Frank G. Pagan

CONSTRUCTIBLE REPRESENTATIONS FOR TWO SEMANTIC RELATIONS

1. Introduction

This paper is concerned with semantic (or lexical) structures of the type used in programs which involve natural language processing, such as question-answering systems. These structures, which generally take the form of graphs representing semantic relations defined on word senses, must satisfy some rather self-evident requirements relating to their linguistic significance and adequacy and to their suitability for computational implementation. In addition, if the lexical subsets in question are to be nontrivial in size, the structures must be *constructible* in some systematic, consistent way, preferably with the aid of a computer. The structures which have been used in existing experimental systems, such as those reported in M. R. Quillian (1969) and R. M. Schwarcz, J. F. Burger, R. F. Simmons (1970), have generally been very restricted, and it has been argued (S. Y. Sedelow, 1972) that it is their lack of constructibility which has precluded the possibility of extending them.

The aim of this paper is to describe two examples of the use of constructibility as a design criterion for semantic structures. The semantic relations of hyponymy and compatibility are introduced in the following section, and suitable representations for them are developed in sections 3 and 4, respectively. Throughout the discussion, the constructibility of a representation is to be interpreted as its amenability to the use of a computational discovery algorithm which would build the structure and have the properties of semi-automatic operation, simple input data, efficiency in the quantity of input, consistency maintenance, and monotonic refinement of the growing structure. The meaning of this terminology will be made more clear in the course of the discussion.

It is far beyond the scope of the paper to consider the development and implementation of practical programs for building large semantic structures. It should also be emphasized that the aim is not the development of linguistic theories or the presentation of *particular* lexical structures. While the linguistic positions adopted in the following section are consistent with much of the existing literature, they are primarily intended to provide a sample context in which the role of constructibility may be illustrated.

2. Hyponymy and Compatibility

The term *hyponymy* has been used by J. Lyons (1968) to denote the common but somewhat vague notion of " semantic inclusion ". A *hyponymy p* of a word sense *q* is a word sense whose meaning is a " subset " of the meaning of *q*; for example, *scarlet* is a hyponym of *red*, and *tulip* is a hyponym of *flower*. Intuitively, if *p* is a hyponym of *q*, then it makes sense to say that *p* is a kind of *q*. Synonymy of word senses may be defined in terms of hyponymy: *p* is a synonym of *q* if and only if *p* is a hyponym of *q* and *q* is a hyponym of *p*. The hyponymy relation is evidently reflexive and transitive.

The importance of hyponymy has been generally recognized by workers in computational semantics (see, for example, M. R. Quillian, 1969, and R. M. Schwarcz, J. F. Burger, R. F. Simmons, 1970). Quillian's system (M. R. Quillian, 1969) is a generalization of a scheme for representing lexical *definitions* of word senses in terms of *supersets* and *properties*, where the superset relation is simply the inverse of hyponymy. When the superset of some sense is modified by the relevant properties, an expression is obtained which is a definition of *paraphrase* of the given sense. As a highly simplified example, suppose that the superset of *bachelor* is *man*, with the corresponding property *unmarried*, and that the superset of *youth* is also *man*, with the corresponding property *young*. A simple calculation would then mark the expressions *young bachelor* and *unmarried youth* as paraphrases, since both are equivalent to the expression *young unmarried man*. Although I have not mentioned many of the details and problems involved (such as the possible forms properties may take), it seems clear that a representation for hyponymy is necessary for such important semantic processes as transformation between equivalent linguistic expressions.

Turning to the *compatibility* relation, the simplest case involves combinations of adjective senses and noun senses to form expressions which may be either meaningful (*hungry man*) or meaningless (*hungry theory*). The two senses may be said to be compatible if their combination is meaningful. It turns out that little can be done with regard to the representation of this relation unless its definition is extended as follows: two adjective senses are compatible if and only if there are noun senses compatible with both. It is useful here to think of noun senses as " atoms " or " constants " and adjective senses as " predicates " which take noun senses as arguments (this is similar to Fillmore's approach [C.J. FILLMORE, 1969]). The second definition of compatibility then involves the intersections of the domains of predicates. This relation, which is reflexive and symmetric, will be the more important as far as constructibility is concerned.

As for the *use* of compatibility, on the other hand, the first definition is the important one. First, in a linguistic or computational system involving the generation or analysis of sentences, it would be useful in excluding meaningless constructions of word senses, and it can be argued that this capability is a fundamental one for any such system. Secondly, the relation provides the ability to *disambiguate* words in the context of their combination with other words. For example, suppose that *hard* has two senses written as *hard (difficult)* and *hard (not soft)*, and that *ball* has the two senses *ball (round object)* and *ball (social event)*. Suppose further that *hard (difficult)* is compatible with *question* and incompatible with both senses of *ball*, and that *hard (not soft)* is incompatible with *question* and *ball (social event)* but compatible with *ball (round object)*. Then, in the expression *hard question*, *hard* must mean *hard (difficult)*. Similarly, in the expression *hard ball*, *hard* must mean *hard (not soft)* and *ball* must mean *ball (round object)*, since the other three combinations are all incompatible.

3. STRUCTURES FOR HYPONYMY

To illustrate some classes of structures for the hyponymy relation, a hypothetical set of seven word senses *a*, *b*, ..., *g* will be used. Their hyponymy relationships are given in Fig. 1 in the form of a matrix: a sense *x* is a hyponym of a sense *y* if and only if there is a 1 in the position whose row corresponds to *x* and whose column corresponds to *y*.

	a	b	c	d	e	f	g
a	1	0	1	1	1	1	1
b	0	1	1	1	1	0	0
c	0	0	1	1	1	0	0
d	0	0	0	1	1	0	0
e	0	0	0	0	1	0	0
f	0	0	0	1	1	1	1
g	0	0	0	1	1	1	1

Fig. 1. *Hyponymy matrix.*

The graph structure which directly corresponds to this matrix is given in Fig. 2.

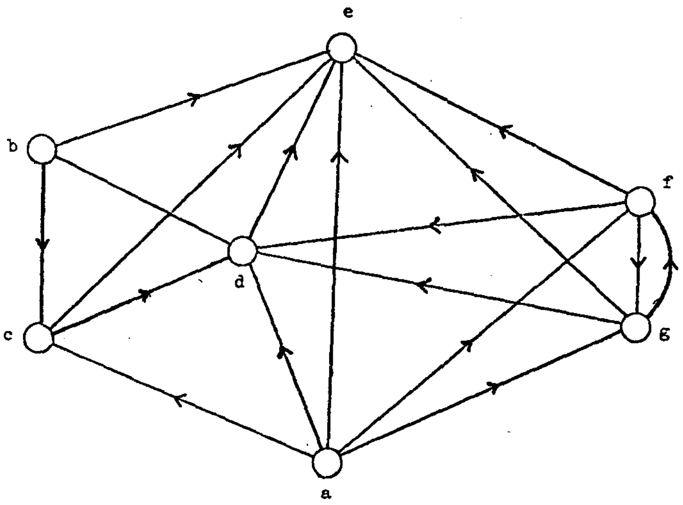

Fig. 2. *F-graph.*

In an *F-graph* such as this, each node corresponds to a sense, and the interpretation of the structure is that x is a hyponym of y if and only if there is an arc from x's node to y's node (strictly speaking, since hyponymy is reflexive, there should be a loop at each node, but these may be omitted without loss of information).

An obvious simplification is otbained by making use of the transitivity of hyponymy in the interpretation of the structure. In a *G-graph* (Fig. 3), then, the nodes are unchanged from the corresponding *F*-graph, but now x is a hyponym of y if and only if there is a *path* from x's node to y's node.

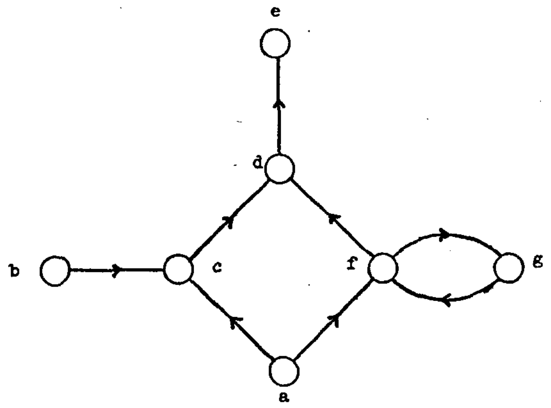

Fig. 3. *G-graph.*

An *F*-graph is the transitive closure of its corresponding *G*-graph, and the *G*-graph has no " redundant " arcs; i.e., arcs from x to y and from y to z imply that there is no arc from x to z. This, together with the fact that synonyms are represented by separate nodes, implies that *G*-graphs are not necessarily unique; for example, the *G*-graph in Fig. 4 is equivalent to that in Fig. 3.

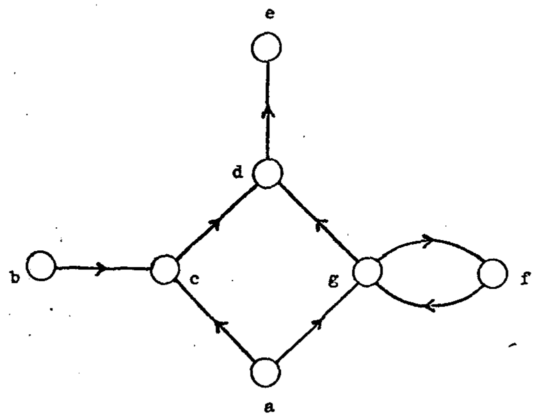

Fig. 4. *An equivalent G-graph.*

This non-uniqueness may be avoided by transforming the *G*-graph into an *H-graph*, in which synonyms are associated with the same node but the interpretation is otherwise unchanged. All *H*-graphs are acyclic, since all the senses on a cycle would have identical hyponymy relationships with all other senses, i.e., they would be synonyms. Thus the nodes form a partially ordered set, and *H*-graphs may be represented by lattice-like Hasse diagrams (D. E. RUTHERFORD, 1965) where each edge is assumed to be directed towards the top of the page, although the arrows are not explicitly present (Fig. 5).

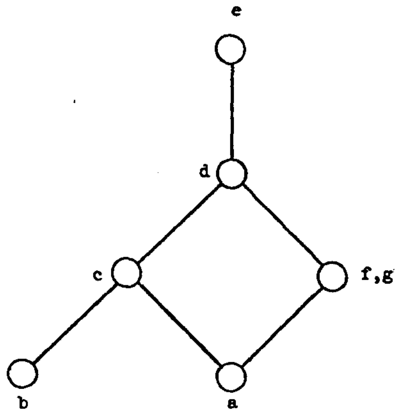

Fig. 5. *H-graph.*

It may easily be seen that *H*-graphs are in one-to-one correspondence with the set of all possible hyponymy matrices (transitive, square, Boolean matrices). They are thus a *fully general* class of structures, and the question arises as to whether the set of linguistically significant structures is actually more restrictive than this. Judging from the exsting literature in pure and applied linguistics, there would seem to be a consensus that hyponymy graphs should in fact be tree-structured. This is apparently the case in Quillian's system (M. R. QUILLIAN, 1969), for example, and some experimental support for the psychological reality of this type of structure is given in A. M. COLLINS, M. R. QUILLIAN (1969). T. G. BEVER and P. S. ROSENBAUM (1970) have also argued for the adequacy of tree structures in this context. In terms of *H*-graphs, the necessary restriction is obtained by requiring that any word sense may be an *immediate* hyponym of at most one other sense; i.e., it may not be a hyponym of two " disjoint " senses.

The class of structures so obtained may be termed *H-forests*, since they are a proper subset of the *H*-graphs.

Although it has been generally adopted in existing models, the restriction to *H*-forests is not without its difficulties. The *H*-graph in Fig. 6 represents an intuitively correct set of hyponymy relationships which is inconsistent with the *H*-forests.

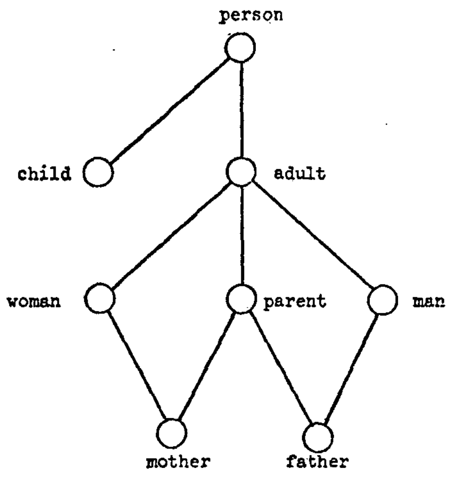

Fig. 6. *Apparent inadequacy of H-forests.*

An interesting but open question is whether such examples can be considered to be rare in natural language, occurring only in very special cases such as kinship terminology. A related question concerns the validity of the assumption that a given sense is *necessarily* an immediate hyponym of at most one other sense, so that hyponymy, defined on this basis, must be tree-structured. It is clear that with the present state of our knowledge a linguistic case can be made for both *H*-graphs and *H*-forests as the appropriate class of structures for the hyponymy relation.

Turning now to the constructibility of the various possible classes of hyponymy structures, it is not necessary to consider *F*-graphs and *G*-graphs, since they may be dismissed in favor of *H*-graphs purely on the grounds of their mathematical properties and economy of representation. Both *H*-graphs and *H*-forests are constructible insofar as one can devise semi-automatic discovery procedures which will, in principle, construct them from basic data obtained from an informant.

This " basic data " would consist of judgments of the relationships between particular pairs of senses: either the first is a hyponym of the second, the second is a hyponym of the first, they are synonyms, or they are related in none of these ways.

The *H*-graph or *H*-forest would be constructed by a process of monotonic refinement; i.e., after the incorporation of each additional word sense, the new structure would be a recognizable generalization of the previous structure. More specifically, suppose that H_i and H_{i+1} are the current structures after i and $i+1$ senses, respectively, x and y are any two senses previously processed, and the *distance* between two senses on the same path is the number of arcs separating them. Then the following statements would hold:

a) If x and y are at the same node in H_i, then they are at the same node in H_{i+1};

b) x and y are not at the same node in H_i iff they are not at the same node in H_{i+1};

c) x is above y in H_i iff x is above y in H_{i+1};

d) if x is a distance d above y in H_i, then x is a distance d or $d+1$ above y in H_{i+1}.

e) if x is a distance d above y in H_{i+1}, then x is a distance d or d-1 above y in H_i;

f) x and y are not on the same path in H_i iff they are not on the same path in H_{i+1}.

The remaining requirements for constructibility are those of input efficiency and consistency maintenance. An obvious point of reference for efficiency is the maximum number of inputs which would be required irrespective of the algorithm and the set of word senses used. This is approximately given by $\frac{1}{2} N^2$, where N is the number of senses, and would be reached in cases where inputs must be obtained for all pairs of senses. The transitivity of hyponymy would allow *H*-graphs to be constructed with better efficiency than this; for example, if a sense z were found to be a hyponym of y, and y were known to be a hyponym of x, then it could be inferred or *predicted* that z is a hyponym of x with no additional input. *H*-forests could be constructed with better efficiency than *H*-graphs since they are a much more predictive class of structures; for example, if z were found to be " unrelated " at the root of a tree, then it would be predicted to be " unrelated " everywhere in the tree.

Because the *H*-graphs are a fully general class of structures, inconsistency of a set of inputs with respect to them could be due only to a violation of transitivity, and the procedure could check for such occurences automatically. In the case of *H*-forests, there is more scope for inconsistency, since any valid *H*-graph which is not also an *H*-forest represents a set of information which is inconsistent with *H*-forests. The procedure would have to obtain redundant inputs in order to maintain a consistency check in this case. The details of testing for and correcting inconsistencies would depend upon the properties of the particular discovery procedure used.

To summarize this section, I have attempted to place in perspective the plausible classes of structures for hyponymy. Both *H*-graphs and *H*-forests satisfy the basic requirement for constructibility. The construction of the latter, however, would be more efficient, so that it would be disadvantageous to use a discovery procedure for *H*-graphs in general if *H*-forests alone are linguistically adequate.

4. Structures for Compatibility

The variety of structure classes for the compatibility relation turns out to be richer than in the case of hyponymy. To illustrate some of these representations, suppose that the compatibility relationships among the seven word senses *a*, *b*, ..., *g* are as shown in the matrix of Fig. 7.

	a	b	c	d	e	f	g
a	1				(symmetric)		
b	1	1					
c	1	1	1				
d	1	0	0	1			
e	1	0	0	1	1		
f	1	0	0	1	1	1	
g	1	0	0	1	0	0	1

Fig. 7. *Compatibility matrix.*

Since the relation is symmetric, it is quite natural to represent it as an undirected graph (an *A-graph*) where the nodes correspond to senses and the edges indicate compatibility. The *A*-graph corresponding to Fig. 7 is given in Fig. 8.

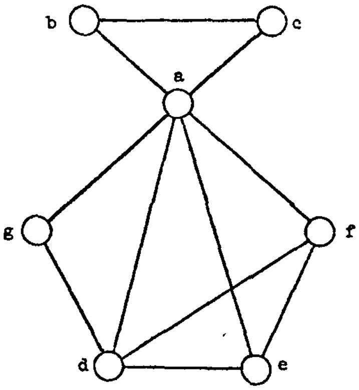

Fig. 8. *A-graph.*

For large sets of senses, *A*-graphs would clearly be unwieldy and uneconomic, and a more compact representation would be highly desirable. One particularly interesting transformation of the matrix or *A*-graph is obtained by grouping senses with the same compatibility properties with respect to all the other senses and placing them at the same node. The interpretation of such a *C-graph* is that two senses are compatible if and only if they are associated with the same or adjacent nodes. The *C*-graph for the present example is given in Fig. 9. It can be easily shown that every *C*-graph is unique and that this class of structures is fully general.

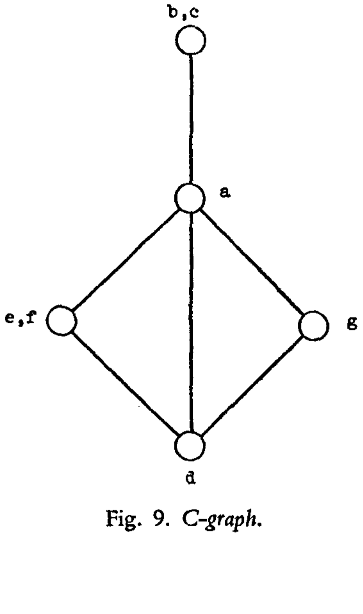

Fig. 9. *C-graph.*

Turning now to more restrictive structures, a more compact representation can sometimes be obtained by transforming the undirected *C*-graph into a directed *D-graph* with the same set of nodes but with the interpretation that two senses are compatible if and only if either they are at the same node or there exists a path between their nodes. In analogy to the situation for *H*-graphs, the nodes of a *D*-graph are partially ordered. Fig. 10 shows a *D*-graph for the example in the form of a Hasse diagram. The *D*-graphs are not a fully general class of structures; i.e., there are *C*-graphs with no corresponding *D*-graphs. Neither are they necessarily unique: it can be shown that some *C*-graphs have more than one equivalent *D*-graph.

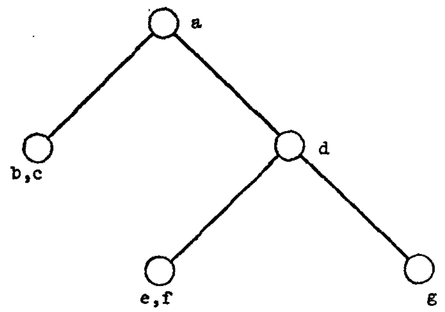

Fig. 10. *D-graph.*

The linguistic significance of these structures lies in their strong similarity to systems of semantic *markers* or *features* (corresponding to nodes) related by *redundancy rules* (corresponding to edges) (J. J. KATZ, J. A. FODOR, 1963). In this case, a feature is *defined* as the set of senses which, in other formulations, would be said to *have* the feature. The relationships among features, and hence the compatibility properties of word senses, have often been represented by structures similar to *D*-graphs; moreover, a survey of the relevant theoretical and computational literature (e.g., R. C. SCHANK, 1969; F. SOMMERS, 1963) would reveal that tree-like structures have nearly always been considered to be adequate for this purpose. The graph of Fig. 10 is in fact such a *D-forest.* A given set of compatibility relationships can be represented by at most one *D*-forest.

The constructibility criterion also bears strongly on the relative merits of these structure classes. Considering the *C*-graphs first, it may easily be shown that this class satisfies all the requirements of constructibility except the crucial one of efficiency. Assuming the availability of basic data in the form of yes-or-no judgments of the compatibility of unordered pairs of word senses, a *C*-graph could be built by a monotonic refinement process: at each stage the current sense would either be added to an existing node or form a new node, possibly in conjunction with the " splitting " of other nodes. Efficiency, however, is the worst possible because the class of structures is fully general, and the judgements obtained for the compatibility of the current sense with some subset of the previous senses have no predictive value for its compatibility with any of the other senses in the graph.

The lack of efficiency in constructing *C*-graphs can be regarded as a corollary of the fact that inconsistencies are impossible. They are possible in the case of *D*-graphs, however, and thus the assumption that they would not arise would make some judgments unnecessary after others became known. But *D*-graphs fail to be constructible because of their non-uniqueness. The *D*-graphs shown in Fig. 11 and Fig. 12 are equivalent,

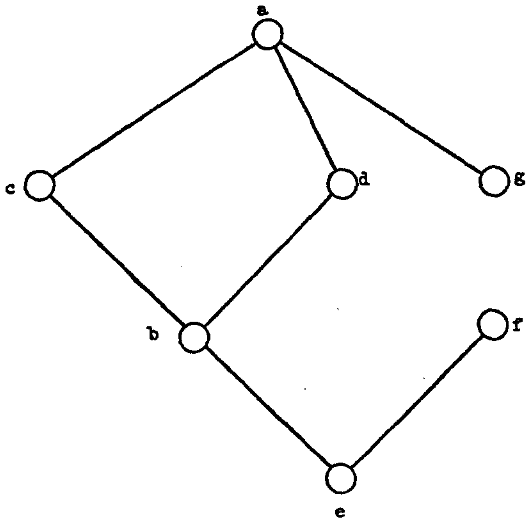

Fig. 11. *A non-unique D-graph.*

and that shown in Fig. 13 represents the same information except that it contains an additional word sense.

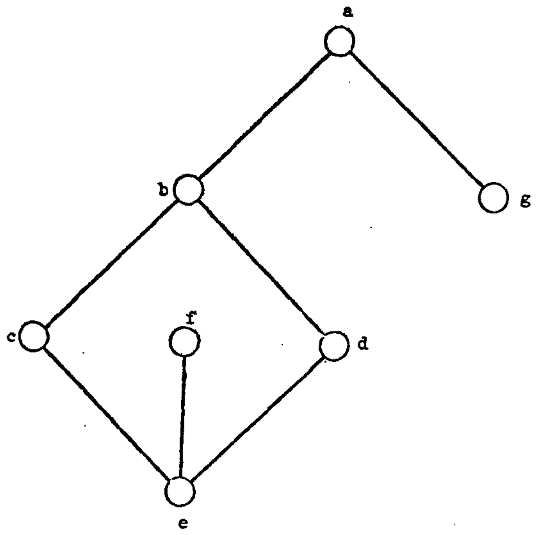

Fig. 12. *An equivalent D-graph.*

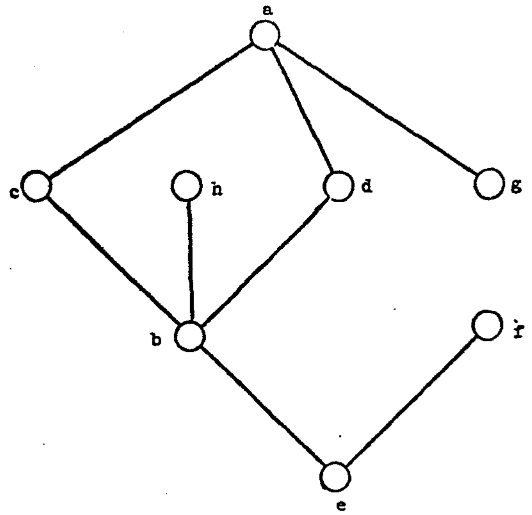

Fig. 13. *A refinement of Fig. 11.*

The latter graph is a simple modification of Fig. 11 but not of Fig. 12, and there is no graph equivalent to Fig. 13 which is easily obtained from Fig. 12. Since a given discovery procedure could construct either of the first two graphs depending upon the order in which the senses were incorporated, the process would in general involve complete reconstruction rather than monotonic refinement.

The *D*-forests, finally, are constructible in all respects. Monotonic refinement is characterized in this case by the following statements, where x and y again are any two senses previously processed and D_i and D_{i+1} are the current forests after i and $i+1$ senses, respectively:

a) if x and y are at the same node in D_i, then they are at the same or adjacent nodes in D_{i+1};

b) if x is a distance d above y in D_i, then x is a distance d or $d+1$ above y in D_{i+1};

c) if x is a distance d above y in D_{i+1}, then x is a distance d or $d-1$ above y in D_i;

d) x and y are not on the same path in D_i iff they are not on the same path in D_{i+1}.

A *D*-forest is a highly predictive structure; i.e., a judgment involving one of its word senses will in general predict the outcome of many other judgments. A detailed estimation of the efficiency of construction would involve the order of the senses, the configuration of the final structure, and the node-scanning strategy used in the discovery algorithm. It is sufficient here to recall, however, that nodes correspond to linguistic features, and hence one would expect that if the number of senses processed were very large, then the number of nodes would be much smaller. Thus all the nodes would appear relatively early in the construction process, and this leads to the conclusion that the total number of judgments would be a nearly linear function of the number of senses. In order to detect inconsistencies, however, some of this efficiency would have to be sacrificed and redundant judgments taken.

To sum up this section, the series of structure classes considered for representing compatibility has led to one class, the *D*-forests, which is constructible. These structures also have several important properties in common with other models in theoretical and computational linguistics.

5. Conclusion

This paper has presented two examples of the use of constructibility as a design criterion for lexical or semantic structures of the type used in natural language processing. This criterion is essential if we ever hope to implement computational systems with broad linguistic competence. It is very encouraging that in the two cases considered the approach has led to structures basically equivalent to those which have actually been used. Although these previous models have generally lacked the property of constructibility, then, it is suggested that this is a fault of design which can be overcome.

REFERENCES

T. G. BEVER, P. S. ROSENBAUM, *Some Lexical Structures and their Empirical Validity*, in R. A. JACOBS, P. S. ROSENBAUM (eds.), *Readings in English Transformational Grammar*, Weltham (Mass), 1970.

A. M. COLLINS, M. R. QUILLIAN, *Retrieval Time from Semantic Memory*, in « J. Verb. Learn. Verb. Behav. », VIII (1969), pp. 240-247.

C. J. FILLMORE, *Types of Lexical Information*, in F. KIEFER (ed.), *Studies in Syntax and Semantics*, Dordrecht (Holland), 1969.

J. J. KATZ, J. A. FODOR, *The Structure of a Semantic Theory*, in « Language », XXXIX (1963), pp. 170-210.

J. LYONS, *Introduction to Theoretical Linguistics*, London, 1968.

M. R. QUILLIAN, *The Teachable Language Comprehender*, in « Communications of the ACM », XII (1969) 8, pp. 459-476.

D. E. RUTHERFORD, *Introduction to Lattice Theory*, Edinburgh, 1965.

R. C. SCHANK, *A Conceptual Parser for Natural Language*, in *Proc. IJCAI*, Washington, 1969, pp. 569-578.

R. M. SCHWARCZ, J. F. BURGER, R. F. SIMMONS, *A Deductive Question-answerer for Natural Language Inference*, in « Communications of the ACM », XIII (1970) 3, pp. 167-183.

S. Y. SEDELOW, *Language Analysis in the Humanities*, in « Communications of the ACM », XV (1972) 7, pp. 644-647.

F. SOMMERS, *Types and Ontology*, in « Phil. Rev. », LXXII (1963), pp. 327-363.