Masquerade: A Cognitive Robotics Game and its First Player

Stephen Pfetsch
University at Buffalo
Email: spfetsch@buffalo.edu

***Abstract*—This paper documents the author's work in a cognitive robotics seminar. The results include both the formulation of a competitive cognitive robotics game and the construction of an autonomous robot. The game is designed to encourage the development of cognitive robots research and technology through direct competition between teams. Designed to be the first robot capable of competing in a simplified version of the game, the autonomous robot constructed for the seminar requires additional work in order to play.**

I. Introduction

Despite multiple decades of work in the field of robotics, there is a persistent barrier between robots that execute procedural behaviors and those that act as autonomous cognitive agents. Despite the level of sophistication reached by the procedural robots, to the point of being able to play soccer as a team, cognitive robots still require significant work to reach similar milestones. It would be productive to move beyond *procedural robotics* to a paradigm that puts knowledge representation and reasoning in control of the robot.

In the past it has been popular to foster the growth of robotics technology through competitions. These exercises range from student activities such as Micromouse [3], Sumobots [4], and Firefighter robots [5] to research-oriented competitions, such as Robocup [6] and the DARPA Grand Challenge [7]. The field of Cognitive Robotics is still a fledgling area of research and thus there are no competitions that are structured to reward advanced reasoning. Over the long term it will be necessary for such events for cognitive robotics to keep pace with other areas of robotics.

This paper proposes a cognitive game to be played by robots. Attention is given to developing rules that encourage memory and reasoning so as to reward participants who push the state-of-the-art in cognitive robotics. Additionally, this paper contains a description of a robot being developed to become the first player in this competition.

II. The Game

Robotics games and competitions have traditionally been used to promote progress in specific areas of robotics. For instance, Robocup was designed to foster the development of advanced team strategies and eventually agile humanoid robotics, while the DARPA Grand Challenge was organized to stimulate research in autonomous vehicles. In a similar vein, this paper proposes a game that encourages teams to compete in building cognitive robotic agents capable of outwitting their opponents. The design of such a competition must take into account several criteria to ensure the activity is relevant to modern cognitive robotics research.

Firstly, special attention must be given to make sure the challenge is feasible with current technology and techniques. It will be desirable to produce a game scenario that allows roboticists and researchers to develop a robot capable of competing in the game according to its basic rules, but leaves ample room for them to continuously improve their algorithms. To further meet this goal, this paper details an initial simplification of the game that may be played with very early robots.

A second criterion to consider is the need for a scenario that is rich enough to foster continued growth as robots improve. In particular, it will be beneficial to make sure that the difficulty of the competition arises from the quality of opponents instead of directly from the rules of the game.

The final criterion under consideration is that the game should go to great lengths to ensure that effective robot strategies naturally must be the result of work toward the theme of the competition. In this case, the use of reasoning should afford a team an advantage over teams that choose to take shortcuts around the cognitive aspect of the competition. By intention, it will be the level of cognitive sophistication that determines the winner of the game.

With the above criteria taken into consideration, an original game scenario has been created. The game is based on the concept of an interactive murder-mystery performance, in which an audience participates as actors reveal a murder and provide clues the audience will need to determine the identity of the killer. In the same vein, this competition has robots interact as courtiers at a masquerade ball, with one of the robots secretly given a mission to kill the "king." The murderer must act inconspicuously while subtly carrying out the crime. Meanwhile, the remaining courtiers must mingle and observe each other's actions as they look for clues as to who is the killer. The murderer wins by committing his crime undetected and a courtier wins the game by revealing the identity of the killer and thwarting the murder.

A. The Playfield

Play for this game would take place within a special arena that has specific zones and objects designated for various activities. The king sits at a throne at one end of the playing field and punch bowl is placed at the other. Additionally, there are several objects scattered around the field, including a pistol, gun powder, and a musketball. While these items may indeed be represented by realistic tangible objects once competing robots have reached a given level of sophistication, proxy objects are to be expected. For instance, early versions of this game may employ color-coded paper squares, advanced games could use drawings or photographs of the items. Similarly, despite the gameplay requiring that robots pick up or manipulate objects, early games will substitute simpler interaction, like touching or driving on top of an item to express holding an item.

B. Murder Conditions

There are several specific methods by which a killer can murder the king. These methods embody specific styles that allow competitors to develop a preferred murder strategy. This also complicates the role courtiers must play in identifying suspicious activity. It is expected that some methods will arise as easier to execute than others.

1) Poisoning the Punch: In order to kill the king by poisoning the punch, a murder must execute the following steps:

1) Drop poison into the punch bowl
2) Wait *X1* seconds
3) Hide the poison bottle
4) Wait *X2* seconds

Depending on the representation of the punchbowl, the act of poisoning the punch would be represented differently. An example representation is a clear bowl filled with red marbles. Poison would be symbolized by black marbles, which would be carried by the robot. A simpler representation might declare a region of the floor to be the bowl, and all robots in that region are considered to be standing at the bowl. In this case, poisoning could be represented by dropping a unique object in this region. After the initial poisoning, there would be a delay period to give other robots an opportunity to discover the poison. Once the delay is period has passed, the killer must deposit another object that represents the poison bottle anywhere on the field. Another delay period is enforced, after which the murder is declared a success. Note that robots competing in this competition would have to be designed or modified to carry and drop the required items, but for simplicity's sake, they must be carried by all robots at all times. Poisoning is considered a stealthy murder because it requires discreet activity without any social interaction.

2) Shooting: A killer may shoot the king directly by completing the following steps:

1) Collect pistol
2) Collect gun powder
3) Collect musket ball
4) Fire at the king

This attack consists of collecting three objects in any order, then approaching the king directly to shoot him. The act of collecting the objects can may be represented by nudging or driving over the object in question, while shooting the king may be represented by nudging the object that represents the king. Since this attack is very direct, it is considered an overt or aggressive strategy. As such, a courtier merely has to notice a robot picking up the objects in order to suspect that he is the murderer. There are no enforced delays because the aggressive killer is not worried about concealing his actions. On the contrary, he intends to shoot the king in a room full of witnesses.

3) Booby trapping: A booby trap is set for the king if murderer performs the following actions:

1) Collect gun powder
2) Speak with the king
3) Place the powder at the thrown
4) Speak with the king
5) Wait X1 seconds

In this case, the attack involves speaking with the king. The act of speaking can be represented by nudging once again, as will be discussed later. A courtier is not allowed to speak to the king until he is considered worthy. As in a historical royal court, popularity will be important in determining a courtier's worth, so implicitly the murderer will have to become popular in the court before attempting this assassination. By comparison to poisoning and shooting, this is the first *social* murder.

4) Dusting with arsenic powder: The final method of murdering the king is as follows:

1) Speak with the king
2) Speak with the queen
3) Obtain the king's scarf
4) Take the scarf to the bathroom and dust it
5) Return the scarf
6) Wait *X1* seconds

This murder concerns the use of a scarf that is awarded to only the most popular courtiers as a sign of respect. There are other uses for the scarf, but the murderer is interested in taking the scarf to a hidden place to coat it with arsenic powder. The killer requires a much higher social standing for this approach than for setting a booby trap. A queen is mentioned in this example as a placeholder for as yet unspecified additional rules to complicate the task of acquiring the scarf. For instance, perhaps before receiving a scarf, the courtier must do something special to impress the queen.

C. Courtier conduct

When robots are not selected as the assassin, they must ensure that they abide by the rules of the court. The killer must follow these rules to blend in with the other courtiers in addition to performing the murder. Firstly, the courtiers must stay well-hydrated or else they will pass out. The punch bowl quenches the courtiers' thirst and therefore they must visit the punch bowl periodically. In the game, thirst translates to a timer that is reset by drinking punch. An external referee computer would be responsible for monitoring a robot's thirst timer and deciding when a courtier has passed out and should be removed.

The second requirement of courtiers is that they socially interact with other robots to maintain popularity. Their social status cannot drop below a certain limit or they will be banished from the court. Additionally, courtiers are not allowed to speak to the king if their popularity is too low. However, once a courtier's status has risen to a very high level, the king will grant them a scarf as a token of respect. The scarf can only be held for a limited time before the king takes it back, but while holding it, a courtier is privy to additional knowledge "gossip" that may help him determine the killer.

Finally, the courtiers must be flexible in their understanding of the court. At the beginning of the game, the parameters of the world will be described to the robots. For instance, if the day is said to be very hot, the robots will experience thirst much faster and therefore they will have less time between trips to the punch bowl. On a given day, the king might declare a taboo that results in immediate dismissal from the court, such as walking in a certain area.

At any time, one courtier may accuse another of being a murderer. A false accusation will cause the accuser to be expelled from the kingdom, but no other courtiers will be told which of them was falsely accused so the play can continue.

III. First Generation Simplification

For the first implementation of this game, it may be necessary to greatly reduce the complexity of the environment and the number of rules. the minimal playfield should include a punch bowl and the king, as well as a few props. However, each item should probably be represented as a colored object or sheet of paper. Robots will also be given unique colored hats to wear to simplify identification. Such a simplification is desirable to keep the focus of the game on cognitive reasoning rather than computer vision.

Taking the Rovio [1] as the reference robot for this game, it is assumed that each robot is controlled remotely by a computer and that all computers and robots are on the same wireless network as a special referee computer. The referee computer will send certain information to the robots that will be reduce confusion amongst the players. At the beginning of the game, each robot will receive a description of the parameters for that game, their status as killer or not, and so on. During the game, the referee will provide artificial stimulus that the robots can trust. Notably, the referee will alert the robots when they have had a sip of punch, when they've picked something up, and it will tell them their current social status. Additionally, the referee will accept accusations from robots and determine whether a robot wins. When deemed necessary, as when a robot is being removed from play, the referee will "freeze time" and expect all robots to cease moving. In this simplified version of the game, a human will provide the referee computer with information when key events occur, such as telling the computer when a robot enters the floor space designated as the punch bowl.

IV. First Agent in the Game

The first robot that was designed and programmed to be used in this competition is a custom modification of a Roomba vacuum cleaner [2] that is capable of complete autonomy.

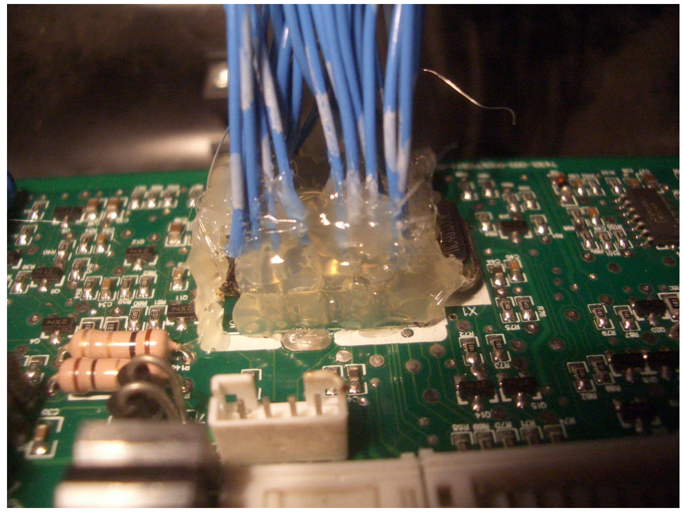

Fig. 1. The Roomba's processor after being rewired.

A. Some Background about the construction of the robot

The robot modified for use in this game does not have any built-in computer control interface off-the-shelf. Instead, it was necessary to first disassemble the Roomba to remove its circuit board and reverse-engineer the wiring of its processor. Building on the information provided by those who have hacked a Roomba in the past [8], the functionality of each pin was determined and documented [9]. Wires were attached to each of the pins of the processor, as shown in Figure 1, to provide a means to override the default vacuum behaviors of the robot. At this point, the author designed and implemented a microcontroller replacement for the onboard processor whose purpose is to enable an ordinary computer to control the Roomba's motors and read its sensor state through a serial port. The microcontroller is seen interfaced to the robot in Figure 2. The microcontroller was incorporated into a circuit that was placed inside the robot so that only a serial cable protrudes from the interior.

In order to perform as an autonomous robot, it was necessary to configure a small, battery-powered computer that would sit on top of the robot. A motherboard with a small form-factor and embedded CPU was selected and it was outfitted with a 4GB Compact Flash card instead of a hard drive. A wireless network adapter was incorporated into the robot to support connections to other computers for observation and control, although the robot is fully capable of controlling itself. Linux was selected as the operating system and various software was installed, including CMU Common Lisp, the SNePS reasoning system, the OpenCV computer vision library, and an OpenSSH server.

The software controlling this robot is intended to be written primarily in Common Lisp with SNePS. It was necessary, however, to develop a library in C that communicated with the microcontroller and simultaneously makes the robot's functionality available to the Lisp programs. This step was completed by first developing a C++ class that encapsulates

Fig. 2. The microcontroller wired to the robot.

Fig. 3. The fully autonomous robot with onboard computer

everything the robot can do and surrounding this class with a CMUCL Alien interface to make the class's methods callable from Lisp. An image of the completed robot is shown in Figure 3.

B. Capabilities of the robot

The robot has numerous infrared sensors and touch sensors. The robot has a wall-following infrared sensor so that it can maintain a certain proximity to a wall while traveling parallel with the wall. It has four individual infrared floor sensors that allow it to detect a cliff and avoid falling. There is an additional infrared sensor on the top of the robot that allows it to detect "virtual walls", which are artificial barriers produced by a device by emitting a beam of infrared light to indicate when the robot has reached a boundary. Several physical sensors are available as well, to detect when the robot's wheels are on the ground and when it has bumped into a wall.

Two wheels are responsible for the movement of the robot. They can move separately in either direction at variable speed, which allows the robot to turn in place, drive forward and backward, and turn with an arbitrary radius. The wheels have encoders to provide feedback on the distance they have rotated, which allows the robot to approximate where it has moved.

Additionally, the robot has a small speaker capable of producing beeps, tones, and simple music. External computers can SSH into the robot via wireless and take control by running programs that implement autonomous behavior or allow interactive control. Additional peripherals such as a camera may be added to the robot in the future using its three USB ports.

C. The Robot's Future Reasoning

Currently the robot is merely a platform on which the first implementation of a masquerade robot will be built. Once a camera is integrated into the robot platform, some elementary vision-based navigation of the playfield will be developed. It may be necessary to incorporate some distinct visual landmarks into the terrain to simplify the robot's identification of its surroundings. Using a low-level algorithm written in C, the Lisp/SNePS layer will receive approximations of its position in the environment so that it can take this into account while acting.

V. Conclusion

The game described in this paper is an approximation of the kind of competition that may be capable of fueling interest in cognitive robotics. In the absence of a definite application for today's state-of-the-art cognitive robots, it would be beneficial to provide a synthetic goal to encourage work in the field. The implementation of the first game player is incomplete, but it should be developed in the coming months.

References

[1] Wowwee Corporation, *Meet Rovio*, http://www.meetrovio.com/.

[2] iRobot Corporation, *iRobot Corporation: Home Page*, http://www.irobot.com/.

[3] *Micromouse Online*, http://www.micromouseonline.com/.

[4] D. Cook, *An Illustrated Guide to American Robot Sumo*, http://www.robotroom.com/SumoRules.html.

[5] Trinity College, *Fire Fighting Home Robot Contest*, http://www.trincoll.edu/events/robot/.

[6] The Robocup Federation, *Robocup Official Site*, http://www.robocup.org/.

[7] DARPA, *Grand Challenge Home*, http://www.darpa.mil/grandchallenge05/.

[8] M. Giambalvo and E. Paradis, *Webplayer-Roomba Integration*, http://web.archive.org/web/20051225014747/heathkit.mondrary.com/virgin/.
[9] S. Pfetsch, *Roomba Reverse-Engineering Spreadsheet*, http://spreadsheets.google.com/pub?key=pRyhXWV13j_qZfM2sEDaFOA.