

Learning Life Cycle in Autonomous Intelligent Systems

Jorge Ierache¹, Ramón García-Martínez² and Armando De Giusti³

Abstract Autonomous Intelligent Systems (AIS) integrate planning, learning, and execution in a closed loop, showing an autonomous intelligent behavior. A Learning Life Cycle (LLC) for AISs is proposed. The LLC is based on three different learned operators' layers: Built-In Operators, Trained Base Operators and World Interaction Operators. The extension of the original architecture to support the new type of operators is presented.

1 Introduction

The autonomous intelligent systems (AIS) evolve from initial theories (set of operators built in by the AIS's programmer) to ones learned from interaction with the environment or other. Given unknown environments, real autonomous systems must generate theories of how their environment reacts to their actions, and how the actions affect the environment. Usually, these learned theories are partial, incomplete and incorrect, but they can be used to plan, to further modify those theories, or to create new ones. Previous work on machine learning applied to problem solving has mainly focused on learning knowledge whose goal was to improve the efficiency of the problem solving task [1].

There is also a current interest in learning state transition probabilities in the context of reinforcement learning. However, few researchers have approached the generalized operators acquisition problem, described as techniques for automatically acquiring generalized descriptions of a domain theory. This issue is

¹ Jorge Ierache

PhD Program. Computer Sc. School. UNLP. Intelligent Systems Lab. FI-UBA. jierache@fi.uba.ar

² Ramon Garcia-Martinez

Software and Knowledge Engineering Center (CAPIS), Buenos Aires Institute of Technology

³ Armando De Giusti

Instituto de Investigación en Informática LIDI, Facultad de Informática, UNLP

crucial when dealing with systems that must autonomously adapt to an unknown and dynamic environment. LOPE (Learning by Observation in Planning Environments) is an AIS implemented architecture that integrates planning, learning, and execution in a closed loop, showing an autonomous intelligent behavior [2].

Learning planning operators (what we will call operators, is also referred to as action models within the reinforcement learning community) is achieved by observing the consequences of executing planned actions in the environment. In order to speed up the convergence, heuristic generalizations of the observations have been used. Also, probability distribution estimators have been introduced to handle the contradictions among the generated planning operators [3] and how sharing among AISs the learned operators improve their behavior [4].

As the natural next step, in this paper we recover the AIS general description (section 2), we propose the AIS learning life cycle for a community of AISs that shares knowledge (section 3) and preliminary conclusions and future research are drawn (section 4).

2 General description of AIS

One of the main objectives of each LOPE agent is to autonomously learn operators (action models) that predict the effects of actions in the environment by observing the consequences of those actions. In order to learn those descriptions, it is able to plan for achieving self-proposed goals, execute the plans, find out incorrect or correct behavior, and learn from the interaction with the environment and other agents. Each agent receives perceptions from the environment, called situations, applies actions, and learns from its interaction with the outside world (environment and other agents). At the beginning, the agent perceives the initial situation, and selects a random action to execute in the environment. Then, it loops by executing an action, perceiving the resulting situation and utility of the situation, learning from observing the effect of applying the action in the environment, and planning for further interactions with the environment when the previous plan has finished its execution, or the system observes a mismatch between the predicted situation by the AIS's operators and the situation it perceived from the environment. The planner basically does a backward chaining search from the initial situation (goal) of the operator with the highest utility in order to find a sequence of operators that will lead from the current state to that goal. If it succeeds, and the probability of its success is greater than a given bound, it executes the plan. If not, it selects the next highest utility operator and searches for a plan. This process loops until it finds a plan for any high utility operator. More details on how the planner works can be found in [3].

In this context, a learned operator O in LOPE [3] is a tuple $\langle C, A, F, P, K, U \rangle$ where: C is the initial situation (conditions), A action to be performed, F final situation (post-conditions), P times that the operator O_i was successfully applied

(the expected final situation F was obtained), K times that the action A was applied to C, U utility level reached applying the action to the initial situation C of the operator.

3. Proposed AISs Learning Life Cycle

Based on the LOPE Architecture, an AIS Learning Life Cycle with three learning layers is presented: [a] Layer BI (Built-In Operators) is the learning layer where the operators are implanted in the LOPE AIS by the AIS programmer, [b] layer TB (Trained Base Operators) is the learning layer where the operators are learned by training (previously designed by AIS programmer and evolutionary learning techniques) and [c] layer WI (World Interaction Operators) is the learning layer where the operators are learned by interaction with the part of the world that performs the environment of the AIS. The proposed learning life cycle is shown in figure 1.

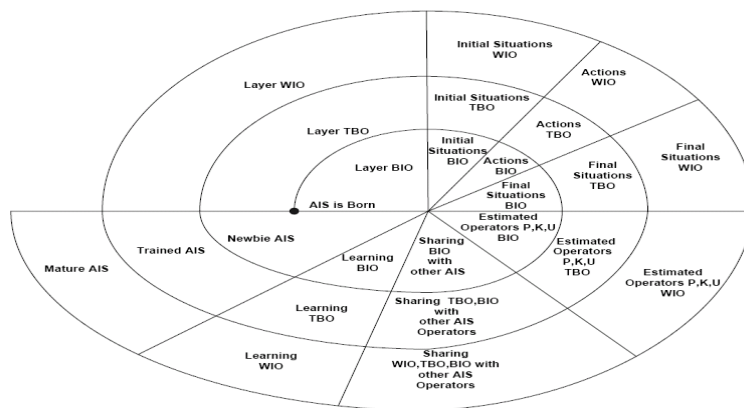


Fig. 1. AIS Learning Life Cycle (LLC)

Although knowledge sources are different, the sensor system and the learned operator structure is always the same. The AIS "born" with the implanted built in operators by its programmer. These operators represent the basal knowledge that allows an initial reactive behavior of the AIS. The operators learned by training facilitate the evolution of the knowledge using a reinforcement mechanism of good operators and the "punish" of bad functioning operators. The terms punish and reward have been borrowed from the field of biological reinforcement rather than from reinforcement learning. The heuristic-generalization algorithm generates a set of new operators according to the generalization heuristics, which are incorporated into the set of planning operators. Since the number of operators that are created can potentially slow down the performance of the learning and planning modules, the system forgets operators with a very low quotient P/K. One

of the main objectives of each LOPE-LLC (LOPE Learning Life Cycle) AIS is to autonomously learn operators (action models) that predict the effects of actions in the environment by observing the consequences of those actions. In order to learn those descriptions, it is able to plan for achieving self-proposed goals, executes the plans, find out incorrect or correct behavior, and learn from initial BI operators, reinforced by TB operators creation and evolved by knowledge sharing based WI operators from the interaction with the environment and other AISs. In TB operators learning each AIS receives perceptions from the environment, called situations, applies actions, and learns from its interaction in the designed training. The AIS perceives the initial situation based in its BI operators, and selects a random action to execute in the environment from its TB operators set. In the WI operators learning process use the BI operators and TB operators. Based on the three layers (BIO, BTO, WIO) of the LLC proposed, the AIS evolves by going around the stages, born, newbie, trained, mature. Each layer includes the following activities: [a] Initial situation of the world (environment and AIS's), [b] Actions based on the AIS operators according to their plans, [c] Foreseen final Situation, [d] Estimate of the AIS operators, [e] Operators sharing with other AISs and [f] AIS learning (regularly), [g] Evolution of the AIS into a new stage. When an AIS is born (initial stage), is provided with programmer's built-in operators. The BIO layer moves around this base, sharing its BI operators with other AIS's, learns and reaches the Newbie AIS stage and then it goes around the BTO layer learning through the training and sharing of the TB and BI operators which allow them to reach the Trained AIS stage, finally it goes around the WIO layer, capable to share their BI, BT, WI operators with the rest of the AIS's reaching the mature AIS stage. Figure 2 shows a schematic view of the LOPE architecture extended on the base of the LLC proposed, which is called LOPE-LLC.

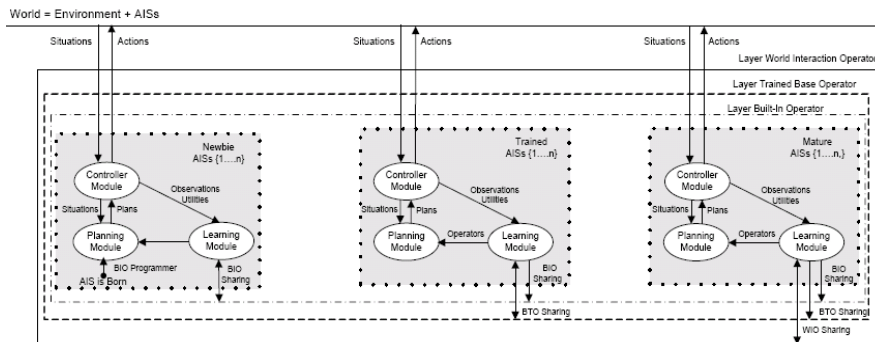


Fig. 2. Architecture of a group of LOPE –LLC AISs

LOPE-LLC AISs can be seen sharing its operators through the different layers according to their stage (born, newbie, trained, mature). Each of the AISs receives as input: situations (perceptions) from the world (environment and AISs); set of actions it can perform and Operators. The output of each AIS is a sequence of over time actions (for the environment) and regularly, the set of operators it learned

from the sharing with other AISs, according to the stage it reaches within the LLC layers proposed.

4 Preliminary Conclusions and Future Research

In this paper, we have presented a learning life cycle for autonomous intelligent systems based on three types of learned operators: built in operators, trained based operators and world interaction; and how these operators evolves in an architecture that learns a model of its environment by observing the effects of performing actions on it. The LOPE-LLC AISs autonomously interact with their environment and with other AISs with the objective of learning operators based on a proposed learning life cycle that predict, with a given probability estimator, the resulting situation of applying an action to another situation.

With respect to the scalability of the approach, we are now performing experiments in a much more complex, noisy, with hidden states, and many AIS domain, such as the robosoccer. The performance world is composed by the Environment (soccer field, ball) and the players of both teams (AIS's), programmed with operators to play the different roles (forward line players, midfield players, defenders, goalkeeper). BI operators of the player (AIS), resulting from the birth of this one thanks to the programmer's action, evolve while sharing with other players (AIS's) of their team or other roboccer teams, allowing the player (AIS) to reach the Newbie stage. BT operators (previously designed by AIS programmer and evolutionary learning techniques) make easier the (AIS) player's evolution into the trained stage, after finishing the activities of the BTO layer of the LLC. The trained player (AIS) through the WI operators sharing with other players (AIS's of its team or another team) reaches the mature player (AIS) stage once the activities of the WIO layer of LLC are finished. We believe that through the use of the probabilities estimations, and the heuristic generalization of operators, we will be able to cope with the complexity of that domain.

References

1. Fritz, W., García-Martínez, R., Blanqué, J., Rama, A., Adobbati, R., Sarno, M. *The autonomous intelligent system*. Robotics and Autonomous Systems, 5, 109-125. (1989).
2. García-Martínez, R. and Borrajo, D. *Planning, learning, and executing in autonomous systems*. Lecture Notes in Artificial Intelligence, 1348, 208-220. (1997).
3. García Martínez, R. y Borrajo, D. *An Integrated Approach of Learning, Planning and Executing*. Journal of Intelligent and Robotic Systems, 29, 47-78. (2000).
4. García-Martínez, R., Borrajo, D., Britos, P., Maceri, P. *Learning by Knowledge Sharing in Autonomous Intelligent Systems*. Lecture Notes in Artificial Intelligence, 4140, 128-137. (2006).