving.

This does not reflect at all one's intuitive feelings about which readings are preferred. Quite the opposite. A more discriminating ordering would require semantic processing, which is exactly what we are trying to avoid in the syntactic approach. Things get even worse when we consider sentences like (17), due to Lakoff (1968):

A Brinks' truck was robbed in the evening, but it (17)  
couldn't have happened during the daytime.

Such examples occur rarely in published writings--only once in the 300 examples examined--no doubt because of the editorial principle that every pronoun must have an antecedent.

2.4. Miscellaneous Assumptions. In the statistical studies summarized below it was first of all assumed that the correct parse was available. This is sometimes a rather dubious assumption because of



the well-known adjunct problem. This may be illustrated by the sentence

Archaeologists found small flint blades with parallel  
sides, struck serially from the cores, which had  
assumed conical shape by the time they were  
discarded. (Watson 19:25)

The correct parse attaches the relative clause "which had assumed . . ." to "blades", but there is no purely syntactic reason for not attaching it to "sides" or "cores". The naive algorithm will choose as antecedent of "they" whichever noun the relative clause is attached to, and will be correct if and only if the parse is correct.

"They": When seeking an antecedent for "they", the algorithm accepts plural noun phrases and collective singular noun phrases. In addition, the algorithm traverses the trees in the given order collecting entities which are selectionally compatible. For example, in

John sat on the sofa. Mary sat before the fireplace.

They faced each other.

the algorithm would pick "Mary" and "John" rather than "Mary" and "the fireplace". Also, when two plurals are conjoined, the conjunction is favored over either plural. In

Human bones and relics were found at this site. They were  
associated with elephant tusks.

"human bones and relics" is picked as the antecedent of "they" rather than either "bones" or "relics".

Quotes: In dialogue it is assumed that the implicit "A said to B . . ." has been recovered before the algorithm is applied to quoted sentences. It is also assumed the algorithm has available the rules that exclude the speaker and listener as possible antecedents of third

person pronouns inside quotes. In

John told Bill that he was lucky

"he" could refer to John, Bill, or someone else; in

John told Bill, "He was lucky."

it could only be someone else.

It should be pointed out that the algorithm and all the features described here, except the correct parse assumption, are well within current computational capabilities (Grishman, et al., 1973; Raze to appear).

2.5. Two Heuristics. Two heuristics which have been proposed for finding antecedents were also investigated. The first is a hypothesis put forward by Klapholz and Lockman (1975) that the antecedent is always found within the last  $\underline{n}$  sentences, for some small  $\underline{n}$ . Charniak (1972) was more explicit and proposed, with reservations,  $\underline{n}=5$ . This is clearly wrong. However, it is of interest to know how often this heuristic does hold and for what values of  $\underline{n}$  it holds often enough to be useful. It turns out that it is possible to make a much stronger statement than either Klapholz and Lockman or Charniak suggested. With  $\underline{n}$  "less than one", a very large majority of the antecedents will be found.

Let the candidate sets  $C_0, C_1, \dots, C_n$  be defined as follows:

$$C_0 = \left\{ \begin{array}{ll} \text{the set of entities in the} & \text{if pronoun comes} \\ \text{current sentence and the} & \text{before main verb} \\ \text{previous sentence} & \end{array} \right.$$

$$C_1 = \left\{ \begin{array}{ll} \text{the set of entities in only} & \text{if pronoun comes} \\ \text{the current sentence} & \text{after main verb.} \end{array} \right.$$

$$C_1 = \left\{ \begin{array}{l} \text{the set of entities in the current sentence and the} \\ \text{previous sentence.} \end{array} \right.$$

$C_n$  = the set of entities in the current sentence and the previous  $n$  sentences.

The frequency with which antecedents were found in  $C_0, C_1, \dots$ , is given below.

The second heuristic was used by Winograd (1972:159) in his blocks microworld system. It is

If the same pronoun occurs twice in the same sentence or in two consecutive sentences, the occurrences are coreferential.

The frequency with which this heuristic worked is also given below.

2.6. Statistical Results. One hundred consecutive examples of pronouns from three texts were examined. The pronouns were "he", "she", "it", and "they". "It" was not counted when referring to syntactically recoverable "that" clauses or occurring in time or weather constructions. The texts were William Watson's Early Civilization in China, pp. 21-69, the first chapter of Arthur Haley's novel Wheels, pp. 1-6, and the July 7, 1975 edition of Newsweek, pp. 13-19, beginning with the article "A Ford in High Gear".

Significant differences may be noted among the texts. Watson is characterized by long, grammatically complex sentences that exercised every step of the algorithm. Its underlying knowledge base is reasonably precise and highly structured. It tends to treat relations and comparisons and not to spend much time describing the behavior or qualities of a particular entity. There is a relative paucity of pronouns--48 pages had to be turned to find one hundred examples--and of the pronouns present, "it" and "they" predominate. It is probably highly typical of technical writing in general.

Wheels, on the other hand, is highly colloquial. The sentences are generally short and simple, frequently being no more than an exclamation. It is full of dialogue. Its underlying knowledge base is our large and ill-defined collection of facts about people, their motivations, and how they move about in American society. "He" is the most frequent pronoun, and large stretches of text trace out a single person's actions and thoughts.

Newsweek has a very rich verbal texture which mixes grammatical complexities and colloquialisms, and draws on a number of very different knowledge bases. There are conscious efforts at clever turns of phrase. It combines some of the worst difficulties of the other texts.

The results of the studies are summarized in tables 1-4.

We see that the candidate set hypotheses is a very strong one. 90% of all antecedents are in  $C_0$  while 98% are in  $C_1$ . Yet there is no useful absolute limit on how far back one need look for the antecedent. One antecedent occurred nine sentences before the pronoun. The pronoun "it", especially in technical writing, can have a very large number of plausible antecedents in one sentence--one example in Watson had thirteen. Any absolute limit we impose might therefore have dozens of plausible antecedents and would hardly be of practical value.

Overall, the algorithm worked in 88.3% of the cases. The algorithm together with selectional constraints worked 91.7% of the time. This is somewhat deceptive since in over half the cases there was only one plausible antecedent. For that reason, the number of examples in which more than one plausible antecedent occurred in the candidate set are tabulated, with the number of times the algorithm worked. Of 132 such conflicts, twelve were resolved by selectional constraints and 96 of

TABLE 1: Summary of Watson

	Number	$C_0$	$C_1$ but not $C_0$	Algorithm works	Algorithm works after Selection	Conflicts before Selection
he	7	7	0	7	7	2
she	0	0	0	0	0	0
it	40	40	0	31	34	34
they	53	50	3	47	51	40
Total	100	97	3	85	92	76

	Algorithm works	Conflicts after Selection	Algorithm works	Winograd applies/ works
he	2	2	2	1 / 0
she	0	0	0	0 / 0
it	26	30	26	7 / 3
they	34	36	34	12 / 6
Total	62	68	62	20 / 9

TABLE 2: Summary of Wheels

	Number	$C_0$	$C_1$ but not $C_0$	$C_2$ but not $C_1$	$C_3$ but not $C_2$	$C_9$ but not $C_8$	Algorithm works	Algorithm works after Selection
he	71	60	8	2	0	1	64	64
she	7	7	0	0	0	0	7	7
it	12	7	2	1	2	0	8	9
they	10	5	5	0	0	0	9	10
Total	100	79	10	3	2	1	88	90

	Conflicts before Selection	Algorithm works	Conflicts after Selection	Algorithm works	Winograd applies/ works
he	13	6	13	6	9 / 9
she	0	0	0	0	0 / 0
it	5	2	5	3	2 / 1
they	3	2	2	2	1 / 0
Total	21	10	20	11	12 / 10

TABLE 3: Summary of Newsweek

	<u>Number</u>	<u>C<sub>0</sub></u>	<u>C<sub>1</sub> but not C<sub>0</sub></u>	<u>Algorithm works</u>	<u>Algorithm works after Selection</u>
he	61	59	2	59	59
she	0	0	0	0	0
it	19	17	2	16	16
they	20	19	1	17	18
Total	100	95	5	92	93

	<u>Conflicts before Selection</u>	<u>Algorithm works</u>	<u>Conflicts after Selection</u>	<u>Algorithm works</u>	<u>Winograd applies/ works</u>
he	16	14	16	14	11/9
she	0	0	0	0	0/0
it	9	5	9	5	3/0
they	10	7	7	5	2/0
Total	35	26	32	24	16/9

TABLE 4: Summary of all three texts

	<u>#</u>	<u>C<sub>0</sub></u>	<u>C<sub>1</sub></u>	<u>C<sub>2</sub></u>	<u>C<sub>3</sub></u>	<u>C<sub>9</sub></u>	<u>Algorithm works</u>	<u>Algorithm works after Selection</u>
he	139	126	10	2	0	1	130	130
she	7	7	0	0	0	0	7	7
it	71	64	4	1	2	0	55	59
they	83	74	9	0	0	0	73	79
Total	300	271	23	3	2	1	265	275

	<u>Conflicts before Selection</u>	<u>Algorithm correct</u>	<u>Conflicts after Selection</u>	<u>Algorithm correct</u>	<u>Winograd applies/ works</u>
he	31	22	31	22	21 / 18
she	0	0	0	0	0 / 0
it	48	33	44	33	12 / 4
they	53	43	45	41	15 / 6
Total	132	98	120	96	48 / 28

the remaining 120 were resolved by the algorithm. Thus, 81.8% of the conflicts were resolved by a combination of the algorithm and selection.

If we look at the results for Watson as typical of the kind of technical article most likely to be analyzed in an automatic fact retrieval system, things look even more encouraging. There were a very high number of conflicts, 76 out of 100 examples, but selection resolved eight of these and the algorithm worked on 62 of the remaining 68. The combination thus yields the correct antecedent in 92% of the cases where there is conflict.

The Winograd heuristic did not perform as well. Out of the 132 conflicts, it was applicable 48 times, of which it returned the correct antecedent only 28 times, or 58.3%. On Watson it worked only nine times out of the twenty in which it applied, for 45%. It should be pointed out, however, that it worked very well on the highly colloquial Wheels --ten successes out of 12 applications. This is not surprising since Winograd proposed it in connection with a system for carrying on dialogues.

#### 2.8. Significance and Lack of Significance of These Results.

Whether we say we have a 92% algorithm, a 91.7% algorithm, or an 81.8% algorithm, the results show that the naive approach is very good. It will be a long time before a semantically based algorithm is sophisticated enough to perform as well, and these results set a very high standard for any other approach to aim for.

Yet there is every reason to pursue a semantically based approach.

First of all, the algorithm does not work. Any one can think of examples where it fails, examples more natural than those presented in Section 2.3. Several will be presented later in the paper. In these

cases the algorithm not only fails; it gives no indication that it has failed and offers no help in finding the real antecedent. What we have done already along this line is as much as we can do. The approach offers no hope of a total solution.

Second, one has a strong feeling that it is not explanatory. This is not necessarily true, for in fact the traversal order may reflect in some underlying way the sense people have of the degree of "active-ness" of the entities involved in a discourse. Yet the semantic approach outlined below seems intuitively more appealing since it depends on very fundamental properties of language.

Third, since it was only published texts that were studied here, there is the suspicion that what was studied was not pronoun use in English or even pronoun use in English writing, but the prejudices of editors. This would not make the study useless, but it would limit its value.

A final and very important reason for investigating the semantic approach described below is that it is all processing that must be done anyway in the analysis of texts.



### 3. The Semantic Approach

3.1. The System for Semantic Analysis of English Texts. A system is described here for the semantic analysis of English texts. It consists of certain semantic operations which access a large set of inferences or axioms expressing the knowledge of the world necessary for the understanding of texts. The operations draw just the inferences appropriate to the text at hand. Their goal is to recognize the structure and inter-relationships implicit in the text.<sup>5</sup>

The input to the semantic analyzer is the text, which we assume syntactic analysis or semantic interpretation rules have already reduced to a predicate notation which exhibits functional relationships. In addition, we will assume the syntactically derivable coreference and non-coreference relations have been detected and recorded; Any antecedent proposed by the methods described below is checked against these relations.

The text in predicate notation consists of (1) a set of entities-- $X_1, X_2, X_3 \dots$  representing the entities referred to in the text; (2) the set of kernel statements which describe properties of the entities by applying predicates, corresponding roughly to English words, to the entities; and (3) an indication of which statement in a sentence is asserted and which are grammatically subordinate, by means of the symbol "|", which is read "such that" or "where". For example, the sentence

The boy is on the roof of the building

would be represented (ignoring tense and definite article).

$\text{on}(X_1 \mid \text{boy}(X_1), X_2 \mid \text{roof}(X_2, X_3 \mid \text{building}(X_3)))$ .

(The  $X_1$  such that  $X_1$  is a boy is on  $X_2$  where  $X_2$  is the roof of  $X_3$  which

is a building.) The information content of this sentence consists of the statements

$$\text{on}(X_1, X_2), \text{boy}(X_1), \text{roof}(X_2, X_3), \text{building}(X_3).$$

In the course of semantic processing, the text is augmented by inferences which the semantic operations have determined appropriate to draw, and interrelated by merging entities standing for anaphors with the entities standing for their antecedents.

The semantic operations work by searching for chains of inference in a Lexicon, which is the store of world knowledge. Associated with each word or predicate in the Lexicon is a potentially large collection of facts, or inferences, which can be drawn from the use of that word. These facts are also expressed in the simple predicate notation. For example, stored with "bank" would be the fact that it is a building

$$(\forall y) (\text{bank}(y) \supset \text{building}(y)) \quad (18)$$

and with "building" the fact that it has a roof

$$(\forall y) (\exists x) (\text{building}(y) \supset \text{roof}(z, y)). \quad (19)$$

The general form of the inferences is

$$(\forall y) (\exists z) (p(y) \supset (q(y, z) \supset r(y, z))) \quad (20)$$

where  $p$  is the word or predicate with which the inference is associated,  $y$  represents its explicit parameters,  $z$  stands for the entities whose existence is also implied, and  $q(y, z)$  represents the other enabling conditions which must be checked before the conclusions  $r(y, z)$  can be drawn. If  $p(X_1)$  occurs in the text and (20) is determined to be appropriate, then  $q(X_1, X_i)$  is looked for in the text for some entities  $X_i$ , and if it is found, the conclusion  $r(X_1, X_i)$  is drawn by adding it to the text. As a matter of computational efficiency it is usual to have no enabling conditions  $q(y, z)$ . Thus, the world knowledge in the Lexicon is keyed by individual words.

Inferences are not drawn freely, but only in response to the specific demands of semantic operations. These demands take two forms:

Forward inferences: From  $p(X_1)$  try to infer something of the pattern  $r(z_1, z_2)$ .

Backward inferences: Find something in the text from which  $p(X_1)$  could be inferred.

Since the Lexicon is potentially quite large, the facts are divided into clusters roughly according to topic. The clusters are given an initial measure of salience according to their anticipated relevance to the text at hand. The measures of salience are modified in the course of semantic processing in response to changes in topic in the text in the following way: When a fact in a cluster is used, the entire cluster is given maximum salience; while the facts in a cluster are not being used, its salience decays. All searches of the Lexicon initiated by the semantic operations are conducted in cluster order.

It may seem at first glance that this device is purely for efficiency. But in fact it is a crucial part of the analysis mechanism. When an operation calls for a chain of inference, there are usually quite a few chains of inference possible, each of which may lead to a different interpretation or a different antecedent. The changing salience measures on the axioms together with the lengths of the chains of inference define a dynamic ordering on the set of chains of inference. The operation then picks the appropriate chain of inference which is first in this ordering at the time the operation is invoked. In this way the inferencing process and hence the interpretation it produces are made highly context dependent.

3.2. The Semantic Operations and Pronoun Resolution. The basic principle of semantic analysis in the system is Joos's Semantic Axiom Number One (Joos 1972), or what might be called the Principle of Knitting. This may be stated

The important facts in the text will be repeated, explicitly  
or implicitly.

That is, we capitalize on the very high degree of redundancy that characterizes all natural language texts. This redundancy is the active principle in understanding natural language. All of the semantic operations are in one way or another realizations of this principle.

There are four primary semantic operations:

1) Detect or Verify the Intersentence Connectives: This is done by comparing the current sentence and the previous text against a small set of common patterns. These patterns are stated in terms of inferences to be drawn from the current sentence and an "eligible" previous sentence, and the modification to be performed on the text if the pattern is matched. The patterns are of two kinds--coordinating and subordinating. Among the coordinating relations are Temporal Succession, Cause, Enablement, Contrast, Parallel, and Paraphrase. One subordinating relation is Example.

As analysis proceeds, a tree-like structure is constructed for the paragraph, with subordinating links building the tree downward and coordinating links building it to the right. A previous sentence is "eligible" if it is on the right frontier of this "tree". For example, in a text  $S_1 S_2 S_3$ , if  $S_2$  is subordinated to  $S_1$ , then  $S_3$  may be coordinated with either  $S_1$  or  $S_2$  (Fig. 3a), but if  $S_2$  is coordinated with  $S_1$ , then  $S_3$  may be coordinated only with  $S_2$  (Fig. 3b).



Fig. 3a

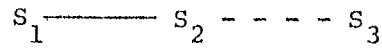


Fig. 3b

The presence of certain conjunctions causes some patterns to be preferred--"and" promotes Temporal Succession and a repetition of the previously recognized pattern, a dash and "i.e." promote Paraphrase, and "but" promotes Contrast.

One variety of Contrast pattern might be stated as follows:<sup>6</sup>

From the assertions of the current sentence and an eligible previous sentence, try to infer statements  $S_1$  and  $S_2$  where

- a) the predicates of  $S_1$  and  $S_2$  are contradictory or lie at opposite ends of some scale;
- b) one pair of corresponding arguments of  $S_1$  and  $S_2$  are identical;
- c) the other pairs of corresponding arguments are "similar" but different.

An example is shown in Section 4.2 in which detecting this pattern leads to pronoun resolution.

2) Predicate Interpretation: Certain predicates make demands on the nature of their arguments. This is particularly true of adverbials and relational terms. This operation seeks to satisfy those demands.

Consider, for example, the phrases "walk out", "walk slowly", and "pleasant walk". The predicates "out", "slow", and "pleasant" all apply to the action of someone's walking, but they each narrow in on different aspects of walking. That is, each demands that a different

inference be drawn from the statement that "X walks". "Out" and "slow" demand their arguments be motion from one place to another, forcing us to infer from "X walks" that "X goes from A to B". "Out" then adds information about the locations of A and B, while "slow" says something about the speed of this motion. "Pleasant", on the other hand, requires its arguments to be an awareness, so we must infer from "X walks" that "X engages in a bodily activity he is aware of". "Pleasant" then says something about the quality of this awareness.

Stored in the Lexicon with many predicates are the inferences which must be drawn from their arguments and the information they add to these inferences. For example, "go( $z_1, z_2, z_3$ )" ( $z_1$  goes from  $z_2$  to  $z_3$ ) must be inferred from the argument of "out". When the statement "out(walk( $X_1$ ))" is encountered in the text, the predicate interpretation operation makes efforts to find a proof of "go( $z_1, z_2, z_3$ )" from "walk( $X_1$ )". The facts in the resulting chain of inference are instantiated, together with the information added by the original predicate.

This operation allows considerable compression in the number of senses that must be stored for each word. "Slow", for example, can be defined as something like "Find the most salient associated motion. Find the most specific Speed Scale for the object X of this motion. X's speed is on the lower end of this scale." This definition is adequate for such phrases as "walk slowly" (the most salient motion is the forward motion of the walking), "slow race" (the forward motion of the competitors), "slow horse" (its running at full speed, generally in a race), "slow watch" (the motion of the hands), "slow business" (the metaphorical motion of the outflow of goods and the inflow of money), and "slow person". This last case is highly dependent on context, and

could mean the person's physical acts in general, his mental processes, or the act he is engaged in at the moment.

This operation also allows us to recover omitted material such as the part from the whole:

He landed on (the roof of) the building  
because "on" demands a horizontal surface for its object; and missing quantity words:

(The price of) Meat is higher this month  
since "higher" requires a real or metaphorical vertical scale.

This operation has a default feature. If a proof of the required inference can't be found, the inference is drawn anyway. At worst it will be instantiated with new and unknown entities for its arguments. The next operation will normally correct for this.

Predicate interpretation frequently aids in pronoun resolution, as will be seen in the next few paragraphs, and in Section 4.1.

3) Knitting: When a statement is instantiated whose predicate is identical to that of a statement already in the text, then the first guess we make is that this is a redundancy. That is, redundancies are occurrences we expect to happen all the time. If no obvious inconsistency is found, we assume the two statements are the same, and we merge them, thereby merging the corresponding arguments.

Knitting frequently leads to the antecedents of pronouns and implicit entities being found before operation 4, whose work it is to find antecedents, is even called. For example, suppose the following text is being processed:

The boy walked into the bank. Moments later he was seen  
on its roof.

Let the bank be represented by the entity  $X_1$  and the "it" of "its" by  $X_2$ . Predicate interpretation on "into" forces us to verify that its object,  $X_1$ , is an enclosure. This could be done by using the fact (18) that a bank is a building and the fact that a building is an enclosure. Since the entire chain of inference is instantiated the statement

$$\text{building}(X_1) \quad (22)$$

becomes part of the text. The second sentence is processed, and predicate interpretation on "roof" demands verification that  $X_2$  is a building. This cannot be verified by the other properties of  $X_2$  since it has no other properties. Therefore the default feature applies and

$$\text{building}(X_2) \quad (23)$$

is simply assumed. The redundancy of (22) and (23) is presumed, knitting occurs, and the two statements and thereby their corresponding arguments  $X_1$  and  $X_2$  are merged. Thus has the antecedent of "it" been identified as "the bank".

4) Identifying entities: This operation seeks to identify the so far unidentified entities. Entities referred to in a text may be arranged in a hierarchy according to their degree of specification:

- a. proper names, including "you" and "I";
- b. other noun phrases, including those with definite, indefinite, and demonstrative articles;
- c. third person pronouns;
- d. zeroed arguments and implied entities.

When a proper noun is encountered, it is identified with any entity in the previous text described by the same proper noun, or if there is none, a new entity is introduced. When a noun phrase tagged by the indefinite article is encountered, a new entity is introduced. The



identification procedure for definite noun phrases is described in Hobbs (1975). Its search step is similar to the search step described below for pronouns.

In order to find the antecedent of a pronoun a backward search through the Lexicon is conducted for a chain of inference that begins at some statement in the text and ends with a known property of the pronoun. Suppose example (21) had escaped the first three operations, and a search for the antecedent of "it" was necessary. The only known property of  $X_2$  is that it has a roof. The Lexicon is probed to see what has a roof, the fact (19) stored with "building" that buildings have roofs is found, and the text is checked for an occurrence of "building". Assume "building( $X_1$ )" has not yet been inferred from "bank( $X_1$ )", so that "building" is not found. The Lexicon is then searched for something which is a building, the fact (18) associated with "bank" that a bank is a building is found, and a bank is mentioned explicitly in the text. The required chain of inference

$$\text{bank}(X_1) \supset \text{building}(X_1) \supset \text{roof}(X_3, X_1)$$

is found. Hence the antecedent of "it" is located.

The difficulty with this search is that it is very expensive. It requires exponential time and the branching factor of the search can be very large. For example, there could be a great many facts in the Lexicon which say that something is a building; gas stations, post offices, dime stores, etc. are all buildings. Therefore in order to cut down on the size of the search, and at the same time to take advantage of the effectiveness of the naive algorithm of Part 2, a bidirectional search (Pohl 1971) is used which starts not only at the pronoun but also at the entity the naive algorithm would choose as

antecedent, in hopes that the two searches will meet somewhere in the middle. Thus, in about 90% of the cases, the search will go quite fast.

Once a plausible antecedent is found, a check is made to insure that the properties of the two entities to be merged are consistent. In practice this is a rather shallow check to verify that there is no obvious inconsistency.

More than one plausible antecedent may be found by the search. If so, we appeal to the Principle of Knitting and choose the candidate that maximizes the redundancy in the simplest possible way. This is done by inferring freely from the properties of the candidates and of the pronoun and picking the candidate that has the most properties of high salience in common with the pronoun.

#### 4. Examples of the Semantic Approach to Pronoun Resolution

In Part 4, four examples are presented in which the semantic approach described in Part 3 results in the antecedent being found. The first example comes from the archaeology text and illustrates knitting working in conjunction with predicate interpretation. The second example, from Newsweek, shows the semantic approach working because of the intersentence operation, where the naive algorithm would fail. The last two are classic examples from Winograd and Charniak.

4.1. The first example is the sentence from the archaeology text

The plain was reduced by erosion to its present level.

or, in predicate notation,

reduce (erode(X1), X2 | plain(X2), X3 | present(level(X3,X4)))

i.e. something, X1, eroding reduces X2 which is a plain to X3 where X3's being the level of X4 is true at present. We must identify the antecedent not only of "it" (X4) but also of the implicit entity which is eroding (X1). (For the sake of simplicity, this sentence has been reduced from

A low hill which rises a short distance north of Chou K'ou Tien represents a small area of limestone which was left above the plain as the latter was reduced by erosion to its present level, some 60 metres below the present top of the limestone tump. (Watson 1966:21)

The real example has five more plausible antecedents, but the solution would be substantially the same as that given here.)

Note that syntactic criteria do not solve the problem, for in the sentence

Walter was introduced by John to his present wife  
 "his" could refer to Walter, John, or someone else. Selectional cri-  
 teria will not work either, for "erosion" can have a present level, as  
 in

Contour farming has reduced erosion to its present level.

Consider now what happens in the course of semantic processing.  
 We apply predicate interpretation to "reduce". This predicate demands  
 of its second argument that it be capable of movement along some verti-  
 cal axis (real or metaphorical), leading us to infer that a plain, being  
 a land form, is characterized by an altitude, i.e. a position on the  
 real vertical axis going from the center of the world outward. "Reduce"  
 then adds the information that a change downward to a third argument  
 X3 has occurred on this axis:

$$\begin{aligned} & \text{become}(\text{at}(X2, X5), \text{at}(X2, X3)) \mid \text{exceed}(X5, X3, X7 \mid \text{Attitude-axis} \\ & \quad (X7)), \text{vertical}(X7) \qquad \qquad \qquad (24) \end{aligned}$$

i.e. the plain X2 at X5 becomes X2 at X3 where X5 exceeds X3 on the  
 Altitude axis X7.

Next the arguments of "reduce" are processed in turn. From the  
 argument X1 of "erode" we must also infer that it is capable of move-  
 ment along a real or metaphorical vertical axis. "Erode" also says this  
 movement is in a downward direction. X1 has no explicit properties,  
 so we cannot infer a vertical axis. Hence we simply assume one to exist.

$$\begin{aligned} & \text{become}(\text{at}(X1, X8), \text{at}(X1, X9)) \mid \text{exceed}(X8, X9, X10 \mid \\ & \quad \text{vertical}(X10)) \qquad \qquad \qquad (25) \end{aligned}$$

Since (24) and (25) are identical except for the temporary entities,  
 and since no contradiction could be derived if we identified the tem-  
 porary entities, the knitting operation applies, and the implicit

entity which is eroding, X1, is identified with the plain, X2, X8 is identified with X5, X9 with X3, and the vertical axis X10 with the Attitude axis X7.

When the third argument of "reduce" is processed, we first apply predicate interpretation to "present". "Present" carries with it the implication that what it describes--X3 being the level of X4--resulted by a "becoming" from some previous state. "Level" demands that its first argument be a point on a vertical scale, and that its second argument be at that point. Thus, we infer

become(X11, at(X4,X3 | on(X3,X12 | vertical(X12))))

i.e. the state X11 changes into the state in which X4 is located at X3 which is a point on a vertical scale X12. This is identified by knitting with (24), thereby identifying X11 with "at(X2,X5)", X3 ("it") with X2 ("the plain"), and vertical axis X12 with the Attitude axis X7.

When at last we invoke operation 4, we find that all entities have been identified except the anaphoric definition noun phrase "the plain".

4.2. The next example comes from Newsweek and illustrates how the intersentence operation aids in pronoun resolution.

The FBI said they had tentative identifications on the fugitives,  
but didn't know where they were.

We wish to find the antecedent of the "they" in the "but" clause.

The naive algorithm does not work on this example. The first entity it would propose is the omitted subject of "didn't know", i.e. the FBI, as if the FBI were lost in a forest and didn't know where they were. Next it would light upon "tentative identifications", as if the FBI had identifications but a clerk had misfiled them. Only at last would the correct antecedent, "fugitives", be reached.

In examining the semantic approach, assume we have the correct parse, conjoining the "but" clause with "they had tentative identifications . . . ." Assume also that they "they" of the first clause and the omitted subject of the second have been recognized as coreferential with "the FBI". The conjunction "but" makes the Contrast pattern the one we try hardest to match. From "the FBI had tentative identifications on the fugitives" we can infer "the FBI had tentatively identified the fugitives". From this can be inferred "the FBI (tentatively) knows the names of the fugitives." This is compared with the assertion of the "but" clause, which paraphrased is "the FBI does not know the location of X". We find that the predicates are contradictory, and the first arguments are the same. Therefore the Contrast pattern will be matched if the second arguments--"the names of the fugitives" and "the location of X"--can be shown to be similar. This can be accomplished simply by assuming "they" and "the fugitives" to be the same.

The search step, operation 4, would locate the antecedent, even if the first three operations failed. The following chain of inference would be discovered:

From "fugitive(X1)" infer "hide-from(X1,X2 | police(X2))". (26)

From (26) infer "cause(X1,not(know(X2,location(X1))))". (27)

From (27) infer "not(know(X2,location(X1)))".

(If something is caused, it holds.) But this is just the property we know about "they".

4.3. The next example is from Winograd (1972:33). Consider

They<sub>1</sub> prohibited them<sub>2</sub> from demonstrating because (28)

they<sub>3</sub> feared violence.

They<sub>1</sub> prohibited them<sub>2</sub> from demonstrating because they<sub>3</sub> advocated  
 violence. (29)

"They<sub>3</sub>" is coreferential with "they<sub>1</sub>" in (28), but with "them<sub>2</sub>" in (29).

In (28), the intersentence operation will seek to link the two clauses, and on account of the conjunction "because" a match with a Cause pattern will be the most sought. One might like to say the Cause pattern is

Find a causal chain from the purported cause to the purported  
 effect,

in this case, from the second clause to the first. But in fact this cannot be done in (28), since a required precondition to someone prohibiting something is that he have the authority to do so, and this cannot be shown from the second clause. Thus a weaker statement of the Cause pattern might be.

Find a causal chain from a prominent fact inferable from the  
 purported cause to a prominent fact inferable from the  
 purported effect.

In a sense we are required to establish the causal link between supersets containing the items rather than between the items themselves. This allows us to recognize as having valid causal links, texts which are not just instantiations of potential theorems in the system.

For the sake of this discussion we will coin a word "diswant", analogous to "dislike". To diswant S is to want not-S. From "x prohibits y" we can infer "x diswants y" and from "x fears z" we can also infer "x diswants z". We will try to establish a causal link from "they<sub>3</sub> diswant violence" to "they<sub>1</sub> diswant (they<sub>2</sub> demonstrate)".

A prominent fact about demonstrations, well-known to anyone aware in the 'sixties, is that frequently

(w demonstrate) cause violence.

A fundamental fact relating the real world with mental and emotional worlds is

$$(x \text{ cause } y \ \& \ z \text{ diswant } y) \text{ cause } (z \text{ diswant } x). \quad (30)$$

This has the interesting effect of transforming a causal link between two events in the real world into the reversed causal link between mental states in which these events are apprehended.

Now the establishment of the causal link is straightforward. In fact, if we are willing to be a bit cavalier about the distinction between causality and logical implication, a three-step resolution proof suffices. In Figure 4, A is "they<sub>1</sub>", B "them<sub>2</sub>", and V "violence", and all are constants. t is "they<sub>3</sub>" and a variable. The unification that permits the final resolution is equivalent to identifying the antecedent of "they<sub>3</sub>" as "they<sub>1</sub>".

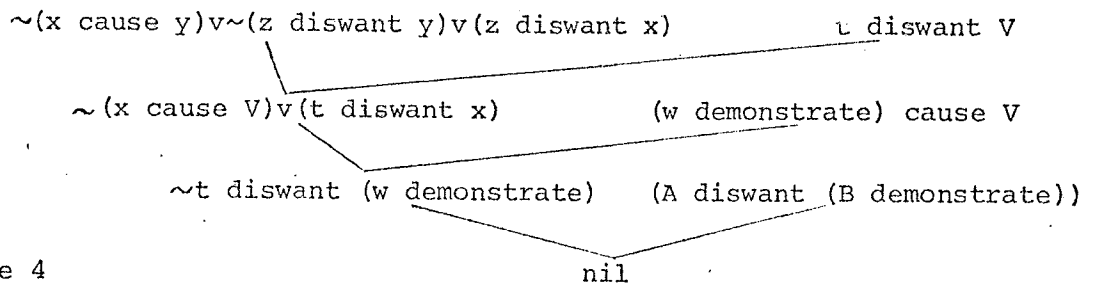


Figure 4

To match the Cause pattern in (29) we seek a causal chain from the second clause to the statement inferable from the first clause that "they<sub>1</sub> diswant (they<sub>2</sub> advocate something)". The chain is on the order of the following: Someone advocating something often causes that something to occur. In particular, they<sub>3</sub> advocating violence may cause violence to occur. Normally, someone, in particular they<sub>1</sub>, will diswant



violence. Therefore, by (30), they<sub>1</sub> will diswant they<sub>3</sub> advocating violence. If we identify they<sub>2</sub> and they<sub>3</sub>, the Cause pattern is matched.

The purpose of these examples is not to advocate resolution theorem-proving, but to show that the required causal chains are readily derivable. In fact, with a large Lexicon, i.e. a large number of axioms; the proof procedure must be much more highly directed than in standard resolution theorem proving, even on an example this small.

The pronoun problem of (28) and (29) could also be solved by proposing both "they<sub>1</sub>" and "them<sub>2</sub>" as plausible antecedents and holding a competition to see which choice maximized redundancy. In (28), the redundancy between "they<sub>1</sub> diswant . . ." and "they<sub>3</sub> diswant . . ." would result in "they<sub>1</sub>" being chosen. While it is true that there are fears associated with demonstrating, e.g. fears of tear gas or fears of the consequences of what one is demonstrating against, these fears are not nearly as salient as the fears associated with prohibiting.

In (29), the prominent fact about demonstrating that someone who demonstrates advocates something would make "them<sub>2</sub>" the choice that maximized redundancy.

One of the most curious aspects of this problem is that it has gone under the name of the "city council" problem. This is because the example was originally stated

The city councilmen refused the demonstrators a permit because  
they feared violence.

This has misled many to believe the problem has something to do with city councils. For example, Winograd has said,

We understand this [i.e. the difference between (28) and (29)]  
because of our sophisticated knowledge of councilmen,

demonstrators, and politics. (Winograd 1972:33)

Of course, the reference can be switched around by loading the context properly. In

The reactionary Republican city councilmen prohibited the lawless motorcycle freaks from demonstrating, because they advocated law, order, domestic tranquility, and the repression of free speech.

"they" probably refers to the city councilmen. But in the terms of our semantic approach, this is because such loading alters the salience of facts in the Lexicon, making a different causal chain more likely.

4.4 The final example, from Charniak (1972, 1974), is in the style of a children's story:

Jack invited Janet to his birthday party.

Janet wondered if Jack would like a kite.

But Bill said Jack already had a kite.

Jack would make her take it back.

The question is--how do we know "it" refers to the kite she is thinking about buying and not the more recently mentioned kite Jack already has.

This is a difficult text, but it is important to be clear about where the difficulty lies. For it does not lie in the pronoun resolution problem in the last sentence. It lies in determining the relation between the first two sentences. And this difficulty is reflected in a slight but perceptible discontinuity the reader senses at that point.

The relation between the two sentences is causal. As in the previous example, a causal relation can be established if a causal chain, built of causal relations stored in the Lexicon, can be found. For

these sentences it might go as follows: Jack inviting Janet to the birthday party causes her to want to come to it. (31) is a "want" version of (30), which may be proposed as a general rule and which is applied here several times.

If y enables or is required for z, then x wanting z           (31)  
causes x to want y.

There are several requirements for a guest at a birthday party. She must give the host a present. To be a present something should be new and it should be liked by the recipient. For it to be new one has to buy it. Now if Jack would like a kite and the present were a kite, then Jack would like the present. Therefore her wanting Jack to like the present causes her to want to know (i.e. wonder) if Jack would like her to buy a kite.

Regardless of the particular facts used and the particular causal chain found, we must somehow flesh out the second sentence to something like

Janet wondered if Jack would like the kite if she bought one  
for him for a birthday present.

Once this is accomplished, pronoun resolution becomes a straightforward matter. We simply take the word "back" seriously.<sup>7</sup> For there to be motion back there must have been motion to. Therefore, in performing predicate interpretation on "back", we look for the most salient motion in the previous text, preferably involving the same agent and object as the motion modified by "back". We find the motion of Janet's buying the kite, and therefore identify "it" with the kite she considers buying.

It is also true that if Jack already has a kite, he at some point bought it. But this fact has not been required in the understanding of the text so far and hence is less salient. Furthermore, the agent of that buying is not the same as the agent in the phrase "her take it back", while the agent of Janet's buying is. If we change the third sentence into

But Bill said Jack had just bought a kite  
it becomes slightly more plausible that the kite referred to is the kite he has, but only slightly, for the agents still differ. If we change the last two sentences to

But Bill said Jack's mother had just bought him a kite.

Jack would make his mother take it back.

the preferred antecedent changes to the kite he already has.

It might be objected that a more salient previous motion is the motion involved in Janet's giving Jack the present. Insofar as this is true, it changes the interpretation of the last sentence to the not implausible

Jack would give it back to her.

However, militating against this interpretation is the influence of the word "make". It seems unlikely that coercion would be necessary for so simple an act as opening one's hands and receiving a harmless object in them.

The intersentence operation is also relevant. The contrast pattern between the second and third sentences, signalled by "but", is verified by noting that the expectation that Jack would like the kite is violated. From this the unacceptability of the kite as a gift can be inferred, and this is the cause whose effect is the return of Janet's

kite in the fourth sentence. Charniak (1974) has laid out in painful detail much knowledge that might go into discovering these intersentence relations.

Another minor aspect that might contribute to Janet's kite being chosen as antecedent is the fact that the kite exists in the hypothetical world of Janet's wondering. Jack's kite exists in the real world. "It" occurs in the hypothetical world dominated by "would". Thus, maximum redundancy is served if we assume "it" to be the hypothetical kite.

## 5. Summary

We have proposed three steps for pronoun resolution.

1. The intersentence relation operation together with knitting.
2. Predicate interpretation together with knitting.
3. The bidirectional search through the Lexicon, using the naive algorithm.

To these two more steps can be added as a fail-safe mechanism:

4. For all the entities of the correct gender in the current sentence and the previous sentence, hold a competition to maximize the redundancy as in Operation 4 of Section 3.2. At least we know the antecedent is there in 98% of the cases. Heretofore, this has been the best solution offered for the pronoun resolution problem.

5. Apply the naive algorithm. This has the advantage that it always gives an answer.

Normally, pronoun resolution will be accomplished by the first two steps. On reflection, this should not be surprising. For the reason a speaker uses a pronoun is precisely because the identity of the entity is obvious without description to anyone who is understanding what is being said, and the first two semantic operations capture, in large measure, what is meant by understanding.

## NOTES

1. I am indebted to Eileen Fitzpatrick for calling this to my attention.
2. If we were looking for head nouns rather than NP nodes, steps 6 and 7 would have a more elegant statement:

Search downward from X on all paths to the left of p  
for a noun.

3. From a computational point of view, Jackendoff's treatment could profit from one modification. He produces a potentially very large number of "optional" coreference readings, rather than simply recording the non-coreference relations and leaving coreference vague, for the semantic component to resolve. For example, on syntactic grounds a sentence with four pronouns, like

He told his brother that he could use his car  
has 15 possible optional readings, most of which can be forced by the appropriate context. A sentence with five pronouns, like

He said he told his brother that he could use his car  
has 41 possible readings. There are sentences, like

John gave Bill a picture of himself  
in which there are exactly two possible antecedents of "himself". Here the two possible readings should be generated. But this is quite different from the situation in

John gave Bill the keys to his car  
where the antecedent could be John, Bill, or any one of an indefinite number of males occurring in an embedding context.

4. A curious circularity occurs here. In

After he<sub>1</sub> robbed the bank, he<sub>2</sub> left town  
The algorithm will choose "he<sub>2</sub>" as antecedent of "he<sub>1</sub>" and "he<sub>1</sub>"

as antecedent of "he<sub>2</sub>".

5. We have chosen to describe the mechanism for semantic analysis in process-oriented terms. That is, we speak of it as a "system", the system consists of "operations", the operations perform "searches" and "draw inferences". These terms are used because they seem to be the most natural. There may be some who think that this means we are not presenting a formalism and that therefore the work is of more practical than theoretical interest. To those, we would simply point out that, as shown by Hobbs & Rosenschein (to appear), there is a straightforward correspondence between expressions in intensional logic and programs in LISP. Thus everything described in Part 3 could be cast into the formalism of intensional logic. To do so would have been quite involved and very unrewarding. It would have produced no insights into the nature of language and would have made the exposition next to impossible to follow. We trust that the interested reader will be able to formalize what is presented in the way and to the degree he sees fit.
6. The general contrast pattern may be as follows: If we call the predicate and arguments of a statement "elements" then
  - 1) one pair of corresponding elements lie at opposite ends of some scale;
  - 2) the other pairs of corresponding elements are identical or can be identified as being members of some small set (i.e. they are similar);
  - 3) there are enough differences in the other pairs to keep the two statements from being contradictory.
7. Charniak (1974) has noticed this and writes about it at great



length. His treatment suffers, however, because his theory lacks a notion of salience and the idea of a predicate making demands on its arguments.

#### ACKNOWLEDGEMENTS

The author wishes to thank Eileen Fitzpatrick for her helpful suggestions. This work was supported by the CUNY Faculty Research Award Program Grant No. 11233.

## BIBLIOGRAPHY

- Bach, E., 1970. Problominalization, *Linguistic Inquiry* 1, 121-2.
- Bolinger, D., 1973. Ambient it is meaningful too, *Journal of Linguistics* 9, 261-70.
- Charniak, E., 1972. Toward a model of children's story comprehension, AI TR-266, Massachusetts Institute of Technology Artificial Intelligence Laboratory.
- Charniak, E., 1974. 'He will make you take it back': A study in the pragmatics of language, Working Paper 5, Institute for Semantic and Cognitive Studies.
- Chomsky, N., 1970. Remarks on nominalization, In: R. Jacobs and P. Rosenbaum (eds.), *Readings in transformational grammar*, Waltham, Mass., Blaisdell, 184-221.
- Culicover, P., 1976. A constraint on coreferentiality, *Foundations of Language* 14, 109-118.
- Cushing, S., 1972. The semantics of sentence pronominalization, *Foundations of Language* 9, 186-208.
- Dik, S., 1973. Crossing coreference again, *Foundations of Language* 9, 306-26.
- Grinder, J., and P. Postal, 1971. Missing antecedents, *Linguistic Inquiry* 2, 269-312.
- Grishman, R., N. Sager, C. Raze, and B. Bookchin, 1973. The linguistic string parser, *Proceedings of the 1973 Computer Conference, AFIPS Press*, 427-34.
- Hinds, J., 1975. Passives, pronouns, and themes and rhemes, *Glossa* 9, 79-106.

- Hintikka, J. and E. Saarinen, 1975. Semantical games and the Bach-Peters paradox, *Theoretical Linguistics* 1, 1-20.
- Hobbs, J., 1975. A general system for semantic analysis of English and its use in drawing maps from directions, *American Journal of Computational Linguistics*, microfiche 32.
- Hobbs, J. and S. Rosenschein, Making computational sense of Montague's intensional logic, to appear.
- Jackendoff, R. 1972. *Semantic interpretation in generative grammar*, Cambridge, Mass., MIT Press.
- Jespersen, O., 1954. *A modern English grammar on historical principles*, part VII: *Syntax*, London, Geo. Allen and Unwin Ltd.
- Joos, M., 1972. Semantic axiom number one, *Language* 48, 257-65.
- Karttunen, L., 1971. Definite descriptions with crossing coreference, *Foundations of Language* 7, 157-82.
- Klapholz, D., and A. Lockman, 1975. Contextual reference resolution, *American Journal of Computational Linguistics*, microfiche 36.
- Kuno, S., 1972. Pronominalization, reflexivization, and direct discourse, *Linguistic Inquiry* 3, 161-95.
- Lakoff, G., 1968. *Pronouns and reference*, Unpublished paper.
- Langacker, R., 1969. On pronominalization and the chain of command, In: D. Reibel and S. Schane (eds.), *Modern studies in English*, Englewood Cliffs, Prentice-Hall, 160-86.
- Lees, R., and E. Klima, 1963. Rules for English pronominalization, *Language* 39, 17-28.
- Olson, D., 1970. Language and thought: aspects of a cognitive theory of semantics, *Psychological Review* 77, 257-73.

- Pohl, I., 1971. Bi-directional search, In: B. Meltzer and D. Michie (eds.), Machine intelligence 6, New York, American Elsevier, 127-40.
- Raze, C. A computational treatment of coordinate conjunctions, to appear in American Journal of Computational Linguistics.
- Rieger, C., 1974. Conceptual memory: A theory and computer program for processing the meaning content of natural language utterances, Stanford AIM-233.
- Ross, J., 1967. On the cyclic nature of English pronominalization, In: To honor Roman Jakobson, The Hague, Mouton, 1669-82.
- Watson, W. 1966. Early Civilization in China. New York, McGraw-Hill.
- Watt, W., 1973. Late lexicalizations, In: J. Hintikka et al. (eds.), Approaches to natural language, Dordrecht-Holland, D. Reidel Pub. Co., 475-89.
- Watt, W., 1975. The indiscreteness with which impenetrables are penetrated, *Lingua* 37, 95-128.
- Wilks, Y., 1975. A preferential, pattern-seeking, semantics for natural language inference, *Artificial Intelligence* 6, 53-74.
- Winograd, T., 1972. Understanding natural language, New York, Academic Press.