

采用 BP 网络辨识航空发动机数学模型

西北工业大学 刘小勇** 樊思齐

【摘要】 运用 BP 网络和实测数据作为学习样本, 对某型航空发动机的数学模型进行了辨识研究。辨识模型输出的结果与实测数据比较误差较小。这种方法收敛速度快、精度高, 结果表明用 BP 网络辨识方法能够得到比较精确的发动机数学模型。

主题词: 航空发动机 模型法 神经网络 非线性 辨识

分类号: O 235

1 引言

航空发动机的数学模型是对其进行复杂控制的基础, 然而航空发动机理论建模困难, 系统辨识方法不失为一种较好的建模方法, 但是以往的辨识方法, 都是在假定航空发动机是线性时不变系统的基础上来进行的。因而这些方法存在着辨识精度差、收敛速度慢等不足。神经网络具有能逼近高度复杂的非线性、并行高速处理、高度容错、联想记忆、学习简便等优点, 因而神经网络为航空发动机的辨识研究提供了重要途径。本文采用 BP 网络来辨识航空发动机的数学模型, 学习的样本来自某型飞机试飞时录取的发动机参数。从对多种不同状态进行辨识的结果来看, 这种辨识方法有精度高、学习时间短等优点, 能够满足工程实际的要求。

2 辨识方法

设航空发动机系统可由下列非线性方程描述

$$N_{l,k+1} = f_l(N_{l,k}, N_{h,k}, M_{f,k}) \quad , \quad N_{h,k+1} = 1 = f_h(N_{l,k}, N_{h,k}, M_{f,k})$$

$N_{l,k}$ 、 $N_{h,k}$ 、 $M_{f,k}$ 分别表示 k 时刻发动机的低压转子转速、高压转子转速和发动机的供油量。

在隐层节点不限的条件下, 三层神经网络(隐层数为 1)能逼近任意非线性, 故采用三层神

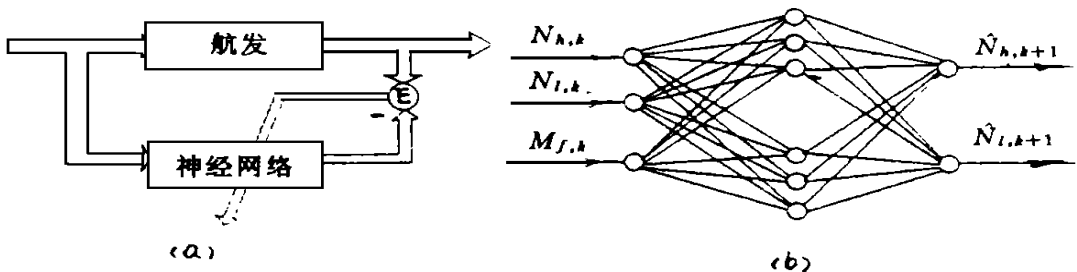


图 1 辨识结构图

神经网络来逼近以上的非线性映射 f_l 和 f_h 。辨识的结构图 1 所示。图中 $\hat{N}_{l,k+1}$ 、 $\hat{N}_{h,k+1}$ 表示神经

网络输出对应于发动机高、低压转子转速的数值。神经网络中隐层节点数为 30。

在神经网络中, 用广义 S 型函数描述神经元的输入输出特性 $O_{i,k} = \alpha_{i,k} / [1 + \exp(-\beta_{i,k} \text{net}_{i,k})]$ 。式中 $\alpha_{i,k}$ 为控制节点输出饱和值, $\beta_{i,k}$ 为调节 S 型函数的斜率, $O_{i,k}$ 为第 k 层第 i 节点的输出, $\text{net}_{i,k}$ 为第 k 层第 i 节点输入。它由下式给出

$$\text{net}_{i,k} = \sum_j [W_{i,j,k} O_{j,k-1}] + \theta_{i,k} \quad (1)$$

式中 $W_{i,j,k}$ 为第 $k-1$ 层第 j 节点对 k 层第 i 节点的权值, $\theta_{i,k}$ 为第 k 层第 i 节点的阈值。

对于给定的输入和输出对, 定义误差为

$$E = \frac{1}{2} \sum_{j=1}^q [d_j - O_{j,n}]^2 \quad (2)$$

式中 n 为神经网络总层数, q 为神经网络输出节点数。 d_j 为实际测量的发动机高、低压转子转速。误差 E 为神经网络输出对应的发动机高、低压转子转速的数值与实际测量的发动机高、低压转子转速之差的平方和, 它利用当前的学习参数值, 在前向通道中计算。

学习过程是通过适当调整学习参数 $W_{i,j,k}$, $\theta_{i,k}$, $\alpha_{i,k}$ 及 $\beta_{i,k}$ 使 E 趋于极小, 即网络的输出值与实测数据的差趋于极小。采用梯度下降法调整学习参数的公式为

$$\begin{aligned} W_{i,j,k}^{(m)} &= W_{i,j,k}^{(m-1)} - \eta_1 (\partial E / \partial W_{i,j,k})^{(m)}, \quad \theta_{i,k}^{(m)} = \theta_{i,k}^{(m-1)} - \eta_2 (\partial E / \partial \theta_{i,k})^{(m)} \\ \alpha_{i,k}^{(m)} &= \alpha_{i,k}^{(m-1)} - \eta_3 (\partial E / \partial \alpha_{i,k})^{(m)}, \quad \beta_{i,k}^{(m)} = \beta_{i,k}^{(m-1)} - \eta_4 (\partial E / \partial \beta_{i,k})^{(m)} \end{aligned}$$

其中 m 为迭代步数, η_1 , η_2 , η_3 , η_4 为学习步长。

利用 (1) 式和 (2) 式可以得到如下偏导计算公式

$$\begin{aligned} \partial \text{net}_{i,k} / \partial W_{i,j,k} &= O_{j,k-1}, \quad \partial \text{net}_{i,k} / \partial \theta_{i,k} = 1, \quad \partial O_{i,k} / \partial \alpha_{i,k} = O_{i,k} / \alpha_{i,k} \\ \partial O_{i,k} / \partial \beta_{i,k} &= O_{i,k} (1 - O_{i,k} / \beta_{i,k}) \text{net}_{i,k}, \quad \partial O_{i,k} / \partial \text{net}_{i,k} = O_{i,k} (1 - O_{i,k} / \alpha_{i,k}) \end{aligned} \quad (3)$$

以上各偏导数都可以在前向通道中各个神经元局部计算。

$$\text{假设} \quad \partial E / \partial \text{net}_{i,k} = -\delta_{i,k}, \quad \partial E / \partial O_{i,k} = -\zeta_{i,k} \quad (4)$$

与 E 有关学习参数的偏导数为

$$\begin{aligned} \partial E / \partial W_{i,j,k} &= (\partial E / \partial \text{net}_{i,k}) (\partial \text{net}_{i,k} / \partial W_{i,j,k}) = -\delta_{i,k} O_{j,k-1} \\ \partial E / \partial \theta_{i,k} &= (\partial E / \partial \text{net}_{i,k}) (\partial \text{net}_{i,k} / \partial \theta_{i,k}) = -\delta_{i,k} \\ \partial E / \partial \alpha_{i,k} &= (\partial E / \partial O_{i,k}) (\partial O_{i,k} / \partial \alpha_{i,k}) = -\zeta_{i,k} (O_{i,k} / \alpha_{i,k}) \\ \partial E / \partial \beta_{i,k} &= (\partial E / \partial O_{i,k}) (\partial O_{i,k} / \partial \beta_{i,k}) = -\zeta_{i,k} O_{i,k} (1 - O_{i,k} / \alpha_{i,k}) \text{net}_{i,k} \end{aligned} \quad (5)$$

上式中除 $\delta_{i,k}$ 和 $\zeta_{i,k}$ 外, 其它所有量均在前向通道计算得到, δ , ζ 通过误差在神经网络反向传播中得到。

考虑输出层- $\delta_{i,n} = \partial E / \partial \text{net}_{i,n} = (\partial E / \partial O_{i,n}) (\partial O_{i,n} / \partial \text{net}_{i,n})$, 利用 (2) 式和 (3) 式有

$$\delta_{i,n} = (d_i - O_{i,n}) O_{i,n} (1 - O_{i,n} / \alpha_{i,n}) \beta_{i,n}$$

利用 (2) 式、(4) 式和 (5) 式得 $\zeta_n = d_i - O_{i,n}$ 对于较低层

$$-\zeta_{i,k} = \partial E / \partial O_{i,k} = \sum_j (\partial E / \partial \text{net}_{j,k+1}) (\partial \text{net}_{j,k+1} / \partial O_{i,k})$$

这里对第 $k+1$ 层所有节点求和, 利用 (1) 式和 (4) 式得简化结果如下

$$\zeta_{i,k} = \sum_j [\delta_{j,k+1} W_{j,i,k+1}], \quad \delta_{i,k} = \zeta_{i,k} O_{i,k} (1 - O_{i,k} / \alpha_{i,k}) \beta_{i,k}$$

$\zeta_{i,k}$ 取决于第 $k+1$ 层的 δ , 计算时从输出层逐层计算到输入层, 在同一层里 ζ 可以并行计算。多层前向神经网络误差反向计算并行运算的意义就在于此, 它大大加快了学习算法的运

行速度, 简化了学习的复杂程度。在运用以上算法时, 对发动机的实测参数进行了无因次化处理, 学习参数 $W_{i,j,k}$, $\theta_{i,k}$, $\alpha_{i,k}$ 及 $\beta_{i,k}$ 的初始值在 $[0, 1]$ 区间上随机选取。隐层节点数初值取为 10, 为使误差能达到要求须通过程序调整节点数和学习次数。最后得到的节点数为 30。

3 仿真分析

本文利用某型飞机在飞行高度为 3 km, 马赫数为 0.67; 飞行高度为 8.6 km, 马赫数为 0.97; 飞行高度为 13 km, 马赫数为 1.47; 以及飞行高度为 18 km, 马赫数为 1.97 等 4 种状态下录取的某型发动机参数, 作为样本数据, 进行了辨识研究。学习步长 η 皆为 0.95, 按 (2) 式计

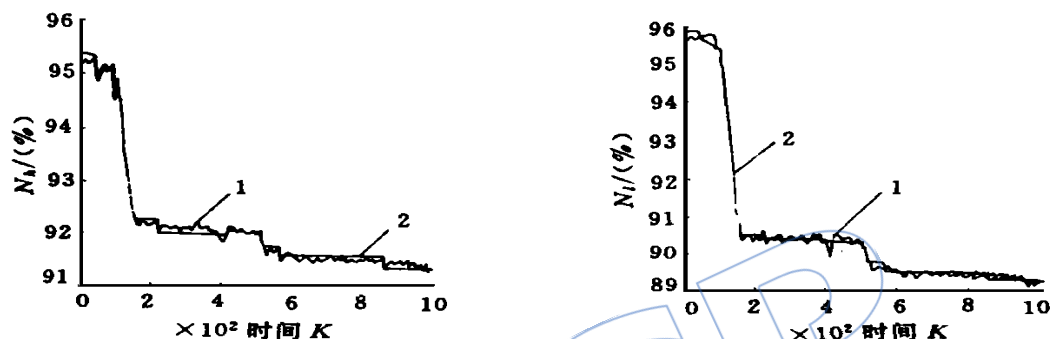


图 2 3 km 辨识模型输出与实测数

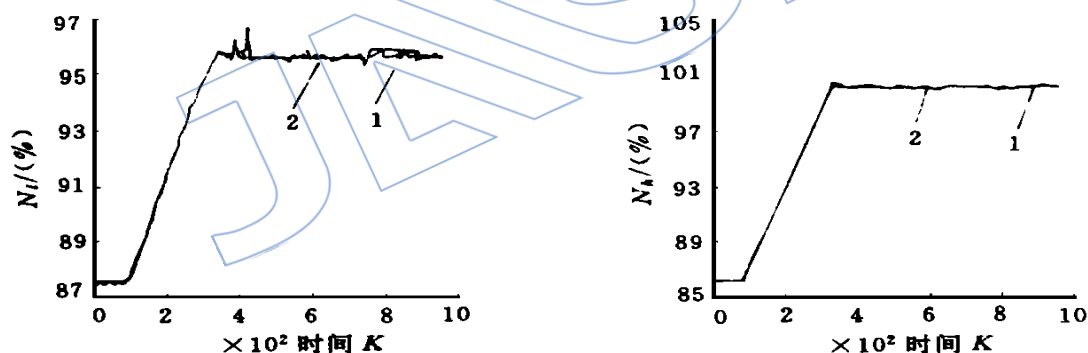


图 3 8.6 km 辨识模型输出与实测数

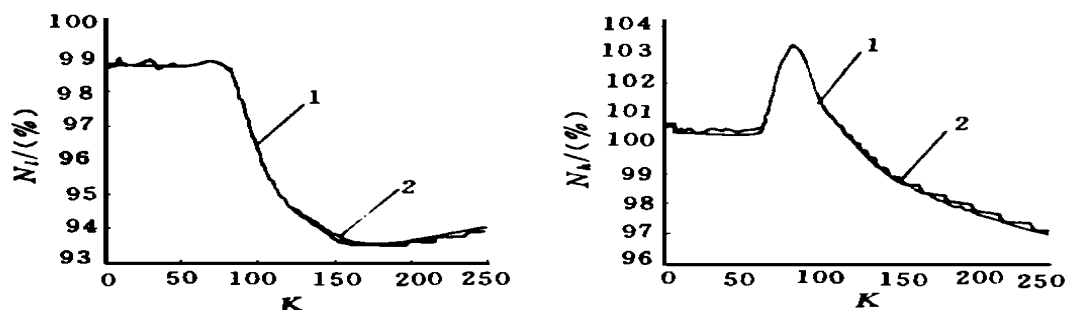


图 4 13 km 辨识模型输出与实测数

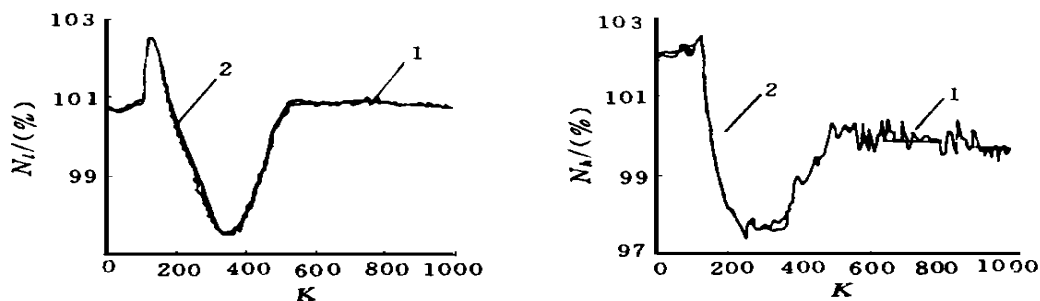


图5 18 km 辨识模型输出与实测数

算的辨识误差均小于 10^{-6} , 辨识网络输出值与实际值的相对误差为千分之一以下, 收敛时间为 18 ms。辨识模型输出的结果与飞行试验数据的比较如图 2- 图 5 所示, 图中曲线 1 为测量数据, 曲线 2 为辨识模型输出。仿真结果表明: 采用 BP 网络辨识发动机数学模型具有精度高、实时性较好等优点。这为航空发动机实现在线辨识和采用新的控制方法提供了重要途径。

参 考 文 献

- 1 Nguyen D H, Widrow B B. Neural Networks for Self- Learning Control Systems IEEE Control Systems Magazine, 1990: 18- 23
- 2 Chen F C. Back- Propagation Neural Networks for Nonlinear Self- Tuning Adaptive Control IEEE Control Systems Magazine 1990: 44- 48
- 3 焦李成. 神经网络的应用与实现. 西安电子科技大学出版社, 1993

(责任编辑 王庆康)