Appl Intell (2012) 36:208–228
DOI 10.1007/s10489-010-0255-y

Formation conditions of mutual adaptation in human-agent collaborative interaction

Yong Xu · Yoshimasa Ohmoto · Shogo Okada · Kazuhiro Ueda · Takanori Komatsu · Takeshi Okadome · Koji Kamei · Yasuyuki Sumi · Toyoaki Nishida

Published online: 24 September 2010

Abstract When an adaptive agent works with a human user in a collaborative task, in order to enable flexible instructions to be issued by ordinary people, it is believed that a mutual adaptation phenomenon can enable the agent to handle flexible mapping relations between the human user's instructions and the agent's actions. To elucidate the conditions required to induce the mutual adaptation phenomenon, we designed an appropriate experimental environment called "WAITER" (Waiter Agent Interactive Training Experimental Restaurant) and conducted two experiments in this environment. The experimental results suggest that the proposed conditions can induce the mutual adaptation phenomenon.

Y. Xu (✉)
Division of Advanced Information Technology & Computer Science, Institute of Engineering, Tokyo University of Agriculture and Technology, 2-24-16 Naka-cho, Koganei-shi, Tokyo 184-8588, Japan
e-mail: xuyong@cc.tuat.ac.jp

Y. Ohmoto · S. Okada · Y. Sumi · T. Nishida
Graduate School of Informatics, Kyoto University, Kyoto, Japan

K. Ueda
Department of System Sciences, The University of Tokyo, Tokyo, Japan

T. Komatsu
International Young Researcher Empowerment Center, Shinshu University, Nagano, 390-8621, Japan

T. Okadome
School of Technology and Science, Kwansei Gakuin University, Osaka, Japan

K. Kamei
Advanced Telecommunications Research Institute International, Kyoto, Japan

Keywords Human-agent interaction · Mutual adaptation · Waiter agent · Think-aloud method

1 Introduction

In recent years, many pet robots, such as SONY's AIBO (Sony Corporation, Japan) and Jetta's Pleo (Jetta Industries Company Limited, USA) have attracted public attention because they are being used in ordinary home environments. Although many people enjoy interacting with these pet robots for the first time, few actually regard them as communicative partners. Furthermore, many people feel bored after playing with them after only a short period. On the other hand, many people play with their pet animals and live together with them for several years without feeling bored. One of the reasons for this is that people can communicate with pet animals but not with pet robots. For example, suppose that an owner is trying to teach a pet dog to sit down when he/she says "sit down." Initially, the dog may not be able to understand the meaning of the word, and therefore, it may ignore the instruction or perform some wrong actions. On finding that the dog cannot understand his/her instruction, the owner may punish it in some manner, perhaps by scolding or hitting it. The owner may repeat the same instruction several times. After several failures, the owner may try some other words with a similar meaning for the same intention. This is considered as adaptation from the owner to the pet dog. If the dog can change its actions to infer the owner's intended action, it can be considered as the adaptation from the pet dog to the owner. If the pet performs the right action, the owner may give it some favorite foods or toys as rewards. The pet dog's adaptation and the owner's adaptation simultaneously occurs. Comparing to the case of a one-way adaptation, the two-way adaptation enables both the pet and the owner to establish interaction protocols in a more natural manner. This type of phenomenon in which both the owner and the pet adapt to each other is called *mutual adaptation*.

In the case of a pet robot, if the owner cannot keep a consistency with the mapping relation of a specific instruction of the human owner and a specific action of the pet robot, it becomes very difficult for the pet robot to finish its task. However, with a real pet animal, it is natural to build a mapping relation between an instruction and an action through interaction instead of continuing to issue the same instruction without considering the pet's understanding. The feature of flexible mapping relation helps not only the owner but also the pet to establish an interaction protocol.

It is reasonable to presume that pet animals must have some unique competence that pet robots do not. This enables them to understand the meaning of their owner's intention even when their owners change the mapping relation between their instructions and the pet's actions during their interaction. It is well known that people have high adaptability and that they often adapt to an environment or other people unintentionally. The competence of taking advantage of people's adaptability should also be useful.

We assume that a mutual adaptation phenomenon may be one of the key factors that enable pet animals and their owners to maintain a sustainable interaction. Owners usually need to spend a lot of time to learn how to teach their pet dogs to perform a simple action such as sit down for the first time. As long as they succeed for the first time, they can often teach their pet to raise its paw, or perform some other actions in a short period of time. They can usually finish this task without being given any predefined mapping relation between their instructions and the dog's actions.

The mutual adaptation phenomenon is a widely existing phenomenon in all fields of artificial adaptive systems that require interaction with human users. The advantages of an agent capable of mutual adaptation include flexible mapping relations, high success rate in establishing interaction protocols, and natural learning procedure.

A protocol implies a commitment that both a human user and an agent prefer to perform the same pair of actions after receiving the same instruction in the same or similar situation. Because this is a general property of all types of adaptive agents, this competence is expected to improve the performance of such agents in various applications.

Some pet robots can learn fixed mapping relations between their available actions and human users' instructions. However, they often fail or are forced to restart their learning when human users change the mapping relation between instructions and their actions during the learning procedure. If the pet robots were to possess the ability to mutually adapt in a manner similar to real pet animals, it will be possible for them to handle such situations even when the human owners change the mapping during their learning. This type of competence may contribute toward manufacturing developmental robots, and it should be helpful in real-world applications where fixed mapping relations are insufficient.

In the research field of AI (artificial intelligence), an adaptive artifact (robot or software program) that can perform some work as a substitute for a human is often called an agent. There are three types of adaptive agents, namely, non-adaptive agent, normal adaptive agent, and mutual adaptive agent. Obviously, it is almost impossible for a human user to expect a nonadaptive agent to extend its ability to perform a new action. Let's consider a human user who needs to teach a normal adaptive agent to do something. Normal adaptive agent can learn to perform a specific action after receiving a specific instruction. In other words, it can learn a fixed mapping relation. However, this mapping relation normally cannot be changed as long as it is established. Because the normal adaptive agent can only learn fixed mapping relations, in order to ensure consistency of the instructions, the human users have to choose the same instruction for a specific intended action to avoid causing inconsistency with previous mapping relations between the instructions and the intended actions. The learning behavior of this type of agent is usually slow, non-developmental, and involves a large learning load to human users.

Mutual adaptation is considered to be one of the promising solutions for the problem of how to enable an agent to learn flexible mapping relations. We argue that an agent can qualify to be a sustainable communicative partner of a human user only if it posses the competence of enabling the mutual adaptation phenomenon. Since people's adaptability is usually far higher than that of agents, it is reasonable to enable an adaptive agent to handle the changes in mapping relations and new situations by taking advantage of human users' adaptability.

To our knowledge, no study on formation conditions and evaluation approaches of the mutual adaptation phenomenon has yet been conducted. Therefore, this study aims to elucidate the formation conditions of the mutual adaptation phenomenon so that it is possible to build a platform for developing an adaptive agent with mutual adaptation competence.

In order to provide a specific task and a working platform to induce the mutual adaptation phenomenon for studying, a collaborative task is designed for a human manager to instruct a waiter agent that can adapt to the human manager in a virtual restaurant, and a novel experimental environment called "WAITER" (Waiter Agent Interactive Training Experimental Restaurant) is developed for the purpose of verification.

The contributions of this study are as follows:

1. Defined the concept of the mutual adaptation phenomenon and hypothesized its formation conditions.

2. Designed a human-agent collaborative task that can induce the mutual adaptation phenomenon.
3. Implemented an appropriate experimental environment for studying the mutual adaptation phenomenon.
4. Conducted experiments to identify the proposed conditions.
5. Created a platform for developing an agent that can handle situations of mutual adaptation.

2 Related works

HAI (Human Agent Interaction) is a new research area that focuses on the problems of: how to design interaction between a human user and an adaptive agent, what factors may affect the interaction, how to implement the agent to enable a natural interaction, etc. Researches on HCI (Human-Computer Interaction) and HRI (Human-Robot Interaction) are highly related to the topics in the area of HAI.

In my opinion, there are three types of research methods to study the mutual adaptation. The first type of research method tries to build a model for agent's adaptation on the basis of traditional research on AI, such as machine learning. The second type of research method tries to build a model for human's adaptation to design an human-centered or human-like adaptive system. The third type of research method focuses on interaction between humans and agents to design an interaction-based adaptive system.

To my knowledge, Yamada is the first person to propose the concept of *mutual adaptation* in the research field of *HAI* [18]. In his early works on the mutual adaptation phenomenon, he conducted an experiment called "mind reading game [20]." In the experiment, a human user and an agent tried to infer the partner's state of mind by recognizing the facial expression of the partner. In his another experiment [19], a human user was asked to train a pet robot AIBO with a method of classical conditioning. Martin et al. [9] developed an intelligent user interface management system that could adapt the interface to a user depending on the information stored in a user model during the execution stage instead of the design stage. Their works preferred to build an adaptive agent system rather than study human's adaptation; therefore, they adopted the first type of method.

Thomaz [12] proposed a socially guided machine learning model based on reinforcement learning, and developed a robot learner that can understand human teaching behavior based on this model. This research focused on building an easy-to-teach robot with respect to a human-teacher and robot-learner framework. In this research, a fixed mapping relation between an instruction and an action is assumed as a precondition. Ana [1] developed an adaptive and intelligent educational system that can automatically learn the best pedagogical policy for teaching students through reinforcement learning. Their research focused more on making the human users express their intentions in a natural way or making the agent express its internal states effectively, therefore, they adopted the second type of method. In contrast, our study focuses more on the flexible mapping relation.

Some studies argued that an artificial adaptive system should work basing on some human-like dynamics and tried to build such a system based on such models. For example, Ogata et al. [11] built a dynamical system based on a RNNPB (recurrent neural network with parametric bias) model that could enable a robot to adapt to its environment in an open-ended way. His works pay more attention to design an adaptive system with a dynamics that can enable the agent to adapt to its environment. In his work, a human user is considered to be a part of the environment rather than an individual interaction partner. Our works consider a human user to be a highly adaptive and individual interaction partner. Miyake [8] interpreted the concept of the mutual adaptation as a mutual synchronization process between a human user and an adaptive machine. Kato et al. [3] developed an adaptable prosthetic hand system and observed some mutual adaptation phenomena between human brains and adaptable machines through f-MRI analysis. Their studies tried to build a biological model to interpret the mutual adaptation phenomenon; therefore, these research adopted the second type of research method.

Some other researchers focused on the role of the mutual adaptation phenomenon in the context of social interaction. For example, Komagome et al. [5] tried to model the mutual adaptation phenomenon between humans and robots and enable a robot to imitate a human's behavior. His research focused on the influence of the mutual adaptation phenomenon on humans and robots and how the phenomenon spreads through imitation behaviors. His research adopted the third type of research method, but did not clarify the inducing conditions. Our research mainly focuses on the inducing conditions of the mutual adaptation phenomenon.

In a previous work [16], we proposed a two-layered mutual adaptation model. Komatsu et al. [6] developed a mutual adaptive speech interface. This interface adopts the cognitive features which humans use for communication and induces and exploits users' adaptation. This research mainly studied the lower layer of the mutual adaptation model, and mainly focused on studying how human users acquire the meanings of voice instructions through paralanguage in a simple TV-game. He also tried to measure the mutual adaptation processes on Akaike's Information Criterion [7]. Although this approach is a promising way to interpret some kinds of mutual adaptation phenomenon, a lack of versatility limits its range of application. Komatsu's research adopted the third type of research method. My research is intended to adopt the third type of method and tried to extends his research to the upper layer and develop an experimental environment in which a human-agent collaborative task can be accomplished by a human user and an adaptive agent without predefined fixed mapping relations between instructions and actions.

3 Concept of mutual adaptation

3.1 Definition

The mutual adaptation competence of the agent affords three advantages, namely, flexible mapping relations, easy establishment of interaction protocols, and a natural learning procedure; therefore, the mutual adaptation phenomenon is expected to improve the efficiency and probability of success of a human-agent collaborative task.

Suppose that there exist two adaptive agents A and B with different adaptabilities. In order to induce the mutual adaptation phenomenon, the following preconditions should first be satisfied:

1. They need to accomplish a collaborative task.
2. Each agent can only access different parts of partially available information about the task.

If any agent has complete or perfect information about the task, it should preferably ignore the reactions of the other. Otherwise, it may try to actively take the initiative. However, the other agent may have no chance to take the initiative. This phenomenon may prevent the occurrence of the mutual adaptation phenomenon.

Next, we will introduce conditions to distinguish the mutual adaptation phenomenon. In order to achieve a common purpose, both agents are required

1. to change behavior to adapt to the partner simultaneously,
2. to estimate the partner's intention with respect to a situation by building the partner's model,
3. and, to form an adaptation loop.

   An adaptation loop implies that if agent A changes some behaviors by adapting to agent B, agent B should change its behaviors as a response. As a result, agent A will change its behaviors again as a response to the changes in agent B's behaviors.

In general, the two agents have to complete a task by gradually adapting to each other. The aforementioned phenomenon is termed *mutual adaptation* in this paper.

3.2 Hypothesis of conditions

One objective of this research is to clarify the conditions inducing the mutual adaptation phenomenon and predict the consequent behaviors. In order to achieve a common purpose, each agent is required

- to receive a feedback or reward from the other agent;
- to express its internal state;
- to establish a common protocol during its interaction;
- and, to decide next action by referring to the partner's past action (explicitly or implicitly build a model for the partner).

The quality of feedback and expression of internal state affect the partner's adaptation. It is obvious that if either agent cannot receive feedback from the other, neither will be motivated to change its actions, and therefore, an adaptation loop cannot be established. A common protocol is necessary for establishing an adaptation loop. Referring to past actions can help build a model of the partner. Without referring to the past action, it is difficult to build a model of the partner, and therefore, it is difficult to predict the partner's action to adapt to the partner in the future.

It is assumed that if some of following inducing conditions are satisfied, the mutual adaptation phenomenon should occur.

- Both agents changed actions or policies as reactions to their partner's behaviors;
- Both agents could take the initiative;
- Changes of action were concatenated with each other within a brief time.

The inducing conditions are different from the conditions that define the mutual adaptation phenomenon, because the former cause the mutual adaptation phenomenon to occur; therefore they should be observed before the occurrence of the mutual adaptation phenomenon.

An agent that exhibits these adaptive behaviors should have a high degree of potential for development. Because the agent can change actions within brief time and communicate feedback with its partner, this ability enables it to handle new situations and build new interaction protocols. Therefore, the agent is capable of developing its adaptability. When dealing with such an adaptive agent, a human user is expected to naturally adapt to the agent.

From an observer's perspective, the mutual adaptation phenomenon can be induced only if an agent exhibits three primary characteristics: autonomy, reactivity, and adaptability. Autonomy implies that the agent can take initiative. Reactivity implies that the agent can react to the partner's actions. Adaptability implies that the agent can change its behaviors over time. Autonomy depends on the agent's ability to take initiative. An agent that can take initiative implies that it prefers to make a beneficial decision than interact with its partner. If either of them only acts by itself disregarding the partner's action, neither of them can be said to take the initiative. We hypothesize that the ability of taking initiative is the first important ability to induce the mutual adaptation phenomenon.

Because reactivity greatly depends on the time factor, and this factor is considered to be one of the important elements for enabling mutually adaptive learning in human-agent interaction [13], a reasonable time-lag should be investigated. If changes in one agent's action cannot be concatenated with the other agent within a reasonable interval, the two agents may simply change their respective actions without caring about the other's response. We hypothesize that the ability of taking reactive actions is the second important ability to induce the mutual adaptation phenomenon.

The result of adaptability depends on an agent's ability to establish a protocol. If two agents can establish a common protocol, it will be helpful to accomplish their collaborative task. We hypothesize that the ability of developing a common protocol is the third important ability to induce the mutual adaptation phenomenon.

A mutually adaptive agent can not only effectively adapt to a specific human user but also generally reduce the load of adaptation for most human users. By extending to a developmental agent, the agent is expected to handle new situations and establish a sustainable relation with human users.

When dealing with an agent with the competence of mutual adaptation, people do not need to pay additional attention to how to interact with the agent from the beginning, and therefore, the load of learning by human users is expected to be reduced. People can learn how to instruct the agent through interactions with it, and the agent can improve its performance over time, thus giving it the potential to develop. Therefore, people can interact with the agents in a manner similar to interacting with a human partner.

Because it is difficult for an agent to achieve complete information in a real-world task, it should be useful to study if an agent with mutual adaptation competence can handle a situation in which it can only achieve partial information of a task.

Having observed the mutual adaptation phenomena in human-human experiments [17], we focus on the mutual adaptation phenomenon in the case of a human-agent in this research.

4 Computational mechanism for mutual adaptation

In order to create an environment to induce the mutual adaptation phenomenon, we employ a synthesis approach by implementing some adaptive agents and observing their interactive behavior with people. This can provide us with a candidate solution to disclose the mechanism of mutually adaptive behavior between a human user and an agent, and help to design intelligent systems that can take advantage of such a mechanism.

In this study, we focus on the case of a human-agent interaction. The formation process of mutual adaptation consists of at least the following five steps:

- Step(1): It is supposed that one agent, labeled as agent-A, plays the role of an instructor and issue instructions to the other agent. The other agent, labeled as agent-B, plays the role of an actor and expresses its internal state through its behaviors;
- Step(2): In order to instruct agent-B to finish a goal X, agent-A needs to change some properties of the instruction (for example, variety, frequency, policy of instruction selection, or time interval between instructions) so that agent-B may change its reaction accordingly;
- Step(3): Both agents will learn X by referring to feedback information such as scores from the environment or changes in the other agent's actions;
- Step(4): If the instructor agent-A realizes that learning on X has been finished or that learning on some other goal, say, Y is more important in the current situation, it may shift its focus of instruction from X to Y. In contrast, if agent-A realizes that learning X has failed, it may return to step(2) to try other methods. Agent-B may also try to take initiative by ignoring the instructions from agent-A;
- Step(5): Repeat step (2) through step (4) and replace X or Y with other proper goals according to the current situation until the tasks are finished successfully or some terminal conditions are reached.

An action implies one or more consecutive primitive movements by the agent. A policy implies criteria of action selection by an agent's decision-making components. A situation implies perception of sensation data that is detected by the agent.

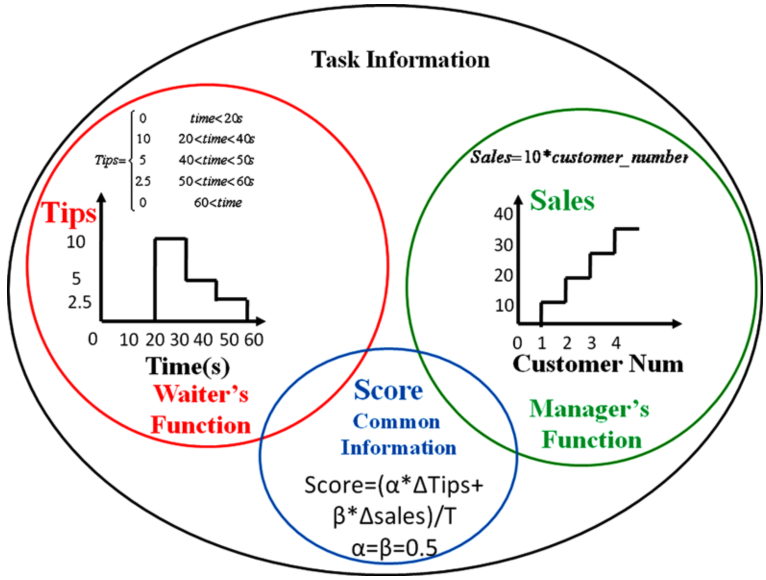
5 Task and experimental environment

In order to explain the concept of mutual adaptation, it is necessary to design a specific task in which an observable mutual adaptation phenomenon can be induced and recorded, so that conditions about the mutual adaptation phenomenon can be analyzed and studied.

In order to provide a specific task to induce the mutual adaptation phenomenon so that it can be studied, we designed a human-agent collaborative task. For the purpose of verification, a novel working platform of the experimental environment is developed. The task scenario used is a restaurant world "WAITER." This is a Matlab-based computer game platform that is designed to investigate the conditions of the mutual adaptation phenomenon. In this task, a manager (human instructor) trains a waiter (agent) in a virtual restaurant to provide service to customers. A human participant always plays a role of a manager. An agent always plays a role of a waiter. The customers are automatically generated by the system at random times.

Two types of experiments, namely, WOZ (Wizard of OZ) agent experiment (experiment 1) and autonomous agent experiment (experiment 2) were conducted. A WOZ agent

Fig. 1 Evaluation function



has less degrees of freedom, and a human WOZ operator partly controls its action-selection policy. In contrast, an autonomous agent can choose its actions by itself.

Two experiments were conducted to investigate the precondition and inducing condition of the mutual adaptation phenomenon.

To facilitate the comparison of different types of agent's behaviors, we designed several modes for the waiter agent. First, a manual mode (M-mode) is designed to study people's adaptation to the experimental environment WAITER. With the exception of the M-mode, several different types of autonomous modes are designed to study the conditions of the mutual adaptation phenomenon. In the WOZ agent experiment (experiment 1), in addition to the manual mode (M-mode), there are 3 autonomous modes, namely, TS-mode (Timely stage Switching mode), OS-mode (Operator stage Switching mode) , and T/OS-mode (Timely or Operator stage Switching mode).

Because the agent in the manual mode does not have any adaptability, the human manager's adaptation to the experimental environment is focused on. In contrast, because the agent in the autonomous mode has adaptability, the participant's adaptation to the agent is focused on, and the mutual adaptation phenomenon is expected to be induced as well.

Generally, there exist three types of agents: agent without adaptability, normal adaptive agent, and mutual adaptive agent. All three types of agents are implemented in the autonomous agent experiment (experiment 2). In the manual mode (M-Mode), the agent without adaptability is implemented. In this case, people's adaptation can be observed to provide the result of people's adaptive behavior that can be used to compare with the behavior of the autonomous agent.

The autonomous agent mode can be further divided into four modes, namely, L-mode (liner prediction mode), R-mode (random mode), B-mode (Bayesian network mode), and O-mode (Observation-phase mode) .

The R-Mode is expected to observe the behavior of a normal adaptive agent that is designed without the competence of encouraging people's adaptation. This type of agent chooses its actions without considering people's reactions, and it is expected to observe how people adapt to such an agent.

For an agent with mutual adaptive competence, three types of modes, namely, L-Mode, B-Mode, and O-Mode are implemented. The agent in the L-Mode employs a simple adaptive policy and adapts to people by following a simple approach. An agent in the B-Mode is expected to utilize a model that is built based on past data.

Furthermore, in order to enable people to rebuild the model for the agent instead of providing instructions for all situations, the O-Mode is implemented with the introduction of several enforcing observation phases during a single trial. It is expected to find difference in people's instruction behaviors before and after the observation phase.

As shown in Fig. 1, either the manager or the agent only has partial information about the task; however, both of them share common information (score). Following this design, the manager and the waiter have to accomplish the task through collaboration. The manager can only access the sales value; meanwhile, the waiter agent can only access the tips value. Neither of them can access the information known to the partner. In other words, both of them can only access different parts of partially available information about the task. Therefore, the preconditions have been satisfied.

It is essential that the degree of satisfaction of the customers and the profit of the restaurant are maximized simultaneously. Here, the degree of satisfaction of the customers

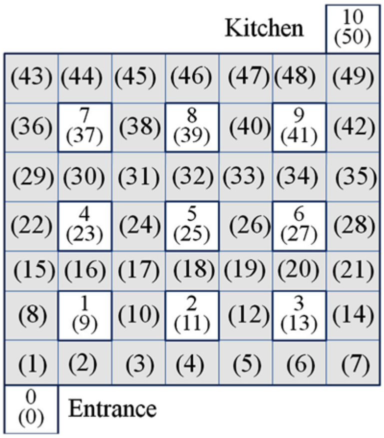


Fig. 2 Seat layout of WAITER environment

is reflected by the amount of tips given to the waiter agent by the customers.

The seat layout of the "WAITER" system is shown in Fig. 2. There are 51 cells in the restaurant. Except for nine seats (Numbers 1 through 9 without parenthesis) and two special cells, namely, the entrance (Number 0) and the kitchen (Number 10), the waiter agent can move in 40 cells (numbers in parenthesis). The manager can instruct the waiter by pressing one of eleven buttons, namely, an entrance button, a kitchen button, and nine seat buttons. The cell Entrance has up to two states: "no new customer" and "has new customer." The cell Kitchen has up to two states: "no dish" and "has dish." Each seat has up to four states: "vacant," "waiting for order," "eating," and "need to clear." The waiter agent has three states: "free," "guiding customer," and "carrying dish."

WAITER system provides a state-action space for a single agent that uses a fixed set of actions on a fixed set of states. This system is defined by $\{L, I, \Sigma, T, A\}$. There exists a finite set of k locations $L = \{l_1, \ldots, l_k\}$. In the restaurant task scenario, $k = 51$; $L = \{l_1, \ldots, l_{51}\}$. Among the 51 cells, the waiter agent can move in 40 cells, whereas the remaining 11 cells are locations for 11 seats, namely, the entrance, kitchen, and nine seats. Let L^A be a set of possible agent locations: $L^A = \{l_1, \ldots, l_{k_a}\}$. Let L^I be a set of possible instruction locations (seat locations): $L^I = \{l_1, \ldots, l_{k_i}\}$. In the restaurant task scenario, $k_a = 40$, $L^A = \{l_1, \ldots, l_{40}\}$, and $k_i = 11$, $L^I = \{Entrance, seat_1, \ldots, seat_9, Kitchen\}$.

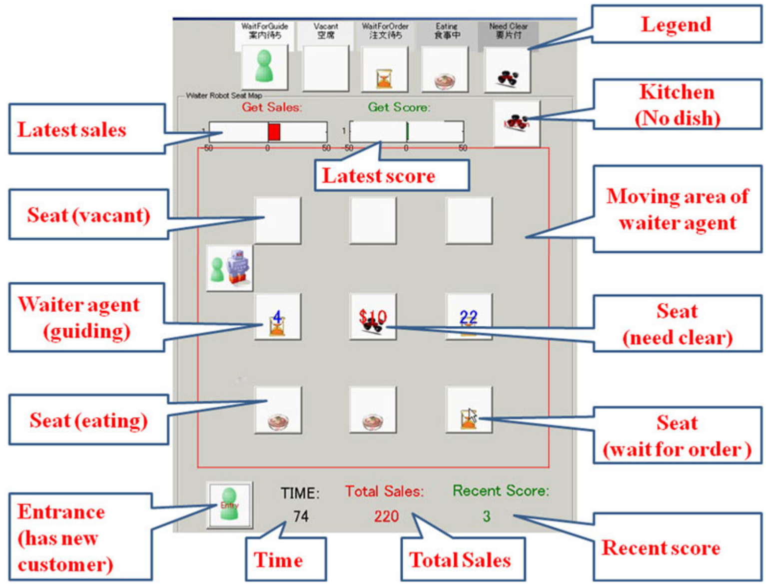
There exists a finite set of states $S = \{s_1, \ldots, s_m\}$. There exists two types of states: agent state and seat state. There exists three types of seats: entrance, kitchen and customer seat. Each of them can be in one mutually exclusive state. Therefore, Ω_i is the set of states s_i and $S^* = (\Omega_{agent} \times \Omega_{entrance} \times \Omega_{kitchen} \times \Omega_{seat_1} \cdots \times \Omega_{seat_9})$ is the entire state configuration space. An agent has three states: $\{free, guiding, clearing\}$. An entrance has two states: $\{No\_customer, Has\_customer\}$; a kitchen, two states: $\{No\_dish, Has\_dish\}$, and each of the nine customer states, four states: $\{Vacant, Wait\_for\_order, Eating,$ and $Need\_clear\}$. Then, the legal set of states is $\Sigma \subset (L^A \times L^I \times S^*)$, and a specific state is defined by (l_a, l_i, ω): the agent's location, $l_a \in L^A$; instruction's location, $l_i \in L^I$; and the state configuration, $\omega \in S^*$. T is a transition function: $\Sigma \times A \mapsto \Sigma$. The action space A is expanded from six actions ($Move$, $Order$, $Guide_{start}$, $Guide_{end}$, $Clear_{start}$, and $Clear_{end}$).

In the initial state, the agent is at the entrance. Successful actions for the waiter agent include guiding a customer to any available vacant seat, placing an order as soon as possible, and clearing the seat when it is idle. The manager can issue instructions by pressing one of 11 buttons. Instead of searching for successful actions or action sequences in the state space, the objective of the task is to encourage the human manager to change the instruction method by adapting to the agent, so that the mutual adaptation phenomenon can be induced.

This is a novel task domain that has sufficient complexity for experiments with adaptive autonomous agents. The restaurant task has of the order of 10,000,000 states, with between 2 and 5 actions available in each state. More precisely, the total number of states in the system is $(L^A \times L^I \times S^*) = 40 \times 11 \times (3 \times 2 \times 2 \times 4^9) = 1{,}384{,}120{,}320$. Because the kitchen's current state and the eating state among seat states have no effect on the agent's actions, they are ignored in this learning system. Therefore, the total number of states can be reduced to $40 \times 11 \times (3 \times 2 \times 3^9) = 51{,}963{,}120$. The graphical user interface (GUI) of the WAITER system is shown in Fig. 3.

In this task, let us consider the first condition mentioned before, "Each agent is required to receive a feedback or reward from the other agent." The human manager can send a reward by changing the instruction actions, for example, by changing the variety, frequency, or timing of issuing instructions or the time interval between consecutive instructions. The manager can receive a feedback by watching the actions of the waiter agent. The waiter agent can receive a reward or feedback by detecting changes in the manager's instructions. With regard to the second condition, "Each agent is required to express its internal state," the waiter agent can express its internal state through its movement, and the human manager can express his/her internal state through the timing of issuing or the frequency of instructions. With regard to the third condition, "Each agent is required to establish a common protocol during its interaction," as long as a relatively stable pair of instruction of the manager and action of the waiter are observed, a common protocol can be considered to have been established. With regard to the fourth condition, "Each agent is required to decide next action by referring to the partner's past action," because human managers unintentionally referred to the agent's past action, as long as the

Fig. 3 GUI of WAITER system



agent can refer to the manager's past action, this condition can be satisfied.

It requires 20–30 s for a seat to change into the "wait for order" state after a customer takes a seat. The tips that the waiter agent can receive depend on the time for which the customer has to wait for service. If it takes less than 40 s to place an order, the waiter agent can receive a maximum tip of 10 points. When the waiting time increases, the tip decreases. When the waiting time is longer than 60 s, the tip value becomes 0 points. The tip value is only accessible by the waiter agent and not by the manager, and therefore, the manager does not know the tip-first rule of the waiter agent. Meanwhile, the manager is instructed to guide more customers into vacant seats and place their orders, and therefore, he/she can increase the sales; however, the waiter agent does not know this. If the waiter agent only moves to the seat where it can obtain maximum tips but ignores the instruction of the manager, it may cause some customers to leave the restaurant after waiting for a long time. Because the score is not only related to the sales value but also to the tips that the waiter receives, if the manager cannot realize that some of the waiter agent's autonomous actions can usually cause more tips, they may not obtain the maximum score. Although the tips are not shown on the GUI of the experimental software, the accumulated sales value and recent changed score value are shown to let the manager know how many customers have entered the restaurant and placed orders. The sales value increases by 10 points once a customer finishes eating. When the score changes every T $(= 10)$ times, the recent score value ($\Delta Score$) will be updated by summing up recent changes in tips and sales.

The results of the experiment include log files that record scores, sales, agent states, seat states, and the manager's pressing-button actions. The corresponding screen-captured video data recorded from the GUI output on the display and videotaped records of conversations between the experimenter (the first author of this paper) and the participants for think-aloud protocol analysis are captured using a video camera during all experiments.

$$Tips = \begin{cases} 0, & time < 20\ \text{s} \\ 10, & 20\ \text{s} \le time < 40\ \text{s} \\ 5, & 40\ \text{s} \le time < 50\ \text{s} \\ 2.5, & 50\ \text{s} \le time < 60\ \text{s} \\ 0, & 60\ \text{s} \le time \end{cases} \tag{1}$$

$$Sales = 10 \times Customer\_Numbers \tag{2}$$

$$\Delta Score = (\alpha \times \Delta Tips + \beta \times \Delta Sales)/T, \quad (\alpha = \beta = .5) \tag{3}$$

In the restaurant task situation, as a typical scenario, the waiter agent acts by following its tip-first policy, whereas the human manager decides instructions by following the sales maximization policy. Neither the manager nor the agent knows the rules of the other. However, both of them share a common score to evaluate their preceding actions, and they adapt to the partner by changing their actions when encountering a similar situation again. This phenomenon depends on whether one agent changes its action, then the other changes accordingly, and first agent again changes its actions as a response, i.e., whether an adaptation loop can be established between the two agents.

6 Experimentation

In order to study the mutual adaptation phenomenon in a human-agent collaborative task, two experiments, namely, experiment 1 and experiment 2, are conducted.

6.1 Experiment 1

6.1.1 Objective

The objectives of experiment 1 are to confirm the design of the WAITER system, observe the mutual adaptation phenomenon, and provide learning data for building a Bayesian network model for experiment 2.

6.1.2 Participants and procedure

12 Japanese students (5 male and 7 female in the age range of 20–26 years; average age is 22.75 years) participated in experiment 1. For convenience, we represent the participants as M1 through M12. All the participants are laymen (with no expertise in machine learning) and have no experience of using the WAITER system.

All the students participated in three modes, namely, M-mode, TS-mode, OS-mode (10 min) and some of them also participated in other modes, namely, eight students in the T/OS-mode (10 min), six students in the OS-mode (20 min), and six students in the OS-mode (30 min).

All participants received the following instructions before the experiment: "This is a virtual restaurant. You are a manager of this restaurant. Here is a waiter robot. Your task is to teach this waiter robot to provide service to customers in the restaurant according to your intention. You can issue your instruction by pressing one of 11 buttons, namely, nine seat buttons, one entrance button, and one kitchen button. The waiter has two modes–manual mode and autonomous mode. In the manual mode, it cannot perform any action without your instruction. In the autonomous mode, it can perform some actions by itself without your instruction. Your target is to try and make the agent act as you think in order to maximize the sales and score."

In experiment 1, the agent has two different modes, more specifically, manual mode (M-mode) and autonomous mode (A-mode). In the manual mode, the waiter agent cannot move without the manager's instruction. In contrast, in the autonomous mode, the waiter agent can decide its action by following the manager's instruction or by choosing an autonomous action. The autonomous mode in experiment 1 can be further divided into the TS-mode (Timely stage Switching mode), OS-mode (Operator stage Switching mode), and the T/OS-mode (Timely or Operator stage Switching mode). In the TS-mode, the ability of the waiter agent is set to switch to a higher level at a predefined time moment (every one-third of the trial). In the OS-mode, the WOZ operator (experimenter) manually controls the timing of switching. In the T/OS-mode, the experimenter randomly selects one method from TS-mode and OS-mode. In this experiment, the ability to clear the seat once and the ability to find the kitchen autonomously was designed to improve gradually along with the increases in the switching operation. The ability of the agent to improve is expected to help the manager participant recognize the changes in the agent (agent's adaptation to the manager) and the advantages of the agent's autonomous function, and therefore, they may change their instruction to adapt to the agent.

Table 1 Result of human-adaptive agent experiment 1

No. of participant	No. of trial	Mode	Time (min)	Instruction usage times M	SD
12	12	M	10	127.56	1.80
	12	TS	10	143.67	4.10
	12	OS	10	116.67	3.21
	8	T/OS	10	160.83	4.97
	6	OS	20	275.67	6.66
	6	OS	30	344.00	11.42

Note. M: Manual mode, TS: Timely switching WOZ mode, OS: Operator switching WOZ mode, T/OS: Timely or operator switching WOZ mode

In order to record the intention of the participant, a think-aloud protocol method was adopted for all trials during all experiments. The think-aloud protocol is a method that is often used to gather data in psychology and a range of social sciences. It involves participants thinking aloud as they are performing a task. Participants are asked to say whatever they are looking at, thinking, doing, and feeling as they go about their task. This enables observers to observe first-hand process of task completion (rather than only its final result). Specifically, in our experiment, the experimenter asked every participant why he/she uses a specific instruction when facing a specific situation or a specific policy at a specific moment, what he/she feels about the reaction of the agent, and so on. Although it is impossible to ask participants to explicitly explain their intentions for every instruction, the videotaped data are useful for analyzing the intention of the participants, especially when it is difficult to interpret the log data together with the screen-captured video data.

6.1.3 Results

Table 1 shows the result of experiment 1. In total, 56 log files, 840 min of screen-captured video files, and corresponding voice and videotaped data were recorded. In this experiment, there was a statistically significant difference in the instruction interval between the first half and the second half of the same trial. In 71.4% (40 out of 56) of the trials, the time interval in the second half was significantly longer than that in the first half ($p = 0.027$, one-tailed test).

Table 2 shows the participants' subjective evaluation results about the changes in the agent's behavior and the participant's instructions during breaks between different trials.

Table 2 Subjective evaluation for behavior changes in experiment 1

Rank item	Rank score	
	M	SD
Change in agent's behavior (Q1)	5.3	1.23
Change in instruction behavior (Q2)	4.6	1.50

Note. Q1: "How much do you think the agent changes its behavior?" Q2: "How differently do you instruct the waiter at the beginning and at the end?" (7 level rank, 1-low, 7-high)

When answering questions Q1:"How much do you think the agent changes its behavior?" and Q2:"How differently do you instruct the waiter at the beginning and at the end?" the participants of experiment 1 gave relatively higher scores (Mean = 5.3, Std. Dev. = 1.23, seven-stage evaluation score with lowest evaluation 1 to highest evaluation 7) to Q1 and a slightly high score (Mean = 4.6, Std. Dev. = 1.5, 7 stage evaluation) to Q2.

This result suggests that many participants recognized the changes in the agent's behavior and also changed their instruction method by themselves. With regard to the reason why they changed their instruction method, 9 out of 12 participants in experiment 1 chose the answer "Because the agent changed its actions." The remaining participants chose the answer "Because I could not predict the agent's action."

Next, we would like to introduce three typical examples. In the first example, participant M4 reduced the number of instructions because she found that the agent could perform the task efficiently without her instructions. In the questionnaire, she answered that "Because I found that the agent's performance improved gradually, I reduced my instruction frequency." This indicates that the adaptation of the agent induced the adaptation of the human instructor.

In the second example, participant M3 found it difficult to instruct the agent to follow her instructions to "clear" (vacate) a seat. When the agent automatically moved to the entrance to guide a new customer to a vacant seat, she realized that the agent may prefer "guiding" customers to "clearing" a seat. Therefore, she changed her instruction policy. She did not give the "clear" instruction until the agent finished guiding a customer. In other words, the human instructor changed the instruction policy to allow the agent to occasionally take the initiative to improve the total efficiency. Thus, the participant changed her instruction method to adapt to the changing behavior of the agent. This result was confirmed by M3's speech in the think-aloud recording data.

However, not all participants could adapt well to the behavior of the agent. For example, participant M6 overestimated the ability of the agent and felt disappointed about its behavior. He kept giving instructions without considering the autonomous function of the agent. This can be proved by the result of the questionnaire. For evaluating the ability of the agent, a seven-level evaluation scale was used (where 1 denotes the lowest ability and 7 denotes the highest ability). M6 selected the value 5 in the questionnaire before the experiment, while he selected the value 2 after the experiment. This low value reflects the participant's strong disappointment toward the adaptability of the agent. This result suggests that overestimation of the adaptability of the agent may cause self assertive participant to feel strongly disappointed and encourage them to issue instructions by disregarding the adaptation of the agent; thus, the induction of the mutual adaptation phenomenon becomes difficult. In other words, whether the mutual adaptation phenomenon can be induced not only depends on the agent but also depends on the human. In the current experimental environment, most participants were observed to change their behavior when they realized the changes in the behavior of the agent.

On the whole, the result of experiment 1 proves that the current design of the agent can make the manager realize the adaptation from the waiter agent and let the manager change his/her instruction. This helps establish an adaptation loop between the waiter agent and the human manager.

6.2 Experiment 2

6.2.1 Objective

On the basis of the analysis results of experiment 1, we improved the WAITER environment and conducted experiment 2. The objective of experiment 2 is to verify the inducing condition and compare the performances of agents implemented with several different adaptive algorithms.

In this experiment, the manager's intention is expressed as the choice of the instruction policy. Specifically, it is assumed that there are three candidate choices of instruction priority policy, i.e., "guide-first," "order-first," and "clear-first" in the waiter agent task. Initially, the manger may have a preference for, say, policy A, and may mainly focus on the instruction of this policy. If the waiter agent adapts well, the manager should adapt to the agent by shifting his/her focus to another policy. This leads to an adaptation loop. If such changes can be observed, it can be considered that the mutual adaptation phenomenon has occurred.

We consider three types of agents, namely, agent without adaptability, normal adaptive agent, and mutual adaptive agent. Further, we implement five modes – a manual mode and four autonomous modes (L-mode, R-mode , B-mode, and O-mode).

6.2.2 Participants and procedure

In total, 25 people (16 male and 9 female, 23 Japanese and 2 Chinese, in the age range of 18–37 years, average age

Table 3 Result of human-adaptive agent experiment 2

No. of participant	No. of trial	Mode	Time (min)	Instruction usage times	
				M	SD
25	25	M	15	188.60	2.31
	25	L	15	329.28	10.96
	24	R	15	334.88	11.12
	19	B	15	323.11	10.49
	8	O	15	346.60	11.61

Note. M: Manual mode, L: Linear prediction mode, R: Random mode, B: Bayesian network mode, O: Observation phase mode

of 21.68 years) participated in the experiment. For convenience, we represent the participants as N1 through N25. All participants completed the trials in the M-mode and L-mode, 24 of them completed the R-mode, and 19 of them completed the trails in the B-mode, and 8 of them completed the O-mode.

In the manual mode, without interference from the agent's autonomous function, the manager's intentions can be reflected successfully, and therefore, data from this mode is considered as a benchmark.

Because both utilizing past experience and predicting future action are considered as useful abilities for an adaptive agent, a linear prediction function and a Bayesian network model are implemented for the two types of autonomous agents. In the L-mode, the agent changes its priority policy by choosing the most frequently used instruction that was recently used by the manager. Candidates for the priority policy are "guide-first," "order-first," and "clear-first." In the R-mode, the agent randomly chooses its priority policy. In the B-mode, a Bayesian network model was built offline in advance based on results of experiment 1. The agent can choose the priority policy by calculating the Bayesian network model. In the O-mode, the manager was forced to stop the instruction every 5 min.

The result of the L-mode should reflect a simple and straightforward changing policy. The B-mode is expected to predict the manager's instruction based on experience. The O-mode is designed to enforce the managers to stop their instructions so that they have a chance to recognize the advantages of the agent's autonomous function. This mode aims to prevent the human manager from issuing instructions all the time, and it is expected to encourage the managers to change their instruction policy.

6.2.3 Results

The results of experiment 2 are shown in Table 3; 25 M-mode trials, 25 L-mode trials, 24 R-mode trials, 19 B-mode trials, and 8 O-mode trials were recorded. In total, 102 log files, 1485 min of screen-captured video files, and corresponding voice and videotaped data were recorded.

Table 4 Subjective evaluation for the behavior changes in experiment 2

Rank item	Rank score	
	M	SD
Change in agent's behavior (Q1)	5.08	1.24
Change in instruction behavior (Q2)	4.96	1.4

Note. Q1: "How much do you think the agent changes its behavior?" Q2: "How differently do you instruct the waiter at the beginning and at the end?" (7 level rank, 1-low, 7-high)

The answers to questions Q1 and Q2 are shown in Table 4; 44% (11 out of 25) participants chose the answer "Because the agent changed its actions," and another 6 chose the answer "Because I could predict the agent's behavior." It suggests that the participants realized both the changes in the behavior of the agent and in their instructions.

A repetitive instruction is considered to be one index to express a manager's activity type and his/her focus of instruction. We classified participants according to the number of times they used repetitive instructions. Repetitive instructions imply consecutive instructions with an interval of no more than 1 s. Figure 4 shows the number of times each participant used instructions. Accordingly, participants are classified into 3 types in this experiment. Participants N4, N5, N13, N15, N17, and N18 are classified into type-1 (active type) because they issued more than 400 instructions; participants N1, N3, N6, N7, N8, N9, N10, N16, N19, N22, and N25 are classified into type-2 (average type) because they issued between 200 and 400 instructions; and participants N2, N11, N12, N14, N20, N21, N23, and N24 are classified into type-3 (passive type) because they issued less than 200 instructions. The thresholds of 200 and 400 are chosen empirically.

The log files recorded the target cell of the agent together with the ordinal numbers of buttons that were pressed by the manager (instructor). The target cell is the position indicated by a button to which the manager intends to instruct the waiter agent to move. It can be one of 11 buttons, namely, the entrance, kitchen, or nine seats. The waiter agent in the current system is designed to ignore autonomous actions to follow the manager's instruction when the manager presses a button to issue an instruction. However, the value of the agent's target cell does not change at the moment when the button is pressed; it changes after the button is pressed. In other words, at the moment the button is pressed, it is still the previous target cell of the agent's autonomous action. A successful hit is recognized only if the current target cell is the same as the manger's intended seat number. The results of the hit rate of autonomous agents in experiment 2 are summarized in Table 5.

Because mainly four types of instructions (guide, order, clear, and eat) are observed, the number of times each type

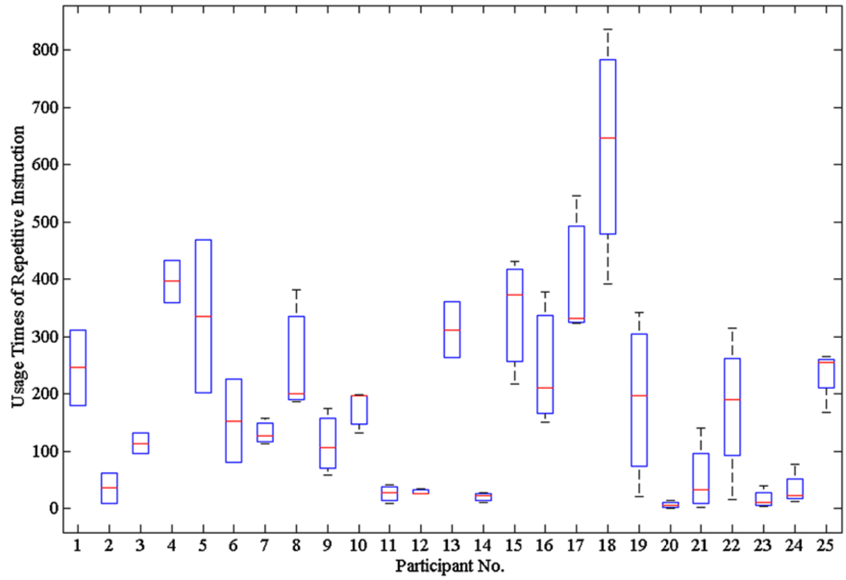


Fig. 4 Number of times participants used repetitive instructions in Experiment 2

Table 5 Hit rate of autonomous agent in experiment 2

Mode	Trial No.	Statistic	Instruction times	Hit times	Hit rate
L-Mode	25	M	325.28	218.32	.661
		SD	183.01	130.65	.096
B-Mode	19	M	323.11	226.63	.661
		SD	171.45	149.31	.105
R-Mode	24	M	334.83	227.17	.637
		SD	204.56	173.15	.1141
O-Mode	8	M	346.63	271.75	.731
		SD	260.37	223.21	.108

Note. L: Linear prediction mode, B: Bayesian network mode, R: Random mode, O: Observation phase mode

of instruction is used is summarized and the rate for all types of instructions are plotted in Fig. 5. In the current task of this experiment, the "eat" instruction is useless, however, we included it in the figure because some participants still pressed the button for seat in the "eating" state to issue an "eat" instruction. In this figure, the changes in instruction rate are roughly divided into three stages by time. The instruction rate can be considered to express the focus of instruction of the manager. As shown in the figure, the manager (N13) changed the focus of instruction over time. In the first stage, the manager mainly focused on the "order" instruction. Because the agent's autonomous function on "placing order" worked well, N13 decreased the focus on the "order" instruction and changed focus to the "clear" instruction. In the last stage, the focus of instruction shifted to "order" again; however, it stayed at a relatively low level as compared to the first stage. Although the rate of the "clear" instruction decreased, it remained at a slightly higher level than in the first stage. Figure 6 shows the result of instruction frequency for the same trial in greater details.

By referring to the screen-captured video data, we found that the interactions between the manager's instruction and agent's action differed in three stages of the trial. In the first stage, because the manager preferred "order" to "guide," more attention was paid to the "order" instruction. Even when a customer was waiting at the entrance, the manager still issued an "order" instruction. Because the agent worked well at guiding customers to a vacant seat, the manager decreased the frequency of issuing the "guide" instruction. Because the agent was initially designed to not "clear" the seat actively, the manager changed the focus of instruction to the "clear" instruction gradually in the second stage. Initially, the manager increased the frequency of instruction (double clicking or continuously pressing the button) to stress the importance of the instruction. When the agent improved its performance of finishing the "order" and "clear" tasks, the manager changed the mental model of the agent and decreased the frequency of the "clear" instruction. This led to a relatively similar rate among the three instructions in the third stage.

In contrast, the second example is the result of changing the instruction frequency of participant N14 of type-3 (passive type with low hit rate) in the B-mode, as shown in Fig. 7. It is quite different from that of N13. In this case, the instruction frequency gradually decreased, but a "guide" instruction was seldom issued to the waiter agent, and in the end, only the "clear" instruction remained at a relatively low frequency. This manager appears to trust the waiter agent to a great extent. Due to the default low priority of the "clear" instruction of the agent, the relatively high frequency of the "clear" instruction remained at the end of this trial.

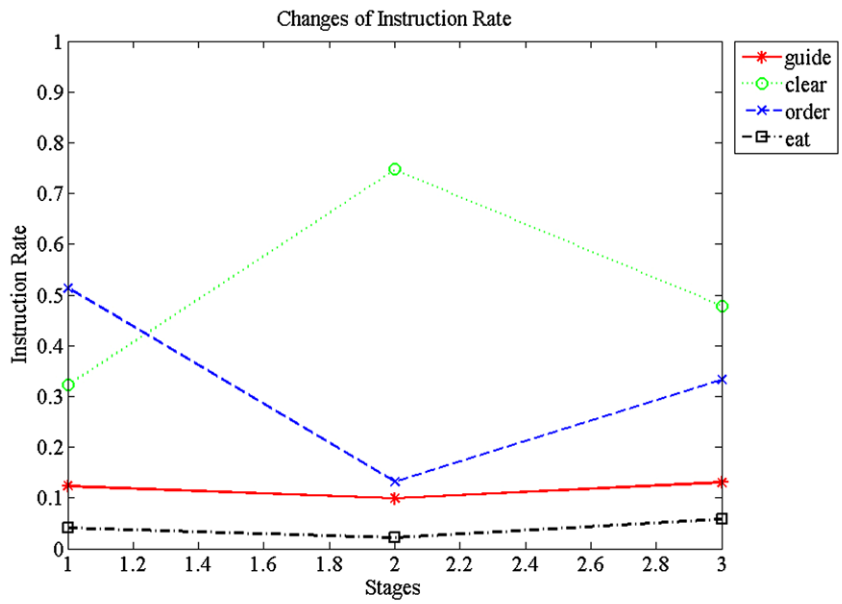


Fig. 5 Changes in instruction rate in B-mode of N13 (active type)

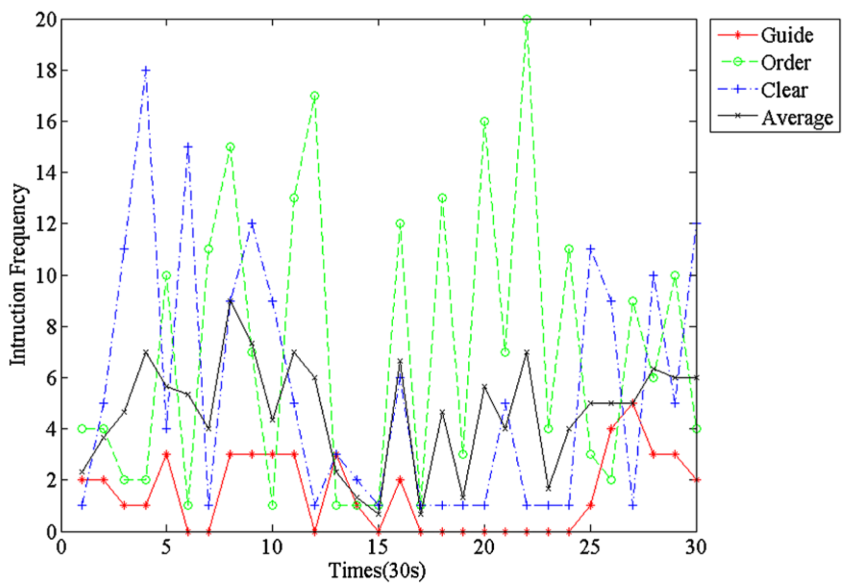


Fig. 6 Changes in instruction frequency in B-mode of N13 (type-1, active type, time window is 30 s)

In the third example, the result of participant N24 of type-2 (average type) in the B-mode is shown in Fig. 8. It appears that no obvious difference exists among three types of instructions. This manager started with a low instruction frequency but found that the waiter agent did not work perfectly; therefore, the manager then increased the frequency of repetitive instructions (continuous button clicking action). Because this manager did not have a strong preference among the three types of instructions, and in particular, he/she did not pay more attention to the "clear" instruction, it resulted in an average distribution of instruction frequency. Figure 9 shows the changes in instruction rate in the three stages.

A comparison of these three examples indicates that the three types of instructions (guide, order, and clear) were learned not separately but simultaneously by the autonomous agent in this experiment. This suggests that the switching of instruction focus by the manger is followed by the waiter agent, and the agent's action also caused changes in the instructions issued by the manager. In other words, an adaptation loop might be established in some of the trials.

6.3 Change point detection

The results of the experiment include log files that record time stamps, agent states, seat states, and button numbers. In order to analyze the log data, it is necessary to choose a suitable method for analyzing time series data consisting of multi-dimensional vectors.

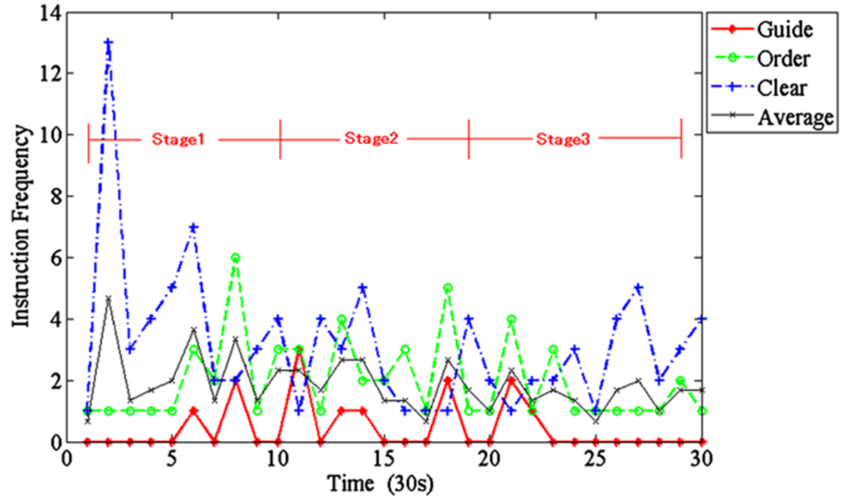


Fig. 7 Changes in instruction frequency in B-mode of N14 (type-3, passive type, time window is 30 s)

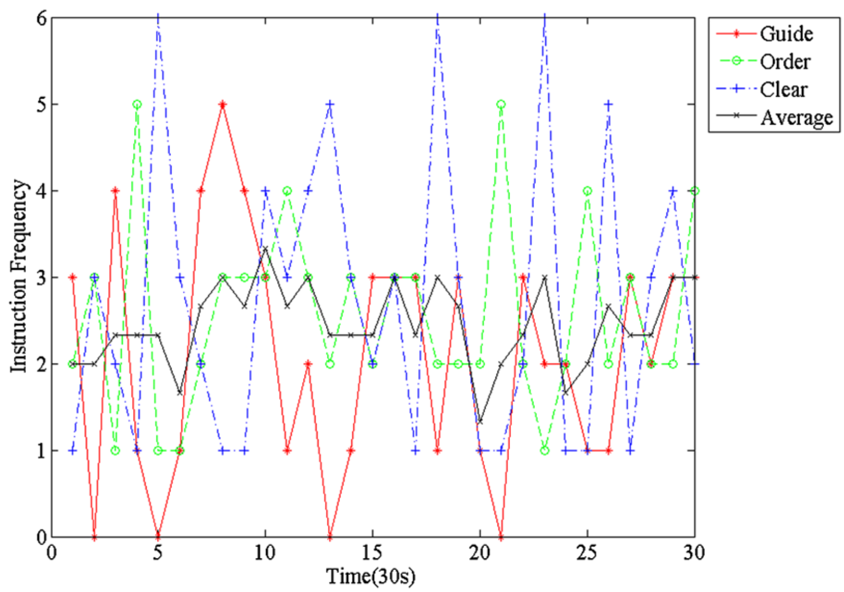


Fig. 8 Changes in instruction frequency in B-mode of N24 (type-2, average type, time window is 30 s)

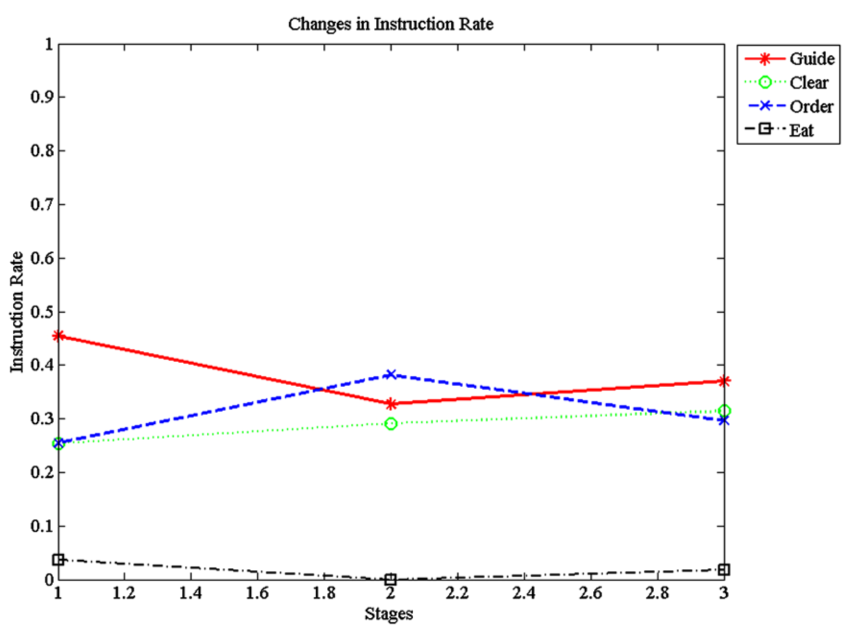


Fig. 9 Changes in instruction rate in B-mode of N24

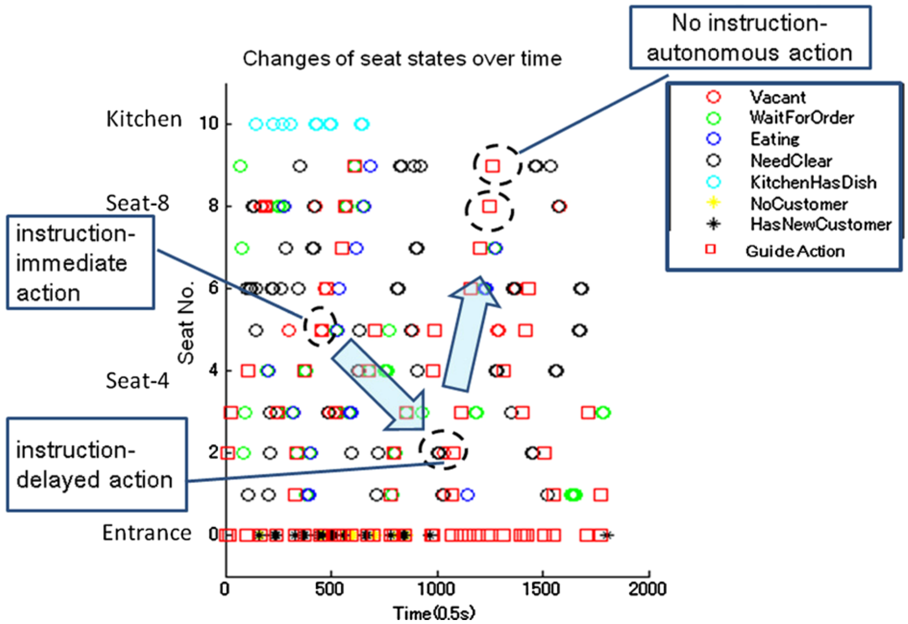


Fig. 10 Seat state distribution with guide instruction of N18 in B-Mode and guide action of agent

The time series instruction data and the time series action data reflect the manager's instruction intention and the agent's action respectively. A general time series change point detection algorithm called SST (Singular Spectrum Transform) [2] is a popular technique for detecting the change point in a time series. In this work, RSST (Robust Singular Spectrum Transform) [10], an improved version of SST, was adopted. In this algorithm, as long as parameter N (the number of previous windows to which it referred) and parameter W (the width of time window) can be determined, the change point in the time series data can be detected with high accuracy.

As the result of the implementation of the experimental environment, the human manager decides instructions by following the sales maximization policy, whereas the waiter agent acts by following its tip-first law. Neither the manager nor the agent knows the rules of the other.

Next, we will introduce an example of participant N18 in the B-Mode. Figure 10 shows how participant N18's instructions and the agent's actions change over time. Figure 11 shows the result of change point detection by processing the time series data of N18's instructions using RSST algorithm ($W = 3$, $N = 3$). Figure 12 shows the result of change point detection by processing the time series data of agent's actions using RSST algorithm ($W = 3$, $N = 3$).

6.4 Motif discovery

In preceding sections, we mentioned some results of the consequent behaviors of the mutual adaptation phenomenon. Although these results can be helpful to distinguish the occurrence of the mutual adaptation phenomenon, they cannot yet provide enough evidence to prove the causality between the human's instructions and the agent's actions. In this section, we will try to discover some motives from the experimental log data. The results of the experiment include log files that record time stamps, agent states, states of seats (entrance, kitchen, and 9 seats), and ordinal numbers of pressed buttons by the manager. The WAITER system saved one record every 0.5 s. Each record can be expressed as a 26-dimensional vector, including following dimensions "session number, time stamp, changed score, total score, changed sales, total sales, changed tips, total tips, agent position, agent target, agent state, agent mode, button-pressed flag, ordinal number of pressed button, button's intention, state of seat No. 1 through No. 11 (where No. 1 denotes the kitchen cell, No. 11 denotes the entrance cell, and No. 2 through No. 10 denotes the nine customer seats)." Therefore, each trial can be expressed as a time series of vectors. For example, a 15-min trial can be expressed as a 26-dimensional vector with a length of $1800 (= 15 \times 60 \times 2)$. We used the following algorithms with the hierarchical clustering method [14] to process the time series data.

```
Algorithm 1 Convert layout vector to
decimal number
1. Open a log file
2. Read in a 11-dimensional seat
layout vector
3. Set each bit of a 11-bit binary
number according to the state of
each seat
4. If current record is not the last
one of the log file, goto step-2
5. Output the time series of
decimal number
```

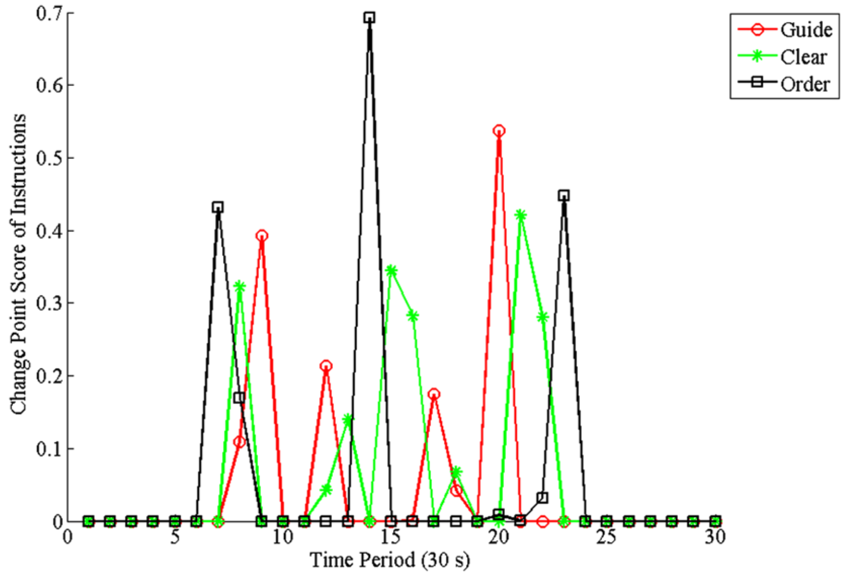


Fig. 11 RSST processing result on instruction data of N18 in B-Mode

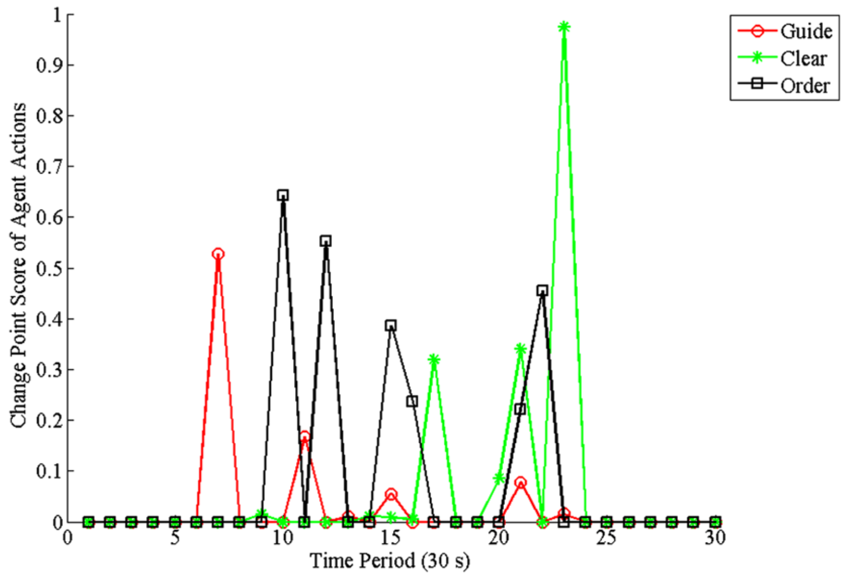


Fig. 12 RSST processing result on action data of N18 in B-Mode

```
Algorithm 2 Preprocessing
1. Open a log file
2. Read in a 26-dimensional vector
3. Call Algorithm 1 to convert each
'11-dimensional layout vector'
into a decimal 'layout number'
4. Convert 'agent position' into
'agent position area'
5. Output the time series of
4-dimensional vector
(layout number, agent position
area number, instruction type,
and instruction No.)
```

```
Algorithm 3 Clustering
1. Set width of sliding window to n
(n=3)
2. Generate time sequence TsL by
sliding window on seat layout
4. Generate time sequence TsR by
sliding window on agent position
5. Set parameters cutoff1 (0.9) and
cutoff2 (0.9)
6. Clustering layout sequence with
cutoff1 by 'hamming' distance
7. Clustering agent position sequence
with cutoff2 by 'hamming' distance
8. Output the time sequence TsResult
including seat layout cluster,
agent position cluster, instruction
type, and instruction number
```

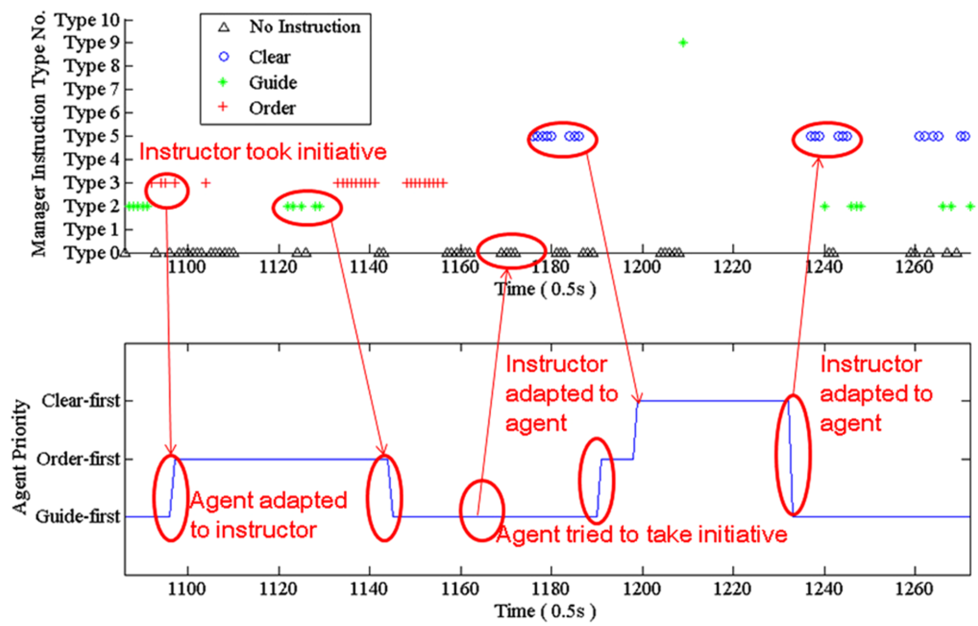


Fig. 13 Change in instruction type and agent priority of N18 in L-Mode

```
Algorithm 4 Motif discovery
1. Load time sequence TsResult
2. Generate a table TableLA for
existing 'layout-agent position'
pair
3. Generate a table TableInst for
existing 'instruction type-
instruction number' pair
4. Divide TsResult into sections by
assigning a section number for
each record according to the
next instruction
5. Count co-occurring times of
'instruction type-instruction
number' pair and 'layout-agent
position' pair with and without
counting repetitive instructions
6. Set frequent times threshold
(default value =10) and output
the motives (most frequent pairs)
```

On the basis of the clustering results, we got the most frequent motives—combinations of "layout-agent position" pair and "instruction type-instruction number" pair for each candidate trial. For example, we selected one sample from the results and extracted the corresponding instructions that were used in the most frequently used situation pair "layout-agent position."

The result is plotted in Fig. 13. We also plotted the moment when the agent switched its priority policy on the same figure. As is evident in the figure, the mutual adaptation phenomenon can be observed. At first, the human manager took the initiative by issuing some "order" instructions after issuing some "guide" instructions. In order to adapt to the instructor, the agent switched its priority from "guide-first" (moment 1086) into "order-first" (moment 1096). Then, the instructor issued "guide" instruction and the agent adapted to the instructor by switching to the "guide-first" again. After that, the human manager stopped issuing any instructions when facing the same situation (layout sequence and agent position area sequence) for a short period of time. It could be considered that the instructor allowed the agent to take the initiative during this period. Because the instructor issued "clear" instruction at relatively high frequency, the agent tried to take the initiative by switching to the "order-first" temporarily. It implies that the instructor took the initiative at this moment. As the instructor kept issuing "clear" instruction after that, the agent switched into "clear-first" to adapt to the instructor. As the agent was designed to not actively take "clear-first" priority. After a while, the agent tried to take the initiative by switching its priority to "guide-first." Then, the instructor increased the interval between consecutive "clear" and "guide" instructions to adapt to the agent. From this example, we can find that both the instructor and the agent can take the initiative and adapt to each other simultaneously. Therefore, it is considered to be a typical example of the mutual adaptation phenomenon.

6.5 Discussion

We evaluated the design of experimental environment called WAITER and confirmed the occurrence of the mutual adaptation phenomenon in experiment 1. Then, we obtained learning data for building a Bayesian network model for experiment 2. From the result of experiment 2, we found that the mutual adaptation phenomenon may not depend on a specific adaptive algorithm. We found that some instructors tried to change the agent's action by changing their instruction policy or changing the instructions frequency and the timing of button-pressing actions. This could also imply that they tried to send feedback to the agent. Because the interval between the instructions of some participants and the action of the agent was observed to change and the agent could take autonomous actions, it can be considered that the human instructor allowed the agent to take the initiative after realizing that the agent can autonomously finish the task efficiently. Further, some participants were observed to take back the autonomy when they found that the action of the agent was not satisfactory. In other words, both the instructor and the agent could adapt to each other by transferring the control. By analyzing the time series data, we found that both the instructor and the agent adapted to each other simultaneously, and referring to accumulated history facilitated the adaptation because the agent with a Bayesian network mode could adapt well. Some instructors changed instructions when faced with a similar situation, and the agents changed priority simultaneously. On the basis of the questionnaire results, we confirmed that the participants realized both the changes in the behavior of the agent and in their instructions.

Although the agent was designed to change immediately to respond to the human user's instructions, there is a time lag between the changes in the agent's action and the changes in the participant's instruction; this time lag may make it difficult for the participants to recognize changes in the autonomous agent. In the L-mode, the linear prediction function enables the agent to trace the user's intention by switching its priority among "guiding a customer first," "placing an order first," and "clearing a seat first"; however, the participants appeared to be unsatisfied by the agent's reaction. The B-mode was implemented based on previous experimental results; it did not exhibit a significant difference from the results of other modes. This suggests that simple mode switching may not be sufficient to satisfy the manager's requirement.

Although it is useless to press the "eat" button in the current task, some participants still used this type of instruction with the intention of "staying at this place." Furthermore, some participants pressed the "eat" button by mistake immediately after the seat changes its state. This type of instruction should be currently classified as noise. These two types of instructions cannot be distinguished. Further study is required on this issue.

The user's implicit intention or interest may have a hierarchical structure, and it might be useful to build a user model by using a UIH (user interest hierarchy) [4] so that the user's intention or interest can be handled in a proper way, thus enabling the agent to adapt to the user in a better way.

In addition, the current version has some limitations, such as limited channel of manager's instructions, insufficient expression of agent's internal states, and unclear method for evaluation of the agent's learning result.

7 Conclusion and future work

In this paper, we introduced the concept of mutual adaptation and designed a human-agent collaborative task to explain this concept. In order to induce and observe the mutual adaptation phenomenon, we hypothesized the conditions of the mutual adaptation phenomenon and developed an appropriate experimental environment "WAITER." The agents and humans can access different part of information about the task. Implemented with a simple adaptive algorithm such as a linear prediction or a Bayesian network model, the adaption of the agent that causes the human instructor change some properties of their instruction, such as the frequency of instructions, the time intervals between instructions, and the focus of instructions. Using this environment, we conducted two human-adaptive agent communication experiments to study the mutual adaptation phenomenon and confirmed its conditions. In the task of a waiter agent, the mutual adaptation phenomenon was observed in some of the trials with respect to some different types of participants.

The switching of the instruction focus and of the initiative were observed as well. The results verified that the difference of ability between humans and agents, the ability of noticing the adaptive competence of the partner, the ability to take initiative and the ability to influence the partner are considered to be approximately sufficient for the occurrence of the mutual adaptation phenomenon.

Both the human instructor and the agent actor were observed to change their instructions or actions and to take initiative during the interactive collaborative task. The experimental results partly verified the proposed hypothesis.

It must be greatly useful to have a comparative analysis between the mutual adaptation and one sided adaptation (or one way adaptation). However, current work mainly focuses on clarifying the conditions inducing the mutual adaptation phenomenon. Therefore, the comparative analysis with one sided adaptation will be one of key points of this research in the next step.

Because this research mainly focuses on clarifying the inducing conditions for the occurrence of the mutual adaptation phenomenon rather than inventing a new learning algorithm. We implemented several types of adaptive agents with existing algorithms and found that the occurrence of the mutual adaptation phenomenon did not depend on a specific learning algorithm. It suggests that as long as the proposed conditions are satisfied, the mutual adaptation phenomenon can be induced.

This research is an exploratory study to clarify the concept of the mutual adaptation phenomenon that has not been intensively discussed in previous works. Maybe it is not very convincing to draw a strong conclusion on the basis of experimental data of 25 participants, the results of the experiments still suggest that the mutual adaptation phenomenon may be induced when proposed conditions are satisfied. This finding will be important for comprehensive studies in future. Even our result may be insufficient to strongly support our hypothesis, it should be helpful to clarify the concept of the mutual adaptation phenomenon.

The change point detection is considered to be an extremely important method in HAI. Current results suggest that the change point detection algorithm has great potential to detect the point when human users' intentions change. In the future, we need to conduct further comprehensive studies that can draw more general conclusions.

In the domains of developmental robotics, it should be useful for robots to evolve and modify their behaviors that enable them to solve a variety of relevant problems. An autonomous mental development approach [15] may be a potential way to develop an autonomous intelligent agent.

References

1. Ana I, Paloma M, Ricardo A, Fernando F (2009) Learning teaching strategies in an adaptive and intelligent educational system through reinforcement learning. Appl Intell 31(1):89–106
2. Ide T, Inoue K (2005) Knowledge discovery from heterogeneous dynamic systems using change-point correlations. In: The SIAM international conference on data mining (SDM 05), pp 571–576
3. Kato R, Yokoi H, Arieta AH, Yu W, Arai T (2009) Mutual adaptation among man and machine by using f-MRI analysis. Robot Auton Syst 57(2):161–166
4. Kim HR, Chan PK (2008) Learning implicit user interest hierarchy for context in personalization. Appl Intell 28(2):153–166
5. Komagome D, Suzuki M, Ono T, Yamada S (2006) A design of robot meme cultural learning, transmitting and creation by human-robot mutual-adaptation. IPSJ SIG Notes ICS (131):7–12
6. Komatsu T, Utsunomiya A, Suzuki K, Ueda K, Hiraki K, Oka N (2005) Experiments toward a mutual adaptive speech interface that adopts the cognitive features humans use for communication and induces and exploits users' adaptation. Int J Hum-Comput Interact 18(3):243–268
7. Komatsu T, Ohmoto Y, Ueda K, Okadome T, Kamei K, Xu Y, Sumi Y, Nishida T (2008) Definition of mutual adaptation processes based on Akaike's information criterion. In: The 7th international workshop on social intelligence design (SID 2008), Puerto Rico
8. Kon H, Miyake Y (2005) An analysis and modeling of mutual synchronization process in cooperative tapping. J Hum Interface Soc 7(4):61–70
9. Martin GR, Jorge M, Agueda V, Marcos GG (2009) Improving accessibility with user-tailored interfaces. Appl Intell 30(1):65–71
10. Mohammad Y, Nishida T (2009) Robust singular spectrum transform. In: Next generation applied intelligence (the twenty second international conference on industrial, engineering & other applications of applied intelligent systems (IEA/AIE 2009)) Taiwan. LNAI, vol 5579. Springer, Berlin, pp 123–132
11. Ogata T, Sugano S, Tani J (2005) Open-end human-robot interaction from the dynamical systems perspective: mutual adaptation and incremental learning. Adv Robot 19(6):651–670
12. Thomaz AL, Breazeal C (2008) Teachable robots: Understanding human teaching behavior to build more effective robot learners. Artif Intell 172:716–737
13. Ueda K (2006) Mutually adaptive learning: a time factor in human-agent interaction. J JSAI (Japanese Society for Artificial Intelligence) 21(6):675–680
14. Ward JH Jr (1963) Hierarchical grouping to optimize an objective function. J Am Stat Assoc 58(301):236–244
15. Weng J, McClelland J, Pentland A, Sporns O, Stockman I, Sur M, Thelen E (2001) Autonomous mental development by robots and animals. Science 291:599–600
16. Xu Y, Ohmoto Y, Ueda K, Komatsu T, Okadome T, Kamei K, Okada S, Sumi Y, Nishida T (2008) Two-layered communicative protocol model in a cooperative directional guidance task. In: The 7th international workshop on social intelligence design (SID2008), Puerto Rico
17. Xu Y, Ueda K, Komatsu T, Okadome T, Hattori T, Sumi Y, Nishida T (2008) Woz experiments for understanding mutual adaptation. AI Soc 23(2):201–212
18. Yamada S, Koh K (2003) Idea: Interaction design for adaptation. J Jpn Soc Fuzzy Theory Intell Inform 15(2):185 (in Japanese)
19. Yamada S, Yamaguchi T (2004) Training aibo like a dog. In: The 13th international workshop on robot and human interactive communication (ROMAN-2004), Kurashiki, Japan, pp 431–436
20. Yamada S, Yamaguchi T (2005) Mutual adaptation of mind mappings between a human and a life-like agent. J Jpn Soc Fuzzy Theory Intell Inform 17(3):289–297

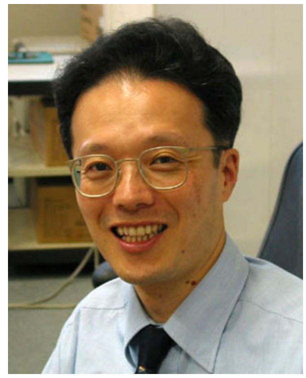
Yong Xu graduated in Computer Software and Application from Suzhou Vocational University in 1993, received his Master's Degree in Circuits and Systems from the Department of Electronic Science and Technology, University of Science and Technology of China in 2001, and received his Ph.D. in Informatics from Kyoto University, Japan, in 2010. He is currently an assistant professor at Division of Advanced Information Technology and Computer Science, Institute of Engineering, Tokyo University of Agriculture and Technology, Tokyo, Japan. His research interests include mutually adaptive interface, human-robot interaction, and human-agent interaction.

Yoshimasa Ohmoto is an assistant professor at Kyoto University. His research interests include human-agent interaction, man machine interface and cognitive science. He received his Ph.D. in Arts and Sciences from the University of Tokyo. He is a member of the Cognitive Science. Contact him at Graduate School of Informatics, Kyoto University, ohmoto@i.kyoto-u.ac.jp.

Shogo Okada received the B.S. degree in engineering from Yokohama National University, JAPAN, in 2003, and the M.S. and Ph.D. degrees in engineering of computer science from the Tokyo Institute of Technology, JAPAN in 2005 and 2008, respectively. From April 2008, he was Project Assistant Professor in the Department of Intelligence Science and Technology, Graduate School of Informatics, Kyoto University. His research interests include pattern recognition, and neural networks, autonomous robotics.

Kazuhiro Ueda is a professor at the University of Tokyo. He received his B.A. degree from the Faculty of Liberal Arts and Science at the University of Tokyo in 1988. He also received M.A. and Ph.D. degrees in cognitive science from the Department of General System Studies at the University of Tokyo in 1990 and 1993. His current research interests include cognitive analysis on scientific problem solving and collaboration, behavioral economics, cognitive analysis of skill acquisition, cognitive neuroscience approach to social cognition and perception such as eye-gaze perception, and cognitive robotics.

Takanori Komatsu is an assistant professor at international young researcher empowerment center, Shinshu University, Japan. He received a B.E. degree in mechanical engineering from Shibaura Institute of Technology, Japan in 1997, and a Ph.D. degree in cognitive and computer science from University of Tokyo, Japan in 2003. His research interest is about human-computer interaction and human-agent interaction, especially utilizing human's cognitive features for establishing smooth interaction between artifacts and users.



Takeshi Okadome is a professor of School of Science and Technology, Kwansei Gakuin University in Hyogo, Japan. His research interests focus on creating contents for the real-world events using sensors. He received the Doctor of Science degree in Computer Science from the University of Tokyo in 1988. During 1988–2009, he stayed in NTT Laboratories as a computer scientist. except two year stay at the planning section in Advanced Telecommunications Research Institute International (ATR). He is a member of the ACM. Contact him tokadome@acm.org.

Koji Kamei is a researcher at ATR Intelligent Robotics and Communication Laboratories in Kyoto, Japan. His research interests focus on sharing of real-world events, including event extraction from sensors and ontology construction for real-world events. Kamei received his ME in electronics and communication engineering from Kyoto University.

Yasuyuki Sumi is an associate professor in the Graduate School of Informatics at Kyoto University. His research interests include intelligent user interface, understanding of human interactions, and experience medium that facilitate human collaborative capturing and sharing their experiential knowledge. Sumi received his MEng and DEng in information engineering from the University of Tokyo.

Toyoaki Nishida is Professor at Department of Intelligence Science and Technology, Graduate School of Informatics, Kyoto University. He received the B.E., the M.E., and the Doctor of Engineering degrees from Kyoto University in 1977, 1979, and 1984, respectively. His research centers on artificial intelligence and human computer interaction. He founded an international workshop series on social intelligence design in 2001. He opened up a new field of research called Conversational Informatics in 2003. Currently, he leads several projects on social intelligence design and conversational informatics. His representative work may be found in Nishida (ed.) Conversational Informatics—An Engineering Approach, Wiley, 2007. He serves for numerous academic activities, including the president of JSAI (Japanese Society for Artificial Intelligence), an associate editor of the AI & Society journal, an area editor (Intelligent Systems) of the New Generation Computing journal, a technical committee member of Web Intelligence Consortium, and an associate member of the Science Council of Japan.