

基于 RBF 神经网络的直接自适应飞行控制器*

孙志兵 戴金海

(国防科技大学 航天与材料工程学院 湖南 长沙 410073)

摘要 :设计了一种径向基神经网络(RBF NN)飞行控制器结构,并给出了相应的控制律和参数调节律。由于调节了 RBF NN 的全部参数(连接权、高斯函数的中心和宽度),得到了很好的控制性能。以 F8 战斗机为控制对象进行了仿真分析,仿真表明,在存在 70% 的模型误差的情况下,该控制器仍然能实现较好的跟踪控制,表现出很好的鲁棒性,远远优于传统的只调节连接权值的算法。

关键词 :径向基函数神经网络;直接自适应;飞行控制器;仿真分析

中图分类号:TP183 文献标识码:A

Direct Adaptive Neuro Flight Controller Based on RBF NN

SUN Zhi-bing, DAI Jin-hai

(College of Aerospace and Material Engineering, National Univ. of Defense Technology, Changsha 410073, China)

Abstract :A flight controller structure based on RBF NN was designed and its control law and tuned law for RBF NN parameters.(center and width of gauss function, weight) were presented. F8 Fight aircraft was taken as the control object and a simulation was thus performed. The results from simulation show that, even under 70% model error, this controller manifests excellent control performance and good robustness. Fully-tuned law presented in this paper is better than the conventional algorithm which only adjusts weight value of RBF NN.

Key words :radial basis function neural network; direct adaptive; flight controller; simulation analyze

自 20 世纪 80 年代至今,基于人工神经网络的控制律与设计方法在飞行器控制领域逐步得到了深入研究。然而主要的应用范例还是集中于 BP 学习算法及其扩展算法。相比较而言,径向基神经网络(RBF NN)应用方面的文献较少^[1-3]。而 RBF NN 应用方面,都是使用经典方法实现网络结构,而且仅仅调节网络连接权。本文采用调节全部参数的 RBF NN,能够更迅速更精确地捕捉系统的动态,更适用于飞行器控制。对于高斯型 RBF NN 而言,调节的全部参数包括连接权、高斯函数的中心和宽度。启用 Kawato 的反馈误差学习方案^[5],以构造一个在线控制结构,实现直接自适应控制,而神经网络则采用全调节(高斯型)RBF NN,取代原来的多层前向网络。运用李雅普诺夫综合法导出稳定的 RBF NN 参数调节律,以保证整个系统的稳定性。

1 控制问题表述

用连续系统描述飞行器动态,有

$$\dot{x} = f(x, u) \quad (1)$$

上式中,假定 $f(\cdot)$ 是光滑的。 x 是 $n \times 1$ 维的状态向量, u 是 $p \times 1$ 维控制向量。不失一般性,假定 $f(0, 0) = 0$,那么原点就是一个平衡状态。控制输入的数量一般会小于状态数($p < n$),在这种条件下只有 p 个状态能被完美地加以跟踪。将 x 分割为 x_l 和 x_r ,那么飞行器动态方向变为:

$$\dot{\begin{matrix} x_l \\ x_r \end{matrix}} = \begin{matrix} f_l(x, u) \\ f_r(x, u) \end{matrix} \quad (2)$$

* 收稿日期:2006-08-29
基金项目:国家部委基金资助项目
作者简介:孙志兵(1969—),男,博士生。

研究的问题是设计一个神经控制器,使给定的 p 个状态 x_t 能够精确地跟踪期望的输入 x_{dt} ,同时其余的状态 x_r 渐进地逼近配平点 x_{dr} 。

如何确定使一个解 $u_d(t)$ 存在的条件,是要解决的控制问题之一。基于早期的研究^[6],为完成上述控制目标,系统必须满足假设 1。

假设 1 $f_t(x, u)$ 在沿着期望轨线 x_d 的所有点的一定领域内,有连续的有限偏微分。且 $\frac{\partial f_t(x, u)}{\partial u^T}$ 是非奇异的。

根据隐函数定理,期望的 $u_d(t)$ 可以表达为

$$u_d(t) = \bar{f}_t(x_d, \dot{x}_{dt}) \quad (3)$$

其中 \bar{f}_t 是 $p \times 1$ 维光滑函数,是 f_t 的逆函数。 $x_d = [x_{dt}^T, x_{dr}^T]^T$ 。

可以运用 RBF NN 控制器来逼近 (3) 式,并利用李雅普诺夫综合法推导 RBF NN 全部参数的调节律。

2 用于全调节 RBF NN 的稳定的调节律

2.1 控制策略与误差动态分析

将误差 e 定义为 $e = x - x_d$,其中 $x_d = [x_{dt}^T, x_{dr}^T]^T$ 。整个系统的误差动态为

$$\dot{e} = \dot{x} - \dot{x}_d = f(x, u) - f(x_d, u_d) \quad (4)$$

进行泰勒展开:

$$\dot{e} = \frac{\partial f(x, u)}{\partial x^T} \Big|_{x_d, u_d} (x - x_d) + \alpha(x - x_d) + \frac{\partial f(x, u)}{\partial u^T} \Big|_{x_d, u_d} (u - u_d) + \alpha(u - u_d) \quad (5)$$

其中 α 代表高阶项。将 $\frac{\partial f(x, u)}{\partial x^T} \Big|_{x_d, u_d}$ 用 $A(t)$ 来替代,将 $\frac{\partial f(x, u)}{\partial u^T} \Big|_{x_d, u_d}$ 用 $B(t)$ 代替,并忽略所有高阶项,得

$$\dot{e} \approx A(t)e + B(t)(u - u_d) \quad (6)$$

为说明概念起见,图 1 中的经典控制器使用一个满足假设 2 的比例控制器。

假设 2 基于稳定的飞行器模型,设计其反馈控制器,形成的闭环控制系统沿着期望的飞行轨迹是稳定的。

仅使用线性比例控制器, $u_p = K_p(t)e$,其误差的动态为:

$$\dot{e} = [A(t) + B(t)K_p(t)]e - B(t)u_d \quad (7)$$

总控制信号是比例控制器和 RBF NN 控制器信号的和:

$$u = K_p(t)e + u_m \quad (8)$$

考虑 RBFNN 的逼近作用,进一步推导得:

$$\begin{aligned} u &= \sum_{k=1}^h \hat{w}_k^T \exp\left(-\frac{1}{\hat{\sigma}_k^2} \|\xi - \hat{\mu}_k I\|^2\right) + K_p(t)e \\ &= \hat{W}^T \phi + K_p(t)e \end{aligned} \quad (9)$$

其中, ξ 为 RBF NN 的输入, $\xi = [x_d^T, \dot{x}_{dt}^T]^T$ 。 I 是单位矩阵。 ϕ 是 $h \times 1$ 维的高斯函数阵,它由中心 $\hat{\mu}$ 和宽度 $\hat{\sigma}$ 确定,字母上面加有符号的,表示计算量。 \hat{W} 是 $h \times p$ 维的权值矩阵(h 是稳态神经元个数), \hat{w}_k 是 \hat{W} 的第 k 行。

误差动态表达形式经推导可变为:

$$\begin{aligned} \dot{e} &= J(t)e - B(t)(\hat{W}^T \phi + \tilde{W}^T \phi + \tilde{W}^T \phi) - B(t)\epsilon_h \\ &\approx J(t)e - B(t)(\hat{W}^T \phi + \tilde{W}^T \phi) - B(t)\epsilon_h \end{aligned} \quad (10)$$

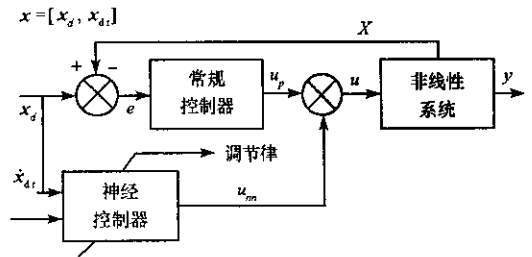


图 1 飞行器神经控制器结构
Fig.1 Structure of flight controller

其中 $J(t) = A(t) + B(t)K_p(t)$, \tilde{W} 为 W^* 与 \hat{W} 之差, 即 $\tilde{W} = W^* - \hat{W}$, W^* 是 $h \times p$ 维的最值权值矩阵。同样 $\tilde{\phi}$ 为 ϕ^* 与 $\hat{\phi}$ 之差, 即 $\tilde{\phi} = \phi^* - \hat{\phi}$, ϕ^* 是 $h \times 1$ 维的高斯函数阵, 它由最佳中心 μ^* 和最佳宽度 σ^* 确定。 ε_h 为固有逼近误差, 认为 ε_h 受限于一个上限(常数) ε_H 。

$$\varepsilon_H = \sup_{x_d \in X} \|\varepsilon_h(x_d, \hat{x}_{dt})\| \quad (11)$$

2.2 用于全调节 RBF NN 的稳定的自适应调节律

选取下面的李雅普诺夫函数, 用以推导出稳定的调节律:

$$V = \frac{1}{2} e^T P(t) e + \frac{1}{2} \text{tr}(\tilde{W}^T \Theta \tilde{W}) + \frac{1}{2} \tilde{\phi}^T \Lambda \tilde{\phi} \quad (12)$$

其中 $P(t)$ 是一个 $n \times n$ 维的时变、对称、正定矩阵。 Θ 、 Λ 均为 $h \times h$ 维的非负定矩阵。李雅普诺夫函数的导数为:

$$\begin{aligned} \dot{V} = & -e^T Q(t) e - \varepsilon_h^T B(t)^T P(t) e + \tilde{\phi}^T [-\hat{W} B(t)^T P(t) e + \Lambda \dot{\tilde{\phi}}] \\ & + \sum_{i=1}^p [-\tilde{w}_i^T \hat{\phi} B_i(t)^T P(t) e + \tilde{w}_i^T \Theta \dot{\tilde{w}}_i] \end{aligned} \quad (13)$$

其中 $Q(t) = -\frac{1}{2} [J(t)^T P(t) + P(t) J(t) + \dot{P}(t)]$, \tilde{w}_i 是 \tilde{W} 和第 i 列, $B_i(t)$ 是矩阵 $B(t)$ 的第 i 列。

如果按照下述方式选取 $\dot{\tilde{w}}$ 和 $\dot{\tilde{\phi}}$:

$$\dot{\tilde{w}}_i = \Theta^{-1} \hat{\phi} B_i(t)^T P(t) e, \quad i = 1, \dots, p \quad (14)$$

$$\dot{\tilde{\phi}} = \Lambda^{-1} \hat{W} B(t)^T P(t) e \quad (15)$$

那么(13)式变为:

$$\dot{V} = -e^T Q(t) e - \varepsilon_h^T B(t)^T P(t) e \quad (16)$$

根据(11)式, 有:

$$\dot{V} \leq -\|e\| \lambda_{\min}(Q) \|e\| + \|e\| \lambda_{\max}(P) \|B(t)\| \varepsilon_H^T \quad (17)$$

令 $\|B(t)\| \varepsilon_H = \delta_{\varepsilon_h}$, 可以直接推导出 \dot{V} 为非正的条件:

$$\|e\| \geq \frac{\lambda_{\max}(P)}{\lambda_{\min}(Q)} \delta_{\varepsilon_h} = E_a \quad (18)$$

根据(14)式进行推导, 得到离散形式的连接权调节律:

$$\hat{w}_k^T(n+1) = \hat{w}_k^T(n) - \tau \frac{\partial \hat{g}}{\partial \hat{w}_k^T} B(n)^T P(n) e(n), \quad k = 1, \dots, p \quad (19)$$

其中 τ 为采样时间, n 为采样指针。下标 k 表示对应于第 k 个隐含神经元的量。

根据(15)式进行推导, 得到高斯函数的中心与宽度的调节律:

$$\hat{\mu}_k(n+1) = \hat{\mu}_k(n) - \tau \eta_1 \frac{\partial \hat{g}}{\partial \hat{\mu}_k} B(n)^T P(n) e(n), \quad k = 1, \dots, h \quad (20)$$

$$\hat{\sigma}_k(n+1) = \hat{\sigma}_k(n) - \tau \eta_2 \frac{\partial \hat{g}}{\partial \hat{\sigma}_k} B(n)^T P(n) e(n), \quad k = 1, \dots, h \quad (21)$$

其中 $g = \hat{W}^T \hat{\phi}$, η_1 和 η_2 是适当选择的标量。在参数调节过程中, 整合一个与(18)式有关的限制策略, 能够保证系统的李雅普诺夫稳定性——即当 e 小于某个指定的误差 e_0 时, 停止更新参数。因为很难直接求解 E_a , 所以人为地选取一个常值误差 e_0 , 当满足 $e_0 \geq E_a$ 时, 只要 $e \geq e_0$, 则(18)式成立。 e_0 要在计算中多次试验选取, 尽可能选小值。

矩阵 $B(t)$ 在调节律中扮演了一个重要角色。所以, 要么采取某种策略来辨识它, 要么依靠鲁棒控制策略(如滑模控制)来保证控制性能。

3 仿真分析

为了进行比较,将全调 RBF NN 控制器用于 F8 战斗机模型的指令跟踪。比较了全调节 RBF NN 控制器以及只调节连接权的 RBF NN 控制器的跟踪性能。

F8 战斗机模型的主要表达式为:

$$x(n+1) = F_1 x(n) + G_1 u(n) \quad (22)$$

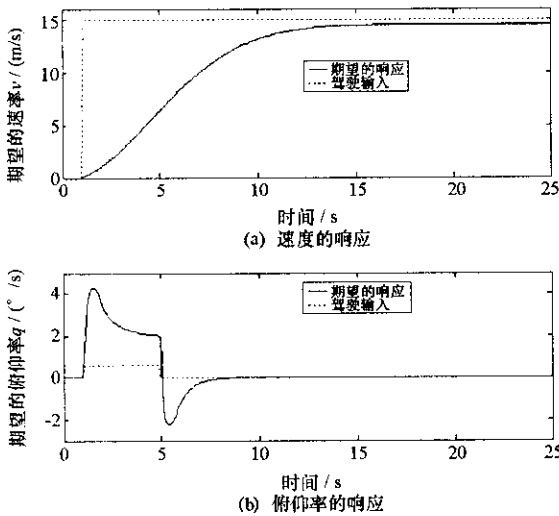


图 2 速度和俯仰率的期望轨迹

Fig. 2 Desired trajectory of velocity and pitch rate

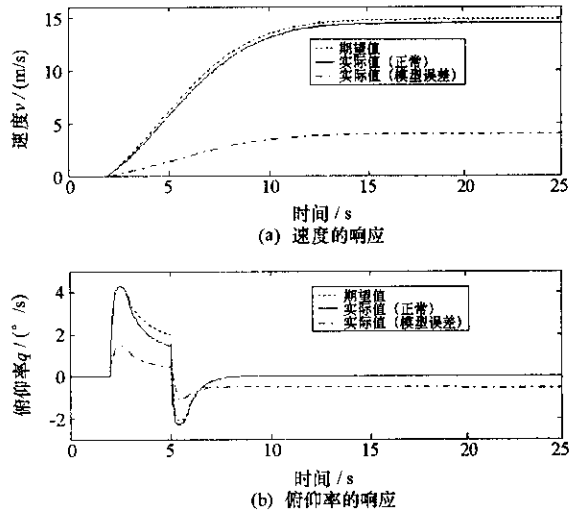


图 3 RBF NN 在 70% 模型误差下的响应

Fig. 3 RBF NN response under 70% model error

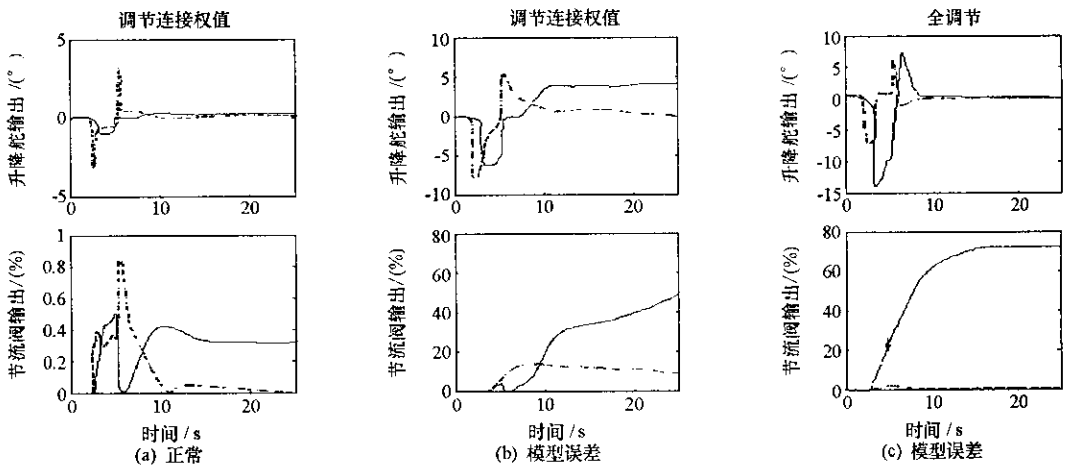


图 4 控制输入(带有 6 个隐含单元的 RBF NN)

Fig. 4 Control input(RBF NN with 6 hidden units)

模型详细情况见参考文献 [8]。仿真中,通过在 $t \geq 0$ 时,将系统矩阵 F_1 和 G_1 的所有元素改变 70% 来模拟模型误差。要求的驾驶输入及期望的响应,分别绘于图 2(a)(b)。图 3、图 4 是存在模型误差时,调节全部参数与只调节连接权时的结果比较,还比较了采用不同隐含神经元时的情况。显然,隐含单元越多,跟踪误差越小。从图 3 还可看出,使用全调节 RBF NN,其跟踪性能远远优于只调节连接权的经典网络。

图 4 表明了 RBF NN 控制器是如何地在线学习系统的逆动态。图中的实线表示 RBF NN 控制器的输出,虚线表示传统控制器的输出。使用只调节连接权的 RBF NN 控制器时,两个控制器在额定条件下的输出以及模型误差条件下的输出,均绘于图中。从图中看出,在初始阶段,经典控制器对总的性

能作出主要贡献,此期间,RBF NN 控制器的输出是变化的,表示它还处于学习过程。在额定条件下,10s 后,稳态输出主要来自于 RBF NN 控制器,而反馈控制器的输出收缩到 0,这说明 RBF NN 已经完成了对系统逆的学习。当存在模型误差时,传统的 RBF NN 无法成功地捕捉系统的逆动态,这是因为使用了错误的中心和宽度参数。经过比较可以看出,使用全调节 RBF NN,能够迅速地捕捉模型动态的改变。

需要指出,如果使用传统的 RBF NN,就必须根据系统的先验知识,精确地确定隐含单元的中心和宽度参数。而使用全调节 RBF NN 时,由于其中心和宽度参数能够在线地进行计算,因此无需根据系统的先验知识来预先确定网络参数,而且能够避免因人为地估计参数带来的性能损失。

仿真分析表明,由于调节了 RBF NN 的全部参数,极大地提高了控制精度。而且鲁棒性很好,当存在很大的模型误差时,仍然能够很好地完成控制任务。考虑高达 70% 的模型误差,是为了表现控制器的强鲁棒性。

值得注意的是,矩阵 $B(t)$ 在调节律中扮演了一个重要角色。所以,要么采取某种策略来辨识它,要么依靠鲁棒控制策略(如滑模控制)来保证控制性能。

参 考 文 献 :

- [1] Calise A J, Rysdyk R T. Nonlinear Adaptive Flight Control Using Neural Networks[J]. IEEE Control Systems Magazine, 1998, 18(6): 14-25.
- [2] Singh S N, Wells W R. Direct Adaptive and Neural Control of Wing-rock Motion of Slender Delta Wings[J]. AIAA Journal of Guidance, Control, and Dynamics, 1995, 18(1): 25-30.
- [3] Byoung S K, Calise A J. Nonlinear Flight Control Using Neural Networks[J]. AIAA Journal of Guidance, Control, and Dynamics, 1997, 20(1): 26-33.
- [4] Sanner R M, Slotine J J. Gaussian Networks for Direct Adaptive Control[J]. IEEE Trans, on Neural Networks, 1992, 3(6): 837-863.
- [5] Gomi H, Kawato M. Neural Network Control for a Closed-loop System Using Feedback-error-learning[J]. Neural Networks, 1993, 6(7): 933-946.
- [6] Narendra K S, Mukhopadhyay S. Adaptive Control Using Neural Networks and Approximate Models[J]. IEEE Trans, on Neural Networks, 1997, 8(3): 475-485.
- [7] Park J, Sandberg I W. Universal Approximation Using Radial-basis-function Networks[J]. Neural Computation, 1991(3): 246-257.
- [8] Sadhukhan D, Feteih S. F8 Neurocontroller Based on Dynamic Inversion[J]. Journal of Guidance, Control and Dynamics, 1996, 19(1): 150-156.

(上接第 29 页)

参 考 文 献 :

- [1] Wingrove R C. A Survey of Atmosphere Reentry Guidance and Control Methods[R]. IAS Paper No. 63-86, 1963.
- [2] Morth H R. Reentry Guidance for Apollo[R]. MIT Instrumentation Lab Report R-532, Jan. 1966.
- [3] Marcus F J, Kriegsmann B A. An Entry Guidance System for the Shuttle Vehicle[C]//IFAC Paper, Session No. 1, 5th IFAC Symposium on Automatic Control in Space, Session No. 1 Genoa, 1973.
- [4] Causey W, Sohoni V. Reentry Guidance for Space Shuttle[C]//IFAC Paper, Session No. 1, 5th IFAC Symposium on Automatic Control in Space, Session No. 1 Genoa, 1973.
- [5] Spratlin R W. An Adaptive Numeric Predictor-Corrector Guidance Algorithm for Atmospheric Entry Vehicles[D]. M.S. Thesis, Dept. of Aeronautics and Astronautics, Massachusetts Inst. of Technology, Cambridge, MA, May 1987.
- [6] Harpold J C, Graves C A. Shuttle Entry Guidance[J]. Journal of the Astronautical Science, 1979, 27(3): 239-268.
- [7] Fuhry D P. Adaptive Atmosphere Reentry Guidance for the Kistler K-1 Orbital Vehicle[R]. AIAA 99-4211, 1998.
- [8] Youssef H, Chowdhry R S, Lee H, et al. Predictor-corrector Entry Guidance for Reusable Launch Vehicles[R]. AIAA 2001-4043, 2001.

