IC13

Ethology-Inspired Advanced Problem Solving Mechanism

Jaroslav Vítků

Dept. of Cybernetics, Faculty of Electrical Engineering, CTU in Prague
Technická 2, 166 27 Prague, Czech Republic

vitkujar@ fel.cvut.cz

ABSTRACT

We built a new architecture of autonomous agent. Our agent is capable of fully autonomous learning. The course of learning is similar to behavior of a newly born animal. The agent learns everything by itself, from basic tasks towards the more complex knowledge. Knowledge is stored in the hierarchy of Reinforcement Learning (RL) modules. Each module represents and abstract action (composed behavior). The planning engine can chain these actions in order to solve even more complex task. The main outcome of this work is new hybrid domain independent planner.

AIMS OF RESEARCH

Goal was to create an autonomous problem solving system (agent) with:

- Autonomous adaptation on a given domain – nothing pre-programmed
- Learn and use this knowledge for further learning and survival
- Ability to deal with the complex domains
- Hybrid approach - from reactive towards the deliberative decisions
- Autonomous system - but be able to specify goals if needed

HIERARCHICAL ARCHITECTURE

Reinforcement Learning: if the agent understands that his action was useful reinforces itself – it corresponds to some decision space (Fig.1). This system, based on [3], is able to **autonomously build the hierarchy of RL engines** based on the interaction with the environment. It supports online autonomous learning in unknown domains. Hierarchical approach enables agent to learn very complex tasks – system learns which variables are important and when.

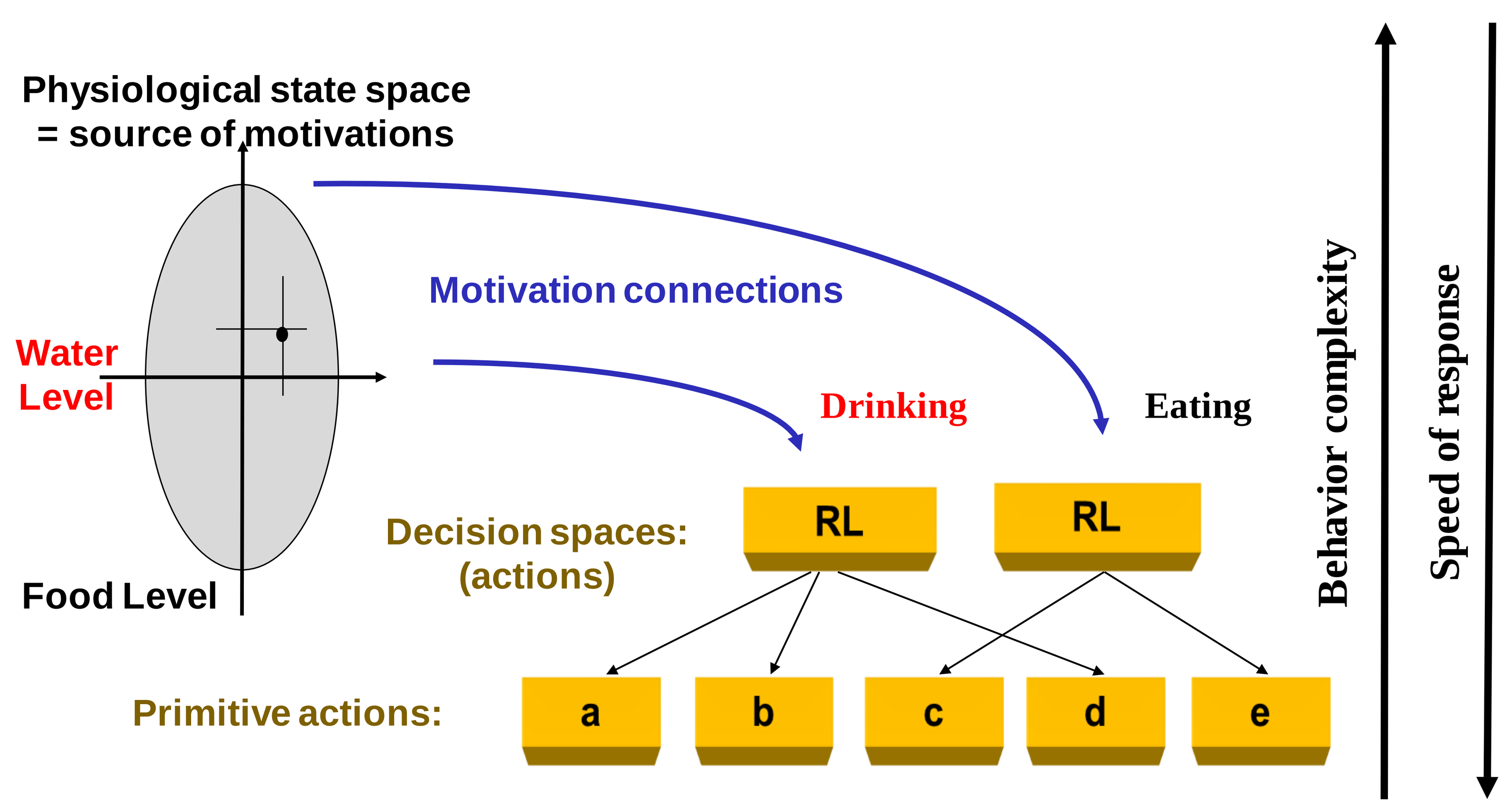


Fig.1 Hierarchical RL (HARM) – To understand what is good and learn it

HYBRID HIERARCHICAL SYSTEM

Now, as the agent knows some behaviors (actions), is able to autonomously **discover the preconditions and effects** of these actions (Fig.2). Therefore these actions can be represented as primitive actions in the language for planner [1], e.g. Stanford Research Institute Problem Solver (STRIPS) language. This enables our agent to deliberatively "think" about these actions and to create complex plans. The course of work of this system is as follows:

1. Autonomously learn some behaviors (hierarchy of RL engines)
2. Translate these actions for planner
3. Set/receive the goal
4. Construct the plan in order to fulfill the goal
5. Execute the plan (Fig.3)

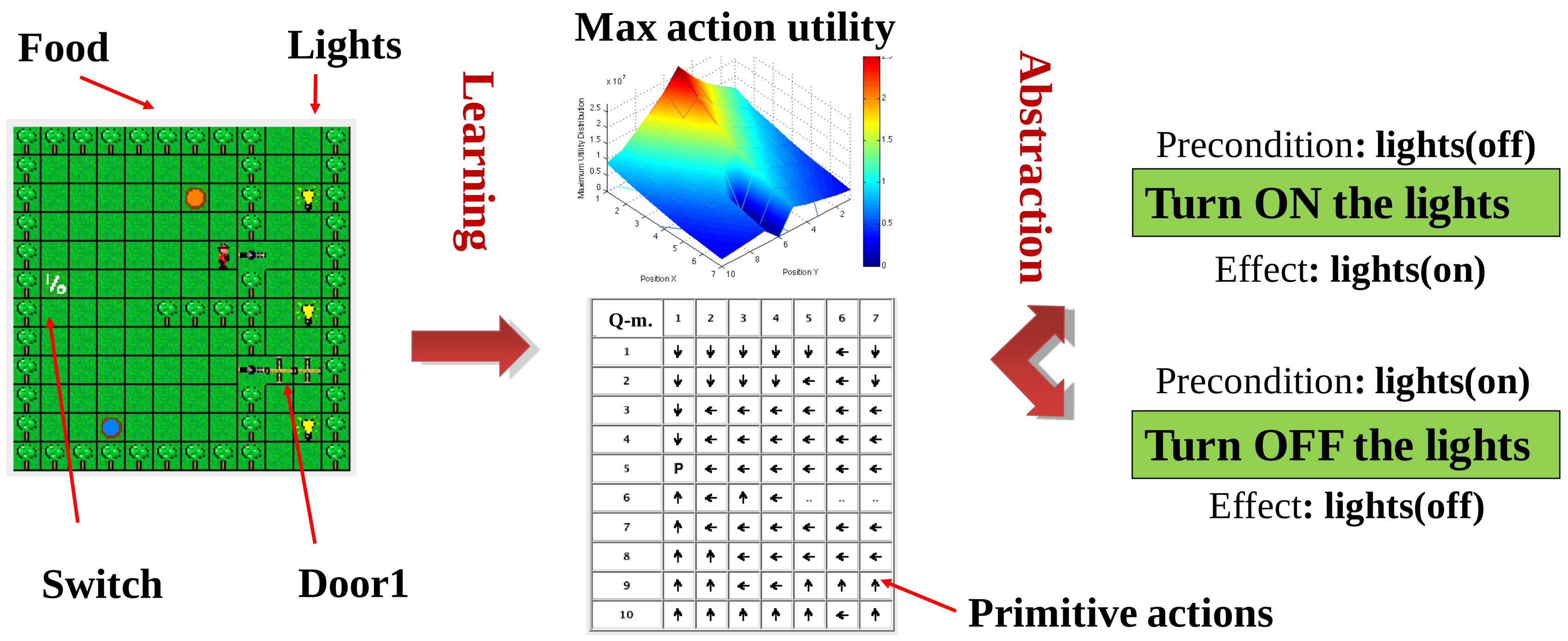
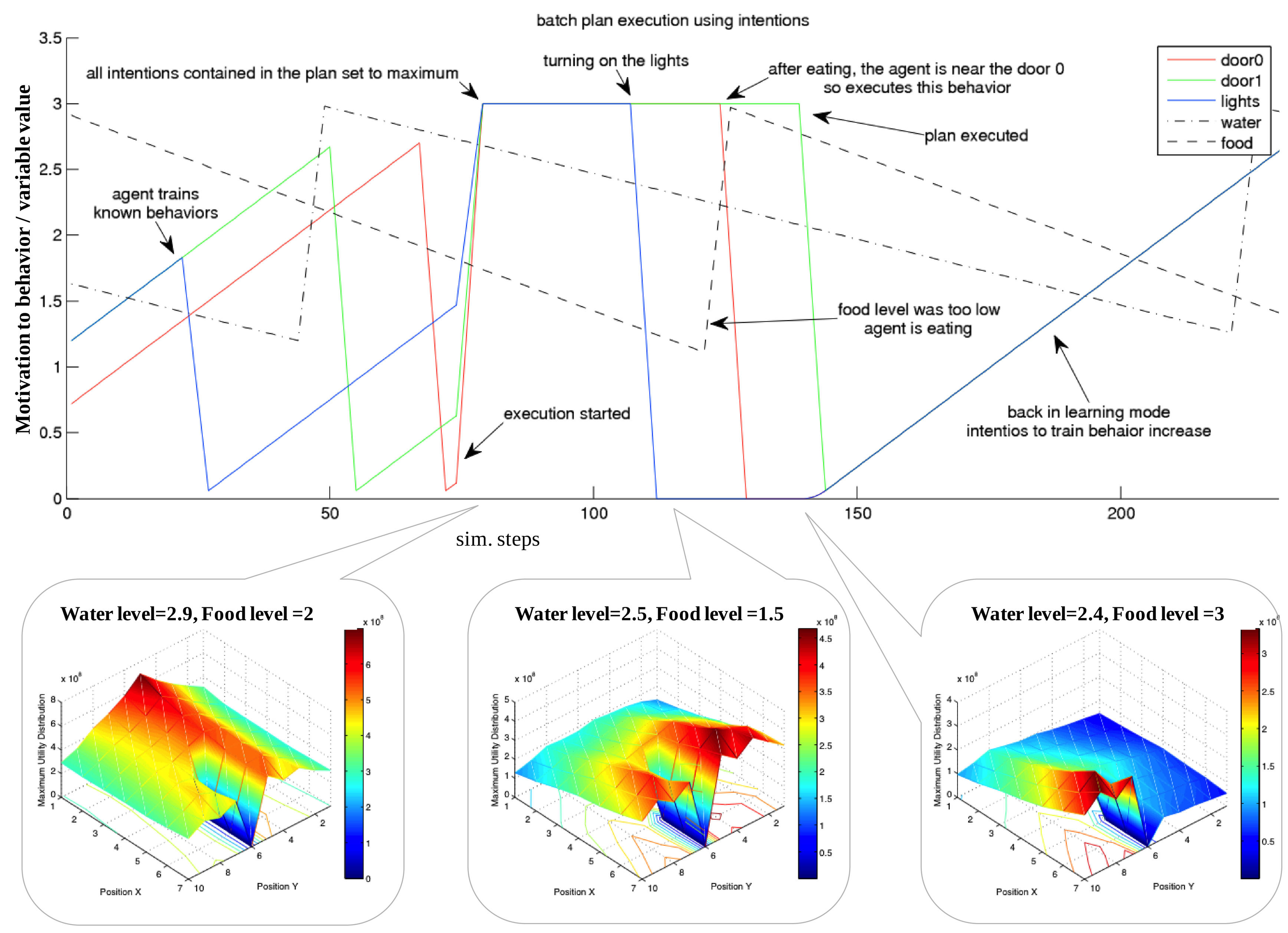


Fig.2 Abstraction of RL to STRIPS language – lights control

PRINCIPLE OF FUNCTION



(a) *Turning on the lights.* (b) *Lights turned on-small hunger.* (c) *One subtask remains.*

Fig.3 Plan execution based on composed action utility, greedy selection:
Turn on the lights - Open the door0 (eat) - Open the door1

CONCLUSION

The benefits against other approaches are: RL itself cannot generate such a complex behavior, classical planner cannot operate in complex domains, hierarchical planners need a priori knowledge about the problem structure. This solver autonomously learns structure of the problem (using the RL hierarchy) and is able to use this knowledge for planning. Because of hierarchical organization of knowledge is our agent able to operate even in very complex domains.

REFERENCES

[1] Vítků, J.: *An Artificial Creature Capable of Learning from Experience in Order to Fulfill More Complex Tasks*, Czech Technical University in Prague, FEE, Diploma thesis, (2011)

[2] Nahodil, P. and Kadleček, D. (2008), *Adopting Animal Concepts in Hierarchical Reinforcement Learning and Control of Intelligent Agents*, In Proc. of 2nd IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, BioRob 2008, Scottsdale, U.S.A, 2008, 122-131, ISBN: 978-1-4244-2882-3.

[3] Kadleček, D.: *Motivation Driven Reinforcement Learning and Automatic Creation of Behavior Hierarchies*, Czech Technical University in Prague, FEE, PhD thesis, (2008)