

LINEARIZATION IN DESCRIBING SPATIAL NETWORKS

The topic of this paper is the way in which speakers order information in discourse. I will refer to this issue with the term "linearization", and will begin with two types of general remarks. The first one concerns the scope and relevance of the problem with reference to some existing literature. The second set of general remarks will be about the place of linearization in a theory of the speaker.

The following, and main part of this paper, will be a summary report of research of linearization in a limited, but well-defined domain of discourse, namely the description of spatial networks.

1. SCOPE AND RELEVANCE

One of the design characteristics of spoken language is its strict temporal ordering, or left-to-right structure. This property may not be convincingly present at the level of phonemes, but it certainly holds at the level of clauses: There is no way to co-articulate clauses; they have to be produced one after another. Since most speech in everyday life involves more than single isolated clauses, it is the rule rather than the exception that a speaker has to make decisions on the ordering of clauses. Such decisions, moreover, are not trivial. There are, in most circumstances, many conceivable ways of ordering the information to be expressed in discourse, but the speaker chooses one ordering rather than another, and this is most probably not a random choice. This choice problem may not be so apparent if the information has a very strict linear structure itself. If somebody reports an accident he has witnessed, a main part of the discourse will reflect the temporal order of the events that took place. Or if one describes a meal (Byrne, 1977), it is reasonable to begin with the first course, and to end with the last one. But, again, these cases are more the exception than the rule. Often, informational structures have no intrinsic linear order. Take apartment descriptions. Linde and Labov (1975) had informants describe the layout of their apartments. Such layouts are two-dimensional structures. They have to be mapped on a linear order of clauses in such a way that the listener can, within certain limits, reconstruct the two-dimensional picture. This mapping involves what I will call a

linearization-strategy, and it appears from Linde and Labov's work, that these strategies are quite systematic. In almost all cases, the informants' strategy is a kind of *tour*: they start at the front door, and move through the apartment room by room. At choice points they take one branch first, and, after finishing it, jump back to the last place of choice in order to select a next branch. This means that the tour is a quite abstract one: these jumps cannot be performed physically. They are mental switching operations which may reveal something important about discourse planning. Another similar case is the way in which people describe their living rooms. Again, an at least two-dimensional structure has to be mapped on a linear order of clauses. Veronika Ullmer-Ehrich (1981) in our institute has found that a main strategy of the subjects in describing their living rooms is to make what we call a *gaze tour*. They position themselves at the door, and gaze along the walls in either left-to-right or inverse fashion. They describe the pieces of furniture one-by-one in the order of the gaze tour. Speakers can often be the captives of their linearization strategy. Several of the informants entirely forgot to mention the furniture in the middle of the room, perhaps because the focus of gaze only followed the walls.

These cases are still relatively simple for the speaker. But what about explaining games? How would one explain the game of chess to somebody who is uninformed? The informational structure is so multidimensional that speakers are not able to set up a linearization strategy which guarantees full transmission of the relevant information. We found that discourse planning is quite chaotic here; the only way, apparently, is to teach the listener while playing.

Several forms of linearization are strongly interactive. The listener's reactions may become highly important for the speaker's linearization. This is, for instance, very apparent from an analysis of discourse planning in the Watergate-tapes (Linde and Goguen, 1977). But also in more monological forms of discourse planning, the listener's role is important. After all, the speaker wants the listener to understand a particular informational structure. He therefore has to take into account the listener's presumed foreknowledge and processing capacities. This will turn out to be an essential issue in the analysis of spatial network descriptions.

Let me finish this set of general remarks by mentioning one other set of linearization studies in the literature. There are at least three empirical studies on how people give road directions (Klein, 1981; Wunderlich, 1981; Munro, 1977). Also here, the speaker makes a tour from source to goal, a tour which is laid out via a system of landmarks. These, and all other cases of spatial

discourse I have seen, are full of deictic devices, which, in their turn, depend strongly on the linearization strategy chosen.

2. THE PLACE OF LINEARIZATION IN A THEORY OF THE SPEAKER

The ubiquity of linearization in everyday language use does not imply that it involves a unified psychological mechanism. If we manage to cut up the speaker's nature at its joint — and I think a theory of the speaker has to do that — it is not self-evident that linearization will come out as a separate limb, or bone. Since, however, a theory of the speaker does not exist, all serious candidates for joints and bones are worth considering. The main distinction I would like to propose for a theory of the speaker (following Kempen, 1977) is between the processes involved in the genesis of the *ideas* underlying speech, and the processes involved in the choice of linguistic *forms* for then-expressions. The first set of processes, which may be called *conceptualizing*, may be conceived of as the set of non-linguistic preliminaries to producing or sustaining an utterance. It should include, among other things, the development of communicative intentions, the selection of the appropriate information from the knowledge base, and the linearization of this information. The second set of processes, *formulating*, give linguistic form to the generated

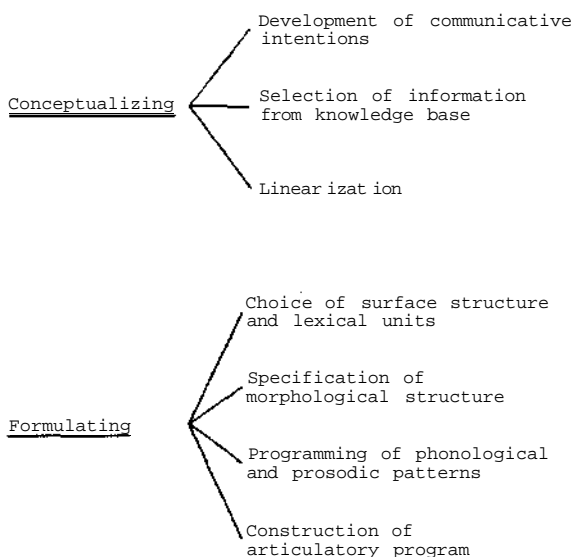


Fig. 1. Processes involved in speaking.

intentions and contents. Among the activities involved will be the choice of surface structure and of lexical units, the specification of morphological structure, the programming of phonological and prosodic patterns, and the construction of an articulatory program. I would like to see as much as possible work done by the conceptual preliminaries, so that the formulating mechanism can operate in a highly automatized fashion on highly specified conceptual input.

In this conception, linearization has to do with the conceptual preliminaries only, i.e., it is a non-linguistic process. It is, therefore, no surprise that linearization can be observed in other human behavior as well, e.g. in walking through a museum, or in playing music.

Even if linearization is a purely conceptual issue, it need not be a unified process. If linearization is fully determined by the content of the discourse, it would still make no sense to study linearization as a relatively autonomous process. But, in my view, there is reason to suppose that linearization shows functional properties which are independent of the knowledge base, and which may turn out to be fairly general for different types of discourse. These properties, I would like to claim, come forth from the economy of short term memory. The present approach is, therefore, orthogonal to content analysis explanations of discourse structure. I will now try and make these claims more concrete by working in some detail through a case of linearization in a simple well-defined domain.

3. THE DOMAIN OF DESCRIPTIONS

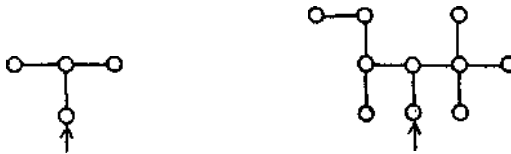
The spatial structures I have worked with are gridlike networks as displayed in Figure 2. They consist of nodes and arcs, and are always connected. The nodes are colored dots of degree 1, 2, 3 or 4, i.e. they have one to four arcs, and are, correspondingly, called single, dual, triple and quadruple nodes. The arcs are equally long, and are arranged either horizontally or vertically. It is helpful to distinguish three types of structure in this domain: The first type is the *linear* structure. It consists of a string of dual nodes, with two single nodes at the ends (see Figure 2a). The second type is *hierarchical*. It contains triple or quadruple nodes (as in Figure 2b). These nodes will, of course, be choice points for the subject who has to describe the network. The third type of structure is the *loop* (see Figure 2c). It creates all sorts of special problems in linearization, as we will see shortly.

In the experimental situation, the subject is visually presented with a network, and is required to describe the network into a tape recorder, beginning

a. Linear networks



b. Hierarchical networks



c. Loops



Fig. 2. Examples of networks used.

at the arrow, in such a way that the listener will be able to draw the network from the tape. It is further mentioned to the speaker that the listener has seen the same example networks as the speaker, so that he knows the domain of spatial structures under concern.

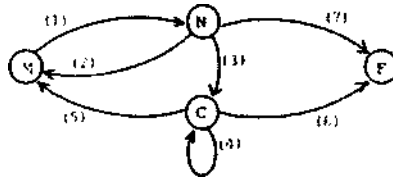
By choosing this domain, I hoped, on the one hand, to capture some of the important aspects of other spatial domains, like city maps, apartment layouts, electric circuits, etc, and on the other hand, the domain would allow for a precise formulation of a linearization model, which could then be tested experimentally. I would now like to present two such models. The first one is a bit more speaker-oriented, the second one more listener-oriented, or "cooperative".

4. A SPEAKER-ORIENTED MODEL OF LINEARIZATION

The first model was constructed so as to capture the main features of what

Linde and Labov observed in their study of apartment descriptions. A description is like a tour, i.e. the moves preserve maximal spatial connectedness. Jumps only occur back to unfinished choice points, and it seems that not only moves, but also jumps are as small as possible, namely back to the last choice point. Whatever the nature of the model, it should fulfill the requirement that it generates at least one complete description for every network in the domain.

I have designed an augmented transition network (ATN) that will do just that. It is given in Figure 3. The ATN consists of four states and a set of transitions between them. In some cases different transitions lead to the same change of state. In the Figure these are collapsed into one arc for the sake of visual simplicity. Each of the transitions has one condition-action pair associated to it. If the condition of a transition is fulfilled, the transition may be made under simultaneous execution of the action.



| Transition | Condition | Action |
|------------|--|---|
| (1a) | network entrance | make entry statement reduce valence of entrance node by 1. |
| (b) | other than net- work entrance | (selectand)describemove, reduce val. of node left and of node entered by 1. |
| (2a) | node val =1 | describe node |
| (b) | node is loop entrance and val >1 | describe node |
| (3) | node valence >1 | describe node |
| (4) | node valence >1 | seek M/F |
| (5) | node valence =1 | - |
| (6) | node valence =0 | - |
| (7) | node valence =0 | describe node |

enter loop if loop has been
recognized at transition (2).

Fig. 3. Speaker-ATN for network descriptions (speaker-oriented).

The network can recursively call itself, as we shall see in a moment. This is a necessary device for dealing with choice points. This recursive property requires a push-down store which keeps the addresses to which the ATN has to pop back after finishing an embedded operation. We must assume that the ATN has something like a semantics, i.e. the operations are performed under reference to the network under concern. One way to put this is that the ATN moves a pointer over the spatial network, and that the actions refer to the information at the pointer. More specifically, the various "describe" actions (see Figure 3) should be true for the information at the pointer. Finally, the ATN keeps a register of so-called node-valences. Initially, the valence of the node is its degree: 4 for quadruple nodes, 3 for triple nodes, etc. Every time a node is entered or left, the valence of the node is reduced by 1. Semantically, this amounts to marking these entries and exists of nodes.

If the network is linear, as in Figure 2a, it is easy to see how the ATN works. Starting in state M (the initial or move state), the ATN will oscillate between states M and N (the node state), and it will describe the successive nodes and arcs by performing the actions that correspond to transitions 1b and 2a. When it finally meets a single node, i.e., a node which after being entered has valence 0, transition 7 is made, and the final state F is reached. In short, the ATN predicts that linear networks are described in a connected way from beginning to end.

If a structure is hierarchical, as in Figure 2b, the recursive power of the ATN has to be used. If, in state N, the pointer is at a choice point, i.e. with valence > 1 , transition 3 has to be made, by which the ATN reaches the choice state C. From here it can only make transition 4, i.e. it has to transverse the network from M to F as an embedded action. The choice state is stored on the push-down store, which means, semantically, that the address of the choice point is stored, while the ATN proceeds to describe one of the branches from the choice point. If such a branch is linear, it is automatically described by oscillating between states M and N, until the end of the string, and there-with state F is reached. Control transfers back to C. If the choice node has only one valence left, transition 5 will be made, and from state M, the final branch from the choice node will be described. If two valences are left, another push operation (transition 4) will be performed, etc. This way of dealing with hierarchical structures means that the ATN will always return to the last unfinished choice point, and will never skip.

Really complicated is the situation with loops. The ATN can handle them in two ways. The first one is what I would like to call the dumb way. Consider the loop on the left in Figure 2c. The ATN enters the choice point. It will

be in state N, and will transit to state C, since the valence of the node is greater than 1 (transition 3). From the choice state it will enter the loop in subroutine-mode, i.e. with the choice address on store. Upon returning to the same node, it finds that the node has valence zero, which is the condition for transition 7 to the final state. Since the push-down store is not empty, control shifts back to state C. The node valence, however, is still zero, so that transition 6 is made, and the final state is reached with empty store. The dumbness of this procedure is that (a) unnecessary storage is involved, and (b) several additional computations have to be made at the end.

The clever way is to recognize the loop, and *not* store the choice point at entering the loop. This is the condition for transition 2b. The ATN just keeps oscillating between M and N, without recursion, and therefore without storage. One could say that a timely recognized loop can be "linearized", or "unfolded". In this way the number of actual choice points can be reduced.

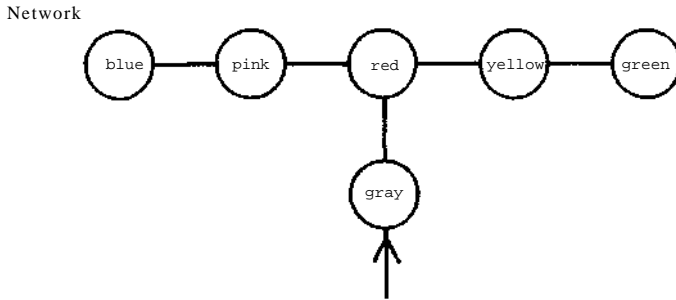
I have a proof (see Acknowledgement) that this ATN generates at least one complete description for every network in the class.

It may be clarifying to see the ATN at work for an actual description obtained in the experiment to be reported furtheron. Figure 4 gives the network described, the (Dutch) description, its English translation, and its breakdown according to the ATN of Figure 3.

Clearly the ATN of Figure 3 is non-deterministic, since it does not prescribe which arc has to be taken first at a choice point. I will return to this after presentation of the second ATN.

In order to complete the discussion of this ATN let us consider the listener's role. The listener can also be modelled as an ATN. The ATN should be able to draw the spatial network on the basis of the description generated by the speaker-ATN. The corresponding listener-ATN is presented in Figure 5. It is an almost complete image of the speaker-ATN. In fact, it functions exactly in a matching fashion as long as the clever loop-procedure is not used. If the clever way is used by the speaker, the listener-ATN will perform the equivalent of the dumb procedure. Only if the speaker were to mention that he is entering a loop (which is not done by the speaker-ATN, but which could easily be added), could the listener also be clever. Apart from this loop-issue, the storage requirements for speaker and listener form a perfect match.

This latter fact means that, if the network gets complicated, the load on the listener may become quite substantial, only because of multiple embeddings of choice-points. The next ATN is designed to make the speaker a bit more cooperative, so that the listener has an easier task to perform.



Description we beginnen bij grijs.
 van grijs kun je naar boven naar rood.
 Van rood kun je naar rechts naar geel
 en nog verder naar rechts naar groen.
 Je kunt vanuit rood ook naar links,
 dan kom je eerst bij roze terecht,
 en nog verder naar links,
 dan kom je bij blauw.

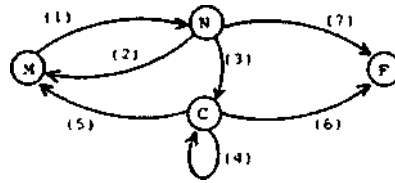
English translation and ATN-breakdown

| Statement | Transition |
|------------------------------|--------------------------------|
| We begin at | Ma) entry statement. |
| gray | (2a) describe node |
| From gray one can go up to | (1b) describe move |
| rod | (3) describe node |
| | (4) seek MF, stack state C. |
| From red one can go right to | (1b) describe move |
| yellow | (2a) describe node |
| and still further right to | (1b) describe move |
| green | (7) describe node |
| | unstack and return to state C. |
| | (5) - |
| From red one can also go | (1b) describe move |
| left, then you first reach | |
| pink | (2a) describe node |
| and still further left. | (1b) describe move |
| then you come to | |
| blue | (7/ describe node |

Fig. 4. Analysis of actual description according to speaker-oriented ATN of Fig. 3.

5. A LISTENER-ORIENTED MODEL OF LINEARIZATION

The load on the listener is caused by the recursiveness of the speaker-ATN. Would it be possible to build a non-recursive speaker? From listening to one subject in a pilot study, I got the idea that this could be done. That subject did not *jump* back to the last choice point, but she *moved* back, step by step. So the return from green to red in Figure 4 would be described as follows:

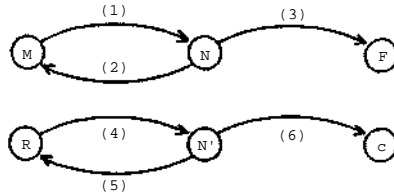


| Transition | Condition | Action |
|------------|--|-------------------------------------|
| (1a) | entry statement | draw entry arc |
| 1b) | movedescription | draw arc from node |
| (2a) | node descr., val =1 | draw node to arc, mark arc exits |
| (b) | node descr., val >1 and loop mentioned | draw node to arc, mark arc exits |
| (3) | node val >1 | draw node to arc, mark arc exits |
| (4) | node val >1 | seek M/F |
| (5) | node val =1 | - |
| (6) | node val =0 | - |
| (7) | node val =0 | draw node |

Fig. 5. Listener-ATN to ATN of Fig. 3.

"From green back again to yellow, and from yellow again to red". Inspired by this procedure, I designed an ATN which can be called listener-oriented, for reasons that will become clear shortly. It is presented in Figure 6.

This ATN consists of two networks. The top one generates description of arcs and nodes, it could be called the "move-network". The bottom one is the "return-network". It generates the return from end-nodes, by mentioning in reverse order the nodes and arcs that had been described before. The ATN is non-recursive, and has, correspondingly no push-down store. Instead, it has a so-called "unfinished node counter". Every time the move-network meets a node with valence > 1, i.e., a choice node with at least two unfinished arcs, the counter value is increased by 1, and one of the arcs is entered (see transition 2b). So, the address of the node is not stored, but only the fact that somewhere there is an unfinished choice node. The ATN proceeds through the network, oscillating between M and N, counting unfinished choice nodes, until it reaches a node of valence 0. It will then transfer control to the return-network, which proceeds by oscillating between R and N', until it meets a node with valence 1 or 2, i.e., an unfinished choice point. If only one arc is still to be done, the ATN enters it, and reduces the counter-value by 1,



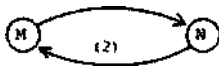
| Transition | Condition | Action |
|------------|------------------------------|---|
| (1a) | network entrance | make entry statement, reduce val of entrance node by 1. |
| (b) | other than network entrance | •(select and)describe move, reduce val. of node left and node entered by 1. |
| (2a) | node val =1 | describe node |
| (b) | node val >1, non loop | describe node, UNC n = n + 1 |
| (c) | loop entry | describe node |
| <d1 | node valence =0, and UNC n>0 | describe node, seek R/C |
| (3) | node valence =0 and UNC n=0 | - |
| (4) | node valence =0 | mention return move along non-r-marked arc, r-mark arc |
| (5) | - | mention node |
| !6a> | node val =1 | mention node, UNC n = n - 1 |
| 1b) | node val =2 | mention node |
| | | •if loop, enter loop |

Fig. 6. Non-recursive speaker-ATN (listener-oriented).

since there is one fewer unfinished choice node. If there are two open arcs left, the count is obviously not reduced. The return-network, prohibits returning twice along the same arc, which is a necessary but unimportant technicality. Loops are hard for this ATN. Without going into details, it should be remarked that loops can only be handled in a way corresponding to the clever procedure of the earlier ATN, i.e., a loop *has* to be recognized by the speaker.

For the listener, this speaker-behavior is really very easy. The corresponding listener-ATN is presented in Figure 7. A listener without short-term memory would still be successful, so to speak.

Though this pair of networks is far easier in terms of memory load than the former pair, a price has to be paid. Firstly, the descriptions will be up



| Transition | Condition | Action |
|------------|------------------------|-------------------------------------|
| (1a) | entry statement | draw entry arc |
| (b) | move description | draw arc from node |
| (c) | return move mention | more along drawn arc |
| (2a) | node description | draw node to arc, mark arc exits |
| 1b) | node mention | check node |

Fig. 7. Listener-ATN to ATN of Fig. 6.

to 50% longer, due to the return moves. Secondly, if loops are missed, completeness of description is no longer guaranteed.

like the former speaker-ATN, the present one is non-deterministic. At choice points the speaker can go one arc or another. I would now like to discuss some principles of choice, which can probabilistically predict which arc will be chosen at choice points. These principles are of a *local* and of a *global* nature.

6. LOCAL CONSTRAINTS ON CHOICE

The purely local situation at a choice point is always one of four possibilities. They are given in Figure 8. The first three situations are binary. In each case, there is one degree of freedom for predicting the choice probabilities. The fourth situation allows for six possible orders of choosing the different arcs, i.e., there are 5 degrees of freedom. A simple theory of local constraints can be designed to describe these 8 degrees of freedom in total with just two parameters. According to this theory, a speaker would, firstly, decide whether he will go straight or not. The probability of going straight, s , is the first parameter. For all cases that the subject cannot or will not go straight, a further decision is taken whether to go right or left, if that choice is still open. The second parameter, then, is the probability r of choosing right. These two parameters will predict all cases in Figure 8. I will return to this in discussing the experimental results.





| <u>Node type</u> | <u>order</u> | <u>Probability</u> |
|---|---|---|
|  | $\left\{ \begin{array}{l} \text{straight} \rightarrow \text{right} \\ \text{right} \rightarrow \text{straight} \end{array} \right.$ | $\left\{ \begin{array}{l} s \\ 1 - s \end{array} \right.$ |
|  | $\left\{ \begin{array}{l} \text{straight} \rightarrow \text{left} \\ \text{left} \rightarrow \text{straight} \end{array} \right.$ | $\left\{ \begin{array}{l} s \\ 1 - s \end{array} \right.$ |
|  | $\left\{ \begin{array}{l} \text{right} \rightarrow \text{left} \\ \text{left} \rightarrow \text{right} \end{array} \right.$ | $\left\{ \begin{array}{l} r \\ 1 - r \end{array} \right.$ |
|  | $\left\{ \begin{array}{l} \text{straight} \rightarrow \text{right} \rightarrow \text{left} \\ \text{straight} \rightarrow \text{left} \rightarrow \text{right} \\ \text{right} \rightarrow \text{straight} \rightarrow \text{left} \\ \text{right} \rightarrow \text{left} \rightarrow \text{straight} \\ \text{left} \rightarrow \text{straight} \rightarrow \text{right} \\ \text{left} \rightarrow \text{right} \rightarrow \text{straight} \end{array} \right.$ | $\left\{ \begin{array}{l} s \cdot r \\ s(1 - r) \\ (1 - s)r s \\ r(1 - s)^2 \\ (1 - s)(1 - r)s \\ (1 - s)^2(1 - r) \end{array} \right.$ |

Fig. 8. Four local choice situations.

7. GLOBAL PRINCIPLES OF CHOICE

Local constraints derive from the structure of the choice node itself. Global constraints emerge from structural properties of the network as a whole. My conjecture is that such global principles of choice are quite general in nature and will apply to linearization in quite diverse domains of discourse.

The various global constraints on choice can be derived from one underlying principle, which I will call the

Principle of minimal effort. Everything else being equal, speakers will prefer to give descriptions which minimize the number and duration of elements on store, and the length of the description. There will be a preference for using structural information whenever it can be instrumental to such minimization.

The first global constraint we derive from this principle of minimal effort is depicted in Figure 9. The constraint can be formulated as follows:

GCI: Everything else being equal, the probability that from a choice node a shorter branch is described before a longer branch is greater than 0.5.

The figure shows a choice situation, and a table of the elements on store if the first (speaker-oriented) ATN linearizes in one way or the other. Clearly, there is a longer duration of storage if the long branch is taken first. The prediction, therefore, is that the shorter branch will have a higher probability to be taken first.

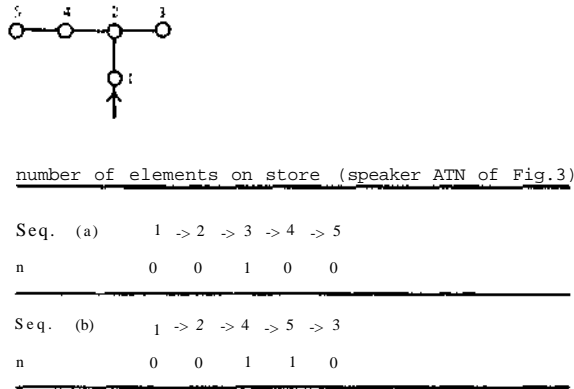


Fig. 9. Duration of load for two linearizations of network.

The listener-oriented speaker will make the same choice, but for a different reason. For him, there is no difference in storage, but clearly the description will be longer if the long branch is described first; this would namely involve a longer return move.

The second global constraint has to do with the number, not the duration of items on store. Contrary to the first constraint, it is only valid for the speaker-oriented ATN.

GC2: Everything else being equal, the probability of branches with fewer embedded choice points to be described before branches with more embedded choice points is greater than 0.5.

The constraint is illustrated in Figure 10. It presents the storage load for

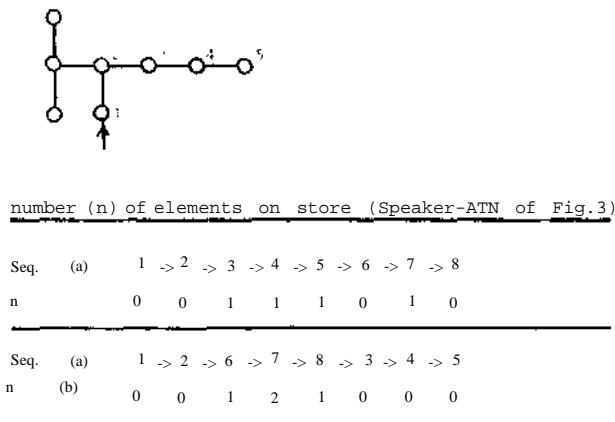


Fig. 10. Size of load for two linearizations of network.

going either right or left first. Maximal storage load for the ATN of Figure 3 is, apparently, less if the linear branch is done before the hierarchical one. Though for this figure the *average* load is the same for the two linearizations, the *maximal* load should be the correct measure, since our short term memory does not resist a continuing but slight load. It is more like a bottle-neck: it breaks down at moments of overload.

It should be noted that this second global constraint is formally identical to Yngve's depth hypothesis. In a similar fashion, it predicts that linearization creates maximally right-branching discourse structures. I believe that this is a very general and fruitful hypothesis about linearization in a large variety of discourse domains.

I mentioned that the listener-oriented ATN would not show this global constraint. The reason is simple: it does not involve any storage.

A last global constraint I want to make is that speakers prefer to be clever in dealing with loops:

GC3: Everything else being equal, the probability that, at choice points, loops are described before other branches is greater than 0.5.

This constraint is exemplified in Figure 11. We have seen that the clever

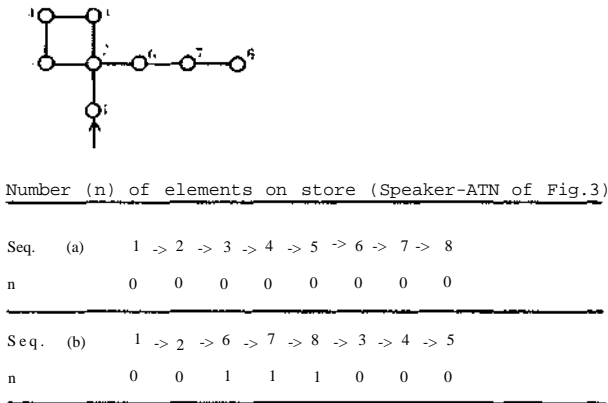


Fig. 11. Size of load for two linearizations of a loop.

procedure is to "unfold" the loop. In that case, no storage of the choice node is necessary, but the requirement is that the loop is done first. This can all be seen in the table of storage load in the figure.

This constraint is a probabilistic one for the first speaker-ATN. For the listener-oriented ATN it is a deterministic requirement, since this ATN only allows for the clever procedure.

8. SOME EXPERIMENTAL FINDINGS

Let us now turn to some of our experimental findings. For the sake of brevity. I will not present the full details of experiment and results here. It should suffice to say that I constructed 53 different networks in order to test the ATNs, and the probabilistic constraints. We had correspondingly, 53 subjects who described all these networks in different orders.

The characteristic difference between the ATNs is that the first one predicts *jumps* back to choice points, and the second one return *moves*. It turned out that of our 53 subjects 33 were exclusively jumpers, 16 were exclusively movers, and only 4 both jumped and moved. It seems, therefore, that there are two very consistent linearization types. The next question is, of course, whether the two ATNs correctly predict other aspects of linearization for these two types of subject. In the following, I will therefore report the results for "jumpers" and "movers" separately. I will proceed as follows. Firstly, I will give some data on linear, hierarchical, and loop structures. Then I will mention some results on local constraints. And finally, the global constraints will be considered.

The experimental set contained 7 *linear networks*. They are given in Figure 12. Both ATNs predict full connectivity here. There is not a single case in the data where the description is discontinuous: all follow the network from node to node.

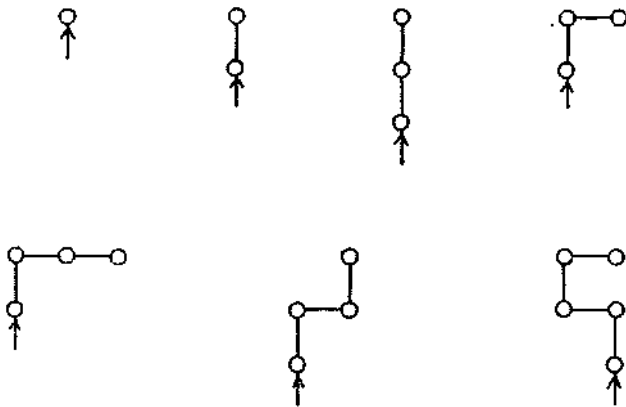


Fig. 12. Linear networks in the experiment.

There were various *hierarchical networks* in the set. One network was especially designed to test the last-in-first-out property of the push-down store. That is, it tested whether the subject always returns to the last unfinished

node. This should be true for both ATNs, although for different reasons. The pattern, given in Figure 13, is embedded 7 times.

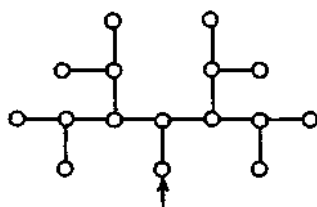


Fig. 13. Multiply hierarchical network.

Of the 33 jumpers, only 3 show deviations in the return order of choice points. Two of these make just one reversal of order, and one subject makes two such inversions.

The 16 movers show no violations whatsoever, which is of course a consequence of strictly adhering to the moving-strategy.

Since the property is quite essential to both ATNs, these results give substantial support to a major feature of their design.

Loops, finally, create problems for both jumpers and movers. The loops used in the experiment are depicted in Figure 14. I will only give a summary of the results for these networks.

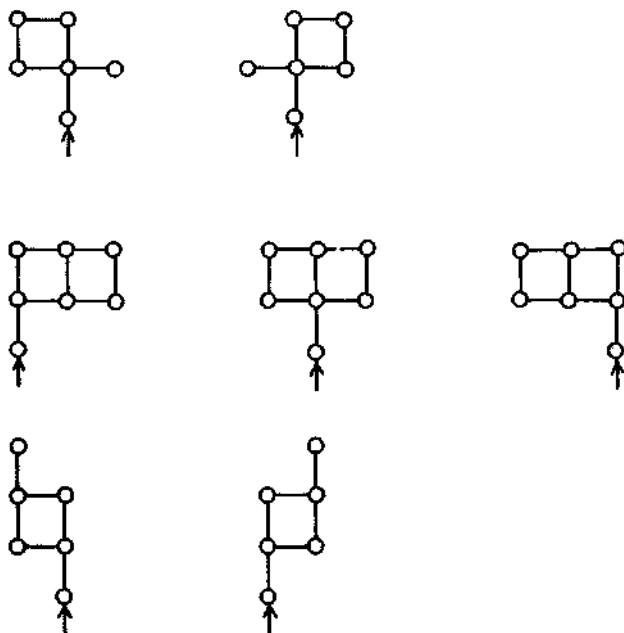


Fig. 14. Loops used for testing ATNs.

Movers, it turns out, only do what their ATN predicts in 50% of the cases. In no less than 33% of the cases, they follow the order predicted, but stop short of a *complete* description. In only 17% of their descriptions does the order really violate the ATN predictions. As expected (see section 5) loops make movers very error-prone.

Jumpers linearize, as predicted by their ATN, in 65% of the cases. In 8%, they stay incomplete, and in 27% of the cases, their order contradicts predictions from the ATN. This is substantial, and requires, in my view, some further theoretical work.

The *local constraints* were tested with the networks in Figure 15. I made a least square estimate for the two-parameter model, and tested the fit by $x^2 \cdot$

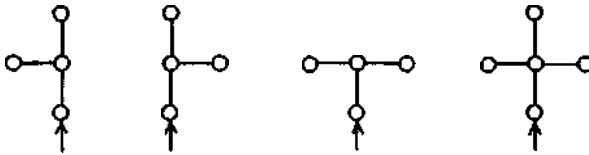


Fig. 15. Networks used for testing local constraints.

Neither jumpers nor movers showed any significant deviations from the model ($.50 > p > .30$ and $.30 > p > .20$, respectively). But the two types differed in their first "straight-on" parameter. Jumpers were more inclined to go straight first ($s = 0.52$) than movers ($s = 0.38$).

The *first global constraint* (GC1) short branches before long branches, was tested with the patterns in Figure 16. The constraint was strongly confirmed for both jumpers and movers for all patterns except 16(e) and (f). It could be the case that the local straight-on decision has precedence over global constraints, but this has to be further studied.

The *second global constraint* (GC2), Yngve's depth hypothesis, was tested with the networks in Figure 17. For jumpers, it was strongly confirmed for all networks, except 17 (c) and (d). It seems that jumpers do seek the least complex branch first, but don't distinguish very much between embedding and form complexity.

The movers, as predicted in section 7, do not follow the constraint for any of the networks. But a surprising additional finding is that they show the inverse for all patterns, i.e. they seem to *seek* complexity. This was not predicted and requires further theoretical analysis.

The *third global constraint* (GC3), finally, which predicts that loops have precedence, was tested with the patterns in Figure 18. Jumpers enter the loop first in 68% of the cases, movers do so in 80%. This seems promising, but care

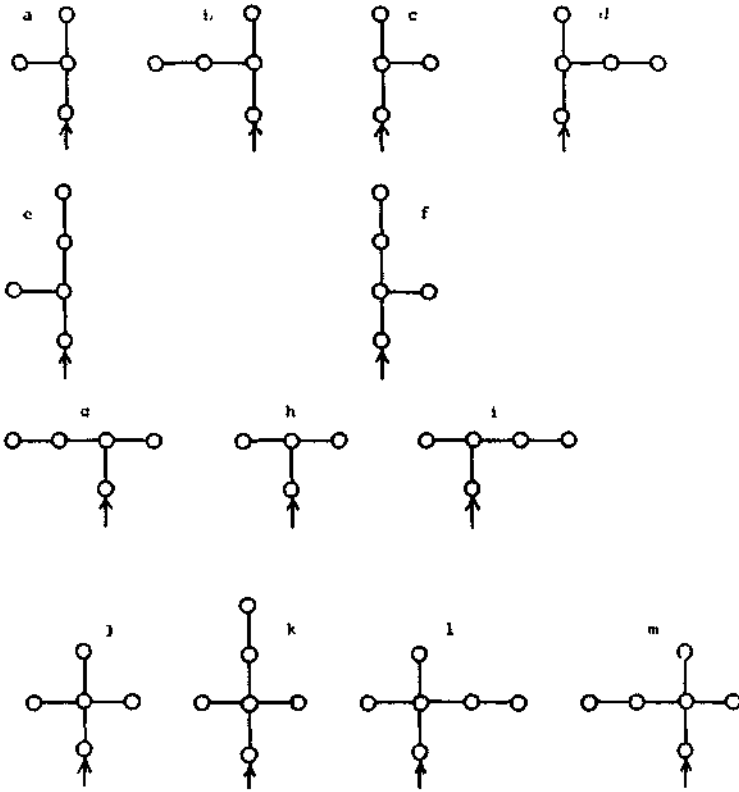


Fig. 16. Networks used for testing first global constraints (GC1).

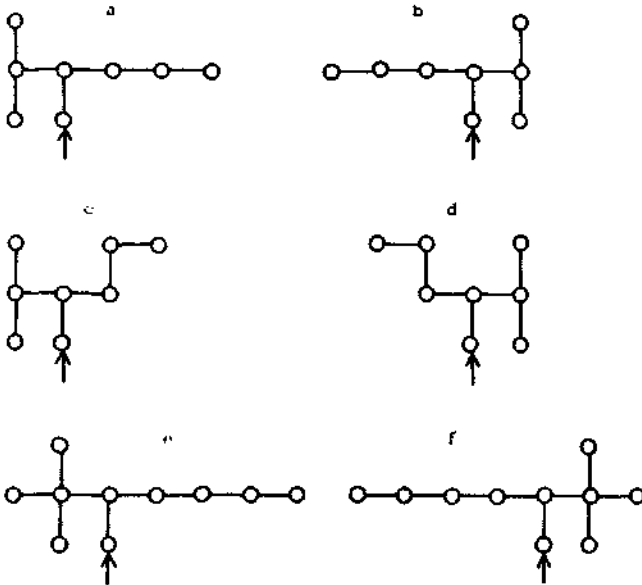


Fig. 17. Networks used for testing second global constraint (GC2).

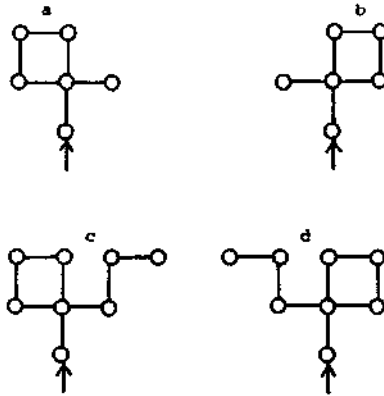


Fig. 18. Networks used for testing third global constraint (GC3).

is necessary in the choosing of an acceptable null-hypothesis. It is not reasonable to put H_0 at 50%, since the speaker can always choose between three arcs, two of which enter the loop. If H_0 is set at 67%, only movers show a slight tendency to prefer the loop. It seems, then, that the clever procedure of "unfolding" the loop is not very popular with our subjects.

9. CONCLUDING REMARKS

It would have been surprising to find that the ATN-models worked in all cases. But up to the level of hierarchical spatial structures, they are in almost faultless correspondence to the speakers' behavior. It should be noted that this correspondence does not depend on the particular format of the ATNs, but on a few general construction principles: connectivity of moves in both ATNs, first-in-last-out storage of choice nodes in the speaker-oriented ATN, no such storage but connected return moves in the listener-oriented ATN. It should not be difficult to realize these same principles in other ways: in the speaker-oriented ATN one could, for instance, reduce the push-down store and the node valence register to a single mechanism (as suggested by Dr. Hendrix during the conference), and similar changes can be made in the listener-oriented ATN without affecting any of the predictions made, and thus without affecting the empirical fit. For loop-structures, however, both ATNs are less accurate, and so will be other variants built according to the same principles. The main deviations from the ATN predictions are, apart from incompleteness, the "cutting up" of a loop, thus treating it as a hierarchical 2-branch structure. Why is it that speakers find it hard to deal with loops

in a connected way? Psychologically, this may have to do with difficulties subjects encounter in maintaining their original deictic orientation while turning through a loop. This suggestion (made by Dr. Partee at the conference) should be followed up since it might lead to an additional construction principle in linearization models (cf. Levelt, 1981).

Taken together, the results so far underline the basic importance for discourse generation of principles such as preservation of connectivity, first-in-last-out treatment of choice points, and a general minimal effort principle based on short term memory economy. My conjecture is that these functional principles of linearization are not limited to this particular domain, but apply more generally to other types of discourse as well. (See J. Mandler, 1978, who shows the working of similar principles in the retrieval of stories.) The combination of a first-in-last-out principle, and minimization of memory load predicts a prevalence of hierarchical right-branching structures in discourse, an interesting property to look for.

Max-Planck-Institut für Psycholinguistik, Nijmegen

ACKNOWLEDGMENT

I am grateful to Wilma Silvius who ran the experiments, and analyzed the protocols. Dr. R. H. Jeurissen, Department of Mathematics, Nijmegen University, provided a completeness proof for a set of linearization devices, including the ATN of Fig. 3. Finally, thanks are due to Dr. Tony Ades who suggested changes and improvements, of which several were realized in this final version of the text.

REFERENCES

- Byrne, R.: (1977) 'Planning meals: problem solving on a real data-base,' *Cognition* 5, 287-332.
- Kempen, G.: (1977) 'Conceptualizing and formulating in sentence production,' in S. Rosenberg (ed.), *Sentence Production: Developments in Research and Theory*, Erlbaum, Hillsdale.
- Klein, W.: (1981) 'Some aspects of route directions', in R. J. Jarvella and W. Klein (eds.), *Speech, Place, and Action. Studies of Language in Context*, Wiley, Chichester.
- Levelt, W. J. M.: (1981) 'Cognitive styles in the use of spatial direction terms', in R. J. Jarvella and W. Klein (eds.), *Speech, Place, and Action. Studies of Language in Context*, Wiley, Chichester.
- Linde, C. and W. Labov: (1975) 'Spatial networks as a site for the study of language and thought,' *Language* 51, 924-939.
- Linde, C. and J. A. Goguen: (1977) 'Structure of planning discourse,' unpublished manuscript.

- Mandler, J.: (1978) 'A code in the node: The use of a story schema in retrieval,' *Discourse Processes* 1, 14-35.
- Munro, A.: (1977) *Speech Act Understanding*, Dissertation, 1977, U.C. San Diego.
- Ullmer-Ehrich, V.: (1981) 'The structure of living space descriptions', in R. J. Jarvella and W. Klein (eds.), *Speech, Place, and Action. Studies of Language in Context*, Wiley, Chichester.
- Wunderlich, D. and R. Reinelt: (1981) 'Telling the way', in R. J. Jarvella and W. Klein (eds.), *Speech, Place, and Action. Studies of Language in Context*, Wiley, Chichester.