Reducing Miscommunication in Spoken Human-Machine Dialogue

Laila Dybkjær, Niels Ole Bernsen and Hans Dybkjær

Centre for Cognitive Science, Roskilde University
PO Box 260, DK-4000 Roskilde, Denmark
emails: laila@cog.ruc.dk, nob@cog.ruc.dk, dybkjaer@cog.ruc.dk
phone: (+45) 46 75 77 11 fax: (+45) 46 75 45 02

Abstract

This paper presents a principled approach to reducing the occurrence of communication failure in spoken language dialogue systems. A set of principles for cooperative human-machine dialogue has been developed based on the development of the dialogue component of a spoken language dialogue system and on human-human dialogue theory. The principles have been tested on the dialogue corpus from a controlled user test of the implemented system. The paper demonstrates how the principles enabled systematic classification and analysis of the user test data on system miscommunication. In addition, the user test confirmed the broad scope of the principles as only minor additions and revisions were needed to provide a complete classification of the test data. The principles may have other uses in addition to that of test data analysis and dialogue evaluation. Potentially, they might serve as guidelines for the design of cooperative dialogue during early dialogue design.

1. Introduction

It is the system designer's task to prevent human-machine miscommunication from seriously damaging the user's task performance. Such prevention is done in two ways. One is to prevent miscommunication from occurring in the first place, the other is to prevent miscommunication, once it has occurred, from producing task failure. Given current speech and language technologies, the possibilities of on-line handling of clarification and repair meta-communication are seriously limited. Furthermore, miscommunication always leads to additional user-system exchanges. It follows that the goal of reducing the amount of miscommunication that will occur is a highly important one. Reduced meta-communication is a source of increased dialogue quality and efficiency. On-line repair and clarification meta-communication will still be needed, of course. In particular the speech recognition capabilities of spoken language dialogue systems (SLDSs) are still fragile. Meta-communication functionality is needed to overcome the effects of system misrecognitions. In addition, users will inevitably provide input which, although recognised by the system, requires clarification or repair dialogue.

This paper proposes principled ways of reducing the occurrence of communication failure in SLDSs and presents a systematic classification of test data on miscommunication. The results presented are based on the development and controlled user testing of the dialogue component of the Danish dialogue system. The system is an SLDS in the domain of flight ticket reservation. The dialogue model of the system had to satisfy several technological constraints which were mainly imposed by the choice of hardware and the speech recogniser, while at the same time being as natural as possible. Those constraints effectively prevented the use of user-initiated domain communication. Fortunately, however, the ticket reservation task is a well-structured task, i.e. the information to be exchanged in order to achieve the task goal is to a large extent predetermined. The ticket reservation task thus lends itself to system-directed *domain communication* in which the user responds to questions asked by the system. With respect to *meta-communication*, on the other hand, the dialogue is mixed-initiative. Whenever needed, users may initiate meta-communication to repair system misunderstanding or lack of understanding by using one of the keywords 'change' and 'repeat'.

Given the approach to dialogue initiative just described, it was crucial to reduce the number of cases in which users might be inclined to take other forms of dialogue initiative, such as asking questions, providing information which the system had not asked for or initiating less constrained forms of meta-communication. This is why the issue of dialogue cooperativity became central to our design of the dialogue structure. We had to optimise system dialogue cooperativity in order to prevent cases such as those described from occurring. To this end, we have developed a set of general principles to be observed in the design of cooperative, spoken human-machine dialogue.

The principles of cooperative dialogue design were developed on the basis of a Wizard of Oz (WOZ) corpus collected during dialogue model development and consolidated through analytic comparison with a body of principles of cooperative human-human dialogue. The principles were then tested on a corpus of dialogues collected during a controlled user test of the implemented Danish SLDS. Section 2 briefly presents the development and consolidation of the principles and describes the user test. Section 3 provides a systematic classification, illustration and over-

view of the user test data based on the principles. Section 4 briefly discusses user errors and Section 5 concludes the paper.

2. Principles of Cooperative Dialogue Design

The system runs on a PC with a DSP board and is accessed over the telephone. It is a walk-up-and-use application which uses robust parsing to understand speaker-independent continuous spoken Danish with a vocabulary of approximately 500 words. The prototype runs in close-to-real-time and is representative of advanced current systems. Comparable SLDSs are (Aust & Oerder 1995; Cole et al. 1994; Eckert et al. 1995).

2.1 Dialogue Model Development and Principles of Cooperative Dialogue

The dialogue model was developed by the Wizard of Oz (WOZ) experimental prototyping method in which a person simulates the system to be designed (Fraser & Gilbert 1991; Dybkjær, Bernsen, & Dybkjær 1993). Development was iterated until the dialogue model satisfied the design constraints on, i.a., maximum and average user utterance length (10 and 4 words, respectively). The dialogues were recorded, transcribed, analysed and used as a basis for improving the dialogue model. The seven WOZ iterations yielded a transcribed corpus of 125 task-oriented human-machine dialogues corresponding to approximately seven hours of spoken dialogue. A total of 24 different subjects were involved in the iterations. Dialogues were based on written task descriptions (scenarios).

A major concern during WOZ was to detect problems of user-system interaction that might lead to miscommunication or actually did so. We eventually used the following two approaches to systematically discover such problems: (i) prior to each WOZ iteration, we matched the scenarios to be used against the current dialogue model by performing designer walk-throughs of the dialogue model based on the scenarios. The model was represented as a graph structure with system phrases in the nodes and expected contents of user answers along the edges. A deviation from the graph would indicate a potential dialogue design problem which should be removed, if possible. (ii) The recorded dialogues were plotted onto the dialogue model graph. As in (i), graph deviations indicated potential dialogue design problems. All deviations were marked and their causes analysed whereupon the dialogue model was revised, if necessary (Dybkjær, Bernsen, & Dybkjær 1996b).

At the end of the WOZ design phase, the problems of interaction uncovered during WOZ were analysed and represented as violations of principles of cooperative dialogue. Each problem was considered a case in which the system in addressing the user had violated a principle of cooperative dialogue. The principles of cooperative dialogue were made explicit, based on the problems analysis. The WOZ corpus analysis led to the identification of 14 principles of cooperative human-machine dialogue based on analysis of 120 examples of user-system interaction problems (Bernsen, Dybkjær, & Dybkjær 1996a; Dybkjær, Bernsen, & Dybkjær 1996b). If the principles were observed in the design of the system's dialogue turns, we hypothesised, this would serve to reduce the occurrence of user dialogue behaviour that the system had not been designed to handle and which might lead to miscommunication.

2.2 Consolidation of the Principles of Cooperative Dialogue

Having developed the principles of cooperative system dialogue, we became aware of the similarity between our work and Gricean cooperativity theory. We analytically compared our principles with Grice's Cooperative Principle (CP) and maxims (Grice 1975). Grice's Cooperative Principle (CP) says that, to act cooperatively in conversation, one should make one's "conversational contribution such as is required, at the stage at which it occurs, by the accepted purpose or direction of the talk exchange in which one is engaged". Grice proposes that the CP can be explicated in terms of four groups of simple maxims which are not claimed to be jointly exhaustive nor to have been generated on a principled theoretical basis other than the CP itself (Figure 1). As a result of the comparison between our initial principles and Grice's maxims, the principles achieved their present form as shown in Figure 1. A detailed comparison with Grice's work is presented elsewhere (Bernsen, Dybkjær, & Dybkjær 1996a). Briefly, the main difference between Grice's work and ours is that the maxims were developed to account for cooperativity in human-human dialogue, whereas our principles were developed to account for cooperativity in human-machine dialogue. Grice focused on the inferences which an interlocutor is able to make when the speaker *deliberately* violates one of the maxims in order to make a dialogue contribution through what Grice calls 'conversational implicature'. Our primary interest rather is in *non-deliberate* violations of maxims and principles. It is exactly when a human or an SLDS non-deliberately violates a maxim that miscommunication is likely to occur. However, whether violated deliberately or non-deliberately, the principles or maxims are the same and their function remains that of helping to achieve the shared dialogue goal as directly and smoothly as possible.

Comparison between Grice's maxims and our principles produced a clear-cut result. The principles include the maxims as a subset. In addition, the principles manifest aspects of cooperative task-oriented dialogue which were not addressed by Grice. The distinction between *principle* and *aspect* is important because an aspect represents the property of dialogue addressed by one or several particular maxims or principles. Finally, the comparison suggested the distinction between *generic* and *specific* principles. As shown in Figure 1, Grice's maxims are all generic. However, a generic principle may subsume one or more specific principles which specialise the generic principle to certain classes of dialogue phenomena. Although subsumed by

DIALOGUE ASPECT	GP NO.	SP NO.	GENERIC OR SPECIFIC PRINCIPLE
Group 1: **Informativeness**	GP1		*Make your contribution as informative as is required (for the current purposes of the exchange).
		SP1	*Be fully explicit in communicating to users the commitments they have made.*
		SP2	Provide feedback on each piece of information provided by the user.
	GP2		**Do not make your contribution more informative than is required.*
Group 2: **Truth and evidence**	GP3		*Do not say what you believe to be false.
	GP4		**Do not say that for which you lack adequate evidence.*
Group 3: **Relevance**	GP5		*Be relevant, i.e. be appropriate to the immediate needs at each stage of the transaction.
Group 4: **Manner**	GP6		*Avoid obscurity of expression.
	GP7		*Avoid ambiguity.
		SP3	*Provide same formulation of the same question (or address) to users everywhere in the system's dialogue turns.*
	GP8		**Be brief (avoid unnecessary prolixity).*
	GP9		**Be orderly.*
Group 5: **Partner asymmetry**	GP10		Inform the dialogue partners of important non-normal characteristics which they should take into account in order to behave cooperatively in dialogue. Ensure the feasibility of what is required of them.
		SP4	Provide clear and comprehensible communication of what the system can and cannot do.
		SP5	Provide clear and sufficient instructions to users on how to interact with the system.
Group 6: **Background knowledge**	GP11		Take partners' relevant background knowledge into account.
		SP6	Take into account possible (and possibly erroneous) user inferences by analogy from related task domains.
		SP7	*Separate whenever possible between the needs of novice and expert users (user-adaptive dialogue).*
	GP12		Take into account legitimate partner expectations as to your own background knowledge.
		SP8	Provide sufficient task domain knowledge and inference.
Group 7: **Repair and clarification**	GP13		Initiate repair or clarification meta-communication in case of communication failure.
		SP9	*Provide ability to initiate repair if system understanding has failed.*
		SP10	Initiate clarification meta-communication in case of inconsistent user input.
		SP11	Initiate clarification meta-communication in case of ambiguous user input.

Figure 1. Principles of cooperative system dialogue. GP means generic principle. SP means specific principle. The principles that were *not* found violated in the user test are indicated in italics. Grice's maxims are marked with an asterisk.

generic principles, we believe that specific principles are important in SLDS dialogue design and evaluation. If generality is all we need, one principle is enough, such as "Be cooperative" or Grice's CP. However, such general expressions are not very helpful in telling us what to look for in the WOZ data or in the data from a user test. The generic principles constitute a distinct improvement and the specific principles provide more focus still.

Some of the specific principles may hold only for spoken human-machine dialogue and not for human-human dialogue. For example, SP3 *(provide same formulation of the same question (or address) to users everywhere in the system's dialogue turns)* should not be practiced in human-human dialogue because this would lead to very monotonous dialogues. Moreover, interlocutors might still interpret the same question in different ways depending on the context. However, human-computer dialogues typically have a very restricted context. This minimises the risk that users will interpret different occurrences of an identically expressed question in different ways. In addition, SP3 has two positive side-effects: (i) since users tend to model the system's vocabulary, SP3 may help limit users' vocabulary; (ii) if the computer behaves too much like a human interlocutor, users may forget that they are talking to a computer or may over-estimate the dialogue skills of the system. This will increase user-system miscommunication.

None of the principles appear to conflict in general. However, concrete SLDS design situations may generate difficult trade-offs. For instance, when designing the introduction to our SLDS we had to trade off GP2 and GP8, on the one hand, and SP4 on the other. The difficult question was how much information is sufficient but not too much, given the immense differences in communicative skills between humans and machines. This question is not made easier by the facts that users are very different and that limited distinction between the needs of novice and expert users (SP7) does not reflect the detailed needs of each single user. Another problem became apparent in the WOZ experiments when the system did not explicitly communicate to users the commitments they had made (against SP1) with respect to change of reservation. This led some users to ask for confirmation. On the other hand, several users had in the previous WOZ iteration complained that the system provided too much information in general (viola-

tion of GP2). Our present conclusion is that users appreciate explicit confirmation of the commitments they make during ticket reservation, i.e. that the confirmation conforms to GP2. In information tasks in which users do not commit themselves to anything, implicit feedback may well be sufficient.

2.3 The User Test: Identification of Dialogue Interaction Problems

When the system had been implemented and debugged, a controlled user test was carried out. In this test, a simulated speech recogniser was used (Bernsen, Dybkjær, & Dybkjær 1995). A wizard keyed in the users' answers into the simulated recogniser. The simulation ensured that typos were automatically corrected and that input to the parser corresponded to an input string which could have been recognised by the real recogniser. In this set-up, recognition accuracy would be 100% as long as users expressed themselves in accordance with the vocabulary and grammars known to the system. Otherwise, the simulated recogniser would turn the user input into a string which only contained words and grammatical constructs from the recogniser's vocabulary and rules of grammar.

The test was based on 20 different scenarios which had been designed to enable exploration of all aspects of the task structure. Twelve novice subjects, mostly professional secretaries, participated in the user test. The subjects conducted the dialogues over the telephone in their normal work environments. They were given a total of 50 particular tasks based on 48 scenarios. A *task* consists in ordering one or more tickets for one route. The number of recorded dialogues was 57 because subjects sometimes reiterated a failed dialogue and eventually succeeded with the task. A *dialogue* is one path through the dialogue structure.

Each dialogue was recorded and all transactions between the individual system modules were logged. The recorded dialogues were transcribed and analysed. The analysis aimed at detecting problems of dialogue interaction and was done as follows. Based on the dialogue structure, a template was built which contained the system's questions. For each scenario, normative system questions and user answers were filled into the template. The key contents of the actual dialogues were then plotted into the template (Dybkjær, Bernsen, & Dybkjær 1996a). Comparison between normative and actual system and user utterances led to the identification of three main classes of interaction problems: (1) linguistic problems, (2) problems of dialogue interaction and (3) other problems, such as cases of system breakdown. (2) splits into (a) dialogue design problems and (b) user errors. The following section focuses on (a).

PRINCIPLE VIOLATED	COOPERATIVITY PROBLEM	No.	TF	CAUSE/REPAIR
GP1	**Less information than required provided by system** (final question too open; withholding important information, requested or not).	19		System question design (4). System response design (15).
GP3	**False information provided by system** (on departures).	2		Database design.
GP5	**Irrelevant information provided by system.**	2	1	Speech recognition design.
GP6	**Obscure system utterance** (grammatically incorrect response; departure information).	7		System response grammar design (1). System response design (6).
GP7	**Ambiguous system utterance** (question on point of departure).	2		System question design.
GP10	**System requirements not followed** (indirect response, change through comments, asking questions, answering several questions at a time).	33		Unreasonable system demands on users. Improve the system to handle the violations.
SP2 (GP1)	**Missing system feedback on user information.**	2	1	System response feedback design.
SP4 (GP10)	**Missing or unclear information on what the system can and cannot do** (system does not listen during its own dialogue turns).	33	1	Speech prompt design.
SP5 (GP10)	**Missing or unclear instructions to users on how to interact with the system** (under-supported user navigation: use of 'change'; round-trip reservations).	2	1	User instruction design.
SP6 (GP11)	**Lacking anticipation of domain misunderstanding by analogy.**	3		User information design.
SP8 (GP12)	**Missing system domain knowledge and inference** (temporal inference; inference from negated binary option).	4		System inference design.
SP10 (GP13)	**Missing clarification of inconsistent user input** (system jumps to wrong conclusion).	5		System clarification question design.
SP11 (GP13)	**Missing clarification of ambiguous user input** (system jumps to wrong conclusion).	5	2	System clarification question design.

Figure 2. Typology of the 119 problems of cooperative dialogue design identified in the user test. The number of occurrences of each problem is shown as are the responsibilities for transaction failure (TF) per problem type. The rightmost column shows the global cause(s) of the problems and hence what needs to be repaired to prevent them from occurring.

3. Classification of Dialogue Design Problems

The principles presented in Figure 1 were used to classify the dialogue interaction problems that had been identified in the user test. Sixteen of the 13 GPs and 11 SPs (Figure 1) were found violated. In the following, the identified dialogue interaction problems are analysed and discussed as violations of the principles for cooperative dialogue. We then discuss problem classification and the coverage of the principles. Finally, the non-violated principles are examined to answer the question why they were not found violated in the data.

3.1 Violated principles

Figure 2 presents an overview of the problem types that were identified in the user test. The figure only contains 13 GPs and SPs. However, although no direct violation of GP11, GP12 and GP13 was found, these principles are regarded as violated because one or more of the specific principles subsumed by each of them were violated. For each violated GP and SP, the identified cooperativity problems are briefly characterised with global indications of the causes and repair of those problems.

In the analysis, each identified design problem was (a) characterised with respect to its *symptoms*, (b) a *diagnosis* was made, and (c) a *cure* was proposed. Representative examples of violations of each of the principles are presented in Figures 3 to 15. In these figures, **S** means symptom, **D** diagnosis and **C** cure. S means system and U user. In several cases, the presented user-system exchange (**S**) contains more than one problem and thus violates more than one principle. However, each figure only diagnoses a single dialogue design problem. We shall sometimes refer to the co-existing problems in an exchange without providing a full symptom-diagnosis-cure description of them.

S: S: There is no departure at 21:00, the closest other red departures are at 19:45 and 22:40. Do you want one of these departures? U: Yes. S: Which? U: Repeat; S: There is no departure at 21:00. Which?.
D: The system only repeats part of its previous response excluding the list of possible departures.
C: Let the system repeat its entire previous utterance since it is impossible to know in advance which part a user will be most interested in having repeated.

Figure 3. A dialogue design problem from Dialogue G33a. The violated principle is GP1: Make your contribution as informative as is required (for the current purposes of the exchange).

S: S: At which time? U: Morning. S: The green morning departures are fully booked (happens twice).
D: There is no grammar failure. According to the flight file the first departure on the given day is 9:15 which is not considered morning but 'before noon' by the database, and which is not fully booked. Therefore, the system's message should have been that there are no morning departures.
C: Find and repair the bug in the database.

Figure 4. A dialogue design problem from Dialogue G13b. The violated principle is GP3: Do not say what you believe to be false.

S: S: At which date will the journey start? U: The journey starts on January 23. S: Today is not the 23rd of January (happens twice).
D: A system misrecognition (grammar failure) triggers an error message which may seem rather irrelevant to the user who did not mention today at all.
C: Two possibilities: 1. The grammar should handle the input in question (the journey start on January 23). 2. Use threshold values (score values) to evaluate the probability of correct recognition and start repair if the score is too low rather than producing a system response which in all likelihood is based on misunderstanding.

Figure 5. A dialogue design problem from Dialogue G22a. The violated principle is GP5: Be relevant, i.e. be appropriate to the immediate needs at each stage of the transaction.

S: S: At which time? U: He wants to depart at 10:45. S: There is no departure at 10:45, the closest other departure is at 17:00. Do you want this departure?
D: The system provides misleading information since the departure at 17:00 is the only departure that day. The user is in doubt whether or not there is an earlier departure.
C: The system should provide non-obscure information, e.g. by saying that the departure at 17:00 is the only departure that day.

Figure 6. A dialogue design problem from Dialogue T53a. The violated principle is GP6: Avoid obscurity of expression.

S: S: Where does the journey start? U: It starts in Hellerup.
D: Hellerup is not the name of an airport. It is a suburb of Copenhagen and is not in the system's vocabulary. The user considers the point of departure to be the traveller's home rather than the airport. This might be a priming effect from the scenario.
C: Option 1. Add more names (hardly feasible). Option 2. The system should ask for the departure airport and not just where the travel starts (or ends).

Figure 7. A dialogue design problem from Dialogue T52a. The violated principle is GP7: Avoid ambiguity.

S: U: Yes, please. I would like to have 9:40. S: 22:40. At which date will the return journey start? U: No, 9:40.
D: The user is too occupied with the problem to remember to use the keyword 'change'.
C: 'Change' is not natural. Prefer mixed-initiative metacommunication.

Figure 8. A dialogue design problem from Dialogue T22a. The violated principle is GP10: Inform the dialogue partners of important non-normal characteristics which they should take into account in order to behave cooperatively in dialogue. Ensure the feasibility of what is required of them.

S: S: Do you want a return ticket? U: No, we will book that one later. [Pause] **It's okay. They are not going back.** S: Red discount may be obtained on certain weekday departures and green discount may be obtained in weekends. In case of red discount the home journey may only start after two days at the earliest. Are you particularly interested in discount?
D: There is no direct system feedback on the user's answer to the question whether s/he wants a return journey. Hence it is difficult to detect misunderstandings at once unless one knows the system/domain well. Discount can only be obtained on return tickets.
[Continued from previous page.]

However, if the user does not know this (which in the current case s/he apparently does not), the misunderstanding will only show later in the dialogue when the system asks for a return date.
C: Provide system feedback to all user answers.

Figure 9. A dialogue design problem from Dialogue T21a. The violated principle is SP2: Provide feedback on each piece of information provided by the user. Boldface shows overlapping user speech.

S: S: At 19:50. U: **Yes.** S: You have now booked ...
D: Phatic. A common way for humans of expressing that one follows the conversation and is interested during the interlocutor's turn. Pauses in particular may be filled in this way. However, the system does not listen during its own turns, and users were not informed of this.
C: Inform users that the system does not listen during its own turns.

Figure 10. A dialogue design problem from Dialogue T41a. The violated principle is SP4: Provide clear and comprehensible communication of what the system can and cannot do. Boldface shows overlapping user speech.

S: S: At which time? U: At 18:10. S: There is no departure at 18:10. The closest other departures are at 18:00 and 22:00. Do you want one of these departures? U: No. S: At which time? U: 15:45 from Esbjerg.
D: The user is supposed to book a round-trip ticket which s/he, as is common, perceives as a return ticket. Round-trip tickets, however, can only be handled by the system as two separate single reservations. When asked about the home departure hour for the second time, the user tries and fails to input a departure airport for the home journey that is different from the arrival airport for the out journey. The system does not provide information on how to handle roundtrips.
C: The system should inform users on its limitations with respect to roundtrip reservations.

Figure 11. A dialogue design problem from Dialogue G54a1-2. The violated principle is SP5: Provide clear and sufficient instructions to users on how to interact with the system.

S: S: Do you want more? U: Did I get a discount?
D: The user wants discount and does not know that this is unavailable on one-way journeys.
C: The system should take into account users' background knowledge by mentioning that, if the user has chosen one-way tickets, discount cannot be obtained.

Figure 12. A dialogue design problem from Dialogue T13a. The violated principle is SP6: Take into account possible (and possibly erroneous) user inferences by analogy from related task domains.

S: S: At which time [out journey]? U: A quarter past seven. ... S: At which time [home journey]? U: A quarter to six. S: The home journey must not be earlier than the out journey.
D: Hour of departure for home journey. The system checks the temporal relation with the out journey. The home departure time 'a quarter to six' is understood as am by default. However, it is not ambiguous in the present context and should have been understood as 17:45.
C: Provide a general solution to the class of problems exemplified by: if out and home journey are on the same day and the home departure time is 'a quarter to six' then understand the answer as '17:45'.

Figure 13. A dialogue design problem from Dialogue T14a. The violated principle is SP8: Provide sufficient task domain knowledge and inference.

S: S: Are you particularly interested in discount? U: Yes. S: At which date will the journey start? U: Monday. S: Monday January 16. At which time? U: 7:20. S: There is no departure at 7:20. The closest other red departure is at 10:50. Do you want this departure?.
D: There is, in fact, a departure at 7:20 but it does not offer discount. The system implicitly assigns priority to discount over departure time without sufficient reason.
C: S should ask U about priority, e.g.: "7:20 is not a discount departure. Red discount can be obtained on the departures at x, y and z. Which departure do you want?"

Figure 14. A dialogue design problem from Dialogue G14a. The violated principle is SP10: Initiate clarification meta-communication in case of inconsistent user input.

S: U: Around 7 o'clock. S: 19 o'clock. At which date will the return journey start? U: Change.
D: All departure times which may be understood as being before noon (am), e.g. 7 o'clock, are automatically understood as such. If there is no flight, the dialogue description automatically adds 12 hours to the departure time. If there is a flight at this time, input will be interpreted as being pm, e.g. 19 o'clock.
C: Ask the user for clarification in case of ambiguous temporal input which cannot be resolved by context.

Figure 15. A dialogue design problem from Dialogue G32a. The violated principle is SP11: Initiate clarification meta-communication in case of ambiguous user input.

Figure 2 shows that, as might be expected, a certain principle was sometimes violated in different ways. This was not the case, however, for GP5, GP7, SP2, SP5, SP6, SP10 and SP11. Representative illustrations of the violations of these principles are shown in Figures 5, 7, 9, 11, 12, 14 and 15, respectively. The principles which were violated in different ways were the following:

GP1 *(make your contribution as informative as is required for the current purposes of the exchange)* was violated in three different ways. An example of how the system withholds requested information is shown in Figure 3. Figure 12 shows the effects of a final question which is too open, and Figure 14 shows a case in which the system withholds important (non-requested) information.

GP6 *(avoid obscurity of expression)* was violated in two ways. Figure 6 exemplifies the most common violation. The second violation (one case only) was as follows. The system produced the output: "There is a departure at 9:10 and 11:50 sold out." This output is due to incorrect system grammar. The intended meaning is that only the 9:10 departure has free seats whereas the departure at 11:50 is already fully booked.

GP10 *(inform the dialogue partners of important non-normal characteristics which they should take into account in order to behave cooperatively in dialogue. Ensure the*

feasibility of what is required of them) was violated in four different ways: asking to change something through comments rather than through the authorised keyword 'change', cf. Figure 8; asking questions, cf. Figure 12; answering several questions at a time, often through providing two temporal expressions in the same utterance, such as date and hour of departure; and providing indirect responses, such as answering 'cheap' to the question of hour of departure. The most frequent violations were changes through comments and answering several questions at a time.

SP4 *(provide clear and comprehensible communication of what the system can and cannot do)* violations were mainly of the type exemplified in Figure 10, i.e. phatic expressions indicating that the user agrees to what the system is simultaneously saying. It does not matter that the system does not listen to such phatic expressions. In a few cases, however, users tried to make corrections while the system was speaking. Typically, users discovered that the system had not heard them but in one case a user did not. This resulted in transaction failure.

SP8 *(provide sufficient task domain knowledge and inference)* violations were mainly of the type illustrated in Figure 13. One case was different. In this case, the user's reply to the system's binary-option question about tickets delivery was understood as 'by mail'. The user then asked for this option to be changed. Instead of simply providing the alternative 'delivered in the airport' as feedback, the system repeated its binary-option question.

3..2 Classifiability and coverage

The large majority of dialogue design problems could be straightforwardly categorised as violations of specific principles. It is only to be expected, however, that some problems are borderline cases which may alternatively be classified as violations of different principles. Figure 6 shows an example which was categorised as a violation of GP6 *(avoid obscurity of expression).* Arguably, this example may instead be considered a violation of GP3 (do not say what you believe to be false). Obscurity and falsehood can be difficult to distinguish from one another.

The user test confirmed the broad coverage of the principles with respect to cooperative spoken user-system dialogue. Only three additions had to be made to the principles established during WOZ. Two specific principles of meta-communication were added, i.e. SP10 and SP11 in Figure 1, cf. Figures 14 and 15. Since meta-communication had not been simulated during WOZ and the WOZ corpus therefore contained few examples of meta-communication, this came as no surprise.

More interestingly, we had to add a modification to GP10, namely that it *should be feasible* for users to do what they are asked to do. For instance, in its introduction the system asks users to use the keywords 'change' and 'repeat' for meta-communication purposes and to answer the system's questions briefly and one at a time. Despite the introduction, a significant number of violations of those instructions occurred in the user test. Users asked questions (Figure 12), provided indirect answers, answered several questions at a time and attempted to make changes through full-sentence expressions rather than by saying 'change' (Figure 8). Almost all of these cases led to misunderstanding or non-understanding. These violations of clear system instructions were first categorised as user errors. However, upon closer analysis they were re-categorised as system problems. We believe the reason for those unwanted user behaviours to be the following. Although the system has clearly stated that it has some non-normal characteristics due to which users should modify their natural dialogue behaviour, this is not cognitively possible for many users. In an extreme example: had we asked users to always use exactly four words in their responses to the system's questions, this would clearly have been a cognitively impossible demand on users. Similarly, what the system's introduction asks users to do turns out to be unrealistic given the dialogue behaviour that is natural to most people.

3.3 Non-violated principles

Eight of the 24 principles were not found violated in the user test. Figure 16 explores why. Most of the principles in question are either very easy to follow during dialogue design once it has been decided to follow them (SP1, SP3, SP9); or it is difficult to tell from observed cooperativity

PRINCIPLE	COOPERATIVITY PROBLEM	COMMENTS
GP2	System provides more information than required.	Difficult to test through identified cooperativity problems.
GP4	System provides information for which it lacks evidence.	The system cannot directly commit this error. The SP10 and SP11 problems indirectly raise issues of this kind.
GP8	System is too verbose.	Difficult to test through identified cooperativity problems.
GP9	System provides disorderly discourse.	Great care taken during dialogue design.
SP1 (GP1)	System is not fully explicit in communicating to users the commitments they have made.	Easy to ensure once it has been decided to follow SP1.
SP3 (GP7)	System does not provide same formulation of the same question to users everywhere in its dialogue turns.	Easy to ensure once it has been decided to follow SP3.
SP7 (GP11)	System does not separate when possible between the needs of novice and expert users.	Difficult to test through identified cooperativity problems.
SP9 (GP13)	System does not initiate repair when it has failed to understand the user.	Repair ability is easy to provide once it has been decided to follow SP9.

Figure 16. Why some principles were not found violated in the user test.

problems whether or not they have been violated because they must be massively violated for a cooperativity problem to occur (GP2, GP8, SP7). With respect to non-massive violations, users tend to suffer in silence during the dialogue and complain afterwards. An example of this was found in the WOZ experiments. The fact that GP2 *(do not make your contribution more informative than is required)* and GP8 *(be brief)* had been violated became apparent from users' complaints that the system talked too much. The problem was solved by removing superfluous information and constructing briefer system utterances.

4. User Errors

Not everything that goes wrong during user-system dialogue happens because of errors made by the dialogue designers. Users also make errors. Some of the user error types found in the user test corpus, such as scenario misunderstandings, have limited real-life significance and several of them cannot be prevented, such as slips and thinking-aloud. In particular two types of error, however, were sources of severe miscommunication. These errors occurred when users ignored system feedback and when they responded to a question different from the clear question posed by the system. Although such errors cannot be completely avoided, their number may be reduced by making subjects pay more attention to the system's feedback and questions. For a full account of user errors in the user test see (Bernsen, Dybkjær, & Dybkjær 1996b).

5. Conclusion

Two further lines of investigation must be pursued in order to test and improve the completeness and practical usefulness of the presented principles of cooperative dialogue design. First, it cannot be excluded at this stage that the principles are somehow tied to the task domain and dialogue complexity of our particular SLDS. Analysis of dialogue problems caused by systems of different dialogue complexity or which address different task domains may reveal additional specific or even generic principles as well as flaws in the way the current principles have been expressed. Secondly, principles of cooperative dialogue are not necessarily the same as practically applicable design guidelines. We therefore need to investigate how the cooperative principles can be made to work as guidelines in dialogue design practice.

Acknowledgements. The Danish dialogue system was developed in collaboration between the Center for Person-Kommunikation at Aalborg University (speech recognition, grammar), the Centre for Language Technology, Copenhagen (grammar, parsing), and the Centre for Cognitive Science, Roskilde University (dialogue and application design and implementation, human-machine aspects). We gratefully acknowledge the support from the Danish Research Councils for the Technical and the Natural Sciences.

References

Aust, H. and Oerder, M. 1995. Dialogue control in automatic inquiry systems. *Proceedings of the ESCA Workshop on Spoken Dialogue Systems*, Vigsø, 121-124.

Bernsen, N.O., Dybkjær, H. and Dybkjær, L. 1995. Exploring the limits of system-directed dialogue. Dialogue evaluation of the Danish dialogue system. *Proceedings of Eurospeech '95*, Madrid, 1457-60.

Bernsen, N.O., Dybkjær, H. and Dybkjær, L. 1996a. Cooperativity in human-machine and human-human spoken dialogue. *Discourse Processes*, Vol. 21, No. 2, 213-236.

Bernsen, N.O., Dybkjær, L. and Dybkjær, H. 1996b. User errors in spoken human-machine dialogue. To appear in *Proceedings of ECAI '96 Workshop on Dialogue Processing in Spoken Language Systems*, Budapest.

Cole, R., Novick, D.G., Fanty, M., Vermeulen, P., Sutton, S., Burnett, D. and Schalkwyk, J. 1994. A prototype voice-response questionnaire for the US Census. *Proceedings of the ICSLP '94*, Yokohama, 683-686.

Dybkjær, H., Bernsen, N.O. and Dybkjær, L. 1993. Wizard-of-Oz and the trade-off between naturalness and recogniser constraints. *Proceedings of Eurospeech '93*, Berlin, 947-50.

Dybkjær, L., Bernsen, N.O. and Dybkjær, H. 1996a. Evaluation of Spoken Dialogues. User Test with a Simulated Speech Recogniser. *Report 9b from the Danish Project in Spoken Language Dialogue Systems*. Roskilde University, February. 3 volumes of 18 pages, 265 pages, and 109 pages, respectively.

Dybkjær, L., Bernsen, N.O. and Dybkjær, H. 1996b. Designing co-operativity in spoken human-machine dialogue. To appear in *Proceedings from the Second Workshop on Human Comfort and Security*, Springer Research Report.

Eckert, W., Nöth, E., Niemann, H. and Schukat-Talamazzini, E. 1995. Real users behave weird - Experiences made collecting large human-machine-dialog corpora. *Proceedings of the ESCA Workshop on Spoken Dialogue Systems*, Vigsø, 193-196.

Fraser, N.M. and Gilbert, G.N. 1991. Simulating speech systems. *Computer Speech and Language* 5, 81-99.

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