

# 采用 BP 神经网络记忆模糊规则的控制<sup>1)</sup>

应行仁 曾南

(中国科学院自动化研究所,北京)

## 摘 要

本文提供了一种比模糊推理更为自然的方式使用人们的经验知识,通过一组神经元不同程度的兴奋表达一个抽象的概念值,由此将抽象的经验规则转化成多层神经网络的输入-输出样本.通过 Back-Propagation 学习算法使得网络记忆这些样本.控制器以“联想记忆”方式使用这些经验.本文介绍了控制器的构造方法,给出了控制仿真结果,并讨论了这种控制器的特点和发展前途.

**关键词:** 神经网络,智能控制, Back-Propagation, 模糊控制.

## 一、引 言

在实践中,许多受控的对象缺乏精确的数学描述或具有时滞、非线性等复杂的关系,难以用现有的理论方法分析和控制.但是它们往往可以由人们根据一些经验知识较好的控制.因此,如何通过计算机利用这些经验知识自动地进行控制,便是一种非常实用、有效的技术,同时也是智能控制的一个基础课题.

自从 1973 年 Zadeh<sup>[1]</sup>, 1975 年 Mamdani<sup>[2]</sup> 的工作之后,模糊推理成为人们表述经验知识的常用工具.至今, Mamdani 的模糊逻辑控制器的设计方法和 Zadeh 的模糊推理规则仍被人们在研究中反复提及,它们几乎是现在利用经验知识控制系统的唯一方法.

但是,模糊推理并不能十分令人满意地表达经验知识. Zadeh 的推理规则仅仅是与人们思维比较接近的一种人为的规定.模糊量也缺乏一个明确的准则用以确定隶属度函数曲线.实际上,对于采用最大隶属度决策方法的模糊逻辑控制器,其控制律对于隶属度函数曲线形状的变化并不敏感.它们只有极小部分的信息是有用的,绝大部分的变化对控制律毫无影响,以至采用普通逻辑控制器就可以很好地代替它<sup>[3]</sup>.

在人们的思维中,抽象的“概念”对应着一个模糊量.抽象的经验知识是联系它们的一个映射.从生理的观点,人们的知识是大脑神经网络所实现的某些记忆.它是由神经元之间不同强度的联系来实现“概念”之间的联系,而“概念”则是具体的观测量或控制量所引起的一些神经元不同程度的兴奋.依照这个观点,本文提出一种新的表达经验知识的方法.并以此构造一个可以利用抽象经验规则的控制器的.

本文于 1990 年 6 月 12 日收到.

1) 国家自然科学基金 No.68974016 资助的课题. 本文曾在 1990 年全国控制理论及应用年会(杭州)上宣读.

## 二、结构与表达

控制器的构造由图 1 所示。它与模糊逻辑控制器有着相同的框架，但各部分的实现方法不同。Rule Base 储存抽象的经验规则，由多层神经网络实现向量空间的一个映射。Fuzzier 将控制器的输入转化成模糊量，其输出送往神经网络的输入节点。Defuzzier 将神经网络输出所表示的模糊量转化成具体的控制量。控制器在使用之前，通过“概念”的表达方法，将抽象的经验规则转化成样本，采用 Back-Propagation 算法对神经网络进行训练，使它记忆这些经验知识。控制器按照“联想记忆”方式来使用这些经验，实现控制。

抽象的经验规则是由下面这样的一组逻辑语句组成：

if ( $E = A_i$  and  $CE = B_i$ ) then  $U = C_i$ .

这里  $E$ ,  $CE$  和  $U$  是逻辑变量，在控制中可以解释为误差、误差变化和控制； $A_i$ ,

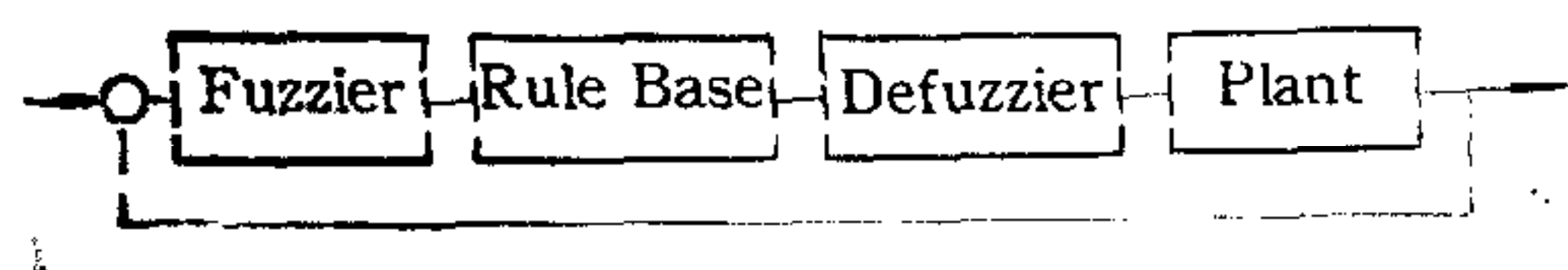


图 1 控制器结构框图

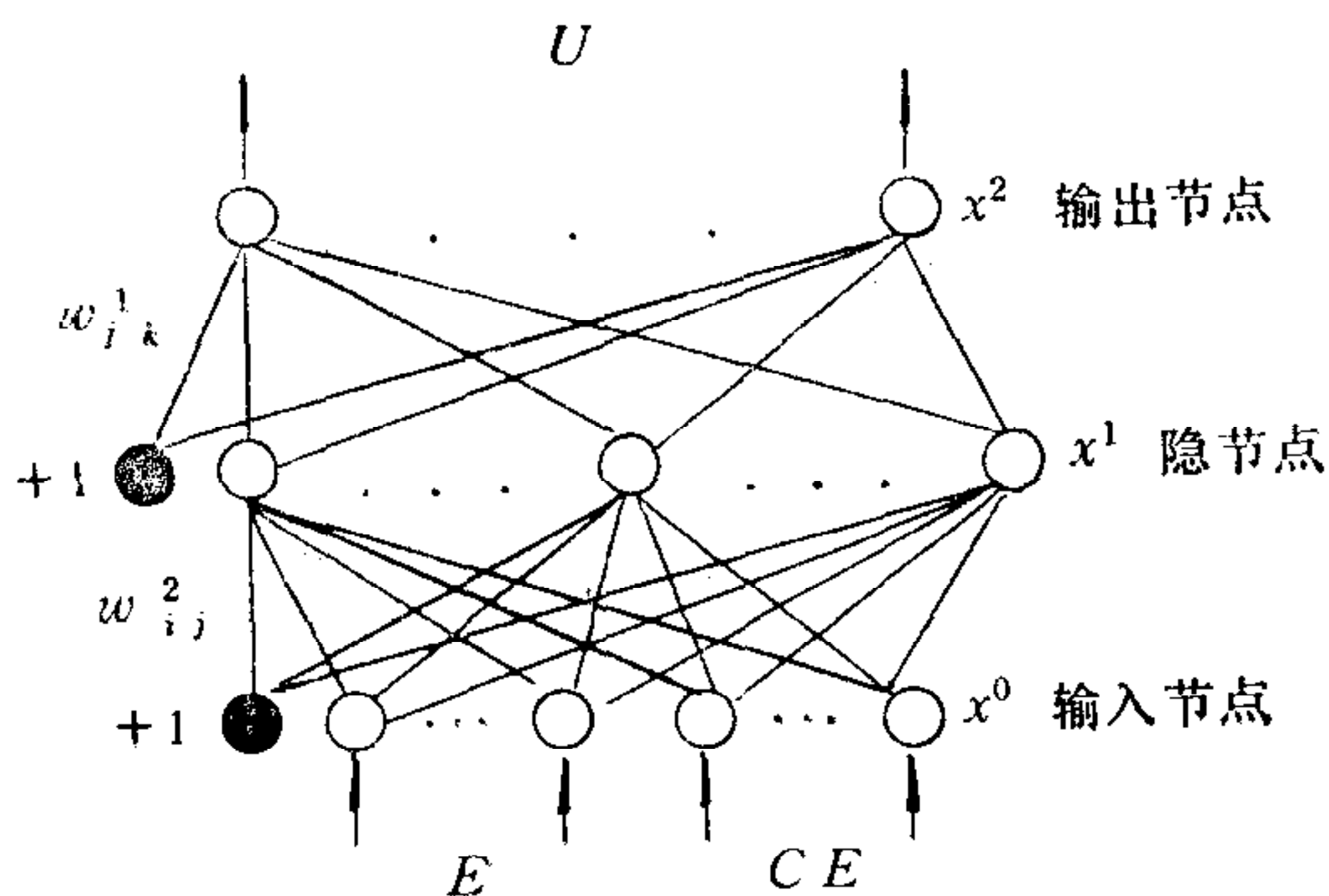


图 2 三层神经网络的作用

$B_i$  和  $C_i$  是对各自变量论域上的模糊(常)量。当然，逻辑语句中的条件和作用可以不限于上述几个分量。

对于模糊逻辑控制，上述逻辑语句采用 Zadeh 的模糊推理规则来实现。在本文，它们由神经网络来实现。Rule Base 在这里是一个三层神经网络，如图 2 所示。条件变量  $E$  和  $CE$  对应网络输入的两组节点，而作用变量  $U$  对应着网络的所有输出节点。网络的输入、输出节点分别代表它所在变量论域中的一个名义值。例如：网络有 15 个输出节点，节点  $x_1^2, \dots, x_{15}^2$  分别代表控制量从 -7 到 7 的一个整数值。

与某个逻辑变量对应的一组节点上的数值表示为一个“概念”。每个节点上可以是  $[0, 1]$  区间中某个数值，说明代表它名义值的神经元的兴奋程度。它是其节点的名义值在这个“概念”所对应模糊量的隶属度。

“概念”所对应的模糊量的隶属度函数是一个高斯函数

$$G(x, \mu, \sigma) = \exp(-(x - \mu)^2 / 2\sigma^2),$$

由均值  $\mu$  和均方差  $\sigma$  确定。例如： $U$  上模糊量  $PM$  的  $\mu = 4, \sigma = 2$ ，则它所对应的“概念”在 15 个输出节点上的数值分别为  $x_i^2 = G(i - 8, 4, 2), i = 1, 2, \dots, 15$ 。

依照上述方法，一旦确定了各个“概念”的均值和均方差，便可以把抽象的经验规则转化为神经网络的一组输入、输出样本。采用 Back-Propagation 学习算法训练神经网络，调整网络的权重  $w$ ，使得网络能够实现这组样本的输入输出的对应映射。

Back-Propagation 三层神经网络的权重按照下列方法确定：网络的权重  $w$  按随机

方式初始化(例如: 随机取  $[-1, 1]$  间的数给每个  $w$  赋值), 然后根据每个样本值与网络输出的误差, 轮番逐个学习修改网络的权重, 直至达到某个规定的条件. 对于样本  $(x, y)$ , 令  $x^0 = x$  作为网络的输入, 依照图 2 的公式, 得出对应网络输出  $x^2$ . 网络按梯度法修改权重

$$\Delta w = -\alpha \frac{\partial \|y - x^2\|^2}{\partial w},$$

则有学习算法

$$\Delta w_{ij}^k = \alpha \delta_i^k x_j^{k-1} \quad i, j, k.$$

对于三层网络

$$\delta_i^1 = \left( \sum_j w_{ij}^2 \delta_j^2 \right) x_i^1 (1 - x_i^1),$$

$$\delta_i^2 = (y_i - x_i^2) x_i^2 (1 - x_i^2).$$

这里常数  $0 < \alpha < 1$ , 称为学习因子;  $w, \delta, x$  的上标表示网络的层数. 关于 Back-Propagation 神经网络详见文献 [4, 5].

三层神经网络的输入和输出节点数按照量化的精度来设计. 中间隐节点数与记忆的样本数有关. 一般来说,  $s$  个隐节点可以用来准确记忆  $s + 1$  个不同的样本<sup>[6]</sup>.

Fuzzier 将控制器的输入模糊化. 在这里它将控制系统中实际测量的误差  $e$  和误差变化值  $ce$  转化成“概念”值. “概念”所对应模糊量的隶属度是  $e$  或  $ce$  实测值的均值  $\mu$  的高斯函数  $G(x, \mu, \sigma)$ , 这里的均值  $\sigma$  由设计者确定;  $x$  是神经网络输入节点的名义值. 实测的反馈量按上述方式转化成神经网络输入节点的接收数值.

Defuzzier 将控制的“概念”值转化成具体的控制. 如同模糊控制一样, 可以有几种设计方案. 最简单的一种是直接采用输出最大的那个节点的名义值. 它类似于采用最大隶属度决策方法的模糊逻辑控制器. 本文在仿真实验中采用了这种方法.

Back-Propagation 神经网络能够记忆训练过的样本. 当网络输入为非样本时, 其输出与该输入较近的样本的输出非常相似. 这是一种对样本的“联想记忆”<sup>[7]</sup>. 控制器以这种方式来使用经验.

### 三、仿真计算

考虑倒立摆系统: 一根竖直的长杆, 下端由铰链与小车连接. 长杆与垂线的偏角为  $\theta$ , 控制力  $F$  加在小车上. 控制的目标是使长杆在初始扰动后还能保证竖直. 倒立摆的运动方程如下:

$$\ddot{\theta} = \frac{mg \sin \theta - \cos \theta (F + m_p l \dot{\theta}^2 \sin \theta)}{\frac{4ml}{3} - m_p l \cos^2 \theta}$$

其中  $\theta$  为偏离角; 杆质量  $m_p = 0.1\text{kg}$ ; 车、杆总质量  $m = 1.1\text{kg}$ ; 杆长的一半  $l = 0.5\text{m}$ ,  $g$  为重力加速度. 控制经验规则采用在模糊控制中常用的 Mamdani 规则<sup>[2]</sup>, 如图 3 所示.

逻辑变量  $E$  和  $CE$  的模糊量  $NB, NM, NS, O, PS, PM, PB$  的均值  $\mu$  分别为  $-6, -4, -1, 0, 1, 4, 6$ . 逻辑变量  $U$  的模糊量  $NB, NM, NS, O, PS, PM, PB$  的均值分别为  $-7, -4, -1, 0, 1, 4, 7$ . 所有模糊量高斯曲线的  $\sigma = 2$ .

		误差变化 $CE$						
		$NB$	$NM$	$NS$	$O$	$PS$	$PM$	$PB$
误差 $E$	$NB$							
	$NM$	$PB$				$PM$		$O$
	$NS$			$PM$		$O$	$NS$	
	$O$	$PM$		$PS$	$O$	$NS$		
	$PS$	$PS$		$O$	$NM$		$NM$	
	$PM$	$O$		$NM$	$NB$			
	$PB$	$O$		$NM$	$NB$			

图 3 控制经验规则表

神经网络输入有 26 个节点, 分别对应着  $E, CE$  从  $-6$  到  $6$  所有整数的名义值. 输出有 15 个节点, 对应着  $U$  从  $-7$  到  $7$  所有整数的名义值. 神经网络输入输出节点上的名义值乘以量化因子对应着观察量和控制量的实际值. 隐节点有 50 个. 网络随机选取  $[-1, 1]$  区间中的实数来初始化权重. 采用学习

因子  $\alpha = 0.25$ , 经过 177919 次的学习, 记忆了控制经验规则表所产生的 49 个样本, 所

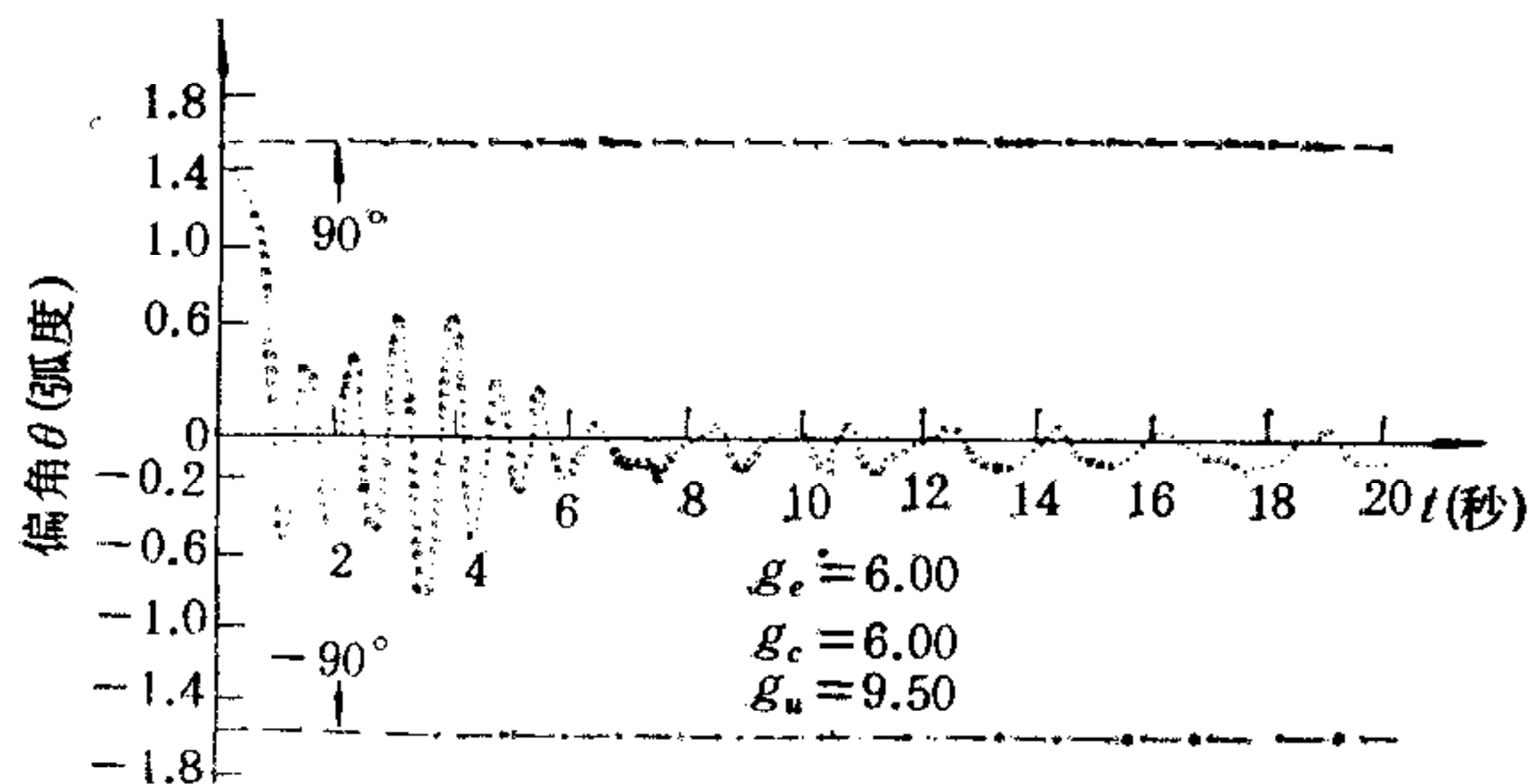


图 4 初始扰动为 1.4 弧度 ( $80.2^\circ$ ) 时的控制效果

有样本值的误差小于 0.05.

图 4 表示初始扰动为 1.4 弧度 ( $80.4^\circ$ ) 时控制的结果. 在这里, 仿真步长为 0.05 秒, 采样周期为 0.1 秒. 误差、误差变化和控制量的量化因子  $g_e = 6, g_c = 6, g_u = 9.5$ . 倒立摆较好地控制在摆动范围不大于  $\pm 0.13$  弧度 ( $7.4^\circ$ ) 的状态. 这个摆动是由较粗的量化所造成的. 如果采用较细的量化, 摆动可以控制在较小的范围. 图 5 表示  $g_e = 60, g_{ce} = 60, g_u = 1$  时的控制效果. 在这里初始扰动为 0.05 弧度 ( $2.9^\circ$ ), 系统很快进入幅

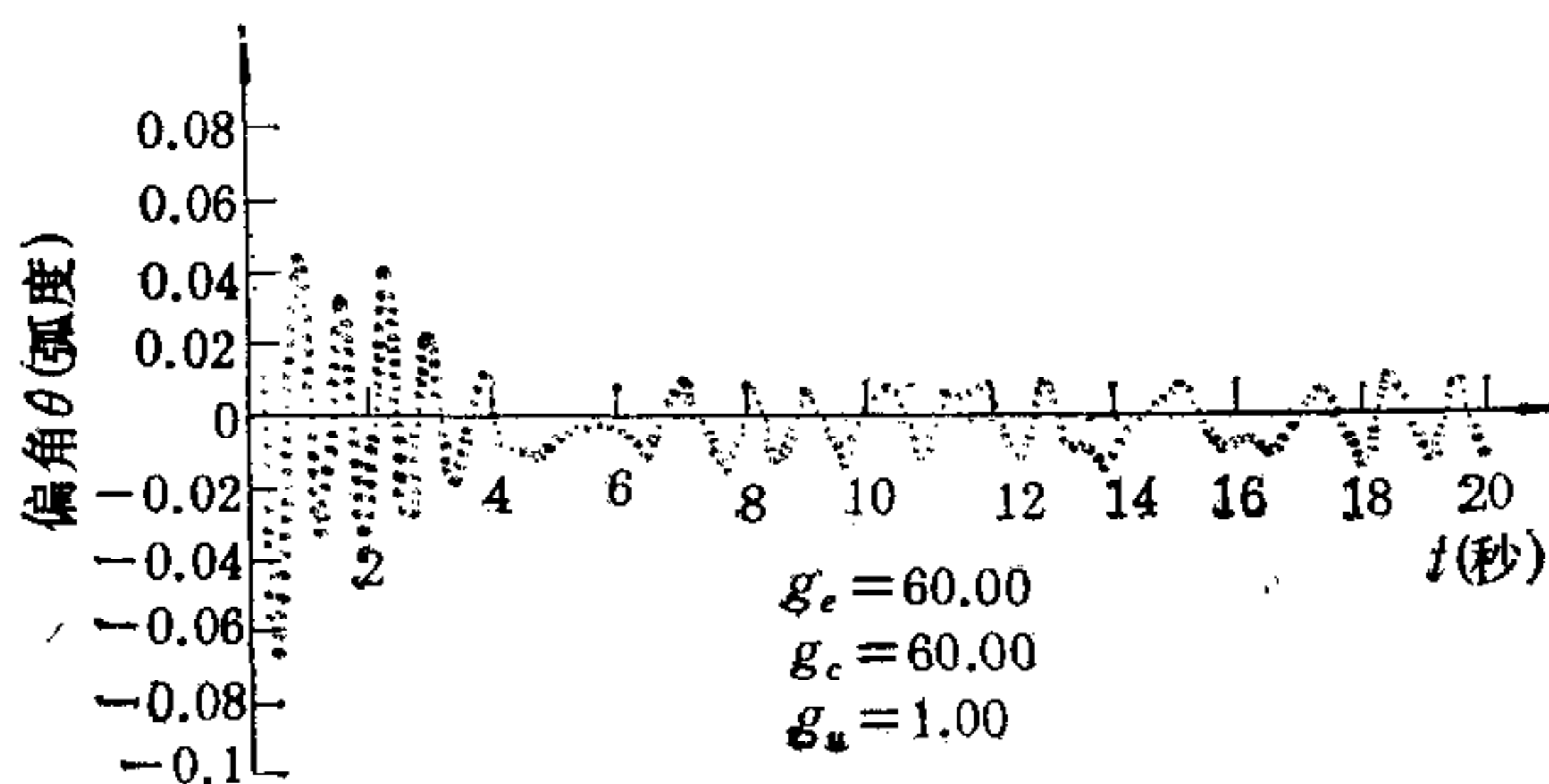


图 5 初始扰动为 0.05 弧度 ( $2.9^\circ$ ) 时的控制效果

度为 0.01 弧度 ( $0.57^\circ$ ) 的摆动状态. 在实际应用中, 可以根据扰动的幅值选取合适的量化因子, 或采用自动切换方式来取得更好的控制. 我们比较了采用相同控制规则和量化因子的模糊控制, 其控制效果与神经网络控制器十分相近.

## 四、讨 论

本文提供了一种用神经网络来记忆抽象经验规则的方法。这种方法比常用的模糊逻辑方法更符合人类认识的模式。仿真计算表明,用这种方法设计的控制器能够很好地利用经验知识控制一个非线性系统。当采用更多条件和作用(多输入,多输出系统)的复杂经验规则时,只需规定神经网络的输入输出节点代表的含意,无需改变整个控制器的结构和算法,因此具有广泛的通用性。

对于模糊逻辑控制器,当控制经验增加时,计算量将随之增加。而神经网络的经验知识是通过训练网络的权重得到记忆。整个训练可以离线进行,或直接复制性能良好网络的权重。神经网络作用的计算量与经验知识的多少无关。我们可以利用尽可能多的经验知识,并容许在其中包含少量甚至是错误的经验。在给这些样本相同机会的训练中,少量与大多数正确经验相矛盾的错误经验在记忆中将自动被排除。随着神经网络研究的发展和硬件的出现,这种方法将显示出更多的优点。

Fuzzier 和 Defuzzier 也可以很容易用神经网络来实现,这样它们与 Rule Base 可以组合成一个多层神经网络。在硬件实现中,这种并行、分布式的计算是非常迅速的。如果将 Defuzzier 的逆向功能设计成与 Fuzzier 一样,加在控制器两端的具体的量可以通过它们变成一对“概念”值加在 Rule Base 两端。这意味着我们可以采用“示教”的方式来训练控制器。以此可能做成通过示教学习的控制器。这种功能又为具有自学能力的控制器提供了实现的基础。

## 参 考 文 献

- [1] Zadeh, L. A., Outline of a New Approach to the Analysis of Complex Systems and Decision Processes, *IEEE Trans, System Man Cybern.*, 1 (1973), 28—44.
- [2] Mamdani, E. H. and Assilian, S., A Fuzzy Logic Controller for a Dynamic Plant, *Int. J. Man Machine Stud.*, 7(1975), 1—13.
- [3] 应行仁,采用最大隶属度决策的 Fuzzy 逻辑控制器分析,模糊数学, 4(1982), 1—12.
- [4] Lipmann, R., An Introduction to Computing with Meural Nets, *IEEE Assp Mag.*, 4(1987), 4—22.
- [5] Rumelhart, D., McClelland, J. DDP Group, Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. I & II. MIT Press, London, 1986.
- [6] 应行仁,三层神经网络隐单元与样本记忆的关系,模式识别与人工智能, 3(1990), 29—44.

## A CONTROLLER IMPLEMENTED BY RECORDING THE FUZZY RULES BY BP NEURAL NETWORKS

YING XINGREN ZENG NAN

(Institute of Automation, Chinese Academy of Sciences)

### ABSTRACT

A more natural way of using the human experiences than the fuzzy reasoning is provided in this paper. An abstract concept is expressed by a set of neurons with different exciting degrees. So, the abstract experience rules are transformed to the input-output samples of multi-layer neural network, and these samples are recorded in the network by Back-Propagation algorithm. The controller utilizes these experiences according to associative memory. The design, simulation result, feature and further development of this controller are also discussed.

**Key words:** Neural network; intelligent control; back-propagation; fuzzy control.