An Autonomous Assistive Robot for Planning, Scheduling and Facilitating Multi-User Activities

Wing-Yue Geoffrey Louie, *IEEE Student Member*, Tiago Vaquero, Goldie Nejat, *IEEE Member*, J. Christopher Beck

***Abstract*— In this paper we present the development of a novel multi-user human-robot interaction (HRI) system architecture to allow the social robot Tangy to autonomously plan, schedule and facilitate multi-user activities while considering the users' schedules. During scheduled activities, the robot is able to interact with a group of users by providing both group-based and individualized assistance based on the current state of the activity and the needs of the individual users engaged in the social interactions. Such planning and scheduling of daily activities of a social robot while reasoning about multiple user schedules has not yet been addressed in the literature. Herein, the HRI multi-user activities we consider are a series of Bingo games. System performance experiments presented in the paper validate the use of the proposed multi-user system architecture in: 1) planning and scheduling daily Bingo games for Tangy to facilitate while considering the individual schedules of the users, and 2) determining the appropriate behaviors of the robot with respect to individuals and groups of people while providing game reminders prior to a Bingo game starting and also while facilitating the game itself.**

I. INTRODUCTION

A robot is defined to be social if it interacts with humans following behavioral norms expected by people [1]. From a human-robot interaction (HRI) perspective, one of the existing open grand challenges is to empower a personal robot with the social functionality to engage autonomously in activities with more than one user. These interactions are particularly challenging, as a robot must be able to identify and manage the collective needs of the individuals within a group during such activities [2]. Furthermore, in order for a robot to effectively implement such group activities, it must also be able to autonomously plan and schedule the activities around the daily schedules of these multiple individuals.

Recently, a handful of social robots have been designed to interact with multiple users during different types of activities including as museum tour guide [3],[4] and bartender. For example, in [3], the humanoid robot Alpha was designed to engage groups of people during museum tours by using visual perception, sound source localization and speech recognition to shift its gaze direction between different people. The shifting of the robot's gaze direction from one person to a second person was verified in laboratory experiments. Questionnaire results of people interacting with Alpha at a university science fair showed that most people found the robot to have human-like eye gaze, gestures and facial expressions, as well they felt that the robot was aware of them. In [4], several robots provided guided tours through a museum to groups of people. The robots used a tour planning algorithm to determine which topics to discuss based on interests of the groups, the amount of content known for each topic and shared topics with the other robots. If necessary, the robots would also swap people among groups if these people could benefit from more interesting topics presented by the other robots. Simulations conducted with two virtual robots and two groups of virtual people showed that this approach was able to optimize content coverage of the topics of interests for the people in the groups. In [5], a bartender robot was developed which used planning to interact with two customers during a drink ordering scenario. Namely, customer states were determined from speech and visual inputs which were then used by a high-level planner to determine the robot's behaviors and the order in which to serve the customers. Subject testing showed that most of the customers were able to successfully order a drink from the bartender robot.

In our work, we consider the scenario where social robots must autonomously organize, facilitate and monitor multi-person activities throughout the duration of the day. This problem requires not only reasoning about which activities the robot should implement (i.e., planning) but also reasoning about when these activities should occur (i.e., scheduling), where the latter takes into account the daily schedules of the individuals (as well as the robot) to facilitate multiple group-based activities. Therefore, considering the daily schedules of the people of interest is important for a robot to effectively plan, schedule and facilitate group activities throughout the day. Such integration of planning and scheduling for autonomous robots engaged in multiple social HRI activities has not yet been addressed in the literature.

In addition to scheduling activities, a social robot is also required to provide reminders to the individuals participating in a group activity prior to the start of the activity. Furthermore, important to group-based activities, the robot must be able to not only engage the group as a whole in the activity, but also be able to provide individualized assistance when needed. In this paper, we present the design of a novel multi-user HRI system architecture for the mobile social robot, Tangy, in order for the robot to be able to plan, schedule and facilitate group-based recreational activities. Namely, the objective of this work is to design Tangy to be capable of: 1) both planning and scheduling a set of

*This research has been funded in part by the Natural Sciences and Engineering Council of Canada (NSERC).

W.G. Louie, T. Vaquero and G. Nejat are with the Autonomous Systems and Biomechatronics Laboratory, Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, ON, Canada. G. Nejat is also affiliated with the Toronto Rehabilitation Institute, Toronto, ON, Canada (e-mail: geoffrey.louie@mail.utoronto.ca, tvaquero@mie.utoronto.ca, nejat@mie.utoronto.ca).

J.C. Beck is with the Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, ON, Canada (email: jcb@mie.utoronto.ca).

recreational group activities throughout the course of a day while considering the schedules of multiple users; and 2) interacting with a group of users during a scheduled recreational activity by providing both group-based and individualized assistance based on the current activity state and the needs of the individual users engaged in the interaction.

II. Automated Planning & Scheduling of Robot Activities

Research on *Automated Planning and Scheduling* (P&S) investigates computational techniques and systems capable of autonomously reasoning about what actions to perform (and in which order), when to perform them, and with what resources to achieve a set of goals. Such automated reasoning is considered an essential component of intelligent behavior and, consequently, an essential part of autonomous machines such as robots [6]. The *integration* of P&S has been investigated over the past several years [7] in such robotic applications as office assistant robots [8] and elder care robots [9],[10]. With respect to HRI applications, the existing work has mainly focused on automated reasoning about the schedule of a single user (e.g., [10]) or about task requests from multiple users without considering their specific individual schedules (e.g., [11]). For example, the Pearl robot [10] uses the Autominder system [12] to incorporate an elderly person's plan of daily activities and what he/she is currently doing to determine which reminders should be provided and when. However, the Autominder system does not consider multiple users. In this work we address the integration of planning and scheduling of the activities of a social robot while reasoning about multiple user schedules.

III. The Tangy Robot's System Architecture

In our work, we address two challenges with respect to designing a socially assistive robot for group-based activities: 1) integrating planning and scheduling of the activities while considering the individual schedules of the users, and 2) HRI with a group of people while providing both group and individual assistance during the course of an activity. In this section we present our unique system architecture for the robot Tangy in order to allow the robot to autonomously plan, schedule and execute multi-person activities taking into account the schedules of the individuals of interest. In what follows, we present the development of the multi-user HRI system architecture for the robot for an envisioned Bingo game HRI scenario. We have chosen Bingo as our group activity as this game provides a social setting to promote interactions with the robot and other users, as well as Bingo has been shown to improve cognitive abilities (e.g., memory, recognition, and visual search) for players of all ages [13].

A. HRI Scenarios

The proposed HRI scenario consists of the Tangy robot organizing and implementing a series of Bingo games. The successful implementation of these games requires Tangy to schedule groups of individuals for different Bingo sessions, and then, prior to the start of each game, provide reminders to these individuals based on their available schedules.

In order to provide game reminders, Tangy needs to decide when the user is available for a reminder and where the user will be at that time. The robot will navigate to a user's location based on his/her schedule and then prompt the user to attend an upcoming Bingo game by telling him/her its time and location. When a Bingo game is ready to commence, the robot will arrive at the appropriate location and act as the game facilitator by calling out Bingo numbers and providing assistance to players. Assistance is provided in the form of: 1) repeating to users Bingo numbers that have already been called out, and 2) reviewing players' cards in order to prompt them to mark missed called numbers as well as to verify winning Bingo cards and celebrate with winners. Tangy will need to manage multiple users during a game and distinguish the players in order to provide individualized assistance as needed.

B. The Social Robot Tangy

The social robot Tangy has a human-like upper torso which is mounted on top of a differential drive mobile platform, Fig. 1. Tangy uses a synthesized voice, body language and gestures, as well as a 12" touch screen tablet to interact with people. The robot has 2 degrees of freedom (DOF) for the neck to allow for shaking and nodding gestures, 3 DOF for the eyes to allow each eye to independently move left and right as well as move up and down together, and 1 DOF to open and close the mouth which is synchronized to speech output. Each arm has 2 DOF in the shoulder, elbow, wrist, and grippers to allow the robot to display both body language and pointing gestures during interactions. The tablet on Tangy's chest is used to display such activity information to players as the called out Bingo numbers. Tangy has an ASUS Xtion PRO LIVE Sensor mounted on its chest, and two 2D Axis M1031-W cameras in its eyes and one 2D Logitech Pro C920 camera located on the top of its head to monitor the Bingo game and identify users. The mobile base also consists of a Hokuyo URG-04LX-UG01 laser range finder mounted on a tilting platform used for navigation.

C. Multi-User System Architecture

The proposed modular multi-user system architecture is presented in Fig. 1. The architecture is used to schedule a set of requested Bingo activities for a day and then provides Tangy with the plans needed to: 1) remind multiple users about the scheduled games prior to game commencement, and then, 2) engage multiple groups of users in the Bingo activity itself. The system architecture described in this work was designed and implemented using the Robot Operating System (ROS) framework [14].

The system architecture is composed of: 1) a *centralized server* responsible for high level planning and scheduling of the robot activities for the Bingo games, and 2) the *robot Tangy controller* which is responsible for autonomously executing the activities requested by the server. Their modules are described below.

1) Activity Request Manager

The *Activity Request Manager* manages and monitors Bingo activity requests sent by the system administrators. Given a set of new Bingo game requests, the manager sends them as a goal to the *Automated Planning and Scheduling (P&S) system* in order to check whether the requests can be fulfilled by Tangy, i.e., if a feasible plan exists. The goal

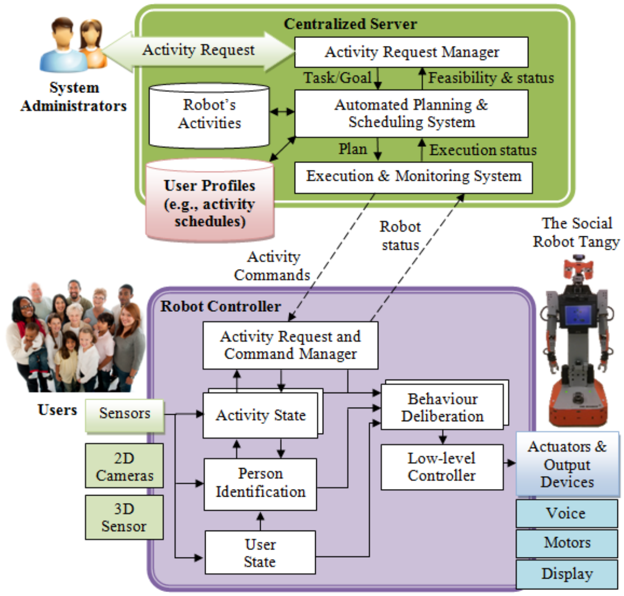


Figure 1. The Proposed Multi-User System Architecture for Tangy

sent by the manager refers to the completion of the Bingo activities by the end of the day. If a valid plan is determined, the *Activity Request Manager* monitors the status of each request until its completion.

2) *Automated Planning & Scheduling System*

The *Automated Planning and Scheduling system* is the main intelligent component of the centralized server. It performs both planning and scheduling of Tangy's activities to fulfill the given goal from the activity request manager.
In this work we have incorporated the domain-independent OPTIC temporal planner [15] to reason about the activities of the users and Tangy, the durations of these activities, the available resources to Tangy (e.g., battery level) and activity time constraints. OPTIC is a forward-chaining partial-order temporal planner that uses Simple Task Network (STN) and mixed integer program (MIP) to reason about time constraints and preferences. This general purpose planner reads as input a P&S problem described in Planning Domain Definition Language (PDDL) [16]. The *P&S system* is not dependent on any one particular type of temporal planner and can incorporate any planner that utilizes PDDL.

The *P&S system* uses the following input information to plan and schedule the robot activities as illustrated in Fig. 1:
1) The Bingo activity requests are first provided by the *Activity Request Manager* via the system administrators. Each Bingo activity is specified by its location and duration as well as the intended users. Scheduling this activity is the goal of the planning and scheduling process;
2) The specifications (conditions and duration) of the activities are provided by the *Robot's Activity database*. This information is represented as PDDL specifications of actions which include preconditions (e.g., a robot can only interact with a user if he/she is available), effects (e.g., battery consumption of the robot during operation) and durations (e.g., a bingo game is 60 minutes long). In this work the robot is able to perform three main activities: *remind* a user about a game that will take place at a future time and location; *play Bingo* at a certain location with a set of users; or *recharge* the robot's battery at a location that has a charging station. For all these activities, the robot has to *move* to a target location to execute the activity;
3) The user schedules are provided by the *User Profile database*. They represent the availability of users and their locations;
4) The robot status is provided to the *Execution and Monitoring system* by the robot's *Activity Request and Command Manager*. The robot status includes its location, battery level, and the activity progress;
5) The topology of the environment is provided to the *P&S system* in an embedded xml file. The topology includes rooms, corridors, and the distances between the locations.

The aforementioned input information is translated to a PDDL problem specification. In particular, user schedules are represented as timed initial literals (TILs) [16] in the problem instance. These TILs describe time frames in which each user is either available or unavailable for HRI (in this case, game reminders or actually playing Bingo). In fact, an HRI with Tangy can only occur when users are available right before the interaction starts and during the interaction. In addition to this constraint, the Bingo activity can only occur if all the participating users have been reminded by the robot prior to the game. Moreover, a battery level constraint is applied to the robot activities: the robot can only execute an activity if it has enough remaining battery life to do so and be able to reach its charging station.

The resulting PDDL representation generated and translated by the *P&S system* is then sent to the OPTIC planner. The resulting *plan* (i.e., an ordered list of time-stamped activities) from the temporal planner is processed by the *P&S system* and sent for execution by the robot. The status of the execution of an activity is sent from the *Execution and Monitoring system* to the *P&S system* for evaluating whether re-planning is required.

3) *Execution & Monitoring System*

The execution of the plan generated by the *P&S system* is managed and monitored by the *Execution and Monitoring system*. Each time-stamped activity of the plan is sent as an activity command to the robot at the specified time. A server clock is used to check which activity should be sent to the robot. In each cycle of the server clock, the *Execution and Monitoring system* tracks activity progress through robot feedback. If a robot activity fails (e.g., the robot cannot find a person to deliver the reminder, it cannot reach a target location, or the game has been interrupted) or the battery level of the robot is too low, the *Execution and Monitoring system* informs the *P&S system* that re-planning and re-scheduling is needed.

4) *Activity Request and Command Manager*

The *Activity Request and Command Manager* onboard the robot receives the activity command sent by the centralized server and provides the robot status (including the activity status) back to the server. Each activity command is used to identify the type of activity the robot should implement (remind, play Bingo or recharge), the location and duration of the activity, and the target users. This information is sent to the *Activity State* and *Behavior Deliberation* modules which behaviors. During the execution of a command activity, the *Activity Request and Command Manager* receives activity state progress from the *Activity State* module and reports this information back to the server.

5) Activity State

The *Activity State* module monitors the state of either the reminder activity or the Bingo activity depending on which activity has been requested by the server. This module is initialized when the *Activity Request and Command Manager* has sent a request to begin one of these activities.

Given a request to remind a user at a certain location about an upcoming Bingo game, the reminder activity has two states: 1) the user has been notified of the scheduled game, and 2) the user still needs to be notified. If a user has been identified at the specified location and a reminder has been delivered the activity state will be "user notified"; or it will be "user still needs to be notified" otherwise.

The Bingo activity states that are identified within this module include: 1) a winning Bingo card, 2) an incorrectly marked Bingo card, 3) a Bingo card missing marked numbers, 4) a correctly marked Bingo card (that is not a winning card), and 5) an occluded Bingo card. The Bingo cards each have a unique symbol located on their top right corner in order to identify which card each player has. In addition, each unique Bingo card has a 5x5 grid of random numbers ranging from 1 to 75 with one "Free Space" in the center of the grid. Red circular markers are used to mark the called out Bingo numbers. If five matching numbers are marked in a row, column, or diagonal line on a single card, the player is said to have a winning card.

The above Bingo card states are determined once the robot recognizes that a player requires assistance via the *User State* module and the robot has navigated to the user. The robot captures a set of *n* 2D images (provided by the Axis camera) of the user's card on the table. A user's card is first identified in the captured image by determining the unique symbol on a player's card using Speeded-Up Robust Features (SURF) [17]. This is followed by a Hough Transformation based methodology to accurately identify the locations of circular markers on the Bingo card grid. Marked grid squares are then identified by applying a nearest neighbor classifier between the identified grid square and marker centroids. Bingo numbers that have been called out are then compared with the numbers marked on a player's card to provide a classification of the activity state. In cases where Tangy has identified the location of less than 25 grid squares the card is considered as occluded (e.g., the card is outside of the robot's camera view or a hand is partially covering the card) and the robot will request that the user move his/her card closer on the table to the robot.

6) Person Identification

The *Person Identification* module identifies individual users in order to provide reminders or assistance during the Bingo activity. In both activities this occurs after the robot has navigated to the user's location. The OKAO™ Vision software library [18] is utilized to identify users based on facial shape models containing eyebrow, eye, nose, mouth, and face contour features utilizing images provided by the Logitech camera. These face models are then compared to face models stored in a database of each user to identify the individual users the robot is interacting with. Herein, user faces can be identified within a 2.25m range of the robot and with face orientations varying within 30^o, 20^o, and 360^o in yaw, pitch and roll rotations from a direct frontal view of a user's face. After a user has been identified, Tangy will greet the user and proceed with the next step for each activity.

7) User State

The *User State* module is utilized only during the Bingo activity and monitors the players during the game to determine whether they require assistance as defined by a raised hand gesture. Herein, this gesture is referred to as the assistance request gesture. 3D sensory information provided by the ASUS Xtion sensor is used to identify this gesture by first determining the 3D joint locations of each player from his/her detected and tracked skeletons using the OpenNI software development kit [19]. A spatial relationship between each player's elbow, wrist, and head joint is then used to classify whether a player has raised his/her hand. This approach is capable of tracking an unlimited number of users and is only limited by the processing power and field of view of the sensor [20]. Once a player has been identified to require assistance, Tangy will navigate to a location in front of the player (determined using the 3D sensory information) and then begin identifying him/her.

8) Behavior Deliberation

The *Behavior Deliberation* module determines the appropriate behaviors for Tangy for both the reminder and Bingo activities. Namely, the module selects these behaviors using a finite state machine (FSM) to achieve the overall objective of reminding all users about their scheduled Bingo activities and to facilitate the Bingo activity itself.

The Reminder activity has two distinct sequential behaviors: 1) navigating to where a user is located, and 2) providing a reminder to the specific user. These behaviors are determined using the FSM shown in Fig. 2. Input into the FSM is from the *Activity State* module. When the robot has navigated to a user's location, the robot will identify the user and greet him/her by name. It will then remind the user about the scheduled Bingo activity that day by telling him/her the location and time of the activity while displaying the location and time on its screen. Example robot behaviors for the reminder activity are presented in Table I.

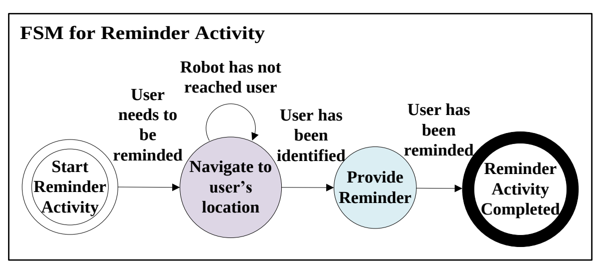


Figure 2. Reminder Activity Finite State Machine

TABLE I. EXAMPLE ROBOT BEHAVIORS FOR THE REMINDER ACTIVITY.

Behavior Type	Example Behavior
Navigating to a user's location	Robot moves from its present location to the location of the person whom it needs to remind. (The identity of the user and his/her location is provided by the centralized server)
Providing a reminder to a user	*"Hi John, the next scheduled Bingo activity is in the games room at 2PM."* (Robot waves when greeting the user and points at its screen where the location and time of the Bingo game is displayed)

The main behaviors for the robot facilitating the Bingo activity are separated into three categories during the game, Fig. 3: 1) calling out Bingo numbers, 2) providing assistance, and 3) providing social utterances. At the beginning of a game, Tangy greets all players and introduces itself. The robot then begins calling out random Bingo numbers. If it is identified that a player requires assistance, the robot will stop number calling and navigate to the table in front of this player. The robot will then greet the identified player by name and determine what type of assistance the player needs. If a player has either incorrectly marked a Bingo number or did not mark a called Bingo number, Tangy will provide help by either requesting the player to unmark the incorrect number (while pointing to the number displayed on its screen) or to mark the missed number (while pointing to the number on the card). If a player has Bingo, the robot will celebrate the winning card by congratulating the player. If a player has asked for assistance, however, the card has been correctly marked, Tangy provides encouragement to the player to continue the game. In situations where the robot's view of a player's Bingo card is partially occluded, Tangy will request that the player bring the card closer to it on the table. After completing an assistance behavior, the robot will return to the front of the room to continue calling out Bingo numbers if no players have won the game. To promote the social dimensions of playing Bingo, Tangy also provides jokes and positive statements every five minutes (while calling out Bingo numbers). Example robot behaviors are presented in both Table II and Fig. 4.

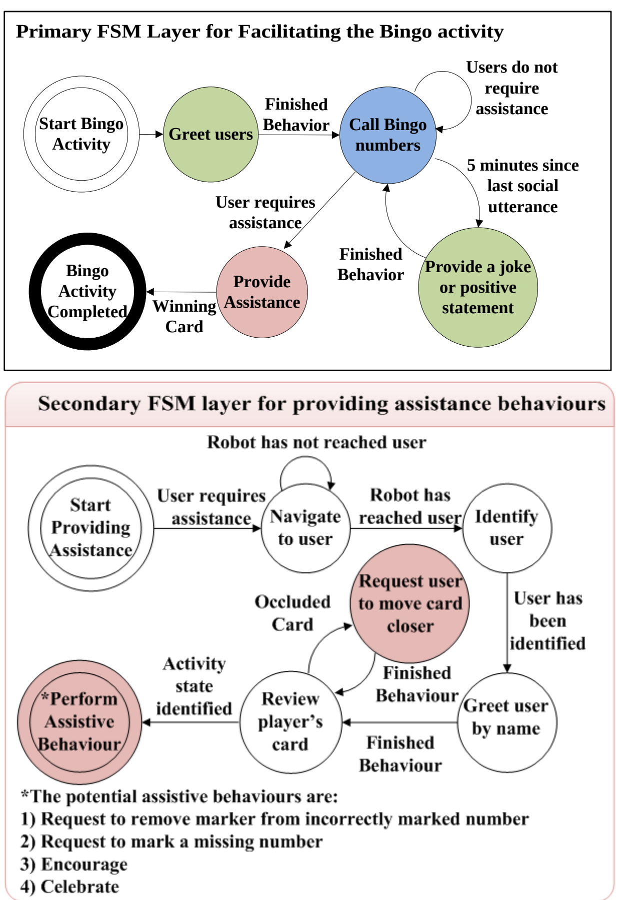
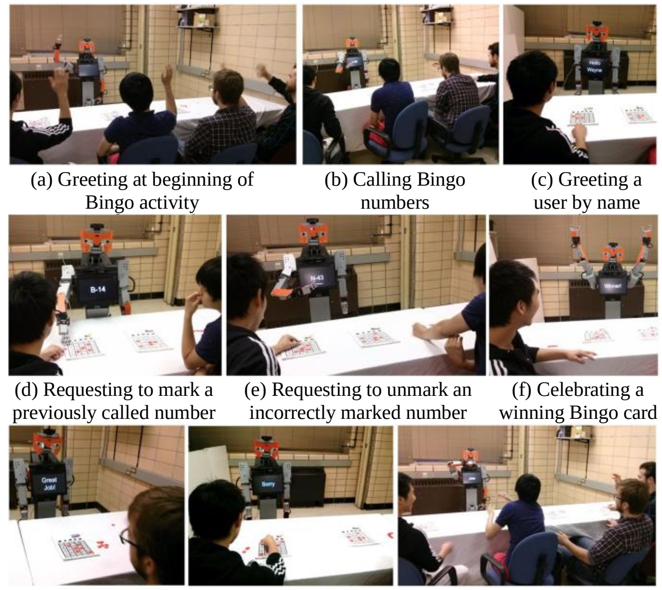


Figure 3. Bingo Activity Finite State Machine



(a) Greeting at beginning of Bingo activity
(b) Calling Bingo numbers
(c) Greeting a user by name
(d) Requesting to mark a previously called number
(e) Requesting to unmark an incorrectly marked number
(f) Celebrating a winning Bingo card
(g) Providing encouragement to a player who does not have Bingo
(h) Requesting player to move his/her card closer to the robot
(i) Telling a joke

Figure 4. Example Robot Behaviors for the Bingo Activity

TABLE II. EXAMPLE ROBOT BEHAVIORS FOR THE BINGO ACTIVITY.

Behavior Type	Example Behavior
Provides social utterances *1) Greeting*	"Hi everyone, My name is Tangy. Are you ready to play Bingo?" (Robot waves at players)
2) Joke	"How do you organize a space party? You Planet!!" (Robot brings its hand to its mouth and giggles)
3) *Positive statements*	"I am having so much fun playing Bingo with all of you today!"
Calls out Bingo numbers	"The next number is I-16. Please mark I-16 on your Bingo card with a red marker." (Robot points to number displayed on its tablet)
Provides assistance 1) *Help: a) Request to mark a missing number on the Bingo card*	"Hi Wayne, B-14 was previously called. Please mark this number on your card with a red marker. Wow, you are getting close to having Bingo" (Robot points to missed number on player's card)
b)Request to remove marker from incorrectly marked number	"Hi Wayne, unfortunately N-43 has not been called yet. Please remove the red marker covering this number on your card." (Robot points to number displayed on its tablet)
2) Celebrate	"Congratulations! You've won BINGO!" (Robot raises both its arms straight up and sways them side to side in a celebration dance)
3) Encourage	"All the numbers on your card have been marked correctly. Keep up the great work. You are very close to getting Bingo!"(Great job! displayed on robot's tablet)
4) Request to move card closer to robot	"Unfortunately, I cannot see your card. Would you please kindly move the card closer to me on the table so I can see it?"

9) *Low-Level Controller*

The overall behaviors of Tangy are implemented using the low-level controller. The controller consists of Navigation and Interaction layers. The Interaction layer generates the robot's verbal (synthesized voice) and non-verbal (graphical display and gestures) communication during both the reminder and Bingo activities. The Navigation layer allows the robot to autonomously move within its environment to the locations of interest. We utilize the ROS navigation software package [21] to perform autonomous navigation using 3D information from a Hokuyo URG-04LX-UG01 laser range finder mounted on Tangy's base.

IV. Experiments

We conducted a system performance experiment of the proposed multi-user system architecture. The system performance testing will allow us to evaluate the ability of the architecture to: 1) plan and schedule daily Bingo activities for the robot Tangy to facilitate while considering the individual schedules of the users, and 2) determine the appropriate behaviors of the robot with respect to individuals and groups of people while providing reminders prior to a game starting and during a Bingo activity itself.

A. *Tested Scenarios*

The following settings are considered as input into our centralized server:

1) A single day commences at 8am (when a potential user wakes up) and ends at 7pm (when all scheduled activities should be completed). Initial planning and scheduling takes place prior to the start of the day. The schedules of users and the requests for Bingo games (i.e., location, duration, users) are known a priori.

2) Across the user schedules, there are three mandatory breaks during the day in which all users are considered unavailable: breakfast (8am-9am), lunch (12pm-1pm), and dinner (5pm-6pm).

3) Each user has his/her own distinct schedule for the day with two or three 1-hour appointments already scheduled in addition to the aforementioned meals. When not at these appointments, the users are located in their own rooms within a single floor of a building.

4) There are four players that interact in a single Bingo game which always occurs at the *games room*.

5) An indoor environment topology is specified a priori which includes corridors that connect users' locations, the charging station of the robot and the Bingo game room.

Given the above assumptions, the five scenarios tested in this work are the following: *Scenario 1*- 4 users and 1 Bingo activity request; *Scenario 2*- 8 users and 2 Bingo activity requests; *Scenario 3*- 12 users and 3 Bingo activity requests; *Scenario 4*- 16 users and 4 Bingo activity requests; and *Scenario 5*- 20 users and 5 Bingo activity requests.

Our performance metrics for these scenarios with respect to the planning and scheduling system are: 1) the runtime for producing schedules for Tangy's activities, and 2) the number of states evaluated by the planner to find a solution to the planning and scheduling problem. Moreover, we analyze whether: 1) the server, through the *Execution and Monitoring system*, effectively sends the activity commands to the robot, and 2) the robot's activity request and command manager assigns the appropriate activity to the robot.

Once Tangy's onboard controller has received an activity to implement from the server, we investigate the robot's effectiveness in implementing the appropriate behaviors for that particular activity. Due to space constraints, herein, we only present the physical implementation of the most complex scenario, *Scenario 5*, on the robot with twenty adults ranging in age from 21 to 32 ($\mu = 24$, $\sigma = 3.4$). Prior to robot interactions, the test scenario was explained to the participants.

To provide game reminders, Tangy navigated down a hallway connected to different rooms where users were located (room doors were opened in the experiment) based on the plan generated by the *P&S system* and, once the robot reached the target user location, the robot identified the user and provided the appropriate reminder, Fig. 5(a)(b). All users were reminded prior to their assigned Bingo game. Bingo games were conducted in a games room at the end of the hallway. At the beginning of each game, the robot would situate itself at the front of the room facing the four participants. Participants were seated 4 m from the front of the room behind a row of tables, Fig. 5(c). Each participant was provided with one Bingo card and 25 red circular markers. Interactions would begin with the robot greeting the participants and would end when the robot identified a player`s winning Bingo card. For these experiments, we measured the robot's ability to determine its appropriate behaviors during the Bingo activity based on the identified user, and the recognized user and activity states.

(a) Rooms where users are located (b) Providing a reminder (c) Bingo activity with four players

Figure 5. Reminder and Bingo Activity Scenarios

B. *Results and Discussion*

Table III presents the performance of the *P&S system* using the OPTIC temporal planner with respect to the five investigated scenarios. The planner solves all the planning and scheduling problems in less than 110 seconds. The planner was able to provide valid schedules for the robot in order to deliver reminders and facilitate games at available times during the day.

TABLE III. Temporal Planner's Performance for Each Scenario

Scenarios	Runtime (s)	Number of States Evaluated
Scenario 1	1.86	1764
Scenario 2	16.34	8183
Scenario 3	26.12	10879
Scenario 4	35.84	11976
Scenario 5	109.02	33768

Table IV presents the assignment of the activities to the robot as generated and requested by the server for the aforementioned five tested scenarios. As can be seen in the table, all of the activity commands were successfully sent to Tangy's *Activity Request and Command Manager* which then appropriately assigned these activities to the corresponding robot controller modules. In all the scenarios, robot recharging was not determined to be a necessary activity to implement given the amount of Bingo activity requests. However, the temporal planner did consider this while planning the robot HRI activities.

The performance of the robot in the physical implementation of *Scenario 5* for the reminder and Bingo game activities are presented in Tables V and VI. For the reminder activity, a success rate of 100% was obtained for navigating to a user's location, while a 95% success rate was obtained for identifying the user and providing the reminder.

TABLE IV. SENT ACTIVITY COMMAND RESULTS

	Expected Activity from the Robot			Success Rate
Commands sent by the server	*Remind*	*Play Bingo*	*Recharge*	
Remind	60	0	0	100%
Play Bingo	0	15	0	100%
Recharge	-	-	-	-

TABLE V. REMINDER ACTIVITY RESULTS

True Person Identified	True Activity State	Expected Robot Behavior	Success Rate
-	User needs to be reminded	Navigating to a user's location	100%
User Identified	User needs to be reminded	Providing a reminder to a user	95%

TABLE VI. ROBOT BEHAVIOR EXECUTION RESULTS

True User State	True Person Identified	True Activity State	Expected Robot Behavior	Success Rate
Hand not raised	-	Start of Bingo game	Greeting	100%
Hand not raised	-	-	Joke	100%
Hand not raised	-	-	Positive Statement	100%
Hand not raised	-	-	Call out Bingo Number	100%
Hand raised	Player identified	Winning Bingo card	Celebrate	100%
Hand raised	Player identified	Bingo card missing marked numbers	Help: Request to mark a missing number on the Bingo card	80%
Hand raised	Player identified	Incorrectly marked Bingo card	Help: Request to remove marker from incorrectly marked number	100%
Hand raised	Player identified	Correctly marked Bingo card	Encourage	100%
Hand raised	Player identified	Occluded Bingo card	Request to move card closer to robot	100%

One user, even though in the appropriate location, was not properly identified by the *Person Identification* module during the first attempt by the robot. Only after the person actively moved towards the robot, the robot was able to identify this particular user. For the Bingo game, the robot had a 100% success rate at performing all the expected behaviors, except for the behavior to request a user to mark a missing number on his/her card which had an 80% success rate. For this behavior, the robot was unable to identify the Bingo card and in turn asked the player to bring the card closer to the robot rather than directly providing the help assistance. However, after the player brought the card closer to the robot, the robot was able to correctly provide the request to mark a missing number behavior.

V. CONCLUSION

In this paper we present the development of a novel multi-user HRI system architecture for the mobile social robot Tangy to autonomously plan, schedule and facilitate group activities while considering the schedules of multiple users. We consider the scenario where Tangy needs to schedule several Bingo activities throughout the duration of the day, and provide a reminder regarding the time and location of a game to each user prior to game commencement. The results of system performance testing are promising and show that the proposed system was able to schedule Bingo game activities during the day based on the schedules of multiple users as well as provide the robot with appropriate activity plans based on the overall schedule. In addition, our physical experiments with 20 Bingo players demonstrated that Tangy was able to effectively deliver reminders prior to the scheduled games, and execute and facilitate different Bingo games with groups of users while providing individualized assistance. Our future work will include extending the modules of the architecture to include different approaches for re-planning and developing additional robot behaviors for the HRI activities. Experiments will also be conducted with larger groups of Bingo players.

REFERENCES

[1] C. Bartneck and J. Forlizzi, "A design-centred framework for social human-robot interaction," *IEEE Int. Workshop on Robot and Human Interactive Communication*, pp. 591-594, 2004.

[2] D. Traum, "Issues in multiparty dialogues," in *Advances in agent communication*, Berlin, Heidelberg: Springer, pp. 201-211, 2004.

[3] M. Bennewitz et al., "Towards a humanoid museum guide robot that interacts with multiple persons." *IEEE-RAS Int. Conf. Humanoid Robots*, pp. 418-423, 2005.

[4] A. Hristoskova et al., "Personalized Guided Tour by Multiple Robots Through Semantic Profile Definition and Dynamic Redistribution of Participants." *Conf. Artif. Intell.*, pp. 1-7, 2012.

[5] R. P. A. Petrick and M. E. Foster, "Planning for Social Interaction in a Robot Bartender Domain." *Int. Conf. Automated Planning and Scheduling*, pp. 1-9, 2013.

[6] M. Ghallab, D. Nau, P. Traverso, *Automated Planning: Theory and Practice*, San Francisco, CA, USA: Morgan Kaufman, 2004.

[7] D.E. Smith, J. Frank and A.K. Jonsson, "Bridging the Gap Between Planning and Scheduling," *Knowl. Eng. Rev.*, 15 (1), pp. 1-34, 2000.

[8] M. Beetz and M. Bennewitz, "Planning, Scheduling, and Plan Execution for Autonomous Robot Office Couriers," *Integrating Planning, Scheduling and Execution in Dynamic and Uncertain Environments*, Workshop Notes, 1998.

[9] A. Cesta et al., "Monitoring Older People with the RoboCare Domestic Environment: Interaction Synthesis and User Evaluation," *Comp. Intell.*, 27 (1), pp. 60–82, Wiley Periodicals, 2011.

[10] J. Pineau et al. "Towards Robotic Assistants in Nursing Homes: Challenges and Results," *Robotics and Autonomous Systems*, 42(3-4), pp. 271–281, 2003.

[11] B. Coltin et al., "Dynamic User Task Scheduling for Mobile Robots." *AAAI Workshop on Automated Action Planning for Autonomous Mobile Robots*, pp. 1-6, 2011.

[12] M. Pollack, "Intelligent technology for an aging population: The use of AI to assist elders with cognitive impairment," *AI Magazine*, 26(2), pp. 9–24, 2005.

[13] B. P. Sobel, "Bingo vs. Physical Intervention in Stimulating Short-term Cognition in Alzheimer's Disease Patients," *American J. of Alzheimer's Disease and Other Dementias*, 16(2), pp. 115-120, 2002.

[14] Robot Operating System (2013). Available: http://www.ros.org.

[15] J. Benton, A. Coles, and A. Coles, "Temporal Planning with Preferences and Time-Dependent Continuous Costs," *Int. Conf. on Automated Planning and Scheduling*, pp. 2-10, 2012.

[16] A. E. Gerevini et al., "Deterministic Planning in the Fifth International Planning Competition: PDDL3 and Experimental Evaluation of the Planners," *Artif. Intell.*, 173, pp. 619–668, 2009.

[17] B. Herbert et al., "Speeded-Up Robust Features (SURF)", *Computer Vision and Image Understanding*. 110(3), pp. 346-359, 2008.

[18] Omron. (2007). *OKAO Vision*. Available: http://www.omron.com/r_d/coretech/vision/okao.html

[19] OpenNI. (2010). OpenNi. Available : http://www.openni.org.

[20] PrimseSense Inc. (2011). PrimseSenseTM NITE Algorithms 1.5, Available: http://www.openni.org/wp-content/uploads/2013/02/NITE-Algorithms.pdf.

[21] E. Marder-Eppstein et al., "The office marathon: Robust navigation in an indoor office environment", *IEEE Int. Conf. Robot*, pp. 300-307, 2010.