Robot Deception and Squirrel Behavior: A Case Study in Bio-inspired Robotics

Jaeeun Shim and Ronald C. Arkin
Mobile Robot Laboratory, School of Interactive Computing
Georgia Institute of Technology, Atlanta GA 30308, USA
jaeeun.shim@gatech.edu, arkin@cc.gatech.edu

Abstract
A common behavior in animals and human beings is deception. Deceptive behavior in robotics is potentially beneficial in several domains ranging from the military to a more everyday context. In our research, novel algorithms were developed for the deceptive behavior of a robot, inspired by the observed deceptive behavior of squirrels for cache protection strategies, then evaluating the results via simulation studies. In this paper, we present this bio-inspired algorithm for robot deception and observe whether the algorithm is truly applicable in actual robot systems.

1. Introduction

A common and essential behavior for survival in a variety of intelligent systems ranging from insects to human beings is deception. Many biologists and psychologists define deception in various ways. According to Vrij (Vrij, 2001), deception is "A successful or unsuccessful deliberate attempt, without forewarning, to create in another a belief that the communicator considers to be untrue in order to increase the communicators' payoff at the expense of the other side." Da Waal argued that "Deception can be defined as the projection, to one's own advantage, of an inaccurate or false image of knowledge, intentions, or motivations" in his paper (de Waal, 1992). We can find a simpler definition of deceptive behavior from a paper by Bond and Robinson (Bond & Robinson, 1988) who defined it as "a false communication that tends to benefit the communicator." We have used this straightforward definition in earlier research in our laboratory on deceptive behavior for robots (Wilcox & Jackson, 1998) and we continue to do so here.

In other words, animals act deceptively to gain benefits from others. Biological and psychological findings show that deception plays an important role in providing an evolutionary advantage (Bond & Robinson, 1988). It appears in higher-level primates to involve a theory of mind mechanism (Cheney & Seyfarth, 2008). We argue that robots can also potentially gain advantage over adversaries by possessing deceptive behaviors. For example, it is obvious that the use of deception is important with respect to the military context (Meehan, 1988). We further posit that to achieve more socially intelligent robots operating in the presence of humans, we must develop robots that interpret, generate, and respond to deceptive behavior. Therefore, we investigate deception in robotics using approaches inspired by biological findings (Wilcox & Jackson, 1998; Davis & Arkin, 2012).

In this article, a novel approach for deceptive behavior by a robot is presented, inspired by observations of squirrels in cache protection strategies (Steele, et al., 2008). Section 2 reviews relevant animal and human deceptive behaviors and existing research in robotic deception. In Section 3, specific deceptive behaviors in food hoarding and protection strategies of squirrels are introduced. In Section 4, a computational model enabling robots to emulate these deception behaviors of squirrels is integrated into *MissionLab*, and the results are discussed in Section 5, including both simulation and real robot experimental results. Section 6 concludes the article. We note that we are well aware of the ethical implications associated with robotic deception and our perspective on this subject is discussed elsewhere (Arkin, 2011).

2. Related Work

2.1 Animal Deception

Animals use various forms of misinformation. These deception mechanisms, achieved by sending false signals either intentionally or unintentionally, are essential for animals' survival. For example, camouflage and mimicry are well known in many species. By resembling other animal species or inanimate objects, animals transmit misinformation to others so that they can avoid detection by both predators and their prey. While camouflage or mimicry are examples of unknowingly deceiving, a deceptive behavior can include seemingly more intentional misinformation.

Many deceptive behaviors are observed from different animals ranging from insects to primates. The spider genus *Portia*, which preys primarily on other spiders, deceives their prey by vibrating the web in ways that resemble a small insect getting ensnared. When the web's resident spider comes to investigate the insects, *Portia* preys on it (Wilcox & Jackson, 1998). According to Ristau's experiment (Ristau, 1991), another interesting deceptive behavior appears in piping plovers. These birds exhibit a "broken-wing display" deceptive behavior. By feigning an injured wing and hopping farther and farther from the nest, birds lead the predator away from their young, thus protecting them.

Primates are the species most commonly ascribed with the ability to deceive (Cheney & Seyfarth, 2008; Gouzoules & Gouzoules, 2002). For example, chimpanzees have multiple deceptive behaviors with several different objectives. When chimpanzees find fruit, they do not move directly towards it so that they do not give any indication to the competitors that they have noticed its location. This food protective strategy is not dissimilar to the one described for squirrels later in this article. Deceptive behavior of chimpanzees is also observed during interactions with humans. According to one observation, a chimpanzee feigned having his arm stuck in the bars of his cage in order to lure a zookeeper nearby. As soon as the human entered to help free his arm, he leaped onto the zookeeper (de Waal, 1992).

Another relevant class of deceptive behavior occurs in the food hoarding strategies of animals. Food hoarding (caching) is an important type of animal behavior needed for their survival through periods when nourishment is not readily available. In particular, these caching behaviors are commonly observed in rodents such as hamsters or squirrels (Jenkins, Rothstein, & Green, 1995). This caching behavior is of particular interest as it can also be useful in a robotic context. In this article caching and protecting resources are investigated for application in a resource protection strategy. In the military domain, robots might face this situation, where it is important to discourage an adversary from discovering a protected site, so the application of these bio-inspired animal food protection behaviors can be particularly beneficial.

In this research, we focus specifically on the observed deceptive behavior of squirrels while they protect cached food acquired during hoarding (Steele, et al., 2008). Recent research in the field of biorobotics implemented a robotic squirrel (Joshi, Johnson, Rundus, Clar, Barbour, & Owings, 2011). In this study, *robosquirrel* are successfully used for long-term studies on rattlesnake behavior during squirrel encounters. Even though it provided a model of some aspects of a squirrel's behavior vis-à-vis predation, it did not include a squirrel's deceptive behaviors. Our research focuses on employing this form of deception to guide robots in their interaction with other competitive agents. Section 3 describes this set of behaviors in more detail.

2.2 Human Deception

Deception is common not just in animals but also in humans. Therefore, understanding human deception is also important to design robot deception capabilities. Human deception often requires planning and second-guessing when compared to what most animals are capable of. Many psychologists have offered various perspectives. According to Vasek (Vasek, 1984), the development of deception follows the development of other skills used in social understanding such as perspective taking, communicational/linguistic skills, and understanding one's own and other's intentionality. In other words, deception is a good indicator of the Theory of Mind mechanism (Baron-Cohen, 1995). Therefore, the ability to deceive can be used to assess children's developmental disabilities such as autism (Baron-Cohen, 2007).

Human deception routinely happens in sports. Work by Mawby and Mitchell (Mawby & Mitchell, 1986) illustrates principles used in sports to enact or avoid deception. For example, many players misinform opponents thereby redirecting the opponents' actions, and teams use tactics of deception that require the coordinated actions of several players. Recent work analyzed the anticipation skills of deceptive movement in sports (Jackson & Warren, 2006) that aimed to predict opponent's intended directions quickly and accurately. Research of this sort enables players to train using virtual agents that have anticipation skills. As a result, human players can improve their deceptive abilities in rugby, soccer, and handball (Brault, Bideau, Craig, & Kkulpa, 2010; Dessing & Craig, 2010; Vignais, Kulpa, Craig, Brault, Multon, & Bideau, 2010).

People have consistently used deception in warfare to cloak their intentions and movements (Hawthorne, 2006; Meehan, 1988; Gerwehr & Russell, 2003). In the military sense, the term "deception" is applicable to "any planned measure undertaken for purposes of misleading or deceiving the enemy" (Sexton, 1986). Often a "story" is an essential instrument for executing a military deception. Here, "story" means a detailed scenario of "that which you want the enemy to think in order to make him do what your commander wants him to do." Because the military typically utilizes relatively fixed scenarios drawn from histories of thinking and acting, military deception is feasible and frequently implemented.

For animal instances, deception can be defined as "a false communication that tends to benefit the communicator (Bond & Robinson, 1988)." In other words, animals usually act deceptively to accrue benefits for themselves. An interesting aspect of human deception is that people sometimes deceive for the benefit of the deceived (Lewis & Saarni, 1993; Barnes, 1994; Ariely, 2012). For example, the use of placebos aims for a beneficial effect on patients, one form of deception employed by doctors/nurses among others. It is important to focus on this aspect when we consider a robot's deceptive capabilities in human-robot interaction to achieve socially-intelligent robot agent (Shim & Arkin, 2013).

2.3 Robot Deception

Endowing robots with the capacity for deception has significant potential utility (Wagner & Arkin, 2011), affording advantages comparable to its use of animals. Clearly, deception behaviors are useful in the military domain (Hawthorne, 2006; Meehan, 1988). Sun Tzu stated in *The Art of War*, "All warfare is based on deception". Military robots capable of deception could mislead opponents in a variety of ways. As both individual and robot teams become more prevalent in the military's future (Wilcox & Jackson, 1998), robotic deception can provide new advantages apart from the more traditional one of force multiplication. In other areas, such as search and rescue or battlefield triage, deceptive robots might also add value, for example, for calming victims or patients when required for their own protection.

Conceivably even in the field of educational robots, a deceptive robot teacher may potentially play a role in improving human learning efficiency (Shim & Arkin, 2012).

Despite its ubiquity in nature and its potential benefits, very few studies have been conducted on robot deception to date. Floreano's research group (Floreano, Mitri, Magnenat, & Keller, 2007) demonstrated robots that evolve deceptive strategies in an evolutionary manner, learning to protect energy sources. Their work illustrates the ties between biology, evolution, and signal communication and does so on a robotic platform. They showed that cooperative communication evolves when robot colonies consists of genetically similar individuals. In contrast, when the robot colonies were dissimilar the robots evolved deceptive communication signals.

Lisy et al. (Lisy, Zivan, Sycara, & Pechoucek, 2010) have studied deception by an adversary, when the mobile sensing agents are coped with each other. During the mission, opponents frequently use deception to distract the agents of sensor team. To increase the efficiency of the team, they analyzed and modeled the adversary's deception using game theoretic approaches, and demonstrated the success of their model via empirical study.

Wagner and Arkin (Wagner & Arkin, 2011) used interdependence theory and game theory to develop algorithms that allow a robot to determine both when and how it should deceive others. More recent work at Georgia Tech explores the role of deception according to Grafen's dishonesty model (Johnstone & Grafen, 1993) in the context of bird mobbing behavior (Davis & Arkin, 2012).

Terada and Ito (Terada & Ito, 2010) demonstrated that a robot is able to deceive a human by producing a deceptive behavior contrary to the human subject's prediction. These results illustrated that an unexpected change of a robot's behavior gave rise to an impression in the human of being deceived by the robot.

Other research shows that robot deception behavior can increase users' engagement in robotic game domains. Work at Yale University (Short, Hart, Vu, & Scassellati, 2010) illustrated increased engagement with a cheating robot in the context of a rock-paper-scissor game, demonstrating greater attribution of mental state to the robot by the human players when participants played against a cheating robot. At Carnegie Mellon University (Vazquez, May, Steinfeld, & Chen, 2011) a study showed an increase of user's engagement and enjoyment in a multi-player robotic game in the presence of a deceptive referee. By declaring false information to game players about how much players win or lose, they observed whether this behavior affects a human's general motivation and interest based on the frequency of winning, duration of playing, etc. These results indicate that deceptive behaviors are potentially beneficial not only in the military domain but also in an everyday context.

3. Deceptive Behaviors in Food Hoarding

The deceptive behavior of squirrels in terms of their food hoarding strategies is now addressed. Food hoarding is an important behavior for many animal species, such as birds and rodents, which are mainly comprised of two parts: caching and protecting the food. The deceptive component falls in the food protection phase.

3.1 Cache Formation

Food caching activity in animals ranges widely from highly dispersed (scatter hoards) to highly clumped (larder hoards). Scatter hoarders cache a few items in many small/scattered caches. On the other hand, larder hoarders place most of the food in one or a few central locations referred to as middens. The evolution of the particular hoarding strategy for a species depends on the abilities of individuals to defend their caches against pilfering (Gerhardt, 2005). According to observation, animals use a larder hoarding cache strategy when their competitors are conspecifics (same species) or they are weaker animals; however, when potential competitors are heterospecific or stronger adversaries, animals tend to use a scatter hoarding strategy (Gerhardt, 2005).

3.2 Cache Protection

After hoarding food items, animals begin to protect their resources from pilfering by patrolling the caches. First, animals move around the caching areas and check whether the cached food items are safe. However, animals generally change their behavior after they experienced pilfering.

One general food protection behavior of animals is changing the locations of its food items. According to Preston (Preston & Jacobs, 2001; Preston & Jacobs, 2005), after kangaroo rats experienced pilfering from conspecific or heterospecific competitors, they moved the location of their food items.

Of particular use for our research is an interesting deceptive behavior observed in the food protection strategy of certain squirrels (Steele, et al., 2008). Social context (i.e., presence or absence of competitors) appears to be pivotal to the expression of cache protection behaviors. Deceptive behavior in the tree squirrel has been observed with respect to food protection.

While patrolling, tree squirrels visit the cache locations and check on their food (Figure 1a). However, if potential competitors are detected nearby, the animals visit several empty cache locations. This deceptive behavior attempts to confuse competitors regarding the food's location, so they can protect against the loss of their hoarded food. After the potential competitors leave the territory, the tree squirrels move the location of their stored food items, if pilfering has occurred (Figure 1b).

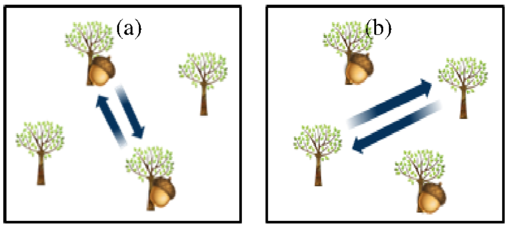


Figure 1. Cache Protection (a) True Patrolling and (b) Deceptive Patrolling

4. Computational Model and Implementation

A bio-inspired behavior-based model (Arkin, 1998) of squirrel caching and protecting behaviors for application to robotic systems is now presented. Simulation studies and real robot experiments were performed in *MissionLab*[1], a mission specification software package developed by the Mobile Robotics laboratory at Georgia Tech (MacKenzie, Arkin, & Cameron, 1997). *MissionLab* provides a graphical user interface that enables users to easily specify behavioral states and the control transitions between states, yielding a finite state acceptor (FSA), which can then be compiled down to executable code for both simulations and robots (MacKenzie & Arkin, 1998).

Each behavior component is an assemblage, a coordinated aggregation of primitive behaviors. The new caching and patrolling behaviors created are combined with pre-existing behaviors, such as avoiding obstacles, moving toward an object, or injecting randomness (noise).

The computational model determines how robots behave in resource caching and protecting scenarios as inspired by squirrel behaviors. The model consists of two main parts - caching behavior and

[1] *MissionLab* is freely available for research and educational purposes at: http://www.cc.gatech.edu/ai/robot-lab/research/*MissionLab*/

patrolling (protecting) behavior. The simulation and robot experiments are based on the interactions between two robotic agents: a "squirrel" robot (resource storer) and a competitor robot (resource pilferer).

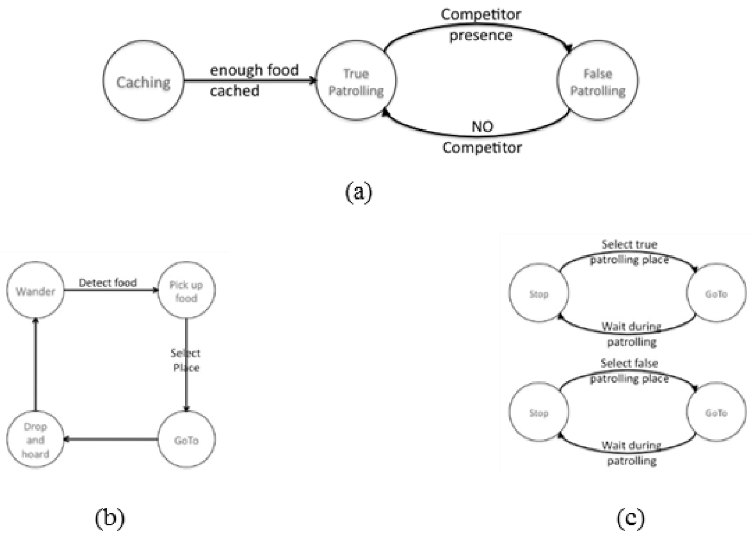


Figure 2. Abstract level of Finite State Acceptors for squirrel robot's behaviors: (a) High-level FSA: caching behaviors of squirrels, (b) sub-FSA: food hoarding, and (c) sub-FSA: food patrolling

Figure 2 illustrates the squirrel robot's storing and patrolling behaviors in an abstract level. Figure 2 (a) shows the high-level view of the squirrel robot's entire behavior strategies. It starts from the caching behavior, but if any of the caching location is enough by the food items, it transitions to the patrolling strategy. In the patrolling strategy, if the competitor robot is nearby, the squirrel robot performs the deceptive patrolling strategy. Otherwise, the true patrolling strategy is procedure. Figure 2(b) and 2(c) illustrate the caching and the true/deceptive patrolling sub-strategies briefly. In the following subsections, we will explain each strategy with more implementation details.

4.1 Caching Strategy

Many research groups (Bradski, 2000; Sugawara, Kazama, & Watanabe, 2004; Lerman & Galstyan, 2002), including ours (Balch & Arkin, 1994), have studied foraging behavior in robotics. In the caching simulation, one robot is required to store the scattered resources in safe locations. Figure 3(b), illustrates the high-level FSA model. To implement the caching behaviors, we used several robot behaviors: Wander, GoTo, DetectObject, PickUpObject, and DropObject. Detailed explanations of each behavioral state and trigger are shown in Appendix A.

With these states and triggers, the squirrel robot's caching strategy was implemented in simulation. Figure 3 illustrates *MissionLab* implementation of squirrel robot's caching behaviors. First, the robot wanders around searching for food items. When the robot detects a food item during foraging, it is picked up. Each food item has yellow mark; therefore, by detecting yellow objects, the pick up behavior is triggered. After picking up the item, the robot selects the caching location to store this item in based on a pre-defined probabilistic distribution. Probabilities of each caching location indicate how often the robot visits each particular location. If the probability of the location is higher than the others, it indicates that the robot will cache more food items in this location. When the simulation includes k caching locations, we set the probabilities $p_1, p_2, \ldots, p_k$ arbitrarily with the sum of p_1 to p_k equal to 1.

To select the next caching location for a newly acquired item according to this pre-defined probability distribution, a weighted roulette wheel algorithm is used (Holland, 1992). This probabilistic transition model between behavioral states has been previously used successfully in developing models of wolf pack predation in our laboratory (Madden, Arkin, & McNulty, 2010). After selecting the specific caching place out of several choices, the robot moves to the selected location and drops the item there. The robot repeats this strategy until the "enough food cached" (a predefined parameter) trigger is activated in any caching location.

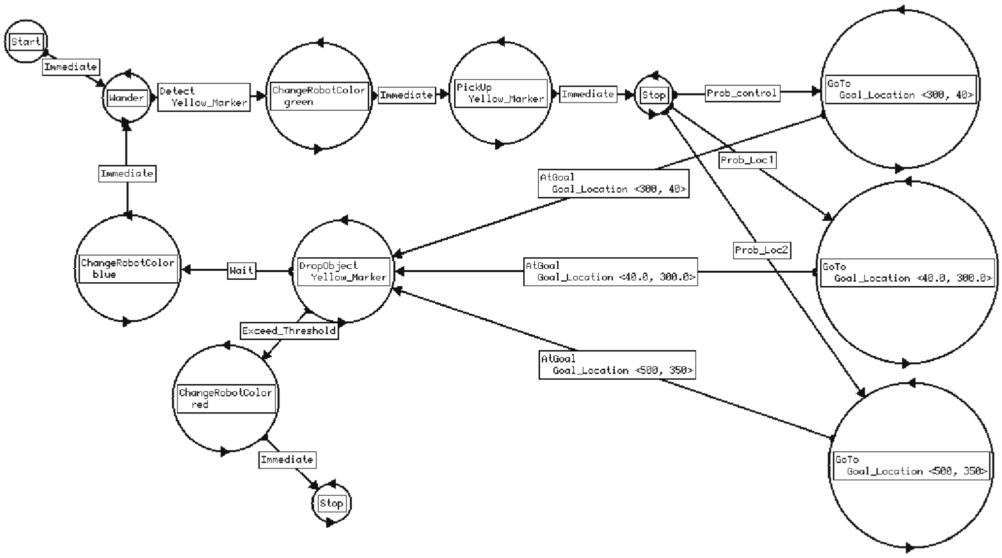


Figure 3. Finite State Acceptors for squirrel robot's food storing strategy in *MissionLab* (implementation for simulation studies)

4.2 Protection Strategy

4.2.1 True Patrolling Strategy

After caching is complete, the robot begins to patrol between the caching locations to protect the resources. The behaviors of the robot include goal-oriented movement, selecting places, and waiting behavior (Figure 3c).

Initially, the robot employs the true patrolling strategy by selecting one of the true patrolling locations. To select the patrolling location, the trigger calculates which of the many caching locations the robot should patrol. The calculation results in a random cache selection based on the transition probabilities among the places. Again, we used the probabilistic transition model as in the caching behavior. In patrolling, the transition probabilities are first determined by the number of previously cached items. If a place has more items, the probability that a robot will visit that location is higher. Therefore, the transition probabilities are calculated by the following equation:

$$P_{ij} = \frac{\# items_j}{\sum\limits_{1 \le k \le n, k \ne j} \# items_k}$$

Here, P_{ij} is the transition probability that indicates that the location j is selected as the next patrol location when the current location is in location i. n is the total number of locations and $\#item_x$ indicates the number of food items in location x. The next patrol state is determined based on these transition probabilities. When the squirrel robot reaches a cache, it calculates the transition probabilities to other patrolling locations and decides on one of the transitions using a weighted roulette algorithm (Holland, 1992; details can be found in Appendix B). When a robot arrives the cache, it remains there for a finite amount of time similar to the patrolling behavior of an actual squirrel. In the true patrolling strategy, the time spent at the cache is determined by the number of food items in that place. If a place contains n food items, the robot stays there for n seconds. At the end of the waiting phase, the robot selects the next patrolling location using the probability transition model discussed above and heads off to the next patrolling state.

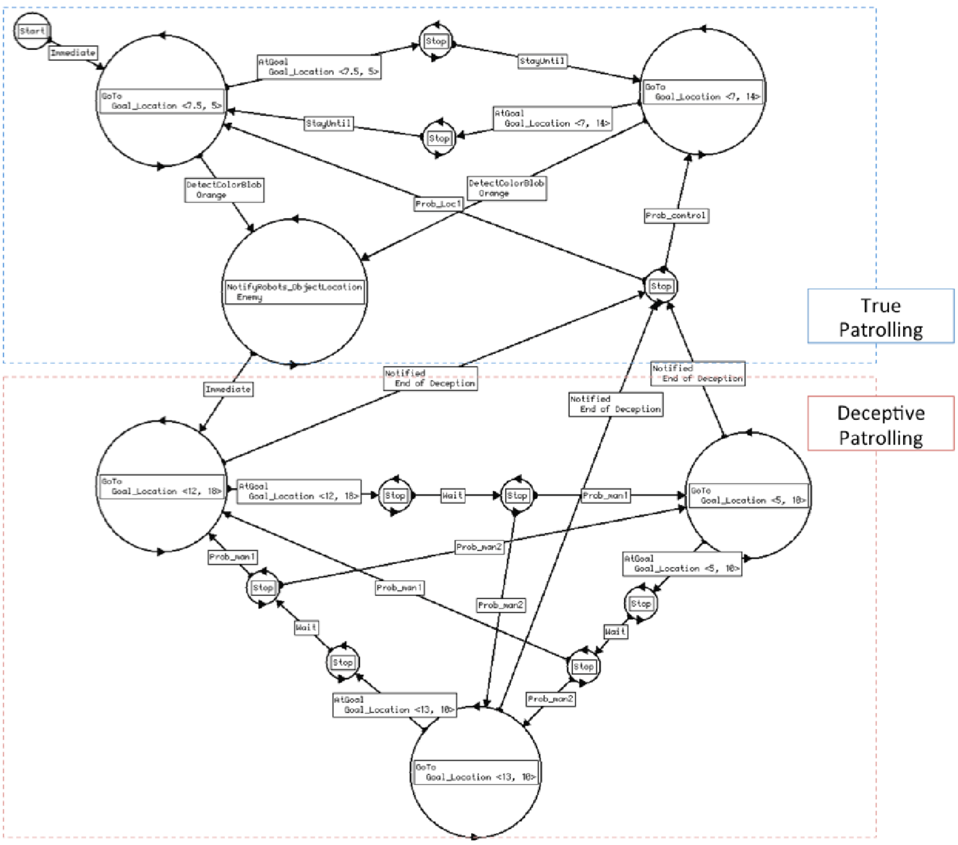


Figure 4. Finite State Acceptors for squirrel robot's patrolling strategy in *MissionLab* (implementation for real robot experiments)

Figure 4 illustrates the *MissionLab* FSA implementation of a squirrel robot's patrolling strategy. This FSA was used for the real robot experiments. Due to the limitations of lab space, we used two true caching locations and three deceptive caching locations. Therefore, as shown in figure 5, in the true patrolling part, the robot moves back and forth between two locations, and, as a result, we don't have to use the probabilistic transition model here. When the robot detects the competitor robot, which has an orange marking, it transitions to the deceptive patrolling strategy. Here, we set three deceptive locations, and the transitions among these three locations are determined by the transition probabilities model that were explained above. Further details of deceptive patrolling strategy will be discussed in the next section.

4.2.2 Deceptive Patrolling Strategy

When the squirrel robot detects the presence of a competitor, deceptive behavior is triggered and the squirrel robot patrols the false (empty) caching locations to attempt to deceive the competitor. All objects and robot agents are marked by specific colors. Therefore, in our implementation, the deceptive patrolling strategy is activated by the DetectColorBlob trigger. In the deceptive patrolling strategy, the squirrel robot moves to and stays among the different deceptive caching locations. These locations actually include no food items, and the squirrel robot tries to mislead the competitor robot by visiting these false places. Again, the selection of deceptive locations is also calculated by transition probabilities. Here, the transition probabilities among the false locations are set as uniform distributions. In other words, the probabilities of each location are distributed equally.

As shown in Figure 4's deceptive patrolling part, the squirrel robot selects one deceptive caching location among several places based on transition probabilities. When the robot arrives in the deceptive caching location, it stays there for a while (time to stay is empirically set) to show the deceptive patrolling behavior to the competitor. After patrolling, it again determines the next deceptive patrolling location and repeats the patrolling behaviors. When the competitor robot is no longer detected in the vision of squirrel robot, the end of deception trigger is activated and it returns to the true patrolling strategy.

4.3 Competitor Robot Behavior

A competitor robot has a simple strategy in the current scenario (Figure 5), where it wanders around the map to try and find the squirrel robot. When it detects the quarry, it determines whether it is at a potential caching location or not, by observing how long the squirrel robot stays in place. Since the squirrel robot takes time to patrol the caching place proportional to the number of food items, the competitor robot obtains evidence of the caching area based on the robot's time onsite. Therefore, if the duration is over a threshold, set empirically, the competitor robot recognizes the place as a caching area. The competitor robot then goes to the detected location and pilfers. In our robot implementation, in the pilferage step, the competitor robot confirms whether the detected location truly contains the items by discriminating food items based on the colors. If it confirms the caching location, it sends the alert message to the system that it has found the cached item. If it determines the location doesn't include the caching item, it returns to "wander" state and repeats the detecting process again. Figure 5 shows the FSA of the competitor robot in the real robot experiment. As shown in this figure, the robot used GoTo, Wander, MoveTowardObject, Detect, and Alert behaviors for the competitor robot's strategy.

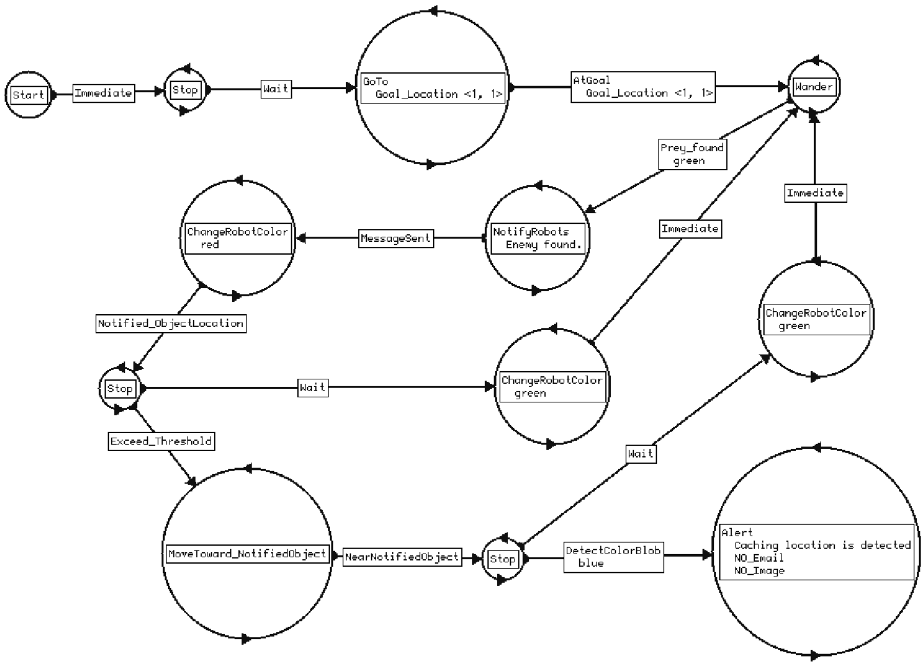


Figure 5. Finite State Acceptors for competitor robot's hunting strategy in MissionLab (implementation for real robot experiments)

5. Experimental Results[2]

To evaluate our deception algorithm, several experiments were performed both in simulation and with real robot systems. For the simulation results, the algorithm was implemented in *MissionLab* as shown in Section 4 and the performance of the algorithm evaluated by comparing the competitor robot's cache pilferage times with and without the squirrel robot's deception capabilities. Real robot experiments were also performed. First, we tested whether the algorithm is successfully applied to the robot system in a simple scenario. In addition, we evaluated the performance of robot's deception capabilities in the real world system by observing whether and how often the competitor robot is deceived by the squirrel robot.

5.1 Simulation Results

A simple scenario of the squirrel-like deceptive behavior was simulated in *MissionLab*. The simulation environment is shown in Figure 6. Yellow-colored food items were randomly placed around the map. In this simulation, the robot detects these food items by discriminating colors. Three caching places and three empty places were chosen arbitrarily.

First, the robot finds a food item and stores it in the pre-defined caching places as shown in Figure 6a. When the number of the cached items is over a threshold for any of the caches, the state of the robot switches to cache protection. If a competitor is not present, it patrols the true caching locations (Figure

[2] Experiment videos are available at: http://www.cc.gatech.edu/ai/robot-lab/hunt/squirrelProject.html

6b) Otherwise, the deceptive patrolling strategy is activated, and the robot moves to the empty caching places (Figure 6c).

To evaluate the approach, the performance was evaluated by measuring the time duration until the competitor robot detects the true caching places and begins pilfering. The same scenarios without deceptive behaviors formed the baseline. Comparing the baseline results to the measured time when deception is active, serves as an evaluation of its effectiveness. The simulation was run 30 times per each condition: with and without deceptive behaviors.

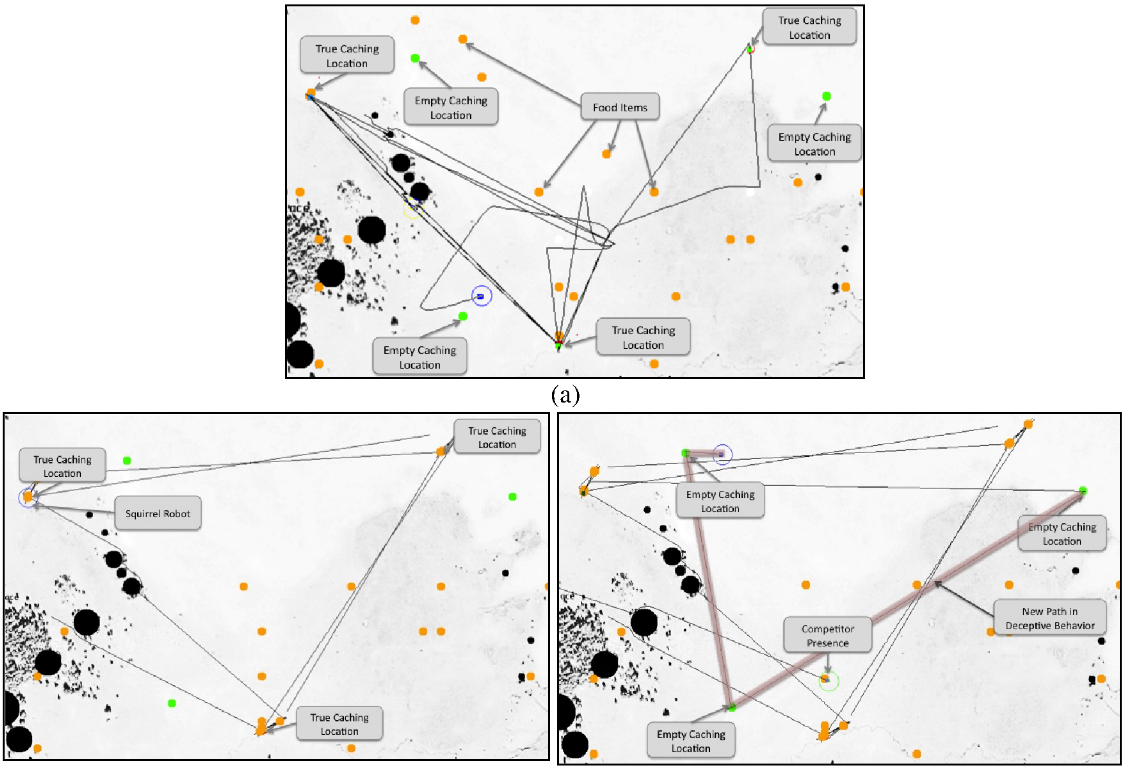


Figure 6. Simulation Results. (a) Caching, (b) True patrolling, (c) Deceptive patrolling strategies

In each trial, all the other conditions except deceptive behaviors are the same. Even though the number of cached items varies in each trial, it maintains the same two conditions - with and without deceptive behaviors. Table 2 and Figure 7 show the simulation results. The average time to successful pilferage when the squirrel robot includes deceptive behavior is 10.4 minutes (std: 3.04), compared to the average time duration without deception is 7.69 minutes (std: 2.91). The statistical analysis yielded 0.0009 p-value (< 0.05) with the Student's t-test, a significant difference between the results of the two conditions.

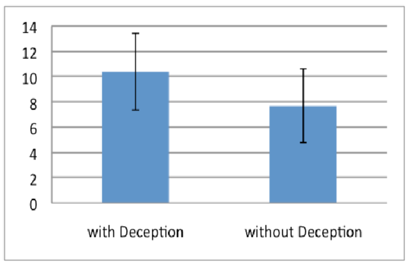


Figure 7. Average time to pilferage

Table 2. Simulation results of first ten of 30 trials: time duration until competitor successfully pilferages resources in contexts; (a) with deceptive behaviors and (b) without deceptive behaviors. (Measurements given in minutes).

Deception \ Traits	1	2	3	4	5	6	7	8	9	10
(a) With	8.76	12.73	5.92	9.25	12.33	10.24	10.97	7.8	15	11.79
(b) W/O	6.79	7.80	10.82	3.13	5.42	11.02	12.03	6.08	5.83	8.48

5.2 Robot Experiment Results

From the simulation results, we proved that our algorithm for robot deception is beneficial in resource protection situations. To evaluate the algorithm further, we also ran real robot experiments. This experiments consisted of two phases. In the first phase we aimed to evaluate whether the squirrel robot's deceptive behavior was properly applied to the real robot system. In the second phase, we added a more complex competitor robot to the experiment and observed whether the competitor robot is truly deceived by the squirrel robot's deception behavior.

We used two pioneer robots for the robot experiment: one as a squirrel robot and the other played the role of a competitor robot. The squirrel robot used an additional camera sensor for detecting the competitor robot and the caching locations (Figure 8). The external omni-directional camera enabled the squirrel robot to observe 360° views of the scene. We used Kumotek Robotics' omni-directional sensor with a Chameleon CCD camera[3].

The experimental environment was set up as shown in Figure 8. The blue baskets indicate the true caching locations for the resources, and the yellow cones represent the empty caching locations. Both robot experiments contain simple scenarios for testing the deceptive cache protection strategies, omitting the caching strategies. In other words, we assume that resources are already cached in the two true caching locations (positions of blue baskets).

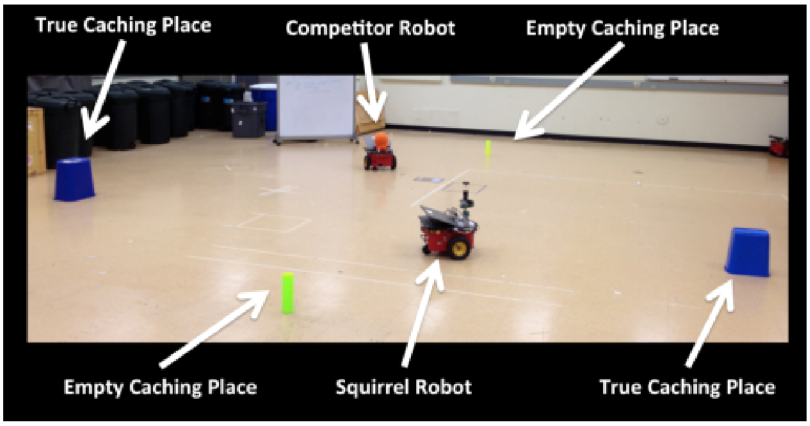


Figure 8. Robot Experiment Layout

[3] Omni-directional Sensor: Kumotek's VS-C450MR-TK standard model http://www.kumotek.com/
Chameleon CCD camera from Point Grey: http://www.ptgrey.com/products/chameleon/

5.2.1 Robot Experiment Phase 1

In phase 1, we intended to test whether our deception algorithm was successfully applied to the real robot system. During this experiment, the squirrel robot patrolled true caching locations to observe the cached items. Since there were two caching locations in this simple scenario, the squirrel robot conducted back-and-forth movements in this states (Figure 9a-c). During patrolling, the competitor robot was teleoperated to approach the squirrel robot. An orange-colored ball was mounted on the competitor robot, since the squirrel robot discriminated objects based on colors. According to the algorithm, the squirrel robot should change its behavior to deceptive patrolling when it detects the competitor robot.

As figure 9 shows, we observed that the algorithm is successfully working in this simple experimental scenario. At the end of scenario, the squirrel robot detected the competitor (Figure 9d-f), where it changed it true patrolling to deceptive patrolling. It started to move to the empty caching locations, and repeated these behaviors until the competitor robots left the area (Figure 9g-i).

(a)
GoTo Cache 1

(b)
Patrol Cache 1

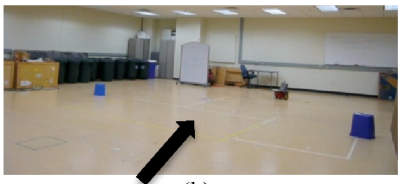
(c)
GoTo Cache 2

(d)
GoTo Cache 1 & Competitor Approaching

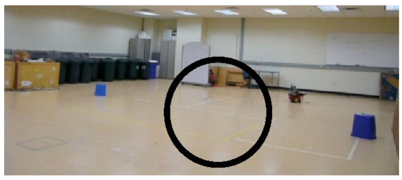
(e)
Detect competitor

(f)
Change Behavior

(g)
Start Deceptive Behavior



(h)
GoTo Empty Cache 1



(i)
Patrol Empty Cache 1

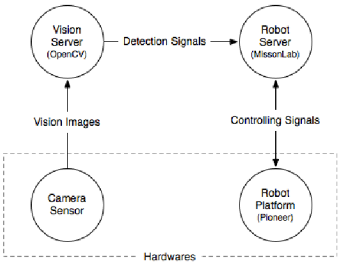
Figure 9. Robot Experiment Scenario:
(a)~(c) True Patrolling, (d)~(f) Competitor Detecting, (g)~(i) Deceptive Patrolling strategies

5.2.1 Robot Experiment Phase 2

In phase 1 we confirmed that our deception algorithm is working properly using teleoperation of the competitor. However, the more important evaluation is whether the algorithm is effective in deceiving other robots, as shown in simulation, i.e., we have to test not just the algorithm, but the performance of this deceptive capability.

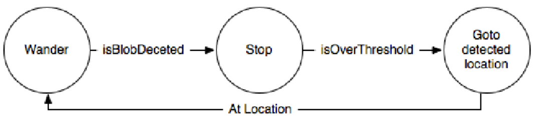
Now, it was required to design the competitor robot's general behavior for pilfering. In the first test phase, the experimenter teleoperated the competitor robot as it only needed to approach the squirrel robot. For realizing an autonomous (not teleoperated) competitor robot, we designed and implemented a new FSA and behaviors.

Figure 10b illustrates the behaviors of the competitor robot in these robot experiments. As shown here, the competitor robot wanders around the lab environment until it detects the squirrel robot. When it detects the squirrel robot, it observes the squirrel robot's location and determines whether the squirrel robot stays at a certain position for a while. If the squirrel robot's location does not change during the pre-defined time period, the competitor robot determines the squirrel robot is currently patrolling a cache, and proceeds to that location. This pre-defined time period is empirically determined and set by experimenter. If the competitor determines the other robot is not patrolling, the competitor robot goes back to the wander behavior and repeats the above behaviors.



Detection Signals from Vision Server	
Boolean isBlobDetected	true if the pre-defined blob is detected
Boolean isOverThreshold	true if the blob stays in one location over threshold time
Float[] blobLocations	Blob location (x,y,z)

(a) Robot System Architecture and Signals



(b) FSA for the Competitor Robot

Figure 10. Competitor robot system designs for the robot experiment phase 2

Figure 11. Competitor robot detects the squirrel robot and approaches

Object	Assigned Color
Competitor Robot	Orange
Squirrel Robot	Light Green
Caching Location	Blue

Table 3. Pre-defined Color Data

For this experiment, the competitor robot also used the external camera as a vision sensor. As shown in Figure 10a, this external camera enabled the competitor robot detected the squirrel robot by sending detection signals. The competitor robot's vision system discriminates the squirrel robot using pre-defined colors (Table 3). We assigned orange to the competitor robot and light green to the squirrel robot, by placing green cones on the squirrel robot and an orange ball on the competitor robot (Figure 11).

The vision server continuously receives images from the camera sensor, discriminating the blobs and determining their locations using the OpenCV color-blob detection algorithm (Bradski, 2000). When the vision server detects a blob (isBlobDetected = true) and determines the location, it compares the current location to the detected blob's location and determines how long this blob stays in the same location. If the time to stay in the same location is larger than the pre-defined threshold, the server determines that the squirrel robot is patrolling and sends this information to the competitor robot's main server (isOverThreshold = true). In our test, we set this pre-defined threshold as three seconds empirically. The environmental settings and the behaviors of the squirrel robots were the same as those for the first phase.

To test the performance of the deception algorithm, we measured how often the squirrel robot successfully deceived the competitor robot in this scenario. In the experiment, the squirrel robot and the competitor robot ran based on their FSA and behaviors as explained above. The competitor robot was declared to have successfully pilfered the cached items if it found the true caching locations within the specific time period, t. This experiment time t is one of the independent variables and it ranged from one to ten minutes. In each trial, we ran the experiment five times and observed how many times the competitor robot successfully pilfers the true locations for all 5 runs. Thus, the maximum successful pilfering can be five in each case. The experiments were performed under two different conditions; the squirrel robot with deceptive capabilities and the squirrel robot without deceptive capabilities (another independent variable). The results are shown in Table 4.

deception \ min	1	2	3	4	5	6	7	8	9	10
(a) With	0	2	2	3	5	5	5	5	5	5
(b) Without	0	2	5	5	5	5	5	5	5	5

Table 4. Robot experimental results: number of successful pilferages (out of five) in each time period (a) with deceptive behaviors and (b) without deceptive behaviors

As the time allowed increases, the number of successful pilferages also increases and converges to the maximum pilferage number, five. Based on this convergence rate, the performance of the algorithm can be evaluated. Faster convergence to the maximum pilferage number indicates that the algorithm enables the competitor robot to find the true caching locations more easily and more rapidly. In other words, slower convergence rates illustrate that the squirrel robot can protect resources longer and better.

To determine the convergence rate, we plotted the experimental results as shown in Table 5. By observing the plot, we formulated the estimation graph of each result with the following equation, $y = \alpha + \beta \cdot e^{cx}$. This function is calculated using a non-linear least-squared regression method.

The graphs in table 5 show the results from the experimental data and their estimation functions. Column (a) includes the experimental results when the squirrel robot uses deception. In contrast, column (b) illustrates the results without deceptive behaviors. The green lines show the original experimental results, which are the number of successful pilferages (out of five) for each time period, t. The red lines indicate the regression functions for convergence. The figure also includes the exponential equations and the computed convergence rate.

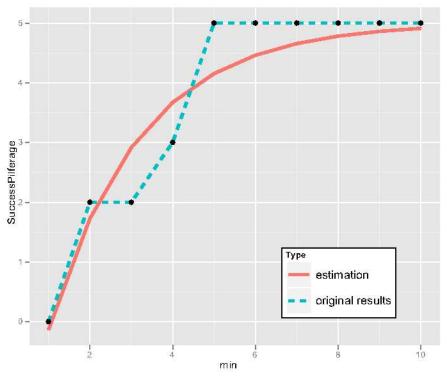
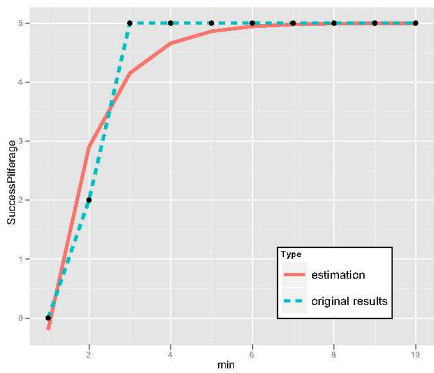
	(a) With Deception	(b) Without Deception
Results		
Est. Eq.	y = 5 – 12.86 exp(-0.9x)	y = 5 – 8.06 exp(-0.45x)
Conv. rate	-2.2188	-1.1035

Table 5. Experimental Results Analysis: the graphs, estimation functions, and convergence rates of the results

In the exponential function, $y = \alpha + \beta \cdot e^{cx}$, the convergence rate depends on the exponentiation, parameter c. Simply, 1/c can determine the rate of convergence and a larger value of the convergence rate indicates a faster convergence speed. As shown in the results from Table 5, the convergence rate of the "with deception" condition is -2.2188, which is smaller than the convergence rate under the "without deception" condition (-1.1036). Thus, it was observed that the squirrel robot using deception could protect the true caching locations longer than without deception capabilities. Therefore, we conclude that our deception algorithm is successfully working in the real robot systems.

6. Conclusions and Future Work

In this paper, a novel approach was presented for deception in robots, focusing on how to preserve resource gains. This approach was inspired from biological findings, i.e., deceptive behaviors of eastern grey squirrels during cache protection. Computational algorithms were developed applying these deceptive behaviors to robots. In the evaluation phase, several simulations were run and it was found that the deceptive behaviors worked effectively and enabled robots to perform better with than without deception. In addition, robot experiments were performed to evaluate whether the algorithm is truly applicable to the real robot systems with two pioneer robots. The experiments consisted of two parts: 1) applying the algorithm to the robot system and 2) testing the performance of the algorithm with the real robots. From the results, we could conclude that the algorithm is successfully applied to the robots and the robot's deception capability works properly to protect the resources in the real robot experiments.

The current version of our algorithm only addresses a scenario with one deceiver robot and one competitor robot (which is a realistic case). However, to be more general, it should include multiple autonomous agents. In the foraging strategy, robots may need to determine the probabilistic distribution for storage locations based on their safety with respect to resource protection instead of a pre-defined distribution. This remains for future work.

As this research focuses on deceptive behaviors for robots in the military domain, where robots may hide and protect resources from humans or other autonomous agents, this deceptive behavior can be beneficial. We will potentially extend our research more towards other more human-friendly environments. In this respect, we are exploring robot deception in Human-Robot Interaction (HRI) contexts. If a robot acts deceptively in HRI situations, can the deceived human partners obtain an advantage, and not just the deceiver? The main hypothesis is whether the robot deception can benefit the deceived human partners in Human-Robot Interaction (HRI) contexts. We have already published preliminary research in this area (Shim & Arkin, 2013).

Providing deceptive behaviors to robots leads to ethical questions, such as whether it is ethical for robots to deceive humans for any purpose. We have addressed this topic elsewhere (Arkin, 2011). This requires considerable discussion in a broader community, which we actively encourage.

Acknowledgement

This work was supported in part by the Office of Naval Research under MURI Grant # N00014-08-1-0696.

References

Ariely, D. (2012). *The (honest) truth about dishonesty*. HarperCollins Publishers.

Arkin, R. (1998). *Behavior-based Robotics*. MIT Press.

Arkin, R. (2011). The Ethics of Robotic Deception. *1st International Conference of the International Assoc. for Computing and Philosophy*.

Balch, T., & Arkin, R. (1994). Communication in Reactive Multiagent Robotic Systems. *Autonomous Robots , 1* (1), 27-52.

Barnes, J. A. (1994). *A pack of lies: Towards a sociology of lying*. . Cambridge University press.

Baron-Cohen, S. (2007). I cannot tell a lie - what people with autism can tell us about honesty. *In Character-a journal of everyday virtues* .

Baron-Cohen, S. (1995). *Mindblindness: an essay on autism and theory of mind*. MIT Press/Bradford Books.

Bond, C., & Robinson, M. (1988). The evolution of deception. *Journal of nonverbal behavior , 12* (4), 295-307.

Bradski, G. (2000). The openCV library. *Dr. Dobb's Journal of Software Tools* .

Brault, S., Bideau, B., Craig, C., & Kkulpa, R. (2010). Balancing deceit and disguise: How to successfully fool the defender in 1 vs. 1 situation in rugby. *Human movement science , 29* (3), 412-25.

Cheney, D. L., & Seyfarth, R. M. (2008). *Baboon metaphysics: The evolution of a social mind*. Chicago: University of Chicago press .

Davis, J., & Arkin, R. (2012). Mobbing Behavior and Deceit and its role in Bio-inspired Autonomous Robotic Agents. *International Conference on Swarm Intelligence. 7461*, pp. 276-283. Brussels, Belgium: Springer Berlin Heidelberg.

de Waal, F. B. (1992). Intentional Deception in Primates. *Evolutionary Anthropology , 1* (3), 86–92.

Dessing, J., & Craig, C. (2010). Bending it like beckham: How to visually fool the goalkeeper. *PloS One , 5* (10), e13161.

Floreano, D., Mitri, S., Magnenat, S., & Keller., L. (2007). Evolutionary Conditions for the Emergence of Communication in Robots. *Current Biology , 17*, 514-519.

Gerhardt, F. (2005). Food Pilfering in Larder-Hoarding Red Squirrels (Tamiasciurus Hudsonicus). *Journal of Mammalogy , 86* (1), 108-114.

Gerwehr, S., & Russell, G. W. (2003). *Unweaving the Web: Deception and Adaptation in Future Urban Operations*. RAND Corporation.

Gouzoules, H., & Gouzoules., S. (2002). Primate Communication: By nature honest, or by experience wise? *International Journal of Primatology , 23* (4), 821-848.

Hawthorne, L. (2006). *Military Deception*. Joint Publication.

Holland, J. H. (1992). *Adaptation in Natural and Artificial Systems* (2nd Edition ed.). MIT Press.

Jackson, R., & Warren, S. (2006). Anticipation skill and susceptibility to deceptive movement. *Acta Psychologica , 123* (3), 355–371.

Jenkins, S. H., Rothstein, A., & Green, W. C. (1995). Food Hoarding by Merriam's Kangaroo Rats: A Test of Alternative Hypotheses. *Ecological Society of America , 76* (8), 2470-2481.

Johnstone, R., & Grafen, A. (1993). Dishonesty and the Handicap Principle. *Anim. Behav , 46* (4), 759–764.

Joshi, S. S., Johnson, R., Rundus, A., Clar, R. W., Barbour, M., & Owings, D. H. (2011). Robotic Squirrel Models. *IEEE Robotics and Automation Magazine , 18* (4), 59 - 68.

Lerman, K., & Galstyan, A. (2002). Mathematical Model of Foraging in a Group of Robots: Effect of Interference. *Autonomous Robots , 13* (2), 127-141.

Lewis, M., & Saarni, C. (1993). *Lying and deception in everyday life*. The Guilford press.

Li, S., & Guo, Y. (2012). Distributed source seeking by cooperative robots: All-to-all and limited communications. *IEEE International Conference on Robotics and Automation* .

Lisy, V., Zivan, R., Sycara, K., & Pechoucek, M. (2010). Deception in Networks of Mobile Sensing Agents. *Proceedings of the 2010 Conference on Autonomous Agents and Multi-Agent Systems*. Toronto, CA.

M. Vazquez, e. a. (2011). A deceptive robot referee in a multiplayer gaming environment. . *In Collaboration Technologies and Systems (CTS) International Conference*.

MacKenzie, D., & Arkin, R. C. (1998). Evaluating the Usability of Robot Programming Toolsets. *International Journal of Robotics Research , 4* (7), 381-401.

MacKenzie, D., Arkin, R., & Cameron, J. (1997). Multiagent Mission Specification and Execution. *Autonomous Robotics , 4* (1), 29-57.

Madden, J., Arkin, R., & McNulty, D. (2010). Multi-robot System Based on Model of Wolf Hunting Behavior to Emulate Wolf and Elk Interactions. *Proc. IEEE International Conference on Robotics and Biomimetics*, (pp. 1043 - 1050). Tianjin, China.

Matsuzoe, S., & Tanaka., F. (2012). How smartly should robots behave?: Comparative investigation on the learning ability of a care-receiving robot. *Proceedings of the 21st IEEE International Symposium on Robot and Human Interactive Communication*, (pp. 339 - 344). Paris.

Mawby, R., & Mitchell, R. W. (1986). *Feints and rues: an analysis of deception in sports, Deception: perspectives on human and nonhuman deceit*. . SUNY press.

Meehan, W. J. (1988). *FM 90-2 Battlefield Deception*. Army Field Manuals .

Preston, S. D., & Jacobs, L. F. (2005). Cache decision making: the effects of competition on cache decisions in Merriam's kangaroo rat. *Journal of comparative psychology , 119* (2), 187-96.

Preston, S. D., & Jacobs, L. F. (2001). Conspecific pilferage but not presence affects Merriam's Kangaroo rat cache strategy. *Behavioral Ecology , 12* (5), 512-523.

Ristau, C. (1991). Aspects of the cognitive ethology of an injury-feigning bird, the piping Plover. In C. A. Ristau, *Cognitive Ethology: The Minds of Other Animals* (pp. 91-126). Hillside, NJ: Erlbaum.

Sexton, D. J. (1986). *The theory and psychology of military deception*. SUNY press.

Shim, J., & Arkin, R. C. (2013). A taxonomy of robot deception and its benefits in HRI. *Proc. IEEE Systems, Man and Cybernetics Conference*. Manchester, England.

Shim, J., & Arkin., R. C. (2012). Biologically-inspired deceptive behavior for a robot. *12th International Conference on Simulation of Adaptive Behavior (SAB2012)*, (pp. 401-411). Odense, DK.

Short, E., J. Hart, M. V., & Scassellati., B. (2010). No fair!!: an interaction with a cheating robot. Proc. *5th ACM/IEEE international conference on Human-robot Interaction*, (pp. 219 - 226). Osaka, Japan.

Steele, M. A., Halkin, S. L., Smallwood, P. D., McKenna, T. J., Mitsopoulos, K., & Beam, M. (2008). Cache protection strategies of a scatter-hoarding rodent: do tree squirrels engage in behavioural deception? *Animal Behaviour , 75* (2), 705-714.

Sugawara, K., Kazama, T., & Watanabe, T. (2004). Foraging behavior of interacting robots with virtual pheromone. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, *3*, pp. 3074 - 3079. Sendai. Japan.

Terada, K., & Ito, A. (2010). Can a Robot Deceive Humans? *Proceeding of the ACM/IEEE International conference on Human-robot interaction*, (pp. 191 - 192). Osaka, Japan.

U.S. Department of Defense. FY 2009-2034 Unmanned Systems Integrated Roadmap.

Vasek, M. E. (1984). *Lying: The Development of Children's Understanding of Deception*. Clark University.

Vazquez, M., May, A., Steinfeld, A., & Chen, W.-H. (2011). A deceptive robot referee in a multiplayer gaming environment. *Collaboration Technologies and Systems (CTS), 2011 International Conference on* (pp. 204 - 211). Philadelphia, PA: IEEE.

Vignais, N., Kulpa, R., Craig, C., Brault, S., Multon, F., & Bideau, B. (2010). Influence of the graphical levels of detail of a virtual thrower and the perception of the movement. *Teleoperators and Virtual Environments , 19* (3), 243-252.

Vrij, A. (2001). *Detecting lies and deceit: the psychology of lying and the implications for professional practice*. New York: John Wiley & Sons.

Wagner, A., & Arkin, R. (2011). Acting deceptively: Providing robots with the capacity for deception. *International Journal of Social Robotics , 3* (1), 5-26.

Wikipedia. (n.d.). Retrieved from http://en.wikipedia.org/wiki/File:Eastern_Grey_Squirrel.jpg

Wilcox, R. S., & Jackson, R. (1998). Spider-Eating Spiders. *American Scientist , 86* (4), 350.

Appendix A: Essential States and Triggers in MissionLab

State/Trigger	Descriptions
GoTo (parameter: location x, y)	The robot is move to the parameterized location (x,y).
Wander	The robot is wandering around the specified map.
State and Triggers for Squirrel Robot	
DetectColorBlob (parameter: color)	This trigger is activated when the vision server detects the parameterized color (i.e., isBlobDetected() in the vision server is activated).
StayUntil	The robot stop until *n* seconds, where *n* is determined by the number of cached items.
Prob_con	The robot determines the distribution of transition probabilities of the current locations and it is probabilistically likely to select one of the location among current places.
Prob_loc	The robot is probabilistically likely to select one of the location among current places.
State and Triggers for Competitor Robot	
Prey_found (parameter: color)	This trigger is activated when the squirrel robot is detected from the vision server. This trigger is the modification from DetectColorBlob trigger.
Notified_ObjectLocation	When the competitor robot receives the location information of the detected prey from the vision server, it processes to the next step.
Exceed_Threshold (parameter: time)	If the maintaining time of the notified object is over the parameterized time (i.e., isOverThreshold() in the vision server is activated), this trigger is activated.

Appendix B: Probabilistic Transition

Transitions based on the existence of probabilities to simulate environmental properties such as number of cached items.

First, P_{ij} is the transition probability that indicates the location j is selected as the next patrol location when the current location is in location i. In addition, n is the total number of locations and $\#item_x$ indicates the number of food items in location x.

$$P_{ij} = \frac{\#\,items_j}{\sum_{1 \le k \le n, k \ne j} \#\,items_k} \qquad \text{- (1)}$$

Based on the determined probabilities, a weighted roulette wheel algorithm is applied to decide the next transition. Using the transition probabilities, the proportion of the wheel is assigned to each of the possible selections. Then, if the randomly generated number is fitted to one of the proportion, it decides as the next transition:

R = random number between 0 and 1

$$NextLocation_i = \begin{cases} Location_1 & , \quad R < P_{i1} \\ Location_2 & , \quad R < P_{i1} + P_{i2} \\ \vdots & \end{cases} \qquad \text{- (2)}$$

Appendix C: Parameters used in the real robot experiments

Squirrel Robot's parameter setting in MissionLab	
Move_to_location gain	0.9
Wander_gain	0.0
Avoid_obstacle_gain	0.9
Avoid_obstacle_sphere	0.7
Avoid_obstacle_safety_margin	0.2
Max_velocity	1.5
Base_velocity	1.5

Competitor Robot's parameter setting in MissionLab	
Move_to_location gain	0.9
Wander_gain	0.5
Avoid_obstacle_gain	0.9
Avoid_obstacle_sphere	0.7
Avoid_obstacle_safety_margin	0.2
Max_velocity	1.0
Base_velocity	1.0