

## 基于改进的神经元 PID 战术网络拥塞控制

张丽丽<sup>1</sup>, 王玉惠<sup>1</sup>, 陈哨东<sup>2</sup>, 吴庆宪<sup>1</sup>

(1. 南京航空航天大学, 南京 210016; 2. 光电控制技术重点实验室, 河南 洛阳 471009)

**摘要:** 未来信息化的网络中心战中, 网络拥塞问题成为制约战争信息有效传输的瓶颈。单神经元 PID 拥塞控制算法中, 系统的稳定性和超调量等性能对于神经元的增益系数  $K$  的依赖性很大, 结合无需辨识的自适应控制算法, 动态地调整增益系数  $K$ ; 另外, 为了进一步提高控制器品质, 加入调整因子在线调整神经元权值的学习率; 最后将改进后的算法应用到战术网络拥塞控制中。仿真结果验证了所提算法的有效性。

**关键词:** 战术网络; 拥塞控制; 单神经元; 自适应控制

**中图分类号:** V271.4; TP393 **文献标志码:** A **文章编号:** 1671-637X(2013)10-0014-04

## Tactical Network Congestion Control Based on Improved Neuron PID

ZHANG Lili<sup>1</sup>, WANG Yuhui<sup>1</sup>, CHEN Shaodong<sup>2</sup>, WU Qingxian<sup>1</sup>

(1. College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China;

2. Science and Technology on Electro-Optic Control Laboratory, Luoyang 471009, China)

**Abstract:** In the information network centric warfare, the network congestion problem becomes the bottleneck of warfare information transformation. In single neuron PID congestion control, the neuron gain  $K$  is the main factor affecting the system performance of stability and overshoot, which is totally depended on expertise and is highly subjective. In this paper, the neuron gain  $K$  is adjusted dynamically combined with identification-free adaptive control algorithm. In addition, in order to improve the quality of the controller, the adjustment factor is introduced to adjust the learning rate of neuron weight. The simulation results in the tactical network verified the effectiveness and feasibility of the improved algorithm.

**Key words:** tactical network; congestion control; single neuron; adaptive control

### 0 引言

在现代战争中, 信息是构成战斗力的重要因素, 信息化的重要特征表现为网络化, 网络已经成为指挥控制、侦察预警、联合打击等国防信息基础设施的命脉。美军也于 1999 年首次提出了全球信息栅格的概念, 并于次年正式向国会报告要求启动全球信息栅格项目, 打造一种可以向世界各地美军传送数据的万维网。由于网络的重要性, 参与到网络中的各种信息越来越多, 造成了信息量的迅猛增加, 不可避免的就出现了网络拥塞

问题。拥塞问题产生的根本原因是资源的相对不足, 用户提供的网络负载超过了网络的资源和处理能力<sup>[1]</sup>。

鉴于拥塞的直接发生地是在路由器上, 因此, 研究路由器上的拥塞控制策略具有更大的实际意义。由此出现了一系列基于路由器的拥塞控制算法, 主动队列管理(AQM) 算法因其能够预测拥塞的发生, 并能把队列长队稳定在一个比较低的值而得到了广泛的关注。AQM 算法主要包括 RED 算法、RED 变形算法、BLUE 算法、AVQ 算法等<sup>[2]</sup>。文献[3]提出了比例积分拥塞控制器, 与之前的算法相比, 具有较快的动态响应能力, 而且能够将队列保持在期望值附近, 又因其具有结构简单、易于实现等特点而被广泛应用。随着对神经网络的深入研究, 出现了具有自适应、自学习能力的 PID 算法, 这种算法不仅结构简单, 而且能够适应外界环境变化, 具有更强的鲁棒性。但是这种算法依然存在一定的问题, 系统的响应对于神经元的增益系数  $K$

收稿日期: 2012-10-09

修回日期: 2012-10-29

基金项目: 国家自然科学基金(11102080); 中央高校基本科研业务费专项资金(NP2011049; NZ2012005); 航空科学基金资助项目(20095152028)

作者简介: 张丽丽(1988—), 女, 山东济宁人, 硕士生, 研究方向为战术网络拥塞控制。

的依赖性很大,  $K$  值越大, 系统的响应越迅速, 但是超调会增大, 甚至会使系统不稳定, 减小  $K$  值, 系统的快速性变得很差。另外, 单神经元 PID 控制器中学习速率的取值基于大量的仿真与实验, 在控制过程中是恒定的, 这限制了控制器品质的提高。

基于以上分析, 本文建立了战术网络的模型, 分析了单神经元控制器和 PID 控制器的对应关系, 并进一步分析了单神经元控制器中增益系数和神经元权值系数对于控制器品质的影响, 由此提出了对神经元增益系数和权值系数的调整算法, 并仿真验证了该改进算法的有效性。

### 1 战术网络拥塞控制模型

由于战术网络是 Internet 信息流的传输, 因此对于战术网络的模型问题, 本文采用文献[4]在流体理论的基础上建立的拥塞窗口在主动队列管理机制下的动态模型。这个模型描述了网络特征变量的平均值之间的关系, 例如, TCP 平均窗口大小、平均往返时延和路由器平均队列长度。它可以用以下一组非线性的微分方程来描述

$$\begin{cases} \dot{W}(t) = \frac{1}{R(t)} - \frac{W(t)W(t-R(t))}{2R(t-R(t))}P(t-R(t)) \\ \dot{q}(t) = \frac{N(t)}{R(t)}W(t) - C(t) \end{cases} \quad (1)$$

式中:  $W$  (单位为包) 为期望的拥塞窗口大小;  $R$  (单位为 s) 为往返时间, 其大小等于排队时延与传输时延之和, 即  $R = \frac{q}{C} + T_p$ ,  $T_p$  为传输时延;  $P$  为路由器丢包概率/包标记概率;  $q(t)$  (包) 为路由器实时队列长度值;  $q_0$  (包) 为期望的路由器队列长度;  $N$  为 TCP 网络连接数目;  $C$  (包/s) 为链路带宽。假定上述模型中, TCP 网络连接数为常数, 即  $N(t) = N$ , 链路带宽为常数, 即  $C(t) = C_0$ 。选择  $(W, q)$  作为系统的状态变量, 路由器队列长度  $q(t)$  为输出变量,  $P$  为输入变量。在工作点  $(W_0, q_0, p_0)$  处, 对系统采用小信号分析法进行局部线性化可得

$$\begin{cases} \delta \dot{W} = - \left[ \frac{W_0 \cdot \delta W(t) + W_0 \cdot \delta W(t-R_0)}{2R_0} P_0 + \frac{W_0^2}{2R_0} \cdot \delta P(t-R_0) \right] \\ \delta \dot{q} = \frac{N}{R_0} \cdot \delta W(t) + N \cdot W_0 \cdot \left[ -\frac{1}{R_0^2 C} \delta q(t) \right] \end{cases} \quad (2)$$

将  $W_0 = \frac{R_0 C}{N}$ ,  $P_0 = \frac{2}{W_0^2}$ ,  $R_0 = \frac{q_0}{C} + T_p$  分别代入式(2)

中, 化简之后可以得到

$$\begin{cases} \delta \dot{W} = - \left[ \frac{N}{R_0^2 C} \cdot (\delta W(t) + \delta W(t-R_0)) + \frac{R_0 C^2}{2N^2} \cdot \delta P(t-R_0) \right] \\ \delta \dot{q} = \frac{N}{R_0} \cdot \delta W(t) - \frac{1}{R_0} \delta q(t) \end{cases} \quad (3)$$

对上式进行 Laplace 变换, 可以得到 TCP 的特性和队列的特性分别为

$$\begin{cases} P_{TCP}(s) = \frac{R_0 C^2}{2N^2} / \left( s + \frac{N}{R_0^2 C} \right) \\ P_{QUE}(s) = \frac{N}{R_0} / \left( s + \frac{1}{R_0} \right) \end{cases} \quad (4)$$

图 1 所示为拥塞控制系统。

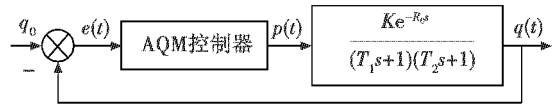


图 1 拥塞控制框图

Fig1 Congestion control block diagram

考虑到实际战术网络系统中存在延时, 因此可以将拥塞控制系统的传递函数表示为

$$G_p(s) = \frac{C^2}{2N} e^{-R_0 s} / \left[ \left( s + \frac{N}{R_0^2 C} \right) \left( s + \frac{1}{R_0} \right) \right] \quad (5)$$

式中:  $T_1 = \frac{R_0^2 C}{2N}$ ;  $T_2 = R_0$ ;  $K = \frac{R_0^3 C^3}{4N^2}$ ; 其他各参量物理意义同上。

### 2 改进单神经元 PID 算法

连续系统 PID 控制器的数学表达式为

$$u(t) = K_p \left[ e(t) + \frac{1}{T_i} \int e(t) dt + T_d \frac{de(t)}{dt} \right] \quad (6)$$

将上式离散化之后得到

$$u(k) = K_p e(k) + K_i \sum_{i=1}^k e(i) + K_d (e(k) - e(k-1)) \quad (7)$$

式中:  $K_p$  为比例系数;  $K_i$  为积分系数;  $K_d$  为微分系数;

$$K_i = \frac{K}{T_i}, K_d = K_p T_d。$$

增量式数字 PID 控制算式为

$$\Delta u(k) = K_p [e(k) - e(k-1)] + K_i e(k) + K_d [e(k) - 2e(k-1) + e(k-2)] \quad (8)$$

PID 控制器合理地选择  $K_p$ 、 $K_i$ 、 $K_d$  的值, 就可以使得系统稳定, 并使队列的输出稳定在一个较低的值附近。但是传统的 PID 控制器对于对象模型的精确性依赖程度很高, 对于复杂多变的系统, 传统 PID 控制不能达到很好的效果, 参数的选择对于控制器的性能影响很大, 系统鲁棒性不好, 而且参数的选择要根据专家经

验,并大量地试凑,这都造成 PID 控制器的局限。单神经元具有较高的自学习自适应能力,将单神经元的权值分别与 PID 控制器中的比例积分微分系数相对应,则 PID 控制器就可以在线地自适应调整系数,增加系统的自适应能力和鲁棒性;而且单神经元的输入是期望值与实际值的误差信号,对于系统模型的依赖性降低,增加了控制器的鲁棒性。图 2 所示为神经元控制器的工作过程。

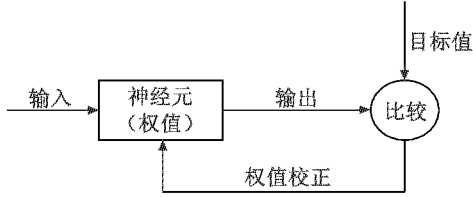


图 2 单神经元工作图

Fig. 2 Single neuron working diagram

由图 2 可以看出,假定输入信号为无误差信号,经神经元控制器之后输出值与目标值进行比较,如果误差依然存在,则进行神经元权值的调整,直至输入误差信号为零。单神经元的控制<sup>[5]</sup>如图 3 所示。

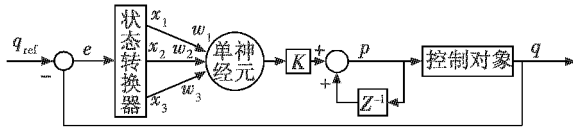


图 3 单神经元控制框图

Fig. 3 Single neuron control diagram

图中: $q_{ref}$ 为给定的队列长度值; $q$ 为系统实际输出的队列长度值;选择单神经元 PID 控制器的输入  $x_1, x_2, x_3$  分别为误差信号

$$\begin{cases} e(k) = q(k) - q_{ref} = x_1(k) \\ e(k) - e(k-1) = \Delta e(k) = x_2(k) \\ e(k) - 2e(k-1) + e(k-2) = \Delta^2 e(k) = x_3(k) \end{cases} \quad (9)$$

单神经元的输出

$$\Delta u(k) = K \sum_{i=1}^3 w_i(k) x_i(k) \quad (10)$$

式中: $K$ 为神经元比例系数; $w_i(k)$ 为权值。神经元控制器的输出作为控制信号输入被控对象,即

$$u(k) = P(k) \quad (11)$$

神经网络的学习分为导师学习和无导师学习两种形式<sup>[6]</sup>:无论是哪一种学习,都要通过调整神经元的权值来实现。设权值矢量  $w = (w_1, \dots, w_n)^T, n=3$ , 输入样本为  $x = (x_1, \dots, x_n)^T, n=3$ , 神经元学习算法的内容就是要确定神经元的权值调整量  $\Delta w(k)$ , 并得到权值的调节公式为<sup>[7]</sup>

$$w_j(k+1) = w_j(k) + \Delta w_j(k), \quad j=1, 2, 3 \quad (12)$$

单神经元的学习规则是神经网络的核心,能够反

映该控制器的学习能力,常用的包括无导师 Hebb 学习规则、有导师 Delta 或 Widrow-Hoff(最小二乘)学习规则等。将无导师 Hebb 学习规则与有导师 Delta 学习规则结合起来,构成了有导师 Hebb 学习规则为

$$w_j(k+1) = w_j(k) + \eta(d_j - y_j)xy_j, \quad j=1, 2, 3 \quad (13)$$

式中: $d_j$ 为导师信号,即期望队列输出; $y_j$ 为系统实际输出。本文中选择不导师 Hebb 学习规则来对权系数进行调整<sup>[8]</sup>,故单神经元 PID 控制器输出规范化处理后为

$$u(k) = u(k-1) + K \sum_{i=1}^3 \bar{w}_i(k) x_i(k) \quad (14)$$

其中归一化权值为

$$\bar{w}_i(k) = w_i(k) / \sum_{i=1}^3 |w_i(k)| \quad (15)$$

权值的计算公式如下

$$\begin{cases} w_1(k+1) = w_1(k) + \eta_I z(k) u(k) x_1(k) \\ w_2(k+1) = w_2(k) + \eta_P z(k) u(k) x_2(k) \\ w_3(k+1) = w_3(k) + \eta_D z(k) u(k) x_3(k) \\ z(k) = e(k) \end{cases} \quad (16)$$

式中: $u(k)$ 为单神经元控制器的输出; $\eta_I, \eta_P, \eta_D$ 分别为学习速率<sup>[9]</sup>。

由控制器公式可以看出,控制器的输出效果与比例系数、初始权值和学习率的选择有关。由于神经元有自学习功能,因此初始权值的选择对于控制器的输出影响不大,一般选择一个较小的值。对于比例系数  $K$ ,其对控制品质的影响引言中已分析过;另外,由公式也可以看出,权值的修正速度取决于学习率  $\eta_I, \eta_P, \eta_D$ 。目前,单神经元控制器的学习速率大都是基于大量的仿真和实验,在整个控制过程中保持恒定不变,这也限制了控制器品质的提高。

因此本文中结合无需辨识的自适应控制算法<sup>[10]</sup>,动态地调整增益系数  $K$  和学习率  $\eta$ ,调整规则如下所述。

1) 对于增益系数  $K$  的自适应调整规则。

$$K(k) = \begin{cases} \frac{cK(k-1)}{T_v(k-1)}, & \text{sgn}[e(k)] = \text{sgn}[e(k-1)] \\ 0.75K(k-1), & \text{sgn}[e(k)] \neq \text{sgn}[e(k-1)] \end{cases} \quad (17)$$

式中: $T_v(k) = T_v(k-1) + \Delta T_v(k)$ ,是控制器参数,一般初值选择一个较小的常数。它的校正方法如下: $\Delta T_v(k) = L \text{sgn}[|\Delta e(k)| - T_v(k-1) |\Delta^2 e(k)|]$ ,其中  $0.025 \leq c \leq 0.05, 0.05 \leq L \leq 0.1$ 。改进后的控制器增益系数能够根据误差进行在线校正,实现自适应调整。

2) 加入调整因子后对于学习率的调整规则<sup>[11]</sup>。

$$\begin{cases} \bar{\eta}_P = \lambda \eta_P, \bar{\eta}_I = \lambda \eta_I, \bar{\eta}_D = \lambda \eta_D, & \text{当 } |e(k)| \leq |e(k-1)| \\ \bar{\eta}_P = \rho \eta_P, \bar{\eta}_I = \rho \eta_I, \bar{\eta}_D = \rho \eta_D, & \text{当 } |e(k)| > |e(k-1)| \end{cases} \quad (18)$$

式中: $\bar{\eta}_P, \bar{\eta}_I$ 和 $\bar{\eta}_D$ 分别是调整之后的学习率; $\lambda$ 和 $\rho$ 是调整因子。对于学习率的调整规则是判断权值的修正

能否使得  $|e(k)|$  减小。 $\lambda$  和  $\rho$  是调整因子,如果当前时刻误差值小于或等于前时刻误差值,学习率乘以一个小于 1 的系数  $\lambda$ ,反之,则乘以一个大于 1 的系数  $\rho$ 。根据以上对于增益系数和学习率的调整,得到改进后的单神经元 PID 控制策略。

### 3 仿真及结果分析

考虑战术网络拥塞控制系统(5),采用控制器(14)、增益系数  $K$  的调整规则(17)和权值调整规则(18),进行闭环控制系统仿真。

系统参数选择如下:链路带宽  $C = 10^5$  包/s,往返时间  $R_0 = 0.03$  s,网络连接数  $N = 30$ ,由此可以求得被控对象传递函数为  $G(s) = \frac{7500000e^{-0.03s}}{(0.03s + 1)(1.5s + 1)}$ 。

给定期望队列长度  $q_0 = 100$ ,控制器设计中所需的参数设定为  $K = 0.18555 \times 10^{-5}$ ;  $K_p = 2.58392 \times 10^{-6}$ ;  $K_i = 2.9829 \times 10^{-6}$ ;  $K_d = 3.50735 \times 10^{-8}$ ;  $\eta_p = 0.80 \times 10^{-4}$ ;  $\eta_i = 0.10 \times 10^{-7}$ ;  $\eta_d = 0.10 \times 10^{-6}$ ;  $\lambda = 0.8$ ;  $\rho = 1.05$ 。

图 4 给出了未改进之前的系统队列长度的输出以及系统实际输出队列长度与期望队列长度的误差,图 5、图 6 给出了自适应调整增益系数和加入调整因子之后的系统队列长度输出以及系统实际输出队列长度与期望队列长度之差。

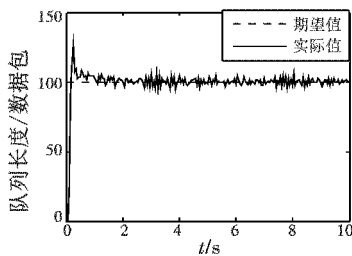


图 4 未改进之前的系统输出

Fig. 4 System output before improvement

由图 4 和图 5 可以看出,改进后的单神经元 PID 控制器输出曲线超调量有明显的降低,并且能够很快地跟随给定的队列长度  $q_0$ ,且响应曲线平滑。由图 6 的系统误差曲线可以看出,改进之前,误差曲线在零值附近持续抖动;而改进之后的控制器误差很快就能到达零值。仿真结果表明,改进后的算法有效提高了控制器品质。

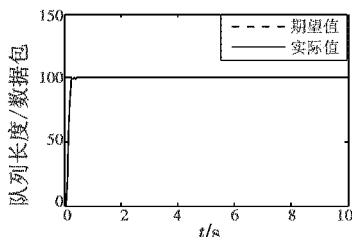


图 5 改进之后的系统输出

Fig. 5 System output after improvement

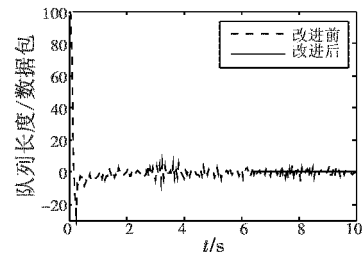


图 6 改进前后系统输出误差比较

Fig. 6 Comparison of system output error

### 4 结束语

本文针对战术网络的拥塞控制问题设计了改进的神经元 PID 控制器。考虑到单神经元 PID 算法中增益系数  $K$  和神经元权值系数学习率对于控制品质的影响,结合无需辨识自适应控制算法进行了改进,由上述分析和仿真结果可知,经过改进后的单神经元 PID 自适应控制策略能够有效降低有效抖动、误差。因此,本文所提算法为未来战术网络的拥塞控制提供了技术支撑。

### 参考文献

- [1] 潘清,胡欣杰,张晓清. 网络中心战装备体系[M]. 北京:国防工业出版社,2010.
- [2] 孙金生,李彬. 模糊神经元自适应 PID 控制 AQM 算法[J]. 东南大学学报:自然科学版,2010,40(s1):109-115.
- [3] 方红,王剑云. 基于电阻炉的单神经元 PID 控制器的设计及仿真[J]. 自动化与仪器仪表,2011,5:55-60.
- [4] MISRA V, GONG W, TOWSLEY D. Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED[C]//In proceeding ACM/SIGCOMM 2000, Sweden, 2000:151-160.
- [5] 尹凤杰,井元伟. 一种基于速率的单神经元自适应 PID 拥塞控制方法[J]. 控制与决策,2005,20(11):1225-1228,1234.
- [6] TAN L S, PUGH A C, YIN M. Rate-based congestion control in ATM switching networks using a recursive digital filter[J]. Control Engineering Practice, 2003, 11(10):1171-1181.
- [7] 张世韬,杨凤,郝骞. 单神经元 PID 控制器研究及仿真[J]. 机械工程与自动化,2009(3):69-70.
- [8] 贾立,陶鹏业,邱铭森. 基于模糊神经模型的自适应单神经元控制系统设计[J]. 华东理工大学学报:自然科学版,2008,34(1):135-139,143.
- [9] 刘金琨. 先进 PID 控制及 MATLAB 仿真[M]. 3 版. 北京:电子工业出版社,2011.
- [10] 赵建华,沈永良. 一种自适应 PID 控制算法[J]. 自动化学报,2001,27(3):417-420.
- [11] 孙夏娜,余群明,周兵,等. 主动悬架单神经元模糊 PID 控制策略与仿真[J]. 系统仿真学报,2009,21(8):2165-2168.