Strojniški vestnik - Journal of Mechanical Engineering 56(2010)11, 718-727
UDC519.6:004.85

Paper received: 27.04.2010
Paper accepted: 22.10.2010

Cloud Computing for Synergized Emotional Model Evolution in Multi-Agent Learning Systems

Tristan Barnett* – Elizabeth Ehlers
Academy for Information Technology, University of Johannesburg, South Africa

Machine learning is a technology paramount to enhancing the adaptability of agent-based systems. Learning is a desirable aspect in synthetic characters, or 'believable' agents, as it offers a degree of realism to their interactions. However, the advantage of collaborative efforts in multi-agent learning systems can be overshadowed by concerns over system scalability and adaptive dynamics.

The proposed Multi-Agent Learning through Distributed Artificial Consciousness (MALDAC) Architecture is proposed as a scalable approach to developing adaptable systems in complex, believable environments. To support MALDAC, a cognitive architecture is proposed which applies emotional models and artificial consciousness theory to cope with complex environments. Furthermore, the cloud computing paradigm is employed in the architecture's design to enhance system scalability. A virtual environment implementing MALDAC is shown to enhance scalability in multi-agent learning systems, particularly in stochastic and dynamic environments.

Keywords: multi-agent learning, cognitive architecture, emotional models, scalability, intelligent agent, cloud computing

0 INTRODUCTION

Multi-agent systems (MAS) have the innate advantage of using the collaborative efforts of multiple, interacting agents to satisfy goals and solve problems. The application of MAL to complex environment is important for application domains such as embodies agents in the form of real-world robotics, computer-human interaction, social simulation, synthetic characters and agents situated within virtual environments. However, the dynamics in MAL systems pose several challenges. Such dynamics are exasperated in complex environments that are analogous to the real-world since agents may be imbued with features such as reinforcement learning, affective guidance and cognition. Efficient cooperation mechanisms are therefore critical for *multi-agent learning* (MAL) systems to cope with the dynamics of these environments.

Consequent to having multiple learning and interacting agents, major problems are presented by MAL, including adaptive dynamics, scalability and problem decomposition [1]. These problems must be considered in MAL design in order to realize its functional advantage over single-agent learning systems, particularly in complex task environments where the agent program is more computationally demanding.

Adaptive dynamics refers to the fact that multiple agents not only adapt according to their own agent state but also in reference to the state of other agents. Agents may select goals greedily or inappropriately as a result, hindering the entire team of agents. *Scalability* is another concern in a MAS, as computational and storage requirements increase exponentially with the increased burden of interactions with other agents. *Problem decomposition* refers to the process of solving a complex problem by dividing it into a set of smaller problems. An important feature of dynamic environments is that it is often the case that multiple goals are to be satisfied and goals must be selected with other agents taken into consideration. Hence, problems must be divided efficiently between agents.

Cognitive architectures can provide a means of coping with complex environments which demand arbitrary, rather than domain-specific, task completion [2]. However, currently available cognitive architectures are not typically well-oriented toward multi-agent learning, where adaptive dynamics, scalability and problem decomposition are a concern.

The proposed *Multi-Agent Learning through Distributed Artificial Consciousness* (MALDAC) Architecture to cope with the critical aspects of adaptability and cooperation in large

*Corr. Author's Address: University of Johannesburg, PO Box 524, Auckland Park, 2006, Johannesburg, South Africa, tdd.barnett@gmail.com

teams of learning agents situated in complex environments which exhibit properties of the real world. MALDAC proposes the *Context-based Adaptive Emotions Driven Agent* (CAEDA), which uses two processing algorithms, *Adaptive Consciousness Layering* and *Adaptive Impulse Modeling* (ACLAIM), to contextualize perceptual knowledge and learn cooperatively with other cognitive agents.

1 SURVEY

An analysis of the current state of the art in MAL as well as its relation to reinforcement learning, affective computing and cognition will be discussed in Section 1. Section 1.1 discusses multi-agent learning and the relationship between the communication and cooperation. Emotional models and sociability discussed in 1.2 can be employed to improve goal selection and learning. Section 1.3 identifies some novel approaches devised to increase the adaptability and realism of agent learning. Finally, Section 1.4 discusses appropriate computer architectures for agent-based systems.

1.1 Multi-Agent Learning

Multi-agent learning (MAL) consists of several agents attempting to solve a machine learning problem through cooperative or competitive interaction [1]. Learning is a desirable aspect for agents deployed in virtual environments as it offers better adaptability and a degree of realism paramount to the simulation of agent sociability. A number of developments seek to improve the learning capacity of agents. Features such as heterogeneity and scalability are increasingly becoming a concern in modern MAS applications.

The distributed nature of multi-agent systems introduces several design issues into their development, including problem decomposition, cooperation and communication [1], [3] and [4]. MAL issues can be strongly interrelated and hence can be handled or counteracted in unison. *Cooperation* and *communication* in particular have a large impact on the success of the system [1] and are also strongly connected since cooperation can be handled by appropriate communication [3].

RL techniques typically applied to single-agent systems may be invalid in the multi-agent domain due to individual agents' inconsideration of the actions of other agents [1] and [5]. Effective cooperation is hence necessary for successful MAL implementations.

Communication is a viable means of attaining cooperation. Synthetic characters also require some degree of interaction for realistic simulations of interaction, thus communication is unavoidably necessary and important in most MAL systems. Minimizing communication overhead, however, is an important consideration to enhance MAL computational efficiency.

1.2 Emotional Modeling and Socialization

Emotional models are used to cope with goal priority by warranting motivations for actions. Emotional agents automatically adapt goal priority based on goals. Optimal agent behavior can be enhanced by emotional models in scenarios where multiple goals must be satisfied [6]. Currently, few cognitive systems take into account the effects of emotion and its role in reasoning [2]. However, appraisal theory has been used by Marinier and Laird to enhance traditional reinforcement learning algorithms to indicate that emotion offers a more flexible goal selection routine in learning applications [7]. Although the research was restricted to a single-agent domain and simplified task environment, it indicates that cognitive learning approaches may benefit from the use of emotion.

Emotional models in multi-agent systems require special attention [6] and [7]. Motivation-based approaches typically imbue agent state data with an emotional association to guide action selection. This approach can be extended to multi-agent systems by requiring that agents satisfy a social component to their emotional model. The social component ensures that communication takes place so that agents can cooperate with one another. Breazeal uses temporally-bound, goal-related drive processes to influence emotion [8]. To cope with interaction, however, a "social drive" was used, where the agent becomes obliged to either interact or cease interaction. A better approach is to allow the agent to be directly motivated by the intrinsic advantage of exchanging information, as communication offers the opportunity to refine on current knowledge.

Communicating emotional states enhances the realism of interactions between agents, particularly within synthetic character simulations. Tomlinson and Blumberg present a framework to allow emotional human-agent socialization which influence agents to form relationships with other agents [9]. The framework is highly appropriate for synthetic character development due to the interactions being more realistic to the user.

Emotion can be modeled using either the *categorical approach*, which uses fixed categories of emotion, or the *dimensional approach*. The dimensional approach represents emotion as a vector on a dimensioned space which offers flexible permutations of emotion and is easier to model mathematically. Tomlinson and Blumberg use the dimensional approach to enhance learned state-action pairs, effectively enhancing agents' interactions with other synthetic characters [10]. Although this research supported social learning, emotion was not considered as an integral cognitive component of learning and reasoning.

1.3 Cognition and Multi-Agent Learning

Motivational and cognition-based approaches are often applied to synthetic characters and robotics due to the demand of choosing between changing goals in a real-world or similarly dynamic virtual environment. Cognitive architectures offer increased realism and adaptability in situated agents and enhance the generality of their application.

A cognitive architecture can be defined as a model with a structural definition of an intelligent agent's behaviour based on the mental processing of humans or animals [2]. *Cognition* refers to the synergised effects of such faculties as learning, motivation, emotion and reasoning. Hence, the representation, organisation, utilization and acquisition of knowledge are the focal points in cognitive architecture design and are typically based on a physiological model. However, the approaches can suffer from adaptive dynamics, scalability and problem decomposition problems when applied to multi-agent scenarios. Furthermore, emotion as discussed in Section 1.2 is often omitted from cognitive architectures.

CAEDA proposes the use of artificial consciousness (AC), which attempts to transform percepts into subjective and contextualized components of information. It is assumed that this will give rise to metaphorical semantics and causal associations. One approach to AC is to augment sensory data as it traverses through various layers of perceptual preprocessing [11].

An important distinction in successful models of consciousness is that they comprise of interacting but specialized functional modules from which consciousness can emerge [12]. MALDAC, discussed in Section 3, autonomously integrates distributed cognitive modules of different agent. Hence the cognitive architecture is essentially distributed across multiple, interacting agents for enhanced robustness.

1.4 Web Services and Agents

To provide a scalable communication infrastructure, computational burdens of MAS can be distributed over the Internet. Agents can both utilize and act as a provider of a web service to enhance the scalability of intelligent systems [13] to [15]. Through web services and Internet-wide infrastructures it may be possible to handle communication in MAS environments in a scalable manner [13] to [16]. Agent-based technologies also lend themselves well to grid technologies due to common goals such as robustness, service-orientation and automation [17]. Cloud computing is a novel paradigm which improves on grid computing and service-orientation by providing a scalable and virtualized communication infrastructure with dynamic resource allocation [18].

2 PURPOSE

The purpose of MALDAC is to address cooperation of agents in MAL implementations through a structured, cognition-based communication approach. The CAEDA cognitive architecture is proposed to enable the deployment of MALDAC in complex, arbitrary task environments. Furthermore, the cloud computing paradigm is proposed to enhance the system's scalability. The motivational state data of agents is communicated to allow agents to interpret the goals and intentions of other agents. MALDAC thereby supports cooperative learning behaviour by ensuring goal selection is aligned across agents.

The focus application domain for MALDAC is that of a synthetic character simulation to illustrate the functional advantage of emotional models for coping with the dynamics of MAL systems. The model also provides an enhanced understanding of cloud computing through diversification of its applications.

2.1 Cognitive Models of Consciousness

MALDAC utilizes the adaptability of cognition to devise a scalable collaborative learning system with a flexible application domain. The flexibility of MALDAC is achieved by accommodating for the adjustment of the number of simultaneous processes used by each agent while learning.

The success of an artificial consciousness system can be evaluated through continuous replacement of functional modules [12]. This measure of success makes the development of cognitive MAL systems well suited to the cloud computing paradigm, where system components can be imported from web services over the Internet. MALDAC exploits the opportunity of using on-demand functional modules, massively improving system scalability and adaptability.

2.2 Web Service and Communication

Bargelis et al. propose an intelligent interfacing module of process capability (IIMPC) to improve activity integration and mitigate the necessity to introduce new solution processes [19]. Knowledge integration provided improved support for modeling in virtual environments in the manufacturing domain. However, the system was constrained in terms of its application domain. There is hence a need to direct future research at generality of application.

MALDAC is a multi-agent system, which is distributed by nature and hence must consider knowledge integration. To ensure generality of application, knowledge integration is supported by emotion and cognitive processes. As previously mentioned, agents should be directly motivated by the knowledge exchange that takes place during interaction. To mitigate the cost of communication, agents should only communicate because they require information, not because they feel obliged to.

MALDAC agents hence only send requests for interaction when (1) current homeostasis functions remain unsatisfied and no action plan has yet been discovered and learned by the agent, or (2) when the MALDAC web service agent was previously unable to provide adequate assistance to the agent. Communication to the web service will hence only take place when the various modules of cognition are inadequate for a given problem or when the agent believes that the web service agent would benefit from information the agent has discovered. The amount of functional attention the server gives to an agent is dependant on the relative urgency of that agent's needs or on the relevance of the data in relation to a currently unresolved problem.

3 DEVELOPMENT AND APPLICATION

The goal of MALDAC is to provide a scalable implementation for highly adaptive agents, particularly in partially observable, stochastic and dynamic environments. Section 3 discusses the architecture in detail and explains its experimental application to synthetic agents behaving rationally in a simulated 3D environment. The CAEDA Architecture is proposed as a cognitive architecture to allow for flexible deployment of multi-agent learning services. CAEDA maintains three core assumptions. Firstly, that there is a relationship between motivation, emotion and learning. Secondly, that there is a functional advantage in a duality of conscious (deliberate) and unconscious (automatic) behavior and that this differentiation is continuous. Finally, that the interaction of long-term explicit (deliberate), long-term implicit (reflexive) and short-term (working) memory are vital to effective general-purpose learning.

3.1 Adaptive Emotional Models and States

To cope with adaptability, the architecture uses emotional models. Section 3.1 discusses the system's emotional model implementation. The dimensional approach to emotional models uses a vector in n-dimensional space to represent a multitude of emotional combinations. The wellknown Pleasure-Arousal-Dominance (PAD) model used by Tomlinson and Blumberg in "AlphaWolf" [10] consists of a vector defined on a three-dimensional plane with the axes 'pleasure', 'arousal' and 'dominance'. With these axes one can program approach-avoidance, fear-confidence and positive-negative valence responses to environmental stimuli. Goal selection is based on the selective balancing of emotional indicators, hence keeping the overall emotional state of the agent in *homeostasis*.

The CAEDA agent adopts the PAD dimensional approach by using multiple regulatory emotional state vectors (see Fig. 1) and associating vectors with the CAEDA agent's knowledge elements. In Fig. 1, circles indicate emotional states of interest, whilst their size represents their relative intensity. The vector V will cause the strongest drive in the agent, since it is furthest from homeostasis points and, in particular, the dominant homeostasis point U.

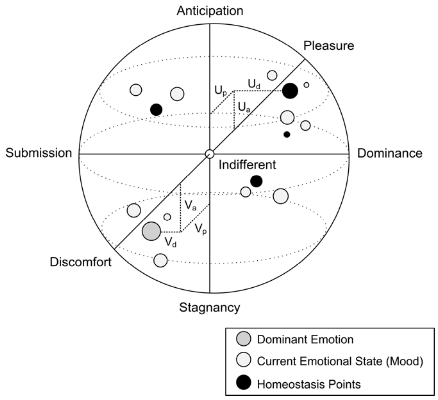

Fig. 1. *Conceptualized view of the PAD emotional model used by CAEDA*

CAEDA adapts the concept of "drive functions" used by Breazeal [8]. Drive functions are regulatory controls that govern agent behavior. Drive functions represent low level needs (goals) of the agent, such as the need for sustaining energy and completing its designed purpose. The CAEDA agent uses a drive function to regulate emotion with drive values derived from an exponential function of the homeostatic deviation. The intensity at which the agent selects to satisfy emotions that are out of homeostasis is based on homeostasis drive functions. The value of homeostasis is determined by continuous analysis of related drive functions and adjusted automatically. In a synthetic character, for example, the discomfort (pain) emotion is triggered by "hunger" percepts. The sight of food is associated with "hunger satisfying" percepts and therefore triggers a pleasure-anticipation (hopeful) emotion. Finally, an opponent nearby the food item may trigger a submission (hesitation) or dominance (aggression) emotion based on if the opponent is perceived as stronger or weaker.

A change in a drive function results in a mapped affect on the agent's emotions. This mapping is initially based on default settings by the designer, such as the emotion of discomfort being experienced during low energy levels or as the result of a potentially damaging collision on the agent's body. These mappings adapt as the result of learned emotional associations during ACL processing discussed in Section 3.2. A submissive emotion may become associated with collisions as the result of experiencing typical pain and an anticipation emotion when a source of energy is found. In this scenario, submissive behavior causes a fear of moving fast in difficult to navigate areas – not because speed is related to the pain caused by a collision itself, but because of the learned association that collisions at high speed yield greater pain. Therefore the walls are not avoided altogether as it would be with ordinary pain-association, but simply treated more cautiously as the result of submission emotion.

The activity of CAEDA agents is motivated by rectifying deviations from homeostasis drive functions (see Fig. 2). The function that evaluates to the greatest drive value will take precedence when presented with associated stimuli. Drive is evaluated as:

$$f(x) = d \cdot |d|^s, \text{ where } d = x - h . \quad (1)$$

In Eq. (1), $|d|$ takes the absolute deviation from homeostasis and is used in an exponential function such that the deviation increases the overall drive exponentially. The absolute deviation is raised to the exponent constant s, the stability factor, which determines the strictness that the drive function be maintained. Lower values will yield less drastic variations of drive at low homeostasis deviations, as seen in Fig. 3. Emotional stability can hence be maintained using restrictive homeostasis functions. This is important as it prevents extreme emotional responses that are undesirable for the application domain. For example, a synthetic character that is only slightly hungry should only opportunistically pursue food. Only when very hungry should it aggressively seek to satisfy this need.

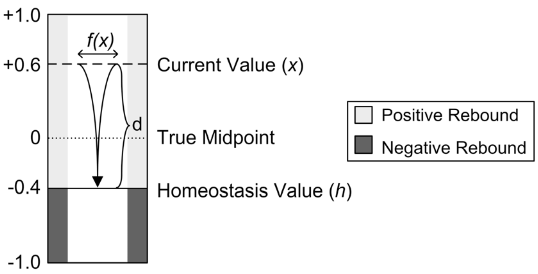

Fig. 2. *Homeostasis drive function of the CAEDA agent*

Drive function are defined to specify the agent's functional needs, which in turn affect emotion and hence goal selection of the agent. Agents develop disposition to environmental queues based on the configuration of homeostasis vectors, hence obtaining rational goal selection.

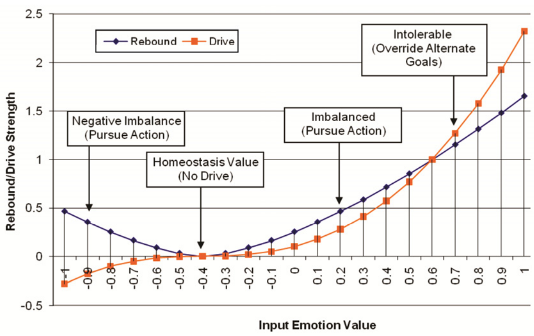

Fig. 3. *CAEDA Drive Function with Homeostasis h=-0.4 and Stability s=1.5*

3.2 Cognition and Knowledge Representation

Section 3.2 discusses how percepts in CAEDA are filtered and transformed into learned knowledge with contextualized semantics and how this knowledge is retrieved from the agent's memory. The percept filtering and transformation process is important as it determines which percept sequences to transmit to other agents and why such data will contribute to effective cooperation in learning efforts.

Gielingh proposes that knowledge can be continually refined in an iterative cognitive cycle [20]. In the cycle, knowledge is developed through an impression of the environment and thereafter a hierarchical action selection structure is developed as a solution. The system confirmed efficiency gains in continual task improvement. However the system was supported by human input. CAEDA requires that learning methods need no human input or parameterization. Hence, CAEDA has to independently determine which information is relevant.

Most perceptual information sensed from the environment is irrelevant to the agent's current goals. Determining which sequences of percepts are of interest to the agent requires placing percept data in context with what the agent already knows – a process which will be called *contextualization*. This allows noise to be ignored and associations between novel percepts and recognized percepts to be built.

The approach to knowledge representation the CAEDA agent uses is inspired by the principal of "reconstructive evidence" used by forensic investigators. The intention is to relate data to the point of being able to identify causality. The types of reconstructive evidence the system should consider include:

- *Temporal data* to limit the life-span of percepts and place events in sequence relative to one another.
- *Relational data* to associate percepts and hence build percept sequences.
- *Functional data* to identify the percepts' relation to the agent's goals and intentions.

Contextualization takes place by traversing percepts through various layers of cognitive processing, adding and modifying temporal, relational and functional data (see Fig. 4). Nodes are classified based on their priority and are terminated, acted upon or undergo AIM processing. AIM involves (a) adding temporal data, (b) determining relevance to drive functions and (c) creating emotional associations. Drive function values and emotional state vectors are hence updated in AIM. AIM also updates node priority values. Nodes of intermediate priority that neither result in action nor in termination are subject to communication.

Artificial consciousness approaches often use a layered approach to perceptual processing [11], where different layers deal with either reflexive or more deliberate perceptual tasks. The shortfall of the layering technique is the vagueness behind determining which percepts are addressed by which functional module. Langley et al. propose that new research in cognitive architectures should allow for knowledge utilization to be able to dynamically change between deliberate and reflexive modes based on the situation [2]. The ACL model the hence follows this suggestion as it may better support learning in dynamic environments. In the ACL approach, percepts traverse between levels of consciousness continuously based on the subjective experiences of the agent (see Fig. 5). Precepts are contextualized based on consciousness rating and priority. Nodes such as *D(t)* are terminated due low priority relative to time. Learning can occur on varying levels to address learned reflexive responses or deliberate behavior.

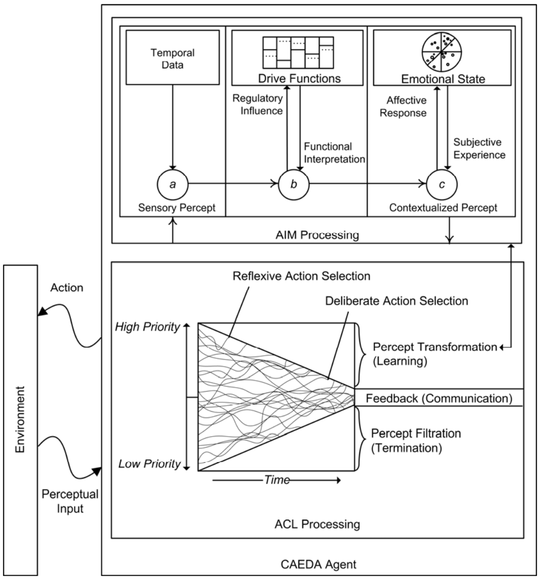

Fig. 4. *Model of the CAEDA agent with ACL and AIM processing methods*

In ACL, input percepts result in the creation of perceptual elements, called *nodes*. Nodes consist of a priority and consciousness rating, an emotional tag and links to other nodes. Links between nodes have a strength value and an elapsed time value. *Priority* increases when emotional imbalances are triggered. Nodes with high priority are more likely to be acted on and are committed to memory. Low priority nodes decay over time and are eventually terminated. *Consciousness rating* is increased over time and determines the amount of contextualization a node is subjected to, which in turn allows longer action sequences to be developed.

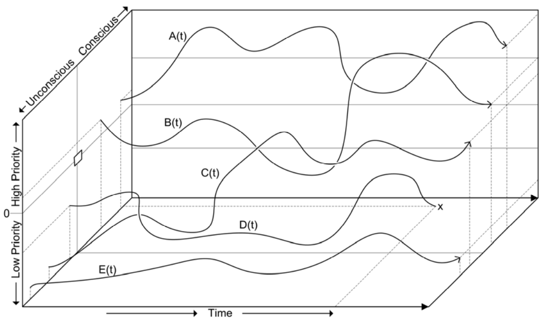

Fig. 5. *Precepts subjected to the ACL model*

Consciousness rating hence affects the level (reflexive or deliberate) cognitive attention. Unconscious perceptual nodes result from sensory input and form low-level habituated responses which guide reflexive behavior. Reflexive actions are triggered first because reflexive behavior is subjected to the earliest processing. Nodes with emotional values that are strongly out of homeostasis in reference to the agent's current emotional state are shifted closer to the conscious layer. Actions associated with nodes are initially random but are refined when action sequences are determined to have higher utility in solving the imbalance of homeostasis functions. With more deliberate actions, sequences of reflexive behaviors can be chained. The chaining process allows action sequences to be learned. ACL also maintains a separation of long-term memory from short-term memory. The continued contextualization of percepts increases their lifespan and thereby commits them to long-term memory. Consciousness rating determines whether long-term memories are explicit or implicit, allowing for differentiation of reflexive or deliberate learned action sequences.

3.3 Motive for Communication

As previously stated, emotional models are typically applied to agent-based systems to enhance goal selection. Communication acts are also viable goals that allow agents to cooperate with others [3]. However, in order to minimize communication burdens on the system, the MAS system designer must ascertain as to what constitutes as an appropriate communication act. It is possible to use a social drive in which agents are obliged to communicate periodically [8]. It is also possible to implement procedures that force agents to communicate when knowledge is of relevance to themselves or others [3]. In the case of learning systems however, knowledge exchange itself should be a significant goal of the agent. Learning is a more difficult goal to realize because the reward is intangible. To realize this objective, emotional state data is directly involved in communication acts in MALDAC.

Contextualized percepts with high consciousness and high priority ratings are transferred to other agents (indicated as “feedback” in Fig. 4). They contain motivational data through emotional tags and are sent as groups of linked percepts which allow causal relations to be built by a receiving agent. These received percepts are processed by the recipient using the ACLAIM method and are therefore prioritized and memorized in a typical fashion. Following processing, the receiving agent will “empathize” with the other agent, understanding its emotions and intentions and hence acting with other agents’ motives in mind. Agents manage conflicting goals through an evaluation of relative purpose. Furthermore, evidence for need satisfiers can be shared between agents. For example, assuming that two agents needed to cross the same single-lane bridge, the agent with the lowest emotional imbalance will yield.

3.4 Cloud-Based Agent

To enhance scalability, MALDAC uses a cloud-based service agent through which CAEDA agents can leave behind information for others to access, with the web service agent acting as a middleman. This indirect communication is less of a burden to multi-agent systems and promotes scalability [1]. Furthermore, web services allow agents to access and integrate diverse aspects of cognitive function, allowing agents to themselves adapt to specialized tasks. Agents with differing functional requirements can be augmented by subscribing the agent to web services with the appropriate cognitive modules.

Figs. 6 and 7 illustrate MALDAC with a web service agent communicating with a CAEDA agent. The web service agent searches its knowledge repository and returns nodes pertaining to the problem. The response nodes may contain a solution action sequence or simply additional contextualization data that the querying agent might be able to utilize to develop a solution.

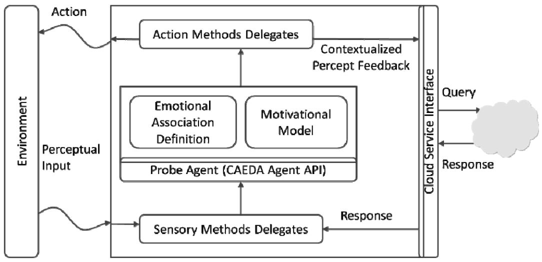

Fig. 6. *Client-side MALDAC architecture*

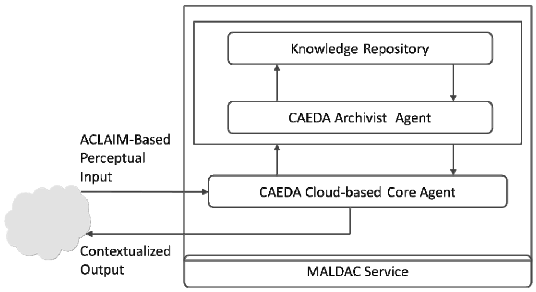

Fig. 7. *Server-side MALDAC architecture*

An interesting feature of MALDAC is the communication of motivational states of agents to align agent goal selection, as agent goal selection is based on the agent’s emotional state as well as being influenced by others agent’s emotional states. Any agents connected to MALDAC will inadvertently synchronize their goal selection to support team members and take action when expected to by the team. Communication is hence mitigated, with only data relevant to agent goal selection being transmitted. This optimization is a result of ACLAIM, where reflexive knowledge is left to individual agents but deliberate knowledge is candidate to communication. The MALDAC-based web service agent stores knowledge gathered by other agents to build a shared repository of knowledge. This repository serves to collect and provide successful, learned behaviors in an on-demand manner to CAEDA agents and allows continuous improvement of learned patterns of behavior.

4 RESULTS AND DISCUSSION

The HIVE simulator has been developed as a complex virtual environment for CAEDA agents and as a test bed to MALDAC. Robotic entities are presented with a maze, randomly generated using a recursive backtracking algorithm (see Fig. 8). Goals are placed throughout the maze. Agents must collaboratively learn the locations and times at which goals are accessible and learn a compromise on the allocation of the limited resources. The simulator is designed to recreate the ever-changing needs and daily demands of synthetic characters.

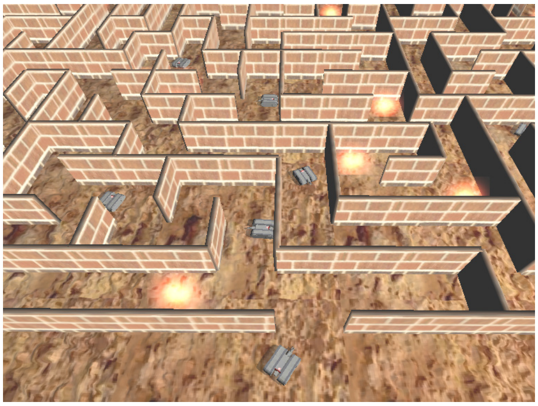

Fig. 8. *Exploration robots in the HIVE simulator*

To simulate extreme dynamics in the environment, a weather system that presents threats to the agent has been developed. The input of a human controller contributes additional dynamics to the scenario. Agents must achieve their own defined purpose by keeping their subsystems in homeostasis. The multiple and dynamic goals include collision avoidance; exploration and sample gathering tasks; stored energy conservation and solar recharging; and internal temperature regulation. The CAEDA API has been developed and controls robotic entities. The simulator has been integrated with a MALDAC hosted on a private computing cloud, with a specialized CAEDA agent acting as the web service agent.

Enhanced agent activity is acquired via interactions with the web service agent using ACLAIM. Heterogeneity is supported, as agents need only maintain learned knowledge that directly pertains to their own behavior. The system implementation is shown to be robust, as agents maintain their own copy of learned knowledge. Web service agent modification immediately affects and benefits the client implementation without client's software needing changes due to the standardized communication mechanisms.

MALDAC has improved on alternate cognitive architectures by integrating support for cooperative multi-agent learning. Furthermore, the scalability often lacking in cognitive architecture has been carefully considered in MALDAC. Finally, MALDAC has employed emotion as an integral mechanism of rationality and learning, which is often neglected in cognitive architecture [2].

However, current human interaction in MALDAC has extended only to environmental manipulation. Human interaction could be introduced to support the training of agents. A fundamental part of the client agent is choosing when to access the web service. Should multiple web services that support this architecture be available to the agent, it would be advantageous for the agent to be capable of learning the reliability of various web services for solving specific types of problems that the agent encounters. A more standardized communication scheme will also benefit the deployability of MALDAC. A further investigation of the agent's cognitive module selection process is also needed.

5 CONCLUSIONS

MALDAC provides a scalable solution for cognitive multi-agent learning architecture, suitable for partially observable and dynamic environments. MALDAC offers promise for a scalable cognitive MAL system, in which trained modules of cognitive processing could be distributed across geographically separate systems. Furthermore, MALDAC advances research of cognitive architectures by integrating the robustness of multi-agent learning and adaptability of emotional learning.

6 ACKOWLEDGMENTS

The financial assistance of the National Research Foundation (NRF) of South Africa towards this research is hereby acknowledged. Opinions expressed and conclusions arrived at, are those of the authors and are not necessarily to be attributed to the NRF.

7 REFERENCES

[1] Panait, L., Luke, S. (2005). Cooperative multi-agent learning: The state of the art. *Autonomous Agents and Multi-Agent Systems*, vol. 11, p. 387-434.
[2] Langley, P., Laird, J.E., Rogers, S. (2009). Cognitive architectures: Research issues and challenges. *Cognitive Systems Research*, vol. 10, no. 2, p. 141-160.
[3] Vlassis, N. (2007). A Concise introduction to multiagent systems and distributed artificial intelligence. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, vol. 1, p. 1-71.
[4] Sycara, K.P. (1998). Multiagent systems. *AI magazine*, vol. 19, p. 79-92.
[5] Shoham, Y., Powers, R., Grenager, T. (2004). Multi-agent reinforcement learning: a critical survey. *AAAI Fall Symposium on Artificial Multi-Agent Learning Conference Proceedings*, p. 89-95.
[6] Maria, K.A., Zitar, A.Z. (2007). Emotional agents: A modeling and an application. *Information and Software Technology*, vol. 49, p. 695-716.
[7] Marinier, R.P., Laird, J.E. (2008). Emotion-Driven Reinforcement Learning. *Proceedings of the 30th Annual Conference of the Cognitive Science Society*. Austin, TX: Cognitive Science Society, p. 115-120.
[8] Breazeal, C. (2003). Emotion and sociable humanoid robots. *International Journal of Human-Computer Studies*, vol. 59, p. 119-155.
[9] Tomlinson, B., Blumberg, B. (2002). Social Synthetic Characters. *Focus: Education*, vol. 36, p. 5.
[10] Tomlinson, B., Blumberg, B. (2003). Alpha Wolf: Social learning, emotion and development in autonomous virtual agents. *Lecture Notes in Computer Science*, p. 35-45.
[11] Arrabales, R., Ledezma, A., Sanchis, A. (2009). CERA-CRANIUM: A test bed for machine consciousness research. *International Workshop on Machine Consciousness*, Hong Kong, p. 1-20.
[12] Arrabales Moreno, R., Sanchis de Miguel, A. (2008). Applying machine consciousness models in autonomous situated agents. *Pattern Recognition Letters*, vol. 29, p. 1033-1038.
[13] Richards, D., Sabou, M., van Splunter, S., Brazier, F.M.T. (2003). Artificial intelligence: A promised land for web services. *The Proceedings of the 8th Australian and New Zealand Intelligent Information Systems Conference ANZIIS*, p. 10-12.
[14] Wijngaards, N.J.E., Overeinder, B.J., Van Steen, M., Brazier, F.M.T. (2002). Supporting internet-scale multi-agent systems. *Data & Knowledge Engineering*, vol. 41, p. 229-245.
[15] Gaspari, M., Dragoni, N., Guidi, D. (2004) Intelligent web servers as agents, from: *http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.1.1642&rep=rep1&type=pdf,* accessed on 2009-08-08.
[16] Ermolayev, V., Keberle, N., Plaksin, S., Kononenko, O. (2004). Towards a framework for agent-enabled semantic web service composition. *International Journal of Web Services Research*, vol. 1, p. 63-87.
[17] Foster, I., Jennings, N.R., Kesselman, C. (2004). Brain meets brawn: Why grid and agents need each other. *Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multi-Agent Systems*, p. 8-15.
[18] Vaquero, L.M., Rodero-Merino, L., Caceres, J., Lindner, M. (2009). A break in the clouds: towards a cloud definition. *SIGCOMM Comput. Commun. Rev.*, vol. 39, p. 50-55.
[19] Bargelis, A., Kuosmanen, P., Stasiskis, A. (2009). Intelligent interfacing module of process capability among product and process development systems in virtual environment. *Strojniški vestnik - Journal of Mechanical Engineering*, vol. 55, no. 9, p. 561-569.
[20] Gielingh, W. (2008). Cognitive product development: a method for continuous improvement of products and processes. *Strojniški vestnik - Journal of Mechanical Engineering*, vol. 54, no. 6, p. 385-397.