In: American Journal of Computational Linguistics, Microfiche 77, 1978. S. 53-67.

GLANCING, REFERRING AND EXPLAINING IN THE DIALOGUE SYSTEM HAM-RPM

W. WAHLSTER, A. JAMESON, W. HOEPPNER

Project: 'Simulation of Language Understanding'
Germanisches Seminar der Universität Hamburg
von-Melle-Park 6, D-2000 Hamburg 13, West Germany

SUMMARY

This paper focusses on three components of the dialogue system HAM-RPM, which converses in natural language about visible scenes. First, it is demonstrated how the system's communicative competence is enhanced by its imitation of human visual-search processes. The approach taken to noun-phrase resolution is then described, and an algorithm for the generation of noun phrases is illustrated with a series of examples. Finally, the system's ability to explain its own reasoning is discussed, with emphasis on the novel aspects of its implementation.

1. THE TREATMENT OF VISUAL DATA

The natural language dialogue system HAM-RPM[1] converses with a human partner about scenes which either one or both are looking at directly (or have a photograph of). At present the system, which is implemented in FUZZY (LeFaivre 1977), is being tested on two domains: the interior of a living room and a traffic scene.

Since it is assumed that both partners begin the dialogue with relatively little specific knowledge about the scene, most of the specific information used by the system during the conversation must be obtained by a process more or less analogous to looking at the scene. We have found it worth while to make the analogy quite close, requiring the system to retrieve its visual data by doing something like casting a series of glances centered on various points in the scene.

Fig. 1 is a schematic drawing of a section of our traffic scene, showing a tree with a parking lot in front of it. How easy is it to recognize the various objects in Fig. 1 when glancing at point A? CAR9 and CAR8 will be about equally easy to recognize as cars. TREE4 will probably be recognized more easily, since it is equally close to point A, and very large, and since there are no similar types of objects. On the other hand, CAR3 will be less easily recognizable, since it is farther away. MAN4 is probably too far away to be recognizable as a man at all (he is recognizable only from the points nearest him, as is shown by the four arrows pointing away from him).

Just this information is stored in HAM-RPM in a separate associative network corresponding to point A. In all, there are about a hundred such small networks (represented by the small dots in Fig. 1), corresponding to possible glances at the scene. The statements about the nature of the various objects which are recognizable from the point in question are ordered, in a way characteristic of the FUZZY programming language, in terms of their recognizability, so that they will automatically be retrieved in that order.[2]

[1] The system's overall structure is described in (v. Hahn et al. 1978) as are the goals and methodological principles which guide the research within the project.

[2] These networks are implemented as CONTEXTs in the sense introduced by the language CONNIVER.

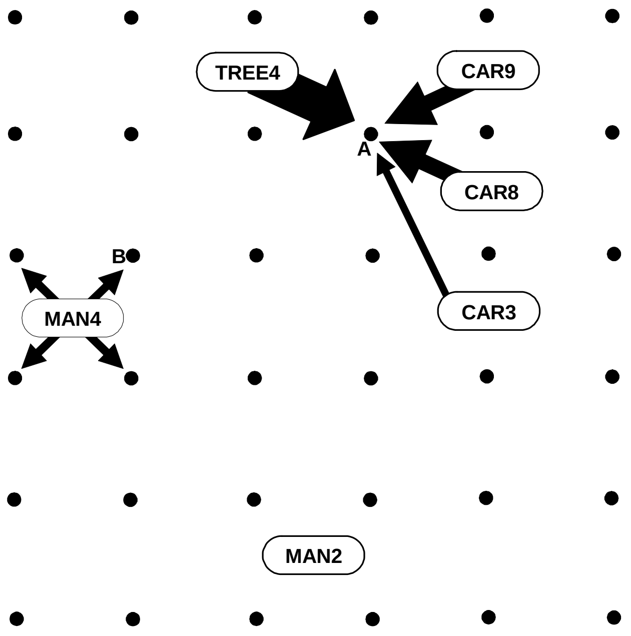


Fig. 1: "The man in front of the tree"

A simple example will show how the data stored in this way can be used by the system. When interpreting the definite description *the man in front of the tree,* assuming that TREE4 is the one meant, the system enters several CONTEXTs in front of TREE4, within each retrieving the internal names of the men recognizable from that point. It doesn't find MAN4 until it has entered the CONTEXT corresponding to point B. It then enters a couple more, and, finding no further men, assumes that it has found the referent of the definite description. Information not only about the respective types of the various objects, but also about their other attributes is stored in a similar way.

Why is it worth all this trouble to make the system sensitive to the recognizability of the various facts about a scene from the various points within it? After all, the facts themselves could be stored very straight forwardly.

Our principal justification is that, for a dialogue system which is supposed to communicate effectively with a human partner, the bare facts about the scene are less important than the way the partner himself would be likely to perceive them. If only the facts themselves are known, information may be lacking which is essential for the production of a communicatively adequate response. For example, the definite description whose interpretation was just sketched was, strictly speaking, ambiguous, as there is a second man in front of the tree whom the system would have considered to be the referent of the description if MAN2 hadn't been there. Yet the system didn't even notice this ambiguity, since it stopped shortly after finding the first man.

To be sure, the resolution of such ambiguities could also be achieved by giving the system general information on the recognizability of objects for human beings and letting it choose on that basis which of the potential referents of the description was the one which the partner was most likely to have intended to refer to. Instead of doing this, we have made the system itself a model of its partner, so that instead of referring to a model, it only has to 'be itself' or 'act naturally', in order to communicate effectively.[3]

[3] Two of the reports (v. Hahn 1978a, 1978b) which have been issued by the HAM-RPM group deal with the question of the nature of the relation between the dialogue-partner model and the human partner in some detail.

In addition to the interpretation of ambiguous utterances, there are other situations in which this approach can be applied elegantly (Fig. 2).

	SITUATION	INFORMATION
1)	Interpretation of an ambiguous definite description	Which object the speaker is probably referring to
2)	Generation of a definite description	Which reference points will be easy for the listener to find
3)	Description of a part of a scene	Which objects the speaker might be interested in hearing about

Fig. 2

When describing the location of an object with reference to other objects, the system will usually find a number of potential reference points; in general, it should mention those which are visually easiest for the listener to find. This is likely to happen if it itself finds these reference points particularly easily. When answering a vague question, such as a request to describe what is on the other side of the street, the system will have to select among the many visible facts those which the listener might be interested in hearing about. In many cases, these will be the visually most salient facts.

2. NOUN-PHRASE RESOLUTION

Two of the components of HAM-RPM which make use of the visual data are those responsible for noun-phrase resolution, that is, the determination of the potential referents of a noun phrase, and noun-phrase generation, that is, the construction of noun phrases to identify objects uniquely.

The procedures which resolve noun phrases work on the shallow structure of the input sentence. This is what is obtained after multiple-word phrases and idioms have been replaced with canonical expressions, the words have been looked up in the lexicon, and a simple morphological analysis has been performed.

A definite noun phrase is recognized within the shallow structure as a structure consisting of a definite article, possibly one or more attributes, a noun, and possibly a relative clause (Ritchie 1977). In a way reminiscent of Winograd's SHRDLU (Winograd 1972), processes involving semantics and pragmatics are activated in HAM-RPM as soon as possible during the analysis of the input sentence.

The noun-phrase interpreter tries to find a unique referent for each definite noun-phrase by using the knowledge stored in the conceptual and referential semantic networks and performing visual search algorithms. For example, the definite description *The picture hanging to the left of the red chair,* referring to Fig. 3, is replaced with the internal object-name PICTURE1 in the shallow structure of the sentence. This strategy can save a good deal of unnecessary processing: if no object is found which satisfies the description, there is no further parsing, but rather feed-back to the conversational partner. In the case where more than one potential referent is found, the one mentioned most recently is assumed to be the referent. If none of them has been mentioned recently, the system asks the partner for further details, assuming, as it were, that he does have some particular object in mind. These details take the form of a noun phrase, which may be either complete or elliptical. Further attributes of the intended object may be specified, it may be characterized in terms of its spatial relations to other objects, or the noun originally used in the description may be replaced with a more specific one.

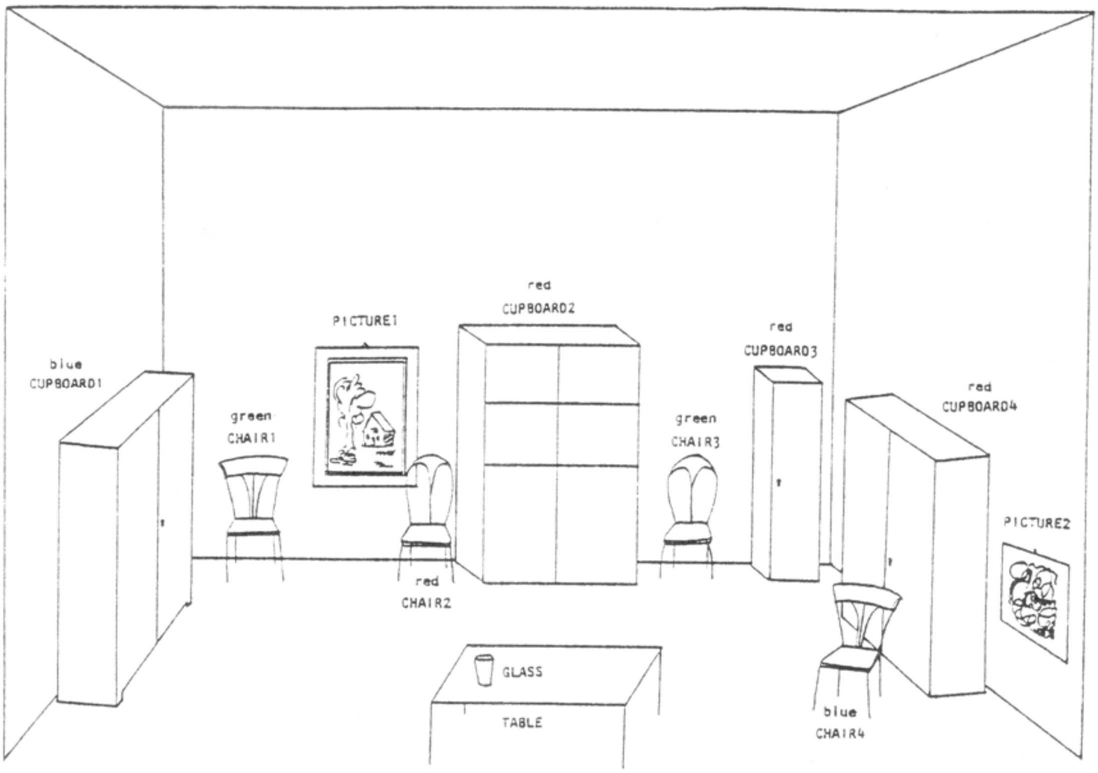


Fig. 3

Not all noun phrases, of course, can be replaced immediately with a specific referent. One such case is exemplified by the description *the chair in front of the red cupboard*. Applied to the scene in Fig. 3, the noun phrase the red cupboard cannot be replaced, because there is more than one red cupboard, but it cannot be ignored, either, because there is more than one chair. The entire noun phrase can only be interpreted when it is recognized that there is only one pair of objects which stand in this relation to one another.

Another case where a definite noun phrase can't simply be replaced directly by its referent is the generic description with definite article, as in the sentence *The chair is something to sit on*. Lately we have been thinking about what formal features of a sentence might be helpful in recognizing such descriptions (see Grosz 1976).

Two clues which tend to favour a generic interpretation are the absence of any referential attribute and the presence of an adverb such as *usually* or *normally*. On the other hand, a generic interpretation becomes somewhat less plausible if the noun phrase is the object of a local preposition, as in *on the chair*; if the sentence is in the past tense; or if the verb can be generally classified as one involving visual perception or spatial relations. We assume that, no matter how many weak inference rules of this sort are incorporated into the system, there will still be some ambiguities which can only be resolved by other means, including interaction with the speaker.

A general goal in this connection is a sort of compatibility between noun-phrase resolution and noun-phrase generation, in the sense that the system should be able to understand any kind of noun-phrase that it can generate, and vice versa.

3. NOUN-PHRASE GENERATION

The method we have developed for the inverse process, noun-phrase generation, is distinguished from earlier approaches mainly in three respects.

The first is its use of what might be called a 'worst-case-first' strategy. The second is the way it takes into consideration the ease with which the listener will be able to interpret the description it generates, when more than one uniquely identifying description is possible (Herrmann & Laucht

1976). The third is its use of complex spatial relations to deal with the 'worst cases', that is, those in which several objects are indistinguishable on the basis of their properties alone.

Let's examine a few examples of the behavior of the algorithm. First, two trivial cases.

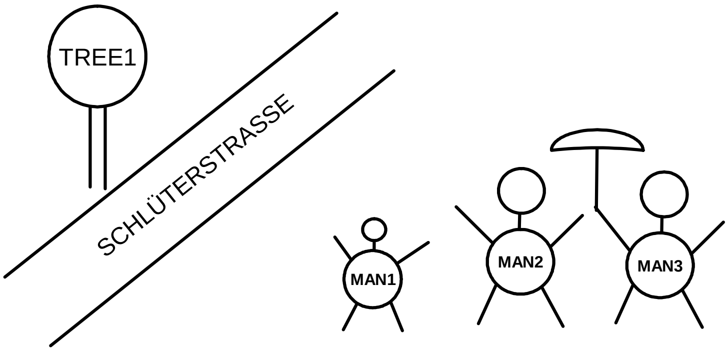


Fig. 4

The street in Fig. 4 has a proper name, and is thus referred to simply as *Schlüterstrasse*. The tree is the only one in the discourse world, and hence is identified as the tree. The number of interesting possible strategies becomes greater when the object to be described is one of several belonging to the same conceptual class. Consider for example MAN1 in Fig. 4. The system looks among its properties for one which distinguishes it from MAN2 and MAN3, and describes it *as the small man*. A similar process underlies the generation of the noun phrase *the big man with the umbrella* to refer to MAN3.

Note that the system uses redundant labels. This is a consequence of the sequential nature of its noun-phrase generation: First, the property 'big' is found. When the system notices that there is another big man in the scene, it looks for a further distinguishing property and finds the umbrella. This property would in fact be adequate in itself, but the system doesn't attempt to find a minimal characterizing set of attributes. This sort of redundancy, which is often found in human beings, saves time both in the generation and in the interpretation of definite descriptions.

HAM-RPM frequently uses negative characterizations of various kinds, as, for example, when MAN2 is described as *the big man without an umbrella*. Now let's turn to some more complex problems of noun-phrase generation. So that the pictures don't get too cluttered, we will use examples from a simple domain of geometrical figures (Fig. 5). Consider CIRCLE1 in Fig. 5. Note that there are two green circles in the scene. The presence of several objects which are indistinguishable on the basis of their attributes alone is the worst case which can occur. The reason why we have spoken of a 'worst-case-first' strategy is that the system checks for this case early, rather than trying immediately to construct a simpler characterization such as those in the last few examples given.

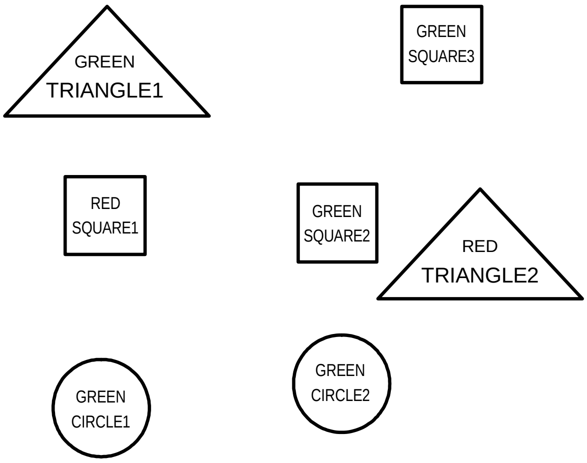


Fig. 5

Informal observation shows that human beings also often notice the presence of identical objects in a scene immediately. The only way to distinguish these two circles is by reference to spatial relations, for example, *the green circle in front of the red square*.

We may note in passing two ways in which the form of a description may be constrained by the form of the question which is being answered. First, properties which have been presupposed in the question should not be mentioned in a description. Consider the question *Which square is red?*. The answer *The red square* is clearly unacceptable, so instead the system answers *The square in front of the green triangle* (= SQUARE1 in Fig. 5). A second constraint of this sort is that the system should not produce circular descriptions. For example, when answering the question *Where is the red triangle?*, the system should not answer *To the left of the green square which is to the right of the red triangle*, although each half of this description is perfectly natural when considered in isolation.

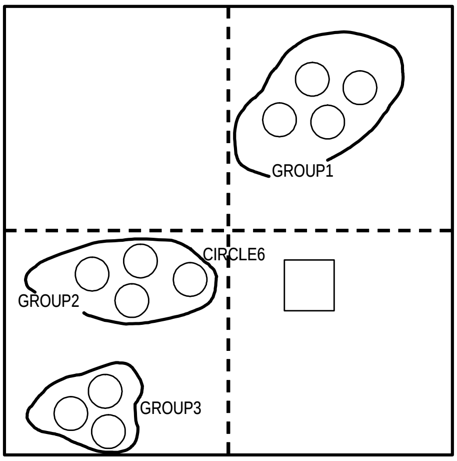


Fig. 6

It sometimes happens that objects chosen as spatial reference points in a description in turn have to be identified with the help of other reference points. For example, CIRCLE2 in Fig. 5 is described as *the green circle in front of the green square which is to the left of the red triangle*. As this example shows, the products of such recursive applications of the generation algorithm can soon become difficult to understand. We have made the maximum allowable depth of recursion a parameter which can be set to various values for experimental purposes.

Perhaps the most difficult problem in noun-phrase generation is the unique identification of an object when there are objects with exactly the same properties in its immediate neighbourhood (Fig. 6). This is a task which often causes difficulties even for a human speaker. To set the stage, suppose that CIRCLE6 in Fig. 6 is to be identified. The system first describes its position relative to the other circles in its group: *the right-hand circle*. Then it identifies the group of which CIRCLE6 is a member within the scene as a whole, distinguishing it first from GROUP2: *in front and to the left* and then from GROUP3: *which is to the left of the square*. Thus the complete description is *The right-hand circle in front and to the left which is to the left of the square.*

To put the point more generally, complex scenes sometimes have a hierarchical structure in which groups of similar objects serve as units which have to be identified in much the same way that objects themselves are. The remarks we have made about circular descriptions and recursion depth apply on the level of groups as well.

Concluding this sketch of HAM-RPM's noun-phrase interpreter and generator, we would like to stress that all these algorithms are domain-independent.

4. EXPLANATION

Although all of the examples discussed up to now have involved some sort of description of visible aspects of a scene, HAM-RPM frequently makes use of general knowledge and inference rules to draw conclusions.

For example, the system might be asked *Is the parking zone tarred?*, where the parking zone in question, though part of the scene, is hidden from view. It would then try to answer the question using approximate inferences based on fuzzy knowledge (Wahlster 1978), concluding that the parking zone might very well be tarred, because a parking zone is in a sense a part of a street, and streets, like thoroughfares in general, are usually tarred. Inferences which stand on such shaky ground as this one are of limited use to the conversational partner unless the system can describe the reasoning which underlies them.

Furthermore, not just any description will be satisfactory: the system ought to act in accordance with the following three maxims, as formulated by (Grice 1975):

1. Make your contribution as informative as is required.
2. Don't make your contribution more informative than is required.
3. Be relevant.

Thus, describing an inference chain in every detail will not in general be communicatively adequate, if some of the inferences are essentially definitional, and hence conceptually trivial. Only when the dialogue partner has repeatedly requested details about inferences will it be sensible to mention all of them.

Now let's look at the way we have tried to achieve these goals in HAM-RPM, using the example just given. Three processes are essential. First, while the reasoning is being performed, a sort of trace of the inference process is stored in a separate data base called INFERENCE-MEMORY. Second, after an explanation of the conclusion has been requested, this part of memory is traversed to find those of the assumptions used which are on a communicatively appropriate level of detail. Finally, these assumptions are expressed in natural language.

An essential role in the first two of these phases is played by the meta-knowledge associated with each inference rule which is available to the system. As you can see from the two inference-rule definitions in Fig. 7, one such piece of meta-knowledge concerns the degree of uncertainty associated with the rule. The most interesting piece of meta-knowledge in this situation is the specification of a particular FUZZY procedure demon. These demons enforce during the application of an inference rule global control regimes specified by the programmer (LeFaivre 1977). In particular, one of the things done by TRACE-PROCEDURE-DEMON7 is the storage of the reasoning steps in INFERENCE-MEMORY.

META-KNOWLEDGE:

- Apply the control knowledge coded in TRACE-PROCEDURE-DEMON7
- Don't use instantiations of premises with a degree of belief less than 0.3
- The degree of uncertainty of this rule is 0.5

```
RULE:        If you want to show   (X   IS    Y)
               show that           (X   ISA   Z)
                           and     (Z   IS    Y)
```

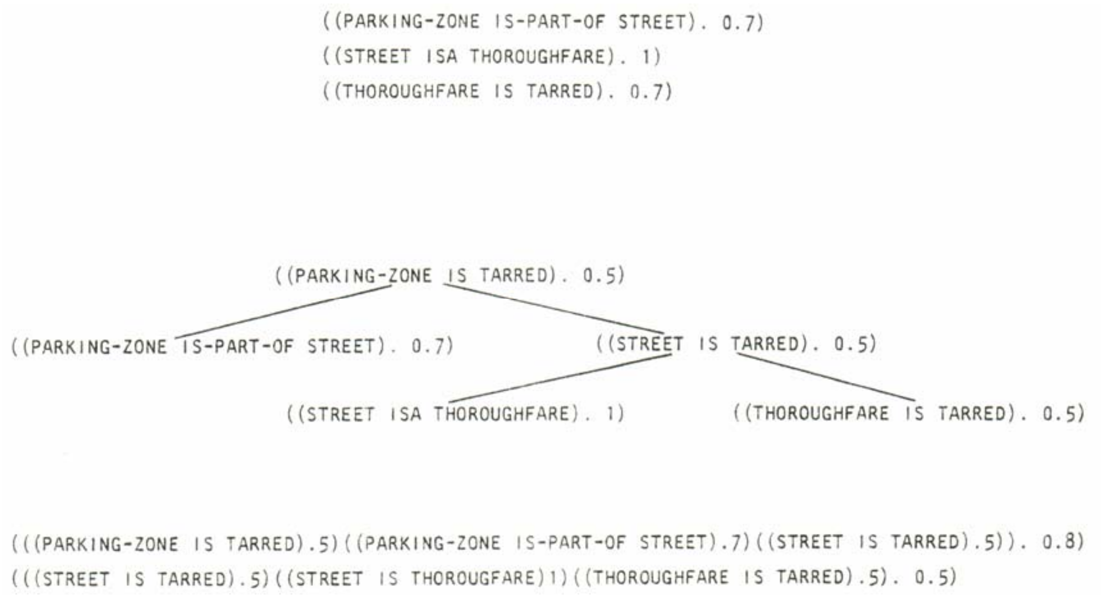
META-KNOWLEDGE:

- Apply the control knowledge coded in TRACE-PROCEDURE-DEMON7
- Don't use instantiations of premises with a degree of belief less than 0.4
- The degree of uncertainty of this rule is 0.8

```
RULE:        If you want to show   (X   IS    Y)
               show that           (X   IS-PART-OF    Z)
                           and     (Z   IS    Y)
```

Fig. 7

Suppose now that the assumptions at the top of Fig. 8 are represented in semantic networks. Applying the two rules in Fig. 7 to them, the system builds up the goal tree[4] in Fig. 8. The internal trace which is built up by the procedure demon is shown at the bottom of Fig. 8. Note that the entries in the inference memory are ordered in terms of the degree of uncertainty of the executed inference procedures. This means that the most uncertain entries will be mentioned first in the explanation, and the most trivial ones probably not at all. This reflects our hypothesis that degree of uncertainty is the most important factor determining the relevance of a step in an inference chain, as far as justification of the conlusion is concerned.



```
((PARKING-ZONE IS-PART-OF STREET). 0.7)
((STREET ISA THOROUGHFARE). 1)
((THOROUGHFARE IS TARRED). 0.7)
```

((PARKING-ZONE IS TARRED). 0.5)
((PARKING-ZONE IS-PART-OF STREET). 0.7)
((STREET IS TARRED). 0.5)
((STREET ISA THOROUGHFARE). 1)
((THOROUGHFARE IS TARRED). 0.5)

```
(((PARKING-ZONE IS TARRED).5)((PARKING-ZONE IS-PART-OF STREET).7)((STREET IS TARRED).5)). 0.8)
(((STREET IS TARRED).5)((STREET IS THOROUGFARE)1)((THOROUGHFARE IS TARRED).5). 0.5)
```

Fig. 8

Our approach to explanation is distinguished from the previous efforts of Winograd in his SHRDLU system (Winograd 1972) and of the MYCIN group (Scott et al. 1977). In SHRDLU each theorem calls the functions MEMORY and MEMOREND explicitly, which manipulate the inference memory. We have improved over this by integrating the management of the inference memory into a higher process, which controls all reasoning processes. The structure of the inference rules themselves is therefore not obscured by the presence of trace commands. Our approach generalizes the corresponding features of MYCIN, in which the conceptual complexity of a rule is a logarithmic function of its certainty factor and the goal tree is traversed in steps whose size is specified by a numerical argument of the WHY command (Davis et al. 1977).

This approach is also related to recent research by Davis in his TEIRESIAS system (Davis 1977) and Sussman in his AMORD (De Kleer et al. 1977) and EL (Stallman & Sussman 1977) projects, in which general problems of an explicit control of reasoning are explored, in that it is based on an explicit representation of control and meta-knowledge, which typically is 'hidden away' in the interpreter and therefore is inaccessible to the inference system.

The explanation facility of HAM-RPM is far from being complete. Ultimately, the system must understand exactly what the dialogue partner failed to comprehend.

ACKNOWLEDGEMENT

This research is currently being supported by the Deutsche Forschungsgemeinschaft.

[4] The conflict-resolution strategy which is used is one which favours more specific ones.

REFERENCES

Davis, R. (1977): Generalized Procedure Calling and Content-Directed Invocation. *Proceedings of the Symposium on Artificial Intelligence and Programming Languages. SIGART Newsletter 64, 45-54*

Davis, R., Buchanan, B., Shortliffe, E. (1977): Production Rules as a Representation for a Knowledge-Based Consultation Program. *Artificial Intelligence 8, 15-45*

De Kleer, J., Doyle, J., Steele, G.L., Sussman, G.J. (1977): AMORD - Explicit Control of Reasoning. *Proceedings of the Symposium on Artificial Intelligence and Programming Languages. SIGART Newsletter 64, 116-125*

Grice, H.P. (1975): Logic and Conversation. *Cole, P., Morgan, J.L. (eds.): Syntax and Semantics. Vol. 3: Speech Acts. N.Y.: Academic,41-58*

Grosz, B. (1976): Resolving Definite Noun Phrases. *Walker, D.E. (ed.): Speech Understanding Research. Stanford Research Institute, Chapter 9*

v. Hahn, W. (1978a): Überlegungen zum kommunikativen Status und der Testbarkeit von natürlichsprachlichen Artificial-Intelligence-Systemen (Some Thoughts on the Communicative Status and the Testability of Natural Language AI-Systems). HAM-RPM Report No. 4, April 1978 (also to appear in: Sprache und Datenverarbeitung)

v. Hahn, W. (1978b): Probleme der Simulationstheorie und Fragepragmatik bei der Simulation natürlichsprachlicher Dialoge (Some Problems of Simulation-Theory and the Pragmatics of Questions in Connection with the Simulation of Natural Language Dialogues). HAM-RPM Report No. 6, May 1978 (also to appear in: Ueckert, H., Rhenius, D. (eds.): Komplexe menschliche Informationsverarbeitung. Beiträge zur Tagung 'Kognitive Psychologie' in Hamburg. Bern: Huber)

v. Hahn, W., Hoeppner, W., Jameson, A., Wahlster, W. (1978): HAM-RPM: Natural Dialogues with an Artificial Partner. *Proceedings of the AISB/GI Conference on Artificial Intelligence, Hamburg, 122-131*

Herrmann, T. & Laucht, M. (1976): On Multiple Verbal Codability of Objects. *Psychological Research, 38, 355-368*

LeFaivre, R.A. (1977): FUZZY Reference Manual. Rutgers University, Computer Science Dept., March 1977

Ritchie, G.D. (1977): Computer Modelling of English Grammar. University of Edinburgh, Computer Science Dept., Report CST-1-77

Scott, C.A., Clancey, A., Davis, R., Shortliffe, E.K. (1977): Explanation Capabilities of Production-Based Consultation Systems. *American Journal of Computational Linguistics, Microfiche 62*

Stallman, R.M. & Sussman, G.J. (1977): Forward Reasoning and Dependency-Directed Backtracking in a System for Computer-Aided Circuit Analysis. *Artificial Intelligence 9, 135-196*

Wahlster, W. (1978): Die Simulation vager Inferenzen auf unscharfem Wissen: Eine Anwendung der mehrwertigen Programmiersprache FUZZY (The Simulation of Approximate Reasoning Based on Fuzzy Knowledge: An Application of the Many-Valued Programming Language FUZZY). HAM-RPM Report No. 5, May 1978 (also to appear in: Ueckert, H., Rhenius, D. (eds.): Komplexe menschliche Informationsverarbeitung. Beiträge zur Tagung 'Kognitive Psychologie' in Hamburg. Bern: Huber)

Winograd, T. (1972): Understanding Natural Language. N.Y.: Academic