Automatic Fuzzy Rule Generation by Simulating Human Knowledge Gathering Process
사람의 지식 축적과정 모사를 통한 자동 퍼지규칙의 생성

Sung Hoon Jung*
정 성 훈*

ABSTRACT

Fuzzy rules, developed by experts thus far, may be often inconsistent and incomplete. This paper proposes a new methodology for automatic generation of fuzzy rules which are nearly complete and not inconsistent. This is accomplished by simulating a knowledge gathering process of humans from control experiences. This method is simpler and more efficient than existing ones. It is shown through simulation that our method even generates better rules than those generated by experts, under fine tuned parameters.

요 약

지금까지 전문가들에의해 만들어진 퍼지규칙들은 종종 모순되었고 불완전하였다. 본 논문에서 우리는 모순되지 않고 거의 완전한 퍼지규칙을 자동생성 하는 새로운 기법을 제안한다. 이러한 퍼지규칙의 자동생성은 제어경험으로 부터 퍼지규칙을 얻어내는 사람의 지식 축적과정을 모사한 방법이다. 이 방식은 기존의 방식들보다 보다 간단하면서도 보다 효율적인 방법이다. 잘 조절된 파라메터상에서 우리의 방식이 전문가들보다 더욱 좋은 퍼지규칙을 생성함을 시뮬레이션을 통하여 보인다.

Ⅰ. Introduction

Development of fuzzy rules is one of the most important parts in designing a fuzzy logic controller [1]. Such rule sets are made by experts in the human-in-loop system. However these rule sets may be incomplete and inconsistent [2]. Several approaches to automatically generate fuzzy rules have been introduced from the early 1990s [3, 4, 5, 6, 7]. However, these methods are quite complex and cannot guarantee a dependable performance of the control. This paper introduces a new methodology for automatic generation of fuzzy rules in an off-line fashion by simulating a knowledge gathering process of humans from control experiences. With knowledges of nonlinear plants, experts are able to control them. The knowledge is gathered from their previous experiences while trying to control the plants. Before human has the knowledges, they know only very simple knowledges. However, they will obtain more and more knowledges about the controlled plant according as the control experiences are gathered more

* 전기 및 전자공학과 한국과학기술원
Department of Electrical Engineering Korea Advanced Institute of Science and Technology

and more. In our algorithms, gathered information will be represented as a set of fuzzy rules for a fuzzy logic controller. This generated set of fuzzy rules fits the control dynamics of the fuzzy logic controller into the nonlinear plant control. Simulation results show that our method generates better rules than those generated by experts, under fine tuned parameters.

Ⅱ. Human Control Knowledge Gathering Process

All humans have some basic innate reasoning power. For example, if a fuzzy positive error exists, we would apply a fuzzy positive control input; if a fuzzy negative error exists, we would apply a fuzzy negative control input. We also try to reduce a steady state error as small as we can. The control results are used to make the control dynamics be more fitted in the nonlinear plants. Thus, the control dynamics is accumulated according to the progression of the control process. With this accumulated information, we can better control the nonlinear plants. To fully understand the dynamics of a plant, we should experience many set points in the operating range.

Ⅲ. Automatic Fuzzy Rule Generation

The innate control operation of humans, mentioned in previous section, is very similar to that of a PI-type controller. However, the control values of humans cannot exactly be the same as those of the a PI-type controller because the control operation of humans are fuzzy. This fuzziness of control inputs can be simulated by adding a perturbation factor using Gaussian random noise to the PI-type control value. As a consequence, the innate human control is simulated on a computer with a *PI-plus Perturbation* control.

This type of human control can be simulated on a computer with a very similar manner to a PI-type controller. That is, this controller just inputs a fuzzy positive value when the current error is fuzzy positive, and inputs a fuzzy negative value when the current error is fuzzy negative. Humans also try to reduce the steady state error. To make fuzzy control inputs, we add a perturbation factor to the PI-type control value using Gaussian random noise.

To cover the entire operating range, a sufficient number of time series of controlled results should be obtained. Such time series contain partial information about the dynamics of a controlled plant. This information should be combined to makeup a nearly complete set of dynamics of the plant. In this paper, our method simply adopts the average values of all data in correspondence with the error and change of error parts of gathered data. Algorithm 1 shows a time table generation algorithm.

Using the n time series, fuzzy rules are generated as shown in Algorithm 2.

This method can be applied to any nonlinear plants if the closed loop system does not unstable at all set points.

Ⅳ. Simulation Results and Discussion

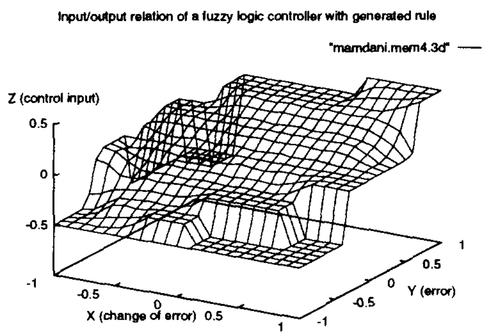
Three highly nonlinear plants are employed to show the performance of our method. These nonlinear plants are given by difference equations as:

$$\textit{plant 1}: y(k+1) = \frac{y(k)}{1+y^2(k)}\ u(k) + u(k) \tag{1}$$

$$plant\ 2\ :\ y(k+1) = \frac{y(k)}{1+y^2(k)}\ exp(u(k)) + u(k) \tag{2}$$

$$plant\ 3\ :\ y(k+1) = \frac{y(k)}{1+y^2(k)}\ sin(u(k)) + u(k) \tag{3}$$

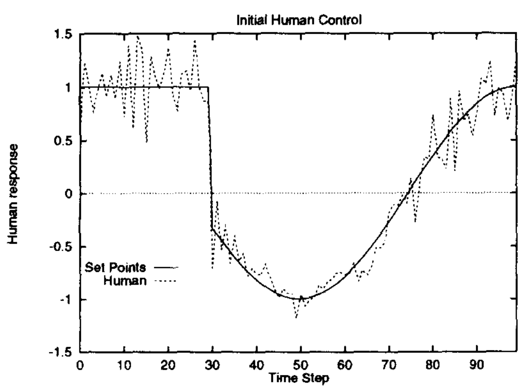
Set points are randomly selected from -1.0 to 1.0 with uniform distribution. The number of control length for one time series is set to 100 and the number of set points for gathering statistics is set to 100. We use the Mamdani's inference method and the LGM(Level Grading Method) defuzzification method [8] to implement a fuzzy logic controller. Seven linguistic values, *NB, NM, NS, ZO, PS, PM, PB*, are also employed. Figure 1 2 3 show the simulation results of the three nonlinear plants, respectively. Each figure (a) shows the nonlinear control dynamics of the fuzzy logic controller, which is generated by the fuzzy rules shown in each figure (b). Each figure (c) shows the control result of a human control, i.e., $PI + Perturbation$ and (d) shows the control result of the fuzzy logic controller using generated rules. The nonlinear mapping enables a fuzzy logic controller to adequately control the highly nonlinear plant.



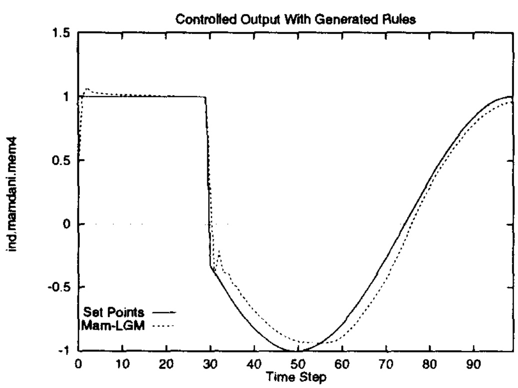
(a)

e \ ce	nb	nm	ns	zo	ps	pm	pb
nb	nb	nb					
nm	nb	nm	nm				
ns	zo	ns	ns	ns	ns		
zo		zo	zo	zo	zo	zo	zo
ps		ps	ps	ps	ps	ps	zo
pm			pm	pm	pm	pm	
pb						pb	pb

(b)

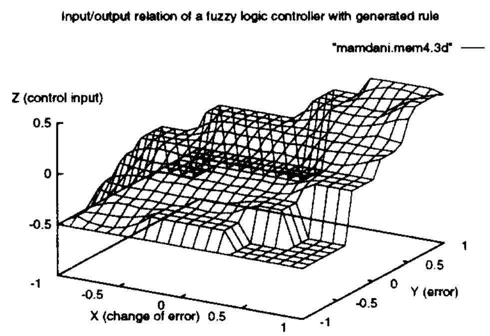


(c)



(d)

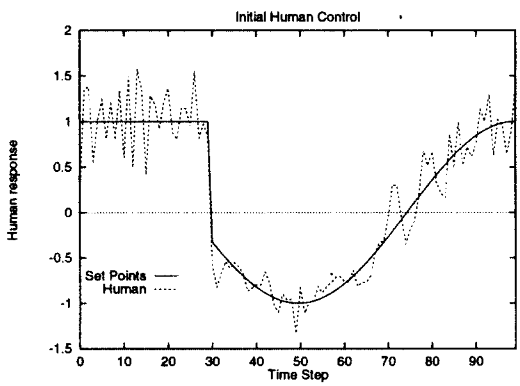
Fig. 1 [PLANT 1] (a) Input/Output relation of a fuzzy logic controller, (b) Generated rule table, (c) Simulated human control (d) Controlled output of an FLC using generated rules



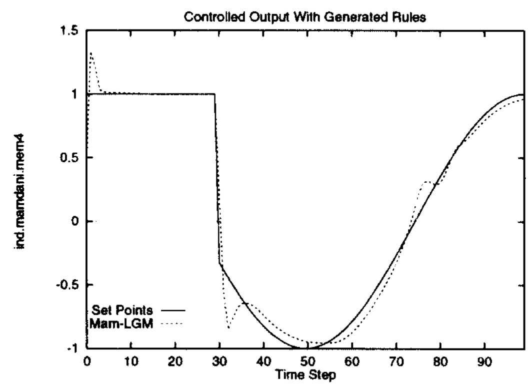
(a)

e \ ce	nb	nm	ns	zo	ps	pm	pb
nb		nb					
nm		nm	nm	nm			
ns		ns	ns	ns	ns		
zo		zo	zo	zo	zo	zo	
ps			ps	ps	ps	ps	ps
pm					pm	pm	pb
pb						pb	

(b)

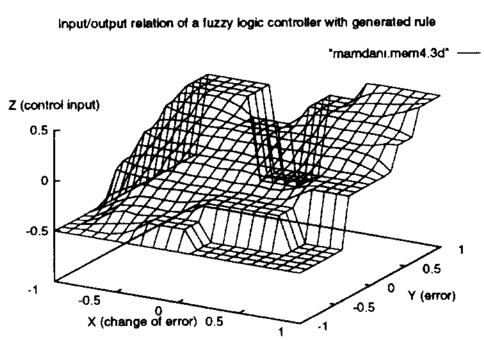


(c)



(d)

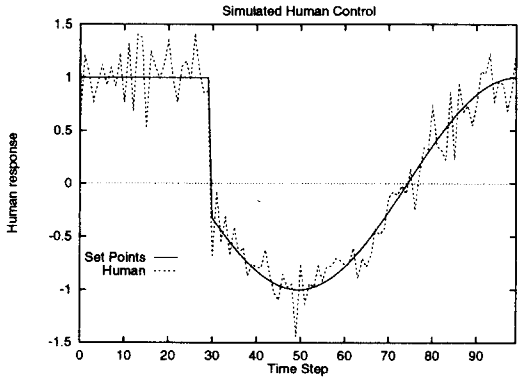
Fig. 2 [PLANT 2] (a) Input/Output relation of a fuzzy logic controller, (b) Generated rule table, (c) Simulated human control (d) Controlled output of an FLC using generated rules



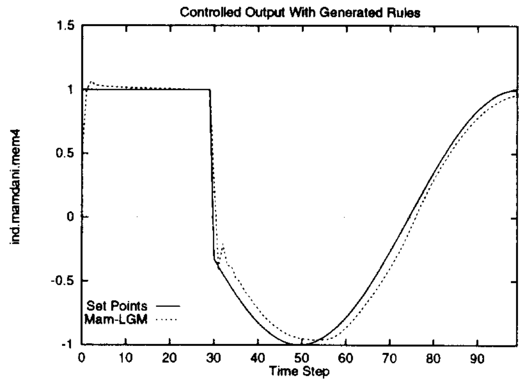
(a)

e \ ce	nb	nm	ns	zo	ps	pm	pb
nb	nb	nb					
nm	nb	nm	nm				
ns		ns	ns	ns	ns		
zo		zo	zo	zo	zo	zo	zo
ps		ps	ps	ps	ps	ps	
pm		pm			pm	pm	
pb						pb	pb

(b)



(c)



(d)

Fig. 3 [PLANT 3] (a) Input/Output relation of a fuzzy logic controller, (b) Generated rule table, (c) Simulated human control (d) Controlled output of an FLC using generated rules

The simulation results show that our method generates more efficient fuzzy rules which map well the control dynamics of the plant even for highly nonlinear plants. These results are better than those produced by human experts. This is because computers are more accurate in detecting all dynamics of the plant. However, man can adapt to the dynamic variation of the plant itself or its environment in an on-line fashion.

V. Conclusion

This paper proposes a new fuzzy rule generation paradigm which is based on the knowledge gathering process of humans. First, our algorithm produces a set of time series using a PI-type like control method with perturbation, and then generates the fuzzy rules using these time series. Our method is applied to a highly nonlinear plant to measure the performance. Although our method is simpler and more tractable than existing approaches, the simulation results are excellent for some parameters. More elaborate operations to extract the dynamics and adapt to the environment are left for future research.

참 고 문 헌

1. Witold Pedrycz, *Fizzy Control and Fuzzy Systems*, Research Studies Press, 1989.
2. Wonseek Yu and Zeungnam Bien, "Design of Fuzzy Logic Controller with Inconsistent Rule Base," *5th IFSA World Congress*, vol. 2, pp. 1370-1373, 1993.
3. Chin-Teng Lin and C. S. George Lee, "Neural-network-based fuzzy logic control and decision system," *IEEE Transactions on Computers-Special Issue on Artificial Neural Networks*, vol. 40, no. 12, pp. 1320-1336, dec 1991.
4. Jyh-Shing R. Jang, "Self-Learning Fuzzy Controllers Based on Temporal Back Propagation," *IEEE Trans. on Neural Networks*, vol. 3, no. 5, pp. 714-723, Sept. 1992.
5. Chuen-Chien Lee, "Intelligent Control Based on Fuzzy Logic and Neural Net Theory," *Proceedings of the International Conference on Fuzzy Logic*, pp. 759-764, July 1990.

6. Chin-Teng Lin and C.S. George Lee, "Neural-Network-Based Fuzzy Logic Control and Decision System," *IEEE Trans. on Computers*, vol. 40, no. 12, pp. 1320-1336, Dec. 1991.
7. Hamid R. Berenji and Pratap Khedkar, "Learning and Tuning Fuzzy Logic Controllers Through Reinforcements," *IEEE Trans. on Neural Networks*, vol. 3, no. 5, pp. 724-740, Sept. 1992.
8. Sung Hoon Jung, Kwang Hyun Cho, Tag Gon Kim, and Kyu Ho Park, "Defuzzification Method for Multishaped Output Fuzzy Sets," *Electronics Letters*, vol. 30, no. 9, pp. 740-742, Apr. 1994.