17th European Symposium on Computer Aided Process Engineering – ESCAPE17
V. Plesu and P.S. Agachi (Editors)

Adaptive Controller Tunning by Kalman Filter for Advanced application to a Fermentation Process

Elis R. Duarte,[a] Laércio Ender,[b] Daniel I. P. Atala,[a] Rubens Maciel Filho.[a]

[a]*State University of Campinas, 13.083-970, Campinas, SP, Brazil, eduarte@feq.unicamp.br*
[b]*Regional University of Blumenau,Campus II, 89010-971, Blumenau , SC, Brazil, ender@furb.br*

Abstract

The present work proposes a control parameter adjustment technique using Kalman filter applied to an adaptive control strategy. The control algorithm is based on neural networks with on-line learning to compute the next action of the manipulated process variables. The penalization parameters of the control actions are on-line optimised by the Kalman filter. The control strategy was tested on an extractive alcoholic fermentation process and the results showed the great potential for successful application.

Keywords: Control Strategy, Kalman Filter, Fermentation Process.

1. Introduction

The control of biotechnological processes is not an easy task mainly due the complex nature of the microbial metabolism, as well as to the non-linearity of its kinetics, consequently this process required a control method by which the process objectives can be achieved reality and quickly.

Most process control applications consist on not only keeping controlled variables at their set points but also keeping the process from violating operational constraints. The former is particularly important when there are changes in set points and the processes are multivariable and non-linear. While good advances have been obtained using conventional adaptive controller algorithms, when high performance operation is required, more sophisticated controllers are required. In many industrial systems, they're strong interactions among the process variables as well as internal changes. To cope with the former changes algorithm self-tuning capabilities is needed.

The artificial neural networks (ANN´s) are instuments very interesting because the nets approach allows taking into account process nonlinearities as well as variable interactions. Extensively employed in control process [1-5].

In fact, most of the advanced control algorithms have controller parameters set-up off-line and the computer plays a marginal task in terms of real time controller parameters identification. Taking this into account, this work presents a control strategy based on neural networks with on-line and real time learning of the nets and parameter adjustment using Kalman filter.

An adaptive algorithm with parameter adjustment using Kalman filter in working in a real time basis is developed and implemented.

2. Control Strategy

For the development of the present work feedforward architecture with backpropagation learning was used. In all the neural networks one hidden layer was considered. Historical input-output data obtained by simulation were used to off-line training a dynamic neural networks model and an inverse process dynamic model. The dynamic network is trained to represent the forward process dynamics. There is a time window for the neural network to forget past value. The inputs of the network are the current and past values of the controlled and manipulated variables and the outputs of the network are the one step ahead prediction of the process outputs, in according with equation (1)

$$Y(k+1) = f\big(Y(k),\ldots,Y(k-n+1),U(k),\ldots,U(k-m+1)\big) \tag{1}$$

The inverse process dynamics acts as a controller of the strategy. The inputs are the setpoints of the closed loop for the next sampling time; past controlled and manipulated variables and the outputs of the neural network are the manipulated variables for the next sampling instant.

The weights of the neural network of the controller are adjusts in such way to minimize the estimated global error ($Yr - \hat{Y}$). Considering that the estimated error is based on a neural model, it is necessary to have a model that represents with fidelity the dynamic behaviour of the process. When the quadratic error of the neural model outputs is smaller than the desired tolerance, this model is used in the optimization routine. If the quadratic error becomes larger in relation to a determined error, the controller makes use of the standard weight (weight of the off-line learning) to generate the control action for this sampling instant.

Initially, the neural networks of the controller and the process model were off-line trained with input-output response data from the process. This initial training ensures that the neural controller will be able to provide relatively accurate control output signals and process output response.

To guarantee the good dynamic representation of the process through neural networks, a strategy formed by three neural networks representing the dynamic behaviors acting parallelly was adopted. The first is formed by weights of the off-line learning, here denominated of standard weights; the second, is initialized with the standard weights and it is submitted at the on-line learning. Whenever the standard weights present better performance, this net has its weights substituted by the standard weights. The third is initialized with the standard weights and continually is submitted to the on-line learning at each sampling time. The neural network that presents the smallest quadratic error in the representation of the vector that contains the last inputs/outputs of the process is used in the control strategy in this sampling time [6].

2.1. Methodology

A first order filter was used in the reference of the process (setpoint) and a penalization in the control action was used. The objective of this work is to adjust the penalization parameters in the filter of the control actions and a robust behavior of the process output is expected. The penalization of the control actions is given by the equation (2):

$$u(i,t) = \lambda_i \; u(i,t-1) + (1-\lambda_i)\; u_{ANN}(i,t) \tag{2}$$

where:
u(i,t): control action i, applied to the process in the time t;
u_{ANN}(i,t): control action i, obtained by the neural controller in the time t;
λ_i: penalization parameter for the manipulated variable i estimated by Kalman Filter ($0 \leq \lambda_i < 1$).
The adjustment or tuning algorithm of the parameter λ is based on the standard Kalman filter. To be able to adjust λ a dynamical system has to be created which can observe the state of the parameter λ as in:

$$\begin{aligned} \lambda_{i,k+1} &= \lambda_{i,k}\left(+\, w_{i,k}\right) \\ z_{i,k} &= C\; \lambda_{i,k}\left(+\, v_{i,k}\right) \end{aligned} \tag{3}$$

where $w_{i,k}$ and $v_{i,k}$ are random variables with a normal distribution of N(0,Q) and N(0,R) respectively. $Z_{i,k}$ is the measurement related to the state $\lambda_{i,k}$. Normally the noise of a parameter state is zero, but a small process noise results in a more stable filter.
The observation equation of the λ is based on the past and not on the present data. Thus the algorithm changes λ in a feedback way. It would be desirable that the penalization parameters are raised, when the process is changing rapidly. This assures that the process will not show oscillatory behaviour. It is used an intuitive approach, which is based on the definition of the relative gain, which results in the following identification system.

$$\begin{aligned}\lambda_{1,k+1} &= \lambda_{1,k}\left(+w_k\right)\\ \lambda_{2,k+1} &= \lambda_{2,k}\left(+w_k\right)\\ \lambda_{3,k+1} &= \lambda_{3,k}\left(+w_k\right)\end{aligned} \tag{4}$$

$$\phi\begin{bmatrix} abs\left(y_{1,k}-y_{1,k-1}\right)\\ abs\left(y_{2,k}-y_{2,k-1}\right)\\ abs\left(y_{3,k}-y_{3,k-1}\right)\end{bmatrix} = \begin{bmatrix} J_{11} & J_{12} & J_{13}\\ J_{21} & J_{22} & J_{23}\\ J_{31} & J_{32} & J_{33}\end{bmatrix}\begin{bmatrix} abs\left(u_{1,k}-u_{1,k-1}\right) & 0 & 0\\ 0 & abs\left(u_{2,k}-u_{2,k-1}\right) & 0\\ 0 & 0 & abs\left(u_{3,k}-u_{3,k-1}\right)\end{bmatrix}\begin{bmatrix}\lambda_{1,k}\\ \lambda_{2,k}\\ \lambda_{3,k}\end{bmatrix} \tag{5}$$

where J_{ij}*'s* are the coefficients of the jacobian matrix of the input/ouput variables and ϕ is a velocity factor. The jacobian matrix was calculated at each sample time and ϕ is a project parameter.

The absolute value is taken as the penalization parameter has a positive value. In case of the derivatives of the responses are large it is necessary to put a velocity factor, ϕ. Small value of the parameter results in a reduction of the penalization parameter λ. The velocity factor is a parameter which still has be tuned manually. But the number of parameters to be tuned will be reduced.

2.2. Case study

As a study case its was used a 3x3 multivariable process, represented by an alcoholic fermentation plant, proposed by Silva *et al.* [7], where the manipulated variables were the feed stream flow rate (Feed), cells recycle rate (R), flash recycle rate (r). The control variables were: product concentration (P), substrate concentration (S) and cell concentration (X).

2.3. Results & discussions

The velocity factor is a parameter which still has be tuned manually, while it is wanted to eliminate the tuning of the suppression factor. For this case the better velocity factor was 0.2. While the number of parameters to be tuned for MIMO system is reduced in this way, it is still not the ideal case where no parameter has to be tuned. Therefore it has to be continued the search for other ways to identify the suppression factors, which results in no introduction of another parameter.

The servo problem as well as the regulator problem was undertaken. The regulator problem was caused by the change in the feeds concentrations of substrate (So) and temperature (To) (Figure 1 and 2). In the servo case four perturbations are done according Figure3.

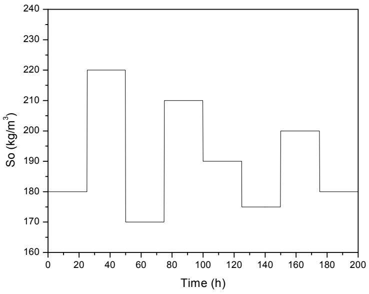


Figure 1. Regulator Problem (So).

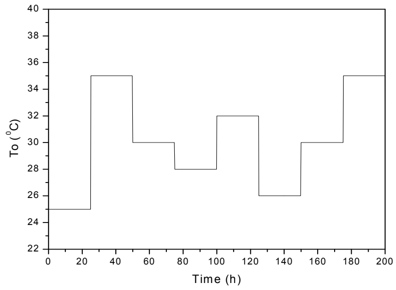


Figure 2. Regulator Problem (To).

It is presented in Figure 3 the control of the product concentration and Figure 4 presented the control of substrate and microorganism concentration.

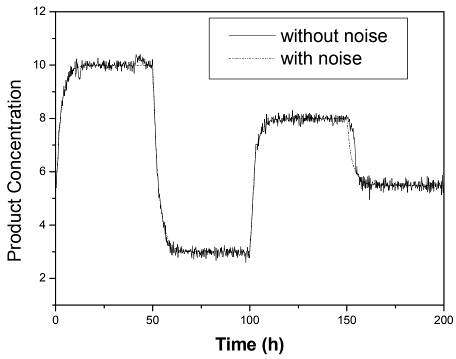


Figure 3. Servo Problem Control P.

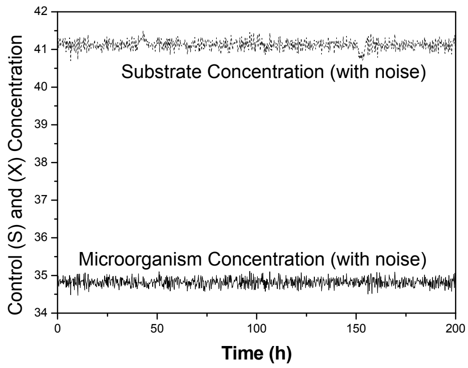
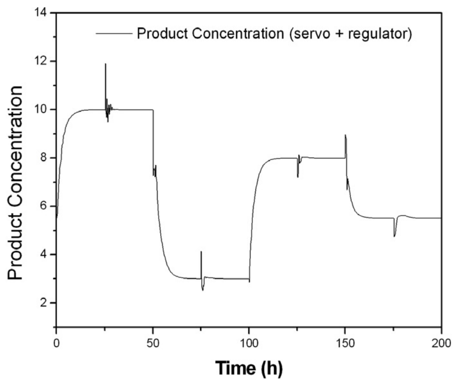


Figure 4. Servo Problem Control S and X.

Figures 5 and 6 presented the results for servo plus regulator problem.



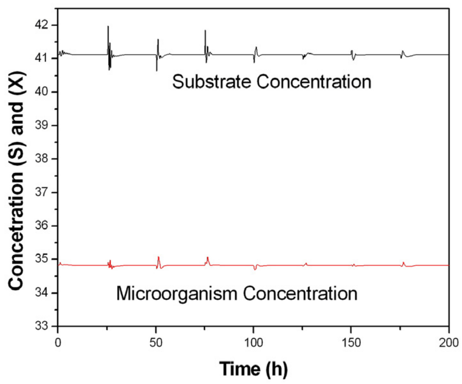


Figure 5. Servo Plus Regulator Problem (P). Figure 6. Servo Plus Regulator Problem (S) and (P)

The evolution of the estimated penalization parameters obtained by the Kalman for servo problem and servo plus regulator problem are show in the Figures 7 and 8. It can be observed the penalization parameter (λ) changes mainly when the process changes, because the applied identification scheme which is a function of process.

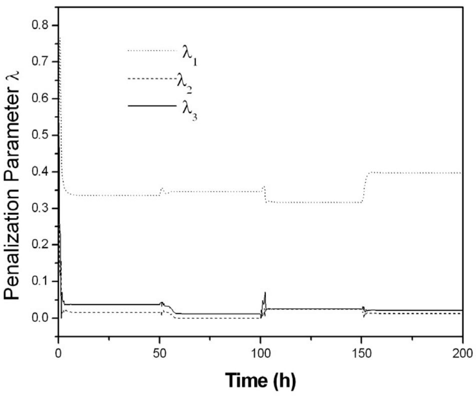


Figure 7. Evolution of λ for Servo Problem.

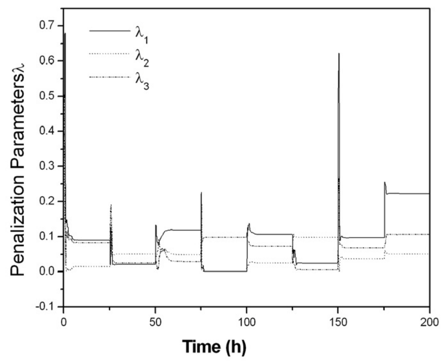


Figure 8. λ for Servo Plus Regulator Problem.

3. Conclusions

The proposed control algorithm has shown to be robust for the analysed disturbances, promising to have a great potential to be used in control strategies of large scale systems. It is noted the importance of the adjustment of the penalization parameter factor to be able to cope with change in process operations. The estimation algorithm of the penalization parameter, will determine successfully the rate of change of the system due to the control action. In case of MIMO processes the number of parameters to be tuned manually is lowered for the algorithm. Still, it has to be searched for better ways of identifying the suppression factor, which do not lead to introduction of new parameters.

References

1. Bhat, N. & McAvoy, T. J. Computers & Chemical Engineering, (1990), No.14, 573.
2. Baughman , d. R. & Liu, Y. San Diego: Academic Press, 1995.
3. Ender, L., Maciel Filho, R. In:3rd European Congress of Chemical Engineering. 2001.
4. Gadkar, K. G., Mehra, S., Gomes, J. Computers & Chemical Engineering, (2005),29, 1047.
5. Hussain, M. Artificial Intelligence in Engineering, (1999), No.13, 55.
6. Ender, L., Maciel Filho, R. Computers & Chemical Engineering, (2000),24, 937.
7. Silva, F. L. H., Rodrigues M. I., Maugeri F. *, J. Chem. Biotechnol.* (1999), No. 2, 176.