文章编号：1000 - 8152(2002)04 - 0777 - 04

一种快速对角回归神经网络控制算法

扈宏杰，尔联洁，刘金琨
(北京航空航天大学 自动控制系，北京 100083)

摘要：文[1]定理 1 给出了一个基于 Lyapunov 函数的三层对角回归神经网络(DRNN)任意权参数学习速率的自适应调整算法，而推导各层权自适应学习速率时没有严格满足定理 1 成立的必要条件，故没能找到各学习速率的准确范围.依据文[1]定理 1，精确给出了各权向量及权矩阵学习速率的调整算法，结果表明 DRNN 应具有更大的学习速率，对应更加快速的收敛算法.给出了相应的仿真结果.

关键词：对角回归神经网络；自适应学习速率；权向量及权矩阵；收敛性

中图分类号：TP183　　　**文献标识码**：A

Fast algorithm for diagonal recurrent neural networks control system

HU Hong-jie, ER Lian-jie, LIU Jin-kun
(Department of Control, Beijing University of Aeronautics & Astronautics, Beijing 100083, China)

Abstract: Convergence Theorem 1 in Ref.[1] was given for three layers diagonal recurrent neural networks (DRNN) by introducing a Lyapunov function. Because the essential condition to Theorem 1 was neglected upper limits of learning rates for every weight vectors and matrix were not attained. Much bigger learning rates of all weight vectors and matrix are deduced precisely on the basis of convergence theorem 1 in Ref. [1], so a fast iterative algorithm is obtained. Simulation results are included.

Key words: DRNN; adaptive learning rate; weight vector and weight matrix; convergence

1 引言(Introduction)

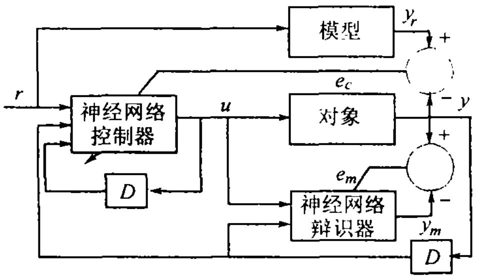

图 1　DRNN 控制系统
Fig. 1　DRNN control system

文[1]中给出了基于对角回归神经网络(DRNN)的模型参考自适应控制方案，其结构如图 1 所示.其中，神经网络控制器(DRNNC)和神经网络辩识器(DRNNI)均采用了图 2 所示的三层结构，隐含层为回归层.网络的所有输入与所有回归神经元之间的连接权采用统一的学习速率进行迭代.各层权的通用学习速率的迭代算法(文[1]定理 1)是依据 Yabuta[2]的误差理论，基于 Lyapunov 函数的收敛定理而得出，但在权矩阵学习速率的推导过程中违背了文[1]定理 1 成立的条件.在本文中，严格按照文[1]定理 1 的结论，从权向量的角度出发，重新推导了各权向量及权矩阵学习速率的调整算法.

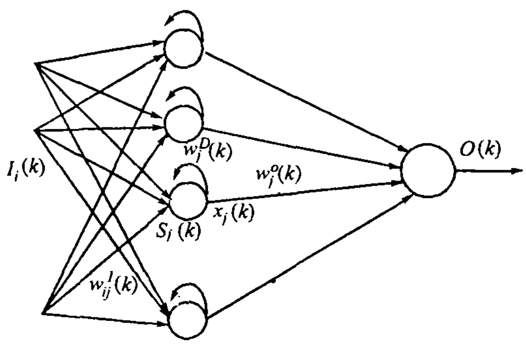

图 2　DRNN 结构
Fig. 2　DRNN structure

2 DRNN 迭代算法(Iterative algorithm for DRNN)

DRNN 的数学模型为：

收稿日期：2000 - 09 - 19；收修改稿日期：2001 - 05 - 09.

$$\begin{cases} O(k) = \sum_j W_j^O X_j(k), \\ X_j(k) = f(S_j(k)), \\ S_j(k) = W_j^D X_j(k-1) + \sum_i W_{ij}^I I_i(k). \end{cases} \tag{1}$$

$I_i(k)$ 为 DRNN 的第 i 个输入；$S_j(k)$ 为第 j 个回归元的输入之和；$X_j(k)$ 为第 j 个回归元的输出；$f(\cdot)$ 为 Sigmoid 函数；$O(k)$ 为 DRNN 的输出，若 DRNN 为控制器 DRNNC，则 $O(k) = u(k)$. 若 DRNN 为辩识器 DRNNI，则 $O(k) = y_m(k)$. W^D 和 W^O 分别为 DRNN 的回归层和输出层的权向量，且二者皆为 n_h 维；W^I 为 DRNN 的输入层的权矩阵. 现将 W^I 表示为：

$$W^I = [W_1^I \quad W_2^I \quad \cdots \quad W_j^I \quad \cdots \quad W_{n_h}^I]. \tag{2}$$

其中，W_j^I 为 DRNN 的输入与第 j 个回归元之间所构成的权向量，且为 n_I 维. 其中，n_I 和 n_h 分别为 DRNN 的输入个数和隐含层回归元个数.

DRNNI 权迭代算法采用如下指标函数：

$$J_m = \frac{1}{2}(y(k) - y_m(k))^2 = \frac{1}{2}e_m^2(k). \tag{3}$$

DRNNC 权迭代算法采用如下指标函数：

$$J_c = \frac{1}{2}(y_r(k) - y(k))^2 = \frac{1}{2}e_c^2(k). \tag{4}$$

由负梯度下降法及链式法则可得 DRNNI 的任一权向量迭代算法如下：

$$W(k+1) = W(k) + \eta_I\left(-\frac{\partial J_m}{\partial W}\right) = W(k) + \eta_I e_m(k)\frac{\partial O(k)}{\partial W}. \tag{5}$$

其中，η_I 为权向量 W 的学习速率.

同理，可得 DRNNC 的任一权向量迭代算法如下：

$$W(k+1) = W(k) + \eta_c\left(-\frac{\partial J_c}{\partial W}\right) \approx W(k) + \eta_c e_c(k) y_u \frac{\partial O(k)}{\partial W}. \tag{6}$$

其中，η_c 为权向量 W 的学习速率，$y_u \equiv \partial y_m(k)/\partial u(k)$[1] 表示对象的输出对输入的灵敏度.

令网络输出对网络任一权向量 W 的变化率为

$$g(k) = \partial O(k)/\partial W(k). \tag{7}$$

又由式(1)可知：

$$\frac{\partial O(k)}{\partial W_j^O(k)} = X_j(k), \tag{8}$$

$$\frac{\partial O(k)}{\partial W_j^D(k)} = W_j^O P_j(k), \tag{9}$$

$$\frac{\partial O(k)}{\partial W_{ij}^I(k)} = W_j^O Q_{ij}(k). \tag{10}$$

其中，

$$P_j(k) \equiv \partial X_j(k)/\partial W_j^D(k) = f'(S_j)X_j(k-1), \tag{11}$$

$$Q_{ij}(k) \equiv \partial X_j(k)/\partial W_{ij}^I(k) = f'(S_j)I_i(k). \tag{12}$$

且由 $0 < f'(S_j) < 0.5$ 及 $0 < X_j(k) < 1$，故 $0 < P_j(k) < 0.5$.

2.1 DRNNI 迭代算法(Iterative algorithm for DRNNI)

文[1]定理 1 中给出了一个基于 Lyapunov 函数的各层权所通用的学习速率调整法

$$0 < \eta_I < \frac{2}{g_{I,\max}^2}. \tag{13}$$

其中

$$g_{I,\max} := \max_k \| g_I(k) \|,$$

$$g_I(k) = \partial O(k)/\partial W_I(k),$$

$\|\cdot\|$ 定义为欧几里德范数(Euclidean-norm)，$W_I(k)$ 为一个 n 维向量，且为式(13)成立的一个必要条件. 而文[1]在利用式(13)推导输入层权的学习速率时，直接推导权矩阵的学习速率，违背了这个必要条件，因而没有得出最优结果.

为严格依据式(13)的结论，从权向量的角度出发推导各权向量及矩阵的学习速率，首先定义如下权向量和权矩阵：

$$W_I^O = [w_{I,1}^O, w_{I,2}^O, \cdots, w_{I,j}^O, \cdots, w_{I,h_I}^O]^{\mathrm{T}}, \tag{14}$$

$$W_I^D = [w_{I,1}^D, w_{I,2}^D, \cdots, w_{I,j}^D, \cdots, w_{I,h_I}^D]^{\mathrm{T}}, \tag{15}$$

$$W_I^I = [W_{I,1}^I \quad W_{I,2}^I \quad \cdots \quad W_{I,j}^I \quad \cdots \quad W_{I,h_I}^I]. \tag{16}$$

其中，$W_{I,j}^I = [w_{I,1j}^I, w_{I,2j}^I, \cdots, w_{I,ij}^I, \cdots, w_{I,n_Ij}^I]^{\mathrm{T}}$，$n_I$ 为 DRNNI 的输入数，h_I 为 DRNNI 隐含层回归元数，$W_{I,j}^I$ 为 DRNNI 的输入与隐含层第 j 个回归元之间的连接权所构成的权向量，W_I^O 和 W_I^D 分别表示网络的输出层和回归层的权向量，W_I^I 表示输入层的权矩阵，则有如下 DRNNI 收敛定理.

定理 1 设 η_I^O，η_I^D 和 $\eta_{I,j}^I$ 分别表示 DRNNI 的权向量 W_I^O，W_I^D 和 $W_{I,j}^I$ 的学习速率. η_I^I 表示权矩阵 W_I^I 的学习速率. 如满足 $0 < |W_{I,j}^D| < 1, j = 1,2,\cdots,h_I$，且学习速率按如下方式选择：

$$0 < \eta_{I,j}^I < \frac{8}{n_I}\left[\frac{1}{w_{I,j\max}^O I_{I,\max}}\right]^2, \ j = 1,2,\cdots,h_I, \tag{17}$$

$$0 < \eta_I^I < \frac{8}{n_I}\left[\frac{1}{W_{I,\max}^O I_{I,\max}}\right]^2, \tag{18}$$

$$0 < \eta_I^O < \frac{2}{h_I}, \tag{19}$$

$$0 < \eta_I^D < \frac{8}{h_I}\left[\frac{1}{W_{I,\max}^O}\right]^2, \tag{20}$$

则迭代算法收敛.其中

$$w_{I,j\max}^O := \max_k | w_{I,j}^O(k) |,$$
$$W_{I,\max}^O := \max_k \| W_I^O(k) \|,$$
$$I_{I,\max} := \max_k \| I_I(k) \|,$$

$\|\cdot\|$ 定义为无穷范数(sup-norm).

证　为求 $W_{I,j}^I$ 的学习速率,定义 $Q_{I,ij}(k) \equiv \partial X_j(k)/\partial W_{I,ij}^I(k)$. 由式(10)可得 DRNNI 的输出对权向量 $W_{I,j}^I$ 的变化率为

$$\left\|\frac{\partial O(k)}{\partial W_{I,j}^I(k)}\right\| = \| [w_{I,j}^O Q_{I,1j}(k), w_{I,j}^O Q_{I,2j}(k), \cdots, w_{I,j}^O Q_{I,ij}(k), \cdots, w_{I,j}^O Q_{I,n_j}(k)]^{\mathrm{T}} \| \leqslant | w_{I,j}^O | 0.5\sqrt{n_I}\, | I_{I,\max}(k) |. \tag{21}$$

定义

$$w_{I,j\max}^O \equiv \max_k | w_{I,j}^O(k) |,$$
$$I_{I,\max} := \max_k \| I_I(k) \|,$$
$$| w_{I,j}^O | \leqslant w_{I,j\max}^O,$$

则有:

$$\left\|\frac{\partial O(k)}{\partial W_{I,j}^I(k)}\right\| \leqslant 0.5\sqrt{n_I} I_{I,\max} w_{I,j\max}^O. \tag{22}$$

由式(13),可得 $W_{I,j}^I$ 的学习速率为

$$0 < \eta_{I,j}^I < \frac{8}{n_I}\left[\frac{1}{w_{I,j\max}^O I_{I,\max}}\right]^2. \tag{23}$$

即式(17)得证.

定义 $W_{I,\max}^O := \max_k \| W_I^O(k) \|$, 则

$$w_{I,j\max}^O \leqslant W_{I,\max}^O.$$

可选 $W_{I,j}^I$ 的学习速率为

$$0 < \eta_{I,j}^I < \frac{8}{n_I}\left[\frac{1}{W_{I,\max}^O I_{I,\max}}\right]^2 \leqslant \frac{8}{n_I}\left[\frac{1}{w_{I,j\max}^O I_{I,\max}}\right]^2. \tag{24}$$

取 W_I^I 的学习速率为

$$\eta_I^I = \min_j | \eta_{I,j}^I |,\ j = 1,2,\cdots,n_I. \tag{25}$$

即式(18)得证.类似地可得式(19)和(20).

2.2 DRNNC 迭代算法(Iterative algorithm for DRNNC)

设 n_C 为 DRNNC 的输入数, h_C 为隐含层回归元数, $W_{C,j}^I$ 为 DRNNC 的输入与第 j 个回归元之间的连接权所构成的权向量.参见文[1]中定理 3,类似定理 1 的证明方法,可得 DRNNC 的收敛定理如下:

定理 2　设 η_C^O, η_C^D 和 $\eta_{C,j}^I$ 分别表示权向量 W_C^O, W_C^D 和 $W_{C,j}^I$ 的学习速率, η_C^I 表示权矩阵 W_C^I 的学习速率. 如果满足 $0 < | W_{C,j}^D | < 1, j = 1,2,\cdots,h_C$, 并且,学习速率的选取满足如下条件:

$$0 < \eta_C^O < \frac{2}{h_C S_{\max}^2}, \tag{26}$$

$$0 < \eta_C^O < \frac{8}{h_C S_{\max}^2}\left[\frac{1}{W_{C,\max}^O}\right]^2, \tag{27}$$

$$0 < \eta_{C,j}^I < \frac{8}{n_C S_{\max}^2}\left[\frac{1}{w_{C,j\max}^O}\right]^2,\ j = 1,2,\cdots,h_C, \tag{28}$$

$$0 < \eta_C^I < \frac{8}{n_C S_{\max}^2}\left[\frac{1}{W_{C,\max}^O I_{C,\max}}\right]^2. \tag{29}$$

其中,定义

$$W_{C,\max}^O := \max_k \| W_C^O(k) \|,$$
$$w_{C,j\max}^O \equiv \max_k | w_{C,j}^O(k) |,$$
$$I_{C,\max} := \max_k \| I_C(k) \|,$$
$$I_C(k) = \{r(k), u(k-1), y(k-1)\},$$
$$S_{\max} \equiv \frac{h_c}{2} W_{I,\max}^O W_{I,1\max}^I \text{[1]},$$
$$W_{I,1\max}^I := \max_k \| W_{I,1(k)}^I \|,$$
$$\| W_{I,1(k)}^I \| := \max_j \| W_{I,1j}^I(k) \|,$$

$\|\cdot\|$ 为无穷范数(sup-norm).

证　(略).

3　仿真结果(Simulation results)

设被控对象的模型为

$$y(k+1) = \frac{y(k)}{1 + y^2(k)} + u^3(k). \tag{30}$$

取辩识网络 DRNNI 的输入为

$$I_I(k) = \{u(k), y(k-1)\},$$

DRNNI 的结构取为 2-5-1,初始学习速率均选取为 0.1.设被控对象的控制输入为 $u(k) = 0.5\sin(8\pi k)$,从图 3 及图 4 为 DRNNI 的辩识结果.

针对图 1 所示的控制结构,取控制网络 DRNNC 的输入为

$$I_C(k) = \{r(k), u(k-1), y(k-1)\},$$

取参考模型为

$$y_r(k+1) = 0.6 y_r(k) + r(k),$$

被控对象的模型仍为式(30),DRNNC 的结构取为 3-7-1,初始学习速率均选取为 0.1,指令信号为 $r(k) = \sin(2\pi k/25) + \sin(2\pi k/10)$ 时,图 5 及图 6 给出模型跟踪控制结果.

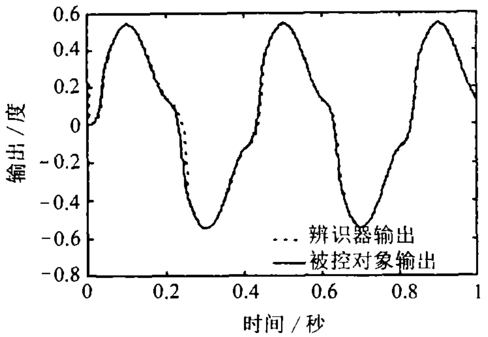

图3 文[1]的辨识结果
Fig. 3 Identification result by means of [1]

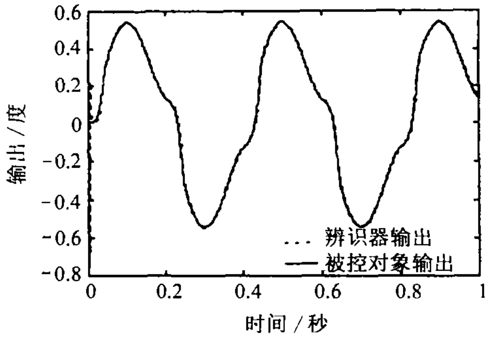

图4 本文方法的辨识结果
Fig. 4 Identification result by means of this paper

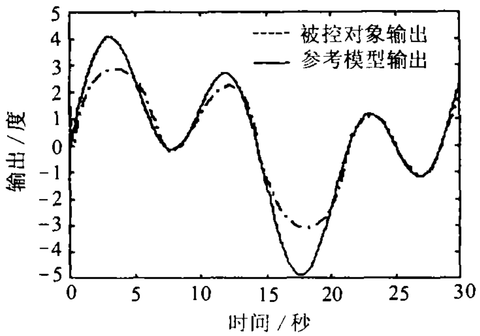

图5 文[1]方法的跟踪结果
Fig. 5 Tracking result by [1]

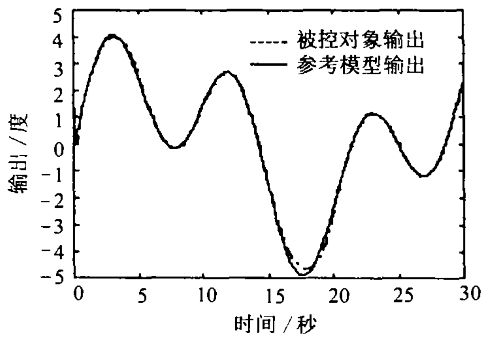

图6 本文方法的跟踪结果
Fig. 6 Tracking result by means of this paper

4 结论(Conclusion)

比较本文定理1及2与文[1]中的定理2及4,结合仿真结果可见,该迭代算法具有更大的学习速率.由于在迭代算法稳定收敛的前提下,收敛速度正比于迭代步长(学习速率)的大小,故本文得出了一种基于 Lyapunov 函数的快速 DRNN 的权参数的收敛算法.

参考文献(References)

[1] Ku Chao-Chee, Lee Kwang Y. Diagonal recurrent neural networks for dynamic system control [J]. IEEE Trans. Neural Networks, 1995,6(1):144-156

[2] Ybuta T, Yamada T. Learning control using neural networks [A]. Proc. IEEE int. Conf. Robotics and Automation [C]. Saramento, CA, USA,1991,740-745

本文作者简介

應宏杰 1962年生.博士,讲师.研究方向为运动系统的VSS、神经模糊控制及其飞行仿真系统的设计与实现. Email: xuchunmei1030@sohu.com

尔联洁 1938年生.博士导师.研究方向为飞行仿真系统的设计、实现及相应的控制策略.

刘金琨 1965年生.博士.副教授.研究方向为智能控制.