The Research of Motion Planning of Humanoid Robot

Piao Songhao, Hong Bingrong, Wang Liang, Geng Zhen, Cai Zesu, Huang Qingcheng,
School of Computer Science and Technology, Harbin Institute of Technology, Harbin 15001, China
(Tel: 86-451-6413388; Fax : 86-451-6221048; E-mail: piaosh@hit.edu.cn)

Abstract: This paper proposed a strategy for humanoid robot motion planning using GA-based fuzzy neural network controller, which is stable self-learning fuzzy neural networks control system. The system is composed of two parts: (1) A FNNC which use GA to search optimal fuzzy rules and membership function; (2) A supervisor which use gradient learning algorithm to train the network weights. And we apply this controller to robot motion planning. The simulation result shows its effectiveness.

Keywords: Motion planning, Fuzzy neural network, GA algorithm, Humanoid robot

1. Introduction

One of the most important problems in mobile robot control is motion planning. There are some methods that solve motion planning problems, such as artificial potential method. In this paper, we use fuzzy neural network method to control robot motion planning.

Fuzzy controllers can work in the situation where there is a large uncertainty or unknown variation. Many studies regarding finding the rules and tuning the membership function parameters of fuzzy models have been reported [5]-[10]. In order to obtain more powerful learning capacity, the fuzzy neural network (FNN) is proposed, which has very strong robustness and adaptability [1]-[4]. The FNN, in the form of feedforward multilayer network, combines the fuzzy logic controller and neural network structure to form an integrated neural-network-based-fuzzy logic control and decision system. In the FNN, the fuzzy rules and membership function may be expressed by the weights and node parameters, and the optimal value can be obtained through learning algorithm [17].

Karr, Freeman, Meredith [11] proposed a combination of fuzzy logic and Genetic Algorithm (GA). Unlike the neural network learning, GA does not need the teaching signal to obtain the fuzzy controller [14]. GA finds fuzzy rules using the payoff for the success/failure of its control actions. GA was applied to identification of hierarchical structure of fuzzy model [12] from given input-output pairs of data. S. Matsushita and A. Kuromiya [13] have applied GA to selection of input variables of FNN hierarchical model. The method is effective in the case where the target system has a strong nonlinearity.

This paper presents a fuzzy neural network controller (FNNC) by a new GA-based method. The genetic-based fuzzy approach presents powerful optimization and synthesis tools for a controller design. At last we apply this controller to soccer robot motion planning.

2. The Fuzzy Controller

A fuzzy controller can be built using a multilayer fuzzy neural network, as shown in Fig.1. The Fuzzy Neural Network Controller (FNNC) structure is shown in Fig.2. The control system has a total of four layers. A controller with two input variables (e, △e) and a single output (y) is considered here for convenience. Here, e is error signal and △e is change of the error signal.

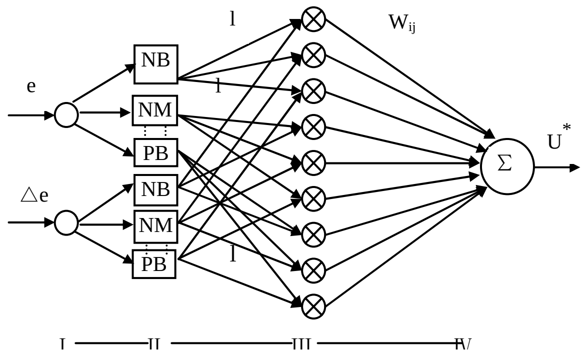

Fig.1. Topology of the fuzzy controller

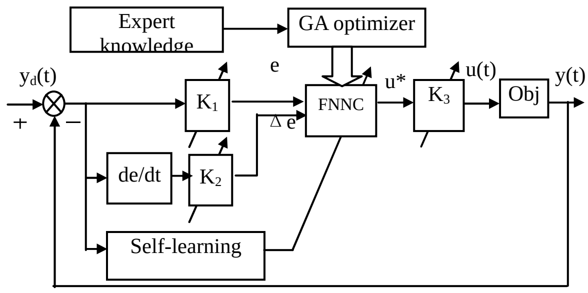

Fig.2. The fuzzy control

The controller gives control output y by the "IF-THEN" rules which are represented as follows:

$$\text{Rule i: IF e is } A_{i1} \text{ and } \Delta e \text{ is } A_{i2} \text{ THEN y is } w_i \tag{1}$$

$$y = \frac{\sum_{i=1}^{n} u_i * w_i}{\sum_{i=1}^{n} u_i} = \sum_{i=1}^{n} \overline{u_i} * w_i \tag{2}$$

Where A_{i1} and A_{i2} (i =1,…,n) is the membership functions in the antecedent part, and w_i is a real number in the consequent part. The result of fuzzy reasoning y for the input e and Δ e is calculated as follows:

$$u_i = A_{i1}(e) * A_{i2}(\Delta e) \tag{3}$$

$$\overline{u_i} = \frac{u_i}{\sum_i u_i} \tag{4}$$

Where u_i is a membership value of i-th inference rule.

Accordingly, there are two nodes in layer Ⅰ and one node in layer Ⅳ. Nodes in layer Ⅰ are input nodes that directly transmit input signals to layer Ⅱ. Layer Ⅳ is the output layer. Nodes in layer Ⅱ is a "term nodes" and they act as membership functions to express the input and output fuzzy linguistic variables. In the premise part, the Gaussian function is adopted here as a membership function. Where a_{i1} , a_{i2} are the centers of the Gaussian function, and b_1 ,b_2 are the widths.

$$A_{i1}(e) = \exp\left[-\left(\frac{x_i - a_{i1}}{b_1}\right)^2\right] \tag{5}$$

$$A_{i2}(\Delta e) = \exp\left[-\left(\frac{x_i - a_{i2}}{b_2}\right)^2\right] \tag{6}$$

The two fuzzy sets of the first and second input variables consist of n_1 and n_2 linguistic terms, respectively. The linguistic terms, such as positive big (PB), positive medium (PM), positive small (PS), zero (ZE), negative small (NS), negative medium (NM), negative big (NB), are numbered in the term nodes. Hence, $n_1 + n_2$ nodes are included in layer Ⅱ.

Each node of layer Ⅲ is a "rule node" and represents a single fuzzy control rule. In total, there are $n_1 \times n_2$ nodes in layer Ⅲ to form a fuzzy rule base for two linguistic input variables.

3. The Hybrid Learning Scheme

3.1 FNNC Learning Method Based GA

GA is a probabilistic and stochastic algorithm based around Darwinian principal of survival, i.e., survival of the fittest [15]. It employs heuristic search methods based on a model of natural evolution developed in population genes. By means of probabilistic operations such as crossover and mutation, the space of solution can be widely searched [16]. In this section, the proposed means is used to optimize the inference rules and the shapes (a_{i1} , a_{i2} , b_1 and b_2) of the membership functions by GA. The following steps are employed to generate and handle a set of strings (population).

Step 1. Initialization: The first step in GA is coding, which maps a finite-length string to the searched parameters. We first generate an initial population containing N strings, where N is the number of strings in each population. In this paper, the binary coding scheme is used, because it is easy and quick. Each string represents a set of possible inference fuzzy rules.

Step 2. Fitness function: In the step, each string is decoded by an evaluator into an objective function. This function value, which should be maximized by the GA, is then converted to a fitness value. Here, the fitness function is defined by

$$F(x_i) = 1/E \tag{7}$$

$$E = \frac{1}{2}\sum\left(\overline{u_i} - u_i\right)^2 \tag{8}$$

Where E represents the root-mean-square error between the network outputs and the desired outputs for the string. The aim of the GA optimizer is to maximize the above fitness function.

Step3. Reproduction: Reproduction is a process in which individual strings are copied according to their fitness values, i.e. based on the principle of survival of the fittest. This operator is an artificial version of natural selection. Through reproduction, strings with high fitnesses receive multiple copies in the next generation while strings with low fitnesses receive fewer copies or even none at all.

Step 4. Crossover. Reproduction directs the search towards the best existing individuals but does not create any new individuals. In nature, an offspring is rarely an exact clone of a parent. It usually has two parents and inherits genes from both. The main operator in GA to work on the parents is crossover, which happens for the selected pair with a crossover probability. In the paper, we use the two-point crossover population are mated at random, and two crossover points (two bits position) are randomly selected. This reproduction stage takes place with a probability of p_c.

Step 5. Mutation: Mutation operation is an operation that makes the genetic search random and it involves modification of genes to their opposite value in a random fashion with a probability of p_m. In the paper, we select the two-point mutation operation, and two mutation points (two bits position) are randomly selected.

Step 6. In the proposed GA algorithm, the stopping

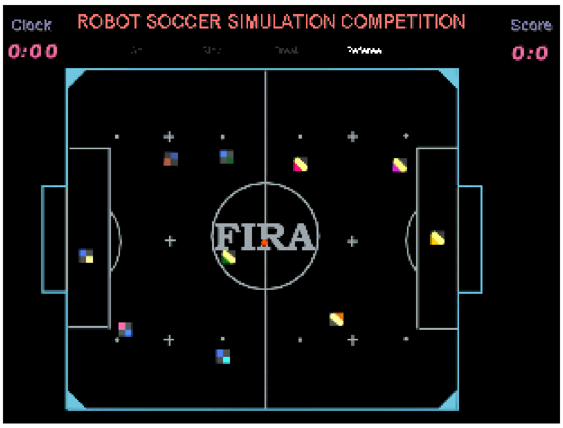

Fig.3. the interface of simulation platform

criterion is the execution of a certain number of generations without any improvement in the best fitness value. Finally, proper fuzzy logic rules are obtained by encoding the best string into a set of fuzzy rules.

3.2 FNNC Learning Method Based Gradient Descent

The FNNC system uses gradient descent algorithm to train the local parameters of network weights. The cost function J (W) is defined as follows:

$$J(W) = 1/2\,[\varepsilon(W,k)]^2 = 1/2\,[Y(k) - \ddot{Y}(k)]^2 \qquad (9)$$

Y(k) is the desired output, and Ÿ(k) is the actual output. ε (W, k) is the error. Then the gradient of error in Eq.(8) with respect to an weighting vector W becomes

$$\frac{\partial J(W)}{\partial W} = -\varepsilon(W,k)\frac{\partial \ddot{Y}(W,k)}{\partial W} \qquad (10)$$

$$W(k+1) = W(k) + \eta(k)\left(-\frac{\partial J(W)}{\partial W}\right) \qquad (11)$$

The weight can be adjusted by using a gradient method. Where η is a learning rate.

4. Simulation

Because the linguistic terms, such as positive big (PB), positive medium (PM), positive small (PS), zero (ZE), negative small (NS), negative medium (NM), negative big (NB), are used, there are 49 rules (see Table 1) in the system.

Table 1. Fuzzy control rules

e \ w	Δ e						
	N B	N M	N S	Z O	P S	P M	P B
N B	- 6.0	- 6.0	- 4.0	- 6.0	- 4.0	- 4.0	-4.0
N M	- 6.0	- 4.0	- 2.0	- 4.0	- 4.0	- 4.0	-2.0
N S	- 4.0	- 2.0	- 2.0	- 2.0	0.0	2.0	4.0
Z O	- 6.0	- 4.0	- 2.0	0.0	2.0	4.0	6.0
P S	- 4.0	-2.0	0.0	2.0	2.0	2.0	4.0
P M	2.0	4.0	4.0	4.0	2.0	4.0	6.0
P B	4.0	4.0	4.0	6.0	4.0	6.0	6.0

The values in the table are set as the initial connection weights W for the FNNC. The central points of the fuzzy sets NB, NM, NS, ZE, PS, PM, PB are –6, -4, -2, 0, 2, 4, 6, respectively. The width values of the membership function are all unity.

In order to test our proposed method, we develop a simulation platform using VC++6.0. In the test we let one robot avoid one or some robots (see Fig. 3).

5. Conclusion

This paper proposed a strategy for humanoid robot motion planning using GA-based fuzzy neural network controller. The control system is composed of two parts. A FNNC uses GA to search optimal fuzzy rules and membership function. And a supervisor uses gradient learning algorithm to train the network weights. We apply this controller to robot motion planning. The simulation result shows that the method is effective in optimization and good in performance.

References

[1]Glenn Healy and David Slater, “Computing Illumination-Invariant Descriptors of Spatially Filtered Color Image Regions”, IEEE Trans. Image Processing, vol.6, no. 7, pp.1002-1013, 1997

[2]Heng-Da Cheng and Ying Sun, “A Hierarchical Approach to Color Image Segmentation Using Homogeneity”, IEEE Trans. on Image Processing, vol.9, no.12, pp.2071-2082, 2000

[3]Brian V. Funt and Graham D.Finlayson, “Color Constant Color Indexing”, IEEE Trans. Pattern Anal., vol.17, no. 5, pp.522-529, 1995

[4]Yong-Beom Lee, Bum-Jae You and Seong-Whan Lee, “A Real-time Color Based Object Tracking Robust to Irregular Illumination Variations”, Proceeding of the 2001 IEEE International Conference On Robotics & Automation, pp. 1659-1664

[5]Gi-jeong Jang and In So-Kweon, “Robust Object Tracking Using an Adaptive Color Model”, Proceeding of the 2001 IEEE International Conference On Robotics & Automation, pp. 1677-1682

[6]J.Yang and A.Waibel, “A Real-time Face Tracker”, Proceedings of IEEE Workshop on Applications of computer Vision, PP.142-147, 1996

[7]J.Yang, W. lu and A.Waibel, “Skin-Color Modeling and adaptation”, Proceedings of ACCV’98, vol. 2, pp. 687-694, 1998

[8] C.Amoroso, E.Ardizzone, V..Morreale, and P.Storniolo, “A New Technique for Color Image Segmentation”, Proceedings of International Conference on Image Analysis and Processing, pp. 352 –357, 1999

[9]S.Feyer and A.zell, “Detection, Tracking, and Pusurt of Humans with a Autonomous Mobile Robot”, Proceedings of IEEE International Conference on Intelligent Robots and Systems, pp. 864-869, 1999

[10] Chi Zhang and P. Wang, “A New Method of Color Image Segmentation Based on Intensity and Hue Clustering”, Proceedings of 15th International Conference on Pattern Recognition, vol.3, pp. 613 –616, 2000

[11]Rolf Adams and Leanne Bischof, "Seeded Region Growing", IEEE Trans. Pattern Anal., vol.16, no.6, pp.641-647, 1994

[12]R.Kyle Justice and Ernest M.Stokely, "3-D Segmentation of MR Brain Images Using Seeded Region Growing", 18th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, vol.3, pp. 1083 –1084, 1997

[13]Hong Peow Ong and J.C.Rajapakse, "Fuzzy-region-segmentation", Proceedings of International Joint Conference on Neural Networks, vol. 2 , pp. 1374 -1379 2001

[14] D.Guliato, R.M.Rangayyan, W.A.Carnielli, J.A.Zuffo, and J.E.L.Desautels, "Segmentation Of Breast Tumors in Mammograms by Fuzzy Region Growing" Proceedings of the 20th Annual International Conference of the IEEE Engineering in Medicine and Biology Society , vol. 2, pp. 1002 –1005, 1998

[15] S. A. Hojjatoleslami and J. Kittler, "Region Growing: A New Approach", IEEE Trans. Image Processing, vol.7, no. 7,

[16]M.I.Chowdhury and J.A.Robinson, "Improving Image Segmentation Using Edge Information", Conference on, Electrical and Computer Engineering, vol. 1, pp. 312 –316, pp.1079-1084, 1998

[17]Mie Sato, S.Lakare, Ming Wan, A.Kaufman, and M.Nakajima, "A Gradient Magnitude Based Region Growing Algorithm for Accurate Segmentation**",** Proceedings of 2000 International Conference on Image Processing, vol.3, pp. 448 – 451, 2000