Bioautomation, 2004, vol. 1, pp. 57-66

Optimal Control of a Fed-batch Fermentation Process by Neuro-Dynamic Programming

Tatiana Ilkova, Stoyan Tzonkov

Centre of Biomedical Engineering, Bulgarian Academy of Sciences
105, Acad. G. Bonchev Str., Sofia 1113, Bulgaria

E-mail: tanja@clbme.bas.bg
tzonkov@clbme.bas.bg

Abstract: *In this paper the method for optimal control of a fermentation process is presented, that is based on an approach for optimal control-Neuro-dynamic programming. For this aim the approximation neural network is developed and the decision of the optimization problem is improved by an iteration mode founded on the Bellman equation. With this approach computing time and procedure are decreased and quality of the biomass at the end of the process is increased.*
Keywords: *Fermentation process, Neuro-dynamic programming, Optimal control.*

Introduction

The optimal control of the Fermentations Processes (FP) usually depends on the presence of complex, non-linear dynamic model of the system because of this is difficult to realize to working-out of the problem, which is very important to practical realize [7,9]. The problems of the optimization difficulties arise even in offline optimal control, especially when a model is a high dimensional, one with larger diapason of the investigations, lager composition of the variables also of technological and optimizations' restrictions [10].
One approach for solving to -problem of dynamic optimization is Dynamic Programming (DP), and is successfully applied to fermentation processes [3,4,8]. However, the approach is largely considered impractical as analytical solution of resulting dynamic program is seldom possible and numerical solution suffers from the "curse of dimensionally" [3,4].

Neuro-Dynamic Programming (NDP) is suggested as a method to alleviation the "curse of dimensionally". The name neuro-dynamic programming expresses the reliance of the methods of this article on both DP and neural network (NN) concepts [2]. The method is successfully

applied for optimal control of batch's and continuous FP in the last years, as the computing time was decreased about 2/3 and increases of the quantity of the desired products was gotten [7,8].

In this paper an optimal control of a fed-batch FP process by NDP is presented and compared it with the DP optimal control.

Formulation of the optimization problem

The problem of dynamic optimization includes minimization of the following expression [8]:

$$\min_{u_0,\ldots,u_{p-1}} \sum_{i=1}^{p-1} \phi(x_i,u_i) + \bar{\phi}(x_p)\,(1), \qquad (1)$$

with constraints: $g(x_i,u_i) \geq 0.0$;

$0 \leq i \leq p-1$, $\bar{g}(x_p) \geq 0$;

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

for a given initial state x_0 and a function constant input $u(i) = u_i \; i \cdot h \leq \tau < (i+1) \cdot h$, h is the sampling time, x_i-represent the value of x at the stage i^{th} (т.е. $x(t)$ в $t=h.i$), ϕ-single state cost function and $\bar{\phi}$ is the terminal state cost function and whey are defined on the multitude of the real numbers, and f is continuous and differentiated function.

Bellman Equation

The DP involves stage-wise calculation of the *cost-to-go* function to arrive at the solution, not just for a specific x_0 but for general x_0. For (1), the *cost-to-go* at each stage is defined as [3]:

$$J_i = \min_{u_{p-i},\ldots,u_{p-1}} \sum_{j=p-i}^{p-1} \phi(x_j,u_j) + \bar{\phi}(x_p). \qquad (2)$$

and the next step:

$$J_{i+1} = \min_{u_{p-i-1},\ldots,u_p} \sum_{j=p-i-1}^{p} \phi(x_j,u_j) + \bar{\phi}(x_p).$$

If we mark $J_i = L \cdot (F_h(x,u))$, where $F_h(x,u)$ is the resulting state after integrating the differential equation for one sample interval with the starting state of x and constant input of u, L is continuous and differentiated function, then:

$$J_{i+1} = \min_{u_{p-i-1},\ldots,u_p} \sum_{j=p-i-1}^{p} \phi(x_j,u_j) + L \cdot (F_h(x,u)) \qquad (3)$$

In conformity with the "principle of the optimum" of Bellman [1] "*The tail policy is optimal for the tail subproblem"*, at each stage the calculation of the *cost-to-go* function can be done as:

$$J_i(x) = \min_u \phi(x,u) + J_{i-1}(F_h(x,u)), \quad (4)$$

Equation (4) is calculated from i=1 to i=p

Such as, the pertinent terminal needs to be imposed at each stage. Once received, the *cost-to-go* function has optimal decision for a general state x_0 according to "principle of the optimum" [1].

The objective of DP is to calculate numerically the optimal cost function J. This computation can be done off-line, i.e., before the real system starts operating. In very few cases can we solve the stage-wise optimization analytically to obtain a closed-form expression for the *cost-to-go* function. An optimal policy, that is, an optimal choice of u for each i, is computed either simultaneously with J, or in real time by minimizing in the right-hand side of Bellman's equation. It is well known, however, that for many important problems the computational requirements of DP are overwhelming, mainly because of a very large number of states and controls. In such situations a suboptimal solution is required.

Cost-to-go Approximation

The traditional approach to solving the Bellman equation involves gridding of the state space, solving the optimization (4) for each grid point, and performing the stage-wise optimization until convergence. Exhaustive sampling of state space can be avoided by identifying relevant regions of the space through simulation under judiciously chosen suboptimal policies. The *policy improvement theorem* states that a new policy that is greedy with respect to the *cost-to-go* function of the original policy is as good as or better than the original policy (the now policy is given when moment value of the *cost-to-go* function is least, i.e. the new policy is defined by expression:

$$u(x) = \arg \min_u \phi(x,u) + J_i(F_h(x,u))),$$

where arg $G(u,x,i)$ is arguments of the function$\in R^{m+n+r}$, and $u \in R^m$, $x \in R^n$ and $i \in R^r$, and it is

improvement of the original policy. When the new policy is as good as the original policy, the above equation becomes the same as Bellman optimality equation (4). When the iterations

converge, this off line-computed *cost-to-go* approximation optimal control calculation for the bioreactor.

NDP Algorithm

NDP method is suboptimal methods that center around the approximate evaluation of the optimal cost function *J*, possibly through the use of neural networks and/or simulation [2]. For description of the algorithm the next symbol is used: *J*-represents *cost-to-go* values, $\tilde{J}(x)$-a function approximation relating *J* to corresponding state *x*, $()^i$ iteration index for cost iteration loop, *k*-discrete time. Then it can be written: $\tilde{J}(k) \equiv \tilde{J}(x(k))$.

If the system starts with a given policy (some rule for choosing a decision u at each possible state *i*), and its approximately evaluate the cost of that policy (as a function of the current state) by least-squares-fitting a scoring function $\tilde{J}(k)$ to the results of many simulated system trajectories using that policy and simulation of the process with chosen suboptimal policies under all representative operating conditions. The *cost-to-go* function is calculated using the simulation data for each state visited during the simulation, as for each closed loop simulation yields us data *x*(0), *x*(1), …, *x*(*N*), where *N* is sufficiently large for the system to reach equilibrium, calculated of one-stage cost $\phi(k)$ for these points. Then *cost-to-go* is the sum of single stage costs from the next point - $J(k) = \sum_{i=k+1}^{N} \phi(i)$. This is step simulation. After we preparing of a NN to the data approximate the *cost-to-go* function – denoted as $\tilde{J}^0(x)$, as a smooth function of the states. The NN is chosen with dependent on number of system equation, Bellman's equation and time. A new policy is then defined by minimization in Bellman's equation, where the optimal cost is replaced by the calculated scoring function, and the process is repeated. To improve the approximation, perform the following iteration (referred to as the cost iteration) until convergence:

- With the current *cost-to-go* approximation $\tilde{J}^i(x)$ is calculated $\tilde{J}^{i+1}(k)$ for the given points of *x* by following equation:

$$\tilde{J}^{i+1} = \min_u \phi(x,u) + \tilde{J}^i(F_h(x,u)), \tag{5}$$

which is based the Bellman equation. This step is approximation.

Update of the policy. It may necessary to increase the coverage of the state space i.e. more suboptimal simulations with the updated policy with suboptimal policy for improvement of

cost-to-go approximation are used to increase the coverage or the number of data points in certain region of state space.

Application of the NDP for optimal control of a fermentation process

The process of a fed-batch fermentation of *E. Coli* is examined in this work. The fermentation is leaded in a bioreactor with mixing in mesophyle regime. The parameters of the process are showed in Table 1 [6, 7].

Table 1 Parameters of the process

Parameter	Value
Temperature	35°C
pH	6,8
Gassing rate	275 1/h
Oxygen around	35 %
Glucose in Batch	2,5 g/L
Volume	1,5 g/L
Glucose in Feeding Solution	100 g/L
Stirrer speed at start	900 rpm
Stirrer speed at end	1500 rpm

The Model of the Process

The program was made for parametric identification and estimate of the parameter of the model on the basis of non-linear regression. On the basis of this program the following kinetic model was received [5]:

$$\frac{dX}{dt} = \mu.X - \frac{F}{V} \tag{6}$$

$$\frac{dS}{dt} = \frac{F}{V}(S_0 - S) - \eta.X \tag{7}$$

$$\frac{dV}{dt} = F\ , \tag{8}$$

where: $\mu = \mu_{\max} \cdot \frac{S}{k_S + S}$; $\eta = \frac{\mu}{Y} + k_m$;

and X, S and V are continuous and differentiated functions, η- specific consummation rate of substrate, 1/h; μ- specific grown rate, 1/h; μ_{max}- maximum specific grown rate, 1/h; V- volume, L; Y- yield coefficient, %; F- feeding rate, L/h; k_s-Monod's saturation constant for substrate, 1/L; S_0-initial substratum concentration, g/L; X- concentration of biomass, g/L; S- concentration of substrate, g/L; k_m-inhibition constant, -.

The initial conditions are:
$S(0)$=2,6 g/L, $X(0)$=0,11657 g/L, $V(0)$=1 L.
The parameters of the model have the next values [5]:
k_m=-0,6015; μ_{max}=0,4671; k_S=0,0742; Y=0,4843.

Formulation of the Problem Optimization

As is well-known for fermentation process, relatively little a change in the speed of feed can take process to switch over toward undesired stability state (especially steeply disturbance in F). The control objective is, therefore, to drive the reactor from the low biomass steady state to the desirable high biomass yield state. It may be viewed as a step change in the setpoint at time t = 0 from the low biomass to the high biomass yield steady state.

Working-out of the optimization problem:

Simulation with Using of Suboptimal Control

Four values of F=[0,0035,…,0,016] was examined, that can cover the possible rang of variation. For each of the parameter values, the reactor was started at three different $x(0)$ values around the low biomass yield steady state. We obtained 100 data points for each run. Thus a total of 1200 data points were obtained.

Approximation

A functional approximation relating *cost-to-go* with augmented state was obtained by using neural network with five hidden nodes, six input and one output node. The neural network showed a good fit with mean square error of 10^{-3} after training for 1000 epochs.

Improvement Through Bellman Iterations

Improvement to the *cost-to-go* function is obtained through iterations of the Bellman equation (4). This method, known as cost iteration (or value iteration), is described in section 2.4. The

solution of the one-stage-ahead cost plus *cost-to-go* problem, results in improvements in the cost values. The improved costs were again fitted to a neural network, as described above, to obtain subsequent iterations $\tilde{J}^1(x)$, $\tilde{J}^2(x)$, and so on …, until convergence. Cost was said to be "converged" if the sum of absolute error was less than 5% of the maximum cost. The cost converged in 4 iterations for our system.

Results

The results are shown in Fig.1 and numerical comparison in Table 2. The method was tested for various *F*. Representative results for a single *F* value of 0,146 are shown.

First two rows in the Table 2 represent the online performance of the two approaches, DP and NDP. In the Table 2, the last two columns show the comparison between the two schemes; the first four columns represent the control algorithm, the number of data points used in obtaining *cost-to-go* function, the number of cost-iterations and the number of hidden nodes in the neural network approximation of converged cost function.

Table 2 The result of the optimization

Method	Number of data points	Cost Iterations	Number hidden nodes	Total cost (at $x(0)$)	Time, s
DP	1200	6	N. A.	26,7	1067,0
NDP	1200	4	5	24,03	130,0

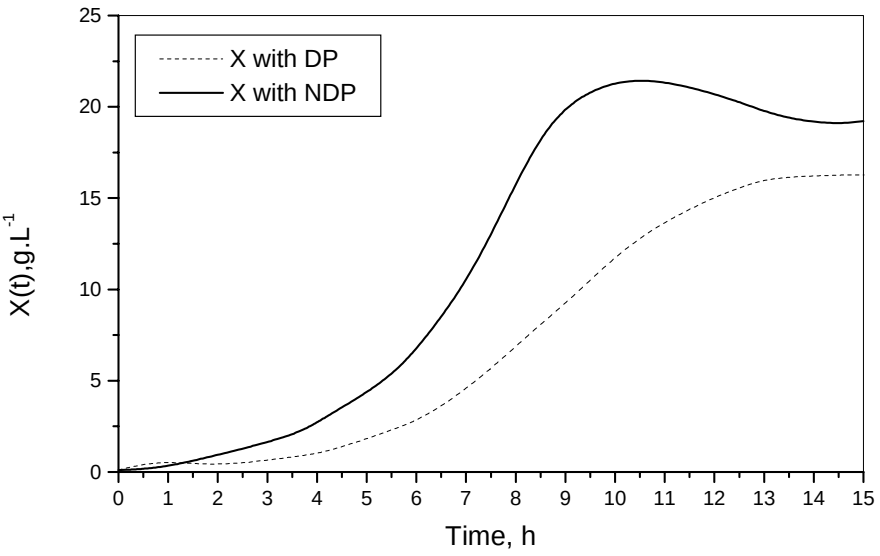

Fig.1 Concentration of the biomass, optimized with DP and NDP

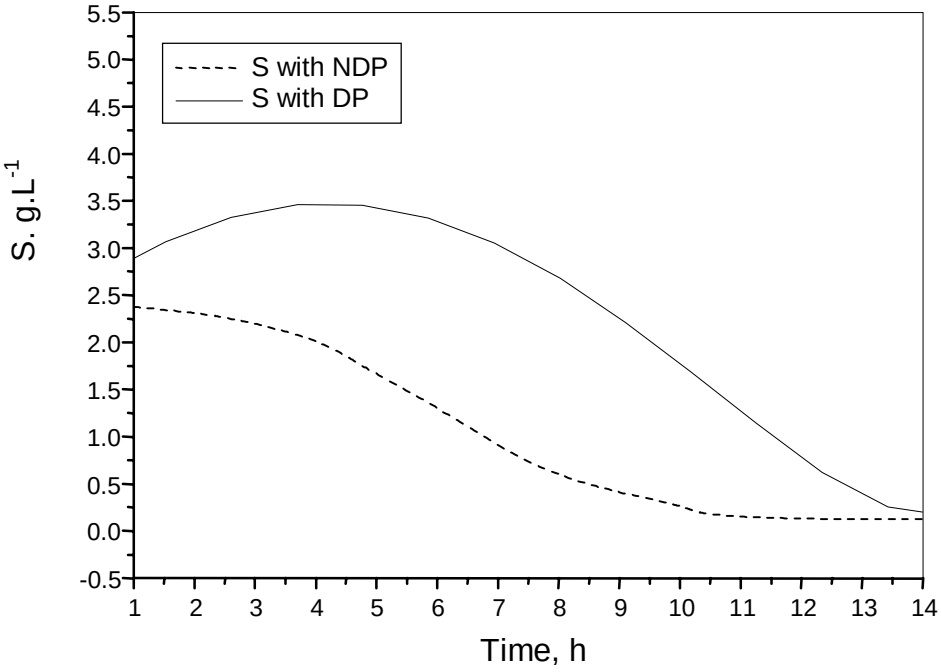

Fig. 2 Concentration of the substrate, optimized with DP and NDP

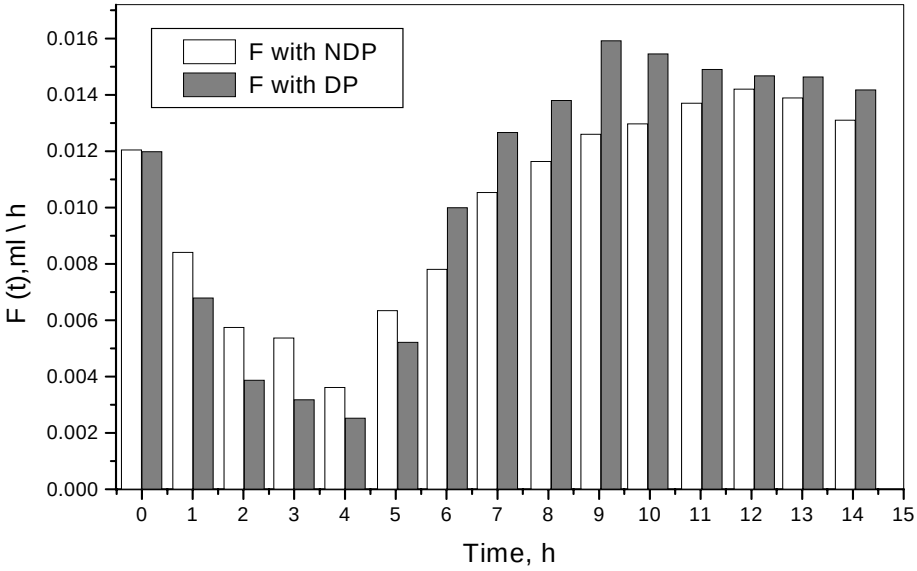

Fig. 3 Feeding rate received with DP and NDP

Discussion

From Fig.1, Fig.2 and Table 2. it is noticed:

1. Great increase of quantity of the biomass at the end of the process is reached with using the method of NDP, comparison with method of the DP. When NDP is applied the biomass quality is 32.19%, when DP is applied the biomass quality is 24,19%.
2. In Table 2 it shows that with NDP operation are decreased time for optimal control, comparison with method of the DP. When NDP is applied the optimization procedure finishes for 130,0 s, when DP is applied the optimization procedure finishes for 1067,0 s.
3. From Fig.2 it shows that the using quality feeding optimized with NDP is less than using quality feeding optimized with DP. In this way the substrate is better utilized, that decrease production cost price.
4. From Table 2 and Fig.2 it shows that with using NDP the process should be stopped at 10.30 hour and with that the process economical effectiveness is raised.

Conclusion

The special features and the characteristics of the FP essence make difficult its optimal control. In this paper a method for optimal control of FP is presented, that is based on the contemporary and effective approach for optimal control-NDP. The NDP expresses the reliance of the methods of this article on both DP and neural network concepts and it was proposed such as one methodic for alleviation of “curse of dimensionally”. For this aim the approximation neural network was developed and the initial approximation *cost-to-go* function is further improved by an iteration mode founded on the Bellman equation with that decreased computing time, increased quality of the product at the end of the process and decreased substrate utilization with comparison of DP method.

Acknowledgments:

The investigations are partially supported by National Science Fund by grant N^0*TH1314/2003.*

References:

1. Bellman R. (1965). Dynamic programming. New Jersey, Princeton University Press.
2. Bertsekas D., J. Tsitsiklis (1996). Neuro-dynamic programming. Belmont-MA, Athena Scientific.
3. Bertsekas D. (2000). Dynamic programming and optimal control, Vol. I, 2nd Edition, Belmont-MA, Athena Scientific.
4. Bertsekas D. (1995). Dynamic programming and optimal control, Vol. II, Belmont-MA, Athena Scientific.
5. Ilkova T., M. Petrov (2002). Modelling and Fuzzy Optimization of Fed-batch Fermentation Process. Chem. Biochem. Eng. Q.,16, No 4, pp. 173-178.
6. Hitzmann H., J. Brandt (1994). Knowledge-Based Fault Detection and diagnosis in flow-injection analysis, Analytical Chim. Acta, 29, pp. 29-40.
7. Kaisare N. , J. Lee, J. H. Lee (2002). Simulation Based Optimization Nonlinear Optimal Control: Application to a Microbial Cell Reactor. - IFAC World Congress, Barcelona, Spain, 17 July, pp.345-369.
8. Lee J., B. Cooley (1997). Recent Advances in Model Predictive Control. - Chemical Process Control, V, pp. 201–216b.
9. Petrov M. (1998). Fuzzy Optimization of Mass Transfer in Stirred Tank Bioreactors, Biotechnol& Biotechnoll Eq., 12, N 1, pp. 120-124.
10. Tzonkov St., S. Stoyanov, Tz. Georgiev (1994). Bases of the Automation Control of Biotechnological Processes, TU-Sofia. (in Bulgarian).