

# A Computational Model of Incremental Utterance Production in Task-Oriented Dialogues

Kohji Dohsaka

NTT Basic Research Laboratories  
3-1 Morinosato-wakamiya,  
Atsugi, Kanagawa, 243-01 Japan  
dohsaka@atom.br1.ntt.jp

Akira Shimazu

NTT Basic Research Laboratories  
3-1 Morinosato-wakamiya,  
Atsugi, Kanagawa, 243-01 Japan  
shimazu@atom.br1.ntt.jp

## Abstract

This paper presents a computational model of incremental utterance production in task-oriented dialogues. This model incrementally produces utterances to propose the solution of a given problem, while simultaneously solving the problem in a stepwise manner. Even when the solution has been partially determined, this model starts utterances to satisfy time constraints where pauses in mid-utterance must not exceed a certain length. The results of an analysis of discourse structure in a dialogue corpus are presented and the fine structure of discourse that contributes to the incremental strategy of utterance production is described. This model utilizes such a discourse structure to incrementally produce utterances constituting a discourse. Pragmatic constraints are exploited to guarantee the relevance of discourses, which are evaluated by an utterance simulation experiment.

## 1 Introduction

Dialogues occur in real-time and so are susceptible to time constraints. For example, dialogue participants must produce utterances under time constraints where pauses in mid-utterance must not exceed a certain length. Moreover, participants are inference-limited (Walker and Rambow 1994). Due to time constraints and limits in inference, dialogue participants cannot help producing utterances incrementally. Incremental utterance production is characterized like this: speakers produce utterances while deliberating what to say, and refine what they will say while articulating the first part of their utterances.

The incremental strategy of utterance production plays a crucial role in spoken dialogues in two respects. First, it helps speakers to satisfy time constraints on pauses. This is crucial since lengthy pauses imply the transition of a turn from the cur-

rent speaker to others. Second, it helps hearers to easily understand utterances since it enables the piecemeal transmission of information.

This paper presents a computational model of incremental utterance production in task-oriented dialogues. This model produces utterances to propose the solution of a given problem while simultaneously solving the problem in a stepwise manner. To satisfy time constraints on pauses, this model starts utterances even when the solution has not been fully determined and refines on the solution during the articulation of utterances.

We present the results of an analysis of discourse structure in a corpus of Japanese task-oriented dialogues and show that the fine structure of discourse prevails in spoken dialogues and the predominant discourse structure contributes to the incremental strategy. Based on such a discourse structure, this model incrementally produces utterances constituting a discourse. However, the incremental strategy is subject to generating irrelevant discourses. To guarantee the relevance of discourses, this model utilizes pragmatic constraints and a context model, which are evaluated in an utterance simulation experiment.

## 2 Related Research

Recent studies of human speech production (Levelt 1989) show that human speakers frequently use the incremental strategy of utterance production. This paper is concerned with a computational model of incremental utterance production.

Computational models for the incremental syntactic formulation of a sentence were proposed (Kempen and Hoenkamp 1987; De Smedt and Kempen 1991). Although incremental syntactic formulation is an important issue, we do not address this here.

POPEL is a parallel and incremental natural language generation system (Finkler and Schauder 1992; Reithinger 1992). In POPEL, the “what to say” component determines the content to be generated and gradually carries it over to the “how to say” component, which formulates a sentence incrementally. POPEL can generate dis-

- (1) aiko-ishida made desune / (2) itte/ <hai>  
 PN to COPULA go  
 (to The Aiko-ishida station) (go)
- (3) sokode basu nandesuga / êto  
 then bus COPULA FILLER  
 (then bus) (uhm)
- (4) morinosato-aoyama-iki toiu <hai> basu ga  
 PN named bus SUBJ  
 aru-node /  
 exist-CAUSAL  
 (as there is a bus named morinosato-aoyama-iki)
- (5) sore ni notte-moratte / <hai>  
 it OBJ get on  
 (get on it)

(Note: <hai> shows that the dialogue partner inserts an utterance to provide acknowledgment.)

Figure 1: Part of transcription of dialogue

courses using contextual information. However, it does not allow for the fine structure of discourse prevailing in spoken dialogues. We present a computational model of incremental utterance production using the fine structure of discourse.

Carletta, Caley, and Isard (1993) proposed an architecture for time-constrained language production. As for phenomena peculiar to spoken dialogues, they focused on hesitation and self-repair. Although our model can produce filler terms and repair prior utterances, our chief concern is the fine structure of spoken discourse, which is closely related to incremental utterance production.

### 3 Discourse Structure Analysis

We analyzed the discourse structure in a corpus of task-oriented dialogues, which were collected by the following method. The subjects were ninety native Japanese. In each dialogue, two subjects, *N* and *E*, were asked to converse by telephone to find a solution to the problem of how *N* could get from one place to another. Subjects were chosen such that *E* had enough knowledge to solve the problem but *N* did not. Fifty dialogues were recorded and transcribed. Fifteen dialogues were randomly chosen for analysis. The discourse structure was analyzed in terms of information units and discourse relations.

#### 3.1 Analyzing information units

Speakers organize the information to be conveyed to information units (Halliday 1994), which are the units for transmission of information. The information units (*IUs* for short) are regarded as minimal components of discourse structure. We assume that *IUs* are realized by grammatical devices: a clause realizes an *IU*, an interjectory word realizes an *IU*, and a filler term shows the end of an *IU*. Figure 1 shows part of the transcription of a dialogue where a dialogue participant proposes a domain plan. The symbol “/” separates the *IUs*.

Table 1: Frequency distribution for information units

Clause	929
Interjectory word	665
PP or NP	279
Conjunction	84
Sequence of PPs or NPs	41
Others	14
Total	2012

Table 1 shows the frequency distribution for the grammatical categories of *IU*, where NP and PP mean noun phrase and postpositional phrase. The average number of NPs in an *IU* as a clause, NP, PP, or sequence of NPs and PPs is 1.01 in the fifteen dialogues. The variance is 0.28.

This result indicates that small *IUs* are frequently used. For example, although *IU* (1) in Figure 1 describes only a part of a domain action, it is regarded as an *IU* since it has a copula (“desu”) and a sentence-final particle (“ne”).

#### 3.2 Analyzing discourse relations

Discourse relations between adjacent discourse segments were examined. A discourse segment is an *IU* or a sequence of *IUs*. For discourse relations, we here adopted those used in Rhetorical Structure Theory (Mann and Thompson 1988) and here followed Hovy (1993) to classify them into semantic and interpersonal ones. Figure 2 shows discourse relations that appear in the discourse displayed in Figure 1. The small *IUs* are hierarchically related. This results in the fine structure of discourse.

Table 2 shows the frequency distributions for discourse relations in the fifteen dialogues. Let us consider the role that the predominant relations, Elaboration, Circumstance, and Motivation, play in the incremental strategy of utterance production.<sup>1</sup> First, Elaboration is exploited to describe domain actions, states or objects in a piecemeal fashion. Elaboration enables speakers to distribute the content to be conveyed among different *IUs*. This relation is useful for the incremental strategy since it allows speakers to begin uttering even when the content has not been fully determined.

Second, Circumstance is the relation between two segments, a nucleus and a satellite. The nucleus describes a domain action or state. The satellite describes the circumstances where the nucleus is interpreted, such as the preconditions of a domain action. There are 41 cases where the satellite describes a precondition of a domain action, which amounts to 68% of all cases. The constituents of a domain action are often referred to in its preconditions. We see a typical case in the relation between (4) and (5) in Figure 1. (5) describes the action of getting on a bus and (4) de-

<sup>1</sup>We found no direct relationship between Sequence and the incremental strategy.

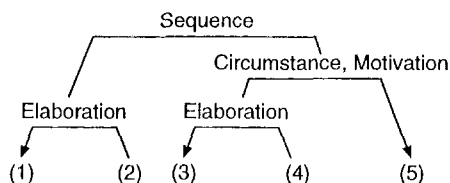


Figure 2: Discourse relations in Figure 1

Table 2: Distribution for discourse relations

Elaboration	305	Motivation	66
Sequence	74	Background	14
Circumstance	60	Evidence	10
Result	25	Interpretation	6
Condition	25	Concession	3
Purpose	2	Enablement	1
Contrast	1		

(a) Semantic relations

(b) Interpersonal relations

scribes the existential status of the bus as the precondition of the action. By utilizing this relation, speakers can distribute the content of a domain action between two *IUs*. They can pick up a constituent of an action and describe it before describing the whole content of the action. Thus Circumstance is useful for the incremental strategy.

Finally, Motivation is mainly used for describing a domain action as a nucleus and motivating addressees to adopt the action by presenting a fact as a satellite. In typical cases, speakers motivate addressees to adopt an action by asserting that its precondition is satisfied. In such cases, Motivation occurs together with Circumstance and contributes to the incremental strategy in the same way as Circumstance.

## 4 The Model

As shown in Figure 3, this model is composed of five modules: a problem solver, an utterance planner, an utterance controller, a text-to-speech converter, and a pause monitor. The problem solver makes domain plans that solve a given problem. The utterance planner makes utterance plans to propose domain plans. Pragmatic constraints and a context model are used to generate relevant discourses. According to utterance plans, the utterance controller sends linguistic expressions to the text-to-speech converter. The pause monitor watches the length of pauses and signals the utterance planner and controller when the pause length exceeds a given length.

These modules work in parallel. Both domain plans and utterance plans are made in a stepwise manner using the hierarchical planning mechanism (Russel and Norvig 1995: Chap.12). This model starts to make an utterance plan before a fully determined domain plan has been obtained. When a pause exceeds the time limit, the utterance planner sends the utterance controller an ut-

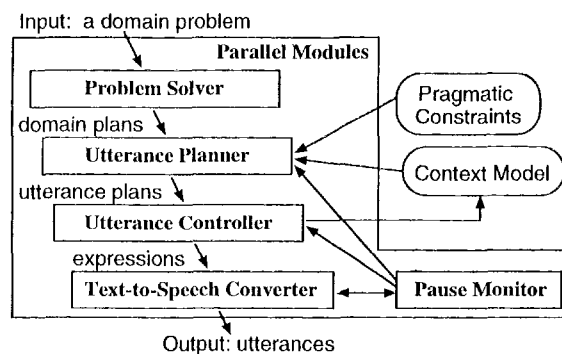


Figure 3: Model overview

terance plan obtained within the time limit. A domain plan is refined during the planning and articulation of utterances. Based on a refined domain plan, the utterance plan is replanned. When the utterance controller is not given utterance plans within the time limit, it produces a filler term.

## 5 Pragmatic Constraints

Pragmatic constraints are required to guarantee the relevance of discourses. This model exploits the following pragmatic constraints.

- (c1) Avoid conveying redundant information.
- (c2) Pronominalize objects in the focus of attention (Grosz and Sidner 1986).
- (c3) Be relevant according to the attentional state.

The context model records the information that has been conveyed and tracks the attentional state. For example, consider the domain action of moving from one location  $l_1$  to another  $l_2$ . To describe such a domain action with verbs such as “iku(go)”,  $l_1$  must be in focus. Otherwise, the description is irrelevant. After such an action has been described,  $l_2$  is in the focus. Moreover, any object marked as a topic becomes a focused one.

## 6 Problem Solving

We outline the problem solver using a sample problem of how to move from the Musashino Center to the Atsugi Center on the map in Figure 4.

The problem solver first makes an abstract domain plan, which is a sequence of three actions  $a1$ ,  $a2$ , and  $a3$ : moving from the Musashino Center to the nearest station by bus, moving to the station nearest the Atsugi Center, and then moving to the Atsugi Center by bus. This plan is written as (r1). The contents of these actions are written as (r2). Expression  $cont(X, Y)$  means that the content of  $X$  is represented as a set  $Y$  of literals.

- (r1)  $plan(\{a1, a2, a3\})$
- (r2)  $cont(a1, \{type(a1, move), source(a1, x1), manner(a1, x2), dest(a1, x3)\})$   
 $cont(x1, \{type(x1, building), named(x1, "musashino sentaa")\})$   
 $cont(x2, \{type(x2, bus)\})$

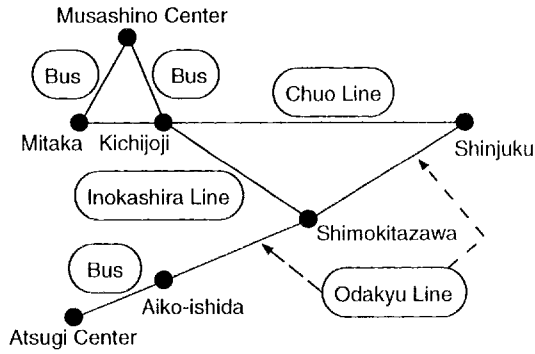


Figure 4: Sample map

```

cont(x3, {type(x3, station), nearest(x3, x1)})
cont(a2, {type(a2, move), source(a2, x3),
         dest(a2, x4)})
cont(a3, {type(a3, move), source(a3, x4),
         manner(a3, x5), dest(a3, x6)})
cont(x4, {type(x4, station), nearest(x4, x6)})
cont(x5, {type(x5, bus)})
cont(x6, {type(x6, building),
         named(x6, "atsugi sentaa")})

```

The problem solver tries to make a more concrete plan. When more than one domain plan is possible, it chooses the domain plan that requires the shortest execution time. In this domain, the domain plan is a sequence of actions  $a4$ ,  $a5$ ,  $a6$  and  $a7$ : moving from the Musashino Center to Kichijoji station by bus, moving to Shimokitazawa station by the Inokashira Line, moving to Aiko-ishida station by the Odakyu Line, and then moving to the Atsugi Center by bus. Part of the content of this plan is represented as follows.

```

(r3) plan([a4, a5, a6, a7])
(r4) cont(a4, {type(a4, move), source(a4, x1),
             manner(a4, x2), dest(a4, x7)})
      cont(x7, {type(x7, station),
             named(x7, "kichijoji")}) .....

```

## 7 Utterance Planning

An utterance plan is a sequence of communicative actions that achieves a communicative goal. It is refined in a stepwise manner. A sequence of surface communicative actions corresponding to the uttering of linguistic expressions is finally planned.

### 7.1 Communicative goals

Generation systems engaging in dialogues must record communicative goals related to communicative actions (Moore and Paris 1994). Communicative goals used here are:

- *persuaded-plan*( $P$ ): dialogue partner is persuaded to adopt domain plan  $P$ .
- *persuaded-act*( $A$ ): dialogue partner is persuaded to adopt domain action  $A$ .

- *described-event*( $E, C, At$ ): domain event  $E$  is described as an event having content  $C$  and attitude  $At$  toward  $E$  is also described.
- *described-obj*( $O, C$ ): domain object  $O$  is described as an object having content  $C$ .
- *described-thema-rel*( $R, O, E$ ): thematic relation  $R$  is described, which domain object  $O$  bears to domain event  $E$ .

When the domain plan (r1) is obtained, (r5) is given as the initial communicative goal.

```
(r5) persuaded-plan([a1, a2, a3])
```

### 7.2 Surface communicative actions

Surface communicative actions used here are:

- *surface-desc-event*( $E, C, At$ ): utter expressions to describe domain event  $E$  as an event having content  $C$  and describe attitude  $At$  toward  $E$ .
- *surface-desc-obj*( $O, C, R$ ): utter expressions to describe domain object  $O$  as an object having content  $C$  and bearing thematic relation  $R$  to a certain event.

### 7.3 Planning utterances based on the fine structure of discourse

An utterance plan is elaborated using action schemata and decomposition methods. An action schema consists of an action description, applicability constraints and an effect.<sup>2</sup> It defines a communicative action. A decomposition method consists of an action description, applicability constraints and a plan. It specifies how an action is decomposed to a detailed plan.

The following schema (r6) defines the communicative action of proposing a domain plan by using Sequence. The decomposition method (r7) specifies how the action is decomposed to a sequence of finer actions.<sup>3</sup>

```

(r6) Act(propose-acts-in-seq(*P),
        Constr: plan(*P),
        Effect: persuaded-plan(*P))
(r7) Decomp(propose-acts-in-seq[*Act | *Rest]),
        Constr: *Rest ≠ [],
        Plan: [achieve(persuaded-act(*Act)),
              propose-acts-in-seq(*Rest)])

```

In these representations, *achieve*( $P$ ) designates an action that achieves goal  $P$ . Notation [ $H$  |  $L$ ] specifies a list, where  $H$  is the head of the list and  $L$  is the rest. Symbols starting with “\*” represent variables. By applying (r6) and (r7) to the initial communicative goal (r5), the following utterance plan is obtained:

```

(r8) achieve(persuaded-act(a1)),
      achieve(persuaded-act(a2)),
      achieve(persuaded-act(a3)).

```

<sup>2</sup>In this paper, we do not consider a precondition for an action schema.

<sup>3</sup>We have omitted other method to avoid infinite recursive application of the method (r7).

- (r9) Act(*propose-act*(\*A), Effect: *persuaded-act*(\*A))  
 (r10) Decomp(*propose-act*(\*A), Constr: *cont*(\*A, \*C), Plan: *achieve*(*described-event*(\*A, \*C, *proposal*))  
 (r11) Act(*describe-event-by-elaboration*(\*E, \*C, \*At), Effect: *described-event*(\*E, \*C, \*At))  
 (r12) Decomp(*describe-event-by-elaboration*(\*E, \*C, \*At), Constr: \*Thema  $\in$  \*C  $\wedge$   
 \*Thema =.. [\*R, \*E, \*O]  $\wedge$  \*R  $\neq$  type  $\wedge$  *cont*(\*O, \*ObjC)  $\wedge$  \*Rest = \*C - {\*Thema}  
 Plan: [*achieve*(*described-obj*(\*O, \*ObjC)), *achieve*(*described-thema-rel*(\*R, \*O, \*E)),  
*achieve*(*described-event*(\*E, \*Rest, \*At))])  
 (r13) Act(*describe-obj-with-thema*(\*O, \*C, \*R, \*E),  
 Effect: *described-obj*(\*O, \*C)  $\wedge$  *described-thema-rel*(\*R, \*O, \*E))  
 (r14) Decomp(*describe-obj-with-thema*(\*O, \*C, \*R, \*E), Plan: *surface-desc-obj*(\*O, \*C, \*R))  
 (r15) Act(*describe-event-type*(\*E, \*C, \*At), Constr: \*C = {*type*(\*E, \*T)},  
 Effect: *described-event*(\*E, \*C, \*At))  
 (r16) Decomp(*describe-event-type*(\*E, \*C, \*At), Plan: *surface-desc-event*(\*E, \*C, \*At))

Figure 5: Action schemata and decomposition methods for proposing domain action

(r8) is decomposed by applying the action schemata and decomposition methods shown in Figure 5. These schemata define communicative actions for proposing a domain action while elaborating the content of the action in a stepwise manner. They reflect the results of a discourse structure analysis, which show that speakers tend to distribute the constituents of a domain action into different IUs by using ELABORATION. In (r12), notation  $F(X, Y, \dots) =.. [F, X, Y, \dots]$  is used for decomposing term  $F(X, Y, \dots)$  into relation  $F$  and arguments  $X, Y, \dots$ .

When domain objects are linguistically realized by the *surface-desc-obj* in (r14), pragmatic constraint (c2) is exploited to pronominalize focused objects. In addition, according to constraint (c3), the objects that are not in focus need to be topicalized if they must be in focus.

By applying these schemata to the first action in (r8), the following utterance plan is obtained. Thematic relations are chosen in default order when (r12) is applied.

- (r17) *surface-desc-obj*(x1, {*type*(x1, building),  
 named(x1, "musashino sentaa")}, source),  
*surface-desc-obj*(x2, {*type*(x2, bus)}, manner),  
*surface-desc-obj*(x3, {*type*(x3, station),  
 nearest(x3, x1)}, dest),  
*surface-desc-event*(a1, {*type*(a1, move)},  
 proposal).

According to utterance plan (r17), this model can start the following utterances to satisfy the time constraints before obtaining a concrete domain plan such as (r3).

- (u1) *musashino sentaa kara-wa desune/*  
 PN from-TOPIC COPULA  
 (from the Musashino Center)  
*basu de/ moyori-no-eki made/ ikimasu/*  
 bus by nearest station to go  
 (by bus) (to the nearest station) (go)

For brevity, we have omitted action schemata and decomposition methods for utterance planning using MOTIVATION and CIRCUMSTANCE.

#### 7.4 Replanning utterance plans

While planning and articulating utterances using an abstract domain plan, a more concrete domain plan is being made. When a more concrete domain plan is obtained, an utterance plan is replanned. For example, consider the case where a concrete domain plan, (r3), is obtained during the production of utterance (u1). The following utterance plan is replanned:

- (r18) *surface-desc-obj*(x1, {*type*(x1, building),  
 named(x1, "musashino sentaa")}, source),  
*surface-desc-obj*(x2, {*type*(x2, bus)}, manner),  
*surface-desc-obj*(x7, {*type*(x7, station),  
 named(x7, "kichijoji")}, dest),  
*surface-desc-event*(a4, {*type*(a4, move)},  
 proposal).

We assume that plan (r18) is obtained when this model finishes uttering "moyori-no-eki made" in utterance (u1). Then (u1) is interrupted and utterances follow based on (r18). Consequently, the following utterances are produced:

- (u2) *musashino sentaa kara-wa desune/*  
 PN from-TOPIC COPULA  
 (from the Musashino Center)  
*basu de/ moyori-no-eki made/*  
 bus by nearest station to  
 (by bus) (to the nearest station)  
*kichijoji made desune / ikimasu /*  
 PN to COPULA go  
 (to Kichijoji station) (go)

In the above, the redundant information is not restated according to pragmatic constraint (c1). Self-repair occurs: "moyori-no-eki made" is replaced by "kichijoji made".

## 8 Experiments

This model has been implemented in Common Lisp. A logical constraint unification system (Nakano 1991) is used in the planners. The domain planner includes 18 action schemata and 16 decomposition methods. The utterance plan-

- (e1) Musashino sentaa kara-wa desune / basu de / moyori-no-eki made /  
 PN from-TOPIC COPULA bus by nearest station to  
 (from the Musashino Center) (by bus) (to the nearest station)
- (e2) êto Kichijoji made / dete-kudasai /  
 FILLER PN to go-please  
 (erm to Kichijoji station) (please go)
- (e3) êto desune sorekara inokashira-sen de desune / shimokitazawa made / itte /  
 FILLER then PN by COPULA PN to go  
 (erm then by the Inokashira Line) (to Shimokitazawa station) (go)
- (e4) odakyu-sen ni / norikaete / aiko-ishida made / ikimasu / .....  
 PN for change PN to go  
 (change train for the Odakyu Line) (to Aiko-ishida station) (go)

Figure 6: Discourse generated by implemented system

ner includes 16 action schemata and 16 decomposition methods. We evaluated pragmatic constraints in an utterance simulation experiment, where discourses generated with the constraints were compared with those generated without them. A map including 120 locations such as station and 25 railroad lines was used. The pause length limit was 0.5 sec.

When pragmatic constraints were used, this implemented system generated relevant discourses. Figure 6 shows the discourse generated when the problem of moving from the Musashino Center to the Atsugi Center was given. Filler terms such as *êto* were produced to satisfy the time constraints. Pragmatic constraint (c1) was used in (e2), as explained in section 7.4. Constraint (c2) was used to zero-pronominalize stations in the focus of attention. Constraint (c3) was used in (e1) to topicalize the Musashino Center. Topicalization was also used in other cases where the system must shift the focus of attention to the location already described in the preceding discourse. Such cases happened when the system started utterances based on an abstract domain plan, took a long time to obtain a more concrete plan, and then elaborated on a route from a location that was not in focus based on the concrete plan. Without pragmatic constraints, the system generated irrelevant and excessively redundant discourses.

## 9 Conclusion

We presented a computational model of producing utterances incrementally so as not to make excessively long pauses. We presented the results of an analysis of discourse structure and showed that speakers frequently use small information units and exploit the fine structure of discourse that contributes to the incremental production strategy. This model can utilize such a discourse structure to incrementally produce utterances according to pragmatic constraints. These were evaluated by an utterance simulation experiment.

## References

Carletta, Jean; Caley, Richard; and Isard, Stephen.

- (1993). A system architecture for simulating time-constrained language production. Research Paper HCRC/RP-43, University of Edinburgh.
- De Smedt, Koenraad, and Kempen, Gerard. (1991). Segment grammar: A formalism for incremental sentence generation. In *Natural Language Generation in Artificial Intelligence and Computational Linguistics*. Edited by Cécile L. Paris, William R. Swartout, and William C. Mann, Kluwer Academic Publishers, 329-349.
- Finkler, Wolfgang, and Schauder, Ann. (1992). Effects of incremental output on incremental natural language generation. In Proc. of 10th ECAI, 505-507.
- Grosz, Barbara J., and Sidner, Candace L. (1986). Attention, intentions, and the structure of discourse. *Computational Linguistics*, 12, 175-204.
- Halliday, M. A. K. (1994). *An Introduction to Functional Grammar*. Edward Arnold.
- Hovy, Eduard H. (1993). Automated discourse generation using discourse structure relations. *Artificial Intelligence*, 63, 341-385.
- Kempen, Gerard, and Hoenkamp, Edward. (1987). An incremental procedural grammar for sentence formulation. *Cognitive Science*, 11, 201-258.
- Levelt, Willem J. M. (1989). *Speaking: From Intentions to Articulation*. The MIT Press.
- Mann, William C., and Thompson, Sandra A. (1988). Rhetorical structure theory: Towards a functional theory of text organization. *Text*, 8(3), 243-281.
- Moore, Johanna D., and Paris, Cécile L. (1994). Planning text for advisory dialogues: capturing intentional and rhetorical information. *Computational Linguistics*, 19(4), 651-694.
- Nakano, Miki. (1991). Constraint projection: An efficient treatment of disjunctive feature descriptions. In *Proc. of 29th ACL*, 307-314.
- Reithinger, Norbert. (1992). The performance of an incremental generation for multi-modal dialog contributions. In *Aspects of Automated Natural Language Generation*. Edited by Robert Dale, Eduard Hovy, Dietmar Rösner, and Oliviero Stock, Springer-Verlag, 263-276.
- Russel, Stuart, and Norvig, Peter. (1995). *Artificial Intelligence: A Modern Approach*. Prentice Hall.
- Walker, Marilyn A., and Rambow, Owen. (1994). The role of cognitive modeling in achieving communicative intentions. In *Proc. of 7th International Conference on Natural Language Generation*.