Ethological Concepts in Hierarchical Reinforcement Learning and Control of Intelligent Agents

Pavel Nahodil
Department of Cybernetics
Czech Technical University in Prague
Karlovo náměstí 13, 121 35 Prague, Czech Republic
E-mail: nahodil@fel.cvut.cz

KEYWORDS
Agent, Simulation, Control, Reinforcement Learning, Ethology, Behavioral Approach, Artificial Life.

ABSTRACT

This paper integrates rigorous methods of reinforcement learning (RL) and control engineering with a behavioral (ethology) approach to the agent technology. The main outcome is a hybrid architecture for intelligent autonomous agents targeted to the Artificial Life like environments. The architecture adopts several biology concepts and shows that they can provide robust solutions to some areas. The resulting agents perform from primitive behaviors, simple goal directed behaviors, to complex planning. The agents are fully autonomous through environment feedback evaluating internal agent state and motivate the agent to perform behaviors that return the agent towards optimal conditions.

This principle is typical to animals. Learning and control is realized by multiple RL controllers working in a hierarchy of Semi Markov Decision Processes (SMDP). Used model free Q(λ) learning works online, the agents gain experiences during interaction with the environment. The decomposition of the root SMDP into hierarchy is automated as opposed to the conventional methods that are manual. The agents assess utility of the behavior and provide rewards to RL controller as opposed to the conventional RL methods where the rewards situations map is defined by the designer upfront. Agent behavior is continuously optimized according to the distance from the agent's optimal conditions.

INTRODUCTION

Trend in artificial intelligence leans towards communities of robots - agents. These structures appears in nature in all types of complexity starting from genes, cells, multi-cell structures, through plants, animals, groups of animals up to their societies. Similarly to the Nature also in robotics it is obvious that one super-intelligent robot (and therefore expensive) is with its abilities far behind a swarm of small, simple and less intelligent but also the less expensive robots. It is believed, that the power lies in quantity and simplicity. Also the range of possible types of tasks implemented on community of mutual cooperative robots is much wider than in case of one super-robot. Therefore approaches from Artificial Life (ALife) are more often applied for control of community of robots. ALife approach (biological model) of implementation of intelligent behavior is inspired mostly by nature phenomena, instead of classical artificial intelligence (rational approach) which is more concerned about logic, rationality and just partially on algorithms inspired by nature. Another significant difference between AI and ALife approach is in the object of interest. Artificial intelligence is focused on complex thinking for example chess playing, text understanding, disease diagnostic etc. ALife if on the other hand focused on basic elements of natural behavior with strong stress on survival in environment. The most of existing ALife approaches are based on algorithms, which enables robots as artificially created creatures, to evolve and adapt (Kadlecek, Nahodil, 2001).

This article provides novel approach to a single agent architecture design. The primary motivation behind this research was to test the possibility of integrating rigorous methods of reinforcement learning and control engineering with behavioral (ethology) approach to agent technology. This work deals with a single agent architecture, rather then modeling multi-agent system. The main outcome of this research is an architecture called HARM designed for intelligent autonomous agents intended to behave in complex Artificial Life like environments. The term "Artificial Life like environments" prescribe environments where the agents are supposed to perform from primitive behaviors (e.g.: escaping from predators, eating, avoiding obstacles), simple goal directed behaviors (e.g.: finding water reservoirs), to complex planning (e.g.: assembling a shelter, getting through a maze with traps and riddles). The agent continually optimizes its behavior to increase its survival probability. Attainment of this goal depends mainly on how and how fast the agent can avoid danger, manage its own resources and gain resources from the environment. The agent shall be able to react to local disturbances and uncertainties, continuously improve its behavior and adapt to more persistent changes in the environment.

This research has been inspired by biology and by ethology and builds upon three fields:
Ethology/biology, control engineering and *reinforcement learning (RL).*

Proceedings 23rd European Conference on Modelling and
Simulation ©ECMS Javier Otamendi, Andrzej Bargiela,
José Luis Montes, Luis Miguel Doncel Pedrera (Editors)
ISBN: 978-0-9553018-8-9 / ISBN: 978-0-9553018-9-6 (CD)

Ethology and biology gives the concept of animal behavior, homeostasis (self-regulation), motivation, and stimulation. Control engineering provides framework for modeling agent, its internal dynamics, and optimization through environmental feedback. RL [11] provides the agent with capabilities to learn and adapt its behavior in unsupervised manner on the basis of sparse, *delayed reward signals* provided when the agent reaches desired goals. *Hierarchical approach to RL* helps to combat complexity of the learning space that is typical for ALife environments.

DESIGN APPROACHES TO AGENT CONTROL ARCHITECTURE

Design approaches can generally be divided into three primary groups with this research following the hybrid paradigm:

1) **Knowledge Based Architecture (Top-Down):** Knowledge based architectures use knowledge to guide agent behavior. Much of the debate regarding the role of knowledge within an agent centers on how it is represented within the context of the control system. Steels [Steels, 1995] considers knowledge representations to involve „physical structures which have correlations with aspects of the environment representation (relationship with the external world) and thus a predictive power (ability to predict from actual knowledge) for the system“. Agent modules are functional blocks like planning, learning or perception blocks and the behaviors (avoid obstacles, identify object, explore the environment) emerge from the interaction of the modules.
2) **Behavior Based Architecture (Bottom-Up):** Behavior based architectures build up the system of behavior producing modules instead of functional, as is the case in knowledge based architectures. Agent modules are behaviors such as avoid obstacles, identify object, and explore the environment. The functions of the system (planning, learning and others) emerge from the interaction of these modules and the environment. The most important phenomenon of behavior-based architectures is *emergent behavior*. Emergent behavior implies a holistic capability where the sum is considerably greater than its parts. Emergence is the appearance of novel properties in whole system and intelligence emerges from the interaction of the components of the system.
3) **Hybrid Approach (Combined):** Hybrid architectures integrate a knowledge-based component for planning and problem solving with behavioral components that produce robust performance in complex and dynamic domains. Strong evidence exists, that hybrid systems are found in biology, implying that they are compatible, symbiotic, and potentially suitable for use in agent control. The focus of research in this area lies in defining an interface between deliberation and reactivity, which is poorly understood so far. Hybrid models include *hierarchical integration* and *coupled planning and reacting*.

BIOLOGICAL AND ETHOLOGICAL BASIS

This section introduces concepts I have reused from biology and ethology. Research of autonomous agents is being inspired by several disciplines. The study of animals and their behavior is among the main contributors. Animal behavior can be categorized into three major classes:

- **Reflexes** are rapid, automatic involuntary responses triggered by certain environmental stimuli. The reflexive response persists only as long as the duration of the stimulus. Further, the responsive intensity correlates with the stimulus's strength. Reflexes are used for simple activities or highly coordinated ones such as locomotion.
- **Taxes** are behavioral responses that orient the animal toward or away from a stimulus (attractive or aversive.) Taxes occur in response to visual, chemical, mechanical, and electromagnetic phenomena in a wide range of animals.
- **Fixed Action Patterns** (FAP) are time-extended response patterns triggered by a stimulus but persisting for longer than the stimulus itself. The intensity and duration of the response are not governed by the strength and duration of the stimulus, unlike a reflexive behavior.

Motivation: Behaviors are not governed only by environmental stimuli but also by the internal state of the animal (e.g. appetite). Motivation is a function of the interaction between internal and external factors (sources or stimuli). Internal conditions and external objects are both required for motivated behavior to appear. I.e. the incentive-value of an external object (e.g., a cookie) depends on internal conditions (e.g., need for calories).

Needs: Needs are internal physiological and psychological conditions that promote instigating drives. Needs can arise either from deprivation of internal body conditions essential to life (e.g., water or food) or to excessive stimulation. Physiological *need states*, *do produce corresponding drive states,* which motivate behavior and thereby assist animals and humans in maintaining homeostasis. An example of the *need => motivation => drive => behavior => inhibition feedback* loop of an animal is shown on the next scheme (Fig. 1).

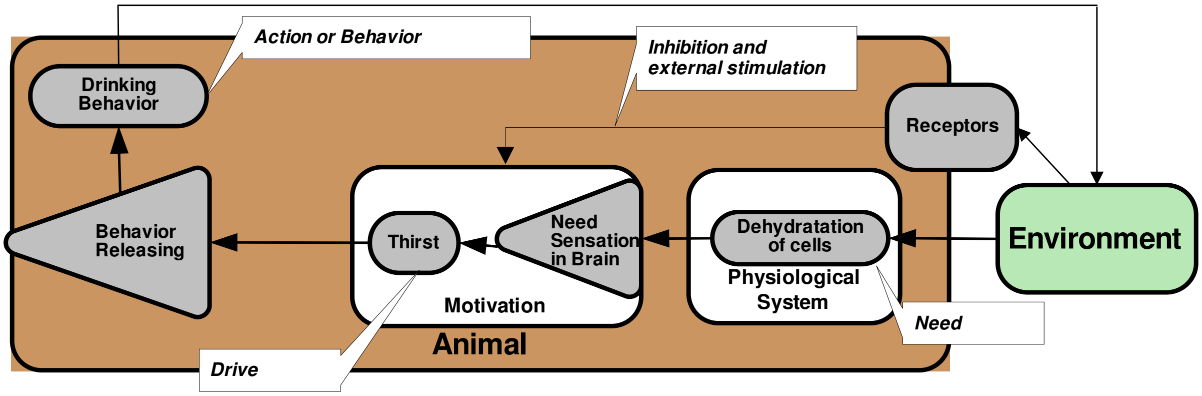


Figure 1: Schema of the need => motivation => drive => behavior => inhibition feedback loop of an animal.

DESIGNED APPROACH

We have chosen the hybrid paradigm to agent architecture design. The system agent-environment works in self-regulation and self-optimization/learning loop. The agent receives feedback on its behavior from the environment. This environment feedback modifies agent's internal state that is together with agent's internal dynamics contained in the physiological system. Both the environment and agent respect physical laws and constraints. These physical laws and constraints are embedded in the physiological system also. The learning and control problem is represented as Semi Markov Decision Process (SMDP) or as multiple SMDPs. The agent learns optimal mapping of actions (or behaviors) to situations for each SMDP.

The framework of Reinforcement Learning (RL) is used for learning. Reinforcement Learning requires a priory mapping of rewards to situations. This mapping needs to be provided by the designer upfront. The common problem of this approach is that the designer brings in subjective experience and usually cannot define optimal mapping respecting both the agent internal model and properties of the environment. I have changed this concept in the sense that the agent evaluates rewards (action utilities) by itself. These rewards are derived from the transition in the agent's physiological space. If the action (or behavior) moves the agent towards its optimal conditions then the reward is positive and negative for reverse direction. Rewards are therefore aligned to real behavior utilities. This approach overcomes the problem of imbalanced and subjective mapping done by human designers. Main optimization criterion that is used during learning and control corresponds to agent's distance to the optimal conditions.

The agent decomposes the root SMDP to a hierarchy of multiple simpler SMDPs to simplify state space and speed up learning convergence. The decomposition leads to multiple SMDPs dedicated to specific behaviors from simple reactive ones to complex composite ones for planning. Motivation block then prioritizes those SMDPs that contain behaviors leading to correction of the agent's internal variables having the longest distance from optimal conditions. In the contrary to the conventional hierarchical reinforcement learning methods where the human designer is required to do the decomposition, I developed concept that allows the agent to make this decomposition automatically.

From a functional point of view, the agent holds properties depicted in Figure 2. Constraining properties of the environment are shown in the same figure.

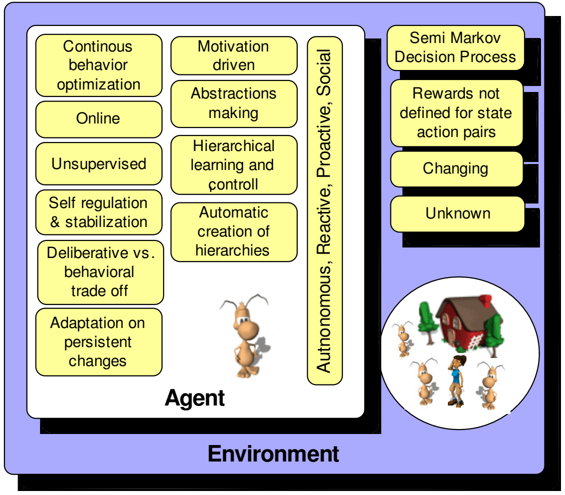

Continous behavior optimization
Online
Unsupervised
Self regulation & stabilization
Deliberative vs. behavioral trade off
Adaptation on persistent changes
Motivation driven
Abstractions making
Hierarchical learning and controll
Automatic creation of hierarchies
Autnonomous, Reactive, Proactive, Social
Agent
Semi Markov Decision Process
Rewards not defined for state action pairs
Changing
Unknown
Environment

Figure 2: Agent requirements and environment constraints

BEHAVIORAL ACCESS

Internal States

Agent properties and their dynamics (agent internal dynamics) is represented as a time invariant deterministic MIMO (Multiple Input-Multiple Output) system with the well known state-space model defined as:

$$\frac{dx}{dt} = f\left(x,u,t\right) \quad \text{and} \quad y = g\left(x,u,t\right) \qquad (1)$$

Where the state $x \in R^n$ and the controls are $u \in R^m$, $x \in X^A \subset R^n$, $u \in \Omega \subset R^m$ and $t_0 \le t \le t_f$.

X^A represents *agent state variables*. An agent's internal state space fulfills the role of the physiological space of animals, and internal state variables correspond to animal's physiological variables. Physiological regulation processes in animals are realized through dynamics of the internal state system. From now on, the term physiological space can be used instead of the term internal state system.

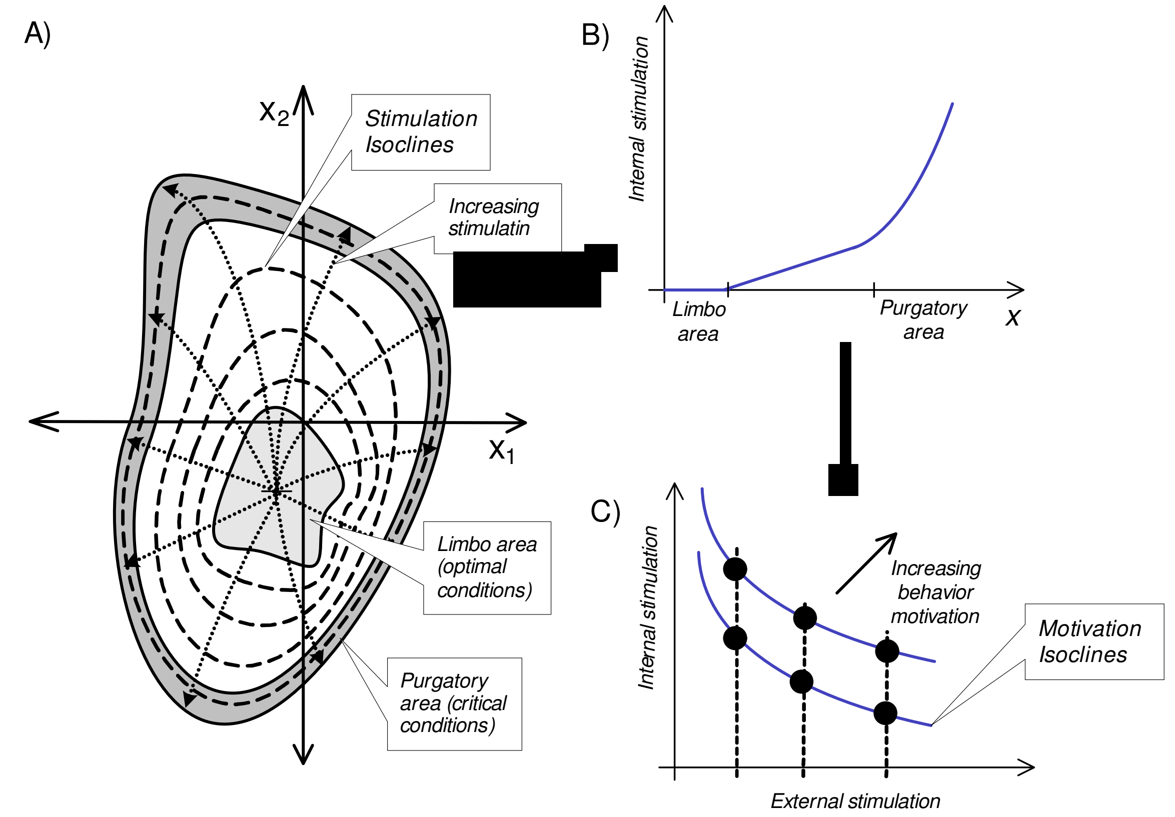


Figure 3: Physiological space of an animal is used correspondingly in presented agent architecture.

Behavior Motivation

Animal's state can be represented as a point in its physiological space. An example of a two variable physiological space (e.g.: x_1 ~ body temperature and x_2 ~ blood pressure) is depicted in Figure 3 A. The state variables are most likely constrained and are determined by the animal physical properties and its environment. The space is bounded by the tolerance limits for these variables. A typical physiological space (Figure 3 A) contains three regions: (i) *limbo* – animal's conditions are almost optimal, (ii) *purgatory* – animal's conditions are critical, and (iii) *normal* area between these two where animal's (agent's) motivation is to perform balancing and behavior motivation grows nearly linearly. As the state approaches the lethal boundary, the motivation to return the animal back to balanced conditions increases. The "balancing" behavior is released when the animal's physiological state is close to the lethal boundary (*internal stimuli*) and the corresponding external releaser appears (*external stimuli*).

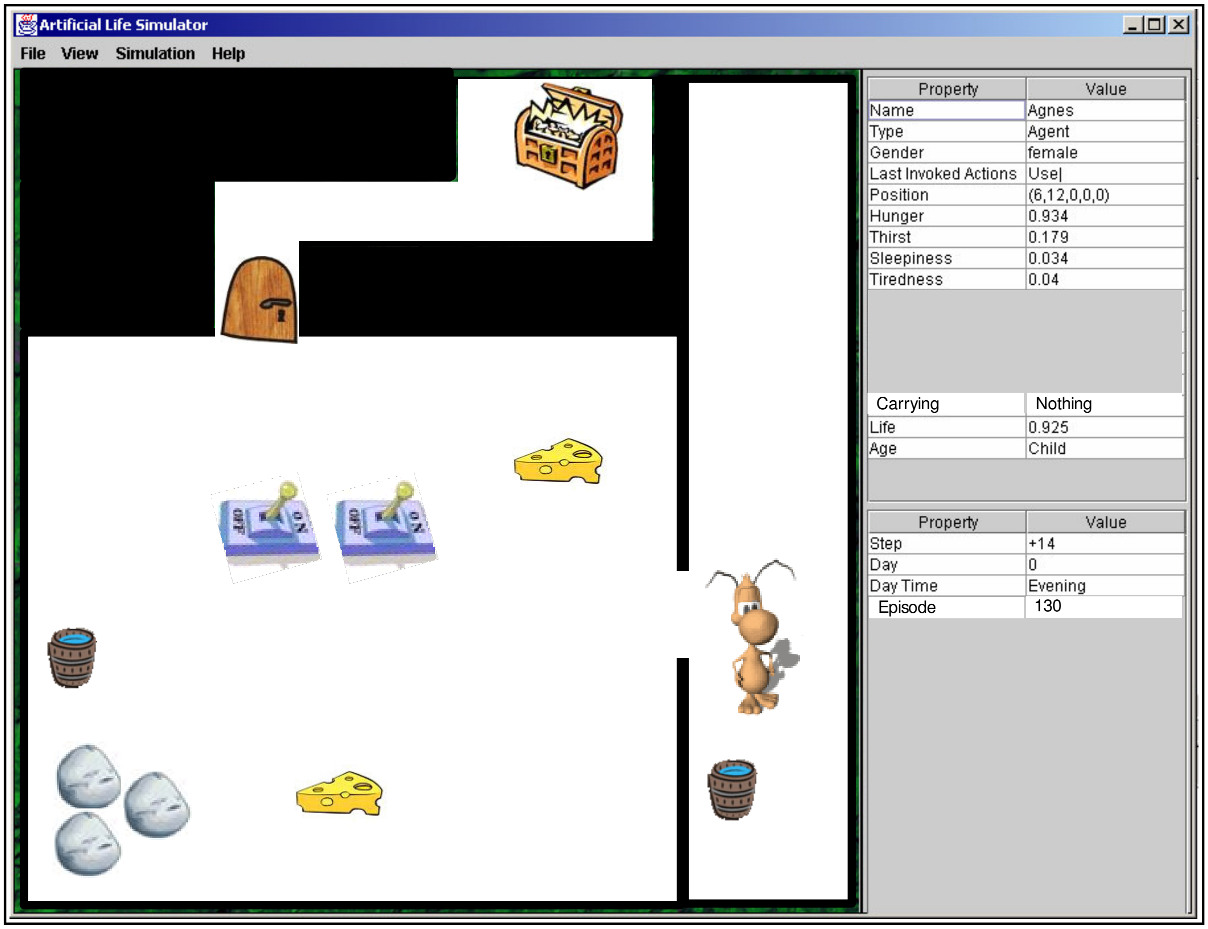
Motivation is a combination of:

- Internal stimulation that comes from the actual position in a physiological space
- External stimulation that comes from environment

Internal stimulations increase from the origin towards boundaries of the physiological space. *Stimulation isoclines* line up the same level of internal stimulation. Figure 3 shows how an internal stimulation derived from multiple state variables and combined with external stimulations. The resulting motivation becomes the primary source for triggering behavior.

Stimulation isoclines line up the same level of motivation. Part 3 B) shows how internal stimulation depends on the agent's position in the physiological space. Part 3 C) shows the resulting motivation that is derived from internal and external stimulation.

Motivation is according to McFarland (McFarland, 1981) a function of internal and external stimulation.



Property	Value
Name	Agnes
Type	Agent
Gender	female
Last Invoked Actions	Use|
Position	(6,12,0,0,0)
Hunger	0.934
Thirst	0.179
Sleepiness	0.034
Tiredness	0.04
Carrying	Nothing
Life	0.925
Age	Child

Property	Value
Step	+14
Day	0
Day Time	Evening
Episode	130

Figure 4: Screenshot captured during experiments.

$$m = m\left(st_{int}, st_{ext}\right) \qquad (2)$$

Where st_{int} is internal stimulation and st_{ext} is external stimulation. I represent stimulations as states of some selected internal and external state variables. Any motivation can result from multiple stimulations, thus the motivation function is a mapping from a set of internal and external variables to a real value:

$$m : X^A \cup X^E \to R \qquad (3)$$

where X^A refers to agent internal variables, and X^E environment variables.

EXPERIMENTS AND OBTAINED RESULTS

Functionality of the designed agent architecture has been assessed in multiple focused experiments. In this paper, we discuss only the experiment demonstrating automatic creation of the hierarchies, learning and performing behaviors of different complexity in this hierarchy. In the experiment "The treasury problem", an agent needs to get to a treasure. The best action graphs are provided for *eat*, *drink*, *drop* and *get to treasure* behaviors. See Figure 4 with Artificial Life Simulator. Agent pushes boulders onto buttons to open the locked door and reach the treasure. In parallel, the agent satisfies its internal needs for food and water.

Functionality of the designed agent architecture has been assessed in multiple focused experiments. In this paper, we discuss only the experiment demonstrating automatic creation of the hierarchies, learning and performing behaviors of different complexity in this hierarchy. In the experiment "The treasury problem", an agent needs to get to a treasure. The best action graphs are provided for *eat*, *drink*, *drop* and *get to treasure* behaviors. See Figure 4 with Artificial Life Simulator. Agent pushes boulders onto buttons to open the locked door and reach the treasure. In parallel, the agent satisfies its internal needs for food and water.

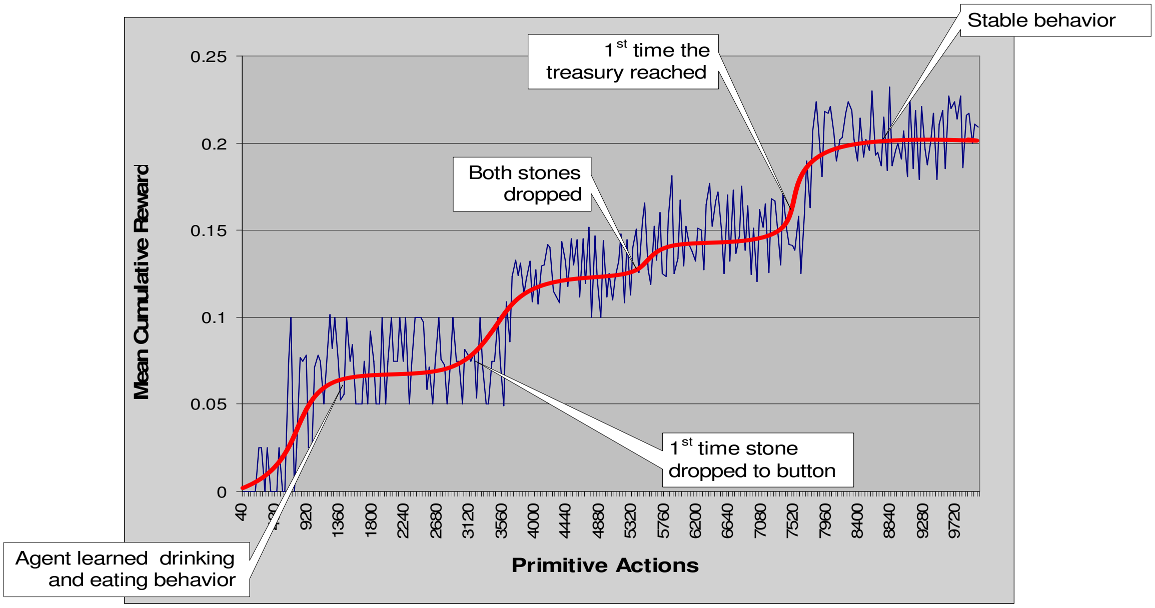


Figure 5. Graph of Mean Cumulative Reward

Note: The agent has no prior knowledge of the environment and no reward values are defined for the problem!

The agent must expose both planned behavior as well as primitive behaviors ensuring agent's survival. The resulting final hierarchy is created in about 8800 primitive actions and the agent is able to quickly find stones, carry them towards button, drop them onto the buttons, open the locked door and get to the treasure. At the same time, the agent can satisfy its needs for food and water. The graph (Figure 5) shows agent's learning convergence for "The treasury problem". The highest cumulative reward is obtained after the agent solves the "Treasury problem" and learns how to get to the treasure and satisfy its needs for water and food. The resulting graph (which shows agent's learning convergence) has staircase shape because of discrete points at which is received a new reinforcement type and the decision space is decomposed consequently.

The resulting hierarchy that the agent learns is depicted in another Figure 6. The agent has three internal variables: Food level, Water level and Obligation (used the express the need for fulfilling deliberative tasks), they produce three motivation functions: Food need, Water need and Fulfillment need.

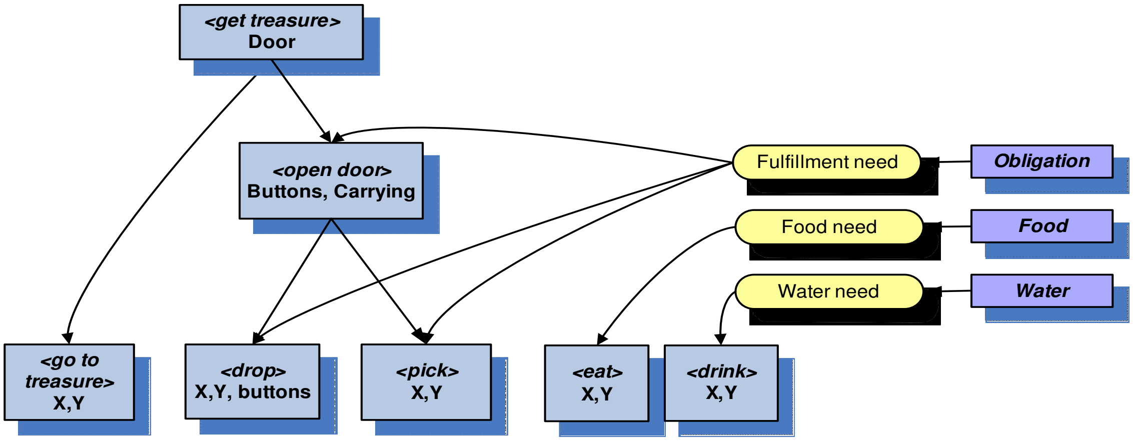


Figure 6: The decision space hierarchy automatically created for the Treasure problem

The root SMDP is automatically decomposed into a hierarchy of seven SMDPs as shown in the Figure 6.

Each SMDP is labeled as <Name> and a set of internal variables that are used for Q(lambda) learning in the concrete SMDP.

CONCLUSION

The main contributions of this research are seen in:
1. Showing how to integrate behavioral (bottom up) and deliberative (top down) approaches in a single agent.
2.Showing an approach how to bridge ethology/biology, agent technology, RL, and control engineering.
3. Showing that some concepts adopted from biology/ethology can be efficiently used as an alternative to classical approach.
4. Showing how to automatically decompose the root problem in a hierarchical RL as opposed to the conventional methods that need manual decomposition.
5. Showing how an agent can assess usability of its behavior and provide reward values to RL controller. This approach is different to the conventional RL methods where the rewards-situations mapping is defined by the designer upfront.
6. Presenting new method for evaluating optimality of agent behavior that is particularly useful in AL domain

ACKNOWLEDGMENT

This research has been funded by the Dept. of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague and by the Centre for Applied Cybernetics under Project 1M0567.

AUTHOR BIOGRAPHY

Pavel Nahodil obtained his scientific degree Ph.D. in Technical Cybernetics from the Czech Technical University in Prague, Czech Republic in 1980. Since 1986 he has been a Professor of Applied Robotics at the Department of Cybernetics at the Faculty of Electrical Engineering in Prague. His present professional interest includes artificial intelligence, multi-agent systems, intelligent robotics (control systems of humanoids) and artificial life approaches in general. He is the (co-) author of several books, university lecture notes, tens of scientific papers and some collections of scientific studies. He is also a conferences organizer + reviewer (IPC Member) and a member of many Editorial Boards.

REFERENCES

Balci, O. and R.G. Sargent. 1981. “A Methodology for Cost-Risk Analysis in the Statistical Validation of Simulation Models.” *Communications of the ACM* 24, No.4 (Apr), 190-197.

Barto, A. G. and Mahadevan, S. 2003,). Recent Advances in Hierarchical Reinforcement Learning Discrete Event Dynamic Systems vol. 13(4), pp 341 – 379.

Bertsekas,D.P., Tsitsiklis, J.N. (2007). Comment on Coordination of Groups of Mobile Autonomous Agents Using Nearest Neighbor Rules, Lab. For Information and Decision Systems Report, MIT, IEEE Trans. On Aut. Control, Vol. 52, pp. 968-969, 2007.

McFarland, D., and Bosser, U. (1993). Intelligent Behavior in Animals and Robots. MIT Press, Cambridge, MA.

Kadleček, D., (2008). *Motivation Driven Reinforcement Learning and Automatic Creation of Behavior.* Doctoral Thesis, Supervised by Nahodil, P., Dept. of Cybernetics, Czech Technical University in Prague, Prague, 143 pp.

Kadleček, D., Řehoř, D., Nahodil, P., Slavík, P. (2003). Analysis of Virtual Agent Communities by Means of AI Techniques and Visualization. In Intelligent Virtual Agents, 2003, Heidelberg: Springer Verlag, pp. 274-282.

Kadleček, D., Nahodil, P. (2001). New hybrid architecture in artificial life simulation. Advances in Artificial Life. In Lecture Notes in Artificial Intelligence No 2159, January 2001 Berlin: Springer Verlag. Pp. 143-146.

Luck et. al., 2003. Agent Technology: Enabling Next Generation Computing. A Roadmap for Agent Based Computation.

Nahodil P., Kadleček D. (2004). Animal Like Control of Intelligent Autonomous Agents. In: Proc. Of the 23rd IASTED International conf. on Modelling Identification, and Control. Zurich. Acta Press, pp. 154 -159.

Nahodil, P., Slavík, P., Řehoř, D., Kadleček, D (2004).: Dynamic Analysis of Agents’ Behaviour – Combining Alife, Visualization and AI. In *Engineering Societies in the Agents World IV.* Berlin: Springer, p. 346-359.

Řehoř, D. Kadleček, D. Slavík, P. Nahodil, P. (2003) Dynamic Analysis of ALife Systems Using Visualization and AI Techniques. 4th International Workshop on Engineering Societies in the Agents World, Imperial College London, UK, Springer Verlag. pp. 173-178.

Řehoř, D. Kadleček, D. Slavík, P. Nahodil, P. (2003). VAT - a New Approach to Multi-agent Systems Visualization. In Visualization, Imaging, and Image Processing, Anaheim, California. Anaheim: ACTA Press. pp. 849-854. vol. II.

Singh, S., Jaakkola, T. and C. Szepesvari (2000). *Convergence results for single step on policy reinforcement learning algorithms*. Machine Learning, 38:287-308.