第九章作业

> 专业：计算机科学与技术
>
> 学号：17341178
>
> 姓名：薛伟豪

9-1

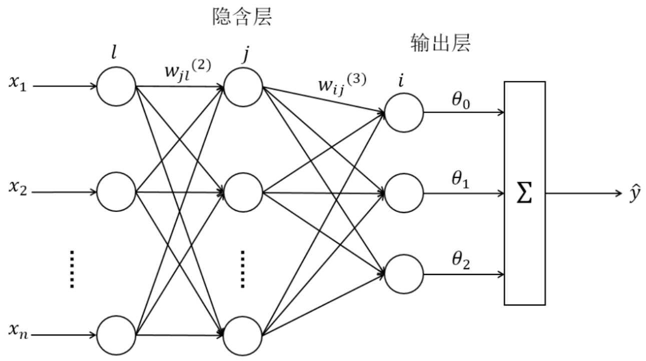
> **参照RBF网络直接模型参考自适应控制算法，试推导BP网络直接模型参考自适应控制算法。**

解：考虑一个如下单输入单输出二阶线性离散系统：

$$y(k)=\theta_0(y,u)*y(k-1)+\theta_1(y,u)*y(k-2)+\theta_2(y,u)*u(k-1)+\epsilon(k) \tag{1}$$

其中$y=[y(k-1),\ y(k-2),\ \ldots],\ \ u=[u(k-1),\ u(k-2),\ \ldots]$，$\epsilon(k)$为系统噪声。

由于参数$\theta=(\theta_0,\theta_1,\theta_2)^T$为系统状态的函数，所以系统实际是非线性的。为在线估计θ，建立一个三层BP神经网络模型，包括输入层、隐含层、输出层。结构如下：



由图可知，BP神经网络输入层节点的输出为：

$$O_l^{(1)}=x_l,\ l=1,2,\ldots,n \tag{2}$$

其中，n为输入层节点的个数，n取决于被控系统的复杂程度。

网络的隐含层输入、输出为：

$$net_j^{(2)}(k)=\sum_{l=1}^{n}w_{jl}^{(2)}(k)O_l^{(1)} \tag{3}$$

$$O_j^{(2)}(k)=f[net_j^{(2)}],\ j=1,\ldots,m \tag{4}$$

其中，$w_{jl}^{(2)}$为隐含层加权系数，$f[\cdot]$为激发函数，$f(x)=\frac{1-e^{-x}}{1+e^{-x}}$，$f'(x)=\frac{1}{2}[1-f^2(x)]$。上角标$(1)(2)(3)$分别对应输入层、隐含层、输出层。

网络的输出层的输入、输出为：

$$net_i^{(3)}(k)=\sum_{j=1}^{m}w_{ij}^{(3)}O_j^{(2)}(k) \tag{5}$$

$$O_i^{(3)}(k) = g[net_i^{(3)}(k)],\ i = 1,2,3 \tag{6}$$

$$O_0^{(3)}(k) = \theta_0,\ O_1^{(3)}(k) = \theta_1,\ O_2^{(3)}(k) = \theta_2 \tag{7}$$

其中，$w_{jl}^{(3)}$为输出层加权系数，$g[\cdot]$为激发函数，$g(x) = \frac{1}{1+e^{-x}}$，$g'(x) = g(x)[1-g(x)]$。

输出层的输出$(\theta_0, \theta_1, \theta_2)$分别是参数的估值$(\hat{\theta}_0(y,u), \hat{\theta}_1(y,u), \hat{\theta}_2(y,u))$，最后得到输出的估值为：

$$\hat{y}(k) = \hat{\theta}_0(y,u) * y(k-1) + \hat{\theta}_1(y,u) * y(k-2) + \hat{\theta}_2(y,u) * u(k-1) \tag{8}$$

取性能指标函数：

$$E(k) = \frac{1}{2}[y(k) - \hat{y}(k)]^2 \tag{9}$$

用梯度下降法修正网络的加权系数，即按E对加权系数的负梯度方向搜索调整，并附加一个使搜索快速收敛全局极小的惯性项，则有：

$$w_{ij}^{(3)}(k) = \gamma w_{ij}^{(3)}(k-1) - \rho \frac{\partial E(k)}{\partial w_{ij}^{(3)}} \tag{10}$$

式中，ρ为学习率，γ为惯性系数。

$$\frac{\partial E(k)}{\partial w_{ij}^{(3)}} = \frac{\partial E(k)}{\partial \hat{y}(k)} \cdot \frac{\partial \hat{y}(k)}{\partial O_i^{(3)}(k)} \cdot \frac{\partial O_i^{(3)}(k)}{\partial net_i^{(3)}(k)} \cdot \frac{\partial net_i^{(3)}(k)}{\partial w_{ij}^{(3)}} \tag{11}$$

由(7)和(8)可得：

$$\frac{\partial \hat{y}(k)}{\partial O_0^{(3)}(k)} = y(k-1),\ \ \frac{\partial \hat{y}(k)}{\partial O_1^{(3)}(k)} = y(k-2),\ \ \frac{\partial \hat{y}(k)}{\partial O_2^{(3)}(k)} = u(k-1) \tag{12}$$

这样，可以得到BP神经网络输出层权值计算公式为：

$$w_{ij}^{(3)}(k) = \gamma w_{ij}^{(3)}(k-1) - \rho \delta_i^{(3)} O_j^{(2)}(k) \tag{13}$$

$$\delta_i^{(3)} = -[y(k) - \hat{y}(k)] \frac{\partial \hat{y}(k)}{\partial O_i^{(3)}(k)} g[net_i^{(3)}(k)],\ i = 0,1,2 \tag{14}$$

同理可以得到隐含层权值计算公式为：

$$w_{jl}^{(2)}(k) = \gamma w_{jl}^{(2)}(k-1) - \rho \delta_j^{(2)} O_l^{(1)} \tag{15}$$

$$\delta_j^{(2)} = f'[net_j^{(2)}(k)] \sum_{i=1}^{3} \delta_i^{(3)} w_{ij}^{(3)}(k),\ j = 1,2,3,\ldots,m \tag{16}$$

根据性能指标$E(k) = \frac{1}{2}[y(k) - \hat{y}(k)]^2 < \epsilon$来判断系统辨识是否结束。通过上述过程来不断地修正输出层和隐含层的权值，使误差满足性能指标的要求。

9-2

> **参照RBF网络的自校正控制方法，设计基于RBF网络的模型参考自校正控制器，并进行Matlab仿真。被控对象为**$y(k) = 0.8sin(y(k-1)) + 15u(k-1)$**，采样周期为**$T = 0.001$**，参考模型为**$y_m(k) = 0.6y_m(k-1) + r(k)$**，**$r(k)$**为正弦信号，**$r(k) = 0.50sin(2\pi kT)$。

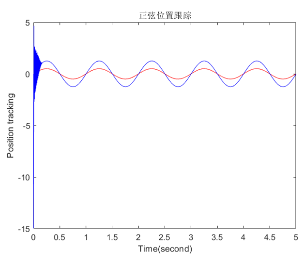
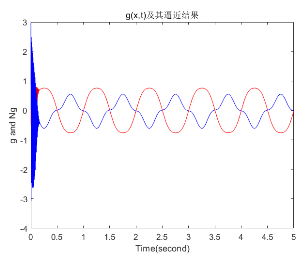
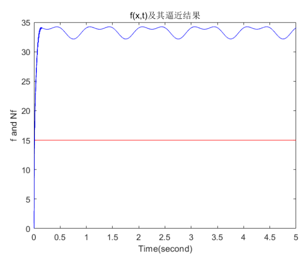
解：依题意，被控对象为$y(k) = 0.8sin(y(k-1)) + 15u(k-1)$，其中$g[y(k)] = 0.8sin(y(k-1))$，$\varphi[y(k)] = 15$。

网络隐层神经元个数$m=6$，初始权值取$W=[0.5\quad 0.5\quad 0.5\quad 0.5\quad 0.5\quad 0.5]^T$，$V=[0.5\quad 0.5\quad 0.5\quad 0.5\quad 0.5\quad 0.5]^T$，高斯函数的初始值取$C_j=[0.5\quad 0.5\quad 0.5\quad 0.5\quad 0.5\quad 0.5]^T$，$B=[5\quad 5\quad 5\quad 5\quad 5\quad 5]^T$，网络权值学习参数为$\eta_1=0.15$，$\eta_2=0.50$，$\alpha=0.05$。仿真程序代码如下：

```
%Self-Correct control based RBF Identification
clear all;
close all;

xite1 = 0.15;
xite2 = 0.50;
alfa = 0.05;
w = 0.5*ones(6,1);
v = 0.5*ones(6,1);
cij = 0.50*ones(1,6);
bj = 5*ones(6,1);
h = zeros(6,1);

w_1 = w;
w_2 = w_1;
v_1 = v;
v_2 = v_1;
u_1 = 0;
y_1 = 0;
ym_1 = 0;

 ts = 0.001;
 for k = 1:1:5000
    time(k) = k*ts;
    r(k) = 0.50*sin(2*pi*k*ts);

    %Practical Plant;
    g(k) = 0.8*sin(y_1);
    f(k) = 15;
    y(k) = g(k)+f(k)*u_1;
    for j = 1:1:6
       h(j) = exp(-norm(y(k)-cij(:,j))^2/(2*bj(j)*bj(j)));
    end
    Ng(k) = w'*h;
    Nf(k) = v'*h;
    ym(k) = 0.6*ym_1+r(k);
    e(k) = y(k)-ym(k);
    d_w = 0*w;
    for j = 1:1:6
       d_w(j) = xite1*e(k)*h(j);
    end
    w = w_1+d_w+alfa*(w_1-w_2);
    d_v = 0*v;
    for j = 1:1:6
       d_v(j) = xite2*e(k)*h(j)*u_1;
    end
    v = v_1+d_v+alfa*(v_1-v_2);
    u(k) = -Ng(k)/Nf(k)+r(k)/Nf(k);
    u_1 = u(k);
    y_1 = y(k);
    ym_1 = ym(k);
    w_2 = w_1;
```

```
    w_1 = w;
    v_2 = v_1;
    v_1 = v;
end

figure(1);
plot(time,r,'r',time,y,'b');
title('正弦位置跟踪');
xlabel('Time(second)');ylabel('Position tracking');
figure(2);
plot(time,g,'r',time,Ng,'b');
title('g(x,t)及其逼近结果');
xlabel('Time(second)');ylabel('g and Ng');
figure(3);
plot(time,f,'r',time,Nf,'b');
title('f(x,t)及其逼近结果');
xlabel('Time(second)');ylabel('f and Nf');
```

仿真结果如下所示：

正弦位置跟踪	$g(x,t)$及其逼近结果	$f(x,t)$及其逼近结果
		

9-3

> **已知一非线性系统**

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

> **给定的期望轨迹为**

$$y_d(k)=sin\frac{2\pi k}{25}+sin\frac{2\pi k}{10}$$

> **试采用RBF神经网络进行自适应控制，其中Jacobian信息由RBF网络辨识，并进行Matlab仿真。**

解：定义跟踪误差为$e(k)=y(k)-y_d(k)$。在$f(x(k))$已知的情况下，设计反馈显性化控制律为$u(k)=y_d(k+1)-f(x(k))-c_1e(k)$。取$c_1=-0.01$，仿真代码如下所示：

```
%Discrete controller
clear all;
close all;
ts = 0.001;
c1 = -0.01;
```

```
u_1 = 0;
y_1 = 0;
fx_1 = 0;

for k = 1:1:20000
    time(k) = k*ts;
    yd(k) = sin(2*pi*k*ts/25) + sin(2*pi*k*ts/10);
    yd1 = sin(2*pi*(k+1)*ts/25)+sin(2*pi*(k+1)*ts/10);
    %Nonlinear plant
    fx(k) = y_1/(1+y_1^2);
    y(k) = fx_1+u_1^3;
    e(k) = y(k)-yd(k);
    u(k) = yd1-fx(k)-c1*e(k);
    y_1 = y(k);
    u_1 = u(k);
    fx_1 = fx(k);
end

figure(1);
plot(time,yd,'r',time,y,'k:','linewidth',2);
title('位置跟踪');
xlabel('time(s)');ylabel('yd,y');
legend('Ideal position signal','Position tracking');
figure(2);
plot(time,u,'r','linewidth',2);
title('控制输入');
xlabel('time(s)');ylabel('Control input');
```

仿真结果如下所示：

位置跟踪	控制误差