In: Proceedings of the First European Conference on Artificial Intelligence, ECAI-82, Orsay 1982, p. 133-138

USER MODELLING IN ANAPHORA GENERATION: ELLIPSIS AND DEFINITE DESCRIPTION

Anthony Jameson and Wolfgang Wahlster

Abstract. This paper shows how user modelling can improve the anaphoric utterances generated by a dialogue system. Two kinds of anaphora are examined: contextual ellipsis and the anaphoric use of singular definite noun phrases. In connection with ellipsis generation, anticipation of the way in which the user would be likely to reconstruct a given utterance can help to ensure that the system's utterances are not so brief as to be ambiguous or misleading. When generating noun phrases to characterize specific objects with which the user is not familiar, the system may take into account the existential assumptions, domain-related desires, and referential beliefs ascribed to the partner. These applications of user modelling are illustrated as realized in the dialogue system HAM-ANS, and some possible generalizations and extensions of the strategies described are discussed.

Keywords. Natural language generation, anaphora, user modelling, ellipsis, definite descriptions, interest-based answering

Introduction

Many writers on dialogue processes have pointed out ways in which it can be advantageous for a participant in a dialogue to be aware of the beliefs, desires, and goals of the other participant and to anticipate the partner's response to one's own utterances.

The importance of the user modelling principle has been demonstrated with its implementation in several AI systems (e.g. Cohen 1978, Allen 1979, Rich 1979; cf. also Jameson/Hoeppner/Wahlster 1980); the present paper focuses on possible roles for user modelling in the generation of utterances which must be interpreted against the background of previous utterances: The types of anaphora to be considered are contextual ellipsis and definite noun phrases.

Anticipation Feedback in the Generation of Elliptical Utterances

Many of the natural language AI systems developed to date have included the capability to process elliptical utterances of the user´s (among the more extended discussions of this topic are those of Grosz 1978 and Kwasny/Sondheimer 1981). As far as the generation of elliptical utterances is concerned, the state of research presents a paradox: On the one hand, most systems whose output takes the form of natural language utterances at all (as opposed, e.g., to lists, tables, or graphic displays), generate utterances which must be regarded as elliptical. But on the other hand, the generation of elliptical utterances has seldom (e.g. by Simmons/Slocum 1972) been viewed as a problem in its own right; elliptical utterances either follow predefined templates or verbalize data structures representing attributes, object names, or other nonpropositional information which have been returned by the part of the system responsible for determining the content of the system's utterance.

When the problem of ellipsis generation is treated explicitly, one of the questions which arises is that of how elliptical an utterance can be without becoming ambiguous or misleading. Consider for example, a dialogue system which plays the role of a hotel manager answering on the telephone the questions of a prospective guest about a particular room:

```
(1) User: Are there three beds and at least one desk
          in the room?
```

If the room in fact has three beds and three desks and the system is to mention the latter fact explicitly instead of simply answering 'Yes', it might respond

```
(2) System: Yes, three.
```

But the user is likely either to interpret “three” as referring to the number of beds or to be uncertain as to the correct interpretation.

The slightly less elliptical utterance

```
(3) System: Yes, three desks.
```

is still somewhat confusing, as it suggests that the system may be answering the question with respect to the desks only while avoiding a comment about the number of beds.

Complete clarity is provided only by an answer which includes considerable overlap with the question:

```
(4) System: Yes, three beds and three desks.
```

But how is the system to know that (2) and (3) would be too elliptical? The following example shows that no general rules relating the form of an elliptical utterance to that of the relevant previous utterance are likely to provide sufficient guidance in all cases.

If the room has three beds and two desks, and the system answers the question (1) with

```
(5) System: Yes, two.
```

no uncertainty of the sort caused by (2) can arise, since the possibility that “two” here refers to the number of beds is ruled out (given the fact that the answer is prefaced by the word 'yes').

It is difficult to conceive how a system could take into account the factors which make (5) acceptable but (2) unacceptable except by anticipating how the user is likely to interpret its utterances.

This conclusion should come as no surprise, since analogous processes have frequently been postulated in connection with other aspects of language production, such as the avoidance of referential and other ambiguities, the control of nuances, and the correction of mistakes in the speaker's own utterances after they have been articulated (see, e.g., Hoenkamp 1980 and Winograd 1977).

The term *anticipation feedback loop* will be used in the following to refer to the type of control algorithm which involves the simulation of the user's response to an utterance of the system's. In order to provide a concrete example of how such a control algorithm can be realized in a natural language dialogue system, we will now illustrate the operation of the ellipsis generation component of the system HAM-ANS, which, in one of its roles, simulates the hotel manager of the above examples (cf. Jameson et al. 1980). The generalizability of some of the features of this component will then be discussed.

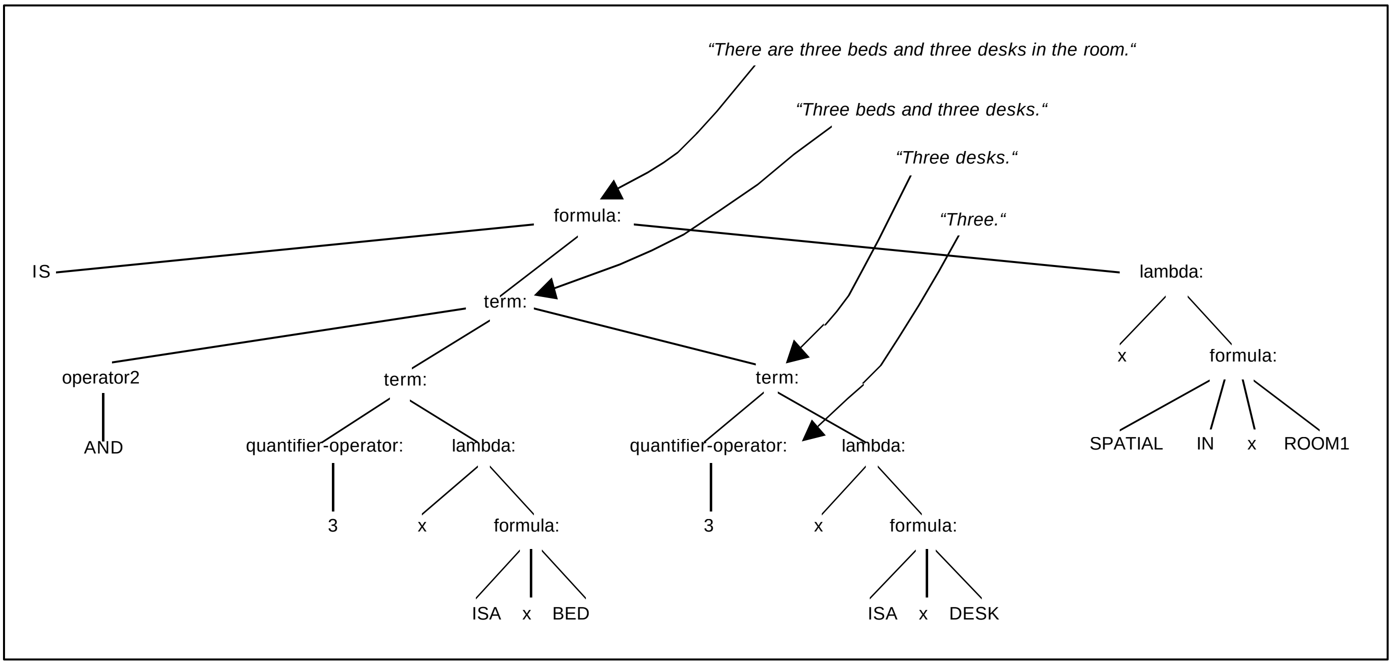

Figure 1: Structure of a formula representing an answer to the question 'Are there three beds and at least one desk in the room?'

Ellipsis Generation in HAM-ANS

The structure which represent the content of utterances to be generated by HAM-ANS are always complete formulas of the logic-based representation language used which express propositions. Because they often overlap considerably with the previous utterance, such formulas are not in general suitable for complete verbalization. On the other hand, there are two advantages that result from basing all language generation on structures representing propositions: (a) Such structures can be processed by the system in various other ways in addition to being verbalized. For instance, they can be added to the part of the user model which consists of the facts assumed to be known to the user. (b) The task of selecting appropriate elliptical formulations can be assigned to a specialized ellipsis generation component which is in a position to deal with complications of the sort illustrated by the examples presented above.

The first step in generating an utterance is the transformation of the formula to be verbalized so that its structure corresponds much more closely to the syntactic structure of the corresponding complete sentence. For example, the answer to (1) is transformed into the formula which is sketched in Figure 1 as a tree structure. A formula which has been transformed in this way has the property that each of its subtrees corresponds to a syntactic constituent of the corresponding natural language utterance. Since the ellipsis generation component is restricted to the generation of elliptical utterances which correspond to single, complete syntactic constituents of the given formula (other types of ellipsis will be mentioned in the next section), the subtrees of the transformed formula thus form the set of structures from which possible elliptical reductions of the formula can be selected.

The actual selection of one such subtree (which may be the entire formula, if no elliptical reduction is possible) is performed by the ellipsis generation component. The formula to be reduced is compared to a stored representation of the previous utterance - in the present example a formula corresponding to (1) which is essentially identical to the structure of Figure 1 except for the subtree representing 'three' in the term for 'three desks' .

Analysing the formulas for both the answer and the question from the top downward, the system makes a list of the successively smaller subtrees (pointed to by the arrows on the right in Figure 1) of the answer formula which can be obtained by eliminating immediate subtrees which are essentially identical in the two formulas. The smallest of the possible reductions found (here: 'three') represents the most concise possible verbalization of the answer which doesn't leave out any information not already contained in the question; but there is no guarantee that it will be interpreted correctly by the user. Rather than verbalizing it immediately, the system therefore attempts to reconstruct it as the user would, i.e. by determining how it fits into the structure of the original question.

In simulating the user's reconstruction attempt, the system activates the same ellipsis reconstruction component that it uses when it itself is interpreting an elliptical utterance of the user's. (This component is in many ways similar to the one described by Grosz 1980). It looks for subtrees within the representation of the previous utterance which are similar in structure and content to the elliptical structure. If there is one such subtree which is clearly a better candidate than all other subtrees, the elliptical structure is substituted for this subtree in the representation of the previous utterance in order to obtain a formula which presumably represents the meaning of the ellipsis. If there are either no such subtrees or two or more approximately equally good candidates, the ellipsis is considered not to be (uniquely) interpretable.

Applying this reconstruction strategy to the possible ellipsis 'three', the system finds as possible corresponding parts of the question not only the subtree representing 'at least one', but also the one representing 'three' within the term for 'three beds'. Since neither of these is judged to be clearly preferable to the other according to the similarity metric used, it cannot be assumed that the user would interpret the answer 'three' correctly.

The ellipsis reconstruction strategy is therefore applied to the next smallest of the possible elliptical structures, the one representing 'three desks'. Again two possible corresponding parts are found: not only the intended subtree, the term for 'at least one desk', but also the conjoined term for 'three beds and at least one desk'. The simpler subtree is judged to be a somewhat more plausible candidate than the conjoined term, being more similar in structure to the hypothetical elliptical utterance, but the difference in plausibility is not great enough to meet the system' s standard. It therefore calls the ellipsis reconstruction component a third time, applying it to 'three desks and three chairs'. This matches the term for 'three desks and at least one chair' in the question formula so closely that it is judged to be unambiguously interpretable and thus suitable for verbalization.

The Generalizability of This Approach

It is important to note how simple the anticipation feedback loop described in the preceding section is: When a candidate utterance is found to be ambiguous, no diagnosis is made of the ambiguity as a basis for the selection or construction of the next candidate; the next candidate is simply taken from the list of possibilities supplied by the ellipsis generation procedures. This kind of simplicity is only possible in a *local* anticipation feedback loop, i.e. one in which the generation procedures for one particular aspect of an utterance are linked with the recognition procedures for the same aspect. An example of a more global loop, in which error diagnosis and the generation of alternatives are necessarily more complicated, is the control structure sketched by Hoenkamp (1980).

HAM-ANS's generation and recognition components presuppose the existence of a previous utterance which the user will refer to when interpreting the system's utterance, i.e. they are restricted to *contextual ellipsis*. (The various forms of contextual ellipsis which HAM-ANS does not handle at present, such as *expansion* ellipsis, would not require major structural changes.) Extension to the generation of *telegraphic ellipsis*, i.e. elliptical utterances which are not linked structurally to any previous utterance, would not be straightforward. In particular, the idea of generating such utterances by deleting constituents of a complete utterance is less natural than it is for contextual ellipsis (cf. Gunter 1963, Betten 1976). On the other hand, if a generation algorithm for telegraphic ellipsis is available which in general produces several candidates, a local anticipation feedback loop can in principle be used to eliminate ambiguous candidates provided, of course, that the system's ellipsis recognition procedures are capable of processing the candidates generated. Although some of the ellipsis recognition strategies developed to date are applicable to telegraphic ellipsis (e.g. those described by Waltz 1978 and Harris 1980), it should be noted that utterances of this type which are generated by natural language dialogue systems often represent phrases which are specific to the role being taken by the system - e.g. 'Your name, please?' in the hotel situation - and which the system may thus not be equipped to deal with as input.

Not all conceivable ellipsis recognition components would lend themselves equally well to incorporation in a local anticipation feedback loop. First, any relevant knowledge used by the recognition component must be represented in a way which is neutral with regard to whether the elliptical utterance comes from the system or from the user. This is not the case, e.g., with the approach described by Bobrow, Kaplan, Kay, Norman, Thompson, and Winograd (1977), which is restricted to the recognition of the user's answers to the system's questions. Second, the recognition component must operate on the same kind of structure as those returned by the system's ellipsis generation component. This would not be the case, e.g., with Kwasny and Sondheimer's (1981) method, which uses the Augmented Transition Network path of the elliptical utterance.

The anticipation feedback loop described above represents a special case of user modelling in that (a) it is not based on any general assumptions about users beyond the implicit assumption that they interpret elliptical utterances in about the same way as the system itself, and (b) it does not involve the accumulation of specific assumptions about the particular user in the course of the dialogue. The user modelling to be described in the remainder of this paper is more typical in both of these respects.

The Role of the User Model in NP-Generation

Much has been written in AI and related disciplines about the comprehension of anaphoric definite descriptions (for a review see Sidner 1979), much less about their generation (for reviews see Wong 1975 and Heidorn 1977), and almost nothing about the influence of the user model on the cognitive processes underlying their generation - the problem to be investigated in this section of our paper.

In the hotel reservation situation mentioned above, where the system takes the part of a hotel manager who tries to persuade the user to book a particular room, the user model plays a major role in the system's NP-generation process[1]; e.g. when tendentious descriptions of the same object can create any of a variety of impressions of it in the mind of the hearer (as when the same bed is described as 'not too hard' or 'not all that soft') or when the system tries to make the user think that the room has several instances of a desired object when it in fact has only one - two phenomena which will be discussed in more detail below.

The degree of complexity of the NP-generation process is influenced by the type of situational context and by the system's conversational goal. It is useful to distinguish two types of situational contexts and two types of goals on the system's part:

(S1) Both partners are familiar with the details of the particular domain being discussed. For example, in one application, in which HAM-ANS is interfaced to a motion analysis system observing a street crossing, it is assumed that either both partners are looking at the scene or the user is familiar with it (or has a photograph of it).

(S2) The listener must be assumed to have no definite prior knowledge about the particular domain underlying the discourse. For example, in the hotel reservation situation at the beginning of the dialogue, the client, unlike the system, has no definite knowledge about the interior of the room being offered.

(G1) The system has no interest beyond helping the user by responding in a slavishly cooperative way. For example, in the scene analysis application the only task of the system is to give accurate information about the scene observed.

(G2) The system's intention is to influence a decision to be made by the dialogue partner. For example, in the hotel reservation situation the system tries to persuade the client to rent the room.

[1] Although the system HAM-ANS includes a pronominalization component, we must limit the present computational exploration to the pragmatics of singular *definite anaphoric reference* to nonpronominal NPs.

In this section we will focus on conversational settings which can be characterized by the combination of (S2) and (G2), since the user model plays a much more prominent role there than in situations characterized either by a combination of (S1) and (G1) or a combination of (S2) and (G1). Although most AI work has been devoted to the latter two combinations, we believe that for future AI applications the first combination may become equally important.

The type of conversational setting to which the hotel reservation situation belongs is characterized by the occurrence of definite descriptions in anaphoric chains initiated by sentences containing an indefinite description (e.g. '... next to an antique armchair ... The armchair...'). We will therefore specify when and how indefinite descriptions are generated by the NP-generation component. To provide a concrete background for the discussion, we first define the task of the NP-generation component in HAM-ANS more precisely and present the technical framework for the proposed computational method.

The Structure of NP-Generation in HAM-ANS

When the verbalization component encounters a logical term in the semantic structure passed to it, the NP-generation component (NP-GEN) is called.

If the term consists of a <quantifier-operator > and a < description > (see Jameson et al. 1980 for details) the complete structure of a definite or indefinite NP has been formed by the answer generation component, so that the only remaining task of NP-GEN is to translate the term into a preliminary NL structure.

If the term contains individual constants (e.g. DESK1) the subcomponent DESCRIBE expands all constants into a term which consists of a <quantifier-operator > and a <description>. Then the NP-generator is called recursively for the translation of the expanded term into a preliminary NL structure as mentioned above.

For example, the term *(6)* is replaced by the term *(7)* (as described below) and then translated into *(8).*

```
(6) (t-o: AND CHAIR1 BED1)

(7) (t-o: AND
      (t-s: (q-qt: A)
         (d-o: AND
         (lambda: x65 (af-a: ISA x65 CHAIR))
         (lambda: x63 (af-a: REF x63 COMFORTABLE))))
      (t-s: (q-d: THE (r: 1  1))
         (lambda: x64 (af-a: ISA x64 BED))))

(8) ((a (comfortable) chair) and (the bed))
```

It is useful to distinguish two possible readings of the NPs generated, depending on whether or not a particular object in the domain of discourse is being referred to: (a) the *extensional reading*, as in *'The TV set* has remote control ' and *'A shower* is available' ; and (b) the *intensional reading*, as in *'The TV set with remote control* is standard equipment in our rooms' and 'Unfortunately, *a shower* is not available'.

The semantic structure of NPs for which an intensional reading is intended is constructed during the answer generation phase; it would clearly be impossible to generate constant terms in this case. On the other hand, most NPs which are intended to have an extensional reading are generated by NP-GEN out of constants using the DESCRIBE component. Since in this paper we are focusing on the influence of the user model on the decision process modelled by the DESCRIBE component, we shall examine only the latter case in more detail.

The DESCRIBE component uses various knowledge sources, including referential and conceptual knowledge, inference rules, and a user model. The role of the following parts of the user model in the NP-generation process will be described: existential assumptions, domain-related desires, dimensional preferences, the partner's referential network, a coreference network, and the specific beliefs and desires of the partner.

To make the discussion more concrete, Figures 2 through 4 show simplified versions of some sections of the knowledge sources which are most important for the presentation to follow.

The Generation of Indefinite Descriptions

As long as the system is not trying to confuse or deceive the user by deliberate referential miscommunication, DESCRIBE generates an indefinite NP if the individual constant is not included in the coreference network, which links the system's referential knowledge to the user model. There are two possible reasons for the presence of such an entry in the coreference network: Either the system has already referred to the object or the existence of the object is implied by the existential assumptions supplied by the user model.

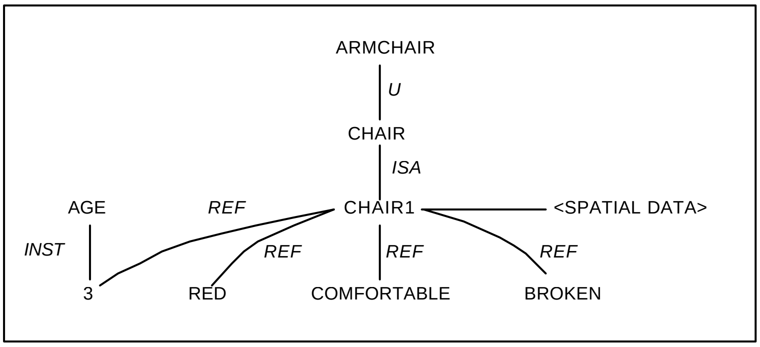

Figure 2: A small segment of the system's referential and conceptual knowledge.

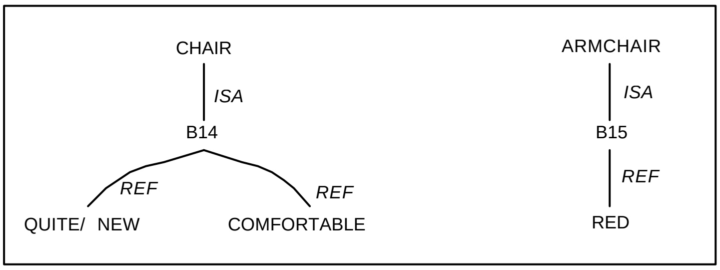

Figure 3: Two entries in the partner's referential network in the user model.

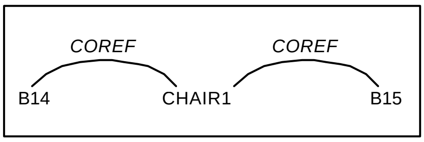

Figure 4: An entry in the coreference network.

In the latter case, the system generates a so-called *pragmatic anaphora*, i.e. an anaphora with no linguistic antecedent. For example, the system may regard as shared knowledge the fact that there are a bed, a door, and a window in the hotel room (see Clark/Marshall 1981). Then, when referring to these objects for the first time it doesn't use an indefinite NP, but rather generates a definite NP as a pragmatic anaphora.

Which existential assumptions are ascribed to the user depends in turn on the user model. For example, if the caller introduces himself as a senior manager, the system will ascribe to the user the existential assumptions associated with luxury hotels, e.g. that the room has a color TV, so that the system can speak of 'the color TV' without its having been mentioned previously.

When producing an indefinite description, the system generates a new individual constant, links it in the coreference network to the coreferential object in its referential knowledge, and stores all the information contained in the description in the partner's referential network.

When generating an indefinite NP, DESCRIBE has to look for an appropriate initial characterization, checking the desires ascribed to the partner. For example, in the given context it would not be advisable to introduce CHAIR1 of Figure 2 as 'a broken chair', but on the basis of a preference ascribed to the partner by default 'a comfortable chair' would be a good choice. If there is more specific information in the user model, e.g. that the client has a strong preference for antique furniture, HAM-ANS can describe CHAIR1 as 'a relatively old chair', taking advantage of its ability to generate vague expressions using hedged relative adjectives (see Wahlster 1981). Otherwise, if the client seems to dislike old furniture, the system would prefer the NP 'a quite new chair', as shown in Figure 3. In the current implementation, the amount of information attached to the introducing indefinite NP is controlled by a parameter called VOLUBILITY. In a more advanced version of HAM-ANS, the value of this variable should depend, among other things, on a continuous reevaluation of the system's success in creating a favorable impression of the room.

The Generation of Definite Descriptions

Once it has used an indefinite description, the system is entitled to use a definite NP. The process of generating anaphoric NPs is governed by the principle of *confusion avoidance* (Herrmann/Laucht 1976). When the object to be referred to is one of several objects which are stored in the partner's referential network and which belong to the same conceptual class, the system looks among the properties stored in the partner's referential net for a subset which distinguishes it from its cohyponyms. Since the search for a uniquely identifying description is independent of the particular role played by the user model, no special algorithm need be developed for this case. (In HAM-ANS the same algorithm is used as in the traffic situation; see Wahlster/Jameson/Hoeppner 1978.)

If there are several properties which refer only to the intended object, we have the case of *multiple codability* studied in psycholinguistics by Herrmann and Laucht (1976). If we omit the possibility of redundant coding from consideration, multiple codability calls for a decision process. HAM-ANS applies the following criteria successively to find a unique property:

- Checking the user model, choose the property which corresponds best to the desires of the partner.

- Choose the property which permits the best subjective discrimination, i.e. the property according to which the objects with which the object in question might be confused differ most strongly.

- Choose the property which has the highest rating on the dimensional preference scale ascribed to the partner (the default preference order in HAM-ANS is simply represented as a list of dimensions, e.g. color, contrast, size, age, ...).

The major new feature of the approach outlined above is that NP-generation in the type of conversational setting under investigation is viewed as an *interest-based* activity which is heavily influenced by the user model. One of the challenging aspects of the approach taken is that the proposed computational methods can also cope with intentional exaggeration and understatement with respect to the cardinality of certain sets of objects.

Suppose, for example, that the client has said 'I trust that there are several chairs in the room' (he plans to hold a business meeting in the hotel room) and HAM-ANS has generated a positive response. In order to reinforce the user´s impression that there are enough chairs, the system describes, e.g., CHAIR1 from Figure 2 (which it has previously referred to as 'quite a new comfortable chair') later in the dialogue with an indefinite NP, namely 'a red armchair' (see Figure 3). The system then generates two instances of a chair in the partner's referential net and links them in the coreference network to the same object identifier.

For an illustration of the understatement tactic, suppose the client wishes a single room but the hotel manager is forced to try to rent him a double room. In this situation the hotel manager can try to conceal the real number of beds in the room. After having referred to BED1, the system refers not only to it, but also to BED2 as 'the bed', so that the coreference net contains the assertions (COREF BED1 B1) and (COREF BED2 B1) and B1 is the only bed in the partner's referential world.

It now becomes obvious why the system must construct the coreference network and the partner's reference network and cannot use the simpler approach of marking in the system's referential knowledge the objects and properties which belong to the shared knowledge: In the preceding examples there is no simple correspondence between objects in the system's referential world and those in the partner's. Only by updating the coreference network can the system ensure consistency in intentional exaggeration and understatement.

In this paper we have viewed NP-generation as a process controlled by the user model. It should be clear that there are other factors such as focus (cf. Grosz 1981) and speech act planning (cf. Cohen 1981) which also play a major role in referent identification.

REFERENCES

Allen, J. F. *(1979): A plan-based approach to speech act recognition*. Unpublished PhD Thesis, Dept. of Computer Science, University of Toronto.

Betten, A. (1976): Ellipsen, Anakoluthe und Parenthesen. Fälle für Grammatik, Stilistik, Sprechakttheorie oder Konversationsanalyse? *Deutsche Sprache, 4*, 207-230**.**

Bobrow, D. G., Kaplan, R. M., Kay, M., Norman, D. A., Thompson, H., & Winograd, F. (1977): GUS, a frame-driven dialog system. *Artificial Intelligence*, 8, 155-173.

Clark, H. H., & Marshall, C. R. (1981): Definite reference and mutual knowledge. In A.K. Joshi, I. Sag, & B. Webber (Eds.), *Elements of discourse understanding*. Cambridge: Cambridge University Press.

Cohen, P. R. (1978): *On knowing what to say: Planning speech acts*. Dept. of Computer Science, University of Toronto, Technical Report 118.

Cohen, P. R. (1981): The need for referent identification as a planned action. *Proceedings of the 7th International Joint Conference on Artificial Intelligence*, Vancouver.

Grosz, B. J. (1978): Ellipsis. In D. E. Walker (Ed.), *Understanding spoken language*. New York: North-Holland.

Grosz, B. J. (1981): Focusing and description in natural language dialogues. In A.K. Joshi, I. Sag, & B. Webber (Eds.), *Elements of discourse understanding*. Cambridge: Cambridge University Press.

Gunter, R. (1963): Elliptical sentences in American English. *Lingua, 12*, 137-150.

v. Hahn , W. , Hoeppner , W. , Jameson , A. , & Wahlster, W. (1980): The anatomy of the natural language dialogue system HAM-RPM. In L. Bolc (Ed*.), Natural language based computer systems*. München/London: Hanser/Macmillan.

Harris, L. R. (1980): ROBOT: A high performance natural language interface for data base query. In L. Bolc (Ed.), *Natural language based computer systems*. München/London: Hanser/Macmillan.

Heidorn, G. E. (1977*): A method for generating noun phrases to identify nodes in a semantic network*. Yorktown Heights, N.Y.: IBM T.J. Watson Research Center.

Herrmann, T., & Laucht, M. (1976): On multiple verbal codability of objects*. Psychological Research, 38,* 355-368.

Hoenkamp, E. (1980): Spontaneous speech as a feedback process. *Proceedings of the AISB-80 Conference Artificial Intelligence*, Amsterdam.

Jameson, A., Hoeppner, W., & Wahlster, W. (1980): The natural language system HAM-RPM as a hotel manager: Some representational prerequisites. In R. Wilhelm (Ed.), *GI-10. Jahrestagung*, Saarbrücken. Berlin: Springer.

Kwasny, S. C., & Sondheimer, N. K. (1981): Relaxation techniques for parsing grammatically ill-formed input in natural language understanding systems. *American Journal of Computational Linguistics, 7*, 99-108.

Rich, E. (1979): *Building and exploiting user models*. Computer Science Dept., Carnegie-Mellon Univ., Technical Report CMU-CS-79-119.

Sidner, C. L. (1979): *Towards a computational theory of definite anaphora comprehension in English discourse*. MIT Al Laboratory, Technical Report TR-537.

Simmons, R., & Slocum, J. (1972): Generating English discourse from semantic networks. *Communications of the ACM, 15,* 891-905.

Wahlster, W. (1981): Implementing fuzziness in dialogue systems. In B. Rieger (Ed.), *Empirical semantics.* Bochum: Brockmeyer.

Wahlster, W., Jameson, A., & Hoeppner, W. (1978): Glancing, referring, and explaining in the dialogue system HAM-RPM*. American Journal of Computational Linguistics, Microfiche 77*, 53-67.

Waltz, D. L. (1978): An English language question answering system for a large relational database. *Communications of the ACM, 21,* 526-539.

Winograd, T. (1977): A framework for understanding discourse. In P.A. Carpenter & M.A. Just (Eds.), *Cognitive processes in comprehension.* Hillsdale, N.J.: Erlbaum.

Wong, H. K. T. (1975): *Generating English sentences from semantic structures*. Dept. of Computer Science, Univ. of Toronto, Technical Report No. 84.