A Task Domain for Combining and Evaluating Robotics and Cognitive Modeling Techniques

Nicholas L. Cassimatis, J. Gregory Trafton,
Alan C. Schultz, Magdalena D. Bugajska, and William Adams
Naval Research Laboratory, Code 5515 and 5513
4555 Overlook Ave. SW
Washington, DC 20375-5337

ABSTRACT

Building systems that integrate different artificial intelligence techniques to achieve a higher level of total intelligence is very difficult. In order to build integrated systems, simplifying assumptions or abstractions are usually made when working in a specific domain. As a result of these assumptions and abstractions, the proper evaluation of integrated artificial intelligence techniques can be quite challenging. We suggest that the domain of hide and seek is a particularly well-suited task for integrating robotics and higher-level reasoning mechanisms such as computational cognitive modeling. Three different instantiations of integrated systems in the "hide and seek" domain, which combine cognitive-level algorithms with lower level algorithms for perception and navigation are discussed.

KEYWORDS: *Robotics, Cognitive Architectures, Cognitive Modeling, Performance Metrics*

1. INTRODUCTION

How do we build intelligent systems, and evaluate the underlying algorithms? There are, generally, two main ways of building intelligent systems. The first possibility is to focus on a relatively small sub domain and build a system or set of algorithms that solve problems in that sub domain very well. Working in this manner generally leads to very efficient methods of solving a relatively narrow set of tasks. There are many researchers (including some of the current authors [1]) who build these kinds of highly specific systems. The second possibility is to build complex systems that solve a larger class of problems but that may be less efficient at each task and perhaps at the whole task. There are some researchers working on these kinds of problems (e.g. [10]), but building integrated systems is very difficult for a number of reasons.

First, the individual techniques are developed using different assumptions about their use (e.g., the input/output relationship with the environment they are in). Second, because most systems are designed for different domains, combining two techniques often involves modifying and adding new domain specific elements to the design. Finally, because each research group's assumptions and domain are so unique, each new group or project must reinvent machinery that is relatively incidental to their main interest. For example, a researcher trying to develop probabilistic reasoning techniques to aid in robot navigation must spend considerable effort acquiring and configuring a robotics platform with the appropriate sensors and actuators to test possible new techniques.

Each of the above integration issues makes it difficult to evaluate the effectiveness of any new technique or system. It is almost impossible to compare two techniques when one assumes video input from the environment and the other assumes sonar; or when one is designed for an office navigation task and the other for air traffic control.

Finally, the structural difficulties of integrating various techniques may encourage researchers to ignore or abstract away from difficult issues that hold back the field's progress. For example, there is concern [5] that when people working on "high-level" artificial intelligence techniques abstract away from perception and mobility issues, or when people working on perceptual and mobility techniques ignore high-level inference, they are actually ignoring the true substance of intelligence which lies at the interface between the two.

In Section 2, we present the task of hide and seek as a particularly well-suited task for integrating and evaluating artificial intelligence techniques. Section 3 describes three different instantiations of integrated systems, which combine cognitive-level algorithms with lower level algorithms for perception and navigation, and which use the hide and seek domain.

2. THE HIDE AND SEEK DOMAIN

In order to address these evaluation and integration issues, we are organizing a substantial amount of our research around the "hide-and-seek" task domain. This domain is forcing us to face the difficult integration problems between "high-level" cognitive architectures (for example, ACT-R [2] and Polyscheme [7]) and systems (such as SAMUEL [9]) for sensing and moving in a physical environment. In this section, we describe how using a robotic platform in this domain allows us to study a surprisingly wide range of issues in intelligence, perception and mobility.

2.1. *Perception and Mobility*

Agents that engage in hide-and-seek obviously cannot avoid the need to address a wide range of problems in perception and mobility. To find their targets, for example, agents must be able to identify (object recognition) and move towards their targets (path planning) without damaging the environment (obstacle avoidance). More generally, any information agents can gather perceptually will help them seek and navigate to their targets and the more efficiently they navigate to the target, the better they will be at hide-and-seek.

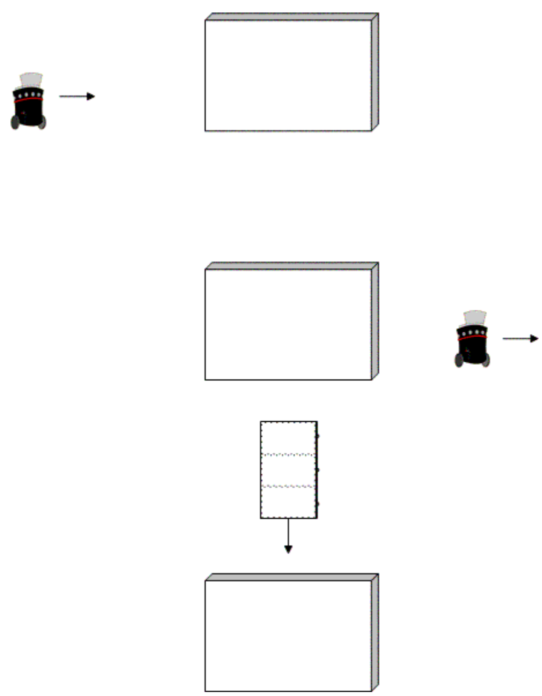

Figure 1. The robot that emerges from behind the screen can be the same robot that went behind the screen from the left because there was nothing behind the screen to block its motion. The barrier did not go behind the screen until *after* the ball did.

2.2. Temporal reasoning

In order to succeed at hide-and-seek, agents must perceive and reason about events that occur during various temporal intervals. The relations of those intervals among each other are important for predicting the outcomes of events and therefore the locations of objects that an agent might be seeking. Figure 1 shows a simple example of this. In the figure, a robot rolls behind an occluding screen and then a robot that looks the same rolls out. Next, someone places a large barrier behind the screen. Because the barrier was placed there *after* the robot-rolling event, you can assume that the space behind the screen was empty *during* the robot-rolling event and that the robot that emerged from the screen is the same as the robot that moved behind the screen.

Figure 2 presents the same scenario, except that the robot rolls behind the screen immediately *after* someone put the barrier behind the screen. In this case the robot that emerged from the screen cannot be the same as the robot the rolled behind the screen because this it did not have time to go around the barrier and it could not go through the barrier.

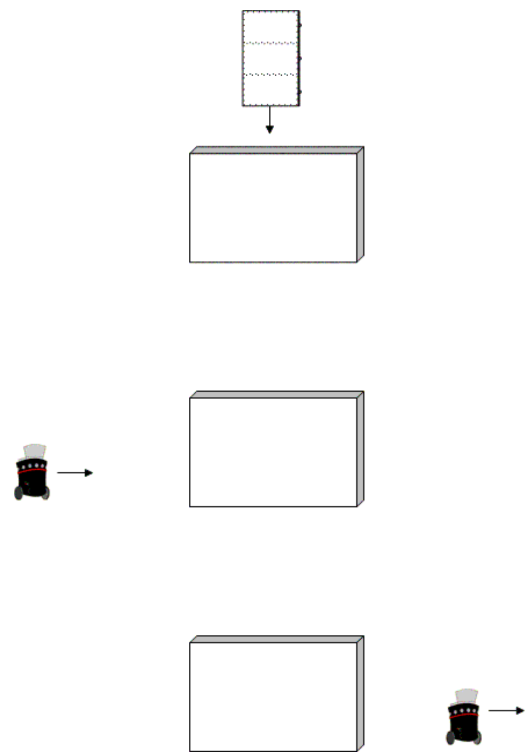

Figure 2. One knows that the robot that rolls out to the right is different from the robot that rolls in from the left because the barrier behind the screen would keep the left robot from rolling out.

In these very simple, illustrative cases and in more complex situations such as hide-and-seek, the task requires agents to make many temporal inferences in order to keep track of seeker or target agents and objects.

2.3. Logical deduction, falsification, default reasoning and explanation

Researchers using logical approaches to artificial intelligence have encountered many difficult issues regarding deduction, falsification, default reasoning and explanation, and they have constructed many sophisticated logical theories to deal with them. The following example shows that even the simplest physical interactions involve these issues.

In Figure 3a, a ball rolls towards a screen. In Figure 3b, it rolls behind the screen, but in 3c it fails to emerge from behind the screen and an object that blocked it is posited behind the screen. One can crudely formalize the inference that the ball should come out of the screen thus:

```
At(ball, left-of-screen, t1) ^
Moving(ball, right, t1) ^
Empty(behind-screen)
→
At(ball, behind-screen, t2) ^
Moving(ball, right, t2).

At(ball, behind-screen, t2) ^
Moving(ball, right, t2) ^
Empty(behind-screen)
→
At(ball, right-of-screen, t3) ^
Moving( ball, right, t3 ).
```

The inference that the ball emerges from the screen depends on the assumption that:

```
Empty(behind-screen).
```

When the ball fails to come out from the screen, you infer that the proposition,

```
Empty(behind-screen),
```

is not true and that there must be something behind the screen blocking the ball:

```
At(something, behind-screen, t2) ^
something != ball.
```

Many traditional issues from the formal logical study of intelligence arise here: what can you assume and why; what does it take to falsify an assumption; when there is more than one explanation for an event; which do you chose; etc. These are the usual issues surrounding explanation and default reasoning and they also occur whenever you try to build an effective hide-and-seek system.

2.4. Belief revision and reason maintenance

Any system that reasons in almost any nontrivial domain must often infer or assume facts that it must later revise. Because the system could have inferred more facts based on the originally assumed fact, revising its belief about the original fact is much more complicated than simply retracting it [8]. The system must retract all beliefs it inferred using the original fact that are not otherwise justified. Building systems that can revise their beliefs correctly has been a challenge for artificial intelligence researchers, even for those trying to build good models of common sense physical interactions.

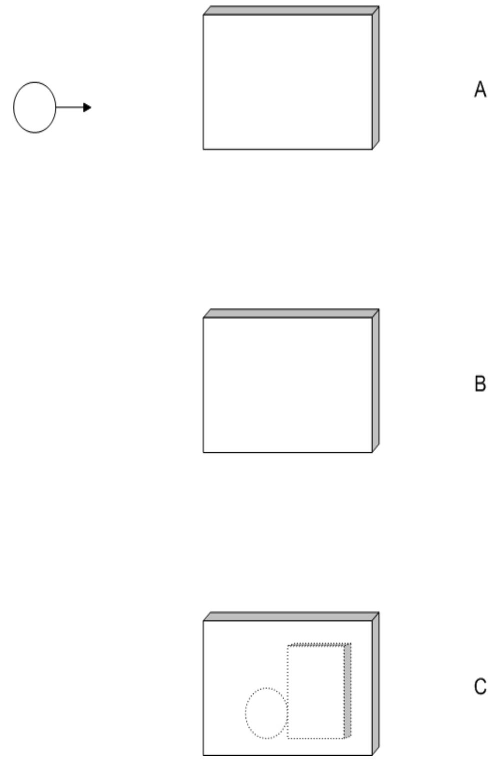

Figure 3. The ball rolls behind the screen (A), but does not roll out (B). There must be an object behind the screen that blocked it (C).

Consider an example. Figure 4a shows a scene where a screen occludes a table. A block is dropped above the table, it falls behind the screen and you infer that it comes to rest on the table. Then, when you are told that there is not just one table, but that there are two separated tables, as in 4b, you must revise your belief about where the ball went when it fell behind the screen. In this case, you assume it fell on the floor.

In general, in order to ascertain the location of any object, an intelligent system must make inferences about the object's location, which often depend on provisional information. For a system playing hide and seek, if the system spends time waiting for or attempting to acquire more definite information, the target would have more time to get away. Thus, any system that engages in the hide-and-seek task must be able to revise provisional beliefs and inferences that followed from it.

2.5. Planning, searching, problem solving

Events often have more than one possible outcome and systems can usually execute more than one action at any given time. The sequence of possible actions and/or inferences about event outcomes creates a huge "problem space" of possible world states and an intelligent system must choose a sequence of actions and/or inferences to achieve an adequate state.

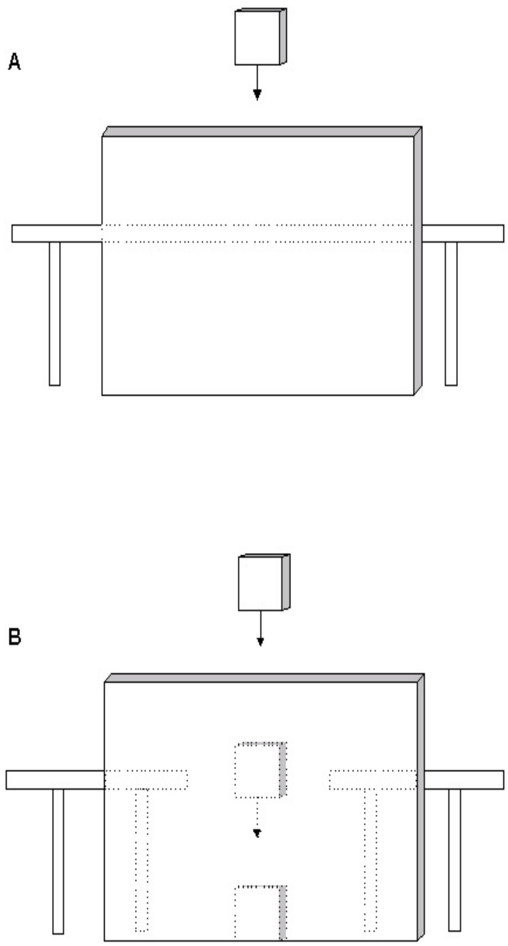

Figure 4. If, as appears in A, there is only one block behind the table, then you infer that it came to rest on the table. When you learn instead that there are two tables with a gap between them, as in B, you infer that the block fell through the gap and rests on the floor.

The need to search through problem spaces is most obvious in the hide-and-seek task when it involves robot mobility issues. Many algorithms for planning complex paths involve creating a visibility or region graph of the space and then searching the graph using traditional artificial intelligence search techniques.

Robots designed for the hide-and-seek task, however, need planning and search for much more than mobility alone. In the following example, we demonstrate that tracking the path of a simple ball can require searching through problem spaces that involve more than just the location of the ball. Figure 5 illustrates a simple physical interaction that requires backtracking search. Behind the screen in Figure 4 are two buckets. On the left, bucket A is filled with water and on the right, bucket B is full of hot coals. Figure 4 also shows a ball falling behind the screen. The ball is white and shaped roughly like a ping-pong ball, though it may be made of plastic or rubber. You see the ball fall behind the screen, though you neither see nor hear any further sights or sounds because the ball is too light to have dislodged anything and the screen masks soft noises. If your task is to figure out if the ball fell into bucket A or B, you might imagine that the bucket fell into bucket B and infer the consequences. To infer the consequence of landing in bucket B, you need to know if the ball is rubber or if it is plastic. You can imagine that it is rubber, infer that you would smell burning rubber, sense that you do not smell anything burning and therefore conclude that the ball is not rubber if it fell in B. Likewise, you can infer that the ball is not plastic if it fell into B because when you imagine a rubber ball lying in burning coals, you imagine a certain smell that you do not perceive. So if the ball fell behind B, it is neither rubber nor plastic. But you know it was one of these, so you know that the ball did not fall into B, but instead fell into A.

Similarly, the hide-and-seek task requires a broad array of search abilities.

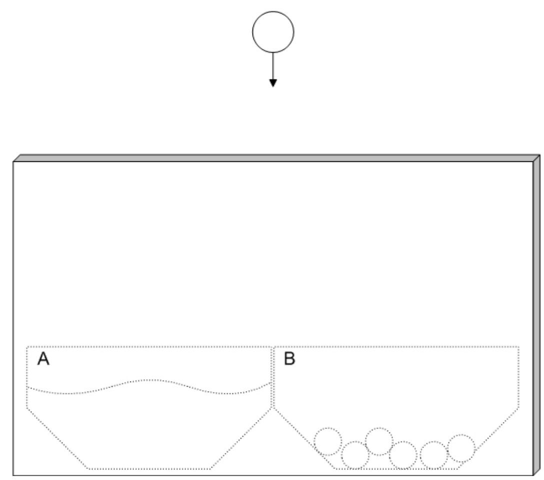

Figure 5. Bucket A is filled with water and bucket B is filled with hot coals. The ball falls into one of the two buckets.

2.6. Probabilistic inference

In many instances of hide-and-seek where events have more than one possible outcome, some are more likely than others. Seeking a target efficiently is often difficult when a scenario involves several possible series of outcomes, because the seeker must decide which of the many outcomes is most likely. Attempts to make such decisions are often called "uncertain reasoning" or "probabilistic inference".

Imagine an example like the last one, with the only difference being that you know more about the probabilities of each uncertainty. Bucket A takes twice the area of bucket B and the odds that the ball is plastic are 5:1. You are certain that you did not hear a splash, but are uncertain whether you smell any new smells. What are the odds that the ball is in A and what are the odds that it is in B?

This example shows that in order to keep track of the most likely positions of targets, agents must engage in probabilistic inferences. Similarly, hide and seek requires similar reasoning.

2.7. Social reasoning, communication and human-machine interaction

When a seeker agent is attempting to find an autonomous intelligent agent, it must be able to reason about that agent's mental state. Depending on the kind of target agent, this implies that competent behavior in the hide-and-seek task requires thought about emotions, beliefs, desires, personality traits, etc.

When a team of seeker agents is searching for a target, then each team member must be able to communicate with other team members in order to coordinate their behavior and execute a coherent search strategy. Depending on the communications abilities of the seeker agents, this could involve language use at all levels: speech recognition to decompose the acoustic signal from other agents into words, syntax and semantics to determine the meaning of the words and pragmatics to understand how the other agent(s) intend an utterance to fit into the larger joint seeking joint project.

When the seeking team contains both humans and robots, then the hide-and-seek task becomes a medium for studying robot and machine interaction with humans.

We believe that the full array of research issues in social reasoning, communications and human-machine interaction can be studied in the hide-and-seek domain.

2.8. Putting it all together

Our claim is that hide and seek is an excellent domain with which to study intelligence using integrated systems. We have presented several needed behaviors that seeker and target agents must have. We now go through a simple example of playing hide and seek to outline where each of these behaviors is needed. Bolded phrases below correspond to the behaviors we have discussed earlier.

Let us assume that we will play hide and seek with a robot. The robot will be the seeker agent ("It") first and search for the target in a room full of boxes, tables, and desks. The target initiates the game with the robot by talking to it (**communicating with it, using human robot interactions, probably language generation and language comprehension**). Next the robot counts to 10 and starts searching for the target agent. The robot must move around the environment in the room while avoiding obstacles (**perception and mobility**). The robot may also draw on its past experience playing hide and seek to determine that some places are better to hide behind and search those places first (**probabilistic inference**).

If the robot searches behind a box first, it may then assume that the target agent will not be there later (unless it moved) (**temporal reasoning**). If the robot then searches the entire room and does not find the target agent, it may reason that the target must have moved to a different (previously searched) hiding place while the robot was searching for it (**logical deduction, falsification, default reasoning, and explanation**). The robot must then decide to search the room again and re-check positions that it had already searched (**belief revision and reason maintenance**). The robot may decide that if the target agent can move around, it should search the room in such a way that allows it to see the maximum (or most likely) places the target agent would move to (**planning, searching, problem solving**). Once the robot finds the target agent, it must tell the target that it was found and perhaps even give it some feedback on how good the hiding behavior was (**social reasoning, communication, and HRI**).

2.9. Hide and seek in the real world

Many real world tasks are instantiations of the basic hide and seek domain. In the military, there are missions that directly require these skills, including ISR (Intelligence, Surveillance and Reconnaissance), RSTA (Reconnaissance, Surveillance, and Target Acquisition), and special operations including concealment. In non-military domains, hide and seek can be found in areas as diverse as Urban Search and Rescue, and inspection of facilities (e.g., searching nuclear containment facility and superfund sites).

3. INTEGRATED SYSTEMS

We now describe the three intelligent systems where high-level algorithms in the form of computational cognitive models are integrated with low-level perception and mobility algorithms, and which use various hide and seek domains. Available results will be briefly described (and the reader directed to appropriate papers for complete results).

3.1. A hybrid reactive/cognitive architecture for micro-air vehicles

We have developed a hybrid cognitive-reactive system that combines more traditional reactive, stimulus-response (S-R) behaviors with cognitive models [4][12]. In this work, we merge a cognitive model and a reactive system into a control system for autonomous vehicles. For this study, the system integrates SAMUEL, an evolutionary algorithm-based rule learning system [9] with ACT-R, a computational cognitive architecture [2]. In our hybrid system, the learning algorithm handles reactive aspects of the task and provides an adaptation mechanism, and the cognitive model handles the higher-level cognitive aspects such as planning and reasoning. The cognitive model also provides cognitive realism of the behavior.

Our hybrid controller was implemented for a simulated distributed micro air vehicle task. In the MAV task, group of vehicles cooperate to perform reconnaissance and surveillance, a version of the "seeking" task. We assumed each vehicle could detect certain ground features below the vehicle and obstacles, including other MAVs, within a defined range to the dies of the MAV. As a group, the MAVs needed to maximize the information gain about the ground features, concentrating on areas of more importance, and minimizing duplication of effort. In previous work, we successfully used genetic algorithms to evolve MAV control rule sets that could accomplish the above surveillance task [3][13].

The cognitive model implemented in ACT-R was based on the data collected during human-subject experiments performed at NRL and described in greater detail in [12]. In those experiments, the human operators would control the MAVs by directing them to goal locations using a point-and-click interface to the simulator. In this study, ACT-R, just like a human operator, was responsible for providing 2D goals to individual MAVs based on the current perception of the world. ACT-R's perception of the environment was closely matched to the perception of the human operator. ACT-R could "see" the position and state of all MAVs, and the position and value of discovered regions of interest.

SAMUEL was used to evolve stimulus-response rules to perform the collision-free reactive navigation behavior for the simulated MAVs. Each MAV used the same behavior evolved by SAMUEL in conjunction with the goals provided by ACT-R to safely navigate to a specified location. The current MAV sensor information is mapped to the conditions of the stimulus-response rules. The action of the rule that is activated specifies the action of the vehicle.

We found the performance of the hybrid controller to be comparable to the performance of the human controller, while allowing more vehicles to be controlled with fewer collisions. The model seems to capture some of the human's behavior and performance, while it also allows for higher levels of reactivity, which the humans were not able to handle. This suggests that our hybrid system is adequately modeling the humans' high-level cognitive functions, and also the difficult low-level reactive aspects.

3.2. Polyscheme

In order to study how to integrate multiple, seemingly incompatible, inferential and representational techniques, we used the Polyscheme cognitive architecture to develop the S6 system [7] that reasons about simple physical events that it perceives. This was especially helpful in understanding how high-level inference and planning techniques might combine with and help perceptual algorithms.

S6 views interactions in a simple physical world through a 2-dimensional projection of that world. S6 keeps track of the identity of objects, infers the character and existence of events it cannot see, predicts the outcome of events, explains events and nonevents and revises its inferences when it receives new information. S6 successfully reasons about many scenarios researchers present to infants and young children in order to study their knowledge of the physical world.

S6 combines specialized representation and inference techniques for identity, time, events, causality, space and paths to successfully deal with a wide range of situations. The knowledge representation schemes S6 uses include scripts, frames, logical propositions, neural networks and constraint graphs. The inference schemes S6 implements include script matching, rule matching, backtracking search, neural network propagation and counterfactual reasoning.

3.3. A learning cognitive model for playing the game of hide and seek

In our efforts to add cognitive models for higher level reasoning to traditional mobile robotics control, and to demonstrate the idea that more effective human-robot interactions are possible by using these computational cognitive models, we are modeling hide and seek behaviors in people, and using these to control a robot.

We have built a simple computational cognitive model of hide and seek. The model is based on a case study of a 3.5 year old learning to play hide and seek, specifically the learning that occurs as the child learns good and poor places to hide. The computational cognitive model is built within the ACT-R framework [2] and models the reasoning the child goes through as she plays and tries different hiding places. This is a very difficult task because there is very little feedback, very few trials, and very few suggestions.

The child went from hiding in a room with her eyes shut ("if I can't see you, you can't see me" strategy) to hiding under an upholstered chair in a different room. The model currently captures aspects of the child's learning by building a schematic representation of hiding and learning that some places are good to use as hiding places (e.g., under is good if the object is opaque; hiding under a piano is bad). The model also uses a simple ontology to reason about hiding given few suggestions ("Do not hide out in the open") and limited feedback ("You hid in a good place"). We are currently in the process of putting this computational cognitive model on a robot to play a simple game of hide and seek.

On the robot, perception is handled with a simplified perception model. Each object in the room is “labeled” with a color target whose color indicates the class of the object (e.g. desk or chair), and whose size indicates the approximate size of the object (so that the model can determine if it is big enough to hide behind. A color camera and a color blob detection algorithm [6] are used to find suitable objects to hide behind.

Low-level mobility of the robot is handled by a system that combines reactive navigation and collision avoidance, explicit path planning, map learning, and localization. This system is described in detail in [11].

4. GENERAL COMMENTS, FUTURE WORK

The proper evaluation of integrated artificial intelligence techniques can be quite challenging. In this paper, we presented the domain of hide and seek as a particularly well-suited task domain for evaluating the integration of low-level, reactive algorithms with higher-level reasoning mechanisms. Three different instantiations of integrated systems that combine cognitive-level algorithms with lower level algorithms for perception and navigation, were described.

We continue to push the integration of computational cognitive models into our systems. We believe that incorporating cognitively plausible behavior will permit more natural interactions between humans and robots. Using computational cognitive architectures and cognitive models can ease the ways in which robots communicate with their human team members, and vice versa. We have been exploring the addition of cognitive models for two goals. First, allowing the robot to use the same representations and qualitative reasoning as the human will allow for more effective and efficient communication. Second, endowing the robot with behaviors based on cognitive models of human performance allows the robot to exhibit behaviors that are similar to how a human might perform a task, thereby enhancing social human-robot interaction. Not only does this improve interactions with the robot’s human team members, but is also critical for robots that need to interact with bystanders. We will test these arguments in future research.

5. REFERENCES

[1] Altmann, E. M. and J.G. Trafton (2002). Memory for goals: An activation-based model. Cognitive Science, 26, 39-83.

[2] Anderson, J. R. and C. Lebiere (1998). The Atomic Components of Thought. Mahwah, NJ: Lawrence Erlbaum.

[3] Bugajska, M. D, A. C. Schultz, J. G. Gregory, S. Gittens, and F. Mintz (2001). "Building Adaptive Computer Generated Forces: The Effect of Increasing Task Reactivity on Human and Machine Control Abilities." In Late-Breaking Papers Proceedings of GECCO-2001, San Francisco, CA.

[4] Bugajska, M. D., A. C. Schultz, J. G. Trafton, M. Taylor, and F. Mintz (2002). "A Hybrid Cognitive-Reactive Multi-Agent Controller." In Proceedings of 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), EPFL, Switzerland; September 30 – October 4, 2002.

[5] Brooks, R. “Intelligence without representation”. Artificial Intelligence, 47:139--159, 1991.

[6] Bruce, J., T. Balch, and M. Veloso (2000). “Fast and Inexpensive Color Image Segmentation for Interactive Robots.” In Proceedings of 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Japan; October 2000.

[7] Cassimatis, N. L. (2002), Polyscheme: A Cognitive Architecture for Integrating Multiple Representation and Inference Schemes. Ph.D. Dissertation, MIT Media Arts and Sciences Section.

[8] Doyle, J. 1979. A truth maintenance system. Artificial Intelligence, 12:231-272.

[9] Grefenstette, J., C. L. Ramsey and A. C. Schultz (1990). "Learning sequential decision rules using simulation models and competition," Machine Learning, 5(4), 355-381, October 1990, Kluwer.

[10] Lenser, S., J. Bruce, and M. Veloso. In A. Birk, S. Coradeschi, and S. Tadokoro, editors, *RoboCup-2001: The Fifth RoboCup Competitions and Conferences*. Springer Verlag, Berlin, 2002.

[11] Schultz, A., W. Adams, and B. Yamauchi (1999). "Integrating Exploration, Localization, Navigation and Planning With a Common Representation," Autonomous Robots, v6 n3, May 1999; IEEE Press, pp. 293-308.

[12] Trafton, J. Gregory, Alan C. Schultz, Magdalena D. Bugajska, Shaun Gittens, and Farilee Mintz (2001). "An Investigation of How Humans and Machines Deal with Increases in Reactivity." Proceedings of Tenth Conference on Computer Generated Forces and Behavioral Representation (CGF&BR), Norfolk, VA; May 15-17, 2001.

[13] Wu, A. S. and A. Agah (1999). “Evolving Control for Distributed Micro Air Vehicles,” proc. Of the IEEE Conference on Computational Intelligence in Robotics and Automation, IEEE, 174-179