

青霉素补料分批培养的人工神经网络模型

方柏山 胡宗定

(天津大学化工系 天津 300072)

摘要 用基于 Δ 规则和最速下降法的反向传播算法构建了一个能够超前1h预测青霉素补料分批培养状态变化的人工神经网络模型。分别考察了不同隐层层数及神经元数、不同步长、不同初始权矩阵及不同收敛准则对人工神经网络模型的影响。研究结果表明:当网络拓扑结构为5-3-5-3,步长为1、初始权值取0.98及收敛准则为 10^{-10} 时,所得到的模型能够很好地描述青霉素补料分批培养过程。

关键词 青霉素,补料分批培养,人工神经网络

既定产品(诸如青霉素类)的生产能力高低在商业上具有很强的竞争性。通过提高管理和控制水平即使稍微降低生产成本,在提高生产利润和保持市场优势方面都是至关重要的。适效的估算技术能够大大地提高管理和控制水平,因此,许多研究者致力于开发用于估算生化过程变量的技术。遗憾的是现行的许多算法因其基于过分简化的模型未能表达生化过程真实的复杂性和非线性而受到限制。一种可供选择的模型化技术——人工神经网络^[1],已成为一种能够有效地模拟和预测高度复杂过程的新技术。

本文正是借助于人工神经网络对青霉素补料分批培养过程进行模拟,以期建立一套能够超前1h预测该过程状态变量变化的神经网络模型,为实现过程的优化控制奠定基础。

1 青霉素补料分批培养动力学模型

为了使青霉素补料分批发酵时的产量最大,必须维持生长速率大于,然而又非常接近于预先确定的最小值。为此必须有规律地进行离线分析及状态预测,以便调节进料量使发酵操作沿着接近于该最佳生长速率的轨迹进行。

基于物料衡算和 Bajpai 及 Reuss^[2]所建立的以葡萄糖为底物补料分批培养生产青霉素 G 的模型,Levisauskas 和 Reuss^[3]提出了如下形式的动力学模型:

$$\frac{dX}{dt} = \mu_x \frac{SX}{K_x X + S} - \frac{X dV}{V dt} \quad (1)$$

$$\frac{dP}{dt} = \mu_p \frac{SX}{K_p + S(1 + S/K_i)} - K_P - \frac{P dV}{V dt} \quad (2)$$

$$-\frac{dS}{dt} = \frac{\mu_x}{Y_x/S} \frac{SX}{K_x X + S} + \frac{\mu_p}{Y_p/S} \frac{SX}{K_p + S(1 + S/K_i)} + m_X + \frac{S dV}{V dt} \quad (3)$$

本文于1995年9月25日收到。

式中,

$$\mu_x = 0.11/\text{h}^{-1}, \quad \mu_p = 0.0055/\text{h}^{-1}, \quad K_x = 0.0060, \quad K_p = 0.0001/\text{g} \cdot \text{L}^{-1}, \quad K_i = 0.1/\text{g} \cdot \text{L}^{-1}, \\ Y_{x/s} = 0.47 \quad Y_{p/s} = 1.2 \quad m = 0.029/\text{g} \cdot \text{g} \cdot \text{h}^{-1}, \quad K = 0.01/\text{h}^{-1}$$

体积 V 。分别为 1.29g/L、69.02g/L 和 8.12L 时,最佳补料时间为 $t = 28.8\text{h}$ 。此时,菌体浓度 $X = 30.09\text{g/L}$ 、底物浓度 $S = 0.00123\text{g/L}$ 和产物浓度 $P = 0.0195\text{g/L}$,按照所给定的最佳补料速率进行操作,发酵时间为 124h 时,最高青霉素产量为 8.57g/L。

2 人工神经网络模型

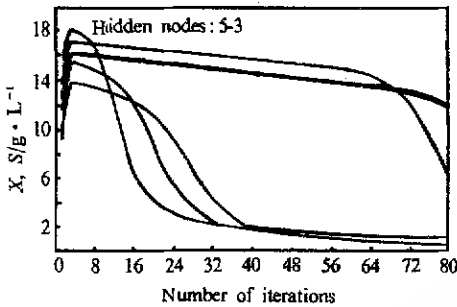
为了预测青霉素补料分批培养过程的状态变化,构建了不同拓扑结构的人工神经网络。按如上所述的 Levisauskas 和 Reuss 优化模型计算出 125 组不同时间下的限制性底物浓度 S 、菌体浓度 X 、产物浓度 P 、补料速率 U 和发酵液体积 V 。以 S_i, X_i, P_i, U_i 及 V_i

(i 为发酵时间)为输入变量, S_{i+1}, X_{i+1} 及 P_{i+1} 为输出变量,以所求出的 125 组数据为样本,对不同构型的人工神经网络进行训练。训练采用基于 Δ 规则和最速下降法的反向传播(back-propagation)算法⁴。传递函数为 Sigmoid 函数。为了便于比较不同条件及考察初始权矩阵对构建人工神经网络的影响,本文按下式设定初始权矩阵:

$$W_{ij} = 0.1 W_{1,1}^{(i-1)}$$

图 1 对于不同隐层神经元数,迭代 80 次离差平方和随迭代次数的变化

Fig. 1 Sum of squared errors in iteration vs number of iterations for different number of hidden nodes when $W_{1,1} = 0.9$, step = 0.4 training up to 80 iterations and $\Delta = 10^{-13}$



式中 i, j 为第 j 个神经元的第 i 个输入。权的调节由下式给出:

$$W_{ij}(t+1) = W_{ij}(t) + \eta \delta_j X_i$$

方程中 $W_{ij}(t)$ 为第 t 次由隐层或输入层神经元 i 到次层神经元 j 的权, X 是神经元 i 的输出或输入, η 是步长及 δ_j 是神经元 j 的修正项:

$$\delta_j = Y_j(1 - Y_j)(d_j - Y_j) \quad (j \text{ 为输出层神经元}),$$

$$\delta_j = X_j(1 - X_j) \sum \delta_k W_{jk} \quad (j \text{ 为任一隐层神经元}),$$

式中: d_j 和 Y_j 分别为输出层 d_j 个神经元输出的期望值和计算值, W_{jk} 为神经元 j 与上游层中第 k 个神经元的权值。收敛准则为离差平方和 SQ_{sum} 的相对误差:

$$\Delta = \left| 1 - \frac{SQ_{\text{sum}}(t)}{SQ_{\text{sum}}(t+1)} \right|,$$

其中 $SQ_{\text{sum}} = 0.5 \sum_{p=1}^p \sum_{j=1}^n (d_{pj} - Y_{yj})^2$,

式中: n 为输出层神经元总数, p 为训练样本数。

3 网络训练及结果

3.1 隐层数及神经元数的影响

取 $W = 0.9$ 、 $\eta = 0.4$ 及 $\Delta = 10^{-13}$, 用前述的 125 组数据分别训练含有一个隐层(其

神经元数分别为 2、6 和 10)及含有两个隐层(其神经元数分别为 5 和 3 及 3 和 5)的人工神经网络。所涉及的权分别为 21、57、93、60 和 56 个。结果如图 1 和表 1 所示:

表 1 对于不同隐层神经元数的训练结果

Table 1 The results of training for different number of hidden nodes

Hidden nodes	2	6	3-5	5-3	10
Number of weights	21	57	56	60	93
Relative speeds	1.00	2.29	2.27	2.43	3.57
$SQ_{sum} \times 100$	29.4	3.23	3.04	4.78	3.15
Number of iterations	21443	64641	45127	42743	62994
Relative time	1.00	6.90	5.12	5.13	10.5

Note: $W_{1,1} = 0.9$, Step = 0.4 and $SQ_{sum} \Delta = 10^{-13}$

神经元数分别为 3 和 5 的人工神经网络进行训练,循环迭代 80 次时其 SQ_{sum} 最大。但当继续迭代至 $\Delta = 10^{-13}$ 时其 SQ_{sum} 最小,而且耗时最短(除了一个隐层的神经元数为 2 外)。对表 1 中相对迭代速度 V 与网络的总权数 N 进行线性回归得:

$$V = 0.257 + 0.0356N \quad (r_{xy} = 0.9997)$$

这表明迭代速度与网络的总权数成正比,而与网络的结构无关。

3.2 步长 η 的影响

取 $W = 0.9$ 、 $\Delta = 10^{-13}$ 及 3.1.1 中所得到的最佳网络结构 5-3-5-3,用同样的数据不同的步长 ($\eta = 0.2, 0.4, 0.6, 1$ 和 2)进行训练,所得结果如表 2 所示:它表明当 $\eta = 1$ 时所获得结果最好。这说明 McClelland 和 Rumelhart^[1]所建议的 $\eta \leq 0.5$ 并非普遍适用。

表 2 学习步长对训练结果的影响

Table 2 Effect of the learning steps on the results of training

Learning steps	0.2	0.4	0.6	1.0	2.0
$SQ_{sum} \times 1000$	4.56	3.04	2.84	0.728	1.93
Number of iterations	26501	45127	36938	96595	26946

Note: network structure 5-3-5-3, $W_{11} = 0.9$ and $\Delta = 10^{-13}$

3.3 初始权矩阵及收敛准则的影响

顾及到初始权矩阵对构造人工神经网络的影响,绝大多数研究者都是以不同套随机产生的权矩阵作为初始权矩阵进行训练。这种做法虽可避免主观性,但有盲目性及难以比较之嫌。为了便于比较初始权矩阵的影响,在上述研究的基础上分别取 $W_{1,1} = 0.9 \pm 0.08$ 进行训练。此外在同样的 $W_{1,1}$ 下取不同的 $\Delta (= 10^{-10}$ 和 10^{-16} , 进行训练,其结果如表 3 所示:它表明在同样的 Δ 下, $W_{1,1}$ 稍有变化, SQ_{sum} 也会随着 $W_{1,1}$ 值的变化而变化。虽然在同样的 $W_{1,1}$ 下,当 $\Delta = 10^{-10}$ 和 10^{-16} 时 SQ_{sum} 值相同。但这并不意味着 SQ_{sum} 与 Δ 取值无关,如当 $\Delta = 10^{-13}$ 时其 SQ_{sum} 值则不同。

表 3 初始权和 Δ 对训练结果的影响

Table 3 Effect of the initial weight and Δ on the results of training

Δ	10^{-10}			10^{-13}			10^{-16}		
W_{11}	0.82	0.90	0.98	0.90	0.82	0.90	0.98	0.90	0.98
$SQ_{sum} \times 1000$	6.79	6.76	6.55	7.28	6.79	6.76	6.55	6.79	6.76
Number of iterations	98600	102081	97374	96595	98600	102081	97374	98600	102081

Note: network structure 5-3-5-3, and step = 1.0

3.4 训练结果

综上所述,取网络结构 5-3-5-3、 $\eta = 1$ 、 $W_{1,1} = 0.98$ 及 $\Delta = 10^{-10}$ 进行训练所得到的结果为好。检验该结果显示:其预测值与期望值非常一致(见图 2)。它表明用所建立的模型能够很好地提前一个小时预测青霉素补料分批发酵过程的状态变量。

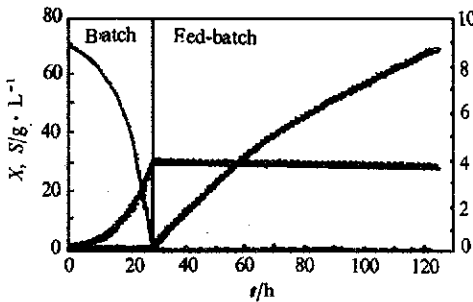


图 2 用于青霉素补料分批培养的神经网络估算
Fig.2 Neural network estimations for
Penicillin fermentation
in fed-batch

局部极值点的可能。

4 讨 论

人工神经网络能够很好地运用于预测青霉素补料分批发酵过程的状态变量,这对于实现该过程的优化控制具有重要意义。输入、输出变量确定后,神经网络优劣受到诸如隐层数、隐层神经元数、初始权的设置、学习步长及收敛程度等因素的影响。避免陷入误差函数的局部极值点是建立人工神经网络成功与否的关键。在训练网络时,有必要全面考虑这些因素的综合影响。随机产生的权矩阵作为初始权矩阵进行训练,虽可避免主观性,但有盲目性之嫌及随机地陷入误差函数

参 考 文 献

- [1]McClelland J L, Rumelhart D E. Explorations in Parallel Distributed Processing: A Handbook of Models, Programs, and Exercises, Cambridge MIT Press, 1986
- [2]Bajaj R K, Reuss M. J Chem Tech Biotech, 1980, (30):332
- [3]Levisauskas D, Reuss M. DECHEMA Biotechnology Conferences, Karlsruhe, Germany, June 1-3, 1992.
- [4]Richard P. Lippmann IEEE ASSP Magazine April, 1987, 4~22.

Artificial Neural Network Model for Penicillin Fermentation in Fed-batch

Fang Baishan Fu Zhongding

(Department of Chemical and Biochemical Engineering,
Tianjin University, Tianjin 300072)

Abstract A neural network model was constructed by using a back propagation algorithm on the delta rule and gradient descent, which could predict the state variables of penicillin fermentation in fed-batch one hour ahead. The influences of the numbers of hidden layer and hidden nodes, learning step, initial weights and convergence rule on the neural network model were investigated respectively. The research results shown that: the model obtained could well describe the process of penicillin fermentation in fed-batch, as the network architecture is 5-3-5-3 topology, learning step is 1, initial weight is 0.98 and convergence rule is 10^{-10} .

Key words Penicillin, fed-batch, artificial neural network