

# 城市单路口两级加权神经网络控制系统设计

徐欣

(上海理工大学管理学院 上海 200093)

**摘要** 城市交通系统是一个非常复杂的非线性系统,很难建立精确的数学模型,而 BP 神经网络具有较强的自学习、自适应的特点,适合复杂的大系统。针对单交叉路口红绿灯控制问题,基于改进的 BP 神经网络算法,同时考虑关键车流和非关键车流信息,提出并设计了两级加权神经网络控制器来进行实时控制。仿真结果表明,本方法优于传统控制方法。

**关键词** 交通控制,神经网络,关键车流

## Design of Hierarchical Weighted Neural Network Control System for City Traffic in Single Intersection

XU Xin

(Management Dept., Shanghai University of Science and Technology, Shanghai 200093, China)

**Abstract** City traffic system is a very complicated non-linear system. It's very difficult to build precise mathematical model and BP neural network has advantage in self-study and self-adaption. In this paper, for the control problem in one intersection, based on improved BP neural network algorithm, considering key traffic flow and nonkey traffic flow, hierarchical weighted neural network controller was proposed and designed. It's used to control traffic on time. The simulation results show that this control method is better than traditional methods.

**Keywords** Traffic control, Neural network, Key traffic flow

### 1 引言

交通系统具有高度的非线性和随机性,是一个复杂的大系统。在我国,许多大中城市以定时控制为主。定时交通控制系统是比较落后的一种交通控制系统,但是由于我国经济建设步伐远远超过交通控制发展的步伐,加上各地区交通的自身原因,到现在为止还有很多大型城市使用这种控制系统。这个定时网络需要确定的控制框架以及对时间要求苛刻的反应,定时系统既不能依据车流状况实时改变,又不能与其上下游相邻路口机协调动作,故常常出现有路权的方向已无车行走,而无路权的方向却停有大量车辆因红灯信号未变而不能行走,造成路口交通能力的大量浪费,形成了城市交通拥堵。目前,定时系统已经越来越不能适应日益变化的交通环境,而交通系统具有非线性、时变性的特点,很难为其建立精确的数学模型。所以,目前交通系统中的智能控制研究十分活跃,并取得了一定成果。

BP 神经网络具有较强的自学习、自适应的特点,适合复杂的大系统。为了加快网络的收敛速度,本文采用了改进的 BP 算法。由于非关键车流信息对车辆延误产生着不可忽视的影响,且与关键车流信息对延误的影响程度又有所不同,因此,本文同时考虑到关键车流和非关键车流信息,提出了设立两级加权神经网络控制器来进行实时控制,使得控制效果更为理想。

### 2 BP 网络基本模型和改进学习算法

BP 网络是运用最广的一种网络,具有较强的自学习、自组织和自适应能力,本文采用了三层网络结构:一个输入层、一个隐含层、一个输出层。BP 网络的学习过程包括正向传播和反向传播,如图 1 所示。其具体算法步骤如下:

1. 初始化;
2. 顺次选取一样本对  $A_k = (a_1^k, a_2^k, \dots, a_n^k)$ ,  $Y_k = \{y_1^k, y_2^k, \dots, y_q^k\}$  提供给网络;

3. 计算中间层各单元的输入,然后通过 S 函数计算中间层各单元的输入  $\{b_j\}$ ;

4. 计算输出层各单元的输入、输出;
5. 计算输出层各单元的一般化误差  $\{d_t^k\}$ ;
6. 计算中间层各单元的一般化误差  $\{e_j^k\}$ ;
7. 修正连接权  $\{v_{jt}\}$  和阈值  $\{\gamma_t\}$

$$v_{jt}(N+1) = v_{jt}(N) + \alpha d_t^k \cdot b_j \quad j=1, 2, \dots, p$$

$$t=1, 2, \dots, q \quad (1)$$

$$\gamma_t(N+1) = \gamma_t(N) + \alpha d_t^k \quad t=1, 2, \dots, q \quad (2)$$

8. 修正连接权  $\{w_{ij}\}$  和阈值  $\{\theta_j\}$

$$w_{ij}(N+1) = w_{ij}(N) + \beta \cdot e_j^k \cdot d_t^k \quad i=1, 2, \dots, n$$

$$j=1, 2, \dots, p \quad (3)$$

$$\theta_j(N+1) = \theta_j(N) + \beta \cdot e_j^k \quad j=1, 2, \dots, p \quad (4)$$

9. 顺次选取下一学习样本对提供给网络,返回到步骤 3,

到稿日期:2009-10-23 返修日期:2009-12-11 本文受科技部国家科技支撑计划项目(编号:2008BADA6B01)资助。

徐欣(1973-),女,博士生,讲师,研究领域为智能计算、信息管理。

直至全部  $m$  个样本对学习完毕;

10. 若误差  $< \epsilon$ , 网络收敛, 若学习次数  $> N$  (预先设定的值), 则网络无法收敛, 学习结束。

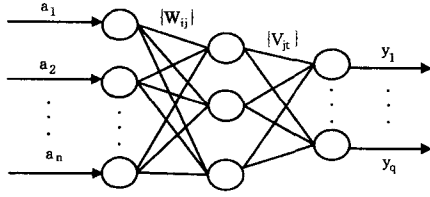


图1 BP网络的学习过程

为了加快 BP 网络的收敛速度, 本文采用了惯性校正法改进的 BP 算法。该算法的主要思想是: 在每一次对连接权进行校正时, 按一定比例加上前一次学习时的校正量, 即惯性项, 加速网络的收敛。具体做法如下: 式(1)和式(3)分别变为式(1<sup>1</sup>)和式(3<sup>1</sup>)。

$$v_{jr}(N+1) = v_{jr}(N) + \alpha d_j^r \cdot b_j + \eta \Delta v(N-1) \quad (1^1)$$

$$w_{ij}(N+1) = w_{ij}(N) + \beta \cdot e_j^i \cdot d^k + \eta \Delta w(N-1) \quad (3^1)$$

其中,  $\eta$  为惯性系数 ( $0 < \eta < 1$ ),  $\Delta v(N-1)$  为前次校正量,  $\Delta w(N-1)$  亦为前次校正量。从式(1<sup>1</sup>)和式(3<sup>1</sup>)可以看出, 惯性项起到了减小网络振荡的作用, 从而加速了网络的收敛。

### 3 单路口红绿灯交通控制系统设计

#### 3.1 两级加权神经网络控制系统结构设计

单路口交通控制的目标是选择适当的绿灯时间, 使车辆延误尽可能小。

如图 2 所示, 单路口的车流信息是由 4 个方向即南向北(SN)、北向南(NS)、东向西(EW)、西向东(WE)的信息组成的。

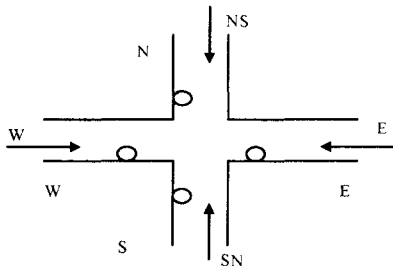


图2 车流信息

选择对交通性能影响最大的两相位关键车流的车辆到达信息作为一级神经网络控制器的输入, 由它们决定绿灯延长时延  $dt$ ; 选择两相位非关键车流的车辆到达信息作为二级神经网络控制器的输入, 由它们决定绿灯延长时延的修正量  $\Delta t$ 。图中的信号检测有 4 个, 分别安装在每条路进口距停车线适当距离的地方, 用于检测 4 个方向的车流信息。控制过程(如图 3 所示)为: 由检测器送来各方向车流的到达信息确定两相位的关键车流和非关键车流, 一级神经网络控制器根据绿灯相关键车流的到达车辆数  $arrive1$  和红灯相关键车流排队车辆数  $queen1$ , 计算出绿灯延长时延  $dt$ ; 二级神经网络控制器根据绿灯非关键车流的到达车辆数  $arrive2$  和红灯相位非关键车流的排队车辆数  $queen2$ , 计算出绿灯延长时延的修正量  $\Delta t$ 。因为非关键车流与关键车流对车辆延误的影响程度不同, 所以, 对  $\Delta t$  加权处理, 得到最终的控制策略: 绿灯延长时延为  $dt + w_i \cdot \Delta t$  ( $w_i \in (0, 1)$ )。

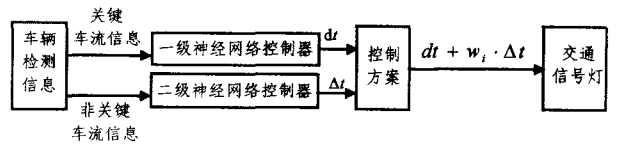


图3 车辆控制过程

#### 3.2 神经网络控制器的设计和实现

##### 3.2.1 BP 网络控制器的结构设计

输入层和输出层的确定

两级神经网络控制器采用相同的结构, 如图 4 所示。

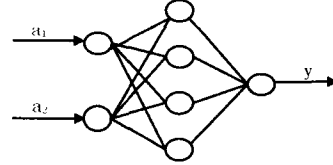


图4 两级神经网络控制器结构

图中第一层为输入层, 输入层神经元个数  $n=2$ , 其输入为  $A=(a_1, a_2)$ 。

对于一级控制器,  $A=(arrive1, queen1)$ , 对于二级控制器,  $A=(arrive2, queen2)$ 。

其中,  $arrive1$  表示绿灯相位关键车流的到达车辆数;  $queen1$  表示红灯相位关键车流的排队车辆数;  $arrive2$  表示绿灯相位非关键车流的到达车辆数;  $queen2$  表示红灯相位非关键车流的排队车辆数;

控制器第三层为输出层, 输出层神经元个数  $m=1$ ; 对于一级控制器, 输出  $Y=dt$ , 即为绿灯延长时延; 对于二级控制器,  $Y=\Delta t$ , 即为绿灯修正时延。

考虑到非关键车流对延误的影响不可忽视, 且与关键车流产生的影响程度又有所不同, 所以控制策略产生的最终绿灯延长时延为  $dt + w_i \Delta t$ , 其中  $w_i$  为权重, 它反映非关键车流对延误的影响程度。经过反复比较、实验,  $w_i$  取 0.6, 可使延误最小。

隐含层个数的确定

选取结构相对简单的三层 BP 网络, 即有 1 个隐层, 隐层采用 Sigmoid 转换函数。

现今并不存在一个理想的解析式来表示隐含层的神经元数, 但是可以提供以下一些参考公式, 在选择最佳隐单元数时可供参考: 因为输入节点为 7, 则根据公式  $n_1 = \log_2 n$  (其中,  $n$  为输入单元数) 和公式  $n_1 = \sqrt{n+m} + a$  (其中,  $m$  为输出神经元数,  $n$  为输入单元数,  $a$  为  $[1, 10]$  之间的常数), 在大量的训练实验中, 隐节点数取 4, 实验效果较好。

##### 3.2.2 BP 网络控制器中的数据预处理

输入数据的预处理

由于 BP 神经网络的隐层一般采用 Sigmoid 转换函数, 为提高训练速度和灵敏性以及有效避开 Sigmoid 函数的饱和区, 一般要求输入数据的值在  $0 \sim 1$  之间。所以, 需要对输入数据进行归一化处理, 将数据处理为在区间  $[0, 1]$  之间的数据。归一化方法有很多种形式, 这里采用如下公式:

$$\hat{x} = \frac{x - x_{\min}}{x_{\max} - x_{\min}}$$

输出数据的反变换处理

如果输出层节点也采用 Sigmoid 转换函数, 则输出变量

也必须作相应的预处理。预处理的方法多种多样,各文献采用的公式也不尽相同。但必须注意的是,预处理的数据训练完成后,网络输出的结果要进行反变换才能得到实际值。

#### 初始权值的选取

由于系统是非线性的,初始权值对于学习能否达到局部最小和是否能够收敛有很大关系。一个重要的要求是:初始权值在输入累加时使用每个神经元的状态接近于零,权值一般取随机数,比较小。

### 4 仿真研究

#### 4.1 仿真条件和仿真实现

仿真时,对车流的假设条件是:各方向的车流以0~1辆/s的来车率随机到达路口(来车率通过随机数产生);绿灯相车流以 $v=0.75$ 米/s的速率离开排队车辆的队列。

关键车流的确定:比较NS(由北向南)和SN(由南向北)两方向的来车率,以来车率大的一方为南北向关键车流,比较EW(由东向西)和WE(由西向东)两方向的来车率,以来车率大的一方为东西向关键车流。

每次仿真时间 $T$ 可自行设定。为了比较控制效果,引进评价指标——延误。

$$\text{延误} = \frac{\text{每个 } T \text{ 内, 四方向总的车辆延误数}}{\text{每个 } T \text{ 内, 四方向总的车辆到达数}} \quad (5)$$

$$\text{可表示为 } \text{delay} = \frac{T_d}{T_a}$$

注:式(5)中总的车辆延误数是指绿灯周期结束时,此周期内绿灯相位没有走完的车辆数。

按照BP网络的一般设计原则,中间层神经元的传递函数为S型正切函数。由于输出已被归一化到区间 $[0,1]$ 之间,因此,输出层神经元的传递函数可以设定为S型对数函数。使用MATLAB利用如下代码创建一个复合上述要求的BP网络:

```
threshold=[0 1;0 1;0 1;0 1;0 1;0 1;0 1];
net = newff(threshold, [19, 1], {'tansig','logsig'},
'traingdx');
```

其中,threshold设定了网络输入向量的取值范围 $[0,1]$ ,网络所用的训练函数为traingdx,该函数以梯度下降法进行学习,并且学习速率是自适应的,是一种改进的BP算法。

仿真过程中,由于网络需要对样本不断地进行学习,实际样本太少可能导致网络学习的不稳定性,因此实际样本越多,模拟的效果越好。输入值和预期输出值的选取,参照交通管制中心给出车流量的相关数据,如: $arrive1=15$ , $queen1=2$ 时,取 $y=12s$ 等(因篇幅所限,数据就不一一列出),训练样本选取 $m=10000$ 个,学习次数 $N$ 为1万次,在图4所示的网络中训练,当误差 $<\epsilon$ ( $\epsilon$ 取0.0002)时,停止训练。这样训练出来的两组权值 $\{w_{ij}\}$ 和 $\{v_j\}$ 有较好的实用价值。

学习算法采用改进的BP算法,加快了网络收敛速度。

#### 4.2 仿真结果比较

下面是用两级加权控制方法进行的30分钟的系统仿真,

随机产生了四方向的来车率,则控制结果如表1所列(只列出前6个周期)。

表1

周期	1	2	3	4	5	6
南北向绿灯时间(s)	20	15	37	35	20	37
东西向绿灯时间(s)	15	30	15	20	12	20

本文在30分钟的仿真时间内就几种控制方法做了一个对比。

用本文给出的两级加权控制方法,平均延误为1.60;如果只考虑关键车流,采用一级神经网络控制,平均延误为2.3,采用二级(非加权)控制,即认为关键车流与非关键车流信息影响同等重要,则平均延误为1.8,可见本文方法取 $W_i=0.6$ ,控制效果最佳。

采用传统的定时控制方法仿真之后,它们的几次比较结果如表2所列。

表2

仿真次数	仿真时间(秒)	定时系统延误	两级加权控制系统延误
1	200	3.28	1.15
2	300	5.20	1.37
3	400	5.96	1.52
4	500	5.20	1.20

可以看出,在同样条件下,本文控制方法是优于定时控制方法的。

**结束语** 本文提出了两级加权神经网络控制方法,用于单路口红绿灯的实时控制。该方法在减少延误方面有较大改进,取得了较好的控制效果,具有广泛的应用前景。进一步的工作是如何使实时性控制更加合理,控制方法更加实用。

### 参考文献

- [1] 陈洪. 单路口交通实时模糊控制的一种方法[J]. 信息与控制, 1997,26(3):227-233
- [2] 石建军,于泉. 现代交通控制相关技术的发展趋势分析[J]. 公路交通科技,2006,9:113-117
- [3] 张锦,高世廉. 中国城市道路交通控制系统发展研究[J]. 云南交通科技,1997,13(2):14-19
- [4] 陈淑燕,周振国,陈森发. 多相位交通信号模糊控制规则的优化[J]. 公路交通科技,2003,12:89-92
- [5] 刘丽莎,黄辉先,汤红忠. 单交叉口模糊交通控制器的研究[J]. 电机与控制学报,2003,2:166-169
- [6] Jain A K. Neural networks and pattern recognition[C]//IEEE Word Congress on Computational intelligence. Orlando, Plenary talks, June 1994
- [7] Werbos P J. Backpropagation: Past and Future [C] // ICNN. 1998:343-353
- [8] 肖秀春,姜孝华,张雨浓. 广义多项式神经网络及其结构自适应确定[J]. 重庆工学院学报:自然科学版,2007,23(7):97-99