Elements of Speech Interaction

Niels Ole Bernsen*, Hans Dybkjær** and Laila Dybkjær*

*The Maersk Mc-Kinney Moller
Institute for Production Technology
Odense University, Campusvej 55,
5230 Odense M, Denmark
nob@mip.ou.dk, laila@mip.ou.dk
phone: (+45) 65 57 35 44
fax: (+45) 66 15 76 97

**Prolog Development Center A/S
H. J. Holst Vej 3-5A,
2605 Brøndby, Denmark
dybkjaer@pdc.dk
phone: (+45) 36 72 10 22
fax: (+45) 36 72 02 69

Abstract

With the spreading of interactive speech systems technologies, a clear need arises for theory which may adequately support the development and evaluation of increasingly sophisticated but still restricted interactive speech systems. This paper presents steps towards a practical bottom-up theory of spoken human-computer interaction. The theory provides a set of interaction elements and takes the form of an incremental task-oriented interaction theory which attempts to anticipate some of the problems to be addressed in developing successive systems generations.

1 Introduction

With the spreading of interactive speech systems technologies, a clear need arises for theory which may adequately support the development of increasingly sophisticated but still restricted interactive speech systems. A complete and applied theory of spoken human-machine interaction would rigorously support efficient interactive speech systems development from initial requirements capture through to the test and maintenance phases. It would include support for interaction model development and implementation, appropriate functionality design, usability optimisation, interactive speech systems evaluation and maintenance. Above all, such a theory would have to be based on the fact that the interaction models of today's interactive speech systems are all *task-oriented*, they enable the system to carry out spoken interaction with users in limited application domains [Smith and Hipp 1994]. When combined with a basic level of meta-communication, or communication about the interaction itself, task-orientation is what enables current systems to successfully undertake spoken dialogue with humans despite their many limitations compared to human interlocutors. These comparative limitations may be briefly illustrated by taking a look at spoken human-human communication.

As humans we learn to perform spoken interaction fluently, effortlessly and efficiently about almost any topic and for almost any purpose. Human-human conversation serves both to organise social life in general and as the basis for more specific types of interaction, such as getting others to do something, obtaining information from them, or solving problems together cooperatively. The ability to perform human-human-quality conversation requires a large number of skills and other characteristics, such as recognition of complex spontaneous speech, a very large vocabulary, reference resolution, referential capabilities building on world knowledge, use of meta-communication when appropriate, and unrestricted language and speech generation. Spoken human-computer interaction, on the other hand, is constrained by the conversational limitations of the computer and rarely has any social function—at least not for the computer. The constraints imply that interaction models for interactive speech systems have to be carefully crafted in order to work at all, even within limited domains. In interactive speech systems development, usability considerations are not a luxury but a dire need. This is one more reason for developing interactive speech theory.

Most spoken or written language interaction theory has so far dealt with unrestricted human-human conversation and has not clearly focused on task-oriented dialogue. While no single, unified interaction theory has yet emerged from the various frameworks and approaches that have been proposed in the literature, parts of these theories and the aspects of dialogue they cover are potentially relevant to the more limited purpose of establishing a task-oriented theory of spoken human-computer interaction. This is true of speech acts theory [Searle 1969], Gricean theory of cooperativity in dialogue [Grice 1975], discourse representation theory [Kamp and Reyle 1993], plan-based approaches to dialogue [Litman 1985, Carberry 1990], Grosz and Sidner's intentional approach [Grosz and Sidner 1986, Grosz et al. 1989], relevance theory [Sperber and Wilson 1987] and rhetorical structure theory [Mann and Thompson 1987a, 1987b], among others. However, a theory of spoken interaction in support of interactive speech systems development and evaluation cannot simply transfer results from interaction theories which deal with unrestricted human-human dialogue. Instead, it is necessary to define the level of interaction which current interactive speech systems are capable of, in order to be able to:

- precisely characterise each individual system including its limitations;
- precisely characterise similarities and differences between current systems;
- support the design and implementation of interactive speech systems;
- define the needs for research and technological development which might help to incrementally improve the capabilities of current interactive speech systems; and
- facilitate the transfer of relevant results from human-human interaction theories.

A theory with these properties may be characterised as a *practical, bottom-up theory of interactive speech systems.* It does not primarily synthesise the existing, often fragile and sometimes conflicting results from spoken human-human interaction theories nor does it primarily aim at specifying the properties of the ideal interactive speech system which we shall not be able to build in the foreseeable future anyway. Rather, the theory departs from the properties of current, comparatively simple interactive speech systems; aims to make sure that these have been understood before proceeding towards more complex systems; incorporates results from existing human-human interaction theory only when relevant to the technological state of the art; and creates the elements of theory needed to support the design of high-level interaction models for specific interactive speech systems.

This paper presents steps towards a practical bottom-up theory of spoken human-computer interaction. The theory provides a set of interaction elements and takes the form of an incremental task-oriented interaction theory which attempts to anticipate some of the problems to be addressed in developing successive systems generations. Incrementality means that novel interaction elements can be added without the rest of the theory necessarily having to be revised.

Section 2 presents a model of the elements of the theory and explains and illustrates these in a walk-through of a spoken human-computer dialogue. Section 3 demonstrates how the theory may be used in characterising interactive speech systems. Section 4 concludes the paper. A more detailed description of the theory can be found in [Bernsen et al. 1998].

2 Elements of interactive speech theory

The goal of interactive speech theory development is to describe the structure, contents and dynamics of spoken human-computer interaction from the point of view of the interactive speech system. On the one hand, users should have a pleasant and efficient conversation, on the other, the theory should have good computational properties and support systems development.

The theory to be presented is far from complete. It is, rather, an organised conceptual toolbox of elements at least some of which need to be taken into consideration when developing today's interactive speech systems. We are aware that the elements and their organisation may be disputed on many points. There simply is no complete, general and accepted theory yet, and even a structured conceptual toolbox is bound to suffer from not-fully-analysed relationships between the elements and types of element it proposes. Conceivably, satisfactory analysis will have to wait until the problem space posed by interactive speech systems development has been explored in much more depth than is currently the case.

Still, there is emerging consensus on several issues, and a number of concepts and techniques have proved useful to the building of interactive speech systems. Figure 1 shows a model of the elements of an interactive speech theory. The elements all appear important and sometimes necessary to the design and construction of interactive speech systems. The model is software-oriented, focusing on the objects or *elements* that go into the system. Hardware, including telephones and microphones, is not included and the same holds for the user's physical work environment. From the point of view of the model, these aspects belong to the many other constraints that have to be taken into account during interactive speech systems specification. In explaining the model below, we shall focus on the elements that are most relevant to the dialogue component and, more generally, to the interaction model of interactive speech systems.

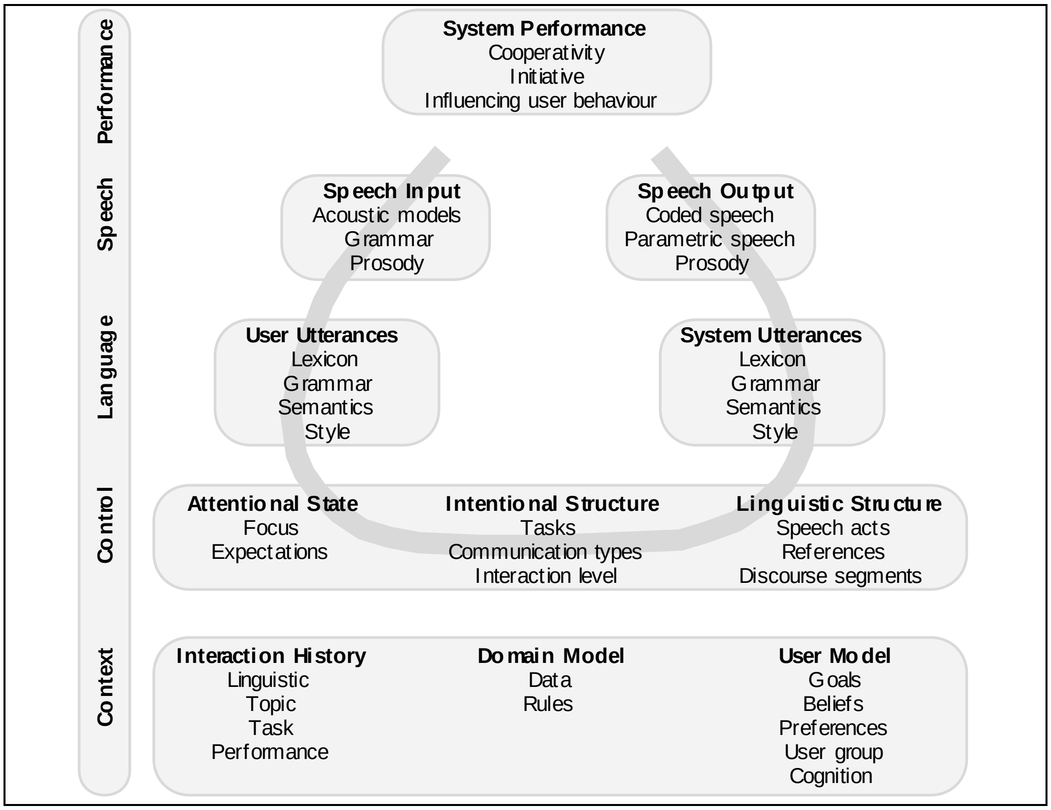

Figure 1: Elements of an interactive speech theory. Element types are shown in boldface. The grey band and grey boxes reflect the logical architecture of interactive speech systems.

The elements of Figure 1 may be used to construct high-level models of interactive speech systems and explain their behaviour. We shall refer to the model in Figure 1 as the *basic speech interaction model.* The model exhibits two modes of organisation:

First, the elements have been organised into five *layers*. At the bottom of the figure, the *context* layer includes aspects of the history of interaction, domain model and user model. At the level above the context layer, the interaction *control* layer includes states of attention as well as the structures defined by the interlocutors' intentions and structural aspects of the linguistic exchanges. System control is largely based on structures at this level. The following, *language* layer describes the linguistic aspects of interaction**.** Then follows the *speech* layer which includes the transformations between speech signals and the symbolic expressions of language. Finally, the *performance* layer is a function of the other layers taken together and includes some general aspects of the system's behaviour.

Secondly, the grey band in Figure 1 indicates the overall processing *flow* among the various types of element—input, intention and attention, output and performance—in a context defined by contextual elements. Developers often refer to elements or element types in terms of the corresponding system modules, such as the recogniser, parser, dialogue manager, inference engine, text generator and player, system performance being replaced by an abstraction of the (physical) user.

It should be noted that some elements, such as lexicon size, user background and cooperativity, are in focus only at design time whereas other elements, such as linguistic structure, interaction history and user goals are run-time constructs which are used dynamically by the system. In the present paper, interactive speech theory will be presented primarily from an operational (or implementational) viewpoint. In Section 3 the theory will be used from a functional viewpoint as a vehicle for characterising interactive speech systems.

Figure 1 reflects a number of interactive speech systems analyses and components from theories of spoken human-human interaction, including [Carberry 1990, Figure 1.1; Eckert and McGlashan 1993, Figures 1 and 2; Smith and Hipp 1994, Figure 3.3; Grau et al. 1994, Figure 1; Jönsson 1993, Figures 7.1 and 7.2; Zue 1994; Aust et al. 1995; Grosz and Sidner 1986; Bunt 1994]. Given these origins, it is clear that the concepts used in the model have been drawn from widely different disciplines, such as linguistics, computer science and cognitive science.

In the following sub-sections we explain and exemplify the elements of Figure 1. Throughout these sub-sections we refer to a human-computer dialogue from the user test of the Danish Dialogue System (Figure 2) for exemplification.

2.1 Performance

Any advanced interactive speech system has many of the elements in Figure 1 but no current system has them all. Together, the elements determine the observable behaviour or *performance* of the system during interaction. The system's performance itself has a number of more or less complex properties that emerge from the nature of the elements and which should be considered during development. We discuss these interdependent properties in terms of the performance elements cooperativity, initiative and the system's influence on user behaviour.

Cooperativity: Habitable user-system interaction requires that both user and system behaviour be *cooperative.* We believe that system cooperativity is crucial to successful interaction model development: it contributes to smooth interaction and reduces the need for meta-communication.

Example: The interaction model of the Danish Dialogue System is task-oriented. It assumes that user and system have a common task, that is, to make flight ticket reservations, and that the aim of user-system interaction is to perform this task in as rational a manner as possible. In support of this, system performance should be cooperative, i.e. the system should enable the interaction to proceed as efficiently as possible towards the common goal. We leave it to the reader to judge the system's cooperative behaviour in Figure 2.

Initiative. The interlocutor who determines the current topic of the interaction is said to have the *initiative* or to control the course of the interaction. It is useful to distinguish between the following modes of interaction from the point of view of who has the initiative. An interactive speech system is called *system directed* if the system has the initiative throughout the interaction; *user directed* if the user has the initiative throughout; and *mixed initiative* if both (or all) interlocutors may take the initiative at some or all points during interaction. These modes of interaction may all be found in today's interactive speech systems except, perhaps, the "free" variety of mixed initiative interaction in which any interlocutor may take the initiative at any time.

Example: It is clear from Figure 2 that the Danish Dialogue System has the overall initiative in the dialogue as shown in (S2b, S3b, S4b, …). The system preserves the initiative through concluding all its turns by a closed question to the user, i.e. a question which asks for a well-defined piece of information, such as in (S2b). However, the user may take the initiative in meta-communication through use of keywords. This sometimes works, as in (U5a), and sometimes this does not work, as in (U4a), cf. on linguistic structure and intentional structure below.

S1a	Hello, this is the DanLuft reservation system.
b	Do you know this system?
U1a	Nope, I don’t. No.
S2a	The system can reserve tickets for Danish domestic flights. You use it by answering the system’s questions. In addition you may use the two special commands *repeat* and *change* to have the most recent information repeated or corrected. The system will only understand you when you answer its questions briefly and one at a time.
b	Please state your customer number.
U2a	The customer number is 4.
S3a	Customer number 4.
b	How many persons will travel?
U3a	Two persons, one adult and one child.
S4a	One person.
b	Please state the id-number of the person.
U4a	It was not one person, it was two persons, one adult and one child.
S5a	Id-number 1, Jens Hansen.
b	Where does the journey start?
U5a	Change.
S6a	First person, id-number 1.
b	Is that correct?
U6a	No, id-number 4.
S7a	Id-number 4, Marie Hansen.
b	Where does the journey start?
U7a	From Copenhagen.
S8a	From Copenhagen.
b	Where does the journey end?
U8a	Karup.
S9a	To Karup.
b	Do you want a return ticket?
U9a	No thanks.
S10a	On which date will the journey start?
U10a	The 11th of February.
S11a	Saturday the 11th of February.
b	At which time?
U11a	Around noon.
S12a	Sorry.
b	I did not understand.
U12a	Around noon, 11:50.
S13a	11:50.
b	You have now booked one ticket for id-number 4, Marie Hansen, from Copenhagen to Karup on Saturday the 11th of February at 11:50. Your reference number is 49.
c	The total price is 680 kroner.
d	The ticket will be available at check-in desk number 19 in the airport. Tickets must be picked up at the latest 20 minutes before departure.
e	Do you want more?
U13a	Yes, please. I would like to book a similar ticket for one person.
S14a	Customer number 4.
b	Is that correct?
U14a	Yes.
S15a	How many persons will travel?
U15a	One person.
•••	(*The user now books a ticket for the second person).*

Figure 2: A dialogue from the user test of the Danish Dialogue System (translated from the Danish). The user interacts with the system for the first time. Turn numbers refer to the transcription of the dialogue with subject (or user) number 13 [Dybkjær et al. 1996]. For ease of reference the turns are sub-divided into utterance units. For instance, “S1” is the first system turn which consists of two utterance units (S1a) and (S1b), and the user’s answer is turn “U1” which consists of one utterance unit (U1a).

Influencing user behaviour. By contrast with the system and its behaviour, users are system-external factors that cannot be controlled directly. The fact is, however, that interactive speech systems are vastly inferior to ordinary humans as communication partners. If users do not realise this, they may have unnecessary difficulty completing their interactive task with the system. Somehow, therefore, a reasonably adequate model of how to interact with the system must be communicated to users. At least the following sources may help users build a reasonable user interaction model:

- Explicit system instructions provided in the system's introduction (cf. S2a in Figure 2) or elsewhere during the interaction.
- Explicit developer instructions, i.e. all sorts of (system-) external information provided to users prior to use of the system, e.g. scenarios or a leaflet on the use of the system.
- Implicit system "instructions".

The latter "instructions" build on the fact that speakers adapt their behaviour to the observed properties of the listener. Some of these "instructions" are provided through the systems vocabulary, grammar and style. Moreover, it appears that people tend to use less sophisticated spoken language when they believe that they communicate with a computer system rather than a human being [Amalberti et al. 1993]. Finally, of course, the system's repair and clarification meta-communication will affect the user interaction model by making some of the system's recognition and understanding difficulties clear to users. However, strong meta-communication facilities do not yet exist in interactive speech systems.

Example: The scenario-based dialogue in Figure 2 shows some cases in which the system's choice of terms probably influenced the user's own choices, such as 'persons' in (S3b) and (U3a) and 'person' in (S4a-b) and (U4a). The system persistently seeks to influence the user's linguistic behaviour through using words that belong to its input *lexicon.* In addition, the user's correct use of 'change' in (U5a) is clearly based on the system's instruction in (S2a).

2.2 Speech

The speech layer concerns the relationship between the acoustic speech signal and a, possibly enriched, text (lexical string). The relationship is not simple. Speech includes a number of prosodic phenomena—such as stress, glottal stops, and intonation—that are only reflected in text in a simplistic manner. Conversely, words and their different spellings as we know them from text, do not have natural expressions in speech.

Speech recognition must cater for extra-linguistic noise and other phenomena, such as that the speech rate varies over time, the speech signal is mixed with environmental noise from other people speaking, traffic and slamming doors, the pronunciation varies with the speaker, and speech from different participants may overlap, for instance with the system's utterances [Waibel 1996, Baggia et al. 1994].

The speech layer includes two types of elements: speech input and speech output.

Speech input

The input to the interactive speech system is an acoustic signal which typically represents a spoken utterance. The transformation of the acoustic signal into some lexical representation, such as a word sequence or lattice, is called *speech recognition*. Basically, speech recognition is a mapping process in which the incoming acoustic signal is mapped onto the system's repertoire of *acoustic models,* yielding one or several best matches which are passed on to linguistic processing. The dominant speech recognition technology uses *hidden Markov models* combined with a dynamic programming technique [Bahl et al. 1983, Rabiner 1988, Kamp 1992]. The acoustic models may represent, for instance, triphones (context-dependent phonemes), phonemes, word forms or entire phrases.

Current speech recognition techniques are typically limited to the extraction of lexical references, excluding information on pauses, stress and *prosody* in general. However, the Verbmobil system uses stress and pauses to support, e.g., semantic disambiguation.

The recognition may assume *isolated words* (words spoken one at a time, clearly separated by pauses), *connected words* (words pronounced as isolated words, but with less stress and no, or little, separation) or *continuous speech* (standard naturally spoken language with contracted words and no separation of words) as input. When accepting connected words and continuous speech, the recogniser has some simple *syntactic model* (or *grammar)* of utterances. Typical examples are *bigrams* (allowed word pairs) and *finite transition network grammars*. The amount of syntactic constraints to impose is a trade-off: syntactic constraints increase the likelihood that input conforming to the model is recognised correctly, but highly constraining syntactic models allow fewer user utterances to be recognised.

Example: An effect of the way the Danish Dialogue System's speech recogniser works can be seen in (U4a-S5a). The speech recogniser expects the user to either provide an id-number (cf. S4b), that is, a number, or to say 'change' or 'repeat' (meta-communication keywords). The recogniser misrecognises (U4a). The actual words used are not among its active *acoustic models* and the grammatical constructs are neither in the active nor in the passive part of its *grammar*. The misrecognised word string, however, still contains three of the four numbers provided but the parser only selects the final one of these, thus making its own contribution to the misunderstanding. The system's speech recogniser is not sensitive to *prosody*.

Speech output

Computer speech is produced by generating an acoustic speech signal from a digital representation.

Hansen et al. [1993] distinguish coded and parametric speech. *Coded speech* is pre-recorded words and phrases which are concatenated and replayed. Coded speech ensures a natural voice and is widely used in voice response systems. Drawbacks are that prosody is impossible to get completely right with today's concatenation technology, and that maintenance of system phrases may be difficult and costly. New phrases to be added must be produced by the speaker who did the previous recording(s), and using the same voice quality, or all words and phrases must be re-recorded.

For *parametric speech* (or synthetic speech), a synthesiser generates an acoustic signal based on a model of human speech. Prosodic features, such as intonation, pauses and stress, may be included in the model and employed on the basis of prosody markers from the system utterance generation inserted on the basis of discourse information [Hirschberg et al. 1995]. Parametric speech makes it easy to generate new system phrases at any time. A drawback is that the parametric speech quality is still low for many languages.

Example: The output speech of the Danish Dialogue System is *coded* as references to pre-recorded phrases that are simply replayed. However, as a recording of system output would have shown, and despite the fact that care has been taken to record phrases uniformly and with an even voice, *prosodic* patterns are still suboptimal.

2.3 Language

Spoken language is very different from written language [Baggia et al. 1994, Waibel 1996]. One of the differences is that people apparently do not follow rigid syntactic and morphological constraints in their utterances. This lack of written-language formality in spontaneous spoken language makes linguistic analysis-by-machine both more difficult than, and different from, analysis of written language. The corresponding, added difficulties involved in the generation of spoken language are less pronounced, if only because human interlocutors are much more capable of decoding the machine's spoken messages.

The language layer includes two types of elements: user (input) utterances and system (output) utterances.

User utterances

The *lexicon* is a list of words, a *vocabulary*, annotated with syntactic (including morphological) and semantic features. The fact that vocabularies of current interactive speech systems are still limited, implies that some application domains cannot be addressed because the required vocabulary is too large.

Grammars describe how words may be combined into phrases and sentences. The input grammar for the application is specified empirically as part of the sub-language identification process. An important goal in input grammar specification is to include all intuitively natural grammatical constructs, possibly up to a certain level of complexity. Users will have little patience with a system which does not accept perfectly ordinary and grammatically simple ways of saying things.

Semantics are abstract representations of the meanings of words, phrases and sentences. In the Danish Dialogue System, syntactic and semantic analysis is done in parallel. Lexical entries are defined as *feature bundles* including lexical value, category (e.g. determiner, ordinal), semantic category (e.g. none, date), gender (e.g. common) and selectional features (e.g. ‘elvte’ can be a month). The grammar has several rules describing the construction of dates. The semantic mapping rules extract semantic values from syntactic sub-trees.

In general, *style* may be analysed in terms of the vocabulary used, which may be formal or informal, slang etc., sentence length, use of adjectives, figures of speech, synonyms, analogies, ellipses, references etc. [Jones and Carigliano 1993]. Style is generally described in terms such as terseness and politeness. In interactive speech systems, user *input style* may be considered an important dependent variable which must be influenced through instruction and example. The aim is to avoid that users address the system in styles that involve lengthy, verbose or convoluted language, such as when users are excessively polite. A system introduction to that effect would appear useful in many cases (cf. Figure 2). Influencing user input style by example is done through the system’s output (see below).

Example: Although the recogniser gets (U3a) in Figure 2 completely right, the *semantic* analysis fails by wrongly choosing the final ‘one’ as the semantic value for the expected number of persons. The problem is caused by the *grammar* which does not accept conjunctions. In (U3a) it would also be difficult for the system to decide if there are four or just two persons who are going to travel because the grammar does not handle co-ordinates such as conjunctions. The word ‘noon’ (U11) is not in the *lexicon*. The general *style* of the user’s utterances is rather terse as required by the system in (S2a). Exceptions are (U4a) and (U13a) which are misrecognised or only partially recognised.

System utterances

The design of system utterances is important to the user’s perception and understanding of, and successful interaction with, the system as well as to how the user will address the system. It is somewhat difficult to distinguish between the effects of output lexicon, output grammar, output semantics and output style. It seems to be a well-established fact that the system’s *style* of speaking influences the way the user addresses the system. If the system is overly polite, users will tend to address the system in a verbose fashion that does not sit well with the need for brief and to-the-point user utterances that can be handled by current speech and language processing [Zoltan-Ford 1991]. Style is a function of, among other things, *grammar* and *lexicon* (cf. above). It seems plausible, therefore, that output grammar and output lexicon do influence the grammar and lexicon to be found in the user’s input. It follows (i) that the output lexicon should not include words which the user may model but which are not in the input lexicon; and (ii) that output grammars should not inspire the user to use grammatical constructs which the system cannot understand.

Example: System utterances in the Danish Dialogue System are constructed using a simple *grammar* that concatenates pre-defined words and phrases. For instance, (S3a-b) is a concatenation of the four words and phrases ‘Customer number’, ‘four’, ‘period’ and ‘How many persons will travel?’ (“period” inserts a short pause). No *lexicon* is used. The system uses a terse and direct *style* of expression.

2.4 Control

Controlling the interaction is a core function in interactive speech systems. Interaction control determines what to expect from the user, how to interpret high-level input structures, consultation of the context elements, what to output to the user, and generally when and how to do what. Being done at run-time, control builds on structures determined at development time. The nature of these control tasks implies that control has to operate on superordinate interaction structures and states. Following [Grosz and Sidner 1986], the interaction model distinguishes three types of superordinate interaction structure and state. The *attentional state* includes the entities in current interaction focus, the *intentional structure* addresses the purposes involved in interaction, and the *linguistic structure* includes characterisation of high-level structures in the input and output discourse.

Attentional state

We use the term *attentional state* [Grosz and Sidner 1986] to refer to the elements that concern what is going on in the interaction at a certain point in time. The attentional state is inherently dynamic, recording the important objects, properties and relations at any point during interaction. The system represents the attentional state as a *focus set.* The focus set includes the set of sub-tasks about which the system is currently able to communicate. The *focus* is the topic which is most likely to be brought up in the next user utterance. For instance, if the system has asked for a departure airport, this topic will be in focus with respect to the next user utterance. If the user instead provides a destination airport this may still be understood if included in the focus set.

Expectations may be attributed to the system if not all sub-tasks are in the focus set. Then expectations serve as a basis for constraining the search space by selecting the relevant subset of the acoustic models, the lexicon and the grammars to be active during processing of the next user input. If the user chooses to address other sub-tasks than those in the focus set, system understanding will fail unless some focus relaxation strategy has been adopted.

Example: The system *focus set* in the Danish Dialogue System comprises the current sub-task, i.e. the one addressed by the system in its latest question and which the user is expected to address in the next utterance, and the user-initiated meta-communication tasks. Based on the system focus, *expectations* concerning what the user will be saying next assist the system in choosing which subset of the acoustic models, the lexicon and the grammars will be used by the recogniser and the parser in decoding the subsequent user utterance. The misunderstanding following (U4a) was partly caused by inadequate system expectations.

Intentional structure

We have chosen the term *intentional structure* [Grosz and Sidner 1986] to subsume the elements that concern tasks and various forms of communication. These elements all concern intentions, or goals and purposes. We distinguish between tasks, communication types, and interaction level. The intentional structure serves to control the transactions of the system.

Intentions can be of many kinds, such as to obtain information, make somebody laugh, or just chat, and are in general not tied to tasks. In today's interactive speech systems, however, spoken human-computer interaction is performed in order for a user to complete one or more tasks. From this task-oriented, shared-goal viewpoint, intentions coincide with task goals.

A single interactive speech system may be able to accomplish several different superordinate tasks. These may all belong to a single domain, such as when the system both performs ticket reservation and provides information on a variety of travel conditions that are not directly related to ticket reservation; or the superordinate tasks may belong to unrelated domains such as the provision of telephone access to email, calendar, weather and stock exchange information [Martin et al. 1996].

We distinguish between *well-structured* and *ill-structured* tasks. Well-structured tasks have a stereotypical structure that prescribes (i) which pieces of information must be exchanged between the interlocutors to complete the task, and often also (ii) a natural order in which to exchange the information. If the stereotype is known, shared and followed by the interlocutors, the likelihood of successful completion of the task is significantly increased. Stereotypical tasks, even when comparatively large and complex, are well-suited for the predominantly system directed or user-directed interaction that is characteristic of today's interactive speech systems. An example is the ticket reservation task stereotype of the Danish Dialogue System shown in Figure 3. This structure conforms to the most common structure found in corresponding human-human reservation task dialogues recorded in a travel agency [Dybkjær and Dybkjær 1993].

Ill-structured or non-stereotypical tasks contain a large number of optional sub-tasks whose nature and order are difficult to predict. An example would be a comprehensive information system on travel conditions. This system would include many different kinds of information at many different levels of abstraction, such as fares, general discount rules, discounts for particular user groups or particular departures, departure times, free seats, rules on dangerous luggage, luggage fees, rules on accompanying persons, pets etc. In specifying the Danish Dialogue System we found that a complex information task of this nature could not be modelled satisfactorily for being accomplished through system directed interaction. The problem was that a user might want a single piece of information which could only be retrieved through a lengthy series of answers to the system's questions. This difficulty might be overcome through more sophisticated interaction models, such as the use of advanced mixed initiative dialogue combined with the use of larger active vocabularies than we had at our disposal.

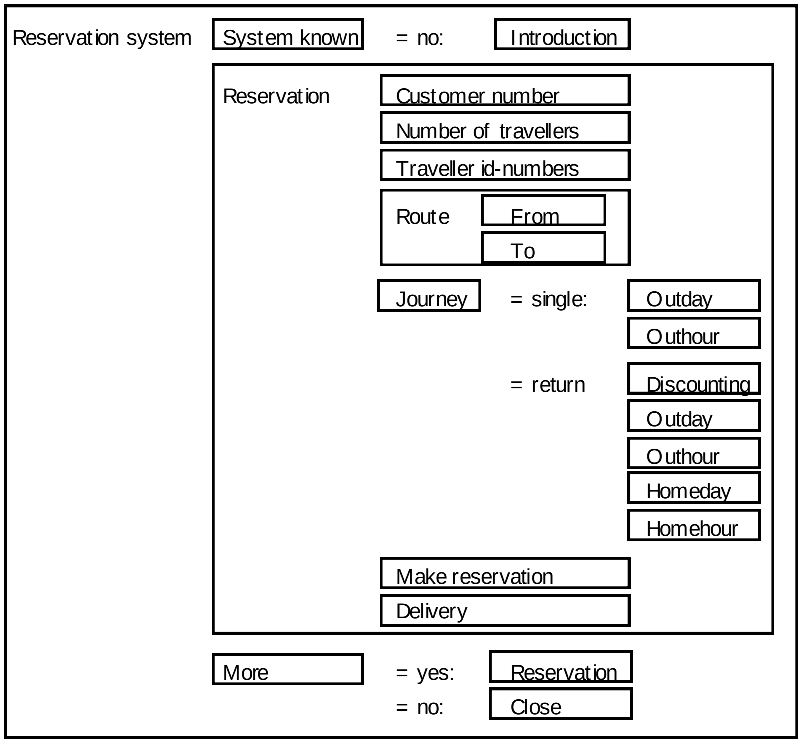

Figure 3: The task structure of the implemented Danish Dialogue System. Meta-communication tasks are not shown. A labelled box indicates a task. If a box A contains another box B then B is a sub-task relative to A. At some points during dialogue the path to follow depends on the user's answer to the most recent system question. In such cases, an answer is indicated as '= [answer]:' followed by the tasks to be performed in this case.

Given a task-oriented approach to interaction theory, there is a relatively clear distinction between three types of interaction between user and system. The first is basic, task-oriented interaction or *domain communication*, which is what the dialogue is all about.

The second interaction type is *meta-communication* which has a crucial auxiliary role in spoken human-machine interaction. Meta-communication serves as a means of resolving misunderstandings and lacks in understanding between the participants during task-oriented dialogue. In current interactive speech systems, meta-communication for interaction *repair* is essential because of the sub-optimal quality of the systems' recognition and linguistic processing of spontaneous spoken language. Similarly, meta-communication for interaction *clarification* is likely to be needed in all but the most simple advanced interactive speech systems.

Domain communication depends on the domain and the dialogue model. Models of meta-communication, on the other hand, might to some extent be shared by applications which are different in task and/or domain [Bilange 1991]. It should be remembered, however, that meta-communication is often domain dependent, such as in "Did you say seven o'clock in the morning?".

In addition to domain- and meta-communication, most interactive speech systems need *other* forms of *communication* which do not belong to either of these two categories.

Finally, the *interaction level* expresses the constraints on user communication that are in operation at a certain stage during interaction. In the extreme, the system may ask the user to spell the input. At the other extreme, no constraints on user input exist beyond those of general user *cooperativity*.

Example: The global structure of the dialogue in the Danish Dialogue System is defined in terms of *tasks*, such as 'reservation' (S2b…S13d), which in their turn include a number of sub-tasks, such as 'customer' (S2b…S3a) and 'route' (S5b…S9a). Note how some tasks, such as 'delivery' (S13d), do not always require user turns. In the dialogue in Figure 2 the reason is that the user has no choice but must pick up the tickets in the airport. If the journey starts more than three days later, the user may choose to have the tickets mailed. *Domain communication* is communication about the task domain and occupies most of the dialogue in Figure 2. As we have seen, users may at any point initiate *meta-communication* to resolve misunderstandings or lack of understanding by using one of the keywords 'repeat' and 'change'. Contrast, for instance, the system's reactions to (U4a) and (U5a). The system ignores the user's meta-communication intention in (U4a) but recognises that intention in (U5a). The system may initiate meta-communication as well, for instance by telling the user that it did not understand what was said (S12a-b). In addition, the dialogue illustrates several phenomena which cannot be characterised as either domain communication or meta-communication, such as the opening greeting "Hello" (S1a), the information about the system itself (S1a, S1b, S2a), and the expressive "Sorry" (S12a). The dialogue in Figure 2 does not show many cases of the system deviating from its standard *level of interaction*. However, following the 'change' command (U5a), the system descends to the more cumbersome, but safer, level of asking for explicit confirmation (S6a-b).

Linguistic structure

The linguistic structure of the interaction includes the elements: speech acts, references and discourse segments.

The *speech act* is a basic unit of conversational theory [Searle 1969, 1979]. All speech acts have *propositional content*, that is, the state of affairs addressed by a particular speech act, such as “departure at 8 o’clock”. Instances of different types of speech acts may have the same propositional content. What distinguishes them, and hence what distinguishes different types of speech act, is what the speakers *do* with their speech. The departure at 8 o’clock, for instance, may be *questioned, promised, ordered* etc.

A particular problem is that speech acts can be indirect as well as direct. In a *direct* speech act, the surface language expresses the intended speech act. An *indirect* speech act is one in which the surface language used does not disclose the “real” act intended by the user. For instance, if someone asks if you have a match, it is likely that the question is not being asked merely in order to be able to record the fact. Rather than being a request for information, this act is a request for the act of providing fire for some purpose, such as lighting a candle. Indirect speech acts remain difficult to identify by machine. Several interactive speech research systems projects have been, or are, wrestling with this problem, such as Esprit PLUS [Grau et al. 1994] and Verbmobil [Jekat et al. 1995].

The handling of *references* (or, strictly speaking, co-references) is a classical problem in linguistics. The problem is that many different words or phrases can refer to the same extra-linguistic entity or entities. Normally, the first occurrence of an expression will make its extra-linguistic reference quite clear. This is not always true but may perhaps be taken for granted in practical, task-oriented written text and spoken discourse. However, given that the first expression has made clear its extra-linguistic reference, language offers many ways of economising with the following, co-referring expressions, i.e. the expressions which have the same extra-linguistic reference as the first one.

State-of-the-art in co-reference handling in current realistic interactive speech systems is that co-reference is not being handled at all but that the problem of co-reference constitutes one of the many reasons why many systems perform word spotting or “robust parsing” rather than full parsing of the users’ input. The point is that co-reference resolution is hard—and not just for machines. However, with the increased sophistication required of the language processing component in interactive speech systems for complex, large-vocabulary tasks, co-reference resolution is becoming an important practical research topic.

Discourse segments are supra-sentential structures in spoken or written discourse. They are the linguistic counterparts of task structure and in the conversational theory of Grosz and Sidner [1986], intentions are restricted to those that are directly related to discourse segments. Each discourse segment is assigned one intention only, the *discourse segment purpose*. Furthermore, the intention as determined by the originator of a given discourse segment must be recognisable by the interlocutors in order to serve as a discourse segment purpose.

Example: At a primitive level, the Danish Dialogue System distinguishes between two types of user *speech acts:* commands and statements. User input in terms of one of the keywords ‘change’ (U5a) and ‘repeat’ is interpreted as commands. All other user input is considered as statements in response to factual system questions. With respect to *reference resolution,* the system handles simple ellipses, such as “From Copenhagen” (U7a) and “Karup” (U8a). The system does not use *discourse segmentation* information.

2.5 Context

Context is of crucial importance to language understanding and generation and plays a central role in interactive speech systems development. The context provides constraints on lexicon, speech act interpretation, reference resolution, task execution and communication planning, system focus and expectations, the reasoning that the system must be able to perform and the utterances it should generate. Contextual constraints serve to remove ambiguity, facilitate search and inference, and increase the information contents of utterances since the more context, the shorter the messages need to be [Iwanska 1995]. Specification of context is closely related to the specific task and application in question. In a sense, each element is part of the context of each other element.

The context layer includes the three generic contextual elements of Figure 1: interaction history, domain model and user model. The interaction history is primarily relevant to the local discourse and used in the dynamic run-time model; the domain model represents the world context in the run-time model; part of the user model is used at run-time whilst other parts are used at development-time only.

Interaction history

An *interaction history* is a selective record of information which has been exchanged during interaction. It is useful to distinguish between at least four types of interaction history.

The *linguistic history* records the surface language, its semantics and possibly other linguistic aspects such as speech acts and the order in which they occurred. The linguistic history encapsulates the linguistic context and is necessary in advanced systems in which the linguistic analysis is no longer context free. For instance, the capture of surface language is needed in cross-sentential reference resolution.

The *topic history* records the order in which sub-tasks have been addressed. The topic history encapsulates the attentional context and is used in guiding system meta-communication.

The *task history* stores the task-relevant information that has been exchanged during interaction, either all of it or that coming from the user or the system, or some of it, depending on the application. The task history encapsulates the task context. It is used in executing the results of the interaction and is necessary in most interactive speech systems. The task history may be used in providing summarising feedback as in the Danish Dialogue System.

The *performance history* updates a model of how well interaction with the user is proceeding. The performance history encapsulates the user performance context and is used to modify the way in which the system addresses the user. Thus the system may be capable of adapting to the user through changing the interaction level.

Example: The *linguistic history* of the Danish Dialogue System is primitive and only records the Boolean contents of the latest system utterance in order to correctly interpret users' "yes" and "no" utterances. For instance, the analysis of (U9a) needs to establish whether "no" means one-way or return. In a different situation, the system might have asked "One-way ticket. Is that correct?" The *topic history* records the order of sub-tasks treated during the dialogue and is used in handling repair and clarification meta-communication as in (U5a). The *task history* stores task-relevant information provided by the user as well as information retrieved from the database. This information is used in the summarising feedback (S13b) and when actually booking the ticket in the flight database, although the current system does not carry out any "real" booking. The system does not use a *performance history*.

Domain model

The domain of an interactive speech system determines the aspects of the world about which the system can communicate. The system usually acts as front-end to some application, such as an email system or a database. The domain model captures the concepts relevant to that application in terms of *data* and *rules*. For instance, during domain-related interaction the system evaluates each piece of user input by checking the input with the application database and/or already provided information stored in the task history. Information retrieved from the application, or provided earlier but to be used now, is checked with the user. The domain model usually has to include both facts and inferences about the application and general world knowledge. Among other things, the database of the Danish Dialogue System contains explicit facts on flight departures, rules stating that the out date must be the same or earlier than the return date, and inference patterns enabling the system to infer dates from input such as "today" (date completion).

A vast literature of general relevance to domain modelling has been produced in disciplines such as artificial intelligence, knowledge bases and expert systems, see [Russell and Norvig 1995]. The interested reader is referred to this literature. Clearly, domain modelling for a particular interactive speech system depends heavily on the application and domain in question.

Example: The *data* of the Danish Dialogue System is consulted after each task-relevant answer from the user. For instance, the system checks that the customer number (U2a-S3a) and the route (U7a-S9a) exist. Additional *rules* define world knowledge that is necessary to the semantic interpretation, such as how to infer the day of the week from a date (S11a).

User model

User modelling is important in interactive speech systems development. The better the system can take aspects such as user goals, beliefs, skills, preferences and cognition into account, the more *cooperative* the system can be [Gasterland et al. 1992]. The general fragility of current speech systems means that they must be particularly carefully crafted to fit the behaviour of their users. Still, even if the subject of user modelling is huge in itself it represents but a single corner of speech interaction models.

At run-time, user *goals* determine which tasks and sub-tasks the system actually has to execute among those that the system is capable of performing. In the Swiss Rail system [Peng and Vital 1996], for instance, the user is assumed to just have one overall goal, namely to obtain train timetable information. Other systems may be capable of satisfying several different general user goals, such as checking emails over-the-phone and consulting an appointment schedule, e.g. [Martin et al. 1996].

The system should model relevant user *beliefs*, i.e. what some or all users believe to be true of the system, the domain and relevant states of affairs in the world. Figure 4 illustrates how crucial a proper understanding of user beliefs can be. After the feedback in (S26a), the Danish Dialogue System assumes that the user accepts the fed back information unless the user subsequently applies the 'change' command. However, the consternated user forgets about the command since the fed back id-number is right but the name is wrong. The user then interprets the system's "Sorry" (S27a) as an acceptance of (U26a) whereas what the system actually meant was "I did not get any relevant information from your utterance". The system should have said (in S27a), for instance, "Sorry, I did not understand. Where does the journey start?" to make sure that the user shares the system's beliefs about the exchange. A system introduction to interaction (cf. Figure 2) is a useful vehicle for modifying the user's expectations with respect to the interaction. More generally speaking, interaction model developers should be prepared to anticipate, sometimes even false, user expectations of many different kinds: concerning the interaction, domain facts, the world etc.

S26a	Id-number 1, Jens Hansen.
b	Where does the journey start?
U26a	No, it is not Jens Hansen, id-number 1 is Lars Bo Larsen.
S27a	Sorry, where does the journey start?
U27a	In Ålborg.

Figure 4: The importance of taking relevant user beliefs into account illustrated from a dialogue with subject number 13.

User *preferences* are options preferred by all, or some, users, such as to let departure time depend on discount availability (domain related), to perform the interactive task in a certain order, or to have the initiative during interaction (interaction related). The latter preference, like many user preferences, may be regarded as a soft constraint, i.e. a constraint that may be ignored at development time if harder constraints have to be satisfied.

User groups represent relevant classifications of potential users. The novice-expert distinction is one such classification. User *expertise* may be characterised along two dimensions: domain novice/expert and system novice/expert. With respect to systems for everyday use, most users can be considered experts to some degree. Thus, most users involved in the development of the Danish Dialogue System were used to book flight (or other forms of transport) tickets. In comparative terms, these users were domain experts although not at the level of travel agents, but they had never before interacted with an interactive speech system. As these users were representative of the intended user population, the system provided little domain help and sought instead to make clear how users should interact with it. In addition to these novice-expert distinctions among users, many other user groupings may have to be taken into account by interactive speech systems developers, for instance distinctions between users from different professional communities, between native and non-native speakers, or between speakers of different dialects. To deal with the latter, the recogniser may apply dialect and language adaptation/identification [Dobler and Ruehl 1995, Hazen and Zue 1994], or do as the Swiss Rail information system does when communication fails: ask the user "Bitte Hochdeutsch sprechen!" ("Please speak High German!").

In addition to user properties such as those mentioned above, developers should keep in mind that users have to perform rapid, situation-dependent *cognitive processing* during interaction and that users' capabilities of doing so are severely limited.

Example: In the Danish Dialogue System the user is assumed to only have the *goal* of making a reservation as is made clear in (S2b). The system models the user's *beliefs* via a status field for each information item. For instance, when starting the second reservation task (S14a), the system, using the task history, assumes that the user believes the customer number to be the same as in the previous reservation task and asks for confirmation (S14a-b) instead of asking anew as in (S2b). Had the user's answer to the return ticket question (S9b) been "yes", the system would have asked if the user has a *preference* for discount fares and their associated departure times. A model of the user serves to guide adaptation to users during the dialogue. Thus the system's introduction (S2a) provides information to the users who lack *expertise* with the system (S1b, U1a). In (U4a) the user forgets to use the keyword 'change' for repair meta-communication, probably due to *cognitive overload* after the misrecognition in (S4a). This suggests that designer-designed keywords, such as 'Change', are a liability in interactive speech systems.

3 Characterising systems

The presentation of speech interaction theory in the preceding section provides few specific choices of means of representation or algorithms. Its primary aim is to offer a conceptual structure for speech interaction theories, models and systems.

In Figure 5 we illustrate the theory's potential for providing high-level system overviews. Writings on systems, parts of systems, and system experiments tend to document only selected parts of the overall system, and the documentation does not have any standard conventions to follow. It is therefore often difficult or impossible to compare results, because of insufficient context, and systems, because of insufficient and incomparable information. One approach to reducing these very real problems is to use a standardised scheme which may provide the minimum information required for describing an interactive speech system in a way which contextualises the results presented and allows comparison with other systems. Figure 5 presents one such scheme which describes the Danish Dialogue System based on speech interaction theory.

The interaction model of the Danish dialogue system
The Danish dialogue system is a realistic research prototype of a telephone based interactive speech system for reservation of Danish domestic flight tickets.

System performance	
Cooperativity	Conformance with the guidelines (Section 4.2).
Initiative	Overall system initiative; users may initiate meta-communication.
Influencing users	Explicit and implicit user instructions; walk-up-and-use system.
Speech input	Continuous; speaker-independent; Danish.
Acoustic models	Based on HMMs; whole word models; approximately 500 words; at most 100 words active at a time; word-accuracy (laboratory) 78%.
Grammar	Bigrams and finite state network mixture.
Prosody	-
Speech output	Normal human voice; Danish.
Coded/parametric	Coded speech.
Prosody	-
User utterances	
Lexicon	Approximately 500 words; lexical entries defined as feature bundles.
Grammar	APSG.
Semantics	Mapping rules extract semantic values from syntactic sub-trees.
Style	Terse.
System utterances	
Lexicon	Pre-defined words and phrases.
Grammar	Simple grammar for concatenating pre-defined words and phrases.
Semantics	-
Style	Terse.
Attentional state	
Focus	Current sub-task plus meta-communication tasks.
Expectations	Predictions sent to recogniser and parser; task dependent parsing.
Intentional structure	
Tasks	Danish domestic flight ticket reservation; well-structured task.
Communication	System-directed domain communication. Mixed initiative meta-communication; users may initiate meta-communication through keywords. System-directed other communication, such as the opening and closing of a dialogue.
Interaction level	Some questions are yes/no or multiple choice, most are general and focused.
Linguistic structure	
Speech acts	Primitive distinction between commands (meta-communication) and statements (answers) in user input; use of commands (questions), and statements for providing feedback, error messages and other information in output.
References	No anaphora resolution; ellipses are handled.
Segments	-
Interaction history	
Linguistic	Only semantic contents.
Topic	Order of exchanges.
Task	Information exchanged.
Performance	-
Domain model	
Data	Timetable, fares, flights, customers, reservations.
Rules	Completions and constraints.
User model	
Goals	Assumed to be flight ticket reservation.
Beliefs	Handled to a moderate extent at run-time.
Preferences	Determined at run-time; the scope is the current reservation task.
User group	System novice/expert distinction; the system’s introduction and discount information is optional.
Cognition	Natural response packages addressed; cognitive overload problem.

Figure 5: High-level description of the Danish Dialogue System (cf. Figure 1).

4 Conclusion

We have described elements of a speech interaction theory. The claim made on behalf of the theory has been rather modest, i.e. that the theory provides a set of concepts which are sufficient for the description of current interactive speech systems technologies at some level of abstraction. Given the diverse origins of these concepts, it is very likely that future theoretical work will produce a more homogeneous set of concepts. For systems development practice, however, this may be of less importance than the achievement of a number of extensions to the theory. One important class of much needed extensions consists of further distinctions and taxonomies. For instance, we still lack an articulate and practically useful typology of interactive speech systems which may help developers conceptualise the “bundle of system properties” they are dealing with. And we lack abstract task concepts from which developers may derive clusters of system properties needed for a system to perform a given task in a particular application domain. The latter example points to a deeper need. It is to transform the interactive speech theory presented in this paper into a practically useful *relational theory*. By this we mean a theory which not only delivers isolated, or only partially related, concepts as we have done above, but which makes explicit the many hidden relations between the different elements of interactive speech systems. Armed with a “complete” relational theory, the developer will be able to derive large numbers of system properties from an initial system specification, thereby avoiding the trial-and-error approach characteristic of much of current interactive speech systems development. It should be added here, though, that nobody can tell at this point how complete an eventual relational theory will turn out to be.

As a final observation, we would like to refer to the distinction between design-time and run-time properties of interactive speech systems. This distinction is not a fixed one. One of the reasons why habitable and natural interactive speech systems remain hard to build, is the amount of user-adaptive crafting to be done at design time and to be tested in lengthy Wizard of Oz simulations, controlled user studies and field trials. The more the system can be made capable of adapting to users at run-time through the use of more sophisticated techniques than those we possess at present, the easier it will become to build truly natural interactive speech systems.

References

[Amalberti et al. 1993] René Amalberti, Noëlle Carbonell, and Pierre Falzon: User representations of computer systems in human-computer speech interaction. *International Journal of Man-Machine Studies*, 38, 1993, 547-566.

[Aust et al. 1995] Harald Aust, Martin Oerder, Frank Seide, and Volker Stenbiss: The Philips automatic train timetable information system. *Speech Communication* 17, 1995, 249-262.

[Baggia et al. 1994] Paolo Baggia, Elisabetta Gerbino, Egidio Giachin, and Claudio Rullent: Spontaneous speech phenomena in naive-user interactions. In *Proceedings of TWLT8*, 8th Twente Workshop on Speech and Language Engineering, Enschede, The Netherlands, 1994, 37-45.

[Bahl et al. 1983] Lalit Bahl, Frederick Jelinek, and Robert L. Mercer: A maximum likelihood approach to continuous speech recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 5, 2, 1983, 179-190.

[Bernsen et al. 1998] Niels Ole Bernsen, Hans Dybkjæ and Laila Dybkjær: *Designing Interactive Speech Systems. From First Ideas to User Testing.* To be published by Springer Verlag 1998.

[Bilange 1991] Eric Bilange: A task independent oral dialogue model. In *Proceedings of the 5th EACL,* Berlin, 1991, 83-88.

[Bunt 1994] Harry Bunt: Context and dialogue control. *Think* 3, ITK, Tilburg University, the Netherlands, 1994, 19-31.

[Carberry 1990] Sandra Carberry: *Plan Recognition in Natural Language Dialogue*. Cambridge, Massachusetts, MIT Press, 1990.

[Dobler and Ruehl 1995] S. Dobler and H.-W. Ruehl: Speaker adaptation for telephone based speech dialogue systems. In *Proceedings of Eurospeech'95*, Madrid, Spain, 1995, 1139-1141.

[Dybkjær et al. 1996] Laila Dybkjær, Niels Ole Bernsen, and Hans Dybkjær: Evaluation of Spoken Dialogues. User Test with a Simulated Speech Recogniser. Report 9b from the Danish Project in Spoken Language Dialogue Systems. Roskilde University, 3 volumes of 18 pages, 265 pages, and 109 pages, respectively, 1996.

[Dybkjær and Dybkjær 1993] Laila Dybkjær and Hans Dybkjær: Wizard of Oz Experiments in the Development of the Dialogue Model for P1. Report 3 from the Danish Project in Spoken Language Dialogue Systems, Roskilde University, 1993.

[Eckert and McGlashan 1993] Wieland Eckert and Scott McGlashan: Managing spoken dialogues for information services. In *Proceedings of Eurospeech'93*, Berlin, 1993, 1653-1656.

[Gasterland et al. 1992] Terry Gasterland, Parke Godfrey, and Jack Minker: An overview of cooperative answering. *Journal of Intelligent Information Systems*, 1, 1992, 123-157.

[Grau et al. 1994] Brigitte Grau, Gérard Sabah, and Anne Vilnat: Control in man-machine dialogue. *Think*, 3, 1994, 32-55.

[Grice 1975] Paul Grice: Logic and conversation. In P. Cole and J. L. Morgan (Eds.), *Syntax and Semantics* Vol. 3: *Speech Acts*. New York: Academic Press 1975, 41-58. Reprinted in Paul Grice: *Studies in the Way of Words*. Cambridge, MA, Harvard University Press, 1989.

[Grosz and Sidner 1986] Barbara J. Grosz and Candace L. Sidner: Attention, intentions, and the structure of discourse. *Computational Linguistics* 12, 3, 1986.

[Grosz et al. 1989] Barbara J. Grosz, Martha E. Pollack, and Candace L. Sidner: Discourse. In Michael I. Posner (Ed.): *Foundations of Cognitive Science.* MIT Press, 1989, 437-468.

[Hansen et al. 1993] Peter Molbæk Hansen, Peter Holtse, Henrik Nielsen, and Niels Reinholt Petersen: Speech synthesis — Teaching a computer spoken language. In *Teleteknik* 1-2, 1993, 52-65.

[Hazen and Zue 1994] Timothy J. Hazen and Victor W. Zue: Recent improvements in an approach to segment-based automatic language identification. In *Proceedings of ICSLP'94*, Yokohama, Japan, 1994, 1883-1886.

[Hirschberg et al. 1995] Julia Hirschberg, Christine H. Nakatani, and Barbara J. Grosz: Conveying discourse structure through intonation variation. In *Proceedings of the ESCA Workshop on Spoken Dialogue Systems*, Vigsø, Danmark, 1995, 189-192.

[Iwanska 1995] Lucja Iwanska: *Summary of the IJCAI-95 Workshop on Context in Natural Language Processing*, Montreal, Canada, 1995.

[Jekat et al. 1995] Susanne Jekat, A. Klein, Elisabeth Maier, Ilona Maleck, Marion Mast, and Joachim Quantz: Dialogue acts in VERBMOBIL. *Verbmobil Report* 65, Universität Hamburg, DFKI Saarbrücken, Universität Erlangen, TU Berlin, 1995.

[Jones and Carigliano 1993] Cerian Jones and Roberto Carigliano: Dialogue analysis and generation: A theory for modelling natural English dialogue. In *Proceedings of Eurospeech'93*, Berlin, 1993, 951-954.

[Jönsson 1993] Arne Jönsson: *Dialogue Management for Natural Language Interfaces. An Empirical Approach*. Ph.D. thesis, Linköping Studies in Science and Technology No. 312, Sweden, 1993.

[Kamp 1992] Yves Kamp: Introduction to Continuous Speech Recognition. Fourth European Summer School in Logic, Language and Information, Essex, England, 1992.

[Kamp and Reyle 1993] Hans Kamp and Uwe Reyle: *From Discourse to Logic*. Dordrecht, Kluwer Academic Publishers, 1993.

[Litman 1985] D. Litman: Plan Recognition and Discourse Analysis: An Integrated Approach for Understanding Dialogues. Technical Report TR 170, University of Rochester, NY, 1985.

[Mann and Thompson 1987a] William C. Mann and Sandra A. Thompson: Rhetorical structure theory: A theory of text organisation. In Livia Polanyi (Ed.): *The Structure of Discourse*. Norwood, NJ, Ablex Publishing Company, 1987, 85-96.

[Mann and Thompson 1987b] William C. Mann and Sandra A. Thompson: Rhetorical structure theory: Description and construction of text structures. In Gerard Kempen (Ed.): *Natural Language Generation. New Results in Artificial Intelligence, Psychology and Linguistics.* NATO ASI Series E No. 135, Chapter 7. The Netherlands, Martinus Nijhoff Publishers, 1987.

[Martin et al. 1996] Paul Martin, Frederick Crabbe, Stuart Adams, Eric Baatz, and Nicole Yankelovich: SpeechActs: A spoken language framework. *IEEE Computer* 29, 7, 1996.

[Peng and Vital 1996] J.-C. Peng and F. Vital: Der sprechende Fahrplan. *Output* 10, 1996.

[Rabiner 1988] Lawrence R. Rabiner: Mathematical foundations of hidden Markov Models. In Heinrich Niemann, M. Lang, and G. Sagerer (Eds.): *Recent Advances in Speech Understanding and Dialog Systems*, NATO ASI Series F: Computer and Systems Sciences, Vol. 46, Springer Verlag, 1988, 183-206.

[Russell and Norvig 1995] Stuart Russell and Peter Norvig: *Solution Manual for Artificial Intelligence: A Modern Approach*. Englewood Cliffs, NJ: Prentice Hall, 1995.

[Searle 1969] John R. Searle: *Speech Acts. An Essay in the Philosophy of Language.* Cambridge, Cambridge University Press, 1969.

[Searle 1979] John R. Searle: *Expression and Meaning. Studies in the Theory of Speech Acts.* New York, Cambridge University Press, 1979.

[Smith and Hipp 1994] Ronnie W. Smith and D. Richard Hipp: *Spoken Natural Language Dialog Systems: A Practical Approach*. New York, Oxford University Press, 1994.

[Sperber and Wilson 1987] D. Sperber and D. Wilson: Précis of relevance, communication and cognition with open peer commentary. *Behavioral and Brain Sciences* 10, 4, 1987, 697-754.

[Waibel 1996] Alex Waibel: Interactive translation of conversational speech. *IEEE Computer*, 1996, 41-48.

[Zoltan-Ford 1991] Elisabeth Zoltan-Ford: How to get people to say and type what computers can understand. *International Journal of Man-Machine Studies* 34, 1991, 527-547.

[Zue 1994] Victor W. Zue: Toward systems that understand spoken language. *IEEE Expert*, 1994, 51-59.