A Two-Layer Model for Behavior and Dialogue Planning in Conversational Service Robots

Mikio Nakano, Yuji Hasegawa, Kazuhiro Nakadai
Takahiro Nakamura, Johane Takeuchi
Toyotaka Torii, Hiroshi Tsujino
Honda Research Institute Japan
8-1 Honcho, Wako, Saitama 351-0188, Japan
{nakano, yuji.hasegawa, nakadai, moo,
Johane.Takeuchi, tory, tsujino}@jp.honda-ri.com

Naoyuki Kanda, Hiroshi G. Okuno
Graduate School of Informatics
Kyoto University
Yoshida Honmachi, Sakyo-ku, Kyoto 606-8501, Japan
naoyuki@kuis.kyoto-u.ac.jp
okuno@i.kyoto-u.ac.jp

***Abstract*— This paper presents a model for the behavior and dialogue planning module of conversational service robots. Most of the previously built conversational robots cannot perform dialogue management necessary for accurately recognizing human intentions and providing information to humans. This model integrates robot behavior planning models with spoken dialogue management that is robust enough to engage in mixed-initiative dialogues in specific domains. It has two layers; the upper layer is responsible for global task planning using hierarchical planning and the lower layer engages in local planning by utilizing modules called experts, which are specialized for performing certain kind of tasks by performing physical actions and engaging in dialogues. This model enables switching and canceling tasks based on recognized human intentions. A preliminary implementation of the model, which has been integrated with Honda ASIMO, has shown its effectiveness.**

***Index Terms*— conversational robot, service robot, behavior and dialogue planning, dialogue management**

I. Introduction

One of our goals is to establish a general model for building intelligent conversational robots that can engage in dialogue with humans to understand their requests and give useful information as well as perform desired behaviors. We call such robots *conversational service robots*. We assume that such robots employ layered architecture [1], [2], in the top level of which is the deliberative planning module, or the *behavior and dialogue planning module*. The objective of this paper is to present a model for this module. It takes as input human speech and gesture recognition results, human locations, human emotion recognition results and other information obtained by interpreting sensor output, and generates multi-modal expressions which include texts, gestures and/or actions.

The most important difference between a conversational service robot and a conversational robot that does not engage in service (hereafter a *conversational entertainment robot*) is that the former must engage in *task-oriented dialogues*. The former must accurately understand human request and correctly perform requested tasks, while the latter focuses on more real-time and emotional reactions rather than accuracy in their behaviors. Previous research on conversational robots, however, has not paid much attention to precise communication as in speech and language processing research.

In the context of speech and language processing research, state-of-the-art spoken dialogue system technologies have made it possible to build computer systems that can engage in complicated dialogues consisting of dozens of turns to perform tasks (e.g. [3], [4]). Although incorporating such technologies are expected to make a conversational service robot more usable, it is unclear how to integrate robot architecture with spoken dialogue system architecture.

This paper presents a new model for the behavior and dialogue planning in conversational service robots. It is called *Multi-Expert-based Behavior and Dialogue Planning* (MEBDP). MEBDP integrates intelligent robot architectures with mixed-initiative, multi-domain spoken dialogue system architectures. MEBDP utilizes components called *experts*, which are specialized for performing certain kind of tasks by performing physical actions and engaging in dialogues in certain domains. With MEBDP, robots understand human requests by engaging in spoken dialogue in a specific domain, and then set the goal that satisfies the requests. The goal may be achieved by physical behaviors, information providing using spoken dialogue, or combinations of both. The goals are stored in an agenda so that the robot can sequentially achieve them.

This paper is organized as follows. The next section describes requirements for the behavior and dialogue planning in conversational service robots. Then we mention previous work and explain the details of MEBDP. Next we describe the current implementation of MEBDP, before concluding this paper with mentioning future work.

II. Behavior and Dialogue Planning in Conversational Service Robots

Conversational service robots are dedicated to performing service tasks in a specific environment such as in a house

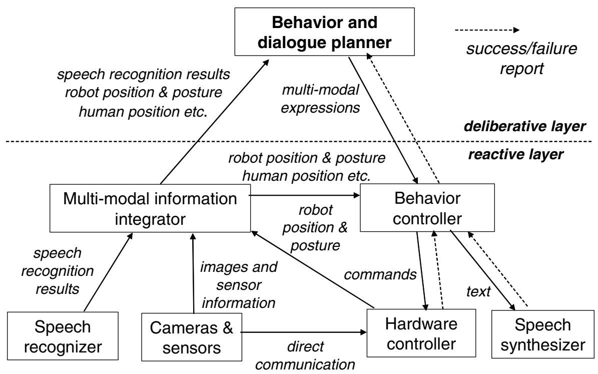

Fig. 1. Architecture for Conversational Service Robots

and in an office. For example, they are supposed to clean up rooms, and collect garbage, and provide weather information. They must be able to engage in dialogues with humans to understand their requests to perform tasks and provide them with some necessary information.

A. Architecture for Service Robots

We assume that the service robots employ conventional multi-layered architecture. Fig. 1 depicts it, although it is simplified. Currently we assume two layers, the upper *deliberative layer* and the lower *reactive layer*.

In the deliberative layer is the behavior and dialogue planning module, on which this paper focuses. It can take as input the recognition results and speaker identification of human utterance and possibly other information from the multi-modal information integrator. At appropriate times, it outputs multi-modal expressions which include texts and physical actions.

The reactive layer interprets and integrates sensor information, and sends it to the deliberative layer. In addition, it behaves according to multi-modal expressions received from the deliberative layer.

B. Requirements

In this paper, we focus on achieving the following features in behavior and dialogue planning module, which we believe is crucial for the better usability of conversational robots.

- *Task-oriented dialogue*
  Unlike conversational entertainment robots, a conversational service robot must accurately understand human requests and provide information by spoken dialogue. Such a dialogue is called task-oriented dialogues. Speech communication in real environments sometimes fails for several reasons such as speech recognition errors and poor quality of synthesized speech. The behavior and dialogue planning module therefore needs the ability to recover from such communication failure.
- *Mixed-initiative dialogue*
  We focused on mixed-initiative dialogues. In mixed-initiative dialogues, not only the system asks humans to specify information required to perform tasks, but also humans can speak in an unrestricted way; for example, humans can ask questions to the system. The dialogue will become inefficient if a conversational robot that performs multiple tasks always starts dialogues with questions to humans. We therefore cannot adopt a system-directed dialogue strategy.
- *Dynamic domain switching*
  Even while engaging in a dialogue in one domain, robots have to switch the domain when humans want to change to another task. If this is not possible, it must be stuck in one domain, and the dialogue cannot proceed.
- *Interruption handling*
  Even while speaking or performing a physical action, robots must react to human interruption utterances. If this is not possible, humans cannot stop or change the robot's actions with speech.

Note that we focus only on human utterances as inputs, and that we do not deal with other kind of inputs such as gesture recognition results and emotion estimation results, although we consider that dealing with such inputs is also important.

III. Previous Work

There have been several pieces of work on building robots that can engage in conversation with humans. They can be classified into two groups.

One group is what we call *conversational entertainment robots*. They are mainly dedicated to entertaining people by chatting with them. Kanda et al.'s robot [5] has situated modules organized in a network. It can perform more than one hundred behaviors including making short utterances. Hoshino et al. presented similar architecture [6]. The behavior in each situated module are relatively simple, and it is not clear whether this architecture enables performing task-oriented dialogues.

The other group is conversational service robots. Recently, some work tried to combine spoken dialogue systems and service robots [7], [8], [9], [10], although they have not addressed how to perform dialogue and behavior planning in an integrated way.

Topp et al.'s robot employs a state-transition model for the highest-level control module, and it has a substate where clarification questions are made when necessary [11]. It is unclear, however, how it can perform domain switching and interruption handling which we desire to achieve.

A robot named *jijo-2* [12] can perform tasks that require physical actions such as delivery as well as engage in task-oriented dialogues in several task domains such as telling office members' current locations and route directions. It can switch tasks and stop navigation based on human utterances. Although it achieves high functionality in robot conversation, it has limitations in that the dialogue management strategies are fixed, and that the dialogue planning is not integrated with global task planning.

IV. MEBDP: Proposed model

A. Incorporating dialogue management

For service robots to understand human requests and providing information through dialogues, they need to perform *dialogue management* [13]. For example, when they understand human questions about weather information, they need to know about the place, the day, information type (such as the weather, the temperature, and the probability of precipitation), and others of the weather information the human wants to obtain. Because not all information is conveyed by only one utterance, they sometimes need to ask humans to fill in missing information. In addition, because there might be speech recognition errors, they need to make a request to confirm some of the recognized information.

Thus far many models and algorithms for dialogue management have been proposed depending on the types of task domains in spoken dialogue system research. Dialogue management techniques are classified according to which kind of representation of *dialogue state* is used. The dialogue state stores the results of user utterance understanding and user intention recognition, dialogue history including system utterances, and all other information concerning the dialogue. Utterance understanding updates the dialogue state according to the language understanding result. The dialogue management component decides the system utterance based on the dialogue state.

For understanding human requests in relatively simple domains, frame-based representations for dialogue states are often used for mixed-initiative dialogues [14], [15]. A frame is a bundle of slots each of which consists of a slot name and a slot value. These slots represent results of user request recognition. For example, a train timetable information system can have slots for departure city, destination city, departure time, and arrival time. In addition to the slot values, grounding information for each slot, and confidence value for each slot can be stored to be used for dialogue management. Here, *grounding information* roughly corresponds to information on whether the information is confirmed or not [16]. Frame-based dialogue management strategies utilize a set of rules that describes which action should be taken given a dialogue state.

Although incorporating such dialogue management techniques into conversational robots seems effective, it is not simple for the following reason. Since service robots need to perform complicated tasks, they need to plan a sequence of subtasks by hierarchical planning. Consider the case where the robot is asked to call person A. The robot needs to find A's location when it does not know, it then needs to go to A's location, and it needs to tell A that he/she is being called. Although sometimes it must replan while performing these subtasks as the situation may change, it is effective to first plan a subtask sequence. This decomposition of tasks into subtasks utilizes plan libraries, which describe how a kind of task can be decomposed. On the contrary, frame-based dialogue management strategies are different in that they do not plan action sequence but choose only the next action. Simply integrating these two types of planning would make the planning process complicated. Since the human speech understanding results are unpredictable and even speech recognition errors may occur, the planner must reconstruct whole plans at the end of each human speech.

B. MEBDP: A Two-Layer Model

The basic idea behind MEBDP is to split the behavior and dialogue planning into two layers. Note that these layers are sub-layers in the deliberative layer in the whole robot architecture. The upper layer, or *task planning layer*, performs global task planning by decomposing requested *tasks* into a sequence of *subtasks* using plan libraries. Here, a task is what humans request the robot to perform.

The lower layer, or *expert action selection layer*, performs subtasks using planners specific to the types of subtasks. We call such planners *experts*. An expert decides what action to perform next until the subtask is achieved or canceled.

The experts which can output actions are called "being in charge." In the current model, for the simplicity of the model, only one expert can be in charge, although we are planning to remove this restriction.

An expert should be prepared for each type of subtask. Any kind of planning framework can be used. If frame-based dialogue management is suitable for a subtask, the expert that performs such dialogue management is used.

C. Types of Experts

In the current model, there are four types of experts.

- *Request understanding experts* (**RU**)
  The subtask performed by these experts is the robot understanding humans' requests so that the robot can provide necessary information or it can behave appropriately. For example, RU in the train timetable information domain can achieve the subtask that the robot understands which information concerning the train timetable the human needs. In most cases, frame-based information state representation is suitable for these experts.
- *Information providing experts* (**IP**)
  These experts perform providing a human with information he/she requested. For example, IP in the train timetable domain accesses an external database to

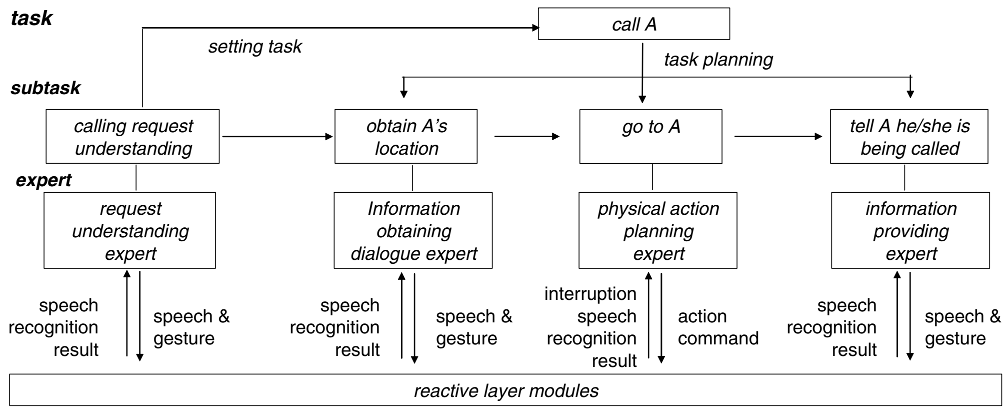

Fig. 2. Change in Subtasks and Experts in the Example

retrieve timetable information requested by the human, and interactively tells it to him/her. It decomposes the information to be provided into small pieces and tells them one by one and checks if the human interrupts. When the human asks for repetition or cancellation, the plan is reconstructed. Hierarchical planning with interruption handling [17] can be used for these experts.

- *Physical action planning experts* (**PAP**)
  These experts work in the same way as IPs except that they plan a physical action sequence, not an utterance sequence. They as well accept human interrupting utterances.
- *Information obtaining dialogue experts* (**IOD**)
  These experts plan dialogues to obtain information necessary for performing a task. For example, one expert in this category performs dialogue management to obtain information about the location of some person or an object. They are similar to RUs in that they obtain information from humans, although the robot mainly asks questions to humans. Frame-based dialogue management is also effective for these kind of experts.

D. Example

Let us consider again the example that a person B asks the robot to call A (Fig. 2). When B speaks to the robot, all the experts try to understand the utterance. Based on those results, the RU expert in the calling domain becomes in charge. It controls the dialogue with the human unless he/she changes the topic. After some exchanges of the utterances, it understands B's request which is set as the goal. Then it plans a subtask sequence, obtaining A's location, approaching A, and telling A that he/she is being called. To obtain A's location, the expert for information obtaining dialogue in the human location domain becomes in charge and engages in the dialogue until A's location is found. Then the PAP expert for moving to A's location becomes in charge. It not only performs moving but also accepts human utterances. If the person who asked to call A tells the robot to cancel calling, it stops moving and goes back. When the robot reaches A, the IP expert in the messaging domain becomes in charge and it tells A that he/she is being called by B.

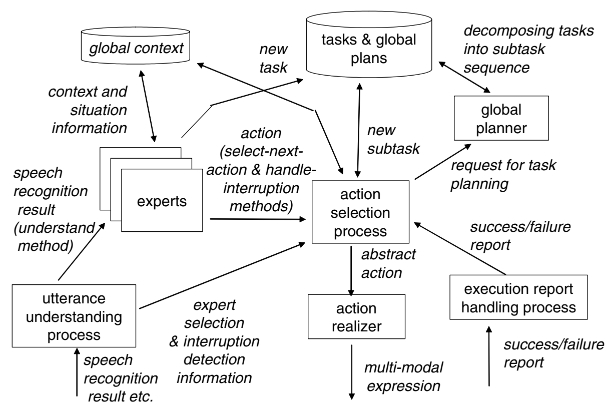

Fig. 3. Modules in MEBDP

E. Model Details

Fig. 3 shows the modules in the MEBDP and their relationship. An expert is a kind of object in the object-oriented programming framework. Each expert has its own internal state, which we call *information state*, as a spoken dialogue system has a dialogue state. In addition to common content of a dialogue state, physical action execution information might be stored in the internal state of experts. Each expert has the following four main methods.

- The *understand* method updates the information state based on the human speech recognition results. This corresponds to context-dependent speech understanding.
- The *select-next-action* method outputs one *abstract action* based on the content of the information state. Here an abstract action is an abstract representation of action

and it consists of the action type and additional attributes in our implementation of MEBDP. Below is an example:

```
[action-type=request-confirmation,
 place=tokyo,
 day=tomorrow,
 information-type=probability-of-precipitation],
```

which will be converted into the system utterance text "Probability of precipitation in Tokyo for tomorrow?" by the action realizer. This method corresponds to dialogue management in the spoken dialogue systems. If it finds that the subtask is completed, this method returns the symbol *finish*. If the expert is an RU, the understanding result is set as a new task. If the expert is an IOD, the obtained information is stored in the global context.

- The *detect-interruption* method determines if the previous human utterance is an interruption to the action being performed when this expert is being in charge.
- The *handle-interruption* method returns the action to be performed after an interruption is detected.

There are three processes that run in parallel, namely, *understanding process*, *action selection process*, and *execution report handling process*.

The understanding process continuously monitors input from the multi-modal information integrator. When it receives a recognition result, it dispatches the result to all experts with the *understand* method. Based on the results of the understanding, it decides which expert should be in charge next, as is done in multi-domain spoken dialogue systems [18]. When an action is being performed, it calls the *detect-interruption* method of the selected expert, and tells the action selection process if the utterance is an interruption.

The action selection process iterates the following procedure.

1) It first checks if there is an expert in charge. If not, it waits for new human utterance to be understood.
2) Otherwise, it calls the expert's *select-next-action* method.
3) Unless *finish* is returned, it sends the returned abstract action to the action realizer, which transforms the abstract action into a multi-modal expression, and outputs it. It waits for the success/failure report from the execution report handling process, and it stores the content of the report in the internal state of the expert in charge. While waiting for the report, if it receives a message that an interruption is detected from the understanding process, it calls the *handle-interruption* method of the expert in charge, and performs this procedure 3).
4) When the subtask is finished, the expert to perform a remaining subtask becomes in charge. If no subtask remains, a new task is decomposed by the task planner into subtask sequences, and the expert for the first subtask becomes in charge.
5) If there is no task to be performed, it waits for a new human utterance.

The execution report handling process receives the reports of the success/failure of action execution from the behavior controller and tells them to the action selection process.

F. Advantages of the Model

MEBDP meets the requirements described in Sec. II-B. Task-oriented, mixed-initiative dialogues are possible because MEBDP employs experts that perform dialogue management for such kind of dialogues. Dynamic domain switching is also possible because which expert should be in charge is determined based on the results of human utterance understanding by all experts. Even while one expert is in charge and engaged in a dialogue in its domain, another expert can be in charge when human utterance in another domain is detected. Interruption can be handled because understanding, action selection, and execution report handling processes run in parallel.

In addition, MEBDP makes it easy to build and maintain the planner, because each expert is designed separately and different planning algorithms can be used. It is possible to add new experts without considering consistencies with existing experts. This is one advantage of MEBDP over *jijo-2* architecture, in which dialogue management strategy is fixed [12].

V. Implementation and Demonstration

We have implemented MEBDP and combined it with the Honda humanoid robot ASIMO, a speech recognition system Julian, and a text-to-speech system FineVoice[1] for demonstration. Julian is a variant of Julius [19] and it utilizes network grammars as the recognition language models. For this demonstration system, microphones and speakers directly connected to the computers are used for the speech input and output. We assume that the system knows who is talking to it, as it has employed neither face recognition nor speaker identification.

The current implementation provides templates of experts so that it becomes easy to create new experts in specific domains. For RU and IOD experts, the templates for the frame-based dialogue management are prepared. The *understand* method consists of *language understanding*, which converts a speech recognition result into semantic representation, and *discourse understanding*, which updates the internal state based on the semantic representation. Developers of experts can configure the language understanding component by preparing the set of utterance patterns and keyword lists as in spoken dialogue system toolkits [20]. Below are examples.

```
utterance patterns:
  action-type: question-weather
    Tell me the weather in *city* *day*
```

[1] FineVoice is a product of NTT-IT Corporation.

```
    I'd like to know *day*'s weather in *city*
  action-type: refer-city
    It's *city*
keywords:
  class: *day*
    today, tomorrow
  class: *city*
    tokyo, kyoto
```

They are used for generating finite-state-transducer-based language understanding components. Discourse understanding and the *select-next-action* method can be implemented with Java programs using predefined methods for operating slots.

For PAP and IP experts, the templates for simple hierarchical planning are prepared. The *understand* method can be configured in the same way as RU and IOD experts, although the internal states are not frames. The *select-next-action* method can be configured by specifying plan libraries for decomposing the subtask into action sequences. Procedures for handling interrupting utterances based on the their understanding results also need to be specified by the developer.

The language model for the speech recognizer is automatically constructed from the utterance patterns and keywords for language understanding components in all experts so that utterances in all the task domains can be recognized. For the action realizer, we utilized template-based generation.

The current demonstration system can perform six tasks, five of which are providing information such as weather information and person's schedules through dialogues, and the other one is calling a person. The speech recognition vocabulary is about 400 words. The task domain selection based on human utterances utilizes hand-written, heuristic rules. Fig. 4 shows an example interaction. For moving in the room where we demonstrate the system, the current system utilizes ultrasonic tags to locate humans, ASIMO, obstacles such as tables and chairs [21]. Therefore it does not have to ask the location of a human before going to him/her, unlike the example in Sec. IV-D.

Note that the interaction is done in Japanese, although the figure contains only English translations. In this example, the task domain is successfully switched. Although this example does not include interruption handling, it can be also demonstrated. If *A* says "you don't need to do that" while ASIMO is moving to the bedroom, it stops moving and returns to *A*.

TABLE I

SOME OF THE EXPERTS IN THE DEMONSTRATION SYSTEM

type	domain	description
RU	weather	Understands questions about weather information. Performs frame-based dialogue management. The frame has ten slots including *information-type*, *date*, and *place*.
IP	weather	Tells requested weather information. Accepts requests for repetition.
RU	schedule	Understands questions about peoples' schedules. Performs frame-based dialogue management. The frame has two slots: *person*, and *date*.
IP	schedule	Tells requested schedule information. Accepts request for repetition.
RU	calling	Understands request for calling a person. Performs frame-based dialogue management. The frame has one slot *person*.
PAP	moving	Performs physical action for moving.
IP	telling message	Tells a message to a person. Waits for his/her acknowledgment.

Agent	Action/Utterance (English translation)	Expert
A	Can you tell me tomorrow's weather in Saitama?	RU in the weather domain
ASIMO	Tomorrow's weather in Saitama?	
A	Yes.	
ASIMO	It will be sunny in Saitama tomorrow.	IP in the weather domain
A	Can you tell me *B*'s schedule?	RU in the schedule domain
ASIMO	*B*'s schedule for tomorrow?	
A	Yes.	
ASIMO	*B* will have a meeting with *C* tomorrow.	IP in the schedule domain
A	Can you call *D*?	RU in the calling domain
ASIMO	I'm going to call *D*. (*move to D's location*)	PAP for moving
ASIMO	*D*, *A* is calling you.	IP for telling messages
D	Okay.	

Fig. 4. Example Interaction

Fig. 5. Dialogue with the Demonstration System

VI. CONCLUSION AND FUTURE WORK

This paper presented MEBDP (multi-expert-based behavior and dialogue planning), a two-layer model for the behavior and dialogue planning module in conversational service robots. The advantage of MEBDP over previously proposed conversational robot planner is that it enables accurate mixed-initiative communication using dialogue management, switching domains, and handling interruptions. The model has been implemented and combined with a humanoid robot. Although the current system is limited in terms of the numbers of tasks and the size of vocabulary for conversation, it demonstrated the effectiveness of MEBDP.

The algorithms for dialogue management and planning used in the experts in our current implementation are simple. It would be effective to incorporate recently developed techniques such as reinforcement-learning-based dialogue management [22] and strategies for avoiding unnecessary confirmation requests [23]. We need, however, to make it clear how to incorporate them into conversational service robotic systems which are more complicated than single domain spoken dialogue systems. We as well need to explore a systematic way to construct rules for task domain selection and rules for out-of-domain utterance detection. Our future work also includes enabling a robot to schedule tasks when humans asks it multiple tasks.

VII. Acknowledgments

We would like to thank Toshio Yokoyama for his support and encouragement. Thanks also go to Kentaro Yamada and Tokitomo Ariyoshi who helped us to build the demonstration system.

References

[1] S. Russell and P. Norvig, *Artificial Intelligence: a Modern Approach (2nd ed.)*, Prentice Hall, 2002.

[2] R. A. Brooks, "A robust layered control system for a mobile robot," *IEEE Trans. on Robotics and Automation*, vol. 2, pp. 14–23, 1986.

[3] A. I. Rudnicky, C. Bennett, A. Black, A. Chotomongcol, K. Lenzo, A. Oh, and R. Singh, "Task and domain specific modelling in the Carnegie Mellon communicator system," in *Proc. 6th ICSLP*, 2000.

[4] S. Seneff, "Response planning and generation in the MERCURY flight reservation system," *Comp. Speech and Lang.*, vol. 16, no. 3–4, pp. 283–312, 2002.

[5] T. Kanda, H. Ishiguro, M. Imai, T. Ono, and K. Mase, "A constructive approach for developing interactive humanoid robots," in *Proc. IROS-2002*, 2002, pp. 1265–1270.

[6] Y. Hoshino, T. Takagi, U. Diprofio, and M. Fujita, "Behavior description and control using behavior module for personal robot," in *Proc. ICRA-2004*, 2004, pp. 4165–4171.

[7] Y. Makihara, M. Takizawa, K. Ninokata, Y. Shirai, J. Miura, and N. Shimada, "A service robot acting by occasional dialog – object recognition using dialog with user and sensor-based manipulation –," *Journal of Robotics and Mechatronics*, vol. 14, no. 2, pp. 124–132, 2002.

[8] T. Yoshimi, N. Matsuhira, K. Suzuki, D. Yamamoto, F. Ozaki, J. Hirokawa, and H. Ogawa, "Development of a concept model of a robotic information home appliance, ApriAlpha," in *Proc. IROS-2004*, 2004, pp. 205–211.

[9] M. Zobel, J. Denzler, B. Heigl, E. Noth, D. Paulus, J. Schmidt, and G. S. and, "MOBSY: integration of vision and dialogue in service robots," in *Proceedings of the Second International Workshop on Computer Vision Systems (ICVS)*, 2001, pp. 50–62.

[10] T. Konashi, M. Suzuki, A. Ito, and S. Makino, "A spoken dialog system based on automatic grammar generation and template-based weighting for autonomous mobile robots," in *Proc. 8th ICSLP*, 2004.

[11] E. Topp, D. Kragic, P. Jensfelt, and H. Christensen, "An interactive interface for service robots," in *Proc. ICRA-2004*, 2004, pp. 3469–3474.

[12] H. Asoh, Y. Motomura, F. Asano, I. Hara, S. Hayamizu, K. Itou, T. Kurita, T. Matsui, N. Vlassis, R. Bunschoten, and B. Kroese, "Jijo-2: An office robot that communicates and learns," *IEEE Intelligent Systems*, vol. 16, no. 5, pp. 46–55, 2001.

[13] M. McTear, "Spoken dialogue technology: Enabling the conversational user interface," *ACM Computing Surveys*, vol. 34, no. 1, pp. 90–169, 2002.

[14] D. Goddeau, H. Meng, J. Polifroni, S. Seneff, and S. Busayapongchai, "A form-based dialogue manager for spoken language applications," in *Proc. 4th ICSLP*, 1996.

[15] J. Chu-Carroll, "MIMIC: An adaptive mixed initiative spoken dialogue system for information queries," in *Proc. 6th Applied NLP*, 2000, pp. 97–104.

[16] P. A. Heeman, M. Johnston, J. Denney, and E. Kaiser, "Beyond structured dialogues: Factoring out grounding," in *Proc. 5th ICSLP*, 1998, pp. 863–866.

[17] K. Dohsaka and A. Shimazu, "System architecture for spoken utterance production in collaborative dialogue," in *Working Notes of IJCAI 1997 Workshop on Collaboration, Cooperation and Conflict in Dialogue Systems*, 1997.

[18] B.-s. Lin, H. m. Wang, and L. s. Lee, "Consistent dialogue across concurrent topics based on an expert system model," in *Proc. 6th Eurospeech*, 1999, pp. 1427–1430.

[19] A. Lee, T. Kawahara, and K. Shikano, "Julius – an open source real-time large vocabulary recognition engine," in *Proc. 7th Eurospeech*, 2001, pp. 1691–1694.

[20] J. Glass and E. Weinstein, "SPEECHBUILDER: Facilitating spoken dialogue system development," in *Proc. 7th Eurospeech*, 2001, pp. 1335–1338.

[21] Y. Nishida, H. Aizawa, T. Hori, N. Hoffman, T. Kanade, and M. Kakikura, "3D ultrasonic tagging system for observing human activity," in *Proc. IROS-2003*, 2003, pp. 785–791.

[22] S. Singh, D. Litman, M. Kearns, and M. Walker, "Optimizing dialogue management with reinforcement learning: Experiments with the NJFun system," *J. Artif. Intel. Res.*, vol. 16, pp. 105–133, 2002.

[23] K. Dohsaka, N. Yasuda, and K. Aikawa, "Efficient spoken dialogue control depending on the speech recognition rate and system's database," in *Proc. 8th Eurospeech*, 2003.