

# Problems of Learning in Multi-Agent Systems

*L. Lhotská, J. Klema, O. Štěpánková*  
*The Gerstner Lab, Faculty of Electrical Engineering*  
*Czech Technical University in Prague*  
*Technická 2, 166 27 Prague 6, Czech Republic*  
*e-mail: {lhotska, klema, step}@labe.felk.cvut.cz*

## Abstract

Multi-agent systems are usually very complex in their structure and functionality. In most of the application tasks, it is difficult or sometimes impossible to determine exactly and correctly behavior and activities of a multi-agent system during its design. Therefore it is important to find a way how to improve system's activity during its operation. This can be achieved by learning agents which modify their behaviour according to their experience. There have to be studied and developed new methods of machine learning which will prove useful for this purpose. The paper reviews the basic problems of learning in multi-agent systems and some approaches applied for their solution.

## Keywords

**Multi-agent systems, learning, multi-agent learning**

## 1 INTRODUCTION

**Distributed artificial intelligence** (DAI) and especially **multi-agent systems** (MAS) have found growing interest both in theoretical research as well as in applications. Many DAI systems have been described in the literature, differing from each other in the entities involved (e.g. with respect to their number, the number of goals pursued by them, the degree of autonomy, and their perceptual, cognitive and effectual skills) as well as in the interactions between the entities (e.g. with respect to frequency, level, and purpose) (Crabtree 1996). Applications cover such areas as robotics, manufacturing, information retrieval, human-computer interaction (Durfee 1994). DAI systems receive considerable attention for two main reasons. First, they have useful properties, such as parallelism, robustness, and scalability. Therefore they are applicable in many domains which cannot be handled by centralized AI systems, in particular, they are well suited for domains which require, for example, resolution of interest and goal conflicts, integration of multiple knowledge sources and other resources, time-bounded

processing of very large data sets, or on-line interpretation of data arising in different geographical locations. Second, they are in accordance with the insight gained in disciplines like AI, psychology, and sociology that intelligence is tightly and inevitably coupled with interaction (Russell and Norvig 1995).

DAI systems are usually very complex in their structure and functionality. In most of the application tasks, it is difficult or sometimes impossible to determine exactly and correctly behavior and activities of a multi-agent system during its design. Complexity of the multi-agent systems results from: **environmental uncertainty** - it is impossible to define all conditions before the system starts to work; **environmental dynamics** - the system exists in an environment whose conditions vary over time; **communication constraints** - every communication link has limited parameters as range and bandwidth and a certain noise level; **degree of clustering** - in case of larger number of agents, it is advantageous to divide them into groups according to their functions, however, this functional grouping is limited; **time stress** - the time for decision making is not infinite, especially in real-time systems the question of quick response plays a vital role; **option multiplicity** - it represents the number of planning options available to each agent (the greater this number is, the more combinations of possible solutions can exist); **density of the solution space** - it represents the ratio of acceptable, conflict-free plans to the number of potential plans.

Agents and their system can be classified according to various criteria (Russell and Norvig 1995), (Wooldridge and Jennings 1996). **Reactive agents** react to changes they perceive in their environment, **deliberative agents** plan and act in goal-directed way, aim of the **utility-driven agents** is to maximize their utility function. When suggesting a taxonomy of agents, (Nwana 1996) applies three minimum basic concepts to be observed on the agents, namely ability to learn, to cooperate and their autonomy. Learning is included since it offers a sound solution to some problems resulting from the complexity of multi-agent systems which can be avoided or at least reduced by introducing the learning ability into the system (Weiss, Sen 1996), (Weiss 1998).

There are several areas where the role of learning seems obvious, namely human-agent communication; problem-solving capabilities, concerning particular problem domain; agent's behavior with respect to the whole system. But this list is far from being exhaustive. There have been distinguished two possible points of view for incorporating the learning ability into a multi-agent (MA) system design, namely learning of a single agent and learning of the whole MA system.

*What an individual agent can learn?* A reactive agent can change its decision function which characterizes the agent's reactions to possible states of the agent's world. What happens on average to a system consisting of reactive agents who learn about other agents is deeply studied in (Vidal, Durfee 1998). A reasoning or **intelligent agent** (Honavar 1998) can improve by learning its knowledge about its environment, about the other agents, about the agents' community or about itself. Agent's knowledge about another agent describes abilities of the other agent, its beliefs (intentions) or its commitments. Very specific problems arise in design of **customizable software agents** (Bradshaw 1997), which are supposed to act on

behalf of a human user. Such an agent tries to become most useful to its human user by learning user's interests by interacting with him/her (Yang et al. 1998).

From the point of view of the whole MA system it seems vital to develop methods which will ensure such behaviour of the MAS system leading to acceptable sharing of resources, maximizing individual profits or minimizing risk of average failure. It is important to know what are the necessary prerequisites for a MA system to learn to solve collectively a complex task which is beyond the reach of any individual agent. This task is closely connected to the problems of shared meaning and mutual understanding in the MA system (Weiss 1998). It is obvious that the field of MAS offers a wide area of both research and applications for machine learning. Let us start by reviewing the classical approaches to machine learning. Hopefully, this will help us identify those problems of learning in MAS which represent a new challenge as they seem to resist classical ML solutions.

## 2 LEARNING

The ability to learn, to adapt, to modify behavior is an inalienable component of human intelligence. We can identify four major machine learning paradigms (Carbonell 1990): **inductive learning** (e.g., acquiring concepts from sets of positive and negative examples), **analytic learning** (e.g., explanation-based learning and certain forms of analogical and case-based learning methods), **genetic algorithms** (e.g., classifier systems), and **connectionist learning methods** (e.g., nonrecurrent "backprop" hidden layer neural networks). These machine learning paradigms emerged from quite different scientific roots, employ different computational methods, and often rely on different ways of evaluating success (Mitchell 1997). They use different data and knowledge representation, as well as different formats of input and output data. In all cases, learning can be defined operationally to mean the ability to perform new tasks that could not be performed before or perform old tasks better (faster, more accurately, etc.) as a result of changes produced by the learning process. The simplest way to describe a task for learning is that of **supervised learning**. In such a case there is available description of a training set consisting of classified examples which express implicitly the difference among the considered classes. The classes are defined in advance by the human user. The goal of learning is to depict this classification explicitly, e.g. in the form of rules. However, it may happen that the information about class membership is absent in instances contained in a training set. That is a frequent case in nature where organisms learn not only from external critics (parents, society) but also from their own evaluation of the history. Under these conditions learning (adaptation) is based on input stimuli only and not on external evaluation of corresponding reaction. This setting is referred to as unsupervised learning (sometimes as selforganization).

### 2.1 Inductive learning

Inductive learning proves useful in tasks where there is available sufficient amount of data that serve as basis for formulation of generalized knowledge. Let us mention basic methods that enable to form a set of classification rules from a set of

training examples. The learning system tries to find such a set of rules that classifies the training examples at best. "Best" is understood with respect to accuracy and understandability. Resulting rules represent generalization of the training examples. The most frequently used basic algorithms are **AQ** (Michalski 1983) and **ID3** (Quinlan 1986). In both of them, the training examples are described by values of a fixed number of attributes and by the corresponding class where the given training example belongs. The inductive learning method used in AQ and ID3 is based on a simple recognition model in which correlations between observable properties and final classification are searched. The properties describing examples, i.e. attributes, can be selected by hand. Recently, there are appearing more sophisticated approaches to attribute selection as this task is crucial in the context of KDD systems (Yang and Honavar 1998). The original algorithms do not use any other problem-oriented knowledge besides the examples. They can be enriched by ability to use a knowledge base (Nunez 1991) or hierarchically structured background knowledge.

Very promising approach is that of ILP - **inductive logic programming** (Muggleton 1992), (Lavrač and Džeroski 1994). In ILP, information available about the observed objects is not limited to their individual properties but this type of learning relies on descriptions of relations among observed objects using a language of first order logic. The resulting classification is expressed in the form of a logic program (e.g. in Prolog). This approach counts on rich background knowledge and it supports even theory revision (De Raedt 1992).

## 2.2 Analytic learning

As opposed to inductive learning, analytic learning is based on learning from few examples (often a single one) and underlying background domain knowledge. There can be identified two larger groups of methods, namely deductive and analogical ones. **Deductive methods** construct a proof of why an instance is an example of a concept using the domain knowledge available. Past problem solving experience (the examples) is utilized to guide the choice of deductive chains (proofs) to perform when solving new problems. **Analytic methods** focus on improving the efficiency of a system without loss of accuracy or generality. They do not extend the library of concept descriptions as a rule. The basic method is **explanation-based learning** (Mitchell 1986). Analogical learning makes use of similarities in problem situations. Previous solutions are then re-used when solving new cases. This approach is applied in **case-based reasoning** (Kolodner 1993). These approaches have been successfully applied in industrial setting (Prasad 1996) to identify e.g. the domain knowledge for an expert system (Lowe 1998).

## 2.3 Genetic algorithms

Genetic algorithms (Souček 1991), (Goldberg 1989) represent empirical approach to machine learning. They have been inspired by a direct analogy to mutations in biological reproduction (cross-overs, point mutations, etc.) and Darwinian natural selection (survival of the fittest individual). Variants of a concept description correspond to individuals of a species, and induced changes and recombinations of

these concepts are tested against an objective function (the natural selection criterion) to see which to preserve in the gene pool. The first step in the solution process - i.e. representation of individual's attributes as combination of "1" and "0" in a binary chain - is very important and must be done very carefully. Then starting with a certain population sample, new generation is created through cross-over or point mutation operations applied randomly to the first generation. Each individual has a certain value of the evaluation function (often called fitness function). Individuals with low value of this function die out. The process usually runs in several dozens or hundreds of generations before it comes to a solution. In principle, genetic algorithms encode a parallel search through concept space, with each process attempting coarse-grain hill climbing. Genetic algorithms are very successful when searching suboptimal solutions of NPC problems (Kubalík and Lažanský 1998).

## **2.4 Connectionist learning methods**

Connectionist learning systems, also called "neural networks" or "parallel distributed systems", have received much attention (Souček 1991). The predecessors of recent neural networks - perceptrons and linear networks - suffered from both theoretical and practical limitations. Introduction of "hidden layers" has enabled to represent intermediate processing and compute nonlinear recognition functions. Basically, there are two types of connectionist systems: those that use distributed representations - where a concept corresponds to an activation pattern spanning, potentially, the entire network - and those that use localized representations where physical portions of the network correspond to individual concepts. In order to evaluate the appropriateness of ML technique to the task at hand, one must ask some detailed quantitative questions, such as comparing the ease of casting training data into acceptable representations, the amount of training data required for sufficiently accurate performance, the relative computational burden of each technique in both training and performance phases, and other such metrics. Besides that there may appear a requirement of readability of learned knowledge. It is obvious from the data and knowledge representation used in individual types of learning systems that it is impossible to "decode" internal knowledge learned by neural networks or genetic algorithms. On the other hand, knowledge encoded in inductive systems, especially in decision trees, is well understood.

# **3 LEARNING AND ADAPTATION IN MULTI-AGENT SYSTEMS**

## **3.1 Introduction**

For most application tasks, and even in environments that appear to be more or less simple, it is extremely difficult or even impossible to fully determine behaviour and concrete activities of a multi-agent system a priori - that is, at the time of its design and prior to its use. This would require, for instance, that it is known in advance which environmental requirements will occur in the future, which agents will be available at the time of requirements occurrence, and how the available

agents will have to interact in response to these requirements. Enumeration of all the possible states of a multi-agent system would cause a combinatorial explosion. As a matter of fact, only a small subset of these states really occurs, but prior knowledge to determine this subset is not usually available.

These kinds of problems resulting from the complexity of multi-agent systems can be avoided or at least reduced by involving the agents with the ability to *adapt* and to *learn* (Weiss 1995). The term adaptation is used in this text as a synonym of learning and no further explicit distinction is made between both of them.

### 3.2 Categories and forms of learning

Learning in multi-agent systems is based on machine learning essentially, but it cannot reuse machine learning algorithms directly due to a different environment. The process of learning in multi-agent systems is determined by (and in many cases is even aimed at) multi-agent system characteristic features. That is, for example, supposed communication and interaction flow among agents connected with exchange of information, changes in dynamic environment, changes of beliefs, intentions and desires of individual agents, shared assumptions, and so forth. All of these features place special emphasis on incrementality and the design of MAS as a whole. Thus most of the ML algorithms have to be extended at least.

The group learning is manifested in better coordination or more effective task and resource allocation. Better coordination is reached by information and knowledge sharing or more effective communication among agents. Task and resource allocation can be improved by learning of **agent specialization** (e.g. agent B always performs task U well), by learning of **group properties** (e.g. agents E and F work well as a team), by learning of **task patterns** (e.g. given type of task is solved more easily if decomposed to two subtasks C and D, and first subtask D is solved) and by learning the **properties of the environment** (e.g. user priorities or machine reliability).

In the introduction we have already mentioned both extreme points of view on the process of learning in MAS distinguished in (Weiss 1995): **isolated (single-agent) learning** - does not rely on the presence of multiple agents, **interactive (multi-agent) learning** - relies or even requires the presence of multiple agents and their interaction.

Since the above mentioned categorization is relatively vague, both concepts are practically considered as principle categories of learning standing on opposite sides of a learning algorithms spectrum. The boundary between them is not sharp and so there is a remarkable amount of systems which are difficult to be assigned into one of the categories. The concept of interactive learning itself can be applied in two different ways. In its stronger and more specific meaning, interactive or multi-agent learning refers only to situations in which several agents learn how to pursue a common learning goal (Bazzan 1998). In its weaker and less specific meaning, it additionally refers to situations in which an agent pursues its own learning goal, but is affected in its learning by other agents (Weiss 1995).

In a similar way, as in the field of ML there exist several criteria that can structure a great amount of learning forms in multi-agent systems. Some of them

are taken directly from the ML field, some of them reflect a special multi-agent view. One of the well-known ML criteria distinguishes learning forms according to the type of **learning feedback** that is available to a learning entity and indicates the performance level achieved so far:

- supervised learning - the feedback specifies the desired activity of the learner; the goal of learning is to match this desired action as closely as possible,
- reinforcement learning - the feedback only specifies the utility of the actual activity of the learner and the goal is to maximise this utility,
- unsupervised learning - no explicit feedback is provided and the goal is to find out useful and desired activities on the basis of trial-and-error process.

These criteria are mentioned again because they offer new and interesting aspects within the multi-agent learning domain. The role of the teacher or critic who provides learning feedback can be played by another agent or group of agents. The system environment can provide this feedback as well. Other criteria relating to multi-agent environment offer additional means for structuring. The **decentralisation** of learning process can vary between two obvious extremes:

- only one of the available agents is involved in the learning process and the learning steps are neither distributed nor paralleled,
- all available agents are involved, and the learning steps are maximally distributed and paralleled.

The criterion of *the purpose and goal of learning* can oscillate between two extremes as well:

- learning aims at an improvement with respect to one single agent, extending its skills and abilities,
- learning aims at an improvement with respect to the group of agents as a unit, their coherence and co-ordination.

According to the *moment of communication among agents* with respect to the learning process, the following approaches can be distinguished (Davies 1996):

- data are gathered into one place before the learning process starts,
- individual agents learn on local data, partial results are shared by means of communication during the learning process,
- agents learn locally and later they share their results, which are then refined and integrated by other agents in light of their own data and knowledge.

Furthermore, the learning forms can be distinguished according to criteria of *agent's involvement in the learning process* (the involvement of observed agent is or is not a necessary condition for achieving the pursued learning goal) or according to *agent-agent and agent-environment interaction* (learning can require only minimal degree of interaction or learning would not be possible without extensive interaction). Moreover, learning does not have to be a permanent and stable process. It can be initialized, accelerated or stopped. An important research topic to be studied is the search for those properties of the MAS and its environment which provide necessary conditions for various types of learning.

### **3.3 Current related and relevant work on multi-agent learning**

In contrast to the fact that learning in multi-agent systems is a relatively new field of study, it is possible to get a remarkable amount of references to work already

done in this field. Recent volume (Weiss 1998) offers an up-to-date reader's guide and a valuable overview of major challenges for machine learning in MAS. Learning in distributed AI systems has been chosen as the topic of a special issue 4/97 of the Journal of Experimental and Theoretical Artificial Intelligence (JETAI). In 1998, the Machine Learning Journal plans a special issue on multiagent learning to be edited by M. Huhns and G. Weiss. A general comparison of distributed problem solving and multi-agent systems can be found in (Durfee 1994). (Davies 1996) deals with more special problems of the agent based approach to data-mining. (Prasad 1996) describes distributed case bases while (Vreswijk 1995) gives attention to the protocols in multi-agent systems.

Possible domains of machine learning algorithms applications within the distributed systems have been studied in The Gerstner laboratory as well. (Klema 1996) suggests procedures aiming to improve communication between heterogeneous agents of DISCIM (Distributed Computer Integrated Manufacturing). These procedures are maintained by single learning agent. (Štěpánková 1998) focuses on tri-base acquaintance model modification by meta-agent within ProPlanT (Production Planning Tool) system (Mařík 1998) in order to achieve flexible adaptation of a distributed agent net. There are considered three central targets for optimisation (Štěpánková 1996) to be improved by activity of the system:

- the single agent behaviour - it concentrates mainly on the agent's reliability judgement with respect to the co-operating partners,
- a specific task to be fulfilled - it tries to find criteria for evaluation of quality of a solution (precision of the offered answer, its cost and availability), these criteria are then used to choose the best agent giving the right answer,
- communication within the agents' community - it searches for clusters of co-operating agents to restructure them or it searches for unintended looping during the search for an answer to a certain task.

#### 4 CONCLUSION

Learning from experience is a powerful technique used by humans to improve their problem-solving ability. It is considered to be an inseparable part of intelligence. Since the ability of learning is fundamental for every intelligent system that should improve its problem solving, it must become a necessary attribute of intelligent agents as well. In multi-agent systems, cooperative learning is necessary in some cases and highly useful in most cases. In addition, for a given number of instances, the predictive power of the agents is much greater in the multi-agent cooperative learning case than in the individual learning case. The penalty to pay for this is that such a system is slower as it has the overhead of cooperation.

We have tried to outline basic problems of learning in multi-agent systems and applicable learning paradigms. It is obvious that each learning method has its advantages and disadvantages and therefore its scope of applicability is limited. Thus, when designing a system for a certain application domain careful choice of



proper methods may represent significant qualitative shift in the performance of the system and its problem-solving ability.

## 5 ACKNOWLEDGEMENT

The research has been supported by the Gerstner Lab project No. VS96047 of the Ministry of Education, Youth and Sports of the Czech Republic and grant No. 102/96/0382 of the Grant Agency of the Czech Republic.

## 6 REFERENCES

- Bazzan, A.: Evolution of Coordination as a Metaphor for Learning in Multi-Agent Systems, In. (Weiss, 98), pp. 117-136
- Bradshaw, J. (ed.): Software Agents, MIT Press, Cambridge, MA 1997
- Carbonell, J. (editor): Machine Learning: Paradigms and Methods. MIT Press, 1990
- Crabtree, B., Jennings, N. (editors): Proceedings of 1st Int. Conf. on the Practical Application of Intelligent Agents and Multi-Agent Technology. The Practical Application Company, London, 1996
- Davies, W., Edwards, P.: Distributed Learning: An Agent Based Approach to Data-Mining, University of Aberdeen, 1996
- De Raedt, L.: Interactive Theory Revision: An Inductive Logic Programming Approach. Academic Press, New York 1992
- Durfee, E., Rosenschein, J.: Distributed Problem Solving and Multi-Agent Systems: Comparisons and Examples, University of Michigan 1994
- Goldberg, D. E.: Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, 1989
- Honavar, V.: Intelligent Agents. In: Encyclopedia of Information Technology (Williams, J. and Sochats, K., eds), Marcel Dekker, New York 1998
- Huhns, M.N. (editor): Distributed Artificial Intelligence. Morgan Kaufmann, San Mateo, CA, 1987
- Klema, J.: Use of Machine Learning Algorithms in Distributed Environment, Research Report, GL – 3/96, CVUT FEL, Prague, 1996
- Kolodner, J.: Case-Based Reasoning. Morgan Kaufmann, California, 1993
- Kubalík, J., Lažanský, J.: On Tuning and Experimental Evaluation of Genetic Algorithms. In Proc. of the XVth EMCSR, Vienna 1998, pp.854-859
- Lavrač, N. and Džeroski, S.: Inductive Logic Programming. Techniques and Applications. Ellis Horwood Ltd., 1994
- Lhotská, L.: Learning in Multi-Agent Systems: Theoretical Issues. In: EUROCAST'97. LNCS 1333, Springer, 1997, pp. 394-405
- Lowe, H., Pěchouček, M., Bundy, A.: *Proof Planning for Maintainable Configuration Systems* in Artificial Intelligence for Engineering Design and Manufacturing, Cambridge University Press, Massachusetts, September 1998
- Mařík, V., Pěchouček, M., Hazdra, T., Štěpánková, O.: ProPlanT - Multi-Agent System for production Planning. In Proc. Of the XVth European Meeting on Cybernetics and Systems Research, Vienna 1998, pp. 725-730

- Michalski,R.S.: Learning from Observation: Conceptual Clustering. In Michalski,R.S., Carbonell,J.G. and Mitchell,T.M. (eds): Machine Learning: An Artificial Intelligence Approach, Morgan Kaufmann 1983
- Mitchell,T.M., Keller,R.M. and Kedar-Cabelli,S.T.: Explanation Based Generalization: A Unifying View, In Machine Learning 1 (1986), pp. 47-80
- Mitchell,T.M.: Machine Learning, Morgan-Kaufman, 1997
- Muggleton,S.: Inductive Logic Programming. Academic Press, New York 1992
- Nunez,M.: The Use of Background Knowledge in Decision Tree Induction. In: Machine Learning, 6, 1991, pp. 231-250
- Nwana,H.S.: Software Agents: An Overview, Knowledge Engineering Review, Vol. 11, No. 3, pp. 1-40, Sept. 1996, Cambridge University Press, 1996
- Prasad,N., Lesser,V., Lander,S.: Retrieval and Reasoning in Distributed Case Bases, Journal of Visual Communication and Image Representation, pp 74-87, 1996
- Quinlan,J.R.: Induction of Decision Trees, Machine Learning Vol.1,pp.81-106, 1986
- Russell,S., Norvig,P.: Artificial Intelligence. A Modern Approach, Prentice Hall,Inc., Englewood Cliffs 1995
- Souček,B. and IRIS Group, Neural and Intelligent System Integration, 1991, A Wiley-Interscience Publication
- Štěpánková,O., Mařík,V., Lažanský,J.: Improving Cooperative Behaviour in a Multi-Agent System, IFIP World Conference on IT Tools, Canberra, 1996
- Štěpánková,O., Pěchouček,M., Mařík,V., Lažanský,J.: KGB Model of CIA, Knowledge-based Goal-driven Behaviour Model of Co-operating Intelligent Agents, 1998
- Vidal,J.M., Durfee,E.H.: Learning Nested Models in an Information Economy. Journal of Experimental and Theoretical Artificial Intelligence: Special Issue Learning in DAI Systems, 1998, <http://ai.eecs.umich.edu/people/jmvidal/papers>.
- Vreeswijk,G.: Open Protocol in Multi-Agent Systems, Technical Report CS95-, University of Limburg, 1995
- Weiss,G., Sen, J. (eds): Adaptation and Learning in Multi-Agent Systems, LNAI, Vol 1042, Springer-Verlag 1996)
- Weiss,G. (ed.): Distributed Artificial Intelligence Meets Machine Learning, LNAI, Vol 1221, Springer-Verlag 1998
- Wooldridge,M.J., Jennings,N.R.(eds): Intelligent Agents. LNAI No. 890, Springer Verlag, Heidelberg 1995
- Yang,J et al.: Mobile Intelligent Agents for Document Classification and Retrieval: A Machine Learning Approach. In Proceedings of EMCSR'98 (Trappl,R. ed.), pp.707-712, Vienna 1998
- Yang,J. and Honavar,V.: Feature Subset Selection Using a Genetic Algorithm. IEEE Expert (special Issue on Feature Transformation and Selection), 1998