A computational model of multi-modal grounding for human robot interaction

Shuyin Li, Britta Wrede, and Gerhard Sagerer
Applied Computer Science, Faculty of Technology
Bielefeld University, 33594 Bielefeld, Germany
shuyinli, bwrede, sagerer@techfak.uni-bielefeld.de

Abstract

Dialog systems for mobile robots operating in the real world should enable mixed-initiative dialog style, handle multi-modal information involved in the communication and be relatively independent of the domain knowledge. Most dialog systems developed for mobile robots today, however, are often system-oriented and have limited capabilities. We present an agent-based dialog model that are specially designed for human-robot interaction and provide evidence for its efficiency with our implemented system.

1 Introduction

Natural language is the most intuitive way to communicate for human beings (Allen et al., 2001). It is, therefore, very important to enable dialog capability for personal service robots that should help people in their everyday life. However, the interaction with a robot as a mobile, autonomous device is different than with many other computer controlled devices which affects the dialog modeling. Here we want to first clarify the most essential requirements for dialog management systems for human-robot interaction (HRI) and then outline state-of-the-art dialog modeling approaches to position ourselves.

The first requirement results from the *situatedness* (Brooks, 1986) of HRI. A mobile robot is situated "here and now" and cohabits the same physical world as the user. Environmental changes can have massive influence on the task execution. For example, a robot should fetch a cup from the kitchen but the door is locked. Under this circumstance the dialog system *must* support mixed-initiative dialog style to receive user commands on the one side and to report on the perceived environmental changes on the other side. Otherwise the robot had to break up the task execution and there is no way for the user to find out the reason.

The second challenge for HRI dialog management is the *embodiment* of a robot which changes the way of interaction. Empirical studies show that the visual access to the interlocutor's body affects the conversation in the way that non-verbal behaviors are used as communicative signals (Nakano et al., 2003). For example, to refer to a cup that is visible to both dialog partners, the speaker tends to say "this cup" while pointing to it. The same strategy is considerably ineffective during a phone call. This example shows, an HRI dialog system must account for multi-modal communication.

The third, probably the unique challenge for HRI dialog management is the implication of the learning ability of such a robot. Since a personal service robot is intended to help human in their individual household it is impossible to hard-code all the knowledge it will need into the system, e.g., where the cup is and what should be served for lunch. Thus, it is essential for such a robot to be able to learn new knowledge and tasks. This ability, however, has the implication for the dialog system that it can not rely on comprehensive, hard-coded knowledge to do dialog planning. Instead, it must be designed in a way that it has a loose relationship with the domain knowledge.

Many dialog modeling approaches already exist. McTear (2002) classified them into three main types: *finite state-based*, *frame-based*, and *agent-based*. In the first two approaches the dialog structure is closely coupled with pre-defined task steps and can therefore only handle well-structured tasks for which one-side led dialog styles are sufficient. In the agent-based approach, the com-

Proceedings of the 7th SIGdial Workshop on Discourse and Dialogue, pages 153–160,
Sydney, July 2006. ©2006 Association for Computational Linguistics

munication is viewed as a *collaboration between two intelligent agents*. Different approaches inspired by psychology and linguistics are in use within this category. For example, within the TRAINS/TRIPS project several complex dialog systems for collaborative problem solving have been developed (Allen et al., 2001). Here the dialog system is viewed as a conversational agent that performs communicative acts. During a conversation, the dialog system selects the communicative goal based on its current belief about the domain and general conversational obligations. Such systems make use of communication and domain model to enable mixed-initiative dialog style and to handle more complex tasks. In the HRI field, due to the complexity of the overall systems, usually the finite-state-based strategy is employed (Matsui et al., 1999; Bischoff and Graefe, 2002; Aoyama and Shimomura, 2005). As to the issue of multi-modality, one strand of the research concerns the fusion and representation of multimodal information such as (Pfleger et al., 2003) and the other strand focuses on the generalisation of human-like conversational behaviors for virtual agents. In this strand, Cassell (2000) proposes a general architecture for multi-modal conversation and Traum (2002) extends his information-state based dialog model by adding more conversational layers to account for multi-modality.

In this paper we present an agent-based dialog model for HRI. As described in section 2, the two main contributions of this model are the new modeling approach of Clark's grounding mechanism and the extension of this model to handle multimodal grounding. In section 3 we outline the capabilities of the implemented system and present some quantitative evaluation results.

2 Dialog Model

We view a dialog as a collaboration between two agents. Agents are subject to common conversational rules and participate in a conversation by issuing multi-modal contributions (e.g., by saying something or displaying a facial expression). In subsection 2.1 we show how we handle conversational tasks by modeling the conversational rules based on grounding and in subsection 2.2 we present how we model individual contributions to tackle the issue of multi-modality. In subsection 2.3 we put these two things together to complete the model description. In this section, we also put concrete examples from the robot domain to clarify the relatively abstract model.

2.1 Grounding

One of the most influential theories on the collaborative nature of dialog is the common ground theory of Clark (1992). In his opinion, agents need to coordinate their mental states based on their mutual understanding about the current tasks, intentions, and goals during a conversation. Clark termed this process as *grounding* and proposed a contribution model. In this model, "contributions" from conversational agents are considered to be the basic component of a conversation. Each contribution has two phases: a *Presentation* phase and an *Acceptance* phase. In the Presentation phase the speaker presents an utterance to the listener, in the Acceptance phase the listener issues an evidence of understanding to the speaker. The speaker can only be sure that the utterance she presented previously has become a part of their common ground if this evidence is available.

Although this well established theory provides comprehensive insight into human conversation two issues in this theory remain critical when being applied to model dialog. The first one is the recursivity of Acceptance. Clark claimed, since everything said by one agent needs to be understood by her interlocutor, each Acceptance should also play the role of Presentation which needs to be accepted, too. The contributions are thus to be organized as a graph. However, this implies that the grounding process may never really end (Traum, 1994). The second critical issue is taking contributions as the most basic *grounding units*. In Clark's view, the basic grounding unit, i.e., the unit of conversation at which grounding takes place, is the contribution. To provide Acceptance for a contribution agents may need to issue clarification questions or repair. But when modeling a dialog, especially a task-oriented dialog, it is hard to map one single contribution from one agent to a domain task since tasks are always cooperately done by the two agents (Cahn and Brennan, 1999). Traum (1994) addressed the first issue by introducing a finite-state based grounding mechanism and Cahn and Brennan (1999) used "exchanges"' as the basic grounding unit to tackle the second critical issue. We combine the advantages of their work and present a grounding mechanism based on an augmented push-down automaton as described below.

Basic grounding unit: As Cahn and Brennan we take *exchange* as the most basic grounding unit. An exchange is a pair of contributions initiated by the two conversational agents. They represent the idea of *adjacency pairs* (Schegloff and Sacks, 1973). The first contribution of the exchange is the Presentation and the second contribution is the Acceptance, e.g., if one asks a question and the other answers it, then the question is the Presentation and the answer is the Acceptance. In our model, a contribution only represents *one* speech act. For example, if an agent says "Hello, my name is Tom, what is your name?" this utterances is segmented into three Presentations (a greeting, a statement, and a question) although they occur in one turn. These three Presentations initiate three exchanges and each of them needs to be accepted by the interlocutor.

Changing status of grounding units: Also as proposed by Cahn and Brennan, an exchange has two states: *not (yet) grounded* and *grounded*. An exchange is grounded if the Acceptance of the Presentation is available. Note, the Acceptance can be an implicit one, e.g, in form of "continued attention" in Clark's term. Taking the example above, the other agent would reply "Hello, my name is Jane." without explicitely commenting Tom's name, yet the three exchanges that Tom initiated were all accepted.

Organization of grounding units: In accordance with Traum we do not think that the Presentation of one exchange should play the role of the Acceptance of its previous exchange. Instead, we organize exchanges in a stack. The stack represents the whole ungrounded discourse: ungrounded exchanges are pushed onto it and the grounded ones are popped out of it. One major question of this representation is: *What has the grounding status of individual exchange to do with the grounding status of the whole stack?* Jane's Acceptance of Tom's greeting has no apparent relation to the remaining two still ungrounded exchanges initiated by Tom. But in the *center embedding* example in Fig. 1, the Acceptance of B1 (utterance A2) contributes to the Acceptance of A1 (utterance B2). These examples show that the grounding status of the whole discourse depends on (1) the grounding status of the individual exchanges and (2) the relationship between these exchanges, the *grounding relation*. These relations are introduced by the Presentation of each exchange because they start an exchange. We identified 4 types of grounding relations: *Default, Support*, *Correct*, and *Delete*. In the following we look at these relations in more detail and refer to exchanges with relation *x* to its *immediately preceding exchange* (IPE) as "*x* exchange", e.g., Support exchange:

Default: The current Presentation introduces a new account that is independent of the previous exchange in terms of grounding, e.g., what Tom said to Jane constructs three Presentations that initiate three default exchanges. Such exchanges can be grounded independently of each other.

Support: If an agent can not provide Acceptance for the given Presentation she will initiate a new exchange to support the grounding process of the ungrounded exchange. A typical example of such an exchange is a clarification question like "I beg your pardon?". If a Support exchange is grounded its initiator will try to ground the IPE again with the newly collected information through the supporting exchange.

Correct: Some exchanges are created to correct the content of the IPE, e.g., in case that the listener misunderstood the speaker and the speaker corrects it. Similar to Support, after such an exchange is grounded its IPE is updated with new information and has to be grounded again.

Delete: Agents can give up their effort to build a common ground with her interlocutor, e.g., by saying "Forget it.". If the interlocutor agrees, such exchanges have the effect that all the ungrounded exchanges from the initial Default exchange up to the current state are no longer relevant and the agents do not need to ground them any more.

Note, once an exchange is grounded it is *immediately* removed from the stack so that its IPE becomes the IPE of the next exchange. This model is described as an augmented push-down automaton (Fig. 2). It is augmented in so far that transitions can trigger actions and a variable number of exchanges can be popped or pushed in one step. There are five states in this APDA and they represent the fact what kind of ungrounded exchange is on the top of the stack. Along the arrows that connect the states the input (denoted as I), the resulting stack operation (denoted as S) and the possible action that is triggered (denoted as A) are given. The input of this automaton includes Presentation (e.g., "defaultP" stands for "Default Presentation") and Acceptance.

A1: What do you think about Mr. Watton?
B1: Mr. Watton? our music teacher?
A2: Yes. (accept B1)
B2: Well, he is OK. (accept A1)

Figure 1: An example of center embedding

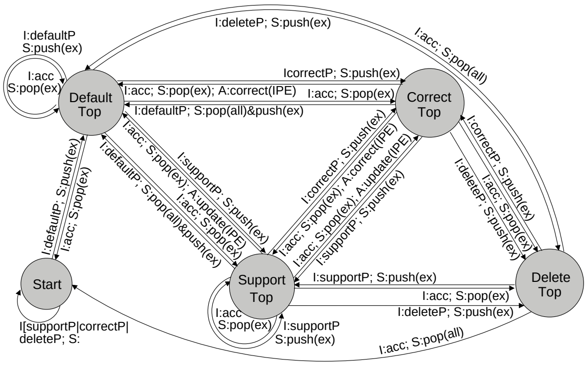


Figure 2: Augmented push-down automaton for grounding (ex: exchange)

As long as there is an ungrounded exchange at the top of the stack, the addressee will try to ground it by providing Acceptance, unless its validity is deleted. For the reason of space, we only explain the APDA with the center embedding example in Fig. 1. Contribution A1 introduces a question into the discourse which initiates a Default exchange, say Ex1. This exchange is pushed onto the stack. Instead of providing Acceptance to A1, contribution B1 initiates a new exchange, say Ex2, with grounding relation Support to Ex1 and is pushed onto the stack. Then contribution A2 acknowledges B1 so that Ex2 is grounded and popped out of the stack. The top element of the stack is now the ungrounded Ex1. Since Ex2 supported Ex1, the Ex1 is updated with the information contained in Ex2 (The music teacher was meant) and B2 then successfully grounds this updated Ex1.

In our model, every exchange can be individually grounded and contributes to the grounding of the whole ungrounded discourse by acting on the IPE according to their grounding relations. This way we can organize the discourse in a sequence without losing the local grounding flexibility. For an implemented system, this means that both the user and the system can easily take initiative or issue clarification questions. To implement this model, however, two points are crucial. The first one is the recognition of the user's contribution type: for every user contribution, the dialog system needs to decide whether it is a Presentation or an Acceptance. If it is a Presentation, the system needs further to decide whether it initiates a new account, corrects or supports the current one, or deletes it. This issue of intention recognition is a classical challenge for dialog systems. We present our solution in section 3. The second point is that the dialog system needs to know when to create an exchange of certain grounding relation by generating an appropriate Presentation and when to create an Acceptance. For that we need to first look at the structure of individual contributions more closely in the next subsection.

2.2 The structure of agents' contributions

To represent the structure of the individual contributions we take into account the whole language generation process which enables us to come up with a powerful solution as described below.

The layers of a contribution: What we can observe in a conversation are only exchanges of agents' contributions in verbal or non-verbal form. But in fact the contributions are the end-product of a complex cognitive process: language production. Levelt (1989) identified three phases of language production: *conceptualization*, *formulation*, and *articulation*. The production of an utterance starts from the conception of a *communicative intention* and the semantic organization in the conceptualization phase before the utterance can be formulated and articulated in the next two phases. Intentions can arise from the previous discourse or from other motivations such as needs for help or information. This finding motivates us to set up a two-layered structure of contributions. One layer is the so-called *intention layer* where communication intentions are conceived. For a robot the communication intentions come from the analysis of the previous discourse or from the robot control system. The other layer is the *conversation layer*. The communication intentions are formulated and articulated here[1]. These two layers represent the intention conception and the language generation process, respectively. We term this two-layered structure of contribution *interaction unit* (IU).

The issue of multi-modality: Face-to-face conversations are multi-modal. Speech and body language (e.g., gesture) can happen simultaneously. McNeill (1992) stated that gesture and speech arise from the same semantic source, the

[1] Since most robot systems use speech synthesizer to generate acoustic output which replaces the articulation process, only formulation is performed on this layer.

so-called “idea unit” and are co-expressive. Since semantic representation is created out of communicative intentions (Levelt, 1989) we assume the communication intentions are the modality independent base that governs the multi-modal language production. We, therefore, extend our structure above by introducing two generators on the conversation layer: one *verbal* and one *non-verbal* generator that represent the verbal and non-verbal language generation mechanism based on the communication intentions created on the intention layer. The relationship between these two generators is variable. For example, Iverson et al. (1999) identified three types of *informational* relationship between speech and gesture: *reinforcement* (gesture reinforces the message conveyed in speech, e.g., emphatic gesture), *disambiguation* (gesture serves as the precise referent of the speech, e.g., deictic gesture accompanying the utterance “this cup”), and *adding-information* (e.g., saying “The ball is so big.” and shaping the size with hands). In our work, when processing users’ multi-modal contributions we focus on the disambiguation relation; when creating multimodal contributions for the robot we are also interested in other informational relations [2]. The structure of an IU is illustrated in Fig. 3.

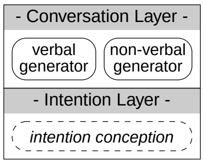


Figure 3: IU

Operation flow within an interaction unit: During a conversation an agent either initiates an account or replies to the interlocutor’s account. The communication intentions can thus be *self-motivated* or *other-motivated*. For a robot, self-motivated intentions can be triggered by the robot control system, e.g., observed environmental changes. In this case, an IU is created with its intention layer importing the message from the robot control system and exporting an intention. This intention is transfered to the conversation layer which then formulates a verbal message with the verbal generator and/or constructs a body language expression with the non-verbal generator. Other-motivated intentions can be triggered by the needs of the on-going conversation, e.g., the need to answer a question, or be triggered by robot’s execution results of the tasks specified previously by the user. The operation flow is similar to that of the self-motivation apart from the fact that, in case of intentions motivated by conversational needs, the intention layer of the IU does not import any robot control system message but creates an intention directly. Note, the IUs that are initiated by the robot and by the user have identical structure. But in case of user initiated IUs we do not make any assumption of their underlying intention building process and the intention layer of their IUs are thus always empty.

With the IUs, we can integrate the non-verbal behavior systematically into the communication process and model multi-modal dialog. Although it is not the focus of our work, our model can also handle purely non-verbal contributions, since the verbal generator does not always need to be activated if the non-verbal generator already provides enough information about the speaker’s intention. Possible scenarios are: the user looks tired (presentation) and the robot offers “I can do that for you.” (acceptance) or the user says something (presentation) and robot nods (acceptance).

2.3 Putting things together

Till now we have discussed our concept of using a grounding mechanism to organize contributions and of representing individual contributions as IU. Now it is time to look at the still open point at the end of the section 2.1: when to create an IU as Presentation and when an IU as Acceptance.

Self-motivated intentions usually trigger the creation of an IU as Presentation with Default relation to its IPE. For example, if the robot needs to report something to the user it can create a Default exchange by generating an IU as its Presentation. The user is then expected to signal her Acceptance. Other-motivated intentions can, according to the context, result in either Presentation or Acceptance. To make the correct decision we developed criteria based on the *joint intention theory* of Levesque et al. (1990) which predicts that during a collaboration the partners are committed to a joint goal that they will always try to conform till they reach the goal or give up. Note, this does not mean that one will always agree with her interlocutor, but they will behave in the way that they think is the best to achieve the goal. This theory can be applied to human-robot dialog in a twofold sense: Firstly, a dialog can be generally seen as a collaboration as Clark proposed. Secondly, the human-robot dialog is mostly task-oriented, i.e.,

[2] This policy has a practical reason: it is much more difficult in computer science to correctly recognize and interpret human motion than to simulate it.

the human and the robot work towards the same goal. With this theory in mind we describe how we process other-motivated contributions below.

The precondition of language production based on other-motivated intentions is language perception. Before reacting, i.e., before creating her own IU, an agent first needs to understand the intention conveyed by her interlocutor's IU by studying its conversation layer. Since we focus on disambiguation function of non-verbal behavior we assume that agents first study the generated verbal information, if the intention can not be fully recognized here, one will further study the information provided by the non-verbal generator (e.g., a gesture) and fuse the verbal and non-verbal information. If the intention recognition is still unsuccessful, the agent can not provide Acceptance for the given IU. If she is still committed to the dialog she will issue a clarification question, i.e., she generates an IU as Presentation that initiates a Support exchange to the current ungrounded exchange. If the intention of her interlocutor is successfully recognized the language perception process ends and the agent tries to create her own IU. As described in subsection 2.2 the creation of the IU starts from the creation of an intention on the intention layer. In case of a robot, the dialog system accesses the robot control system and awaits its reaction to the conveyed information (e.g., a user instruction). Usually, a robot is designated to do something for the user, i.e., the robot is committed to the goal proposed by the user, so we define *the robot can only provide acceptance if the task is successfully executed*. In this case, the robot completes the current IU with the filled intention layer by generating an confirmation on its conversation layer. Afterwards, this grounded exchange can be popped from the stack. If the robot can not execute the task for some reasons, then the current exchange can not be grounded and the robot will take the current IU with the filled intention layer as another Presentation that initiates a Support or Correct exchange to the current ungrounded exchange, similar as the case in Fig. 1. The conversation layer of this IU can thus formulate something like "Sorry, I can't do that because..." and present a sorrowful face. This new Support or Correct exchange is pushed onto the stack. Figure 4 illustrates this process as a UML activity diagram.

In our model we only do general conversational planning instead of domain specific task planning.

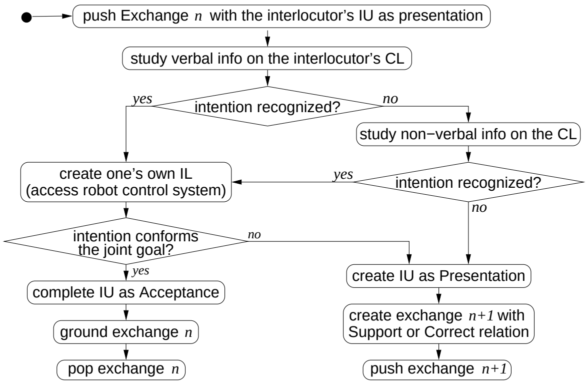

push Exchange *n* with the interlocutor's IU as presentation
study verbal info on the interlocutor's CL
yes
intention recognized?
no
study non−verbal info on the CL
create one's own IL
(access robot control system)
yes
intention recognized?
no
intention conforms
the joint goal?
no
yes
complete IU as Acceptance
create IU as Presentation
ground exchange *n*
create exchange *n+1* with
Support or Correct relation
pop exchange *n*
push exchange *n+1*

Figure 4: Handling other-motivated contribution (CL: Conversation layer; IL: Intention Layer)

What the dialog system needs to know from the robot control system is what processing results it can produce. The association of these results with robot intentions in terms of whether they start a new account, support or correct one, or delete it, can be configured externally and thus easily updated or replaced. Based on this configuration IUs are generated that operate according to the grounding mechanism as described in section 2.1.

3 Implementation

This dialog model was implemented for our robot BIRON, a personal robot with learning abilities. It can detect and follow persons, focus on objects (according to human deictic gestures) and store collected information into a memory. Our implementation scenario is the so-called *home tour*: a user shows a new robot her home to prepare it for future tasks. The robot should be able to learn and remember features of objects that the user mentions and it "sees", e.g., name, color, images etc. Besides, our system was also successfully ported to a humanoid robot BARTHOC for studies of emotional and social factors of HRI (see. Fig. 5).

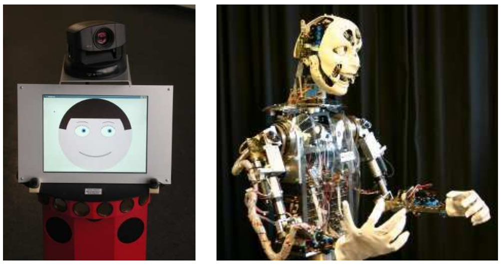
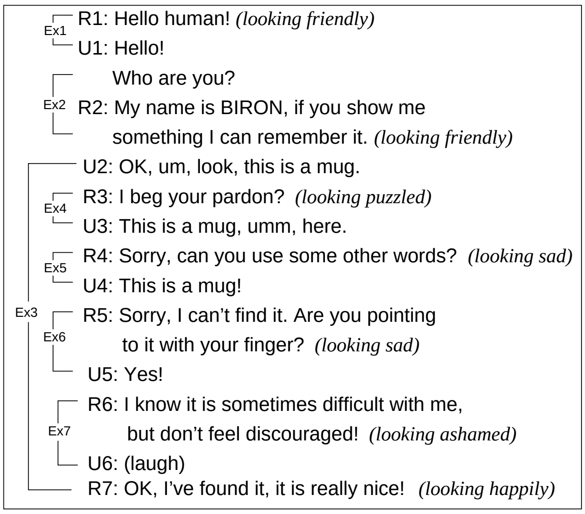


Figure 5: Robots BIRON and BARTHOC

The dialog manager is linked to a speech understanding system which transforms parts of speech from a speech recognizer into a speech-act-based form. To recognize the user's intention, the dialog system classifies this input into 10 categories of three groups according to heuristics, e.g., *instruction*, *description*, and *query* initiate new tasks and thus a new Default exchange; *deletion* and *correction* initiate Delete and Correct exchanges that are related to early exchanges; and *confirmation*, *negation*, etc. can only be responses and are, therefore, viewed as user's Acceptance of exchanges that the robot initiates. The main part of the dialog system is the Dialog Manager that carries out grounding analysis and stack operations. The Robot Control Manager receives messages from the robot control system and calls the Dialog Manager to do relevant operations. This dialog system enables multi-modal, mixed-initiative dialog style and can handle complex repair behaviors. With the example dialog in Fig. 6, which is transcripted from a real interaction session between a user and BIRON, we discuss the two most important features of our system.



R1: Hello human! *(looking friendly)*
U1: Hello!
Who are you?
R2: My name is BIRON, if you show me something I can remember it. *(looking friendly)*
U2: OK, um, look, this is a mug.
R3: I beg your pardon? *(looking puzzled)*
U3: This is a mug, umm, here.
R4: Sorry, can you use some other words? *(looking sad)*
U4: This is a mug!
R5: Sorry, I can't find it. Are you pointing to it with your finger? *(looking sad)*
U5: Yes!
R6: I know it is sometimes difficult with me, but don't feel discouraged! *(looking ashamed)*
U6: (laugh)
R7: OK, I've found it, it is really nice! *(looking happily)*

Figure 6: A dialog example with the extrovert BIRON. (U: user, R: robot, Ex: Exchange)

Taking Initiative and robot personality: Initiatives that a dialog system can take often depends on its back-end application. Since BIRON does not have a task planner which would be ideal to demonstrate this ability we implemented an *extrovert* personality for it (additionally to its *basic* personality) that takes communication-related initiatives. The basic BIRON behaves in a rather passive way and only says something when addressed by the user. In contrast, the extrovert BIRON greets persons actively (R1 in Table 6) and remarks on its own performance (R6). When the robot control system detects a person the dialog system initiates a Default exchange to greet her. BIRON can also measure its own performance by counting the number of Support exchanges it has initiated for the current topic. Since the Support exchanges are only created if BIRON can not provide Acceptance to the user's Presentation (because it does not understand the user or it can not execute a task), the amount of the Support exchanges thus has direct correlation to the robot's overall performance. On the other hand, the more Default exchanges there are, the better is the performance because the agents can proceed to another topic only if the current one is grounded (or deleted). Based on this performance indication BIRON does remarks to motivate users.

Resolving multi-modal object references: It happens quite frequently in the home tour scenario that the user points to some objects and says "This is a *z*". BIRON needs to associate its symbolic name (and eventually other features) mentioned by the user with the image of the object. The resolution of such multi-modal object references (U4-R7 in Table 6) is solved as following: the Dialog Manager creates an IU for the user-initiated utterance (e.g., "this is a cup") and studies the verbal and non-verbal generator on its conversation layer. In the verbal generator, what the pronoun "this" refers to is unclear, but it indicates that the user might be using a gesture. Therefore, the Dialog Manager further studies the non-verbal generator. The responsible robot vision module is activated here to search for a gesture and to identify the object cup. If the cup is found in the scene, this module assigns an ID to the image and stores it in the memory. After the Dialog Manager receives this ID, the processing of the conversation layer of the user IU ends, the Dialog Manager proceeds to create its own IU to react to the user's IU. Problems with the object identification indicate failure of the intention recognition process on the user conversation layer. In this case, the Dialog Manager creates a Support exchange to ask the user which object she refers to and retries it if she does not oppose (R5-R7). This process and the associated multimodality fusion and representation are described in (Li et al., 2005) in detail.

The evaluation of dialog systems for human robot interaction is still an open issue. A robot system is usually a complex system including a large number of modules that claim plenty of processing time or are subject to environmental conditions. For the dialog system, this means that the correct interpretation and transaction of user utterances is by no means a guaranty for a prompt response or successful task execution. Thus, the performance of the dialog system can not be directly measured with the performance of the overall system like most desktop dialog applications. We are still working at evaluation metrics for HRI dialog systems (Green et al., 2006). But the efficiency of our system is already visible in the small effort associated with the porting of this system to another robot platform and in the pilot user study with BIRON. In this study, each of the 14 users interacted with BIRON twice. In the total 28 runs the dialog system generated 903 exchanges for the 813 user utterances. Among these exchanges, 34% initiated clarification questions. This result correlated with the evaluation result of our speech understanding system which fully understood 65% of all the user utterances. 18.6% of the exchanges were Support exchanges created due to execution failure of the robot control system which corresponds to the performance of the robot control system. The average processing time of the dialog system was 11 msec.

4 Conclusion

In this paper we presented an agent-based dialog model for HRI. The implemented system enables multi-modal, mixed-initiative dialog style and is relatively domain independent. The real-time testing of the system proves its efficiency. We will work out detailed evaluation metrics for our system to be able to draw more general conclusion about the strength and weakness of our model.

References

J. Allen, D. K. Byron, M. Dzikovska, G. Ferguson, L. Galescu, and A. Stent. 2001. Towards conversational human-computer interaction. *AI Magazine*, 22(4).

K. Aoyama and H. Shimomura. 2005. Real world speech interaction with a humanoid robot on a layered robot behavior control architecture. In *Proc. Int. Conf. on Robotics and Automation*.

R. Bischoff and V. Graefe. 2002. Dependable multimodal communication and interaction with robotic assistants. In *Proc. Int. Workshop on Robot-Human Interactive Communication (ROMAN)*.

R. A. Brooks. 1986. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, 2(1):14–23.

J. E. Cahn and S. E. Brennan. 1999. A psychological model of grounding and repair in dialog. In *Proc. Fall 1999 AAAI Symposium on Psychological Models of Communication in Collaborative Systems*.

J. Cassell, T. Bickmore, L. Campbell, and H. Vilhjalmsson. 2000. Human conversation as a system framework: Designing embodied conversational agents. In J. Cassell, J. Sullivan, S. Prevost, and E. Churchill, editors, *Embodied conversational agents*. MIT Press.

H. H. Clark, editor. 1992. *Arenas of Language Use*. University of Chicago Press.

A. Green, K. Severinson-Eklundh, B. Wrede, and S. Li. 2006. Integrating miscommunication analysis in natural language interface design for a service robot. In *Proc. Int. Conf. on Intelligent Robots and Systems*. submitted.

J. M. Iverson, O. Capirci, E. Longobardi, and M. C. Caselli. 1999. Gesturing in mother-child interactions. *Cognitive Develpment*, 14(1):57–75.

W. Levelt. 1989. *Speaking: From intention to articulation*. Cambridge, MA: MIT Press.

H. J. Levesque, P. R. Cohen, and J. H. T. Nunnes. 1990. On acting together. In *Proc. Nat. Conf. on Artificial Intelligence (AAAI)*.

S. Li, A. Haasch, B. Wrede, J. Fritsch, and G. Sagerer. 2005. Human-style interaction with a robot for cooperative learning of scene objects. In *Proc. Int. Conf. on Multimodal Interfaces*.

T. Matsui, H. Asoh, J. Fry, Y. Motomura, F. Asano, T. Kurita, I. Hara, and N. Otsu. 1999. Integrated natural spoken dialogue system of jijo-2 mobile robot for office services,. In *Proc. AAAI Nat. Conf. and Innovative Applications of Artificial Intelligence Conf.*

D. McNeill. 1992. *Hand and Mind: What Gesture Reveal about Thought*. University of Chicago Press.

M. F. McTear. 2002. Spoken dialogue technology: enabling the conversational interface. *ACM Computing Surveys*, 34(1).

Y. I. Nakano, G. Reinstein, T. Stocky, and J. Cassell. 2003. Towards a model of face-to-face grounding. In *Proc. Annual Meeting of the Association for Computational Linguistics*.

N. Pfleger, J. Alexandersson, and T. Becker. 2003. A robust and generic discourse model for multimodal dialogue. In *Proc. 3rd Workshop on Knowledge and Reasoning in Practical Dialogue Systems*.

E. A. Schegloff and H. Sacks. 1973. Opening up closings. *Semiotica*, pages 289–327.

D. Traum and J. Rickel. 2002. Embodied agents for multiparty dialogue in immersive virtual world. In *Proc. 1st Int. Conf on Autonomous Agents and Multi-agent Systems*.

D. Traum. 1994. *A Computational Theory of Grounding in Natural Language Conversation*. Ph.D. thesis, University of Rochester.