

Learning by Collaboration in Intelligent Autonomous Systems

Jorge Ierache, Ramón García-Martínez, and Armando De Giusti

Computer Science PhD Program. Computer Sc. School. La Plata National University
Institute of intelligent systems and robotics experimental teaching, FICCTE, UM
Intelligent Systems Laboratory. Engineering School. Buenos Aires University
Institute for Research on Information Technology- LIDI, School of Information Technology,
UNLP
jierache@unimoron.edu.ar

Abstract. Very few learning systems applied to problem solving have focused on learning operator definitions from the interaction with a completely unknown environment. Autonomous Intelligent Systems (AIS) deal with that issue by means of architectures where learning is achieved by establishing plans, executing those plans in the environment, analyzing the results of the execution, and combining new evidence with prior evidence. This paper proposes a selective mechanism of learning allowing an AIS to learn new operators by receiving them from another AIS in a higher stage in the Learning Life Cycle (LLC) with more cycles of interaction in the environment. The proposed collaboration mechanism also considers how to deal with theory ponderation (operators ponderation) and how to include the new operators (provided for) in the set of theories of the receiver AIS. The experimental results show how using collaboration-based learning among AIS provides a better percentage of successful plans, plus an improved convergence rate, than the individual AIS alone.

1 Introduction

We are presenting in this work the results achieved from the application of collaboration from the autonomous intelligent systems (AIS) of robots that are in a higher stage in the learning life cycle of an AIS [1]. 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 AISs. 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 can be used to plan, to further modify those theories, or to create new ones. The 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 [2]; [3]. There is also a current interest in learning state transition probabilities in the context of reinforcement learning. [4] However, few researchers have approached the generalized operator acquisition problem [5], [6], described as techniques for automatically

acquiring generalized descriptions of a domain theory. This issue is 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 [7]. Learning planning operators (what we will call *operators* is also referred to as action models within the reinforcement learning community) are 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. Moreover, probability distribution estimators have been introduced to handle the contradictions among the generated planning operators [8] and how by sharing among AISs the learned operators improve their behavior [9]. In this context, a learned operator O (theory) in LOPE [8] is a tuple $\langle C, A, F, P, K, U \rangle$ where: C is the initial situation (conditions), A is the action to be performed, F is the final situation (post-conditions), P means the times that the operator O_i was successfully applied (the expected final situation F was obtained), K means the times that the action A was applied to C , U means the utility level reached by applying the action to the initial situation C of the operator.

As the natural next step this paper aims to propose the general description of the collaboration model within the framework of the Learning Life Cycle (section 2), to describe the experimentation design and to analyze the results attained (section 3) and to draw some conclusions and future research lines (section 4).

2 General Description of the Collaboration Mechanism

In this section it is presented the description of the collaboration method between an AIS that has reached a higher evolutionary stage and a receiver AIS that is in a lower stage of evolution. The mechanism proposed considers theory ponderation and the inclusion of new theories in the knowledge base of the receiver AIS, provided by the collaborator AIS. Figure 1 shows a diagram of the model oriented to collaboration, on the basis of the LLC (Learning Life Cycle) proposed by the authors [1]. In the framework of the LLC of an AIS, three learning layers are presented: [a] Layer BIO (Built-In Operators) is the layer where the operators are implanted into the “Born” AIS by their creator, [b] Layer TBO (Trained Base Operations) is the one where the operators are learned by the “Newbie” AIS, in the simulation scenario (previously designed) and by evolutionary learning techniques, [c] Layer WIO (World Interaction Operators) is the one where the operators are learned by interaction of the “Trained” AIS with the part of the world representing their operation environment and with the presence of other AISs to turn into a “Mature” AIS.

The objective of the AIS is to autonomously learn operators (action models) that may predict the effects of the actions on the environment, through the observation of the consequences of these actions and to advance in their evolution state (Born, Newbie, Trained, Mature).

The collaboration allows that an AIS may receive knowledge that has been previously acquired by an AIS that is in a higher evolutionary stage within the framework

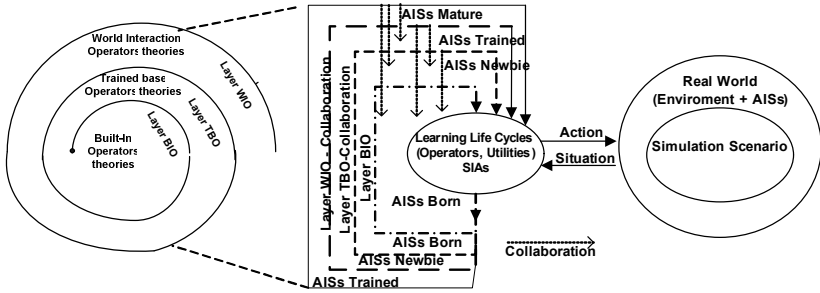


Fig. 1. Collaboration model proposed

of the LLC (Learning Life Cycle). Figure 1 shows the possible levels of collaboration: [a] a Mature AIS that ended layer WIO with a Trained AIS that is at the beginning of layer TBO, or with a Born AIS at the end of layer BIO; [b] a Trained AIS that ended layer TBO with a Newbie AIS that ended layer BIO or a Born AIS starting layer BIO; [c] a Newbie AIS that ended layer BIO with a Born AIS that is starting layer BIO.

When an AIS collaborates, it shares the experience gained, which is the product of both right and wrong performance. Therefore, by following the theory model and its ponderation method, when collaborating with operators it is necessary to reflect this knowledge in the receiver AIS, by maintaining the consistency of its set of theories. A theory (operator) is represented by T: Initial Situation (IS), Action (A), Final Situation (FS), P,K,U. [8]; where P means the quantity of times a theory T_1 was successfully used (the predicted effects were attained) and K means the quantity of times the theory T_1 was used. Finally, U means the level of utility reached when applying the action A to the initial situation (IS) of the theory T_1 . The method employed provides for the concept that the theory successfully used must be reinforced by increasing its P and its K and that the theory unsuccessfully used must be weakened by increasing its K, without increasing its P [8].

In the collaboration process, if a theory exists in the receiver AIS, it is reinforced by adding up its P and K; on the other hand, if a theory of the collaborator AIS does not exist in the receiver AIS, but a similar theory does exist in the latter, this theory weakens, maintaining the same P and adding up the K of the collaborator AIS to its K. Furthermore, the theories of the collaborator AIS that are not in the receiver AIS or are not similar or equal do not show any change, and are only transferred to the receiver AIS with their P and K values. To maintain the soundness of the method of theory ponderation, it is verified that after the collaboration the addition of the Ps of the similar theories in the receiver AIS is equal to the K of any of them.

The collaboration method algorithm is detailed in Pseudo-code 1. The algorithm works in the following way: for each Theory of the Receiver AIS (RA), (1) it tries to find if there exists a similar theory in the set of Theories of the Collaborator AIS (CA); (1.1.) If this is the case, the Ks of similar theories are added up, (1.1.1). Then, if besides the existence of a similar theory there exists an equal theory in the Set of Theories of the Collaborator AIS (SetCA), the Ps of the equal theories are added up. (1.1.2). After repeating this process for each theory existing in the set of Theories of

the Receiver AIS (SetRA), the theories of the set of the collaborator AIS (SetAC) that have not equal theories in the set SetRA are processed.

(2). If a theory of the SetCA has a similar theory in SetRA, (2.1), it is registered in this one with the P of the theory of SetCA and the K of the similar theory of SetRA. If it does not exist a similar theory, it is registered in SetRA, without modifying the P and the K.

Pseudo-code .1 “Collaboration Algorithm”

```

INPUTS: Set Theories AIS receiver (SetAR), Set Theories collaboration
AIS (SetAC).
OUTPUT: Set Theories AIS receiver SetAR
1. FOR EACH THEORY Ti in SetAR
  BEGIN
    1.1 IF exists TJ which belongs to SetAC Similar to TI
    of SetAR THEN
      1.1.1 KTI=KTI+KTJ
      1.1.2 IF exists TJ which belongs to
      SetAC equal to TI of SetAR THEN
        PTI=PTI+PTJ
    END.
  2. FOR EACH THEORY TJ in SetAC
  IF not exists TJ which belongs to SetAC equal to TI which belongs
  to SetAR THEN.
  BEGIN
    2.1 IF exists TI which belongs to SetAR Similar to TJ THEN
    Register TJ with P=PTJ and KTI in SetAR
  ELSE
    Register TJ with P=PTJ and K =KTJ in SetAR
  END.

```

3 Experimentation and Result Analysis

In this section we are presenting the design of the experiments, the results attained and their analysis. Several experiments have been carried out on the basis of the e-puck robot, by using the Cyberbotics' Webots 6 mobile robot simulator developed in the Laboratory of Micro-Informatics of the Swiss Federal Institute of Technology, Lausanne, Switzerland (EPFL). Figure 2 shows the experimentation scenario and the AIS (e-puck robots). The labyrinth-shaped setting is composed of four joined walls that surround the external limits of the setting and internally of a three-cornered labyrinth. In turn, at the end of the labyrinth it appears the type of block obstacle (besides the wall-type).

Two types of utility functions have been applied. The first one was based on the situation perceived by the robot from the information on the distance to the obstacles given by its sensors; considering that the range of each sensor is [0 to 1024], the theory applied by the robot has more “U” utility, the farther the wall or the obstacle is; the nearer to zero (0) the values of its sensors are, the farther from the obstacle the robot will be. Finally, it was transformed the final utility of the theory into the value range [-1 a +1], the higher the value obtained by adding up the values of each sensor,

the nearer to -1 the utility will be. So as to count the successful theories, it was adopted as a criterion the “U” utility value range: $0.75 < U \leq 1$.

The second utility function was based on the action carried out by the robot (speeds applied to each wheel of the robot); the utility was determined on the basis of the speeds calculated by applying Braitenberg quotients [10], compared to the speeds indicated by the action of the theory applied by the AIS for its left and right wheels. The nearer the action carried out by the robot is with regard to the one calculated with the quotients, the higher the utility of the action taken by the AIS (nearer 1) will be; the farther the action is, the lower utility (near -1) it will be. To count the successful theories, it was adopted as a criterion the “U” utility value range: $0.75 < U \leq 1$.

The AIDs constituted by robot autonomous Systems (RASs), during the experimentation, have formulated a set of theories that allow them an “a priori” prediction of the effects of their actions. Upon this experimentation basis, the exchange (cooperation) was performed between Newbie AISs that are in the layer BIO and the collaboration from a Trained AIS that went around the layer TBO to a Newbie receiver AIS located in the LLC (Learning Life Cycle) layer BIO.

For the experimentation, it was applied the knowledge base of theories of three AISs that are represented by: the robot A (born -with 600 simulation cycles, which is starting layer BIO), the robot B (born- with 3000 simulation cycles, which is going around layer BIO), the robot B (newbie- with 6000 simulation cycles, which is finishing layer BIO). On the basis of said theories generated, it was implemented the exchange at the ‘born’ stage in layer BIO, between robot A (born) and robot B (born), the collaboration of robot B (newbie) that is starting layer TBO towards the receiver robot A (born) which is in layer BIO.

The stated theory bases were used by robot E (experimentation robot), on which the experiments with the application of the methods of plan ponderation, mutation and a combination of both were conducted. The neuter AIS was developed on the basis of a born robot that got 600 cycles (robot A). The robots A and B started with their creator (programmer), getting 400 cycles to generate their initial theory base through an initial reactive behavior. The theory bases of the different robots were stored in an XML format.

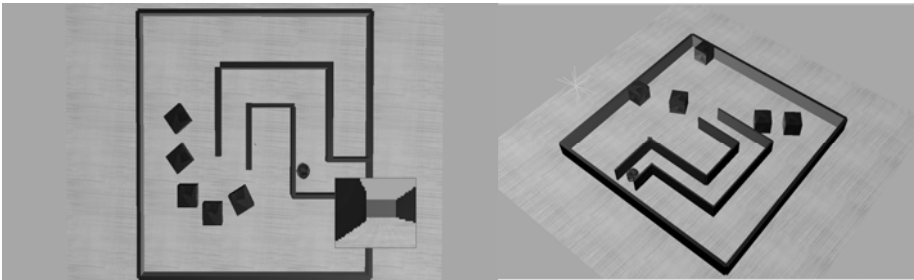


Fig. 2. Experimentation setting of the E-puck AIS

3.1 Experiment Design

The procedure applied in each experiment was performed on the basis of the simulation of the AIS robot for a number of 600 cycles (cycle = perception/action), considering the

starting position of the AIS and the selection of the environment at random. Every 20 cycles, it was generated a report on the status of the AIS variables, as shown in table 1 as an example, thus creating the row I of the matrix containing the variables of the AIS, which was generated in the cycle I*20. The general experimental procedure considers fifty experiments for each case, from which an average matrix resulting from the matrices corresponding to each experimental stage is constructed. The generation of graphs of the experiments presented is carried out on the basis of the information of each average matrix and the interpretation of the results achieved.

Table 1. Structure of the Report on the status of the AIS variables

| Cycles | Situations | New Theories | Theories [0,75 <U < 1] | Quantity of Successful Plans | Quantity of Successful Cycles | % Successful Cycles |
|--------|------------|--------------|------------------------|------------------------------|-------------------------------|---------------------|
| 1 | 1 | 0 | 56 | 0 | 0 | 0 |
| 20 | 2 | 0 | 56 | 8 | 19 | 95 |
| .. | .. | .. | .. | .. | .. | .. |
| 600 | 116 | 472,33 | 128,33 | 459,33 | 193,33 | 32 |

Independent Variables: The following independent variables are regarded: [a] *Time*, the experiments are developed on the basis of this independent variable; a time unit is considered as the period elapsed between one perception of the environment and another one. It is a quantitative variable whose value corresponds to the interval [0,600]; [b] *Threshold of Sensor reading*, the comparison of readings for each of the eight IR sensors in the proximity of the AIS (e-puck robot), is a set of quantitative variables [0,999] for each of the eight sensors of the AIS, from which the readings of the situations perceived throughout the time are compared. The determination of the comparison threshold of each sensor was carried out on the basis of the standard deviation of the readings of the situations registered by the robot in its initial born stage for fifty cycles, showing a reactive behavior; [c] *Threshold of Reading comparison of the effectors*, the speed values for each wheel of the AIS, is a set of quantitative variables [0,999] for each of the two wheels of the robot AIS, from which the readings of the actions (speeds applied to each wheel) are compared. The determination of the threshold for each wheel of the robot AIS is carried out on the basis of the standard deviation of the readings of the actions registered by the robot in its initial stage (born) for fifty cycles, showing a reactive behavior; [d] *Mutation* is a qualitative variable; it indicates if the theory mutation is active, generating new theories from similar theories, [8]; [e] *Plan ponderator* is a qualitative variable that indicates if the behavior of plan ponderation [8] is active; it is used for the selection of the plan to be executed by the AIS; [f] *Theory exchange* is a qualitative variable that indicates if theory exchange [9] is active and is used in the generation and assessment of theories between AISs as a product of exchange (cooperation); [g] *Collaboration* is a qualitative variable (proposed) that indicates if the collaboration between AISs is active, and is used in the generation and assessment of theories, from the theories provided by the action of collaboration of an AIS that is going around a higher LLC (Learning Life Cycle) layer towards a receiver AIS that is going around a lower LLC layer.

Dependent Variables: In the context of the experiments carried out, the following dependent variables are considered: [a] *Quantity of Situations* is a quantitative variable, whose value corresponds to the interval [0,200], and indicates the quantity of different situations that have been generated by the AIS according to the variable “time”; [b] *Quantity of successful theories* is a quantitative variable, whose value corresponds to the interval [0,600] and indicates the quantity of successful theories according to the variable “time”. Successful theories are considered to be those whose “U” utility exceeds the threshold of 0.75; [c] *Quantity of successful plans* is a quantitative variable, whose value corresponds to the interval [0,600]. This variable indicates the quantity of successful plans which have been generated by the system in the time interval: [0, time]. It is the quantity of plans that have reached their end after having gone around all the nodes of the plan chain; each node represents a situation (supposed conditions or predicted effects.) This value can be a rational number, since when a plan is not successful, its partial success is taken; [d] *Quantity of successful cycles* is a quantitative variable, whose value corresponds to the interval [0,600] and indicates the quantity of cycles that have proved to be successful according to the variable “time”; it is a value that is accumulated during the simulation and starts at zero (0) and increases by one (1) whenever the robot has carried out a cycle in which it ended with a U utility theory $> 0,75$ and $U < 1$; [e] *Percentage of successful cycles* is a quantitative variable, whose value corresponds to the interval [0,100] and indicates the percentage of cycles that have proved to be successful according to the variable “time”. This variable indicates the percentage of cycles that have ended with a new theory with U utility: $0,75 < U \leq 1$, at a given instant of the robot simulation which corresponds to the time interval [0, time]; [f] *Quantity of new theories* is a quantitative variable, whose value corresponds to the interval [0,600] and indicates the quantity of new theories according to the variable “time”. The theories considered are only those that were generated during the AIS operation and not those that were accumulated by the AIS in its theory base.

3.2 Graphs and Discussion of the Experimentation Results

To set up the architecture of each AIS that participates in every experiment, the following methods corresponding to independent variables were considered: mutation, plan ponderator, theory exchange between AISs and the proposal of the method of collaboration between AISs. Several experiments that come from the application of the different methods of learning acceleration and their combinations have been developed on the basis of the above-mentioned dependent and independent variables.

Figure 3 shows the comparison graph between situations and theories generated throughout the time, which has been obtained on the basis of the experimentation of an AIS set up with the methods of plan ponderation, mutation, and exchange and the collaboration one proposed. It is shown that even later, when it is observed a tendency to stabilization of the number of situations, the quantity of theories increases faster compared to the quantity of situations (supposed conditions or predicted effects). This shows that the system discovers relations between situations, being the AIS the theory generator.

Here follow the results of the experiments, compared to related works, by means of comparison graphs: the quantity of successful plans (fig.4), the quantity of successful cycles (fig.5), the percentage of successful cycles (fig.6), the quantity of successful

theories (fig.7) and the quantity of new theories (fig.8),under the following method configurations: AIS with plan ponderator and mutation methods, adding the collaboration method proposed (APMEC), AIS with plan ponderator, mutation and exchange methods (APME),proposed by Garcia Martinez et al [9], AIS with with plan ponderator and mutation (APM), proposed by García Martínez, R. y Borrajo, D., [8], Neuter AIS (AN) does not apply any method, proposed by Fritz, W et al [2].

Figure 4 shows the comparison graph of the quantity of successful plans; the AIS that combines the plan ponderator and mutation methods, adding the collaboration method proposed (APMEC), obtained the greater quantity of successful plans throughout the time, followed during the initial period by the AIS that applied plan ponderator, mutation (APM); after the initial period, the AIS that applied theory exchange experiences a slowing down in the quantity of successful plans generated and is surpassed by the AIS with Plan ponderator, mutation and exchange (APME). The lower quantity of successful plans throughout the time was generated by the neuter AIS (AN) that does not apply any method for learning acceleration.

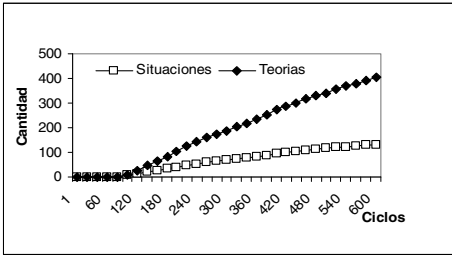


Fig. 3. Quantity of Theories vs situations

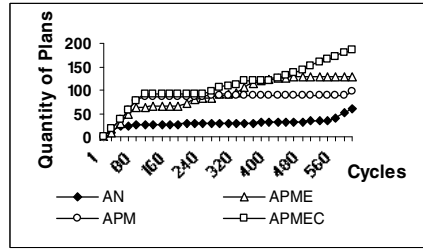


Fig. 4. Quantity of successful plans

Figure 5 shows the comparison graph of “Quantity of successful cycles”, the AIS (APMEC) that applies collaboration obtained the greater quantity of successful cycles throughout the time. The same is observed in figure 6, the comparison graph of “Percentage of successful cycles”, the AIS (APMEC) that applies collaboration obtained the best average percentage of successful cycles throughout the time, with respect to the other methods applied by the AISs (APME, APM, AN).

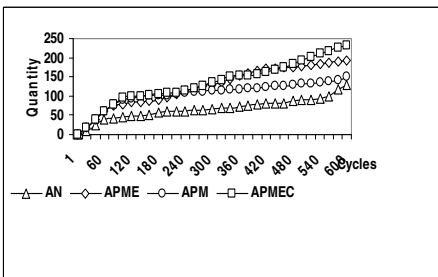


Fig. 5. Quantity of successful cycles

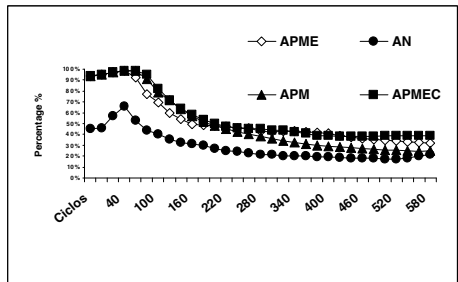


Fig. 6. Percentage of successful cycles

These results contribute to the fact that the AIS that applies collaboration may generate a greater quantity of successful plans, on the basis that it has a greater quantity of accumulated successful theories (theories whose utility is $0.75 < U \leq 1$), being the product of the application of the collaboration method proposed, as it can be observed in figure 7: *Quantity of Successful Theories*.

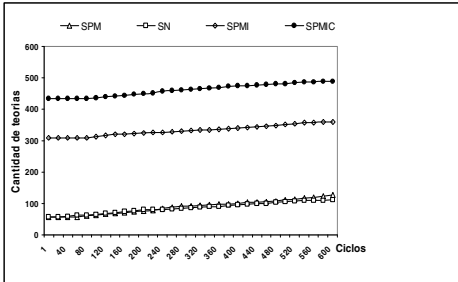


Fig. 7. Quantity of Successful Theories

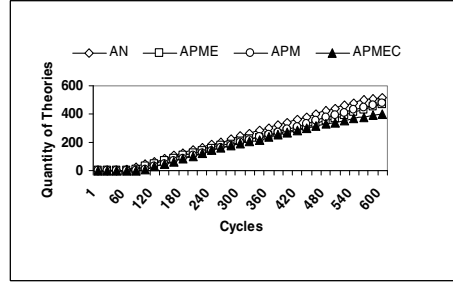


Fig. 8. Quantity of New Theories

Figure 8 shows the graph “Quantity of New Theories”; the greater quantity of new theories generated throughout the time corresponds to the neuter AIS (AN) and the smaller quantity to the AIS (APMEC) that applies collaboration. This is due to the fact that the knowledge base of theories of the neuter AIS is less representative regarding experience than that of the AIS that received collaboration; that is the reason why by having more experience in the operation environment (more quantity of successful theories), the AIS that received collaboration generated the smallest quantity of new theories compared to the rest of the methods that generated new theories during the experimentation.

Other series of experiments were carried out. They considered the individual comparison of each method applied on the basis of a neuter AIS, plan ponderator-mutation AIS, Exchange AIS, Collaboration AIS; the results were satisfactory for the AIS that received collaboration.

4 Preliminary Conclusions and Future Research Lines

According to the results achieved during the experimentation, it is observed that the collaboration produces a greater acceleration in the learning of a receiver AIS, with respect to the methods compared, with an increase of its successful plans and its successful theories throughout the time, as well as the quantity of successful cycles, the percentage of successful cycles and attaining a performance that required the generation of a smaller quantity of new theories, because of having a more representative theory base of its operation environment, attaining the best performance that results from the learning by collaboration. As new research lines we are exploring the mechanisms that may allow to invoke dynamically the collaboration of an AIS that is at a higher evolutionary stage, considering the selection metrics of theory bases of the AISs, based on percentages of successful cycles, quantity of theories with a utility

over 75 %, quantity of successful cycles and quantity of successful plans. Finally, we consider collaboration experimentation from an AIS that comes from a creator or developer different from that of the receiver AIS, given the proposed extension of the LLC (Learning Life Cycle). Possible real life examples of applicability of autonomous systems, are guided robot to avoid obstacles, for example to access the rescue of victims of earthquakes, these robots will share their theories and receive collaboration of robots that are in an upper layer of Learning Life Cycle.

References

1. Ierache, J., García Martínez, R., De Giusti, A.: Learning Life Cycle in Autonomous Intelligent Systems in IFIP International Federation for Information Processing. In: Artificial Intelligence and Practice II, Max Bramer, vol. 276, pp. 451–455. Springer, Boston (2008)
2. 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)
3. Borrajo, D., Veloso, M.: Lazy incremental learning of control knowledge for efficiently obtaining quality plans. *AI Review Journal* 11, 371–405 (1997)
4. Sutton, R.: Integrated architectures for learning, planning, and reacting based on approximating dynamic programming. In: Proc. 7th Int. Conf. on ML, pp. 216–224. Kaufmann, San Francisco (1990)
5. Carbonell, J.G., Gil, Y.: Learning by experimentation: The operator refinement method. In: Michalski, Kodratoff (eds.) *Machine Learning: An AI Approach*, vol. III, pp. 191–213. Morgan Kaufmann, San Francisco (1990)
6. Wang, X.: Planning while learning operators. PhD thesis, School of Computer Science, Carnegie Mellon University (1996)
7. García-Martínez, R., Borrajo, D.: Planning, learning, and executing in autonomous systems. In: Steel, S. (ed.) *ECP 1997. LNCS*, vol. 1348, pp. 208–220. Springer, Heidelberg (1997)
8. García Martínez, R., Borrajo, D.: An Integrated Approach of Learning, Planning and Executing. *Journal of Intelligent and Robotic Systems* 29, 47–78 (2000)
9. Maceri, D., García-Martínez, R., Borrajo, D., Britos, P.: Learning by Knowledge Sharing in Autonomous Intelligent Systems. In: Sichman, J.S., Coelho, H., Rezende, S.O. (eds.) *IBERAMIA 2006 and SBIA 2006. LNCS (LNAI)*, vol. 4140, pp. 128–137. Springer, Heidelberg (2006)
10. Braitenberg, V.: *Vehicles: Explorations In Synthetic Psychology*. MIT Press, Cambridge (1984)