iJET EXPLORE International Journal of Engineering & Technology
CENTRE OF PROFESSIONAL RESEARCH PUBLICATIONS
ISSN 2049-3444

International Journal of Engineering and Technology Volume 2 No. 8, August, 2012

An Implementation of Machine Learning Systems using Fuzzy Distributed Artificial Intelligent Systems

[1]M.Kumarasamy, [2]A.Jebaraj Ratnakumar
[1]DMI College of Engineering, Chennai
[2]Apollo Engineering College, Chennai

ABSTRACT

The main goal of this paper is to develop machine learning systems using fuzzy distributed artificial intelligent systems. The goal of machine learning is to ensemble learning and adaptation abilities of living species in computers; more deeply to program computers to use past experience to solve a given problem. As also stated by Michalski: "Learning is constructing or modifying representations of what is being experienced." In general, machine learning refers to a system capable of the autonomous acquisition and integration of knowledge. Nowadays many powerful methods with different roots, has been introduced in ML such as: neural networks, genetic algorithms, genetic programming, fuzzy logic and also many hybrid approaches as a combination of some aforementioned ones. Using Machine Learning with Fuzzy techniques we implement Fuzzy Q Learning Algorithm.

Keywords: *Fuzzy logic, Artificial Intelligence, Neural Networks, Machine Learning*

I. INTRODUCTION

Fuzzy Logic

Fuzzy logic was proposed on fuzzy set theory to capture the way human beings represent and reason real world knowledge to solve uncertain problems. Uncertainty arises due to some peculiar characteristics like generality, vagueness, ambiguity and incomplete knowledge. A fuzzy set is represented mathematically by assigning to each member in the set a value representing its grade of membership in the fuzzy set, where the grade corresponds to the degree to which the member is similar or compatible with the concept represented by the fuzzy set. In mathematical set theory an element exhibits the property of whether it belongs to the set or not and so they are called as crisp sets. But when it comes to fuzzy sets many degrees of freedom are allowed. A membership function is associated with a fuzzy set so that the function maps every element in the set on to a value between 0 and 1. Fuzzy logic is applicable in the areas of control systems, pattern recognition applications and decision making.

Artificial Intelligence

Artificial intelligence is an area of computer science related with designing, intelligent computer systems. These characteristics we associate with intelligence in human behavior. The term intelligence is less understood. The artificial intelligence tasks are learning, intuition, creativity, inference…and all are partially understood. Some technologies such as expert systems, neural networks, fuzzy logic, cellular automata, probabilistic reasoning exist to solve intelligent problems. Out of these neural networks, fuzzy logic and probabilistic reasoning are known as soft computing. Soft computing was introduced by Lotfi A.Zadeh. Probabilistic reasoning subsumes genetic algorithms, chaos and parts of learning theory. According to Zadeh, soft computing differs from hard computing in its tolerance to imprecision, uncertainty and partial truth. Hard computing methods are based on mathematical approaches and therefore demand a high degree of precision and accuracy in their requirements. In most engineering problems the input parameters are used for obtaining solution to the problem.

Distributed Artificial Intelligence

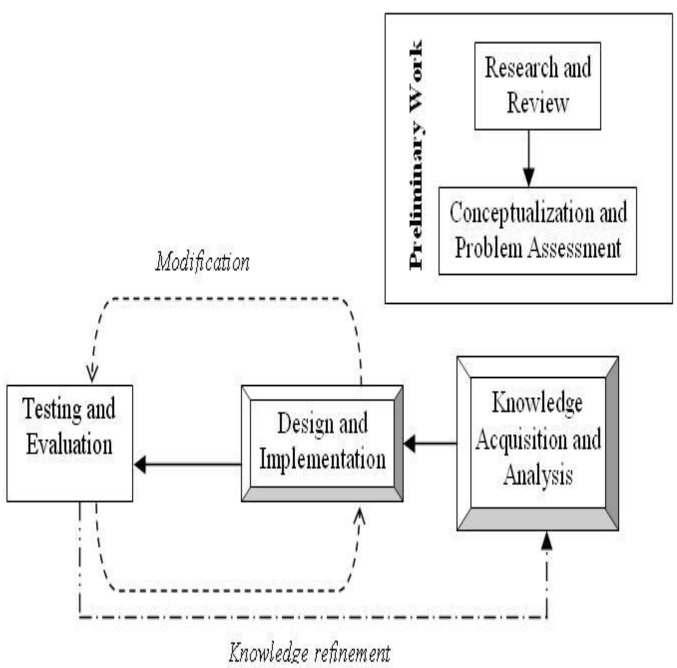
Distributed Artificial Intelligence (DAI) systems can be defined as cooperative systems where a set of agents act together to solve a given problem. These agents are often heterogeneous. Its metaphor of intelligence is based upon social behavior (as opposed to the metaphor of individual human behavior in classical AI) and its emphasis is on actions and interactions, complementing knowledge representation and inference methods in classical AI.

II. OBJECTIVES

This paper addresses learning issues for an autonomous agent in a highly dynamic and noisy environment with huge state space, where conventional techniques are unable to perform. The main objective is developing a learning algorithm, based on existing ML methods, capable of handling both continuous input space and continuous action space; which improves the learning speed and facilitates the agent by high quality learned individual and multi-agent behaviors, while call for low information requirements about the environment. Since the attention is more towards the robotic soccer, the algorithm should be fully suitable to perform in that domain in different levels of abstract multi-agent skills.

III. METHODOLOGY

Reinforcement Learning (RL) has got enormous attention in autonomous robotics and other complex and collaborative areas, for its flexibility and simplicity. On the other hand fuzzy expert systems also have been extensively used in intelligent control problems where mostly traditional methods have poor performance. This thesis is about the utilization of fuzzy theory in RL to enhance learning with more adaptation of RL for continuous and multi-agent domains and to speedup learning process. Mostly the fuzzy RL (FRL) has been used in control engineering rather than in computer science, due its history in control theory, whilst FRL as simple RL provides a general way to solve all RL problems with the use of fuzzy theory. Fig.1 represents System's development framework.



Using Machine learning we collected the informations and put into knowledge base. From the knowledge base we have to analyse the data.

IV. REINFORCEMENT LEARNING

There exist situations in ML field which the resources provided by the problem are so poor and inadequate to utilize supervised learning algorithms. On the other hand in some other cases even there is no precise information about the data on which learning should be done; moreover there might be no former sample data set available, where some unsupervised learning approaches depend on. Having all these limitations fired, mostly reinforcement learning (RL) is the ultimate way out. In unknown situations such these learning should be from experience. RL is a way of learning behaviors for agents by interacting with their environment without any explicit teacher. It has its roots from psychology.

Learning can be categorized in various types some as follows:

- Supervised learning (learning by a teacher)
  -Learning form examples.
  -Learning by taking advice.

- Unsupervised learning (learning without a teacher)
  -Competitive learning.
  -Clustering.
  -Reinforcement learning.

Because of its less demands on the information source provided to the learner, here we are more interested in Reinforcement Learning (RL).

V. REINFORCEMENT LEARNING MODEL

The agent ***B*** in each moment, or time step, *t* perceives its environment state through its sensors denoted as input *it* , and decides to perform an action *at* based on this sensory information and the feedback from the environment *rt* on previous action (reinforcement signal), towards a specified goal, that of course the environment forces it to do so. Therefore the learning's goal simply, is to find best possible action in each state that will result in the maximum total sum of the reinforcement signals, rewards; remember the famous notion of having more score in the basket of the learner from animal psychology. The input function ***I*** is the sensation function of the current state of the environment; it shows the way the agent views its surroundings. ***R*** is the reward function that is mostly unknown to the learner in detail. As a proper formal definition of RL, of course one out of many, it can be: "Reinforcement learning is learning what to do, how to map situations to actions, so as to maximize a numerical reward signal."

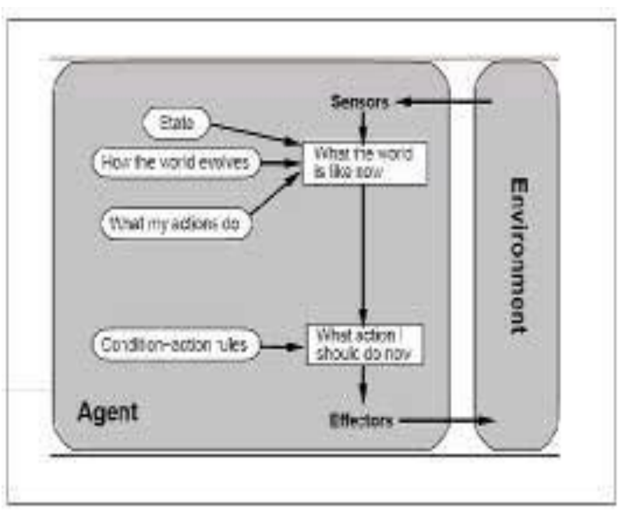
Thus the model has been constructed on top three formal bases:

- Set of environment states, ***S***: It can discrete or continuous, finite or infinite.
- Set of agent actions, ***A***: It can be discrete or continuous.
- Set of scalar reinforcement signals: Either Boolean *{0,1}*, or continuous realvalued.

VI. LEARNING COMPONENTS

Based on what described above, RL problems generally can be thought of consisting three elements as follows:

1. Agent: The learner that can interact with the environment via a set of sensors and actuators.
2. Environment: A time varying generally non-deterministic stationary media, well-expressed by Markov Decision Process (MDP) which will be described in next section. Non-deterministic means the resulting state and reward signal of taking the same action in the same state on two different occasions may differ; whilst stationary means that the probabilities and expectation of getting such results, i.e. special state transitions or rewards, are constant and not time varying.
3. Policy π : A mapping from perceived states set ***S*** to the actions set ***A*** (in psychology stimulus-response rules or associations); it also determines what action should be taken afterwards.



VII. FUZZY Q-LEARNING ACTION SPACE

Framework

FQL is the fuzzy extension of Q-learning which is an online model free optimization of a control policy. FQL algorithm has been developed and they have many things in common and the same in many steps. In FQL structure there is no need to have conclusion values and their eligibility traces. The action quality values *wi* (*k*) will be used for both *t*-optimal value function approximation and action selection policies. So FQL structure consists of:

- Each rule *Ri* in the rule-base has a set of discrete actions associated to it denoted by *U i* .
- Each *kth* action in each rule, *U i* (*k*) , has a quality factor *wi* (*k*) (a weight) which will be adjusted through learning phase. Thus each rule also has a weight vector consisting of its actions qualities denoted by *wi* .
- There is a Q-value associated to the current state-action pair denoted by ~ (, ( )) *Qt Xt Ut Xt* which will be calculated through TS-FIS output.
- There is also an optimal quality value, Q-value, for the current state denoted by * ( ) *Qt Xt* which will be computed through fuzzy inference.

FQL Algorithm

The complete FQL learning process during one time step can be summarized as follows; assume that the current time stamp is *t+1*, the agent has already performed the action *Ut* (*Xt*) at previous time step *t*, and has even received the reward *rt*+1 , then:

1. Fuzzification of the new perceived input state *Xt*+1 .
2. Rule evaluation and truth value computations.
3. Estimation of the optimal Q-value of current state *Xt*+1 based on the learned Q-value function

1. Fuzzification of the new perceived input state X_{t+1}.
2. Rule evaluation and truth value computations.
3. Estimation of the optimal Q-value of current state X_{t+1} based on the learned Q-value function till now (Eq. 6.18 for X_{t+1}): $Q_t^*(X_{t+1})$.
4. *TD Error* $\tilde{\varepsilon}_{t+1}$ calculation (Eq. 6.19):
   $\tilde{\varepsilon}_{t+1} = r_{t+1} + \gamma Q_t^*(X_{t+1}) - \tilde{Q}_t(X_t, U_t(X_t))$.
5. Updating FIS by tuning action qualities (Eq. 6.20): $w_{t+1} = w_t + \beta_t \tilde{\varepsilon}_{t+1} e_t^T$.
6. Perform learning rate update by meta-learning rule of Eq. 6.10.
7. Updating eligibility traces (Eq 6.16).
8. Double ε-Greedy action selection procedure (Eq. 6.12 and Eq. 6.13), and performing the total maximum ε-Greedy action $U_{t+1}(X_{t+1})$.
9. Computing and memorizing Q-value of the current state-action pair X_{t+1} and $U_{t+1}(X_{t+1})$ based on the new Q-value function after tuning action weight parameters values (Eq. 6.17): $\tilde{Q}_{t+1}(X_{t+1}, U_{t+1}(X_{t+1}))$.

VIII. CONCLUSION AND FURTHER RESEARCH

In this paper the general approach of reinforcement learning problems was studied, and specifically focused on generalization enhancements as the advanced RL. The capability of fuzzy inference systems as function approximates were employed in the value function approximation and continuous action interpolator; to be able to effectively generalize over huge state space and also construct more sensible actions. A new methodology based on the previous fuzzy reinforcement learning has been introduced fully in detail. The approach was broken down into different specific methods such as FQL capable of handling both continuities in the state space and action space. In general the methods are powerful enough to tackle the curse of dimensionality and other ordinary RL issues rising in real life and industrial problems. Of course all these have been paid with a reasonable price in a trade-off for more computational cost compared to simple basic RL algorithm.

For further research, for instance, the methodology can be more refined by some sort of fuzzy label automatic generation. Now the rule-base of the FIS will be automatically generated but fuzzy labels should be determined by the user. It should also be noted that having dynamic rule-base and fuzzy linguistic labels will also increase the computational cost, and thereafter limiting the real-time features of the method. Another interesting issue is the introduction of fuzzy reinforcement function (FRF) instead of using multiple reinforcement function; and thus encapsulating different goals form short-term, intermediate to long-term team goals, specifically in multi agent environments with collaborative and adversarial behaviors. This reward then would be a kind of continuous-valued reinforcement, behavioral shaping, and providing smooth transitions in reinforcement; in contrast with multiple reinforcements.

REFERENCES

[1] Antsaklis, P. J., & Passino, K. M. (1993). An Introduction to Intelligent and Autonomous Control, Kluwer Academic Publishers, Norwell MA, 1993.

[2] Barto, A. G., Sutton, R. S., & Anderson, C. W. (1983). Neuronlike adaptive elements that can solve difficult learning control problems, IEEE Transactions On Systems, Man and Cybernetics, September 1983, vol. SMC-13, pp. 834-846.

[3] Chen, M., Dorer, K., Foroughi, E., Heintz, F., Huang, Z. X., Kapetanakis, S., Kostiadis, K., Kummeneje, J., Murray, J., Noda, I., Obst, O., Riley, P., Steffens, T., Wang, Y., & Yin, X. (2003). Robo Cup Soccer Server User Manual: for Soccer Server version 7.07 and later, The Robo Cup Federation, February 11, 2003.

[4] Dautenhahn, K. (1995). Getting to know each other—artificial social intelligence for autonomous robots, Robotics and Autonomous Systems, 1995, 16:333–356.

[5] Dubois, D. and H. Prade (1980). *Fuzzy sets and systems: theory and applications*, Volume 144 of *Mathematics in science and engineering*. Academic Press.

[6] Filev, D.P. (1996). Model based fuzzy control. In *Proceedings Fourth European Congress on Intelligent Techniques and Soft Computing EUFIT'96*, Aachen, Germany.

[7] Jang, J.-S.R. and C.-T. Sun (1993). Functional equivalence between radial basis function networks and fuzzy inference systems. *IEEE Transactions on Neural Networks 4*(1), 156–159.

[8] Kaymak, U. (1994). Application of fuzzy methodologies to a washing process. Chartered designer thesis, Delft University of Technology, Control Lab., Faculty of El. Eng., Delft.

[9] Smith, A. J. (2001). Dynamic Generalisation of Continuous Action Spaces in Reinforcement Learning: A Neurally Inspired Approach, Ph.D. Thesis, Institute for Adaptive and Neural Computing, Division of Informatics, University of Edinburgh, October 2001.

[10] Stone, P. (1998). Layered Learning in Multi-Agent Systems, Ph.D. Thesis, CMU-CS-98-187, School of Computer Science, Carnegie Mellon University, December 15, 1998.

[11] Stone, P., & Sutton, R. S. (2001). Scaling Reinforcement Learning toward RoboCup Soccer, In Proceedings of the Eighteenth International Conference on Machine Learning, Morgan Kaufmann, San Francisco, CA, 2001, pp. 537–544.