Fuzzy Reinforcement Learning Rectilinear Follow-up of Trajectory per Robot

Youcef Dahmani[1] and Abdelkader Benyettou[2]
[1] Laboratoire SIMPA, Ibn Khaldoun University Tiaret, Algeria
[2] Department of Computer Sciences, University of Sciences and Technology of Oran, Algeria

Abstract: *Knowing the action space of an order, the objective consists in distributing this space in a certain set of actions equitably in order to choose the famous action among the candidate ones. This process is ensured by reinforcement learning aided by fuzzy logic. We have established an algorithm applying the fuzzy Q-learning with a fuzzy limited lexicon. We have applied it to a robot for the training of the follow-up of a rectilinear trajectory from a starting point "D" at an unspecified arrival point "A", while avoiding with the robot butting against a possible obstacle. The goal of this work tries to answer the question, in what the reinforcement learning applied to fuzzy logic can be of interest in the field of the reactive navigation of a mobile robot.*

Keywords: *Mobile robot, navigation, reinforcement learning, fuzzy logic, fuzzy Q-learning, fuzzy inference system.*

Received April 21, 2004; accepted August 21, 2004

1. Introduction

The mobility and the autonomy of the robots pose complex problems, as regards generation of trajectory in strongly constrained and not structured spaces [8]; moreover the other problem is of decision-making starting from information sensors vague or incomplete. To this end robots need more sense, decision and technology [10].

It is accordingly that we initially tried to teach the robot how it can follow a rectilinear and straight line trajectory aiming to make the follow behaviour perfectly an object which constitutes one of the modules in the navigation of a mobile robot while considering the behaviour-based architecture [6].

2. Robot Architecture

In the present work, we used a standard architecture [11]. The robot considered is circular having three sensors one in front and one on each side. The angle of sensors orientation chooses are of 45° on both sides of the frontal axis of the robot as shown in Figure 1.

The robot must move along a straight line trajectory, from a starting point "D" to any objective point "A". It must thus learn to follow this trajectory. By these three sensors the robot calculates the length "l" compared to an eventual obstacle, its orientation θ and the angle θ' with respect to the objective as shown in Figure 2.

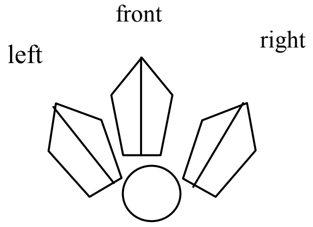


Figure 1. Structure and provision of the sensors.

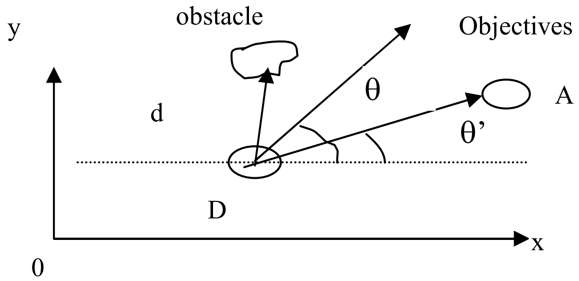


Figure 2. Kinematic model.

3. Navigation

Our aim is to allow the robot, initially to orient its angle directly towards the objective point. Then it must learn how to move along this trajectory by holding its straight line.

The use of fuzzy logic seems to give good results in this kind of problems such navigation without an analytical model of the environment. It remains to notice, as soon as the environment becomes complex, two problems emerge to knowing:

- The difficulty of the construction of the rule basis.
- Refinement of this rule basis.

In order to remedy to these problems, we proposed a model using fuzzy logic and the reinforcement learning. The main concepts in the reinforcement learning, are the agent and environment [3, 4, 5]. The agent has a number of possible actions; the agent improves some actions in the environment which is modeled by a set of states. For some states, the agent receives a signal of the environment called the reward. The task of the reinforcement learning is to find the action which gives the greatest value of the discounted reward called the Q-value. The step passes by two stages:

- A phase of exploration.
- The second phase of exploitation.

3.1. The Reasoning Process

The well known method by the most popular reinforcement learning is Q-Learning where an agent updated successively the quantity Q_i (x, a) which represents the quality of the selected action "a" for the state "x". Within this framework, we used an alternative of Q-Learning associated with fuzzy logic in order to use its properties such the formulation of human knowledge in the form of fuzzy rules and the use of imprecise and vague data as well as possible. Its principle consists in proposing several conclusions for each rule and to associate each potential solution a quality function [2, 7].

R_i: If x_1 is $A_1{}^i$ and ……….. and x_n is $A_n{}^i$ Then
y is u [i, 1] with q [i, 1]=0
or
y is u [i, 2] with q [i, 2]=0
…………………………..
or
y is u [i, N] with q [i, N]=0.

Where $(u\ [i,\ j])^{N}_{j\ =\ 1}$ are potential solutions whose quality is initialized arbitrarily.

The inferred output is given by the formula:

$$U(x) = \frac{\sum_i \alpha_i(x) u[i, PEE\ (i)]}{\sum_i \alpha_i(x)}$$

The quality of this action is:

$$Q(x, U) = \frac{\sum_i \alpha_i(x) q[i, PEE\ (i)]}{\sum_i \alpha_i(x)}$$

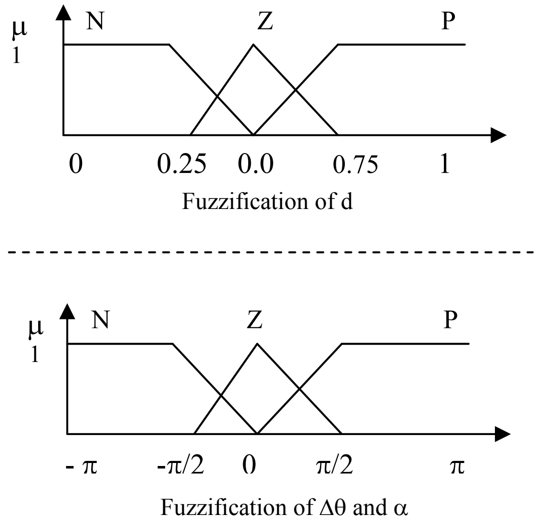
Our approach, is similar in its principle to that previously quoted, except that the set of the suggested actions are not crisp values but fuzzy subsets, because in practice, we can be in the presence of case where the set of the actions to be chosen is not given in known actual values but rather in the form of linguistic terms so as to have the choice between turning slightly or midway. Hence the rules which we will use will take the following form:

R_i: If x_1 is $A_1{}^i$ and ……….. and x_n is $A_n{}^i$ Then
y is B [i, 1] with q [i, 1] = 0
or
y is B [i, 2] with q [i, 2] = 0...
…………………………..
or
y is B [i, N] with q [i, N] = 0.

Where B [i, j] represents the fuzzy subset associated with the rule i and the conclusion j.

3.2. Fuzzification of Inputs and Outputs

In our case, we considered fuzzy linguistic rules with two inputs. Δθ, which is the difference between the course of the robot i. e., its orientation and the objective, (Δθ = θ - θ'), as for the second entry "d", it represents the distance to an obstacle, while the output is the orientation α which the robot must take. Hence, we have the two fuzzy subsets with their fuzzification as shown in Figure 3.



Fuzzification of d

Fuzzification of Δθ and α

Figure 3. Fuzzification of inputs and outputs.

The basic rules of simulator are given by:

R_1: If Δθ est N and d is N Then
α is N with q_{1N}
or
α is Z with q_{1Z}
or
α is P with q_{1P}

In total, we have nine rules:

R_i: If Δθ is A and d is B Then
α is N with q_{iN}
or
α is Z with q_{iZ}
or
α is P with q_{iP}

With A et B are subsets whose linguistic terms can be N (negative), Z (zero), or P (positive).

3.3. Exploration Phase

The first step in reinforcement learning is the exploration, which consists in choosing the best actions progressively. We propose the following diagram block as depicted in Figure 4, and we must follow the following algorithm:

1. Initiate the different qualities q_{iA} by number 0.
2. Repeat for a given number of period “n”.
3. Calculation of the degrees of membership of each input to the various fuzzy subsets:

$$\mu_{Aj}{}^{i}(x_j) \text{ for } j = 1 \text{ to } n \text{ and } i = 1 \text{ to } N$$

4. Calculation of the truth value of each rule, for i = 1 to N:

$$\alpha_i(x) = \min_j (\mu_{Aj}{}^{i}(x_j)) \text{ for } j = 1 \text{ to } n$$

5. Choose an action by the pseudo-stochastic method which is summarized by:
   - The action with better value of q_{iA} has a probability P of being selected.
   - Otherwise, an action is selected randomly amongst all the other possible actions in a given state.
6. Calculation of the contribution of each chosen rule by the pseudo-stochastic method:

$$\mu(\alpha) = \min(\alpha_i(x), \mu_B{}^{I}(\alpha))$$

7. Aggregation of rules:

$$\mu(\alpha) = \max_i (\mu_B{}^{i}(\alpha))$$

8. Defuzzification of the output variable:

$$\alpha = \frac{\int u\mu(u)du}{\int \mu(u)du}$$

9. Calculate the new orientation of the robot:

$$\theta = \theta + \alpha$$

10. Move the robot, and compute the variation

$$\Delta\theta = \theta - \theta'$$

11. Calculate the reinforcement:

$$r = \begin{cases} +1 & \text{if d\_ac>d\_an} \\ -1 & \text{otherwise} \end{cases}$$

Where d_ac is the current distance with respect to the objective following the displacement of the robot, as for d_an is the old distance compared to the objective i. e., the state before the displacement of the robot.

12. Update qualities of the rules which contributed to the variation of the angle α [1, 9] :

$$q_{iA} = (1 - \beta)\, q_{iA} + \alpha_i \; * r * \gamma$$

β: learning rate, γ: delay factor, α_i: truth value of rule i.

13. If “n” is reached or d_ac is small, we stop the learning process.

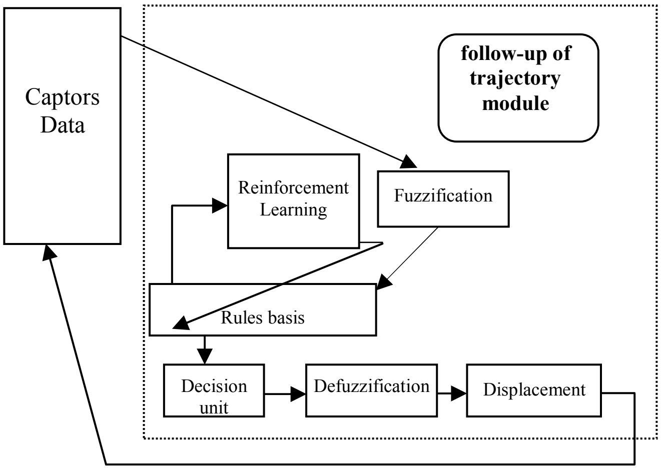


Figure 4. Follow-up of corridor module.

3.4. Exploitation Phase

The optimal policy is obtained by choosing the action which, in each state, maximizes the quality function:

$$u = \arg\max_{u \in U_x} Q^*(x, u)$$

This policy is called “greedy”. However, at the beginning of the learning, the values Q (x, u) are not significant and the greedy policy is not applicable.

For our case, the robot is set to its starting point “A”, and for each displacement, it follows the following step:

- Direct the robot towards its arrival point.
- Repeat.
- Calculate the distance “d” compared to possible obstacles.
- Move by choosing the action of best quality using the fuzzy controller.
- Until reaching the goal.

4. Simulations and Results

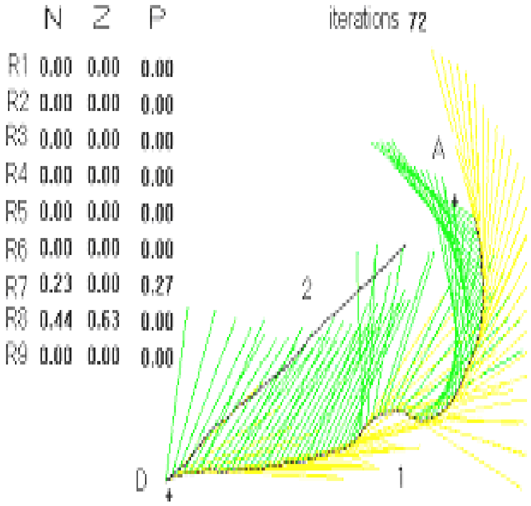
During our work, we have chooses the following coefficients:

- The probability P = 0.9, $\gamma = 0.9$ and $\beta = 0.9$
- Number of passes n = 1000
- The basic rules are given by Table 1.

From Figure 5, we can notice that the trajectory (1) follow-up by the robot at the time of the phase of exploration is different from the trajectory (2) which is carried out at the time of the exploitation phase (after learning).

Table 1. The basic rules.

d \ Δθ	N	Z	P
N	N, q_{1N} Z, q_{1Z} P, q_{1P}	N, q_{2N} Z, q_{2Z} P, q_{2P}	N, q_{3N} Z, q_{3Z} P, q_{3P}
Z	N, q_{4N} Z, q_{4Z} P, q_{4P}	N, q_{5N} Z, q_{5Z} P, q_{5P}	N, q_{6N} Z, q_{6Z} P, q_{6P}
P	N, q_{7N} Z, q_{7Z} P, q_{7P}	N, q_{8N} Z, q_{8Z} P, q_{8P}	N, q_{9N} Z, q_{9Z} P, q_{9P}



	N	Z	P
R1	0.00	0.00	0.00
R2	0.00	0.00	0.00
R3	0.00	0.00	0.00
R4	0.00	0.00	0.00
R5	0.00	0.00	0.00
R6	0.00	0.00	0.00
R7	0.23	0.00	0.27
R8	0.44	0.63	0.00
R9	0.00	0.00	0.00

Figure 5. Exploration phase (1) and exploitation phase (2).

We also notice according to the above table that only the linguistic rules 7 and 8 which contributed in this example hence the change of the coefficients of qualities of only these two rules.

Another teaching which we can draw from this table that:

- Rule 7: If (θ - θ') is N (negative) and d is P Then α is P (positive) which is favored and has the best quality (0.27).
- Rule 8: If (θ - θ') is Z (zero) and d is P Then α is Z (zero) which is favored and has the best quality (0.63).

These two results are completely logical because if the variation tends towards the negative one, it is necessary to apply a positive action to compensate this deviation, and if it is null, it is necessary to maintain the action of term zero i. e., to keep the same angle without deviation. It should be also noted that the iteration count that we needed to find these results is 72.

5. Conclusions and Perspectives

The problem of the learning is always present in the architecture of mobile robot, which led us to use an alternative of the Q-learning with fuzzy limited lexicon.

This study has allowed us to implement a fuzzy approach applied to the reinforcement learning. It gave us good results, in particular in the follow-up of rectilinear trajectory. However, the subject is not ready to be completed.

It is known that, we can always think of carrying out a certain number of works, we can retain following work:

- Realization of a robot able to learn how to avoid obstacles.
- Realization of a robot able to generate actions in conflict.
- Integration of the vague concepts, training by reinforcement learning on all the levels of the architecture of the robot and addition of some alternatives.
- Realization of a simulator allowing to code and simulate the behavior of a robot in an environment a priori unknown.

References

[1] Garcia P., Zsigri A., and Guitton A., "A Multicast Reinforcement Learning Algorithm for WDM Optical Networks," *in Pceedings of the 7th International Conference on Telecommunications-ConTEL*, Zagreb, Croatia, ISBN:953-184-052-0, June 11-13, 2003.

[2] Glorennec P. Y., "*Algorithms of Optimization for Fuzzy Inference Systems: Application to the Identification and the Order*," INSA Rennes 1998.

[3] Glorennec P. Y., Foulloy L., and Titli A., The *Reinforcement Learning, Application for Fuzzy Inference Systems, Fuzzy Order 2*, Treated IC2, Ed Lavoisier, 2003.

[4] Hiroshi I., Masatoshi K., and Toru I., "State Space Construction by Attention Control," *in Proceedings of the International Joint Conference on Artificial Intelligence*, Sweden, pp. 1131-1139, July-August 1999.

[5] Jacky B. and Yumin L., "Path Tracking Control of Non-holomonic Car-Like Robot with Reinforcement Learning," *Lecture Notes in Artificial Intelligence* 1793 MICAI 2000: *Advances in Artificial Intelligence,* Mexico, April 2000.

[6] Joo-Ho L., Guido A., and Hideki H., "Physical Agent for Sensored Networked and Thinking Space," *in Proceedings of the 1998 IEEE, International Conference on Robotics and Automation*, Leuven, Belgium, pp. 838-84, May 1999.

[7] Jouffe L., "*Training of Fuzzy Inference Systems by Reinforcement Methods: Application to the Regulation of Ambiance in a Building of Pork*

Raising," *PhD Thesis*, University of Rennes I, France, 1997.

[8] Maaref H., "*Imperfect Data Treatment in the Setting of the Fuzzy Theory: Contribution to the Navigation and the Localization of a Mobile Robot*," Memory of Authorization to Direct Research, University of Evry Paris, 1999.

[9] Mark D. P., "Reinforcement Learning in Situated Agents: Theoretical Problems and Practical Solutions," *Lecture Notes in Artificial Intelligence 1812,* Berlin, pp. 84-102, 2000.

[10] Michita I., Kazno H., and Tsutomu M., "Physical Constraints on Human Robot Interaction," *in Proceedings of the International Joint Conference on Artificial Intelligence,* Sweden, July-August 1999.

[11] Sergio U. G. and Horacio M. A., "An Application of Behavior-Based Architecture for Mobile Robots Design," *Lecture Notes in Artificial Intelligence 1793, MICAI 2000: Advances in Artificial Intelligence,* Mexico, pp. 136-147, April 2000.

Youcef Dahmani obtained his diploma of computer engineering in 1992 from the University of Sciences and Technologies Oran, Algeria and the MSc degree in 1997 from University of Es Senia Oran. Currently, he is the head of Computer Sciences Department at Ibn Khaldoun University Tiaret and a member of the Signals, Images, and Speech Laboratory. His research areas include optimization of fuzzy rules, artificial intelligence and reactive robotic systems.

Abdelkader Benyettou is a professor of electrical engineering at the University of Science and Technology of Oran (USTO), Algeria. He received the engineering degree in 1982 from the Institute of Telecommunications of Oran, and the MSc degree in 1986 from USTO. In 1987, he joined the Computer Sciences Research Center of Nancy, France, where he worked until 1991 on Arabic speech recognition by expert systems and received the PhD in electrical engineering in 1993 from USTO. His interests are in the areas of speech and image processing, Arabic speech recognition, neural networks and machine learning. He has been the director of the Signal-Speech-Image–SIMPA Laboratory, Department of Computer Sciences, Faculty of Sciences, USTO since 2002.