Neuro-Fuzzy Based Obstacle Avoidance for Autonomous Vehicle

Niken Syafitri[1], Adang Suwandi Ahmad[2], Kusprasapta Mutijarsa[2]
[1]Electrical Engineering Department, National Institute of Technology, Bandung
[2]Electrical Engineering Department, School of Electrical Engineering and Informatics, Bandung Institute of Technology
nikensyafitri@gmail.com, asaisrg@yahoo.com, soni@ee.itb.ac.id

Abstract

The research developed neuro-fuzzy based obstacle avoidance for autonomous vehicle. The method consists of fuzzy system that was equipped with supervised learning and reinforcement learning. Fuzzy system with supervised learning was divided to three method, fuzzy system with Delta Rule (DR), fuzzy system with General Delta Rule (GDR) and fuzzy system with General Delta Rule with Fuzzy Parameter Adaptation (GDRFPA). In DR, three simulations were done to this method. First, simulation was with no boundary value. Second was with boundary value. Third was with parameters that close to the needed output values. In GDRFPA, fuzzy parameter adaptation for learning rate and momentum constant were used. All of method compared to know what the fastest and accurate method in learning.

Keywords: fuzzy system, delta rule, general delta rule, reinforcement learning.

1. Introduction

Autonomous vehicle is the development of mobile robot technology. It is a moving vehicle that has autonomous navigation capability. It can sense its environment, represent its environment as its work environment model, and plan the action. In detail, it has navigation capability which consists of position estimating, environment mapping, path planning, and obstacle avoidance abilities. Autonomous vehicle also can move from the start position to the target position based on the information that obtains from its environment with no collision with obstacle all along the path.

Obstacle avoidance capability is the one of autonomous vehicle navigation capability. In usual, this capability will meet problem when the environment change dynamically. It is different with the navigation case when the obstacle is static. The dynamic environment is more difficult to be controlled. Thus, the basic problem is how to configure obstacle avoidance capability that can solve that problem.

Fuzzy system is one method that is more proposed to configure obstacle avoidance capability. This system can be applied in various situations with no environment model configuration analytically. But, each rule base in this system has certain definition for certain situation. It possible to make rule manually, but it is not easy to configure rule in unknown environment.

To solve the problem in fuzzy system, neural network can be used. It can be used to learn the rule. But, it needs much of representative sets in learning process, in order to be able characterize the environment. Else, it also difficult to get learning pattern which consist of non-contradictive input/output pair. To solve the problem, reinforcement learning was tried. This learning process was just need scalar reinforcement signal as feedback performance from the environment. In detail, scalar reinforcement signal in the form of reward or punishment was given to the system to tell that obstacle avoidance can be done well or not.

Therefore, a neuro-fuzzy method of this research was expected, which could solve the problems in obstacle avoidance capability of autonomous vehicle. There were many of neuro-fuzzy method that could be implemented. But in this research the fuzzy system, delta rule, general delta rule, general delta rule with fuzzy parameter adaptation and reinforcement learning methods were used. They were compared to get the fastest method in learning process, and achieved the minimization criteria.

2. Neuro-Fuzzy Based Obstacle Avoidance

The software of obstacle avoidance used C++ language programming, and it only could use for three sensor groups. Obstacle avoidance system in autonomous vehicle was shared to two parts. First, fuzzy system and supervised learning, they were used in offline learning. Second, fuzzy system and reinforcement learning, they were used in online learning later. The cylindrical autonomous vehicle was used to get the model in this research. Seven sensors were set in half of front circle. They were share to three sensor groups. Fuzzy system was used to process the received inputs. The inputs here were the obstacle distance. The process result was the heading angle of vehicle. Supervised learning was used to update x_{ri} parameter (input membership function parameter of fuzzy system) and b_{2j} (output membership function of fuzzy system). And the reinforcement learning was used to update b_{2j} parameter.

This input membership function that is illustrated in Figure 1 was used for fuzzy system.

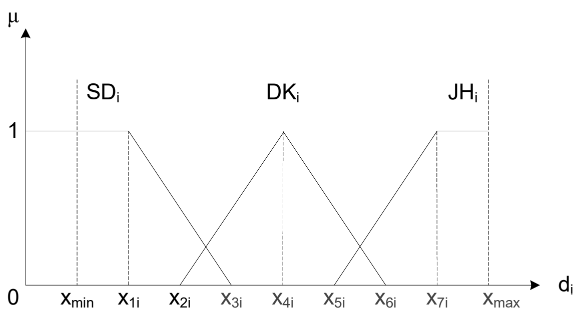

Figure 1. Input membership function

With SD is *Sangat Dekat* (very near), DK is *Dekat* (near), and JH is *Jauh* (far). It represents that the input membership function consists of three functions for each sensor group, which is denoted with i in x_{ri}. Then output membership function is illustrated in Figure 2.

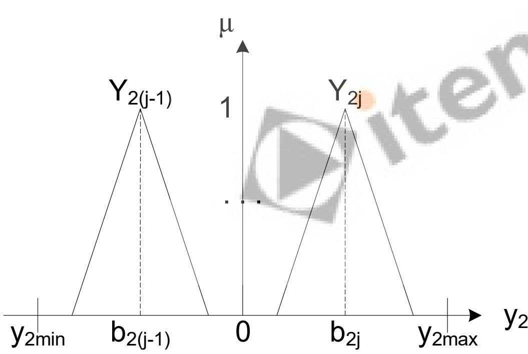

Figure 2. Output membership function

With b_{2j} is center point of output value. The number of center point is depend on input membership function – each show very near, near, and far – and the number of sensor groups. For example, there is three sensor groups, then the number of center point are 3^2=27, or if its number is five, then the center points number are 3^3=243.

Fuzzy system rules were formulated with the statement:

IF d_1 is D_{j1} AND d_2 is D_{j2} AND d_3 is D_{j3} THEN y_2 is Y_{2j},
With j=1,...,27. Larsen's product inference was used for fuzzy inference system, as proposed too in [1][2][3]. Height defuzzification was adopted for the defuzzification process.

For supervised learning, there were use three methods which would be compared. They were Delta Rule (DR), General Delta Rule (GDR), and General Delta Rule with Fuzzy Parameter Adaptation [1]. The minimum objective function was purposed in supervised learning, and used the equation (1).

$$J = \frac{1}{2}\left(y_2 - y_{2d}\right) \tag{1}$$

With y_2 is steering angle which was produced from the calculation, and y_{2d} was output target that was needed.

For the Delta Rule, the change of $x_{ri}(k+1)$ and $b_{2j}(k+1)$ parameter used the equations (2) and (3).

$$x_{ri}(k+1) = x_{ri}(k) - \rho\eta \frac{\partial J}{\partial x_{ri}}(k) \tag{2}$$

$$b_{2j}(k+1) = b_{2j}(k) - \rho\eta \frac{\partial J}{\partial b_{2j}}(k) \tag{3}$$

With ρ is a constant which was obtained from the de-normalization. And η is the learning rate.

For the General Delta Rule, the change used the (4) and (5) equations.

$$x_{ri}(k+1) = x_{ri}(k) + \alpha\Delta x_{ri}(k-1) - \rho\eta \frac{\partial J}{\partial x_{ri}}(k) \tag{4}$$

$$b_{2j}(k+1) = b_{2j}(k) + \alpha\Delta b_{2j}(k-1) - \rho\eta \frac{\partial J}{\partial b_{2j}}(k) \tag{5}$$

With α is the momentum constant.

For the General Delta Rule with Fuzzy Parameter Adaptation, the same equations with General Delta Rule were used. But α and η values were adapted using fuzzy system. The input and output definitions for α and η adaptation are illustrated in Figure 3 and 4. And the rules are described in Table 1.

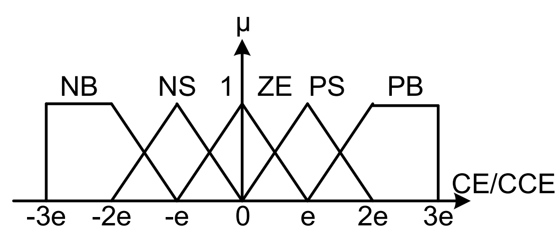

Figure 3. Input membership function of parameter adaptation

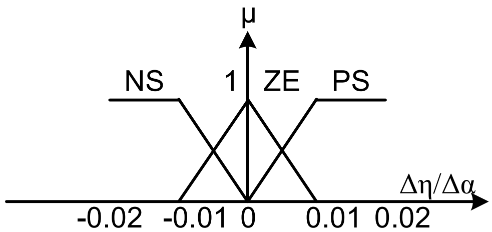

Figure 4. Output membership function of parameter adaptation

Table 1. Parameter adaptation rules

CCE \ CE	NB	NS	ZE	PS	PB
NB	NS, NS	NS, NS	NS, ZE	NS, ZE	NS, ZE
NS	NS, NS	ZE, ZE	PS, ZE	ZE, ZE	NS, ZE
ZE	ZE, ZE	PS, PS	PS, PS	PS, PS	ZE, ZE
PS	NS, ZE	ZE, ZE	PS, ZE	ZE, ZE	NS, NS
PB	NS, ZE	NS, ZE	NS, ZE	NS, NS	NS, NS
	Δη,Δα				

Reinforcement learning steps calculated the external reinforcement signals, then the prediction value $p_m(t)$, internal reinforcement signal, fired strength trace of j rules, then ACE weight, ASE weight, eligibility trace and center point in output membership function.

3. System Test and Analysis

For system test, the first x_{ri} was used as described in Table 2.

Table 2. First x_{ri} values

x_{1i}	x_{2i}	x_{3i}	x_{4i}	x_{5i}	x_{6i}	x_{7i}
25	30	50	55	60	80	85

Then the first b_{2j} values for the first test were -45° to 45°, and the second test were -90° to 90°. Output target value in second test was twice from the first test.

For the Delta Rule test, three way tests were applied. They were the fuzzy system method with Delta Rule with no boundary for x_{ri}, with boundary for x_{ri}, and with boundary for x_{ri} with the b_{2j} value close to output value that needed. Input-output pair for the test was (45, 55, 45)-(-0.23). Result test with no boundary showed that computation is always divergent like illustrated in Figure 5.

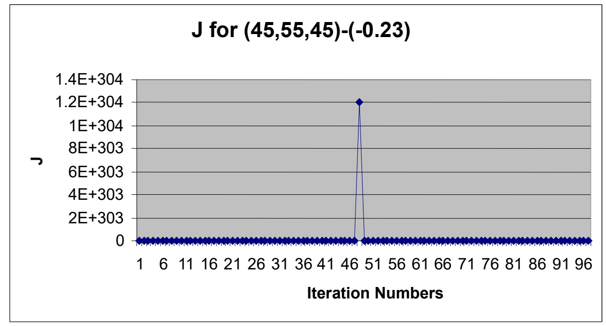

Figure 5. J graphic for DR with no boundary

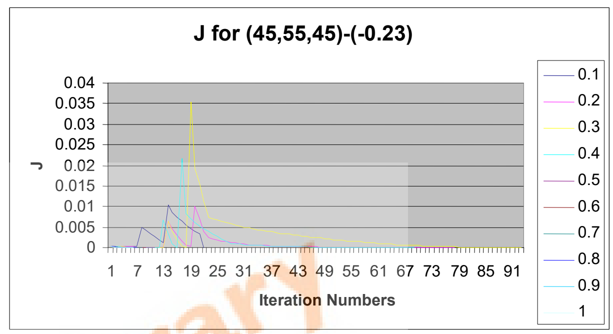

Figure 6. J graphic for DR with boundary

Test result for system with boundary showed that minimum criteria value could be reached, like illustrated in Fig 6. The safe minimum value for first test was 10^{-3}, and for second test was 10^{-2}.

Test result for system with b_{2j} close to needed output showed that most of cases minimum criteria could not be reached, but there was some that can be reached, like illustrated in Fig 7.

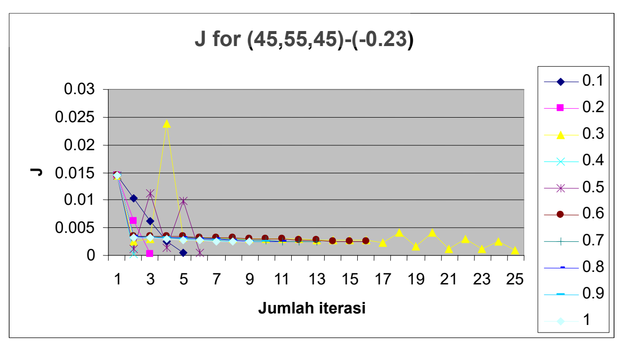

Figure 7. J graphic for DR with b_{2j} close to needed value

For the General Delta Rule method, the minimum criteria 10^{-5} could be reached for the first test, and 10^{-4} for the second test. If GDR was compared with DR, it had smaller iterations, like illustrated in Fig 8.

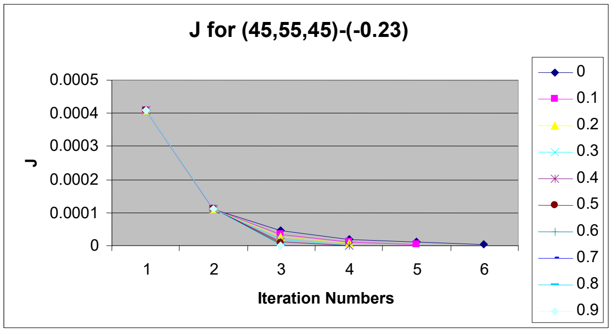

Figure 8. J graphic for GDR

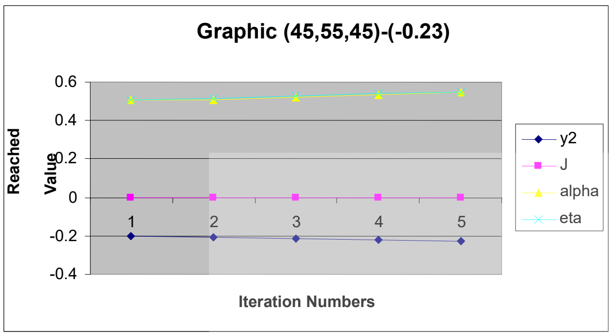

Figure 9. J graphic for GDRFPA

For the General Delta Rule with Fuzzy Parameter Adaptation test, the first η and α value are 0.5. But there were some cases that the first values had to be changed. For 0.5, the changing of η and α was equivalent. For this method, iterations that were done were faster.

For the reinforcement learning, the chosen inputs were 45, 80 and 35. Simulation showed that it could be used.

While ten iterations test was done for inputs d[1]=45 and d[3]=35 continually, and d[2]=40, 39, 38, 37, 36, 35, 34, 33, 32 and 31. y_2 output value was always zero. p_2 and r_{hat} were always showed the changing. r_2 was always -1, meant the external reinforcement signal always indicated that collision was always happened.

4. Conclusion

Based on the simulation tests of obstacle avoidance system, these conclusions could be taken as follow:

1. For fuzzy system with Delta Rule, which was done with three way tests, there were:
   - Fuzzy system with Delta Rule with no boundary for x_{ri} could not be used for obstacle avoidance system, because computations always showed the divergent results.
   - Fuzzy system with Delta Rule with boundary for x_{ri} could be used to obstacle avoidance system. Boundary values could prevent overflow in computations. Then there was inclination that big η value would decrease iterations number in learning process. The smaller η would make smoother weight changing and minimum criteria, although the number of iterations was larger. The changing of b_{2j} parameter and output target in second test would increase minimum criteria.
   - Fuzzy system with Delta Rule with b_{2j} close to needed target value could not be used because in most of cases minimum criteria were too larger. For succeed case, there was inclination that the larger η would decrease the number of iterations in learning process.
2. For fuzzy system with General Delta Rule, there were:
   - Inclination that the larger α would decrease the number of iterations.
   - Iterations number that was needed for learning process is smaller than fuzzy system with Delta Rule.
   - Accuracy for this method was larger than fuzzy system with Delta Rule.
   - For the second test – b_{2j} parameters were changed – minimum criteria were larger.
3. For fuzzy system with General Delta Rule with Fuzzy Parameter Adaptation, there were:
   - Iteration number was smaller than two methods before.
   - The changing of weights and minimum criteria was smoother than two methods before.
   - Accuracy for this method was larger than two methods before.
   - In most of cases, the changing of η was equivalent with the changing of α.
   - For some non-succeed cases η could not fill $0<\eta\leq 1$.
   - For the second test, minimum criteria were larger.
4. For fuzzy system with reinforcement learning, there were:
   - y_2 was always zero, p2 and r_{hat} always changed, r_2 always -1 – meant external reinforcement signal indicated that collision was always happened.
   - Collision in simulation cause of the fast computation cycle for a cycle and some parameters needed to be changed.

References

[1] Ye, C., Yung, N.H.C., Wang, D., A Fuzzy Controller with Supervised Learning Assisted Reinforcement Learning Algorithm for Obstacle Avoidance, *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics,* Vol. 33, No. 1, 17 – 27, February, 2003.

[2] Yung, N.H.C., Ye, C., An Adaptive Fuzzy Approach to Obstacle Avoidance, *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics,* San Diego, 3418 – 3423, 1999.

[3] Yung, N.H.C., Ye, C., An Intelligent Mobile Vehicle Navigator Based on Fuzzy Logic and Reinforcement Learning, *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics,* Vol. 29, No. 2, 314 – 321, April, 1999.