第12卷第5期
1995年10月
控制理论与应用
CONTROL THEORY AND APPLICATIONS
Vol. 12, No. 5
Oct., 1995

Optimizing Fuzzy Logic Controller Using Nelder and Mead's Simplex Algorithm

LI Wei and TAN Qunhua
(National Laboratory for Intelligent Technology and Systems,
Department of Computer Science, Tsinghua University • Beijing, 100084, PRC)
Hartmut Janocha
(Lehrstuhl für Prozessautomatisierung, University of Saarland D-66041 Saarbrücken, Germany)

Abstract: This paper presents a new method for optimizing a fuzzy logic controller. The main idea of this study is to automatically modify the membership functions regarding change-in-error $\dot{e}$, which represent the feedback of velocity. For this purpose, first of all, a set of membership functions are chosen by inverstigating their effects on control performance. Then, the Nelder and Mead's simplex algorithm is adopted to optimize these membership functions during operation. In order to demonstrate the effectiveness of the proposed method, simulation results are reported involving both step and tracking control of a nonlinear plant.

Key words: fuzzy logic control; nonlinear system; optimization technique

1 Introduction

During last twenty years, fuzzy logic (FL) controllers have been successfully applied to many systems with nonlinear dynamic equations or with unknown structure[1~5]. However, the optimization of an appropriate rule base and membership functions is very difficult tasks. One of the widely used methods for optimizing an FL controller is to define membership functions of linguistic variables and to formulate fuzzy rules by control engineers. These membership functions and rules are stored in computer memory and can not be modified during operations[3~8]. Unfortunately, there is no general principle for determining membership functions and rules for each particular system with specific requirements. Another approach is to modify the rule base by self-organizing algorithms automatically according to previous responses until a desired control performance is achieved[2][9]. In self-orgainzing control (SOC), modifying the rule base is to optimize a decision table based on a defined performance index. In such a decision table, however, there are a number of elements which should be optimized during operations (a high dimension optimization problem), it is very hard to optimize the rule base within few learning times.

In [5~7], a hybrid control scheme for a mechanical manipulator has been proposed, which consists of a conventional FL controller and a D controller, as shown in Fig. 1. Based

Manuscript received Mar. 1, 1994, revised Jun. 7, 1995.

n this scheme, in this paper we present a new method for optimizing an FL contorller, which modifies membership functions automatically according to previous responses rather han the rule base. For this purpose, a set of membership functions are characterized by a parameter vector k, and their parameters are optimized by the Nelder and Mead's simplex lgorithm during system operation. The proposed approach is suitable for systems that are weakly defined by analytical models and is more effective than traditional SOC approaches, since it reduces optimizing a decision table into searching for a few of the optimum parame- ers of membership functions. Besides, the adjustment of membership functions using two or three parameters can be easily done by control engineers. To demonstrate the effective- ness of the proposed method, several simulation results are reported involving both step and tracking control of a nonlinear plant.

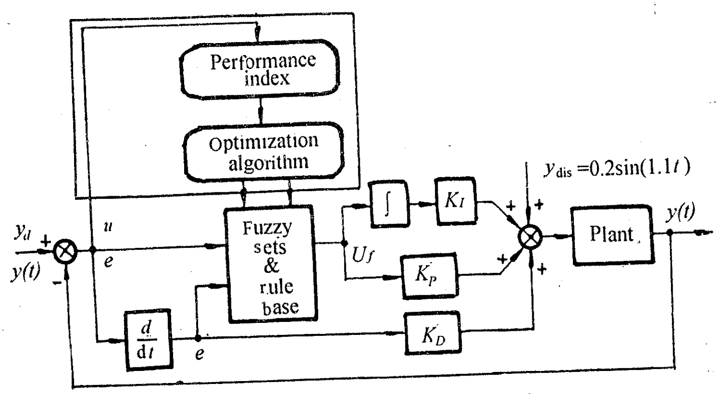

Fig. 1 Control scheme for SOC algorithm

2 Choice of Membership Functions

Using the control scheme in Fig. 1, the control signal u for a plant is computed as follows:

$$u = K_p u_f + K_I \int_0^{t_0} u_f \mathrm{d}t + K_D \frac{de}{\mathrm{d}t}. \tag{1}$$

Where $\frac{de}{dt}$ is a derivative of error $e = y_d - y(t)$, and K_P, K_I and K_D are control parameters. The output from the FL controller, u_f, is determined by the Min-Max inference algorithm and the centroid defuzzification method. Table 1 lists the fuzzy rule base of the FL controller.

Table 1 Fuzzy rule base

$\dot{e}$ \ e	NB	NS	ZO	PS	PB
NB	PB	PB	PB	PS	ZO
NS	PB	PB	PS	ZO	NS
ZO	PB	PS	ZO	NS	NB
PS	PS	ZO	NS	NB	NB
PB	ZO	NS	NB	NB	NB

In general, both membership functions of triangular type and exponential type are widely used in FL control. Obviously, changing these membership functions will affect control performance. A membership function of a symmetrical triangle is characterized by its peaking and zero points (values of one and zero). Since such membership functions can be adjusted only by shifting their peaking and zero points, they are not flexible enough in modifying their shapes. A membership function of an exponential type $\mu(\alpha) = \exp[-(\alpha - \alpha_0)^2/\sigma^2]$ is characterized by its mean α_0 and deviation σ. By increasing σ, the membership functions of this type become flatter. If σ is very large, control performance becomes poor because the corresponding linguistic variables become too "fuzzy". In [5], membership functions regarding $\dot{e}$, which represent the feedback of velocity, are modified by adding some interpolation points to the triangular type. This idea leads to reduce the number of membership functions to be investigated. These studies have shown that the shape of membership functions for desired control performance does not fall on the triangular type nor the exponential type. In this paper, therefore, membership functions regarding $\dot{e}$, are defined by cubic polynomes shown in Fig. 2b, whose shapes can be modified by shifting the "moving" points along the dashed lines, as shown in Fig. 3. The parameters k_b, k_s and k_z, denoted by a vector $\boldsymbol{k} = (k_z, k_s, k_b)$, are used to adjust the membership functions μ(NB) and μ(PB), μ(NS) and μ(PS), and μ(ZO), as shown in Fig. 3a ~ c, respectively. The parameter vector $\boldsymbol{k}$ can be changed in the range $[0.15, 0.85] \times [0.15, 0.85] \times [0.15, 0.85]$. On the contrary, the membership functions regarding e and u_f are chosen to be the triangular type and remain unchanged during operations, as shown in Fig. 2a and 2c, respectively.

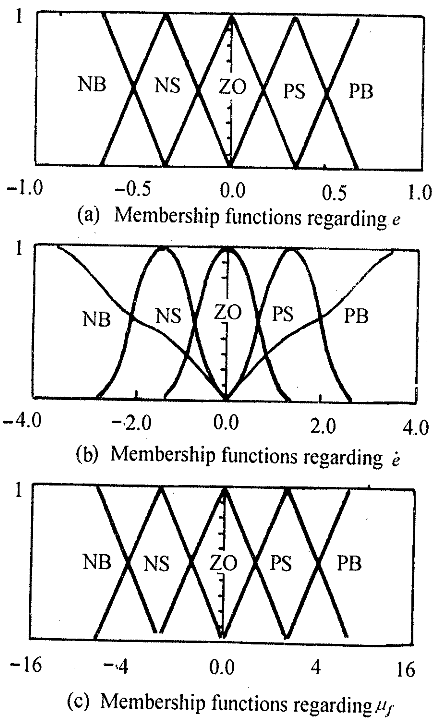

Fig. 2 Membership functions of lingustic variables

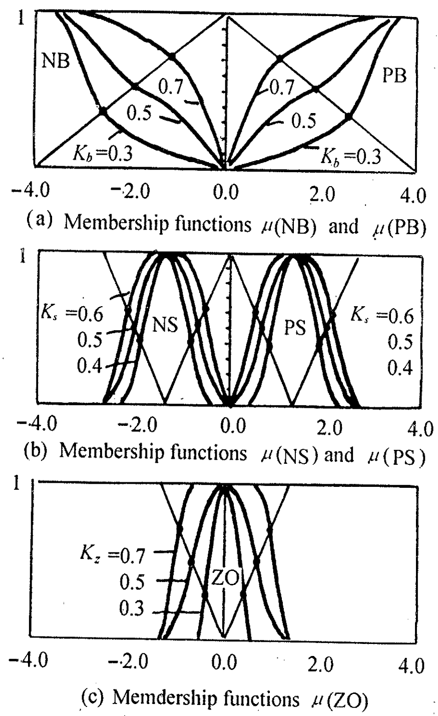

Fig. 3 Different membership functions's shapes

3 Optimizing Algorithm

In the proposed approach, membership functions for desired control performance are etermined by searching the optimum point of the vector $k^* = (k_z^*, k_s^*, k_b^*)$ according to revious responses. Since the parameter vector $k = (k_z, k_s, k_b)$ of the membership functions egarding $\dot{e}$ has the effects on control performance of a plant, $J = f(k_z, k_s, k_b)$ or $J = f(k)$ used to describe response behaviors. In order to achieve desired control performance, we ave to compute the minimum value $f(k^*)$ by searching for the corresponding vector $k^* = k_z^*, k_s^*, k_b^*)$. Many complex systems should be weakly defined, so it is difficult to get an xact analytical model of $J = f(k)$. Because Nelder and Mead's simplex algorithm[11] is uitalbe for solving optimization problems in the absence of analytical models, it is used to ptimize membership functions in operation process. To our knowledge, the effects of (ZO) on control performance are very weak[12], so the parameter k_z is assumed to be onstant c during optimization operation. Computing the minimum value $f(k^*)$ by the simlex algorithm is briefly described as follows:

1) Start with three points $k_1 = (c, k_s^{(1)}, k_b^{(1)})$, $k_2 = (c, k_s^{(2)}, k_b^{(2)})$, $k_3 = (c, k_s^{(3)}, k_b^{(3)})$ and ompute $J_1 = f(k_1)$, $J_2 = f(k_2)$, and $J_3 = f(k_3)$.

2) Find the maximum value J_h, the next maximum value J_g and the minimum value J_l, and the corresponding points k_h, k_g, and k_l.

3) Find the center point of k_h and k_l: $k_f = 0.5(k_h + k_l)$ and evaluate $J_f = f(k_f)$.

4) Reflect k_h, and k_f to find k_r and $J_r = f(k_r)$.

5) If $J_r \geqslant J_g$, proceed to the contraction and find k_c from $k_c = \beta k_h + (1-\beta)k_f$ and $J_c = f(k_c)$, where β $(0<\beta<1)$ is the contraction coefficient. If $J_r < J_g$, find k_c from $k_c = \beta k_r + (1-\beta)k_f$ and $J_c = f(k_c)$.

6) If $J_c < J_h$, replace k_h by k_c, check convergence and if not return to step 2). If $J_c \geqslant J_h$, move to next step.

7) Reduce the size of the simplex by $k_h = k_h + 0.5(k_h - k_1)$ and $k_g = k_g + 0.5(k_g - k_1)$ and calculate J_h and J_g, test for convergence and if not retrun to step 2).

4 Simulations

In order to demonstrate the effectiveness of the proposed method, by numerical simulations we test time responses for step, ramp and sinusoidal tracking control of the following nonlinear plant:

$$\ddot{y} + 2.0\xi\omega\dot{y}y + \omega^2 y^2 = \omega^2 u. \tag{2}$$

In all simulation studies, the initial values of the plant $y(0)$, $y'(0)$, $y''(0)$ are chosen as zero, and the initial points of the parameter vector are chosen as follows: $k_1 = (c, 0.25, 0.25)$, $k_2 = (c, 0.25, 0.75)$, $k_3 = (c, 0.6, 0.6)$, where c remains unchanged in operation process.

Simulation 1 (Step control)

In order to obtain a small overshoot in positive step control, we use the modification of the integral-square-error criterion:

$$J = \begin{cases} \int_0^{t_0} 100.0e^2 dt, & e > 0, \\ \int_0^{t_0} e^2 dt, & e \leqslant 0 \end{cases} \quad (5)$$

as a performance index. Fig. 4 shows the step response to the nonlinear plant ($\omega = 1.0; \zeta = 1.0; c = 0.5$) in the presence of a disturbance $y_{dis} = 0.2\sin(1.1t)$. Since there is a damping to dissipate internal energy of the plant, we chose the parameter $K_P = 1.0, K_I = 1.0$ and $K_D = 0.0$. In this case, $k^* = (0.5, 0.672205, 0.447884)$ with $f(k^*) = 0.2860$ is found by the Nelder and Mead's simplex algorithm. Although the amplitude of the disturbance $Y_{dis} = 0.2$ is quite large in comparison with $y_d = 1.0$, the time response shows a small maximum overshoot $M_P = 0.007$ and a fast settling time $t_s = 600$ms. This simulation result shows that the control scheme can efficiently cancel the disturbance in step control.

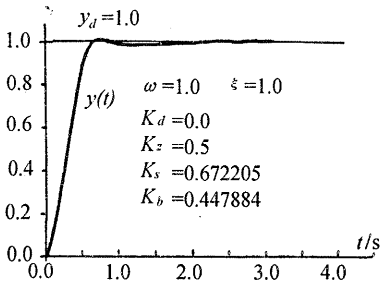

Fig. 4 Time response in the presence of a distubance

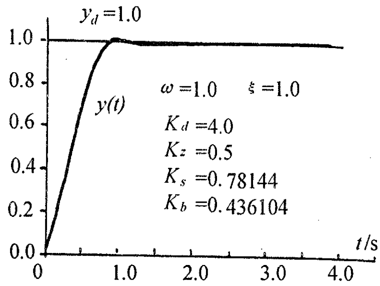

Fig. 5 Time response to an unstable plant

Now, we set $\zeta = -2.0$. In this case, there is no damping to dissipate internal energy of the plant. Although the membership functions μ(NB) and μ(PB) with the maximum value $k_b = 0.85$ produce the strongest negative feedback, the system is still unstable. We choose the parameters $K_P = 1.0, K_I = 1.0$ and $K_D = 4.0$, i.e., we add the D controller to dissipate the internal energy of the plant. Then, the Nelder and Mead's simplex algorithm is used to search for the optimum membership functions in operation process. $k^* = (0.5, 0.78144, 0.436104)$ with $f(k^*) = 0.3336$ is found. Fig. 5 shows the time response with $M_P = 0.0064$ and $t_s = 780$ms.

Simulation 2 (Tracking control)

In tracking control, the standard integral-square-error criterion is used to describe control performance. Fig. 6 shows the ramp tracking response to the plant ($\omega = 1.0; \zeta = 1.0; c = 0.3; K_P = 1.0, K_I = 1.0; K_D = 0.0$) in the presence of a disturbance $y_{dis} = 0.2\sin(1.1t)$. Using the Nelder and Mead's simplex algorithm, $k^* = (0.3, 0.378873, 0.286210)$ with $f(k^*) = 0.0010535$ is found. It can be observed that the control scheme

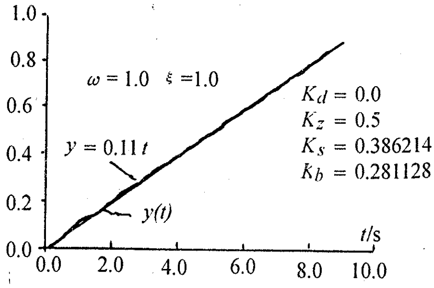

Fig. 6 Time response of ramp tracking in the presence of a disturbance

can efficiently cancel the disturbance to achieve the small tracking error.

Fig. 7 shows the sinusoidal tracking response to the plant with $\omega = 1.0; \zeta = 1.0; c = 0.3; K_P = 1.0, K_I = 1.0; K_D = 0.0$. Using the Nelder and Mead's simplex algorithm. $k^* = (0.3, 0.40474, 0.20)$ with $f(k^*) = 0.0030698$ is found. The simulation result exhibits a small tracking error in the sinusoindal tracking control.

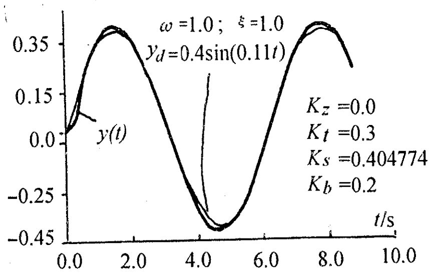

Fig. 7 Time response of sinusoidal tracking

5 Conclusions

This paper presents a new method to automatically optimizing a fuzzy logic controller using Nelder and Mead's simplex algorithm. This method does not need the strict initial values for convergence and is suitable for systems in the absence of analytical models. The simulation results show the effectiveness and the robustness of the proposed method.

The parameters K_P, K_I, and K_D in the control scheme can be determined by Ziegler-Nichols technique[13]. The FL controller in the control scheme is used to improve transient behaviors, e. g. , a small maximum overshoot and a fast setting time . If the FL controller can not stabilize an unstale plant under the strongest negative feedback control, the D controller is added to stabilize the problem if the system is controllable.

References

[1] Scharf, E. M. and Mandic, N. J.. The Application of a Fuzzy Controller to the Control of a Multi-Degree-Freedom Robot Arm. Industrial Application of Fuzzy Control, 1985, 41—61

[2] Wakileh, B. A. M. and Gill K. F.. Robot Control Using Self-Organizing Fuzzy Logic. Computers in Industry, 1990, 15:175—186

[3] Tzafestas, S. and Papanikolopoulos, N.. Incremental Fuzzy Expert PID Control. IEEE Trans. Industrial Electronics, 1990, 5:365—371

[4] Li W.. Fuzzy-Logic-Based Perception Action Behavior Control of a Mobile Robot in Uncertain Environments. International Journal of Engineering applications of Artificial Intelligence, 1994, 5:521—531

[5] Li W. and Zhang B.. Fuzzy Control of Robotic Manipulators in the Presence of Joint Friction and Loads Changes. Proc. of The 1993 ASME International Computers in Engineering Conference, USA, 1993

[6] 李伟等. 一种高效能的机器人模糊控制方案. 自动化学报, 1995, 21(4):400—409

[7] Li W. et al.. Effect of Membership Functions of Linguistic Variables on Control Performance. Proc. of The First Chinese World Congress on Intelligent Control and Intelligent Automation, 1993, 1:866—876

[8] Braae, M. and Rutherford, D. A.. Theoretical and Linguistic Aspects of the Fuzzy Logic Controller. Automatica, 1979, 15:553—557

[9] Procyk, T. J. and Mamdani, E. H.. A Linguistic Self-Organizing Process Controller. Automatica, 1979, 15:15—30

[10] Li, W. A New Self-Organizing Fuzzy Controller. Proc. of The 3rd International Workshop on Advanced Motion Control, at the University of Berkelry, 1994

[11] Nelder, J. A. and Mead, R.. A Simplex Method for Function Minimization. The Comp. Journal, 1965, 7:308—313

[12] Li W. and Wu Z. W.. A Self-Organizing Fuzzy Controller Using Neural Networks. Proc. of The 1994 ASME International Computers in Engineering Conference, 1994, 2: 807—812

[13] Ziegler, j. G. and Nichols, N. B.. Optimum Settings for Automatic Controllers. Transactions of the ASME, 1942, 8: 759—768

利用 Nelder-Mead 单纯形法优化模糊逻辑控制器

李 伟 谭群华

(清华大学计算机系，国家智能技术与系统实验室·北京，100084)

Hartmut Janocha

(Lehrstuhl für Prozessautomatisierung, University of Saarland D-66041 Saarbrücken · Germany)

摘要：本文提出了一种优化模糊逻辑控制器的新方法. 该方法的主要思想是自动地优化误差变化率的隶属度函数，因为这类隶属度函数表征了速度的反馈. 为此，首先定义了一族参数化的隶属度函数. 然后，在系统的运行过程中利用 Nelder-Mead 单纯算法来优化这类隶属度函数. 为了验证所提方法的有效性，报告了控制一非线性被控对象的阶跃与跟踪响应.

关键词：模糊逻辑控制；非线性系统；优化技术

本文作者简介

李 伟 1957 年生. 分别于 1982 年和 1984 年在北方交通大学电气工程专业获学士与硕士学位，并于 1991 年在德国 saarland 大学获工学博士学位. 1993 年被聘为清华大学计算机系副教授. 目前的研究兴趣是：机器人学，行为控制，计算机仿真，模糊逻辑与遗传算法，多传感器融合与集成. 在德国发表"Graphical Simulation and Collision Avoidance of Robots"一书. ASME 和 IEEE 会员.

谭群华 女. 1968 年生. 分别于 1990 年和 1992 年在武汉大学计算机系和哈尔滨船舶工程学院计算机与信息科学系获学士与硕士学位. 现为清华大学计算机系博士生. 目前的研究兴趣是：机器人控制，模糊控制和非线性系统建模.

Professor Hartmut Janocha Born in 1944, studied electrical engineering and headed the Machine and Control Division of the Institute for Production Engineering (IFW) of the University of Hannover from 1985 to 1989. He now heads the Labroatory for Production Automation (LPA) of the University of the Saarland in Saarbrüchen, Germany.