Context representation and reasoning for managing spoken dialogue

Harry Bunt
Computational Linguistics and Artificial Intelligence Unit
Tilburg University
P.O. Box 90153, 5000 LE Tilburg, Netherlands
Email: bunt@kub.nl

Abstract

In interactive systems, the 'dialogue manager' is a central component, whose primary task is to decide what to do next in a dialogue. This decision involves taking a variety of kinds of context information into account, with different logical properties, used in different types of reasoning, and therefore with different demands on representation. In this paper we discuss the representation of the context information needed to support intelligent dialogue management in interactive speech systems. We argue that simple types of context information should be represented in a way that optimally supports efficient simple forms of reasoning, while other types of context information require sophisticated logics for articulate representation and full-blown reasoning.

Using different types of representation for different types of context information creates a problem in situations where a type of context information, that is adequately represented in a very simple form most of the time, exceptionally requires articulate representation and sophisticated reasoning. We indicate how this problem can be resolved, and how a dialogue manager can use several forms of context representation and reasoning for generating dialogue acts, by clearly distinguishing the different ways in which dialogue acts may be constructed and the conditions under which a particular construction is to be used.

Keywords: dialogue context representation, dialogue management, reasoning with context information.

1 Introduction

Dialogue management is the process of deciding what to do next in a dialogue. In human communication, dialogue management is a process that seems to go unnoticed much of the time; only rarely are we away of having to make a decision on what to say next, or on whether to say anything at all. When designing a computer dialogue system, however, we either have to pre-program the possible dialogues according to certain fixed sequence of utterances, or else we have to define a 'dialogue manager', who decides what the system should do next based on a model of the current dialogue context.

Research on the design of intelligent computer dialogue management systems has also inspired investigations of human dialogue management. These investigations have made it clear that human dialogue management is actually highly complex and sophisticated, and suggests that one of the stumbling blocks for the development of high-quality speech dialogue systems is the design of dialogue managers that have some of the sophistication and subtlety of human dialogue management.

In this paper, we will examine the notion of 'dialogue context' in the sense of the information that is relevant for deciding what to do next in a dialogue. We will model the communicative behaviour of dialogue participants using *Dynamic Interpretation Theory,* a theoretical framework for dialogue analysis, that views communication in terms of actions, intended to change the context in certain ways. In this paper we will use Dynamic Interpretation Theory for the analysis of dialogue management phenomena.

This paper is organized as follows. We first consider the basics of dialogue management within the framework of Dynamic Interpretation Theory. We then turn to an analysis of the types of information needed to support effective dialogue management. We identify some of the most important logical and computational properties of these information types and consider the consequences of these properties for computer representations of dialogue contexts. We argue that dialogue management relies crucially on context information, where each type of context information should be represented in a way that optimally supports efficient reasoning.

Finally, we look at the problem, created by the use of different representation forms for different types of context information, in situations where a type of context information, that is adequately represented in a very simple form most of the time, exceptionally requires articulate representation and sophisticated reasoning. We will indicate how this problem can be resolved, and how a dialogue manager can use several forms of context representation and reasoning for deciding on the generation of the appropriate kinds of dialogue acts, by clearly distinguishing the different ways in which dialogue acts may be constructed and the conditions under which a particular construction is to be used.

2 Dialogue management in DIT

Dialogue management, as we said above, is the process of deciding what to do next in a dialogue. This decision must take a lot of considerations into account. Often, participating in a dialogue goes hand in hand with some noncommunicative activity. For instance, the information service representative engaged in a dialogue with an client alternates talking to the client with consulting her computer information system, and the client may be taking notes and consulting his agenda while talking to the information service. A computer dialogue system playing the part of one of the participants also has to perform both kinds of action, and the dialogue manager of such a system therefore has to interact closely with other processes, in particular with processes that gather and combine pieces of information, in short, with *reasoning* processes. Some of this information may be highly complex, as we will see, and require sophisticated logics and inference procedures. For effective dialogue management, it is vital that to keep the representation and processing of conceptually simple context information simple, rather than casting all types of context information into a single formalism for the sake of uniformity.

The two kinds of activity that a dialogue partner is typically engaged in, communicative and noncommunicative, can both motivate a contribution to the dialogue, In Dynamic Interpretation Theory two classes of communicative acts are distinguished, corresponding to these two underlying motivations: *task-oriented* dialogue acts and *dialogue control* acts. Task-oriented acts are motivated by the non-communicative purpose or 'task' underlying of the dialogue; dialogue control acts monitor the interactive process, helping to create and maintain the conditions for smooth and successful communication.

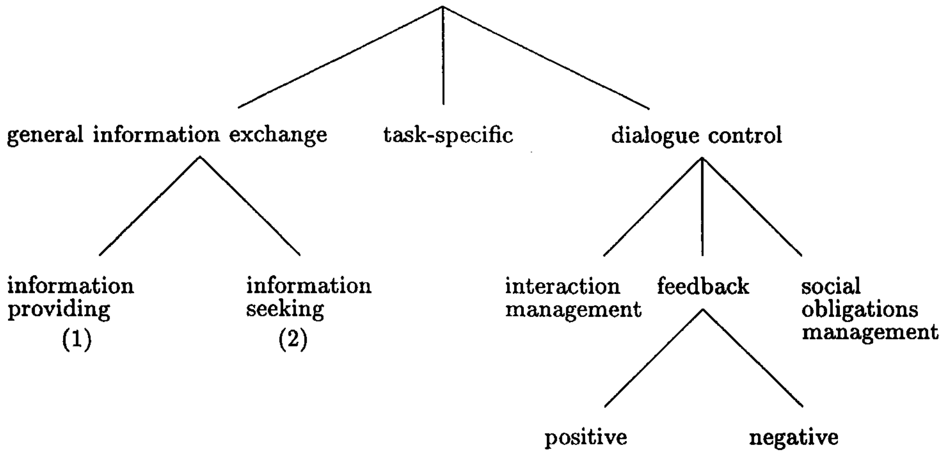
2.1 Dialogue acts

Dynamic Interpretation Theory (DIT) defines both task-oriented and dialogue control acts as context-changing operations. We have defined the concept of a *dialogue act* as the functional units used by dialogue participants to change the context (Bunt, 1994). A dialogue act has a *semantic content,* formed by the information the speaker introduces into the context, and a *communicative function* that defines the significance of this information by specifying how the context should be updated with the information. By saying that dialogue acts are units 'used by dialogue participants', we mean that every communicative function corresponds to a particular set of features of the communicative behaviour that dialogue participants display.

Communicative function can be divided into three categories: those specific for dialogue control, those specific for a particular type of task domain, and those that have a general use for exchanging any kind of information. Task-specific communicative functions are for instance OFFER, ACCEPTANCE and REFUSAL in negotiation dialogues (see Jekat et al., 1995; Alexandersson, 1996).

For dialogue control acts we have proposed a classification into three subsystems, concerned with *feedback, interaction management,* and *social obligation management* (Bunt, 1994). Feedback acts provide information about the processing of partner inputs, reporting or resolving problems (negative feedback), or reporting successful processing (positive feedback). Social obligation management acts deal with socially indicated obligations such as welcome greeting, thanking, apologizing, and farewell greeting. Interaction management acts handle various aspects of the interactive situation, such as taking turns; pausing and resuming; monitoring attention and contact; and structuring the discourse explicitly.

In Bunt (1989) we have presented a hierarchical system of general information-exchange functions, where at the top of the hierarchy we find two subclasses of dialogue acts, those concerned with *information seeking* and those with *information providing.* Figure 1 provides a schematic overview of these subsystems.



(1), (2): hierarchies of communicative functions defined in Bunt (1989).

Figure 1: *Subsystems of communicative functions.*

A general information-exchange function, such as INFORM, can be combined with discourse domain information as semantic content, and form a task-oriented dialogue act, as in (1a), or it may combine with interaction-related information and form a dialogue control act, as in (1b).

(1) a. The next train to Amsterdam leaves at 9.05.

b. I didn't hear you what said.

Both for dialogue control acts and for task-specific dialogue acts there are thus two ways to form them from a communicative function and a semantic content. This is summarized in Table 1. Example (2) illustrates the two ways of forming a task-specific dialogue act, and (3) the two ways of forming a dialogue control act.

(2) a. Twenty-five thousand.

b. I am willing to offer twenty-five thousand.

(3) a. Thank you.

b. I am extremely grateful for your help.

Note that the task-specific functions in an information dialogue are just the information-seeking and -providing functions; therefore, information dialogues constitute the one and only kind of dialogue for which there are in fact *no* task-specific communicative functions.

comm. function	*semantic content*	*dialogue act type*
general info. exchange	task domain information interaction information	task-oriented dialogue control
task-specific	(task domain information)	task-oriented
dialogue control	(interaction information)	dialogue control

Table 1: Possible constructions of dialogue act types

2.2 Dialogue management mechanisms

As mentioned above, communicative acts may be motivated either by the underlying purpose of the dialogue or by the purpose of securing the conditions for effective communication. These motivations form the basis of dialogue mechanisms if we make the following idealizing assumptions:

Rationality Communicative agents act in order to achieve something. They form communicative goals to fulfill their underlying goals and desires; choose their communicative actions so as to optimally further their communicative goals, and organize the interaction so as to optimize the chances of success of their communicative actions.

Sociality Communication is a form of social behaviour, and is thus subject to cultural norms and conventions. An important aspect of this is **Cooperativity**, i.e. the disposition to act so as to be optimally helpful for the dialogue partner, taking his goals, interests and abilities into account.

From these assumptions, the motivations of the various types of communicative act can be derived as follows.

- Task-oriented acts are motivated by the speaker's underlying task goals (Rationality) or by his knowledge of the partner's goals (Cooperativity).
- Dialogue control acts:
  - *Interaction management* acts and *negative feedback* acts are motivated by the drive to communicate successfully (Rationality);
  - *Positive feedback* acts are motivated partly by Rationality and partly Sociality (see below).
  - *Social obligation management* acts are motivated by the desire to honour social obligations (Sociality).

These motivations can all be construed as relations between dialogue acts and aspects of dialogue context. In the next section we consider these aspects of context.

3 Types of dialogue context information

We will follow the division of dialogue acts into task-oriented (TO-) acts and dialogue control (DC-) acts, with the further distinctions indicated in Fig. 1 and below, to consider the types of context information that a dialogue manager needs to generate the various types of acts.

Task-oriented dialogue acts

The generation of a TO-act, being motivated by the underlying dialogue purpose or 'task', depends on the state of the task that the speaker is trying to accomplish. For instance, suppose the speaker is planning a trip, and contacts an information service to obtain travel information. In order to obtain specific information, a question is the appropriate act; the primary condition for asking a question being that the speaker wants to know the information he asks for. Additional conditions for the generation of this act will include the assumption that the information service does have the information requested (which may not be obvious, depending for instance on where the trip is going) and a number of seemingly trivial assumptions (e.g. Searle (1969)'s 'preparatory conditions'). All information-seeking acts share these assumptions. Finer distinctions within the class of information-seeking acts correspond to additional assumptions, such as the CHECK having the condition that the speaker weakly believes the semantic content to be true (see further Bunt, 1989).

While information-*seeking* TO-acts nearly always have their origin in information needs that the speaker experiences in trying to achieve the underlying task, information-*providing* acts often derive from cooperativity: the speaker believes it to be in the hearer's interest to obtain certain information. This requires the speaker to have beliefs about the underlying task and its current state, from the hearer's point of view. For a dialogue manager to generate TO-acts, it must have information about the underlying task and the task domain, as well as about the dialogue partner's information (and this recursively).

Dialogue control acts

The generation of a *feedback act* is triggered by difficulties that the speaker encounters in processing an utterance or by successful completion of such processing (negative and positive feedback, respectively). Generating a negative feedback act seems relatively easy for a dialogue manager, provided that it receives sufficiently clear failure messages from modules that run into difficulties; depending on the seriousness of a difficulty, the dialogue manager should generate a feedback act with the appropriate priority.

The generation of positive feedback acts is conceptually more difficult to regulate, since there is some conceptual unclarity as to why and when such acts should be produced. An optimally efficient dialogue strategy might seem to be one that assumes communication by default always to be successful; positive feedback would then never be necessary. Natural human dialogue, especially spoken dialogue, is full of positive feedback, however, and complete absence of it is experienced as uncomfortable. Some spoken dialogue systems report all the time what they have recognized; this introduces too much redundancy and slows the interaction down too much. Positive feedback seems desirable some of the time, but not all the time.

Three reasons for performing positive feedback acts are:

1. The speaker assesses the dialogue context as 'risky': he has reason to believe that the hearer might think that something in the speaker's processing went wrong, where this is in fact not the case (Rationality).

2. The speaker has not contributed to the dialogue for a considerable period of time; 'it is about time to say something' (Sociality).

3. It is the speaker's turn to contribute to the dialogue, but he cannot immediately come up with something to say. In that case, a positive feedback act can be used as a harmless 'filler'.

Whatever precisely the mechanisms for generating feedback acts may be, they require a speaker's context to contain knowledge of his own processes of recognition, interpretation, evaluation and execution, what we call his *own processing status*, as well as knowledge of the dialogue partner's processing status (for corrective feedback).

It may be noted that the reaction to a negative feedback act often requires knowledge of the preceding discourse; for instance, *"Did you say Thursday?"* and *"Do you mean Friday April the second or Thursday April the first?"* require knowledge of what was said and what was meant, respectively.

Interaction management (IM) acts, at least in spoken dialogues, are primarily concerned with turn-taking, timing, contact, dialogue structuring, and the utterance formulation process. In a corpus of 111 naturally occurring spoken information dialogues (see Beun, 1989) the most important cases of *turn management* were (TURN-GIVING, TURN-KEEPING and (INTERRUPTION). Dialogue participants apparently have a view on who is having the main speaker role ('the turn').

Time management acts are concerned with the amount of time that various activities may take. When a speaker needs some extra time to formulate his contribution, he may produce a STALL act (*"Ehm"*, *"Let's see"*,..); when he needs more time, for instance to collect some information or to deal with an interfering activity, he is likely to produce a PAUSE act, to prevent his dialogue partner from worrying about the reasons of a prolonged silence. The generation of such acts depends on speakers having an estimate of the time needed to perform various activities.

Contact management acts are used by speakers to monitor contact with the dialogue partner both in the 'physical' sense *"Hello?"*), as when checking someone's presence at the other end of a telephone line, and in the sense of giving and having attention. The generation of these acts thus depends on the speaker having beliefs about the physical/perceptual dialogue context and about whether attention is being paid.

Own communication management acts (borrowing Allwood's term; see Allwood et al., 1992) are used by a speaker to deal with disfluencies in his contribution to the dialogue; what is in the literature also referred to as 'self-repair'. The generation of such acts by a dialogue manager depends on the availability of information about difficulties in the output generation processes, which is very similar to the kind of information we have seen for the generation of feedback acts.

Discourse structuring acts are used to explicitly structure the interaction, indicating for example that a certain topic is being closed, that a new topic is going to be addressed, or that the speaker intends to ask a question (*"I have a question."*). The generation of discourse structuring acts is based on the speaker's view of the current context and on his plan for continuing the dialogue. We have argued elsewhere (Bunt, 1999) that a separate representation of topics is not necessary if there is an articulate representation of goals and related aspects of the state of information, especially if the dialogue history keeps a record of goals that have been achieved earlier in the dialogue.

Social obligations management (SOM) acts reflect that natural language communication between human partners is subject to conventions of social behaviour; for instance, when one does something wrong or when is unable to help, one is expected to apologize. And when meeting an acquaintance, one is supposed to greet.

For dealing with such 'social obligations', languages have closed classes of utterances with the property that their use puts a pressure on the addressee to react with an utterance from a related class. For example, a *Thank you"* creates a pressure to say something like *"You're welcome"*, and a greeting creates a pressure to respond with a return greeting. SOM acts characteristically come in pairs, of what in the terminology of the Geneva School is called an 'initiative' and a 'reaction' (see Moeschler, 1985; Roulet, 1985).

In Bunt (1994) we have introduced the notion of *reactive pressures* (RPs) to capture this phenomenon: an 'initiative' SOM-act has as part of its definition a description of the reactive pressure it introduces into the context. To account for the occurrence of initiative SOM-acts, we have further introduced the concept of *interactive pressures* (Bunt, 1996). An interactive pressure (IP) differs from an RP in that it is created not by a particular SOM-act, but by properties of the context that are created in the course of the dialogue.

RPs and IPs lead to the generation of SOM-acts if we assume that communicative agents always aim at resolving such pressures. This is a way of turning the observation made above that speakers have "the desire to honour social obligations" into a mechanism for dialogue act management.

An IP is created by an IP principle, which is a pair consisting of (1) a set of local context conditions that must be satisfied for the pressure to arise; (2) a specification of a communicative function (or class of functions) and a set of constraints on semantic content and utterance form. See Bunt (1996), for more about IP principles.

In sum, this analysis leads us to distinguish the following types of dialogue context information:

1. (a) Knowledge of the task and the task domain. We call this aspect of dialogue context the *semantic* context.
   (b) Knowledge of the dialogue partner's semantic context (and this recursively).
2. (a) Knowledge of one's own processing status.
   (b) Knowledge of the partner's processing status.
3. Knowledge of the preceding discourse; we call this the *linguistic context.*
4. Knowledge for IM-acts:
   (a) Knowledge of the allocation of turns. This can be considered part of the linguistic context.
   (b) Knowledge of the time needed by various processes. This can be considered part of the processing status.
   (c) Knowledge of the physical and perceptual/cognitive 'contact' between speaker and hearer. We call this the *physical/perceptual* context.
   (d) Knowledge of one's own contribution production processes. This can be considered part of one's processing status.
   (e) Knowledge of the discourse structure. This can be considered part of the linguistic context.
5. Interactive and reactive pressures. We call this the (local) *social* context.

It is worth noting that all types of dialogue context information consist of beliefs, knowledge, goals, obligations ('pressures'), and other attitudinal elements. When we speak of 'the dialogue context', we mean in fact always the dialogue context *according to the speaker*. What a speaker does next in a dialogue depends only on how *he* views the context; for the purpose of dialogue management, dialogue contexts are always subjective.

The importance of identifying the above types of context information is that the various information types have different logical properties, and the recognition of these properties allows us to design effective and efficient inference systems to support dialogue management. We now turn to a brief discussion of these properties.

3.1 Logical properties of information types

For computationally adequate modeling of dialogue contexts, the crucial issue is how the context information can be exploited in reasoning processes. Logically complex information should be represented in an articulate way, allowing inference procedures to exploit the logical significance of the articulation, while logically simple information should be represented in a simple way, allowing fast, direct computation instead of searching and complex reasoning. We therefore consider the information types distinguished above with the following two questions:

1. What is the logical articulation and complexity of the information?
2. Do communicating agents use the information in full-blown inferential processing or in simpler, special-purpose processing such as checking the value of a parameter?

Full-blown inferential processing naturally goes hand in hand with articulate expression of information, while simpler processing is appropriate for information with little internal structure. The articulation of information types and the complexity of the associated processing can be investigated empirically by examining the semantic contents of the dialogue acts addressing the various information types, and by investigating the complexity of subdialogues where the various kinds of context information are the topic of conversation.

3.1.1 Semantic context

The semantic content of task-oriented dialogue acts can be quite articulate, reflecting the complexity of the task domain. It is the subject of the most elaborate discussions in a dialogue, that dialogue participants reason about, make plans about, and use to guide their communicative activity in a rational way.

Semantic context information, moreover, is often embedded within recursive belief attitudes as part of an agent's information about his dialogue partner. As such, it is the most complex kind of context information, combining the inherent complexity of task domain with that of nested propositional attitudes. An adequate representation of this information therefore calls for a logically sophisticated formalism with inference machinery.

3.1.2 Processing Status

Processing status information is needed by a dialogue manager for the generation of feedback acts, time management acts, and own communication management acts. This means that an agent's processing status should contain the following elements per process: (1) state of progress; (3) any difficulties encountered; (2) results obtained; (4) estimated time needed for completion.

In natural information dialogues the estimated time needed by a process is never expressed in precise terms; we will therefore consider only the representation of time estimates in the same crude way people do this. This can be accomplished with a simple parameter-value pair. The same holds for feedback acts about reporting the state of progress of a process and for inarticulate own communication management acts.

In negative feedback acts, when processing difficulties are reported, the speaker most of the time signals the failure of a process, as in *I beg you pardon?*, or asks for clarification of a particular item, as in *This Tuesday?*. This may be represented with two parameters: one representing success or failure, and one that may have a problematic item as its value. In the case of negative feedback w.r.t. evaluation, an agent reports conflicts between new information and previously available information. To detect such conflicts may require full-blown inferential processing. It is therefore not surprising that such feedback acts are often expressed not with dedicated feedback functions, but with a general information-exchange function, using a full sentence with articulate semantic contents. In such a case a parameter-value representation would be inadequate. We will deal with this in the next section.

3.1.3 Linguistic context and dialogue memory

Linguistic context, a recording of what has happened in the dialogue, can serve as a *memory*. This has the advantage that all other components of dialogue context do not need to have a memory! Consider, for example, the modelling of a participant's beliefs and intentions relating to the underlying task. It would not suffice to only model the *current* beliefs and intentions, for an agent may sometimes discover that something went wrong, and should then be able to return to a previous state of beliefs and intentions. By associating with each utterance in the linguistic context the changes that the utterance has brought about in the semantic context, we obviate the need to 'remember' these changes in the semantic context; any previous state of the semantic context can be reconstructed from its current state plus the changes recorded in the linguistic context. The same goes for other context components. [1]

The information that participants have about turn allocation is conveniently tegrated with the linguistic context, which can be seen as a 'dialogue history' (cf. Prince and Pernel, 1995), that also records the turns in the preceding dialogue. In order to take the participants' anticipations as to how the dialogue will continue into account, we extend the dialogue history with a planned

[1] Besides this backward-looking aspect, linguistic context in general also has a forward-looking aspect, containing a speaker's discourse plans, which are the basis for discourse-structuring acts. It may also be very practical to use a *context buffer* for storing the most recent dialogue history, especially in view of the fact that complete, definitive processing of inputs often does not occur immediately. Such a buffer has for instance been implemented in the TENDUM dialogue system (Bunt et al., 1984) and more recently in the DENK system, where it is called the 'pending context' (see Ahn et al., 1994; Bunt et al., 1998; Piwek, 1998).

'dialogue future', which can include the representation of the present and future allocation of turns.

This view on the linguistic context has been implemented in the linguistic context model of the PLUS dialogue system; see Bunt and Black (1999).

3.1.4 Physical/perceptual context

Of the physical and perceptual context, which characterizes the ways the dialogue participants can interact with their environment, including each other, the only aspects that can be changed by the dialogue are the effective availability of communication channels and whether the participants are paying attention to each other.

In the case of a telephone dialogue we thus need to represent in the speaker's physical and perceptual context his assumptions about the current availability of the telephone line. Whether the participant who is not speaking is paying attention, is indistinguishable form the availability of the communication channel. The contact management acts we have found in telephone information dialogues confirm this; physical and cognitive 'availability' are not addressed separately. A single parameter is thus sufficient to represent the partner's assumed physical and mental 'presence'. When the participants can see each other, additional parameters may be needed.

3.1.5 Social context

The local social context consists of the 'pressures' to perform SOM-acts, created either by IP rules expressing the conditions under which a certain 'initiative' SOM-act is appropriate, or by a preceding SOM-act. These 'pressures' are in fact partical specifications of dialogue acts, typically containing a communicative function and a context-dependent semantic content, and possibly some constraints on the linguistic surface realization. As such, these pressures are represented by substructures of the data structures (such as nested feature structures) that are used for the linguistic context.

3.2 Information types and reasoning

This brief analysis of the logical properties of the various dialogue context information types leads to the conclusion that a computational model of dialogue context should distinguish the following kinds of structures:

1. formulas in an expressive logical language, like typed lambda calculus with propositional attitude operators;
2. parameters with simple, unstructured values;
3. nested feature structures, or something similar for representing linguistic information.

Parameters may be considered to be a special case of feature structures, so this may actually boil down to just two kinds of representational structures. It should be borne in mind, however, that the reason for distinguishing parameter-value representations is that they allow extremely efficient 'reasoning', consisting simply of checking the value of a parameter, which is all that is required for many dialogue control acts; this should not be lost by embedding these representations in a more powerful formalism, which may be formally elegant but which entails unnecessarily more complex processing.

4 Dialogue control with articulate information

The representation of processing status, perceptual/physical context, and interactive/reactive pressures by simple parameter-value structures has computational advantages; it also has limitations. Consider, for instance, a dialogue system that supplies travel information, and a traveller who asks for departure times of KLM flights from Amsterdam to Beijing on Monday. It so happens that there are no such flights on Monday. The system should then produce a negative feedback act containing that message, like (4b)

(4) a. What are the departure times of KLM flights to Beijing next Monday?

b. There are no KLM flights to Beijing on Monday.

but this semantic content is too complex to be represented by a parameter-value structure; it is in fact a piece of task domain information, and is thus represented like all semantic context. How can the dialogue manager generate such a feedback act? The answer depends on the difference between the two ways to construct a dialogue control act, noted before and indicated in Table 1.

Compare the feedback example (4b) with one where the speaker's uterrance was unintelligible for the system. In that case the system's processing status will contain something like `Recognition:[PROGRESS:fail]`, and from this the dialogue manager will generate a negative feedback act that might be phrased as *"Please repeat"*, an act with zero semantic content. In the case of (4), we assume that a system component that is called to find departure times of flights to Beijing is unable to process a request, and something like `Execution:[PROGRESS:fail]` is recorded in the processing status. On the basis of that information, the dialogue manager might generated a feedback message such as *"Execution of request impossible."* But there is more, for the search for flights to Beijing was initiated by the system trying to construct an answer to (4a), an answer which the system is unable to construct. Now one of the IP rules expressing cooperativity says that if the system is unable to perform a requested action or to answer a question, it should explain the reason why. If the reasoning process, involved in constructing the answer to the user's question 'knows' that there are no flights to Beijing on Monday, it will be able to conclude that for this reason no pertinent departure times can be found, and the dialogue manager will generate an explanatory INFORM act with that content. So the dialogue manager will generate two dialogue acts, to be phrased e.g. as:

(5) a. Execution of request impossible.

b. There are no KLM flights from Amsterdam to Beijing on Monday.

The dialogue manager should be sophisticated enough to schedule these acts in the right order, or even to note that in this case (5b) alone is to be preferred, since it implies (5a).

We started out by considering the generation of an articulate negative feedback act like (4b) - but the question might well be raised whether (4b) is actually a feedback utterance. In its appearance, it is simply an INFORM act. Indeed, there is little reason to consider the utterance in any other way than an INFORM; like other INFORMs, it is generated on the basis of a cooperativity principle expressing that, when an agent has information available that he thinks is in the interest of the partner and which the partner does not have, than this information should be provided.

Similar analyses may be applied to other cases of dialogue control acts with articulate semantic content. Such dialogue acts are invariably constructed with the help of general information-exchange functions (see Table 1), and can be generated as well as interpreted by the dialogue manager simply by taking their communicative functions as such, rather than looking for an interpretation as a dialogue control act. Note that this corresponds with the view that, when dialogue control information is discussed in an articulate manner, then the domain of discourse is temporarily shifted from the underlying task to the interactive process. This in contrast with the use of special-purpose DC functions plus marginal content, where the domain of discourse remains that of the underlying task (as corroberated by the study of topic shifts in spoken information dialogues by Rats, 1996). When a dialogue shifts to interaction-related information, the state of the task-related information is frozen until the dialogue returns there.

5 Concluding remarks

Intelligent dialogue management in mixed-initiative spoken dialogue systems relies on taking a variety of types of context information into account. In order to do this in an efficient manner, we have argued that logically simple context information should be represented in a simple way, and logically complex information in a more articulate way.

Using different types of representation for different types of context information creates a problem in situations where a type of context information, that is adequately represented in a very simple form most of the time, exceptionally requires articulate representation and sophisticated reasoning. We have indicated how this problem can be resolved, and how a dialogue manager can use several forms of context representation and reasoning for deciding on the generation of dialogue acts, by clearly distinguishing the different ways in which dialogue acts may be constructed and the conditions under which a particular construction is to be used.

References

Ahn, R., Beun, R.J., Borghuis, T., Bunt, H.C., & van Overveld, C. (1994) The DENK architecture: a fundamental approach to user interfaces. *Artificial Intelligence Review 8 (2)*, 431-445.

Alexandersson, J. (1996) Some ideas for the automatic acquisition of dialogue structure. *Proc. 11th Twente Workshop on Language Technology.* Enschede: University of Twente, 149-158.

Allwood, J., Nivre, J. & Ahlsén, E. (1992) On the semantics and pragmatics of linguistic feedback. *Journal of Semantics 9*, 1, 1-29.

Beun, R.J. (1989) *The recognition of declarative questions in information dialogues.* PhD Thesis, Tilburg University.

Bunt, H.C. (1989) Information Dialogues as Communicative Actions in Relation to User Modelling and Information Processing. In M.M. Taylor, F. Néel & D.G. Bouwhuis (eds.) *The Structure of Multimodal Dialogue.* Amsterdam: North-Holland Elsevier, 47-73.

Bunt, H.C. (1994) Context and Dialogue Control. *THINK Quarterly 3(1)*, 19-31.

Bunt, H.C. (1996) Dynamic Interpretation and Dialogue Theory. In: M.M. Taylor, F. Néel & D.G. Bouwhuis (eds.) *The Structure of Multimodal Dialogue, Vol. 2.* Amsterdam: John Benjamins.

Bunt, H.C. (1997) Requirements for dialogue context modelling. *Proceedings First International and Interdisciplinary Conference on Modelling and using Context*, Rio de Janeiro.

Bunt, H.C. (1999) Iterative context specification and dialogue analysis. In H.C. Bunt & W.J. Black (eds.) *Abduction, Belief and Context: Studies in Computational Pragmatics.* London: University College Press (forthc.), 73-129.

Bunt, H.C., Beun, R.J., Dols, Linden, J. van der & Schwartzenberg, G. thoe (1984) The TENDUM dialogue system and its theoretical basis. *IPO Annual Progress Report 19*, 105-113.

Bunt, H.C. & Black, W.J. (1999) The ABC of Computational Pragmatics. In H.C. Bunt & W.J. Black (eds.) *Abduction, Belief and Context: Studies in Computational Pragmatics.* London: UCL Press, 1-35.

Bunt, H.C., Ahn, R., Kievit, L., Piwek, P., Verlinden, M., Beun, R.J. and Overveld, C. van (1998a) Multimodal Cooperation with the DENK System. In: H.C. Bunt, R.J . Beun & T. Borghuis (eds.) *Multimodal Human-Computer Communication: Systems, Techniques and Experiments.* Lecture Notes in Artificial Intelligence 1374, Berlin: Springer, 39-67.

Bunt, H.C., Kievit, L., Piwek, P., Verlinden, M., Beun, R.J., Borghuis, T. and Overveld, C. van (1998b) Cooperative dialogue with the multimodal DENK system. In: H.C. Bunt, R.J. Beun, T. Borghuis, L. Kievit & M. Verlinden (eds.) *Proc. 2nd Int. Conference of Cooperative Multimodal Communication CMC/98.* Tilburg: SOBU, Tilburg University.

Jekat, S., Klein, A., Maier, E., Maleck, I., Mast, M. & Quantz, J. (1995) Dialogue Acts in Verbmobil. *Verbmobil Report 65-95*, University of Hamburg.

Maier, E. (1996) Context construction as a subtask of dialogue processing - the Verbmobil case. *Proc. 11th Twente Workshop on Language Technology: Dialogue Management in Natural language Systems.* Enschede: University of Twente, 113-122.

Moeschler, J. (1985) *Argumentation et conversation, éléments pour une analyse pragmatique du discours.* Paris: Hatier.

Piwek, P.L.A. (1998) *Logic, Information, and Conversation.* PhD Thesis, Eindhoven University of Technology.

Prince, V., & Pernel, D. (1995) Several knowledge models and a blackboard memory for human-machine robust dialogues. *Natural Language Engineering 1 (2)*, 113-145.

Rats, M.M.M. (1996) *Topic management in information dialogues.* PhD Thesis, Tilburg University.

Roulet E. (1985) *L'articulation du discours en francais contemporain.* Bern: Peter Lang.

Searle, J.R. (1969) *Speech Act Theory.* Cambridge, UK: Cambridge University Press.