Dylan F. Glas, Takayuki Kanda, Hiroshi Ishiguro, and Norihiro Hagita, Teleoperation of Multiple Social Robots, in *IEEE Transactions on Systems, Man, and Cybernetics -- Part A: Systems and Humans*. Vol. 42, No. 3, pp. 530-544, May 2012.
This is an author's preprint version. The final manuscript is available at DOI: 10.1109/TSMCA.2011.2164243

# Teleoperation of Multiple Social Robots

Dylan F. Glas, Takayuki Kanda, *Member, IEEE*, Hiroshi Ishiguro, *Member, IEEE,* Norihiro Hagita, *Member, IEEE*

***Abstract*— Teleoperation of multiple robots by a single operator has been studied extensively for applications such as search and navigation; however, this concept has never been applied to the field of social, conversational robots. In this paper we explore the unique challenges posed by the remote operation of multiple social robots, where an operator must perform auditory multitasking to assist multiple interactions at once. It describes the general system requirements in four areas: social human-robot interaction design, autonomy design, multi-robot coordination, and teleoperation interface design. Based on this design framework, we have developed a system in which a single operator can simultaneously control four robots in conversational interactions with users.**

**Key elements of our implementation include a control architecture enabling the scripting of conditional behavior flows for social interaction, a graphical interface enabling an operator to control one robot at a time while monitoring several others in the background, and a technique called "Proactive Timing Control," an automated method for smoothly interleaving the demands of multiple robots for the operator's attention. We also present metrics for describing and predicting robot performance, and we show experimental results demonstrating the effectiveness of our system through simulations and a laboratory experiment based on real-world interactions.**

***Index Terms*— Adjustable autonomy, communication robots, human-robot interaction, multiple robots, networked robots, single-operator multiple-robot systems, social robots, supervisory control, teleoperation of social robots**

## I. Introduction

As rapid progress is being made on all frontiers of robotics technology, many of the key components necessary for developing socially-situated autonomous robot systems are falling into place. Field trials of social robots placed in real-world environments such as museums [1, 2, 3], schools [4, 5, 6], and train stations [7], have shown great success and provided valuable insights into real-world social phenomena which cannot be observed in the laboratory.

Yet, inspiring and exciting as it is to see robots operating in the field, the inescapable reality is that social dynamics and recognition problems are complex, and today's technology is not yet capable of supporting a fully-autonomous robot playing a humanlike role in society. Any robot will eventually find itself in unanticipated circumstances, where failure to respond appropriately could lead to socially awkward, money-losing, or even dangerous situations.

Fig. 1. A robot providing route guidance in a shopping mall.

A field trial we recently conducted at a Japanese shopping mall [9] illustrates an example of a social robot application. We placed a humanoid robot in a central public space in the shopping mall for several hours a day, where it chatted with visitors and provided information and route guidance to locations within the mall. Customers were excited by the engaging interactions, and people crowded around the robot every day, waiting for a chance to talk with it (Fig. 1).

Although a large part of the attraction of social robots is their ability to "understand" natural language and engage people interactively, this task is still largely beyond the capabilities of today's robots to achieve without a human operator. Field trials using robots in social settings have often involved some degree of remote control, referred to as the "Wizard of Oz" (WoZ) method [10, 11]. Although pure teleoperation can be valuable for studying human reactions to robot behaviors, it does not necessarily represent progress towards creating fully-autonomous or highly-autonomous social robot systems.

With real-world semi-autonomous robot applications as a goal, our long-term approach is to begin with a partially-autonomous system, and to steadily decrease the role of the operator over time with improvements in robot technology. As robot autonomy increases, it will be possible for one operator to control several robots. The operator-to-robot

Manuscript received ***. This work was supported by the Ministry of Internal Affairs and Communications of Japan.

D. F. Glas is with the Intelligent Robotics and Communication Laboratories at ATR (Advanced Telecommunications Research Institute International), Kyoto 619-0288, Japan. (Corresponding author, phone: +81-774-95-1405; fax: +81-774-95-1408; e-mail: dylan@atr.jp).

T. Kanda is with the Intelligent Robotics and Communication Laboratories at ATR. (e-mail: kanda@atr.jp).

H. Ishiguro is with the Intelligent Robotics Laboratory at the Graduate School of Engineering Science at Osaka University, Osaka 565-0871, Japan, and the Intelligent Robotics and Communication Laboratories at ATR. (e-mail: ishiguro@sys.es.osaka-u.ac.jp).

N. Hagita is with the Intelligent Robotics and Communication Laboratories at ATR. (e-mail: hagita@atr.jp).

This paper is an extended version of a conference paper [8], with the addition of a discussion of system architecture, new experimental results, and an analysis of system performance through simulation.

DOI: 10.1109/TSMCA.2011.2164243

ratio could be considered as one measure of the degree of autonomy of a robot system.

Seen from a commercial perspective, a fleet of ten service robots controlled by a single human operator would be more economically viable than the same number of robots requiring a team of twenty operators, and even highly autonomous industrial robots generally have a human in the loop in a supervisory role. Thus, while full autonomy for social robots is not yet feasible, partial autonomy with a low operator-to-robot ratio could enable social robot applications which would be otherwise impractical.

In this paper, we address the unique challenges of single-operator-multiple-robot (SOMR) operation for the case of social robots. In Sec. III we present a framework in which we categorize and discuss the key issues in designing such a system. Based on this conceptual framework, we implemented a semi-autonomous robot control system for social interactions, enabling a single operator to monitor and control several communication robots at once. The details of our implementation and solutions to key problems are presented in Sec. IV. In the remainder of the paper, we present results showing the effectiveness of our system in simulation, as well as laboratory trials demonstrating that a single operator is able to successfully control up to four robots at once as they simultaneously engage in conversational interactions.

## II. Related Work

In this paper we are exploring semi-autonomous control of multiple robots for social human-robot interaction, by which we mean conversational interaction between a robot and one or more people. In other fields of robotics, such as search-and-rescue or space exploration, many aspects of both single- and multiple-robot teleoperation are active fields of research, but multiple-robot teleoperation has not yet been studied for social robots.

A substantial amount of work has been done regarding levels of autonomy for teleoperated robots. The concept of "shared autonomy" describes a system in which a robot is controlled by both a human operator and an intelligent autonomous system, a concept which has been used in fields such as space robotics [12] and assistive robotics [13]. The concept of "adjustable autonomy", also known as "sliding autonomy," has also been studied, in which varying degrees of autonomy can be used for different situations [14]-[17].

Other teleoperation research has focused on control interfaces for teleoperation. A wide variety of teleoperation interfaces have been created for vehicle control [18],[19], and the unique problems of controlling body position in humanoid robots have also been studied [20].

Several aspects of simultaneous control of multiple robots have also been studied. Hill and Bodt presented field studies observing the effects of controlling multiple robots on operator workload [21]. Sellner et al. studied the situational awareness of an operator observing various construction robots in sequence [22], and Ratwani et al. used eye movement cues to model the situation awareness of an operator supervising several UAV's simultaneously [23].

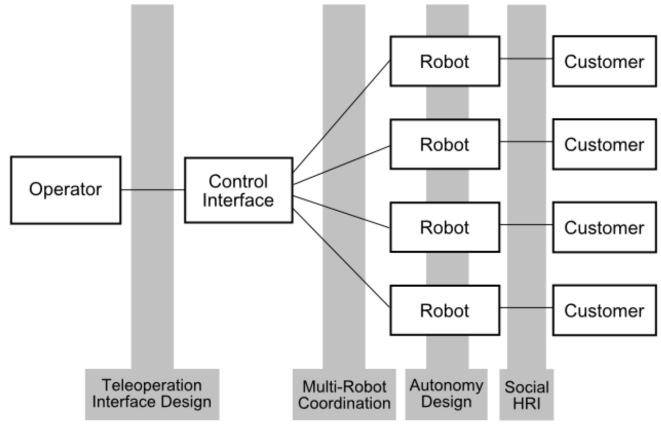


Fig. 2. General overview of multi-robot control system showing key design areas.

A key issue in multiple-robot teleoperation is the concept of "fan-out," which describes the number of robots an operator can effectively control [24]. Crandall and Goodrich have laid a theoretical basis for the modeling of SOMR teleoperation, defining metrics such as Interaction Time (IT) and Neglect Tolerance (NT) to help with calculation of robot fan-out and predicting system performance [25]. Thus far, studies of fan-out in multiple-robot teleoperation have focused on tasks such as search and navigation for mobile robots [26], or target selection for UAVs [27], but not social human-robot interaction.

In this paper we will build upon this research to define a new application domain: the teleoperation of multiple robots for social human-robot interaction tasks. In doing so, we aim to identify ways in which existing SOMR teleoperation principles can be applied to social robots, and to examine ways in which social robots differ from traditional systems.

## III. Issues in Teleoperation of Multiple Social Robots

The teleoperation of multiple robots for social interaction is in some ways analogous to SOMR teleoperation for conventional robots, and in other ways presents new challenges. Extensive research has been done on teleoperation for tasks such as robot navigation, and we have summarized how the issues in teleoperation for conversational social interaction differ from those regarding many kinds of teleoperation for navigation (Table 1). Of these differences, perhaps the most significant is the time-critical aspect of conversational interaction. Time-criticality itself is not unique to social robotics, and time-critical tasks exist, for example, in UAV teleoperation; however in most SOMR systems, robots can buy time by "idling" or "loitering" until an operator becomes available, whereas a robot waiting in silence during a conversation would quickly cause failure of the interaction. Thus time-criticality is a central factor affecting several of the key issues in teleoperation of multiple social robots.

Fig. 2 shows the general organization of a SOMR system for social human-robot interaction. In this paper we will use the terms "operator" and "customer" to describe the roles of

DOI: 10.1109/TSMCA.2011.2164243

TABLE 1. DIFFERENCES IN TELEOPERATION BETWEEN NAVIGATION (FUNDAMENTAL TASKS FOR MOBILE ROBOTS [28]) AND SOCIAL INTERACTION.

| | Navigation | Social interaction (this study) | New problems in social interaction |
|---|---|---|---|
| Operator's role | Obstacle avoidance. Giving current position, path, goals. | Understanding the user's intention and providing required service | |
| Source of input to operator | Scenery + Map | Audition (+scenery) | Cannot monitor multiple sources simultaneously |
| Operator's output (low level control) | Velocity | Utterance, gesture, +(body orientation and position) | Typing and controlling many DOFs for gesturing are very slow |
| Operator's output (abstracted control) | Position (destination) | Behavior (combination of utterance and gesture) | Difficult to prepare for minor cases in advance |
| Consequence of ignoring errors caused by autonomy | Crash into obstacle, or lose the robot. | Person might get lost, buy wrong product, or receive wrong service. | Definitely we should not ignore errors in either case. |
| Can robots wait after an error detected? | Yes, in most cases. | No. Users might soon leave if a robot stops. | An operator should take control of the robot immediately. |
| Can robots anticipate the timing of possible error? | Not usually. | Yes | Most errors are from speech recognition, often after the robot asks a question. |

humans in the system. This choice of terms is not meant to preclude other roles of the humans in the system, e.g. teacher-student or doctor-patient, but is only used to avoid the ambiguity of the term "user".

Four key design areas are identified in the system diagram in Fig. 2. The overall system requirements are driven by the target application, which in this case falls in the domain of **social human-robot interaction**. This area includes the design of the robot's behavior and dialogue with the goal of creating comfortable, natural, and functional interactions between the robot and a customer. To create semi-autonomous robots which can do this, an important issue is **autonomy design**, that is, how operator commands can be reconciled with the autonomous components of the robot control system. Next, due to the time-criticality of social interactions, **multi-robot coordination** is necessary to manage the attention of the operator between robots, and to reduce conflicts between demands for the operator's time. Finally, **teleoperation interface design** is necessary to enable interaction between the operator and the robot, providing the operator with situation awareness and controls for operating the robot.

In this section, we will present design considerations in these four areas and propose metrics for quantifying important characteristics of SOMR systems for social interaction.

### *A. Social Human-Robot Interaction*

The target application of social human-robot interaction drives the design of the entire robot system. As the field of HRI covers a wide range of scenarios, it is important to clearly define the target of this paper.

In this study we are considering conversational humanlike interactions. The task of the robot is primarily dialogue-based, although nonverbal communication and gestures such as pointing may also be essential interaction components.

Some examples of this type of interaction might include a robot shopkeeper which provides information about various products, an information booth robot which gives directions and answers questions in a shopping mall, a tour-guide robot which explains exhibits in a museum, or a public relations robot which greets people and invites them to visit a shop.

In these examples, interactions can be expected to follow a flow which includes alternating phases: one in which a person is asking a question or giving information to the robot, and one in which the robot responds with some explanation or directions.

In the first type of phase, it is the customer's 'turn' to drive the conversation, and the robot (or operator) must correctly recognize the customer's utterances in order to respond appropriately. We call this type of phase a **critical section**, because a recognition failure in this phase is likely to result in a failure of the interaction, such as a customer becoming frustrated with the robot and walking away.

In the second type of phase, it is the robot's 'turn' in the conversation, and the customer is in a listening role. Responding to inputs from the customer is less important during this phase, which we call a **non-critical section**. This is not to say that the customer will never interrupt the robot, but such interruptions are expected to be the exception rather than the rule. Although recognition failures may occur in this phase, we assume that in comparison with critical sections, there is a lower likelihood that they will result in interaction failures.

Understanding this pattern of critical and noncritical sections defined by the social interaction design helps to enable the coordination of operator attention between multiple robots, as we will explain later.

### *B. Autonomy Design*

In semi-autonomous social robot systems, it is important to define how an operator should interact with the autonomous components of the robot's control system. Generally speaking, an operator can direct high-level tasks or identify errors that the system cannot detect autonomously. For social robots, many necessary functions, such as tracking human positions or presenting information through speech and gesture, can be performed autonomously using available technology. Some

DOI: 10.1109/TSMCA.2011.2164243

core background processes, such as emotional dynamics, can also be automated for social robots [29]. It is in the recognition and interpretation of verbal and nonverbal communication and the ability to make common-sense judgments based on an understanding of context that an operator can add the greatest value.

For example, an elderly person in a shopping mall who is holding a map and looking around might need route guidance from the robot; on the other hand, a young person in a plaza looking around in a similar manner might just be looking for friends and not need the service. Although a human operator could easily distinguish between these two cases using intuition, visual cues, and implicit social context, such a recognition task would be quite difficult for a robot to perform autonomously.

An operator can provide input to a semi-autonomous system at several levels. Consider a simple framework for robot control, in which developers create sense-plan-act elements based on a pre-assumed world model. Fig. 3 shows an example of such a system, in which the robot can perform abstracted behaviors composed of low-level actions such as speech and gesture. These behaviors are chosen by decision logic, based on the results of autonomous sensor recognition.

In such a system, three categories of problems tend to occur, which define the three primary tasks of an operator.

*1) Uncovered situations*

The richness and diversity of human behavior makes it difficult to create a predictive model of the world for social interactions. This can lead to many **uncovered situations**, in which a robot does not have appropriate rules or behaviors implemented to act autonomously. Uncovered situations are of particular concern for systems which interact with humans.

Uncovered situations motivated the original use of WoZ, where a dialog system was controlled by a human operator to collect necessary dialog elements [30]. By monitoring the interaction, an operator can provide additional information to the system and improve its world model. The assumption behind this technique is that the robot can ultimately cover all situations after collecting a sufficiently complete world model.

*1) Incomplete autonomy*

Even assuming a good model of the world, there are still cases when we cannot prepare all the necessary sense-plan-act elements. In these cases an operator can be used as a substitute for **incomplete autonomy** and replace those individual elements. Many WoZ studies in HRI are of this type [31].

An example of replacing a *sense* element is speech recognition. Today's speech recognition technologies are unreliable in noisy environments, as observed by Shiomi *et al.* in field trials [32]. It is thus not currently possible to automate this sensing task. However, an operator can be employed to listen to the audio stream and manually input the recognized utterances into the system. Using those inputs, the robot can still perform the plan and act elements autonomously. Other examples in this class could include identifying a person or object, or monitoring the social appropriateness of a robot's actions by observing people's reactions to the robot.

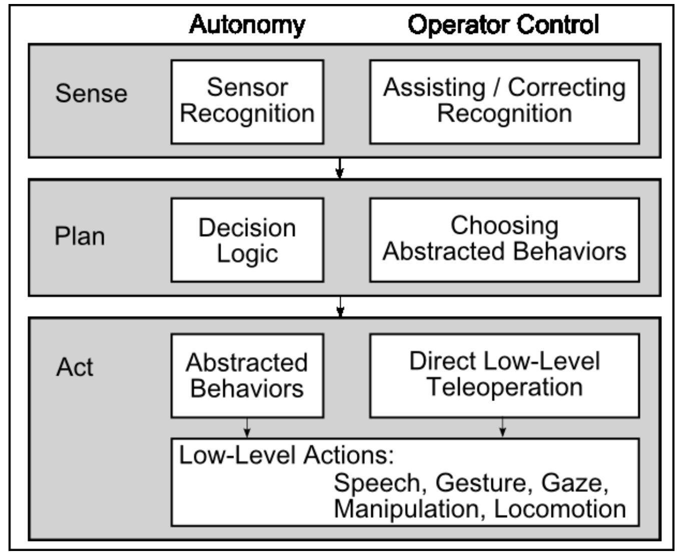


Fig. 3. Autonomy and operator control tasks for sense-plan-act elements in a semi-autonomous robot control system.

An operator could likewise replace a *plan* element. If a robot's action requires particular expertise or authority, such as that of a doctor, technician, soldier, or law enforcement officer, an operator may be required for this step. Here the robot may be able to sense the environment and act on it, but lack the authority or accountability to make the decision to act.

For replacing an *act* element, an example could be a difficult actuation task like grasping. The robot might be able to identify an object to grasp and make the decision to grasp it, but need assistance in actually carrying out the grasping task [33].

Note that for this style of teleoperation, the system can often prompt the operator to perform some action. The operator acts as a "black box" in the system, performing some defined processing task on demand, like any other module in the system.

*2) Unexpected errors*

Finally, it is possible that even if we have prepared a good world model and developed appropriate sense-plan-act elements, the system may not always work as intended. That is, **unexpected errors** may occur during autonomous operation.

In this case, an operator needs to monitor the robot to identify possible errors. In the teleoperation tasks described above, the operator's focus is on the environment and people interacting with the robot, but when monitoring for errors the primary focus is on the performance and behavior of the robot itself.

## *C. Multi-robot Coordination*

As stated in Table I, we assume an operator can only correct errors or provide active support for one robot at a time. Particularly in the case of speech recognition, it is extremely difficult for an operator to concentrate on two or more conversations at once. This restriction makes the operator's attention a limited resource.

In this paper we will model a robot's interaction as consisting of **critical sections**, where there is a high risk of interaction failure and thus a high likelihood that operator assistance will be needed, and **non-critical sections**, which can safely be performed autonomously. Critical sections tend to occur when

DOI: 10.1109/TSMCA.2011.2164243

the actions of the robot depend strongly on recognition of inputs from the customer, and thus the consequences of a recognition error are severe. Critical sections can also occur when there is a high probability that an uncovered situation will arise.

Note that we consider errors in sensor recognition to be equally likely to occur in critical and non-critical sections. However, a recognition error is much more likely to result in interaction failure in a critical section than in a non-critical section. To prevent interaction failures, it is desirable for an operator to be monitoring a robot during critical sections.

A fundamental conflict arises when two or more robots compete for operator attention by entering critical sections at the same time. As noted above, social interactions are time-critical. While the operator serves one robot, the customer interacting with the other robot is made to wait, which will have a negative impact on the quality of service, and possibly even cause failure of the interaction.

In Sec IV-C, we will propose a mechanism for coordinating the interactions of the robots to eliminate such conflicts.

### *D. Teleoperation Interface Design*

An operator has two tasks to perform: first, supervisory monitoring of all robots to identify unexpected situations, and second, assisting individual robots' recognition, planning, and actuation. Supporting both of these tasks provides a considerable challenge for the user interface design.

Both situation awareness and actuation requirements for the user interface differ for these two tasks as follows.

#### *1) Controlling individual robots*

When controlling a single robot, the operator needs to be aware of the robot's individual situation – with whom the robot is interacting, what that person is saying, and what the robot is doing. For simple systems, such as an information-providing robot in a shopping mall, this immediate information may be sufficient for the robot's interactions. For more elaborate systems where the robot has a long-term relationship with the customer, long-term interaction history or personal information about that customer might be required.

This interface also requires actuation controls for correcting sensor recognition, directing behaviors, and performing low-level control such as entering text for the robot to speak in uncovered situations.

#### *2) Monitoring multiple robots*

When acting in a supervisory role and monitoring multiple robots, the operator needs to identify and react to unexpected problems in a timely manner. A summary of the state information about each robot should be presented to the operator in such a way as to make errors and unusual behavior easily recognizable.

As stated in Table 1, an understanding of the conversation flow can make it possible to anticipate when errors in recognition are likely to happen. The highest risk of recognition error occurs during critical sections, so alerting the operator of which robots are in or entering critical sections can help manage the operator's attention most effectively.

TABLE 2. TASK DIFFICULTY METRICS

| Metric | Comments |
|---|---|
| Recognition Accuracy (RA) | Limited by technology; higher RA increases fan-out |
| Situation Coverage (SC) | Limited by scenario predictability; higher SC increases fan-out |
| Critical Time Ratio (CTR) | Determined by interaction design; lower CTR increases fan-out. |

It should also be noted that a summary of the robot's state information might not be sufficient for the operator to accurately identify some errors, so it may be important for the operator to periodically examine the detailed state information for individual robots as well.

### *E. Task Difficulty Metrics*

Finally, it is valuable to have metrics quantifying the capability of the robot system. For multiple-robot systems, a key quantity is the number of robots a single operator can manage, known as "fan-out". High fan-out can be achieved if the robots can operate with high reliability without the support of an operator, whereas fan-out will be much lower if errors are likely to occur, for example, due to poor sensor recognition or high task difficulty. Thus, to predict fan-out, it is important to have metrics which describe the likelihood of error while the robot is unsupervised. In the terminology of Crandall and Cummings, such metrics are classified as "Neglect Efficiency" metrics [26]. In this section, we will define three neglect efficiency metrics reflecting the risk of interaction errors occurring while the robot is unsupervised. These metrics are summarized in Table 2.

#### *1) Recognition Accuracy*

Sensor recognition accuracy (RA) is a fundamental concern for robots in nearly every field. This is also true for social robots, as recognition of the nuances of communicative signals such as speech, gesture, intonation, and emotion in social interaction can be particularly challenging. An estimate of RA can help predict the frequency of unexpected errors in the "sense" element of the robot's control architecture.

The RA of a system should be evaluated in the context of its intended application. Visual recognition accuracy varies greatly with lighting conditions, and audio recognition accuracy is dependent on levels of ambient noise. Variability in interactions can also affect RA. For example, a robot may perform excellent speech recognition while answering a predictable set of questions in an office setting, yet quite poorly in recognizing the unrestricted utterances and emotional signals of children telling stories to the robot at a day-care center.

From a designer's perspective, increasing a robot's RA through better sensors or better recognition technology can reduce the need for operator intervention, which can increase the number of robots a single operator can control. The designer's freedom, however, is typically limited by available technology, and thus RA cannot be increased without bound.

DOI: 10.1109/TSMCA.2011.2164243

TABLE 3. INTERACTION SEQUENCE

| # | Phase | Criticality | Duration |
|---|---|---|---|
| 1 | Simple greeting | Non-critical | 2s |
| 2 | Self-introduction | Non-critical | 3s |
| 3 | Chat behavior | Non-critical | Variable |
| 4 | Offer guidance | Critical | 1s |
| 5 | Wait for question | Critical | 2-10s |
| 6 | Provide guidance | Non-critical | 10-15s |
| 7 | Farewell | Non-critical | 5s |

#### 2) *Situation Coverage*

The next metric we propose is Situation Coverage (SC), which describes the completeness of the "plan" and "act" elements in the robot system. We define a situation to be "covered" if the system would autonomously execute the correct behavior given perfect sensor inputs. Using this definition, SC is defined as the percentage of situations encountered by the robot that are covered.

To be precise, there are actually two aspects to SC, corresponding to the "plan" and "act" elements of the robot control system. $SC_{act}$ describes the percentage of situations encountered by the robot for which an appropriate action has been prepared. $SC_{plan}$ then describes the percentage of situations for which the decision logic has been developed which will trigger those actions.

For example illustrating the difference between $SC_{act}$ and $SC_{plan}$, consider a robot system which includes an implemented action to direct a customer to a supermarket (*i.e.* covered under $SC_{act}$). Assume decision logic has been implemented to execute this action only if a customer asks where the supermarket is. If a customer asks this robot where to buy broccoli, but the robot is not programmed to react to the word "broccoli", this situation is not covered under $SC_{plan}$, and is thus not considered to be a covered situation overall, even though it is covered under $SC_{act}$.

Overall, SC describes a theoretical limit of the system's capacity to operate autonomously. In an ideal system with no recognition errors or unexpected errors, a system with an SC of 70% will be able to successfully complete 70% of its tasks autonomously, and will require operator intervention 30% of the time. When errors are considered, real autonomous performance will fall somewhat below 70%, so SC is useful for describing the upper bound of the system's possible performance, or conversely, a lower bound on the fraction of time during which operator support may be necessary.

In application design, SC is more of a controllable variable than RA. Whereas RA is subject to technological limitations, SC can be increased through human effort. By spending more time researching potential situations the robot may encounter and developing the decision logic and actions to respond to those situations, it is possible to increase a robot's SC.

Given the complexity and variety of real social situations, it is usually impractical to attempt to achieve 100% SC. Instead, an effective strategy for use of partial autonomy would be to design logic and actions to cover the most common situations, perhaps achieving an SC of 90%, and then to rely on operator assistance for the remaining situations.

#### 3) *Critical Time Ratio*

The third metric we will introduce is the Critical Time Ratio (CTR). This is defined as the ratio of the amount of time spent in critical sections to the total duration of an interaction. For tasks with a low CTR, the likelihood of two robots entering a critical section at the same time is correspondingly low, and thus timing control behaviors will seldom be necessary. Tasks with a high CTR are more likely to conflict, which can lead to higher wait times for users and a heavier workload on the operator.

CTR is related to the concept of Robot Attention Demand (RAD) presented by Olsen et al. [34]. RAD represents the fraction of total time that human attention is required. CTR, on the other hand, only describes the pattern of critical and non-critical sections. The degree to which operator attention is required during these critical sections is dependent upon the overall risk of failure, which is in turn based on RA and SC.

For a designer, it is possible to achieve higher fan-out by creating interactions with a low CTR, e.g. by increasing the durations of non-critical sections and minimizing the number of critical sections in the interaction flow. However, this must be done carefully, as reducing CTR also runs the risk of reducing the robot's responsiveness to the customer, and thus reducing the quality of the human-robot interaction.

## IV. IMPLEMENTATION

Using the principles presented in this paper, we developed a system for the teleoperation of multiple robots for social interactions.

In this section we will present the implementation of our system, addressing each of the four design areas presented in Fig. 2: social human-robot interaction, autonomy design, multi-robot coordination, and teleoperation interface design. Finally, we will present an example of how an operator would interact with such a system while controlling multiple robots.

### A. *Social Human-Robot Interaction*

The interaction flow we developed for this study was based on interactions used in our field trials in a shopping mall. Table 3 shows the sequence of conversation phases and their durations. When the robot detected a person in front of it, it would greet the person (1), then introduce itself and explain that it can give directions to locations in the shopping mall (2). After this, the robot would briefly chat about some topic, usually related to current events in the shopping mall or the robot's "experiences" at various shops (3).

As noted earlier, critical sections include situations where a response from the user is expected, whereas non-critical sections include tasks such as greeting, talking, and giving directions, where the robot is primarily providing information. The critical sections in our flow consist of the robot asking where the customer would like to go (4), and then waiting for the customer's response (5).

After the question has been asked, the robot gives guidance (6), then says goodbye to the customer (7). All of these phases are considered noncritical.

DOI: 10.1109/TSMCA.2011.2164243

### B. Autonomy Design

For this study, we created a semi-autonomous robot control architecture which enables an operator to provide commands and assistance to an otherwise autonomous robot system.

#### 1) Robot Platform

We implemented our architecture on Robovie II, a humanoid robot platform developed for human-robot interaction research [21]. It is capable of humanlike expressions with a head that can be moved with 3 degrees of freedom (DOF), arms with 4 DOF each, eye cameras with 2 DOF each, and a wheeled base for locomotion. Each robot also has color CCD eye cameras, a microphone, and several touch sensors.

Audio from the robot's onboard microphone is processed by an automatic speech recognition (ASR) system. In our field trials we have found the ASR system to be unusable because of ambient noise from background music, crowds, and announcements. In our quieter laboratory environment we found it to be more reliable, but accuracy was still observed to be around 60%.

A common difficulty in speech recognition is that the signal-to-noise ratio goes down as distance between the microphone and the person speaking increases. Using headset microphones would certainly improve accuracy, but their use would be impractical for real robots interacting with customers in the field.

#### 2) Robot behavior control

The behavior control system used in this study uses a software framework, illustrated in Fig. 4, in which short sequences of motions and utterances can be encapsulated into discrete units called "behaviors". The programmer then defines a set of transition rules called "episodes" which specify transitions between behaviors [35]. These rules can be based on execution history, like the following examples:

- *If behavior A was executed, execute behavior B next.*
- *If behavior B was executed immediately after behavior A, then execute behavior C next.*

The transitions can also be based on return values of the behaviors. This enables us to incorporate sensor information into the transitions. For example, a "check for customer" behavior could be defined which returns a 1 if a person is detected in front of the robot, and a 0 if no one is detected. This can be used to create a simple waiting loop, as follows:

- *If behavior A returns 0, repeat behavior A.*
- *If behavior A returns 1, execute behavior B next.*

In practice, we have used this framework to create "listen" behaviors which return tens to hundreds of different values based on speech recognition results, as well as action-oriented behaviors such as a "shake hands" behavior which offers to shake hands and returns different values based on the reaction of the person.

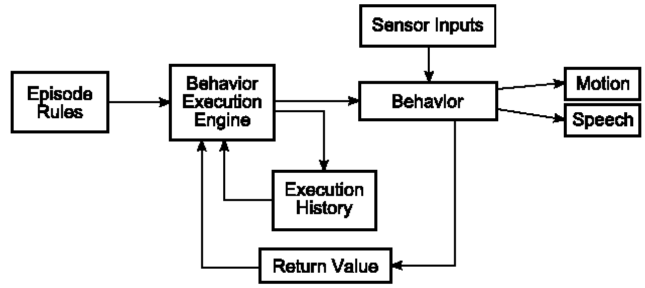


Fig. 4. Behavior execution architecture.

With this framework, if we theoretically assume no errors in sensor recognition and user behavior only within the limits of Situation Coverage, it is possible for the robot to execute any length of behavior chains with full autonomy.

#### 3) Operator intervention

As described in Sec. III-B, the operator needs to be able to intervene in robot operation to deal with uncovered situations, incomplete autonomy, and unexpected errors. This can be achieved either through direct control of the robot at a high or low level, or through correcting the robot's recognition.

##### a) Direct Control

Improvements in the efficiency of robot control can be made possible through layers of abstraction. For example, an operator could specify the individual joint angles for the robot's arm at a low level, or achieve the same result by giving the robot a high-level command, e.g. "point to the left". Most robot systems already incorporate abstractions like this. Joint angles can be grouped into poses, poses grouped into motions, motions and utterances grouped into behaviors, and so on. Similar abstractions have been used in teleoperation systems for navigation [15], [25].

As layers of abstraction are added to the system, the robot usually becomes able to function with a higher degree of autonomy, thus reducing the workload for the operator. When high-level functions are not prepared for a situation, the operator can use low-level functions instead. For example, if there is no behavior prepared for giving directions to a Japanese restaurant, an operator might directly type phrases for the robot to say and control the arms manually to point the way.

##### a) Correcting Recognition

An operator can also choose to correct a robot's sensory recognition errors, rather than completely taking over control of its behaviors. For example, an operator observes a scene where a user says the words "Japanese restaurant", but the speech recognizer fails to pick it up. If the robot has behaviors in place to react to those words, the operator can correct the robot's speech recognition results and allow the robot to complete the interaction as usual.

This kind of control requires less effort from the operator than taking over behavior control in order to generate a guiding behavior for directing the user to a Japanese restaurant.

To give a simple example, when a robot in an idling state detects a person approaching, the episode rules may trigger a transition from the idling behavior to a greeting behavior. After greeting the person, the next behavior might be to offer route guidance and wait for a response. The transition rules would then choose the next behavior based on input from the speech recognition system. If the person asked for directions to a bookstore, and if we assume the speech recognition system correctly recognized the word “bookstore”, the system would then transition to the module for giving directions to the bookstore.

### C. Multi-Robot Coordination

We will consider two possibilities for handling conflicts between robots for the operator’s attention. First, a naïve method of simply alerting the operator of critical sections and second, a technique we call Proactive Timing Control.

In the first approach, each robot can notify the operator of a critical section, and then proceed in its interactions regardless of the state of the operator or other robot(s). If the interaction reaches a point where the robot is unable to respond without operator intervention, the robot will need to stall for time [36] until the operator becomes available.

The robot can simply wait in silence, or it can repeat phrases like, “hmm… hold on… please wait” until the operator can provide assistance. The drawback of this approach is that such behavior might leave a negative impression on a user impatiently waiting for a response.

To avoid making a customer wait in this way, we propose a mechanism for handling the problem of conflicting critical sections, which we call Proactive Timing Control (PTC). This mechanism enables interactions to be coordinated in order to prevent critical section conflicts from arising at all. One means of achieving this is for each robot to send a reservation request to the operator before a critical section begins. If the operator accepts, the robot can proceed to the critical section. Otherwise, the robot performs other behaviors in order to delay entry into the critical section.

This technique changes the robot’s behavior in an important way from the customer’s perspective. When PTC is not used, the delaying behaviors are executed after the user’s “turn” in the conversation, that is, after the user has made a request or asked a question. There, the user is understood to have the initiative, and the robot is expected to react.

With PTC, however, the delaying behavior is executed before the user has spoken, while it is still the robot’s “turn” in the conversation. The robot has not yet relinquished the initiative, and thus the extra behaviors integrate more smoothly into the flow of interaction. The effectiveness of this technique has been demonstrated in a study of the effects of wait time upon customer satisfaction [37].

### D. Teleoperation Interface Design

Teleoperation interface software was developed to enable the operator to control one robot (referred to here as the “active” robot) while monitoring the others in the background. The interface used is pictured in Fig. 5. The four panels on the top left of the screen show the status of each robot, and the operator can click one to begin controlling that robot. Below those panels, the button panel on the left can be used to trigger robot behaviors. The column of buttons to the right of that can be used to correct speech recognition results, and the pop-up window on the right side shows a map of guide destinations from which the operator can trigger guide behaviors.

This interface is nearly identical to that used in [8], and further details of its functionality are explained there. Major differences from that interface include the addition of a map display of the robot’s location (lower right), the addition of a panel showing video from the robot’s eye camera (upper right) and the removal of the buttons for “reserving” a robot without switching to it, as this functionality is not particularly necessary unless PTC behaviors are very long.

### E. Example Interaction

Here we will describe an example of a typical multi-robot control session from our experiment. In this example, the operator is controlling three robots, and the system is not using PTC, *i.e.* there is no attempt to prevent conflicts between robots demanding the operator’s attention at the same time.

First, Robot 1 detects a person approaching. As it begins a greeting behavior, its Interaction Status light changes to yellow and the Countdown Timer on the robot’s status panel begins counting down until the robot expects the human to speak.

The operator clicks on the robot’s status panel to choose Robot 1 as the active robot, and the bottom half of the user interface refreshes to show Robot 1’s current status, behavior history, and speech recognition results. The audio from Robot 1 is also streamed to the operator’s headphones, and the operator listens as Robot 1 introduces itself, “My name is Robovie, and my job is giving directions. Where would you like to go?”

At this point, the operator notices that a person has approached Robot 3 as well. However, the operator stays focused on Robot 1, as its Countdown Timer is just reaching zero. The customer asks where to find an ATM. Unfortunately, due to background noise, Robot 1’s speech recognition was unable to pick up the word “ATM”, and so the operator goes to the expected phrases panel and clicks on “ATM”. Robot 1 then begins giving directions to the customer, and the operator quickly switches to Robot 3, whose countdown timer has almost reached zero.

By this time, a customer has approached Robot 2 and begins asking directions while the operator is still busy helping Robot 3. Robot 2’s Interaction Status light flashes red. By the time the operator finishes helping Robot 3, the customer talking to Robot 2 has already finished speaking. Robot 2’s speech recognition system has picked up the word “hamburger”, which is displayed on its Speech Results display, but the robot has no mapping between that word and a location in the mall. The operator quickly switches to Robot 2, opens the map, and clicks on a restaurant that specializes in hamburgers. Robot 2 then gives directions to that restaurant, as the Interaction Status indicators for Robots 1 and 3 return to green.

DOI: 10.1109/TSMCA.2011.2164243

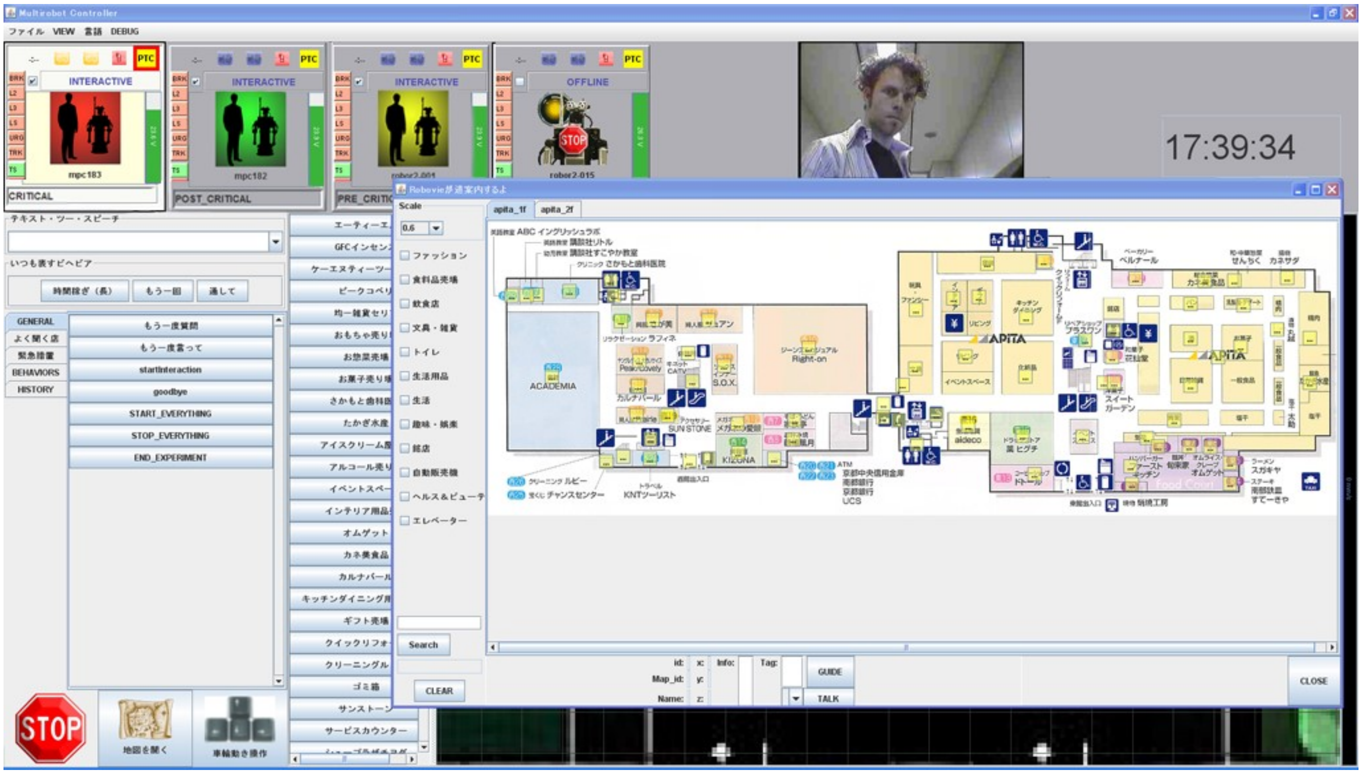


Fig. 5. Teleoperation interface. Panels in the top left show the status of each robot, and can be clicked to select that robot. The video pane on the upper right shows the video feed from that robot's eye camera. The tabbed button panel on the left sends direct commands to the robot. The column of buttons to the right of that panel show expected utterances, allowing the operator to perform manual speech recognition for the robot. The pop-up map on the right allows the operator to select a location for commanding guide behaviors.

## V. Experimental Validation

Preliminary experiments presented in [8] suggested that teleoperation of multiple social robots is possible and useful, and that PTC is a fundamental technique that can support it. The video attachment shows an excerpt from that study, in which four robots are simultaneously controlled by one operator. The video includes audio feeds from all robots, to illustrate the difficulty of controlling four robots at once; however, the operator focused on and controlled only one robot at a time.

The work in [8] was only a preliminary study using internal subjects. We conducted a formal laboratory experiment with unbiased participants to evaluate the feasibility and effectiveness of our approach to teleoperation of multiple social robots, as well as the effectiveness of the Proactive Timing Control technique in particular.

### *A. Laboratory Experiment*

#### *1) Scenario*

For this experiment, we chose route guidance as a realistic example of the kind of task a robot might be assigned to perform. It is easy to imagine a business such as a shopping mall, museum, or theme park placing a robot in a high-visibility location such as a central information booth. This task also lies in an interesting middle-ground between full predictability and open-endedness, and it provides a level of interactivity not found in primarily one-way interactions such as guiding visitors in a museum.

#### *2) Experimental Design*

The experiment was designed to evaluate performance of the operator-robot team while varying two factors. The first factor, *robot-number*, was examined in three levels: *2R*, *3R*, and *4R*, representing the number of robots being simultaneously controlled by the operator. The second factor, *PTC*, was examined in two levels: *with-PTC* and *without-PTC*.

The experiment was designed to evaluate two hypotheses. Our first hypothesis was that our system would improve performance compared with a purely autonomous system, regardless of whether or not PTC was used. To validate this hypothesis, we tested the performance of our system on an absolute scale, comparing *2R*, *3R*, and *4R* trials against two baseline cases: a single-robot case where the operator was always present, referred to as the *1R* condition, and a fully-autonomous case with no operator intervention, referred to as the *A* condition. This comparison was performed separately for the *with-PTC* and *without-PTC* configurations of our system.

We predicted that performance in the *with-PTC* condition should be comparable to that in the *1R* baseline, although performance in the *without-PTC* condition might be lower, particularly for large numbers of robots (*3R*, *4R*).

Our second hypothesis was that the use of PTC in particular would improve performance of the robot team relative to the *without-PTC* conditions, and that this improvement in performance would increase for larger numbers of robots. This was evaluated by a comparison between *with-PTC* and *without-PTC* conditions for each of the *2R*, *3R*, and *4R* cases.

DOI: 10.1109/TSMCA.2011.2164243

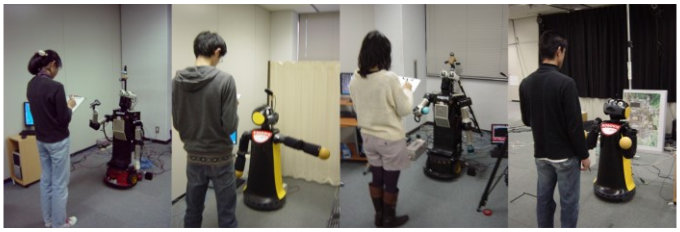

Fig. 6. Four robots operated simultaneously in our experiment.

To test these two hypotheses, our experiment included a total of 8 conditions to be evaluated: *with-PTC* and *without-PTC* variations for each of the *2R*, *3R*, and *4R* cases, and the two baseline cases, for which the use of PTC is not relevant.

*3) Setup*

The behaviors and decision logic for the route guidance scenario were adapted from a recent deployment of our robots in a shopping mall. We used the interaction flow described in Section IV-A, with the chat behaviors (Phase #3 in Table 3) adapted for use as PTC behaviors.

The PTC behaviors consisted of interruptible sequences of short behaviors with an average duration of 4.4 seconds. After each behavior, the sequence could be interrupted or continued based on the presence or absence of an operator.

An example of such a sequence is the following: "Hi, I'm Robovie. / I know many things about this shopping mall. / This week the mall is having a special anniversary celebration. / There are many discount campaigns and exciting activities planned! / There is a 10% off sale in the clothing section. / And next Sunday there will be a classical music concert!" When an operator became available, the sequence could be interrupted after any of these utterances so the robot could begin the critical section by offering to give route guidance, and the entire sequence would flow in a fairly natural way. Four of these PTC sequences were prepared for the experiment, with a maximum possible length of 12 behaviors each, and one sequence was chosen at random for each interaction.

It is important to note that these behaviors were not merely time-killing behaviors. These chat behaviors had originally been part of the natural conversation flow. When the robot spoke about these topics in the field trial, they were relevant to the customers, who enjoyed their interactions with the robot.

The experiment was conducted in our laboratory, using two Robovie-II and two Robovie-R2 robots, as shown in Fig. 6. Each robot also had an automatic speech recognition (ASR) system, which operated in parallel with the operator.

*4) Participants*

16 paid participants played the role of customers in this experiment (12 male, 4 female, average age 22.3, SD=2.5 years). All were native Japanese speakers.

One expert operator, an assistant in our laboratory, was employed to control the robots for all trials. The operator was trained in the use of the control interface and thoroughly familiar with the map of guide destinations prior to the experiment, so we assume negligible improvement in operator performance across trials.

*5) Procedure*

To provide the operator with consistent task difficulty in the different experimental conditions, each trial consisted of 24 interactions in total, *i.e.,* 6 interactions per robot in the *4R* case, 8 in the *3R* case, 12 in the *2R* case, and 24 in the *1R* case. The *A* condition was conducted with four robots but no operator.

In these trials, one "interaction" included a greeting from the robot, possible chat behaviors for PTC, a question from the customer, and a response and farewell from the robot. Eight trials were run on each day of the experiment, one for each of the conditions (*2R-with, 2R-without, 3R-with, 3R-without, 4R-with, 4R-without, 1R,* and *A*).

On the customer side, 4 participants took part in every trial, and each participant interacted with the robots a minimum of 6 times per trial. Participants were assigned evenly across the robots. To achieve even distribution in the *3R* conditions, three participants interacted 6 times each with assigned robots, while one participant moved between the robots, performing two interactions with each. In other conditions, participants did not move between robots.

This experimental procedure was repeated on four days with a different group of 4 customer participants on each day, for a total of 16 participants acting as customers. The order of the eight trials on each day was counterbalanced with respect to both the *robot number* and *PTC* factors.

For consistency in timing, interactions were robot-initiated, with the robot inserting a pause of 0-5 seconds between interactions. To provide a consistent level of workload for the operator, participants continued interacting with the robots for the entire duration of each trial, going beyond the 6 evaluated interactions if necessary.

*6) Evaluation*

There is a causal chain of effects which we expect to produce different results between the *with-PTC* and *without-PTC* conditions. First, the use of PTC should increase the number of critical sections for which the operator is present. This should consequently increase the interaction success rate, because the speech recognition system is used less often. Finally, this improved success rate combined with reduced wait time in the critical section should improve customer satisfaction.

Accordingly, to evaluate the performance of the system, we measured three variables: the rate of operator supervision in the critical section, the overall ratio of successful interactions, and customer satisfaction on a scale of 1 (unsatisfied) to 7 (satisfied). Interaction success (whether the robot had successfully answered the question) and customer satisfaction were reported by participants after each interaction.

### *B. Experimental Results*

The results of this experiment are illustrated in Fig. 7, showing operator supervision during the critical sections; Fig. 8, showing the interaction success rates; and Fig. 9, showing results from the customer satisfaction questionnaire.

DOI: 10.1109/TSMCA.2011.2164243

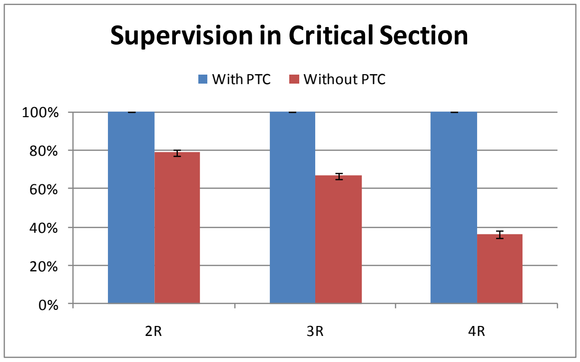

Fig. 7. Operator supervision during critical sections.

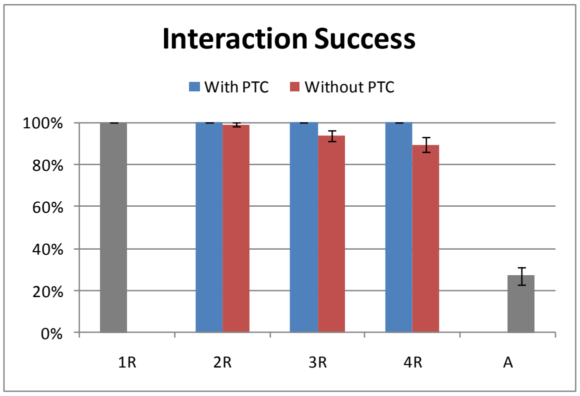

Fig. 8. Interaction success rate. Error bars show standard error.

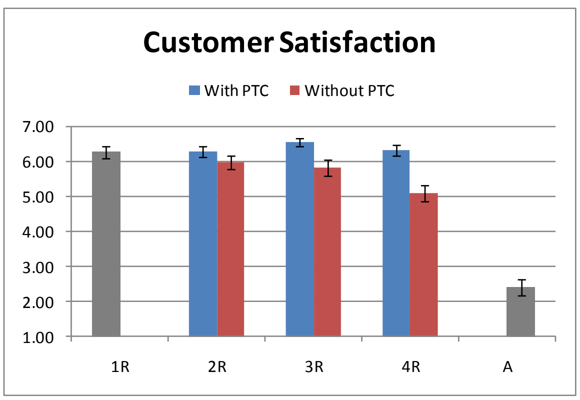

Fig. 9. Customer satisfaction. Error bars show standard error.

*1) Absolute comparison*

To evaluate the absolute performance of the system between *with-PTC* and *without-PTC*, we examined each *PTC* condition separately, comparing the *2R*, *3R*, and *4R* levels of that condition with the *1R* and A baseline cases.

*Operator supervision in critical section:* Due to the use of PTC, the operator availability during critical sections was 100% for every trial in the *with-PTC* condition (Fig. 7). In the *without-PTC* condition, operator availability decreased markedly as the number of robots increased.

*Interaction success:* For the *with-PTC* conditions, 100% of the robot's responses were correct, which is to be expected as the operator was present for all interactions. In the *without-PTC* conditions, the interaction success rate decreased as the number of robots increased, up to a 10% failure rate in the *4R* condition.

In both conditions, there was a significant difference when compared with the autonomous case, which was successful only 27% of the time (*with-PTC* condition: $\chi^2(4)= 327.805,\ p<.01$, residual analysis: *1R*, *2R*, *3R*, and *4R* to *A*: $p<.01$, *without-PTC* condition: $\chi^2(4)= 247.307,\ p<.01$, residual analysis: *1R*, *2R*, and *3R*, to *A*: $p<.01$, and *4R* to *A*: $p<.05$).

*Customer satisfaction:* For the *with-PTC* condition, customer satisfaction did not vary significantly between the *1R – 4R* conditions. A repeated-measures ANOVA revealed a significant difference in the main effect of *robot number* ($F(4,15)= 189.786,\ p<.001$). A Bonferroni test revealed *1R*, *2R*, *3R*, and *4R* to be significantly better than A ($p<.001$), but no significant difference was found among *1R*, *2R*, *3R*, and *4R*.

For the *without-PTC* condition, customer satisfaction did not vary significantly between the *1R – 3R* conditions, but decreased at *4R*. A repeated-measures ANOVA revealed a significant difference in the main effect of number of robots ($F(4,15)= 108.571,\ p<.001$). A Bonferroni test revealed that *1R*, *2R*, *3R*, and *4R* were significantly better than *A* ($p<.001$), and *1R* and *2R* were significantly better than *4R* ($p<.001$ and $p<.01$). The difference between *3R* and *4R* was approaching significance ($p=.077$). There were no significant differences among *1R*, *2R*, and *3R*.

These results confirm our hypothesis that performance in all teleoperated cases would be higher than the autonomous baseline. For the *4R* case, the significant decrease in customer satisfaction for the *without-PTC* condition also agrees with our prediction.

*2) Relative comparison*

To confirm the relative effect of PTC, we directly compared the customer satisfaction for *with-PTC* and *without-PTC* for each level of the number of robots. A paired t-test revealed significant differences for *3R* ($t$=4.442, $p<.001$), and *4R* ($t$=4.986, $p<.001$), and an almost-significant difference for *2R* ($t$=1.813, $p=.090$).

This result is consistent with our hypothesis that the use of PTC will improve performance, and that the performance improvement will be stronger for larger numbers of robots.

## C. Operator Experience

During this experiment, the operator often remarked that she felt a high level of pressure and frustration during the trials without PTC, because she was aware that many robots were entering critical sections at the same time. She said she felt relaxed, and that the interactions seemed to go smoother when PTC was used.

# VI. Simulation

Our laboratory trials provided a practical demonstration of a single operator controlling multiple robots in conversational interactions. However, due to logistical limitations such as the number of robots available, it was not possible to evaluate our system with more than four robots, or to observe the effects of varying parameters such as CTR. We created a simulation based on the interactions observed in our experiment, in order to explore the dynamics of PTC and to make projections about the performance of our system under a variety of conditions.

DOI: 10.1109/TSMCA.2011.2164243

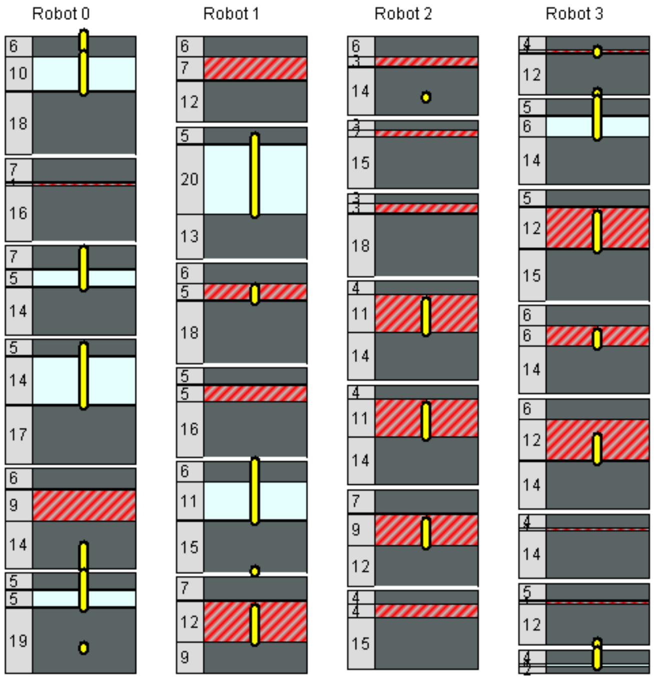


Fig. 10. Examples of simulated interactions *without* Proactive Timing Control. Dark gray boxes represent non-critical interaction phases. Light-colored boxes represent attended critical sections, and diagonally shaded red boxes represent unattended critical sections. Numbers to the left of each phase indicate its duration in seconds. Vertical bars indicate which robot the operator is attending at any given time.

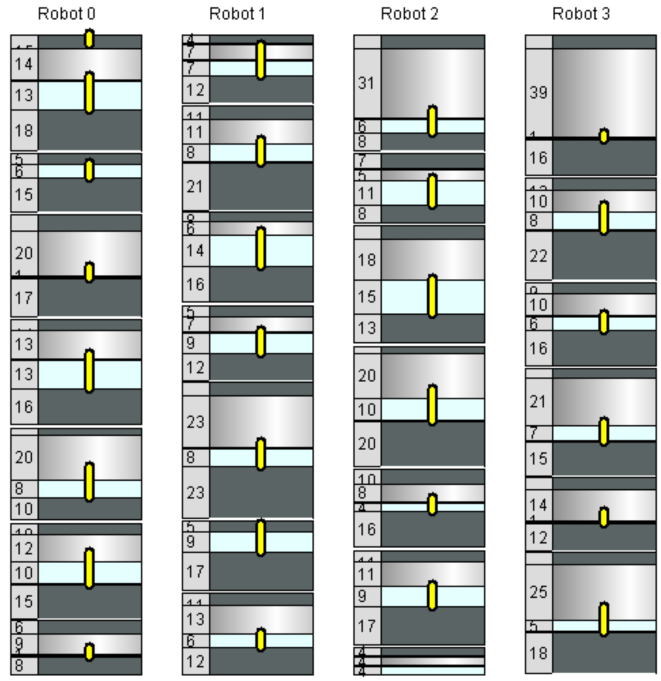


Fig. 11. Examples of simulated interactions *with* Proactive Timing Control. Dark gray boxes represent non-critical interaction phases. Light-colored boxes represent attended critical sections, and boxes with metallic shading represent PTC delay behaviors. Numbers to the left of each phase indicate its duration in seconds. Vertical bars indicate which robot the operator is attending at any given time.

### *A. Interaction Model*

The interaction model used in the simulation represents each interaction as a sequence of phases, as shown in Table 4. The length of each phase is modeled as a normal distribution with mean and standard deviation calculated from the interactions conducted in our experiment.

Interactions normally proceed in sequence through the Pre-Critical, Critical Section, Post-Critical, and Non-Interacting phases. If Proactive Timing Control is being used, then the system will transition to a PTC Behavior rather than a Critical Section if the operator is unavailable.

The simulator included an optional limit on the number of PTC behaviors, instructing the simulator to transition to the Critical Section when the operator becomes available, or after the maximum number of PTC behaviors have been executed.

### *A. Task Success*

Task success is estimated by categorizing each Critical Section as attended or unattended. For our simulation, if an operator is present for an entire Critical Section, it is considered to be attended. If the operator is absent for any fraction of the critical section, it is considered to be unattended. Note that this method of counting is used because it is important to attend a critical section from the beginning in order to guarantee that the customer's question is heard in its entirety. If the operator is late, the speech recognition system may have already provided an incorrect response, or the operator may need to repeat the question.

In our experiment, the operator's accuracy rate during attended interactions was 100%, whereas the speech recognition system's success rate in the autonomous case was 27%. Our simulation thus assumes a response accuracy of 100% for attended interactions and 27% for unattended interactions.

### *B. Operator Allocation*

The simulated operator is allocated to robots according to the following simple algorithm:

- If the operator's current robot is in a critical section, do not switch to a new robot.
- Otherwise, if any other robot is currently in a critical section, switch to the robot which has been in its critical section the longest.
- Otherwise, switch to the robot for which the anticipated critical section begins soonest.

This algorithm is not necessarily guaranteed to be optimal, but it is roughly based on the way operators were observed to operate the system during testing.

Figures 10 and 11 illustrate typical interaction flows with and without Proactive Timing Control.

TABLE 4. INTERACTION PHASES AND DURATIONS

| Interaction Phase | Mean Duration (s) | Standard Deviation (s) |
|---|---|---|
| Pre-Critical | 4.9 | 1.1 |
| PTC Behavior | 4.4 | 1.7 |
| Critical Section | 6.3 | 5.0 |
| Post-Critical | 14.8 | 2.8 |
| Non-Interacting | 0.5 | 0.0 |

DOI: 10.1109/TSMCA.2011.2164243

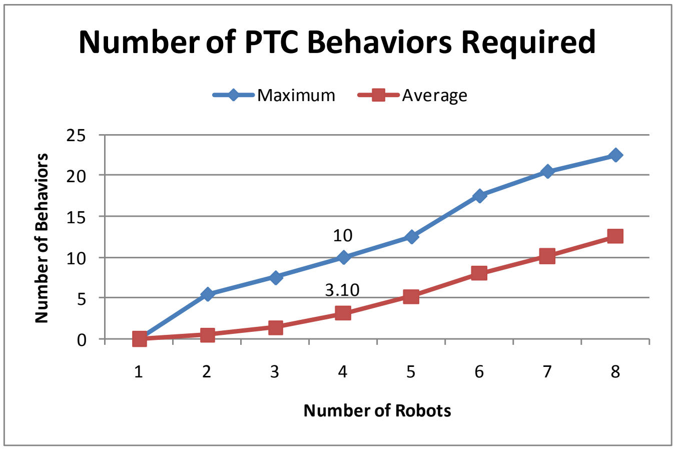


Fig. 12. As the number of robots increases, more PTC behaviors are required to guarantee that an operator can attend all Critical Sections.

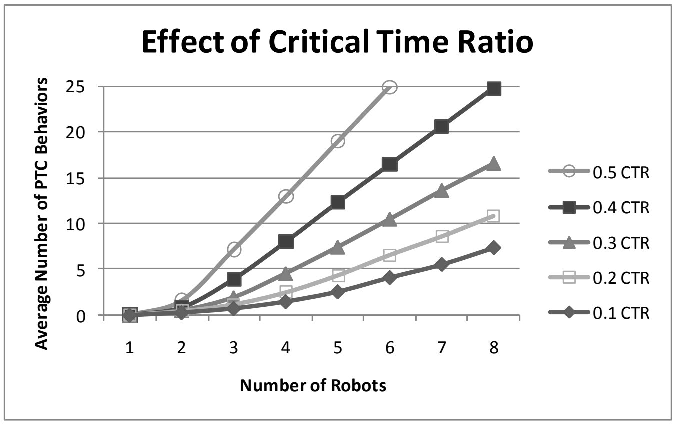


Fig. 13. The average number of PTC behaviors required for a given number of robots increases as a function of Critical Time Ratio.

*Patterns of operation*

Figures 10 and 11 show how PTC dramatically reduces the number of unattended critical sections. The operator in Fig. 10 is only present for the beginning of 31% of critical sections, whereas the operator in Fig. 11 is present for 100%. These diagrams also show the dynamics of the system – at the beginning, when customer arrivals are nearly simultaneous, the operator requires long PTC behaviors to start the interleaving of interactions, but after this point shorter PTC behaviors are sufficient to handle the random variation in interaction lengths. Such a pattern might be observed in a busy case where customers were waiting their turn to talk to the robots.

### C. *Number of PTC Behaviors*

As the number of robots increases, more PTC behaviors will be required, and the average length of interactions will increase. We examined this trend using our simulation.

Figure 12 shows the maximum and average number of PTC behaviors used by our simulated system in runs of 1000 interactions using 1-8 robots. Here, one PTC behavior consists of a short utterance of around 4.4 seconds in length.

The results from this simulation agreed closely with our experimental results, as our operator used a maximum of 10 and an average of 3.9 PTC behaviors for the 4-robot case, compared with a maximum of 10 and average of 3.1 in the simulation.

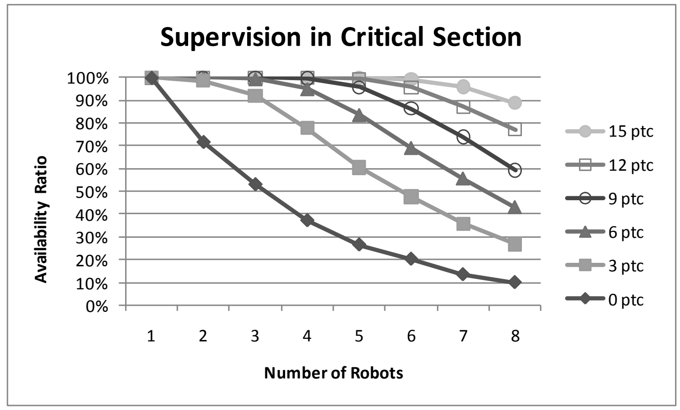


Fig. 14. Variation in operator supervision during critical sections as maximum number of PTC behaviors varies.

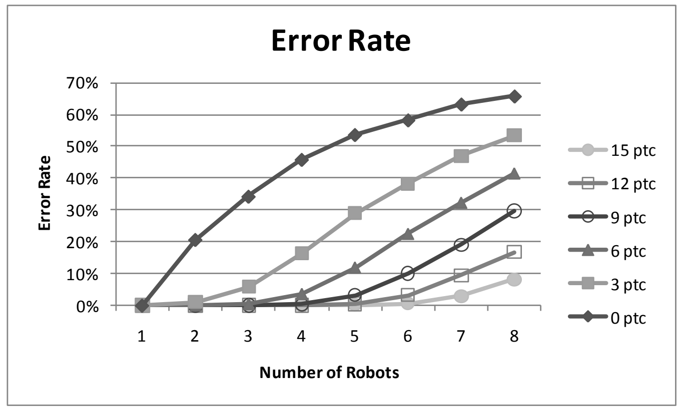


Fig. 15. Change in error rate as maximum number of PTC behaviors varies.

As discussed in Section III, Critical Time Ratio (CTR) is determined by the design of an interaction. A highly interactive robot application would have long critical sections, and thus a high CTR, whereas a robot mostly performing fixed behaviors with less responsiveness to a customer would have a low CTR. Figure 13 shows the average number of PTC behaviors used in our simulations for interactions using a base CTR (not including PTC behaviors) ranging from 0.1 to 0.5. The figure illustrates how an interaction designer can balance the CTR of an interaction with the desired average PTC duration to target a given number of robots.

### D. *Relying on Autonomy*

The results so far assume an unlimited number of PTC behaviors and a target of perfect operator attendance during critical sections. However, the choice of how many PTC behaviors to use can be seen as a tradeoff between the desired level of response accuracy and its cost in terms of design difficulty and extended interaction time. Limiting the number of PTC behaviors causes the system to rely more on autonomy. Figures 14 and 15 show how system performance degrades when the number of PTC utterances is limited.

In these interactions, the limited number of PTC behaviors increases the number of unattended critical sections, and consequently increases the error rate due to failures caused by the autonomous system. For a route guidance application, errors are not acceptable, so the maximum number of PTC behaviors

DOI: 10.1109/TSMCA.2011.2164243

shown in Fig. 12 should be prepared. However, it is conceivable that some conversational robot applications might permit a small number of errors, and so the designer can make the trade-off between PTC duration and target error rate.

As the capabilities of recognition systems improve over time, it may be possible to rely more heavily on autonomy and thus achieve very high performance with minimal use of PTC.

## VII. Discussion

We were actually quite surprised by the positive results of the laboratory experiment and the operator's success in controlling four robots. Theoretical predictions notwithstanding, we had initially expected three robots in a real-world situation to be a challenge and four to be nearly impossible. However, the results from our experiment showed our approach to multi-robot control for conversational interactions to be much more effective than we had anticipated.

Here we will discuss several results from our experiment and how the principles can be generalized to other systems.

### *A. Maximum fan-out*

As the simulation results illustrate, the maximum number of robots an operator can control depends on a variety of factors, including sensor reliability, critical time ratio, maximum number of PTC behaviors, and acceptable error rate. For the most difficult interaction settings in our experiment, the operator was successfully able to control four robots with 90% task success, and for the trials using PTC the operator was 100% successful in conducting all 288 interactions with no errors. Both of these results are dramatically superior to the low 27% success rate of the robots operating autonomously.

### *B. Defining Criticality*

One conceptual model contributing to the success of our system was the division of interactions into critical and non-critical sections. It is fairly straightforward to apply this model to transactional interactions such as giving directions, particularly when a question is followed by a long explanation.

This model can be applied to many kinds of interactions, such as providing information, giving directions, and providing services requested by a customer. It can also be adapted for more complex interactions. For example, if a robot needs to ask a series of several questions, it may make sense to extend the critical section to encompass all of them in a single block. This may result in a small amount of wasted time for the operator while the robot is giving explanations or asking questions, but the operator is also guaranteed to be present for each of the follow-up questions, at a time where it may be awkward to insert delay behaviors.

In the general case, it will be important to consider both the risk of error and the cost of that error, both of which can be continuous variables. These subtleties may become more important in complex or long-term interactions; however, for the simple interactions in this study we will consider only two levels of criticality and model all failures as having equal cost.

### *C. Proactive Timing Control*

From a system-level perspective, the Proactive Timing Control technique improves the operator's span-of-control in two distinct ways. To illustrate this, consider the critical sections of a robot's interactions to be like teeth in a gear, with noncritical sections represented by the gaps between the teeth. For an operator to control two robots, two gears must mesh, that is, the critical sections cannot overlap. The first way PTC achieves this is by synchronizing the gears – that is, holding one gear in place briefly while the other turns, until the critical section of one falls into the gap of the other. Adjustments like this are occasional and probably small. For example, with a hypothetical set of gears with perfectly regular spacing (*i.e.* when the lengths of the conversation phases are fixed) this adjustment would only be made once.

The second way PTC improves interleaving of tasks is by reducing the Critical Time Ratio, that is, by widening the gaps between the gear teeth overall. This is necessary when the time between critical sections is not sufficient to allow the gears to mesh during normal operation. This is a less desirable use of PTC, as delay behaviors must be executed for nearly every interaction. For behaviors such as those in our implementation, the content of the delay behaviors is generally not related to the context of the interaction. Thus, to create more natural interactions, it would be better to reduce the CTR at design time by extending or inserting behaviors relevant to the current interaction, rather than rely on PTC to make up for an insufficient gap between critical sections.

### *D. Limitations*

The user study presented in this work was conducted in a laboratory environment with a pool of 16 customers performing repeated interactions with a robot. The results demonstrate that the proposed technique significantly improves performance, however, the use of PTC in real-world deployments of robots may have a stronger or weaker effect on customer satisfaction due to factors such as the novelty effect of the robots, customers' lack of familiarity with the robot's conversation style due to non-repeated interactions, quality and appropriateness of the robot's utterances, and variation based on the deployment context, *e.g.* whether people in that environment are in a relaxed or rushed mood.

Likewise, the simulation results are based on the user studies, so the results should not be necessarily seen as numerical predictors of customer satisfaction in the field. However, these results do serve to illustrate the dynamics of the system and the effects of varying different parameters, results that will be useful in designing and tuning systems for the real-world deployment.

## VIII. Conclusions

In this study, we have presented a general framework for enabling the simultaneous teleoperation of multiple social robots, focusing on four key design areas: human-robot interaction design, autonomy design, multiple-robot

DOI: 10.1109/TSMCA.2011.2164243

coordination, and teleoperation interface design. While many key aspects of autonomy design and teleoperation interface design are similar to issues faced in other fields of robotics, the areas of human-robot interaction design and multi-robot coordination present many new issues which are unique to social robots.

Based on this conceptual framework, we implemented a robot system to demonstrate the new concept of a single operator controlling multiple robots in simultaneous social interactions. Our laboratory evaluations showed our system to be quite successful, with an operator achieving over 95% task success while controlling up to four robots in one experiment. These results demonstrate the value of our conceptual framework as well as the effectiveness of our specific solutions, such as Proactive Timing Control.

In our experiment, task success and customer satisfaction in every condition were far superior to those attainable by the same system operating in a fully-autonomous mode. Furthermore, our simulation results show that PTC reduces or eliminates conflicts between robots for an operator's attention. Even when PTC behaviors are limited, and the operator is forced to rely on automatic speech recognition some of the time, our simulation results indicate that PTC will provide a substantial increase in task success over a system with no timing control.

Most importantly, we have tested this system using an actual task often performed by our robots in the field, suggesting that this technology can be immediately put to use in real-world field trials. This study introduces the new field of teleoperation for multiple social robots, and several of the topics addressed in this paper are promising areas for further in-depth research.

## References


[1] W. Burgard, A.B., Cremers, D. Fox, D. Hähnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thrun, "The interactive museum tour-guide robot," in *Proc. of the AAAI Fifteenth NationalConference on Artificial Intelligence,*1998. pp. 11-18.

[2] M. Shiomi, T. Kanda, H. Ishiguro, and N. Hagita, "Interactive humanoid robots for a science museum," *IEEE Intelligent Systems*, vol. 22, no. 2, pp. 25-32, Mar/Apr, 2007.

[3] R. Siegwart *et al*., "Robox at Expo.02: A large scale installation of personal robots," *Robotics and Autonomous Systems*, 42, 203-222, 2003.

[4] R. Gockley, J. Forlizzi, and R. Simmons, "Interactions with a moody robot," *ACM/IEEE 1st Annual Conference on Human-Robot Interaction (HRI2006)*, pp. 186-193, 2006.

[5] T. Kanda, T. Hirano, D. Eaton, H. Ishiguro, "Interactive robots as social partners and peer tutors for children: a field trial," *Human Computer Interaction*, Vol. 19, No. 1-2, pp. 61-84, 2004.

[6] J. R. Movellan, F. Tanaka, I. R. Fasel, C. Taylor, P. Ruvolo, and M. Eckhardt,, "The RUBI Project: a progress report," *ACM/IEEE 2nd Annual Conference on Human-Robot Interaction (HRI2007)*, pp. 333-339, 2007.

[7] K. Hayashi et al., "Humanoid robots as a passive-social medium - a field experiment at a train station," *ACM/IEEE 2nd Annual Conference on Human-Robot Interaction (HRI2007)*, pp. 137-144, 2007.

[8] D. F. Glas, T. Kanda, H. Ishiguro and N. Hagita, "Simultaneous teleoperation of multiple social robots," *ACM/IEEE 3rd Annual Conference on Human-Robot Interaction (HRI 2008)*, 2008, pp. 311-318.

[9] T. Kanda, M. Shiomi, Z. Miyashita, H. Ishiguro, and N. Hagita, "An affective guide robot in a shopping mall," *4th ACM/IEEE International Conference on Human-Robot Interaction(HRI2009)*, pp.173-180, 2009.

[10] A. Green et al., "Applying the Wizard-of-Oz framework to cooperative service discovery and configuration," *Proc. IEEE Int. Workshop on Robot and Human Interactive Communication*, 2004.

[11] S. N. Woods, M. L. Walters, K. L. Koay, and K. Dautenhahn, "Comparing human robot interaction scenarios using live and video based methods: towards a novel methodological approach," *Int. Workshop on Advanced Motion Control*, 2006, pp. 750–755.

[12] B. Brunner, G. Hirzinger, K. Landzettel, and J. Heindl, "Multisensory shared autonomy and tele-sensor-programming - key issues in the space robot technology experiment ROTEX," *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 1993, pp. 2123 – 2139.

[13] D. Vanhooydonck, E. Demeester, M. Nuttin, and H. Van Brussel, "Shared control for intelligent wheelchairs: an implicit estimation of the user intention," *Proc. of the 1st International Workshop on Advances in Service Robotics (ASER)*, Bardolino, Italy, pp. 176-182, March 2003.

[14] D. B. Kaber and M. R. Endsley, "The effects of level of automation and adaptive automation on human performance, situation awareness and workload in a dynamic control task," Theoretical Issues in Ergonomics Science, 5(2):113–153, March-April 2004.

[15] M. A. Goodrich, T. W. McLain, J. D. Anderson, J. Sun, and J. W. Crandall, "Managing autonomy in robot teams: observations from four experiments." *ACM/IEEE 2nd Annual Conference on Human-Robot Interaction (HRI2007)*, 2007, pp. 25-32.

[16] P. Scerri, D. Pynadath, and M. Tambe, "Towards adjustable autonomy for the real-world," *Journal of AI Research (JAIR)*, 2002, Volume 17, pp. 171-228.

[17] B. P. Sellner, F. Heger, L. Hiatt, R. Simmons, and S. Singh, "Coordinated multi-agent teams and sliding autonomy for large-scale assembly," *Proc. of the IEEE - Special Issue on Multi-Robot Systems*, Vol. 94, No. 7, July, 2006, pp. 1425 - 1444.

[18] T. Fong and C. Thorpe, "Vehicle Teleoperation Interfaces," *Autonomous Robots*, Vol. 11, No. 1, 2001, pp. 9-18.

[19] H.H. Chiang, S.J. Wu, J.W. Perng, B.F. Fu, and T.T. Lee, "The Human-in-the-Loop Design Approach to the Longitudinal Automation System for an Intelligent Vehicle", *IEEE Transactions on Systems, Man, And Cybernetics Part A: Systems and Humans*, Vol. 40, No. 4, pp. 708-720, July 2010.

[20] N. E. Sian, K. Yokoi, S. Kajita, and K. Tanie, "Whole body teleoperation of a humanoid robot integrating operator's intention and robot's autonomy," *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS2003)*, 2003, pp. 1651-1656.

[21] S. G. Hill and B. Bodt, "A Field Experiment of Autonomous Mobility: Operator Workload for One and Two Robots," *ACM/IEEE 2nd Annual Conference on Human-Robot Interaction (HRI2007)*, 2007, pp. 169-176.

[22] B. P. Sellner, L. M. Hiatt, R. Simmons, and S. Singh, "Attaining Situational Awareness for Sliding Autonomy," *ACM/IEEE 1st Annual Conference on Human-Robot Interaction (HRI2006)*, 2006, pp. 80-87.

[23] R. M. Ratwani, J. M. McCurry, and J. G. Trafton, "Single operator, multiple robots: an eye movement based theoretic model of operator situation awareness," *Proc. 5th ACM/IEEE International Conference on Human-Robot Interaction (HRI2010)*, 2010, pp. 235-242.

[24] D. R. Olsen, Jr. and S. B. Wood, "Fan-out: measuring human control of multiple robots," *Proc.of the SIGCHI conference on Human factors in computing systems*, p.231-238, April 24-29, 2004, Vienna, Austria.

[25] J. W. Crandall and M. A. Goodrich, "Characterizing efficiency of human robot interaction: a case study of shared-control teleoperation," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS2002)*, 2002, pp. 1290- 1295.

[26] J. W. Crandall, M. L. Cummings, "Developing performance metrics for the supervisory control of multiple robots," *ACM/IEEE 2nd Annual Conference on Human-Robot Interaction (HRI2007)*, 2007, pp. 33-40.

[27] M.L. Cummings and P.J. Mitchell, "Predicting controller capacity in remote supervision of multiple unmanned vehicles," *IEEE Systems, Man, and Cybernetics, Part A Systems and Humans*, (2008) 38(2), p. 451-460.

[28] A. Steinfeld, T. Fong, D. Kaber, M. Lewis, J. Scholtz, A. Schultz, M. Goodrich, "Common metrics for human-robot interaction", *ACM/IEEE 1st Annual Conference on Human-Robot Interaction (HRI2006)*, pp. 33-40, 2006.

[29] M. Álvarez, R. Galán, F. Matía, D. Rodríguez-Losada, and A. Jiménez, "An Emotional Model for a Guide Robot," *IEEE Transactions on Systems, Man, And Cybernetics Part A: Systems and Humans*, Vol. 40, No. 5, pp. 982-992, September 2010.

DOI: 10.1109/TSMCA.2011.2164243

[30] N. Dahlbäck, A. Jönsson and L. Ahrenberg, "Wizard of Oz studies: why and how," *International Conference on Intelligent User Interfaces*, 1993, pp. 193-200.

[31] A. Steinfeld, O. C. Jenkins, B. Scassellati, "The Oz of Wizard: simulating the human for interaction research," *4th ACM/IEEE International Conference on Human-Robot Interaction (HRI2009)* pp. 101-107, 2009.

[32] M. Shiomi, D. Sakamoto, T. Kanda, C. T. Ishi, H. Ishiguro, and N. Hagita, "A semi-autonomous communication robot - a field trial at a train station," *ACM/IEEE International Conference on Human-Robot Interaction (HRI2008)*, pp. 303-310, 2008.

[33] N. Sian, T. Sakaguchi, K. Yokoi, Y. Kawai, and K. Maruyama, "Operating humanoid robots in human environments," *Proc. RSS Workshop: Manipulation for Human Environments,* Philadelphia, PA, Aug. 2006.

[34] D. R. Olsen and M. A. Goodrich. "Metrics for evaluating human-robot interaction." *Performance Metrics for Intelligent Systems (PerMIS '03),* 2003.

[35] T. Kanda, H. Ishiguro, M. Imai, and T. Ono, "Development and evaluation of interactive humanoid robots," *Proceedings of the IEEE*, Vol.92, No.11, pp. 1839-1850, 2004.

[36] T. Shiwa, T. Kanda, M. Imai, H. Ishiguro and N. Hagita, "How quickly should communication robots respond?", *ACM/IEEE 3rd Annual Conference on Human-Robot Interaction (HRI2008)*, pp.153-160, Amsterdam, the Netherlands, 2008.

[37] K. Zheng, D. F. Glas, T. Kanda, H. Ishiguro and N. Hagita, "How Many Social Robots Can One Operator Control?", *ACM/IEEE 6th Annual Conference on Human-Robot Interaction (HRI 2011)*, pp. 379-386 Lausanne, Switzerland, 2011.

**Dylan F. Glas** received S.B. degrees in aerospace engineering and in earth, atmospheric, and planetary science from MIT in 1997, and he received his M.Eng. in aerospace engineering in 2000, also from MIT.

He has been a Researcher at the Intelligent Robotics and Communication Laboratories (IRC) at the Advanced Telecommunications Research Institute International (ATR) in Kyoto, Japan since 2005. From 1998-2000 he worked in the Tangible Media Group at the MIT Media Lab. His research interests include networked robot systems, teleoperation for social robots, human-machine interaction, ubiquitous sensing, and artificial intelligence.

**Takayuki Kanda** (M'04) received his B. Eng, M. Eng, and Ph. D. degrees in computer science from Kyoto University, Kyoto, Japan, in 1998, 2000, and 2003, respectively. This author became a Member (M) of IEEE in 2004. From 2000 to 2003, he was an Intern Researcher at ATR Media Information Science Laboratories, and he is currently a Senior Researcher at ATR Intelligent Robotics and Communication Laboratories, Kyoto, Japan. His current research interests include intelligent robotics, human-robot interaction, and vision-based mobile robots.

**Hiroshi Ishiguro** (M') received a B.Eng. and M.Eng. in computer science from Yamanashi University, Japan in 1986 and 1988, respectively. He received a D.Eng. in systems engineering from the Osaka University, Japan in 1991.

He is currently Professor in the Graduate School of Engineering at Osaka University (2002-). He is also Visiting Group Leader (2002-) of the Intelligent Robotics and Communication Laboratories at the Advanced Telecommunications Research Institute, where he previously worked as Visiting Researcher (1999-2002). He was previously Research Associate (1992-1994) in the Graduate School of Engineering Science at Osaka University and Associate Professor (1998-2000) in the Department of Social Informatics at Kyoto University. He was also Visiting Scholar (1998-1999) at the University of California, San Diego and Researcher at PREST of the Japan Science and Technology Corporation. In 2000 he founded Vstone Co. Ltd. He then became Associate Professor (2000-2001) and Professor (2001-2002) in the Department of Computer and Communication Sciences at Wakayama University. His research interests include distributed sensor systems, interactive robotics, and android science.

**Norihiro Hagita** (M'85 – SM'99) Norihiro Hagita received his B.E., M.E., and Ph.D. degrees in electrical engineering from Keio University in 1976, 1978, and 1986.

In 1978, he joined Nippon Telegraph and Telephone Public Corporation (Now NTT). He moved to the ATR (Advanced Telecommunication Research Institute International) and launched the ATR Media Information Science Laboratories in 2001 and the ATR Intelligent Robotics and Communications Laboratories in 2002. He is currently Board Director of ATR and ATR Fellow, director of Social Media Research Laboratory Group and Intelligent Robotics and Communication Laboratories. He is a chair of Technical Committee in Network Robot Forum in Japan and a co-chair of Technical Committee on Networked Robots in IEEE RAS society. He also stayed as a visiting researcher at Prof. Stephen Palmer's lab in University of California, Berkeley (Dep. of Psychology) during 1989-1990. He is currently the director of ATR Intelligent Robotics and Communication Laboratories (IRC) at the Advanced Telecommunications Research Institute International (ATR) in Kyoto, Japan.