Modular Behavior Control for a Cognitive Robot

Palis Ratanaswasd, Will Dodd, *Student Member, IEEE*, Kazuhiko Kawamura, Ph.D., *Fellow, IEEE*, and David C. Noelle, Ph.D., *Member, IEEE*

***Abstract*—We propose a method for a cognitive robot behavior control in which a small number of behaviors are loaded into a workspace, called working memory, where they are combined to generate actions during a task execution. We use the existing components in our cognitive robot architecture, such as the Long-Term Memory, the Short-Term Memory, with the addition of a working memory system and a control mechanism called the Central Executive Agent to create a modular control system. This control method is used to drive the behaviors of our humanoid robot ISAC.**

***Index Terms*—humanoid robot, cognitive robot, cognitive control, behavior control, modular control, working memory**

I. Introduction

IN the future, robots will be required to exhibit robust performance in a wide range of situations. To move towards this goal, the Center for Intelligent Systems at Vanderbilt University has designed a cognitive robotic control system for the humanoid robot ISAC (Intelligent Soft Arm Control) [1]. A goal of cognitive robot is to be fluent in routine operations and become capable of adjusting behaviors in the face of unexpected situations. In order to maintain adaptability in such a complex system, robust control of action must be coupled with the ability to select and focus on important information throughout the task execution. These abilities are known to exist in human as executive functions, and are studied by cognitive psychologists under the rubric of "cognitive control." Cognitive control observed in humans is thought to be useful for a cognitive robot during the action-selection process as it guides the robot through the search for component behaviors that might be combined and used efficiently to execute routine tasks as well as to appropriately respond in novel situations.

Biological evidence of cognitive control in humans can be found in the function of the basal ganglia thalamocortical system, the prefrontal cortex (PFC), and the anterior cingulated cortex (ACC). Basal ganglia are involved in the planning and execution of complex motor and cognitive acts [2], while the PFC is involved in guiding these actions by supporting representations of relevant information from interference due to competing information [3]. The ACC is involved in detecting and helping to resolve response conflicts during a task performance [4]. Recent cognitive psychology research supports the idea that working memory plays a very important role in cognitive control and executive function [5]. Being strongly associated with the PFC, working memory can be viewed as a relatively small cache of task relevant information, grouped into "chunks", that is strategically positioned to efficiently influence behavior [6]. One classical model of working memory is that suggested by Baddely and Hitch in which the control of executive processes is done by a component called the Central Executive [7]. The Central Executive controls two working memory systems, namely the phonological loop and the visuo-spatial sketch pad. These two systems are responsible for both the storage and processing of linguistic and visio-spatial information, respectively.

Cognitive control within ISAC is modeled by the Central Executive Agent (CEA) that interfaces with a working memory system (WMS) which allows task related information to be actively maintained during a task execution. The ability to select suitable information to maintain is provided by a reinforcement learning system. Temporal Difference (TD) Learning [8] is being investigated and will be used for reinforcement learning in the future due to its similarity in function to the dopamine cells that project to the PFC [9].

In this paper, we describe the method used by the CEA and the WMS for the action execution process, including the learning of appropriate behavior combinations that allow the robot to adapt its skills in a manner similar to that used by cognitive control processes in humans. The system will select behaviors based on past experience which will help reduce the computational complexity of movement, and will allow the system to learn which behaviors are best to combine without presenting the robot with every possible combination. The process should have fast response time and with high success

This research has been supported through DARPA/NASA-JSC (Grant #NAG9-1446) and NSF (Grant #EIA0325641). Research was conducted in the Center for Intelligent Systems at Vanderbilt University and at NASA's Johnson Space Center in Houston.

Palis Ratanaswasd and Will Dodd are graduate students in Electrical Engineering at Vanderbilt University, Nashville, TN 37235 USA (e-mails: {palis.ratanaswasd, will.dodd}@vanderbilt.edu).

Kazuhiko Kawamura, Ph.D, is a professor of Electrical and Computer Engineering, and the director of Center of Intelligent Systems at Vanderbilt University, Nashville, TN 37235 USA (e-mail: kaz.kawamura@vanderbilt.edu).

David C. Noelle, Ph.D, is an assistant professor of Computer Science and Psychology at Vanderbilt University, Nashville, TN 37235 USA (e-mail: david.noelle@vanderbilt.edu).

rate as opposed to the system without past experience, where it must analyze the task context and slowly deliberate and plan in order to choose behaviors that best fit the task.

II. Multiagent-Based Cognitive Robot Architecture

ISAC's cognitive robot architecture is a multiagent-based system that incorporates several components inspired by different cognitive processes in humans. Information processing in ISAC is embedded within a multiagent-based software architecture called the Intelligent Machine Architecture (IMA) [1] [10]. ISAC's cognitive abilities are implemented as collections of IMA agents and memory structures, including one important compound agent called the Self Agent.

The Self Agent (SA) represents the robot's "self" by monitoring the robot's own internal state as well as the progress of task execution. The internal representation of the robot self (i.e. the Self Agent) is currently fixed. In the future, it will become dynamic and be updated from time to time which will allow the system to reason and act based on its status and the context of assigned tasks [11]. The SA is responsible for controlling task execution. Similar to human cognitive abilities, we are designing the SA's task execution to take into consideration the internal state of the robot as well as past experiences, emotions, and feelings, all of which are crucial elements of a human's cognitive behavior [12]. Emotional salience serves as both a reward signal and a filter for incoming information from the enormously complex world in which we live [13]. Through emotions and feelings, as well as cognitive direction, the conscious mind concentrates on the features that have proven to be important in the past.

Memory structures are utilized to help maintain the information necessary for immediate tasks and to store experiences that can be used during decision making processes. The short-term memory (STM) stores sensory information on a structure called the Sensory Ego Sphere (SES) [14]. The stored information decays over time. The long-term memory (LTM) stores information such as learned skills and semantic knowledge for retrieval in the future. The WMS is modeled after the working memory in humans, which holds a limited number of "chunks" of information needed to perform a task, such as a phone number during a phone dialing task. It allows the robot to focus attention on the most relevant features of the current task and is closely tied to the learning and execution of tasks [5]. In our implementation, the WMS consists of two functional parts: a set of working memory slots that can be accessed by other agents, and a device to select chunks of memory to place in those slots based on the current task context. The WMS works with a component within the Self Agent called the Central Executive Agent (CEA) to provide cognitive control similar to that seen in humans. This working memory model is based on models from computational neuroscience as discussed in [5]. In the future, we will incorporate the Emotion System to further help provide feedback concerning the behavior of the system. Figure 1 shows the current implementation of ISAC's multiagent-based cognitive robot architecture.

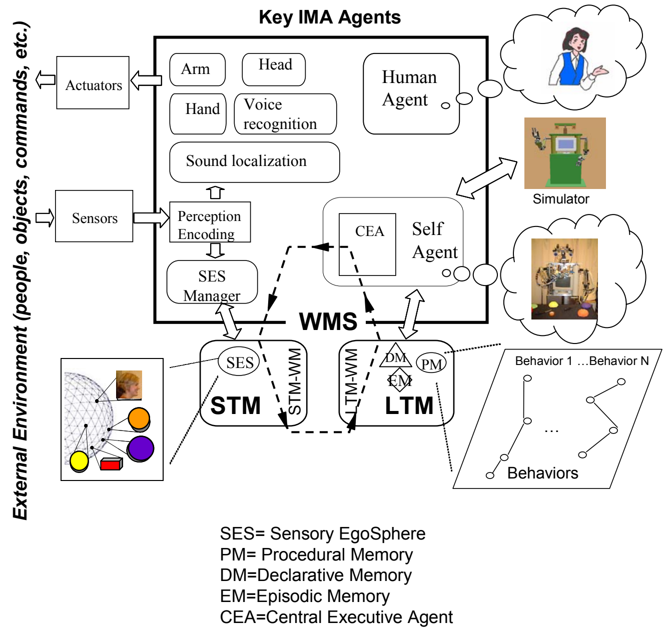


Fig. 1. ISAC's Multiagent-Based Cognitive Robot Architecture

III. Behavior Control and Generation using Modular Control

When a problem is given, most of the time it is easier to solve by decomposing it into smaller ones, allowing each to be solved individually. The method of solving problems in such a modular fashion is common in control and robotics. Wolpert and Kawato use modularity in designing adaptive low-level controllers, allowing a small set of controller modules to be chosen independently to control the system based on the current situation [15]. Likewise, behavior-based robotics benefits from the fact that behaviors can be developed individually and each one yields a different action. Simple behaviors can be grouped to work together, creating a new behavior that can yield more complex deliberation and action [16]. This has been proved to be true in case of mobile robots but it is still an unsolved problem for humanoid robots. In an attempt to construct a high-level action decision process for a cognitive robot, we have developed a modular control system that uses knowledge of its individual behaviors to predict the outcome under a task context. A set of behaviors are selected based on past experience and combined to generate an action. Advantages of implementing modular control are:

--Only a small set of system parameters are required for each behavior, therefore, changes in system configuration affect only a small number of behaviors.

--Adjustments can be made to an individual behavior without affecting the rest.

--Combinations of behaviors allow novel actions to be generated based on the task context without programming them in advance.

--The system focuses on only a small set of appropriate behaviors based on a specific task, which saves system resources.

--The system learns to adapt behaviors it already knows to fit given task requirements.

--The system chooses the best behavior based on task context as well as past experience.

Within the framework of ISAC's cognitive robot architecture, modular control is used to generate an action in response to an assigned task. This process takes place within the WMS, which contains behaviors that are selected based on past experience. That is, the CEA selects a small number of behaviors that are believed to be the best for the current task, and loads them into the WMS based on past experience as measured within a Reinforcement Learning (RL) module. The CEA then activates the modular controller with the selected modules in the WMS. The CEA determines the amount each selected behavior will contribute to a particular action based on its relevancy to the task context, and the CEA produces appropriate weights for the behaviors within the WMS. The combination of control signals is then sent to the robot for action execution. Figure 2 shows the functions of the CEA during an action selection process, based on skill knowledge and past experience.

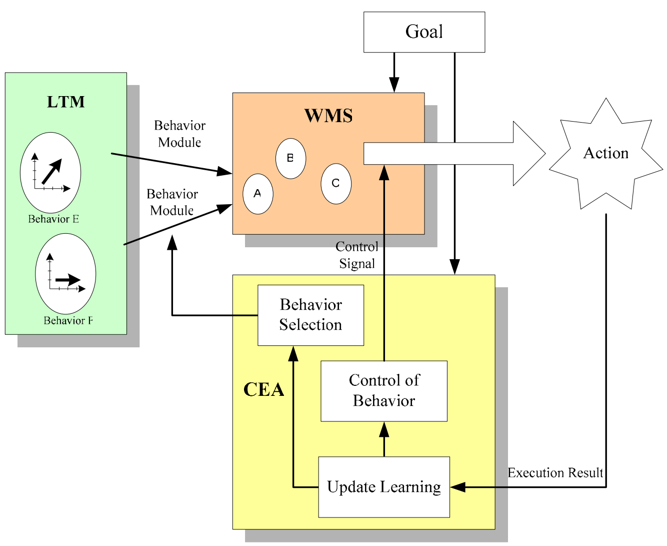


Fig. 2. Functions of the CEA in modular control

An advantage of a modular control approach as compared to other methods is that it allows for a gross reduction of the state-space while retaining the benefits of error-based behavioral control. It prevents the system from having to interpolate and execute a task using several hundred LTM units at the same time, and assists the system with learning to achieve better performance over time.

Figure 3 shows the interaction of components in our cognitive robot architecture with relation to modular control.

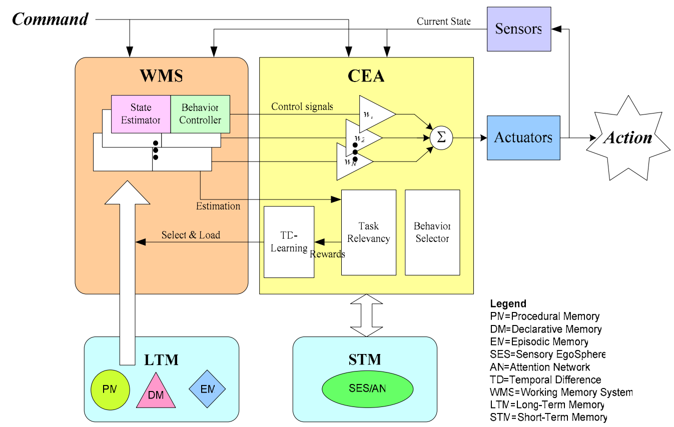


Fig. 3. Modular control in cognitive robot architecture and related components

A. Behavior Selection

ISAC's behaviors are currently taught by human, during which the robot is teleoperated through sequences of actions. The motion streams are recorded and parsed by Spatio-Temporal ISOMAP (ST-ISOMAP) [17], and stored in the Procedural Memory section (PM) in the LTM [18]. At this time, all ISAC's behaviors involve arm movements. In the future, we will include behaviors that control the movements of other actuators, including the head and hands.

Upon the reception of a task assignment, a small set of behaviors in the LTM are selected and placed into the WMS based on ISAC's past experiences with the assigned task. The experience is stored in the form of expected reward in reinforcement learning that the robot has learned to associate the task with behaviors in prior executions. This is under the condition that ISAC already has some initial knowledge on the task before. The result from the execution will be used for computing expected reward for future execution of the same task, as will be described later in section III.E.

The number of behaviors loaded into the WMS at one time is currently limited to three, which is similar to the capacity of human working memory [19]. Restricting to a small number of behaviors helps to limit the size of the search space during action selection, allows only a small amount of system resources to be used, and results in fast response times.

B. Prediction of Behavior Outcome: State Estimation

Within the WMS, each behavior is assigned a state estimator. A state estimator uses the trajectory information associated with its behavior to estimate the execution output of the next sampling time step in the control loop. The estimation is found using an interpolation technique called Verbs/Adverbs – a technique originally used to produce believable motion for computer animation [20]. The Verbs/Adverbs technique uses radial basis functions to interpolate for a set of trajectories that brings the robot closest to the goal state. The estimation is stored in the form of state variables that are sent to the CEA for "relevancy" computation.

C. *Task Relevancies*

In order to select the most appropriate behaviors to perform a task, the CEA calculates the degree of “Relevancy” of a behavior to the given task and compares these quantities across behaviors. The behaviors that have high relevancy are biased by the behavior selector for current action execution.

The relevancy value used in the current modular control system determines how a behavior can bring the state of the system toward the targeted goal state while assuring that the system state at next time step is achievable. To ensure the achievability of the next state, the CEA takes the state estimations and subtracts from the actual current state. The resulting difference, called the estimation error, indicates the amount of movement the robot has to make in order to reach the estimated configuration in the next time step. Smaller estimation error is preferred because the movement is likely to be attainable. Additionally, to determine the relevancy to goal information the CEA determines the distance between the goal state and the state estimation. The resulting error, called the goal state error, indicates how close to the goal the behavior will take the system. The calculation of both the estimation error and goal state error are done individually for each behavior.

--Let the i^{th} behavior that is loaded into the WMS be B_i. Computed at current time t, the state estimation of B_i of next time step $t+1$ is $\hat{x}^i_{t+1}$. The delayed estimation $\hat{x}^i_t$, is evaluated by the CEA to find two types of errors:

-The current state error, $ec_i(t) = x_t - \hat{x}^i_t$

-The goal state error, $eg_i(t) = x^* - \hat{x}^i_t$,

where

- x_t is the actual state of the robot at time t, and

- x^* is the goal state

--Task relevancy, λ_i, for behavior B_i at time t is then

$$\lambda_i = \alpha_c \frac{e^{-|ec_i|^2/\sigma_c^2}}{\sum_{j=1}^{N} e^{-|ec_j|^2/\sigma_c^2}} + \alpha_g \frac{e^{-|eg_i|^2/\sigma_g^2}}{\sum_{j=1}^{N} e^{-|eg_j|^2/\sigma_g^2}}.$$

The scaling factors, σ_c and σ_g, are chosen such that values of the two terms are not too different, and resulting exponentials are not too small to cause computation underflow. The relevancy weights, α_c and α_g, are assigned according to the type of task being executed.

D. *Behavior Selector*

From the relevancies vector $\boldsymbol{\lambda}(t)$, the behavior selector within the CEA calculates a weights vector $\boldsymbol{w}(t)$ that will be used to control the amount that each behavior will be involved during an action. That is, a weight $w_i(t)$ is calculated from a thresholded task relevancy multiplied by a normalization scale factor as shown in Figure 4.

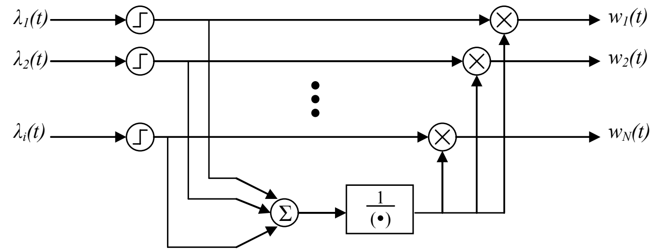
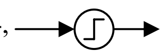


Fig. 4. Weight calculation of the Behavior Selector

In Figure 4,  represents a threshold function

$$Thr(x) = \begin{cases} x, x \geq T \\ 0, x < T \end{cases}.$$

The threshold, T, is a constant that is tuned empirically, based on experimental results.

During action execution, each selected behavior module in the working memory independently produces a control signal that controls the robot along its trajectory. Magnitudes of all the control signals are weighted and summed before finally being sent to the robot controller. As control signals are combined with time varying weights, the resulting action does not resemble only one behavior but adjusts itself based on the task context. The combination of the same set of behaviors will produce a different action when a different task is given.

E. *Reinforcement Learning*

The role of the RL module is to select the behaviors that are appropriate for the current task. A reward is calculated by the CEA after attempting the task with the given set of behaviors. This reward is then fed into the RL module so it can learn which behaviors to select the next time. The reward is equal to the average relevancies of each behavior being loaded during one action execution, discounted by the task completion time. Average relevancies, in this case, indicate how much a set of selected behaviors is suitable for that particular task based on goal information and prediction error. Thus, the most suitable set of behaviors that completes the task most quickly is likely to be selected in the future. ISAC learns which set of behaviors to select for each task. New similar tasks can be built on the learned values.

Currently this is implemented by a simple look-up table that attempts to estimate the utility of each behavior and is used to select the top three behaviors for the task. This selection system is also configured to conduct a random exploration for a small percentage of task attempts in order to prevent initially explored behavior from “hogging” activation, and to guarantee that the system will converge on a usable, reasonably explored state. In the future, we will integrate a more powerful “Working-Memory” temporal difference learning toolkit into the system, which will judge the behaviors as groups through the use of neural networks. This toolkit will also allow for the system to generalize over different contexts and tasks and to learn to group similar behaviors. [6]

IV. EXPERIMENT

We have conducted an initial set of experiments for studying robot behavior learning based on a working memory model and modular control. The first of these experiments is intended to test the following abilities in the context of generating behaviors for a single task:

--The ability to choose appropriate behaviors out of a number of behaviors to perform a task based on past experience

--The ability to switch the priority of behaviors during the course of an action, based on the current state of the robot

--The flexibility to combine behaviors to achieve a new behavior when the stored behaviors are not adequate to perform the task

In the future, the capability of the working memory to filter through large amounts of information and determine what is important for the task will be tested. As ISAC's range of movement is very limited, only a few behaviors can be taught and performed. However, we expect modular control to be much more effective in robots with higher ranges of movement such as NASA's Robonaut. This approach is also not limited to the humanoid robot platform but would work on a mobile platform as well.

A. *Experimental Design*

ISAC is endowed with a set of initial knowledge. This consists of behaviors learned in the past, such as hand waving, reaching, and arm sweeping. These behaviors are currently explicitly taught, though in the future they will be automatically learned by recording common movements as they are generated by the modular controller. This set of behaviors is taught to ISAC prior to the task assignment and have been assigned expected rewards for various tasks performance.

For the task, ISAC is asked to reach to a point on the table from its resting configuration. This point is chosen so that any one of the stored behaviors alone cannot be interpolated to perform this reaching successfully as shown in Figure 5.

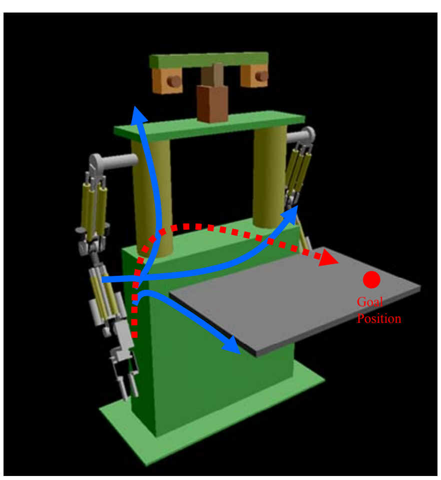


Fig. 5. Motions of stored behaviors (solid lines) and expected behavior motion (dotted line)

As behaviors must be combined in order for ISAC to reach the goal position, this part of the experiment is conducted in simulation to prevent damages that may occur from the generated motion. After we test this part to assure the safety of the motions, we will conduct this experiment on the real robot.

If ISAC cannot reach the goal position within a certain amount of time, it reports that the task is failed; otherwise it will report it as a success. ISAC will be asked to repeat the task for a number of trials to test how the algorithm improves ISAC's action decision by experimenting with various movements and weed out the ones that should be used. As the number of trials increases, the rate of failure per number of trials should decrease.

B. *Results*

When ISAC is presented with the "pointing" task, the CEA looks for behaviors in the LTM with high expected rewards for a task with the keyword "point." The initial expected rewards are given to ISAC as initial experience but are not guaranteed to be efficient. Based on this initial set of rewards, ISAC could not successfully perform the task in the first few trials. However, throughout the experiment, ISAC's behavior selection improved after a number of trials have been performed. The efficiency of the behavior selection process could be observed as the success rate became higher as the number of trials increased. Figure 6 shows the number of trials of each behavior selected by ISAC and corresponding number of success.

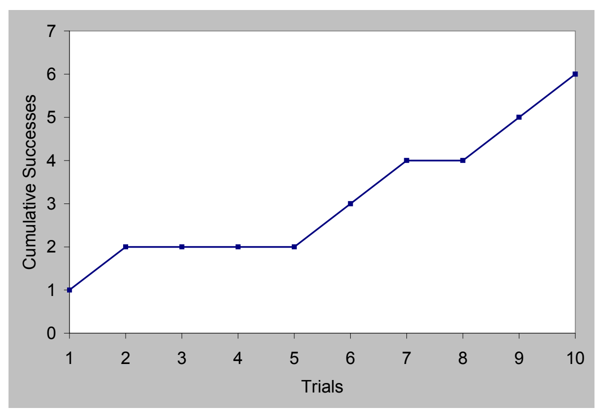


Fig. 6. Number of successful trials during the experiment

One must note that the benefit of faster response time from using WMS was not obvious in this experiment because only a small set of behaviors was presented in the LTM. In the future, when ISAC has learned a large number of behaviors, the use of WMS with reinforcement learning will become more obvious as only a few behaviors that are loaded into the WMS are involved during complex calculation of relevancies while reinforcement learning speeds up the behavior selection process.

This experiment shows another benefit of combining a number of behaviors the WMS to achieve a task as opposed to one behavior being used individually. By observing the trajectory data of the resulting motion, we can see that motions are combined together in the form of a weighted summation of control signals. Since each behavior cannot move toward the goal individually, modular control combines them to create a new motion that has a greater likelihood of reaching goal. Weights are distributed to each behavior differently during the course of execution, showing dynamic adaptability of modular control to the task context. Figure 7 illustrates the weight distribution of behaviors during an action.

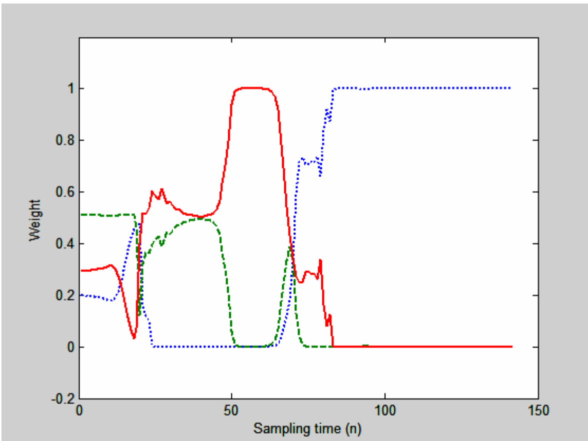


Fig. 7. Weight distribution among behavior motions; each line represents a different behavior.

This preliminary experiment shows some positive points of using modular control in robots. Other features that we are expecting to see in future experiments include the ability to focus on task context and to maintain only relevant behaviors in the presence of a second task.

V. CONCLUSION

We have developed a high-level robot control system that learns its skill knowledge, stores in the form of behaviors within a large library, and generates actions suitable for assigned tasks. In this paper, we introduced a method called modular control, which utilizes a biologically inspired working memory system to hold task related information during task execution. It uses a central control mechanism called the Central Executive Agent to select a small set of behaviors for task execution. The CEA loads behaviors into the WMS and executes them based on reward information of reinforcement learning. Execution results are used to update selection criteria in the future through reinforcement learning.

By using modular control simulation along with a working memory system, we find that the robot is capable of choosing the right set of behaviors after a small number of trials. The behaviors are selected correctly once they are learned, allowing rapid response times in later executions. Motions from executions reveal that the combination of selected behaviors makes use of time-varying weights.

The reported research is currently limited to handle a single task of arm control. Future research will include task switching performance and the controlling of different types of actuators, including the head and hands. Other types of bias will be added to widen the dimension of learned experience, such as an emotion system, so more complex actions can be learned with more quickly.

REFERENCES

[1] K. Kawamura, R.A. Peters II, R. Bodenheimer, N. Sarkar, J. Park, A. Spratley, and K. A. Hambuchen, "Multiagent-based cognitive robot architecture and its realization," *Int'l. Jo. of Humanoid Robotics*, vol. 1, no. 1, Mar. 2004, pp. 65-93.

[2] G. M. Edelman and G. Tononi, *A Universe of Consciousness: How Matter Becomes Imagination*, Basic Books, 2000.

[3] E. K. Miller, J. D. Cohen, "An intergrative theory of prefrontal cortex function," *Annual Review of Neuroscience*, 2001, pp. 167-202.

[4] C. S. Carter, T. S. Braver, D. M. Barch, M. M. Botvinick, D. C. Noll, J. D. Cohen, "Anterior cingulate cortex, error detection and the online monitoring of performance," *Science*, 1998, pp. 747-749.

[5] R. O'Reilly, T. S. Braver, and J. D. Cohen, "A biologically based computational model of working memory", *Models of Working Memory: Mechanisms of Active Maintenance and Executive Control,* (A. Miyake and P. Shah, Eds.) Cambridge; Cambridge University Press, 1999.

[6] M. Skubic, D. Noelle, M. Wilkes, K. Kawamura, and J. Keller, "A biologically inspired adaptive working memory for robots," *AAAI Fall Symp., Workshop on the Intersection of Cognitive Science and Robotics: From Interfaces to Intelligence*, Washington, DC, Oct. 2004.

[7] A. Baddeley, *Working Memory*, 11. Oxford Psychology Series. Oxford: Clarendon Press, 1986.

[8] R. S. Sutton, "Learning to predict by the method of temporal differences," *Machine Learning*, vol.3, 1988, pp.9-44.

[9] T. S. Braver, and J. D. Cohen, "On the control of control: The role of dopamine in regulating prefrontal function and working memory," In S. Monsell & J. Driver, eds., *Control of Cognitive Processes: Attention and Performance XVIII*, Cambridge, MA: MIT Press, 2000, pp. 713-738.

[10] R.T. Pack, D. M. Wilkes, and K. Kawamura, "A software architecture for integrated service robot development," *Proc. of IEEE Systems, Man and Cybernetics*, 1997, pp. 3774-3779.

[11] R. Sanz, "Modelling, self and consciousness: Further perspectives of AI research," *Performance Metrics for Intelligent Systems Workshop (PerMIS)*, Aug. 13-15, 2002, NIST, Washington, DC.

[12] A. R. Damasio, *The Feeling of What Happens: Body and Emotion in the Making of Consciousness*, New York: Harcourt Brace, 2000.

[13] N. Fragopanagos and J. G. Taylor, *Modeling the Interaction of Attention & Emotion, Brain Inspired Cognitive Systems*, Univ. of Stirling, Scotland, UK, Aug. 2004.

[14] R. A. Peters II, K.A. Hambuchen, K. Kawamura, and D.M. Wilkes, "The sensory ego-sphere as a short-term memory for humanoids," *Proc. of the IEEE-RAS Int'l Conf. on Humanoid Robots*, Waseda University, Tokyo, Nov. 22-24, 2001, pp. 451-459.

[15] D. M. Wolpert and M. Kawato, "Multiple paired forward and inverse models for motor control," *Neural Networks*, vol. 11, 1998, pp. 1317-1329.

[16] R. Arkin, *Behavior-based Robotics*. Cambridge, MA: MIT Press, 1998.

[17] O. C. Jenkins and M. J. Matarić, "Automated derivation of behavior vocabularies for autonomous humanoid motion," *2nd Int'l Joint Conf. on Autonomous Agents and Multiagent Systems,* 2003, pp. 225-232.

[18] D. Erol, J. Park, E. Turkay, K. Kawamura, O.C. Jenkins and M.J. Mataric, "Motion generation for humanoid robots with automatically derived behaviors," *Proc. of IEEE Int'l. Conf. on Systems, Man, and Cybernetics,* Washington, DC, Oct. 2003, pp. 1816-1821.

[19] N. Cowan, "The magical number 4 in short-term memory: A reconsideration of mental storage capacity" *Behavioral and Brain Sciences*, 24(1), 2001, pp. 87–185.

[20] C. Rose, M. F. Cohen, and R. Bodenheimer, "Verbs and adverbs: Multidimensional motion interpolation," *IEEE Computer Graphics and Applications,* vol. 18, no. 5, Sep.-Oct. 1998, pp. 32-40.