Tuning an adaptive controller using a robust control approach

Jesse Huebsch and Hector Budman

Department of Chemical Engineering, Waterloo, ON, Canada

Abstract: This paper proposes a technique for tuning of a discrete adaptive controller that is designed based on Lyapunov stability concepts. The tuning is based on the minimization of a performance index that can be calculated from a generalized eigenvalue problem (GEVP). The resulting controller, tuned with the proposed methodology, provides better performance than an adaptive controller based on a Recursive Least Squares Estimator (RLS) during sudden changes in model parameters.

Keywords: Adaptive control, robustness, tuning characteristics

1. INTRODUCTION

Adaptive controllers have been proposed for systems that cannot be accurately modelled using available off-line data. This lack of model accuracy often arises for time-varying systems and for systems for which the model structure is not known a priori. For example, the growth rate term in the mass balance equations of bioreactors is often not accurately known (Zhang and Guay, 2002). Therefore, an adaptive estimator maybe used to estimate this term and then a controller can be designed based on this estimated term.

When the model structure of a process is unknown a priori, a commonly used empirical model suitable for adaptive control design, is given as follows (Sanner and Slotine, 1992):

$$y_{k+1} = \sum_{i=0}^{n} a_{i,k} g(y_{k-i}) + \sum_{j=0}^{n} b_{j,k} h(u_{k-j}) \qquad (1)$$

where y is the state, u is the input or manipulated variable, g and h are pre-specified basis functions that can be linear or nonlinear with respect to y and u and *a's* and *b's* are *a priori* unknown parameters to be estimated on-line from input-output data. When linear basis functions are chosen, the model in equation (1) results in the common ARMA model given as follows:

$$y_{k+1} = \sum_{i=0}^{n} a_{i,k} y_{k-i} + \sum_{j=0}^{n} b_{j,k} u_{k-j} \qquad (2)$$

The fact that the models in (1) and (2) are linear with respect to estimated parameters *a's* and *b's* facilitates the design of estimators with proper convergence and performance properties as explained later in the manuscript. Different types of estimators have been used to estimate parameters using the models given by equations (1) and (2). The most common type is the recursive least square estimator (RLS) that minimizes the sum of square errors between the measured and estimated values of y. The recursive least squares estimator is defined by the following two recursive equations:

$$\hat{\boldsymbol{\theta}}_k = \hat{\boldsymbol{\theta}}_{k-1} + \frac{\mathbf{P}_{k-2}\mathbf{X}_{k-1}}{c + \mathbf{X}_{k-1}{}^T \mathbf{P}_{k-2}\mathbf{X}_{k-1}} \left[y_k - \mathbf{X}_{k-1}{}^T \hat{\boldsymbol{\theta}}_{k-1} \right] \qquad (3)$$

$$\mathbf{P}_{k-1} = \mathbf{P}_{k-2} - \frac{\mathbf{P}_{k-2}\mathbf{X}_{k-1}{}^T \mathbf{X}_{k-1}\mathbf{P}_{k-2}}{c + \mathbf{X}_{k-1}{}^T \mathbf{P}_{k-2}\mathbf{X}_{k-1}} \qquad (4)$$

Where,

$\mathbf{\theta} = [a_1,...,a_n,b_1,...,b_n]$ is the actual vector of parameters assuming that equation (1) is an accurate model of the actual process.
$\hat{\mathbf{\theta}}_k = [\hat{a}_{1,k},...,\hat{a}_{n,k},\hat{b}_{1,k},...,\hat{b}_{n,k}]$ is the vector of estimated parameters at time k
$\mathbf{X} = [y_k, y_{k-1}...y_{k-n}, u_k, u_{k-1},...,u_{k-m}]$ is referred as to the regression vector and is a function of past input-output data.
c is the forgetting factor and it is used to assign a larger weight to new data versus older data.
P is the estimation covariance matrix and it is an indicator of the uncertainty in the parameter estimates.

The estimates obtained with the RLS estimator can be used for control using for example a one-step-ahead controller. When the parameter estimates are assumed to be equal to the actual parameters, i.e. the certainty equivalence principle is applied, the one step-ahead controller is:

$$u_k = \frac{ysp_{k+1} - [0, u_{k-2},...,u_{k-n}, -y_{k-1},...,-y_{k-n}]\hat{\mathbf{\theta}}_{k+1}}{\hat{b}_{1,k+1}} \quad (5)$$

On the other hand, if a more robust controller is desired that accounts for uncertainty in the parameters, a controller referred to as *cautious* (Wittenmark, 1995) can be used that takes into account the uncertainty given by the elements of $\boldsymbol{P_k}$:

$$u_k = g_{cautious}(ysp_{k+1}, \mathbf{X}, \hat{\mathbf{\theta}}_{k+1}, \mathbf{P_k}) \quad (6)$$

Under persistent excitation conditions and for *c=0*, the parameter estimates converge to their actual values whereas the matrix **P** converges to zero. This is a desirable outcome for time invariant systems, i.e. systems for which **θ**=constant. However, this is highly undesirable for time varying systems since, according to equation (3), the parameter's adaptation stops when **P**=0. This undesirable scenario can be partially addressed by selecting a nonzero forgetting factor *c*. However, when *c* is nonzero, may cause to very large or infinite values of **P** in the absence of excitation which may ultimately lead to poor control if the controller given by equation (5) is used or alternatively to controller turn-off if the cautious controller in equation (6) is used. To address this issue, resetting of **P**, based on specific algorithms (Huzmezan et al, 2003)) or resetting at ad hoc selected time intervals, has been proposed. In summary, the tuning of an RLS estimator is challenging for chemical systems where parameters are expected to change in a gradual or step-like fashion, due to for instance occasional changes in operating conditions**.**

An alternative estimator that avoids some of the difficulties related to the RLS estimator is the gradient estimator (Sanner and Slotine, 1992). This type of estimator has been proposed for some chemical engineering applications including adaptive control of bioreactors (Perrier and Dochain, 1993). For this estimator the parameter update equation is as follows:

$$\hat{\mathbf{\theta}}_k = \hat{\mathbf{\theta}}_{k-1} + \mathbf{K}_{k-1}\mathbf{X}_{k-1}^T S_k \quad (7)$$
$$S_k = f(S_{k-1}, X_{k-1}, K_D) \quad (8)$$

Where, **K** is diagonal and is the adaptation gain matrix S_k is the tracking error and is calculated, as shown later in the manuscript, based on Lyapunov stability concepts and, K_D is a tuning constant that determines the rate of convergence of *S*.

The adaptation gain matrix may be constant or time varying. In this study, **K** will be allowed to change with time and it will be referred to as $\mathbf{K}_k$ to indicate its value at time interval *k*.

The obvious advantage of the gradient estimator is that it does not depend on an adapting covariance matrix **P** that presents inherent difficulties as discussed above**.** On the other hand, the algorithm requires proper tuning of an adaptation gain matrix that has great impact on the estimator performance. Often, in the literature, researchers have selected the gain matrix **K** ad-hoc or based on numerical simulations provided that a suitable model is available. However, there are no available techniques to systematically select the elements of **K.** Additionally, the tracking error equation also introduces an additional tuning parameter K_D as shown above. Therefore, a methodology is needed to tune adaptive controllers that used the gradient estimator given by equation (7).

The objective of the current study is to propose a tuning methodology for this type of adaptive controller with a gradient estimator.

A logical choice to select the tuning parameters ***K*** and K_D is to solve, using the information up to interval *k*-1, the following optimization problem:

$$\min_{\mathbf{K}, K_D} E\left\{\frac{1}{N}\sum_{k=1}^{N}(y(k) - y_{setpoint}(k))^2\right\} \quad (9)$$

The expectation of the sum of squares instead of the actual sum is used to account for model uncertainty in the parameters during adaptation and unmeasured disturbances. This problem is closely related to the optimal dual adaptive control problem that search for the optimal trade-off between sufficient excitation for fast model parameter identification versus good tracking properties. In the classical dual adaptive control formulation the minimization is done with respect to the future inputs whereas in equation (9) the cost is minimized with respect to the tuning parameters. However, the two problems are closely related in the sense that the control actions are directly dependent on the tuning parameters.

The problem given by (9) is difficult due to the mathematical expectation that has to be computed for all possible disturbances and in the presence of model uncertainty. Thus, only numerical solutions

have been reported for relatively simple problems and under certain assumptions (Wittenmark, 1995).
In this paper an approximate solution to the problem given by equation (9), based on robust control ideas, is proposed. The idea is to represent the closed loop system by a nominal model and model uncertainty. Using this representation, it will be shown that the problem in (9) can be formulated as an optimization of a set of linear matrix inequalities (LMI's). The paper is organized as follows. Section 2 describes the adaptive control algorithm and the stability and convergence proofs. Section 3 presents the formulation of the tuning problem as an optimization using a set of LMI's. Results and comparisons between the proposed method and an adaptive controller based on RLS estimation are presented in Section 4. Section 5 provides a brief summary and conclusions.

2. ADAPTIVE CONTROLLER ALGORITHM

The controller algorithm presented in this section is a discrete version of an algorithm proposed by Sanner (1992) for continuous systems.

2.1 Definitions

Given a DARMA (Discrete Autoregressive Moving Average) model of a system that is nth order with respect to the state and mth order with respect to the input:

$$y_{k+1} = \sum_{i=0}^{n-1} a_i y_{k-i} + \sum_{j=0}^{m-1} b_j u_{k-j} \qquad (10)$$

The vectors of the parameters a_i and b_j are defined as follows:

$$\mathbf{A} = \begin{bmatrix} a_0 & \cdots & a_{n-1} \end{bmatrix}^T \qquad (11a)$$

$$\mathbf{B} = \begin{bmatrix} b_0 & \cdots & b_{m-1} \end{bmatrix}^T \qquad (11b)$$

The parameter estimate vectors are defined as follows:

$$\hat{\mathbf{A}}_k = \begin{bmatrix} \hat{a}_{0,k} & \cdots & \hat{a}_{n-1,k} \end{bmatrix}^T \qquad (12a)$$

$$\hat{\mathbf{B}}_k = \begin{bmatrix} \hat{b}_{0,k} & \cdots & \hat{b}_{m-1,k} \end{bmatrix}^T \qquad (12b)$$

Let the values of past input and output data be given by the following vectors:

$$\mathbf{Y}_k = \begin{bmatrix} y_k & \cdots & y_{k-n+1} \end{bmatrix}^T \qquad (13a)$$

$$\mathbf{U}_k = \begin{bmatrix} u_k & \cdots & u_{k-m+1} \end{bmatrix}^T \qquad (13b)$$

Then, using equations (11)-(13), the DARMA model given by equation (10), can be reformulated in terms of the input and output vectors as follows:

$$y_{k+1} = \mathbf{A}^T \mathbf{Y}_k + \mathbf{B}^T \mathbf{U}_k \qquad (14)$$

Also for the purpose of designing an implementable controller, let

$$\hat{\mathbf{B}}_k^{old} = \begin{bmatrix} \hat{b}_{1,k} & \cdots & \hat{b}_{m-1,k} \end{bmatrix}^T \qquad (15a)$$

$$\mathbf{U}_k^{old} = \begin{bmatrix} u_{k-1} & \cdots & u_{k-m+1} \end{bmatrix}^T \qquad (15b)$$

A filtered feedback error, to be justified by the stability proof given in the following section, is given as follows:

$$s_k = \frac{-\hat{\mathbf{A}}_k^T \mathbf{Y}_{k-1} - \hat{\mathbf{B}}_k^T \mathbf{U}_{k-1} + 2y_k + (1-K_D)s_{k-1} - \hat{\mathbf{A}}_{k-1}^T \mathbf{Y}_{k-1} - \hat{\mathbf{B}}_{k-1}^T \mathbf{U}_{k-1}}{1+K_D} \qquad (16)$$

For simplicity, an adaptive algorithm based on a one-step-ahead controller will be used. A term proportional to the filtered error, s_k, is added to tune the closed loop response, as follows:

$$u_k = \left(y_{sp} - \hat{\mathbf{A}}_k^T \mathbf{Y}_k - \hat{\mathbf{B}}_k^{old^T} \mathbf{U}_k^{old} + (1-K_D)s_k \right) \cdot \hat{b}_{0,k}^{-1} \qquad (17)$$

In the particular case that $\hat{b}_{0,k}$ is zero during adaptation, the parameters are reset to the values in the previous time interval. In the presence of persistent excitation, it can be shown that the system will eventually converge to the correct values.

The errors in the estimated parameters are defined in the form of deviation variables as follows:

$$\hat{\mathbf{A}}_k = \tilde{\mathbf{A}}_k + \mathbf{A} \qquad (18a)$$

$$\hat{\mathbf{B}}_k = \tilde{\mathbf{B}}_k + \mathbf{B} \qquad (18b)$$

The gradient descent method is used to formulate the parameter update equations where the error used for updating is the sum of the current and past value of the filtered errors, $(s_k + s_{k-1})$:

$$\hat{\mathbf{A}}_k = \hat{\mathbf{A}}_{k-1} + \mathbf{K_A} \mathbf{Y}_{k-1}(s_k + s_{k-1}) \qquad (19a)$$

$$\hat{\mathbf{B}}_k = \hat{\mathbf{B}}_{k-1} + \mathbf{K_B} \mathbf{U}_{k-1}(s_k + s_{k-1}) \qquad (19b)$$

$\mathbf{K_A}, \mathbf{K_B}$ are matrices of adaptation gains, with diagonal structure. Thus, the tuning parameters of the controller are K_D and $\mathbf{K} = [\mathbf{K_A}\ \mathbf{K_B}]$.

2.2 Stability

Assumptions: For simplicity, it is assumed for the following proof that the system is time invariant or its parameters change in a step-like fashion where the time between changes is long enough such as the parameters converge to a steady state value. Also, for the first proof, the tuning parameters K_D and $\mathbf{K}$ are assumed to be constant with time. Later in this section, the proof is expanded to account for tuning parameters that change in time within a finite set of values. For brevity, only a brief description of the proof is presented.

2.3 Stability with constant tuning parameters:

A Lyapunov function made by the combination of the squares of the estimation errors and the filtered error is defined as follows:

$$V_k = \tilde{\mathbf{A}}_k^T \mathbf{K_A}^{-1} \tilde{\mathbf{A}}_k + \tilde{\mathbf{B}}_k^T \mathbf{K}_B^{-1} \tilde{\mathbf{B}}_k + s_k^2 \qquad (20)$$

Substituting equation (17) into equation (14) results in the following:

$$\begin{aligned} y_{k+1} &= \mathbf{A}^T \mathbf{Y}_k + \mathbf{B}_k^{old^T} \mathbf{U}_k^{old} + \\ & b_0 \cdot \hat{b}_{0,k}^{-1} \left(y_{sp} - \hat{\mathbf{A}}_k^T \mathbf{Y}_k - \hat{\mathbf{B}}_k^{old^T} \mathbf{U}_k^{old} + (1 - K_D) s_k \right) \end{aligned} \qquad (21)$$

When the Lyapunov energy converges to zero, $\hat{\mathbf{A}}_k = \mathbf{A}, \hat{\mathbf{B}}_k = \mathbf{B}$ and $s_k = 0$. and then it can be easily shown from equation (21) that: $y_{k+1} = y_{sp}$. For Lyapunov stability it is required:

$$V_{k+1} - V_k \le 0 \qquad (22)$$

Combining equations (18), (20) and (22) and after collecting like terms and completing squares:

$$\begin{aligned} V_{k+1} - V_k = & \left(\hat{\mathbf{A}}_{k+1} - \hat{\mathbf{A}}_k\right)^T \left(\mathbf{K_A}^{-1} \hat{\mathbf{A}}_{k+1} + \mathbf{K_A}^{-1} \hat{\mathbf{A}}_k - 2\mathbf{K_A}^{-1} \mathbf{A}\right) \\ & + \left(\hat{\mathbf{B}}_{\mathbf{k+1}} - \hat{\mathbf{B}}_{\mathbf{k}}\right)^T \left(\mathbf{K_B}^{-1} \hat{\mathbf{B}}_{\mathbf{k+1}} + \mathbf{K_B}^{-1} \hat{\mathbf{B}}_{\mathbf{k}} - 2\mathbf{K_B}^{-1} \mathbf{B}\right) \\ & + (s_{k+1} + s_k)(s_{k+1} - s_k) \end{aligned} \qquad (22)$$

After evaluating expressions (19a) and (19b) at interval k+1 and substituting the result into equation (22), the following results:

$$V_{k+1} - V_k = -K_D (s_{k+1} + s_k)^2 \qquad (23)$$

If $K_D > 0$, Equation (23) guarantees that the Lyapunov function is decreasing with time.

2.4 Stability with time varying tuning parameters

For the optimization problem given by equation (9) it is advantageous to add additional degrees of freedom to the problem by allowing the decision variables, i.e. the tuning parameters K_D and $\mathbf{K}$ ($\mathbf{K_A}$ and $\mathbf{K_B}$) to change with time whereas the stability proof in the previous section assumed that this parameters are constant in time. In this section the stability proof will be extended to account for the situation that the tuning parameters change with time. To the knowledge of the authors it is not possible to prove stability and to solve the optimization in equation (9) for infinite possible values of the tuning parameters. Therefore, it will be assumed that these parameters can only acquire a finite number of values and then a suboptimal solution of (9) will be sought using a combination of these values. Accordingly, a set of tuning parameter values will be defined as follows:

$$\Theta = \{\{K_{D1}\ \mathbf{K_{A1}}\ \mathbf{K_{B1}}\}, \{K_{D2}\ \mathbf{K_{A2}}\ \mathbf{K_{B2}}\} \ldots \{K_{Dn}\ \mathbf{K_{An}}\ \mathbf{K_{Bn}}\}\} \qquad (24)$$

For each element of the set Θ, the system is guaranteed to be stable and the parameters converge as per the proof given in the previous subsection. The key idea to ensure stability when different elements of Θ are considered along time, is to calculate simultaneously on-line the evolution the parameter estimates $\hat{\underline{B}}_k$ and $\hat{\underline{A}}_k$ and the error s_k for all elements of Θ. However, only one specific control action based on one of the elements is actually implemented at any given time. In figure 1 the curves labelled '1' through '5' refer to the Lyapunov function given by equation (20) corresponding to parameters $\lfloor K_{A,1}, K_{B,1}, K_{D,1} \rfloor$ through $\lfloor K_{A,5}, K_{B,5}, K_{D,5} \rfloor$ respectively when, for example, n=5 in definition (24). Then when a new element of Θ (25) is considered, i.e. for a specific $\{K_{Di}\ \mathbf{K_{A\mathit{i}}}\ \mathbf{K_{B\mathit{i}}}\}$, the parameter estimates $\hat{\underline{B}}_k$ and $\hat{\underline{A}}_k$ and the filtered error s_k are reset to the values corresponding to this element of the set. This switch occurring at each time interval may cause a local increase in Lyapunov energy. As an example, refer to the jump in Lyapunov energy between letters 'A' and 'B' in Fig 1. Each curve in Figure 1 corresponds to the progression of the Lyapunov function for a different element of Θ. Clearly, the Lyapunov function ultimately converges to zero despite that temporary increases in this function may occur.

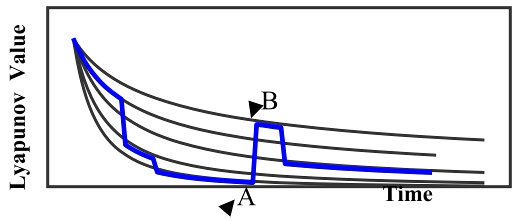

Fig 1: Lyapunov function as a function of time (each line correspond to a different element of the set Θ defined by equation (24))

3. AN APPROXIMATE SOLUTION FOR THE PROBLEM STATED IN EQUATION (9).

As mentioned above, the optimization problem given by equation (9) is very difficult since it has to be solved for all possible disturbances and model uncertainty. Therefore, in this section, an approximated solution to this problem based on a robust control approach is proposed. In order to apply this approach, a nonlinear state space model is formulated, based on definitions (10)-(19) presented above, as follows:

$$\begin{aligned}
y_{i,k+1} &= l_i\left(y_k, d_{i,k}, \widetilde{\mathbf{A}}_k, \hat{\mathbf{A}}_k, \widetilde{\mathbf{B}}_k, \hat{\mathbf{B}}_k, s_k, k\right) \\
s_{i,k+1} &= f_i\left(y_k, y_{sp,k}, d_{i,k}, \widetilde{\mathbf{A}}_k, \hat{\mathbf{A}}_k, \widetilde{\mathbf{B}}_k, \hat{\mathbf{B}}_k, s_k, \mathbf{K}_A, \mathbf{K}_B, K_D, k\right) \\
\hat{\mathbf{A}}_{i,k+1} &= g_i\left(y_k, y_{sp,k}, d_{i,k}, \widetilde{\mathbf{A}}_k, \hat{\mathbf{A}}_k, \widetilde{\mathbf{B}}_k, \hat{\mathbf{B}}_k, s_k, \mathbf{K}_A, \mathbf{K}_B, K_D, k\right) \\
\hat{\mathbf{B}}_{i,k+1} &= h_i\left(y_k, y_{sp,k}, d_{i,k}, \widetilde{\mathbf{A}}_k, \hat{\mathbf{A}}_k, \widetilde{\mathbf{B}}_k, \hat{\mathbf{B}}_k, s_k, \mathbf{K}_A, \mathbf{K}_B, K_D, k\right) \\
y_{sp,k+1} &= \tau_d^{-1} \cdot y_{sp,k} + \left(1-\tau_d^{-1}\right) y_{ref,k}
\end{aligned} \tag{25}$$

Where the disturbance $d_{i,k}$ is an output bounded disturbance and τ_D in the last equation in (25) is the desired closed loop time constant. For example, for a first order system, the first equation in (25) is:

$$y_{k+1} = a \cdot y_k + b \cdot u_k + d_k \tag{26}$$

$$d_k \in [-\delta_D, \delta_D] \tag{27}$$

The other state equations in (25) are derived from equations (10)-(19) and are omitted here for brevity.

The deviations in the model parameters $\widetilde{\mathbf{A}}_{i,k}$ and $\widetilde{\mathbf{B}}_{i,k}$, defined by equation (19), are not known because the actual values of these parameters are not known a priori. On the other hand, bound on these parameters can be obtained on-line by calculating confidence intervals of these parameters based on regression of current and past input output data. Then, based on definition (18) the deviations in parameters with respect to their nominal values $\hat{\mathbf{A}}_k$ and $\hat{\mathbf{B}}_k$ are assumed to be bounded by the identified confidence intervals as follows:

$$\widetilde{\mathbf{A}}_k \in [-\boldsymbol{\delta}\mathbf{A_k}, \boldsymbol{\delta}\mathbf{A_k}],\ \widetilde{\mathbf{B}}_k \in [-\boldsymbol{\delta}\mathbf{B_k}, \boldsymbol{\delta}\mathbf{B_k}], \tag{28}$$

Equations (27) and (28) define an uncertainty set :

$$\Delta = \left\{[-\boldsymbol{\delta}\mathbf{A_k}, \boldsymbol{\delta}\mathbf{A_k}], [-\boldsymbol{\delta}\mathbf{B_k}, \boldsymbol{\delta}\mathbf{B_k}], [-\delta_D, \delta_D]\right\} \tag{29}$$

Liu (1968) has shown that bounds on the stability and performance properties of a nonlinear system can be found from the properties of an equivalent linear time varying system that is given as follows:

$$\begin{aligned}
&\begin{bmatrix} \boldsymbol{\eta}_{k+1} \\ e_k \end{bmatrix} = \begin{bmatrix} \mathbf{E}_k(\boldsymbol{\delta}_i) & \mathbf{F} \\ \mathbf{G} & \mathbf{H} \end{bmatrix} \begin{bmatrix} \boldsymbol{\eta}_k \\ v_k \end{bmatrix} \\
&\boldsymbol{\eta}_0 \text{ is known} \\
&\boldsymbol{\eta}_k = \begin{bmatrix} y_k & s_k & \hat{\mathbf{A}}_k & \hat{\mathbf{B}}_k & y_{sp,k} \end{bmatrix} \\
&v_k = \begin{bmatrix} y_{ref,k} \end{bmatrix} \quad e_k = y_{sp,k} - y_k \\
&\mathbf{F} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1-\tau_d^{-1} \end{bmatrix}^T \\
&\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & -1 \end{bmatrix} \\
&\mathbf{H} = \begin{bmatrix} 0 \end{bmatrix}
\end{aligned} \tag{32}$$

Where the matrix $\mathbf{E}_k$ is given by the linear combinations of matrices $\mathbf{E}_{i,k}$ obtained from the Jacobian of the nonlinear system given by equations (25) calculated at each one of the possible combinations of the uncertainty set vertices defined by equation (29) as follows:

$$\left[\frac{\partial \boldsymbol{\eta}_{k+1}(\delta_i)}{\partial \boldsymbol{\eta}_k}\right] = \left[\mathbf{E}_{i,k}(\delta_i)\right] \tag{33}$$

Since the derivatives in (33) can be shown to be linear with respect to the uncertainty elements described in (29), then:

$$\left[\mathbf{E}_k\right] = \sum_{i=1}^{L} \alpha_{i,k} \left[\mathbf{E}_i(\delta_i)\right], \tag{34}$$

$$\sum_{i=1}^{L} \alpha_{i,k} = 1,\ \alpha > 0$$

For example, for a first order system, there are 8 possible combinations of the uncertainty bounds according to (34a) and (35) and correspondingly 8 possible matrices $\mathbf{E_i}$ are calculated at each time interval k.

Defining the ratio between the feedback errors to the input setpoint changes $y_{ref,k}$:

$$\gamma > \frac{\|e\|_2}{\|v\|_2} \tag{35}$$

Then, a bound on γ for all the models defined by equation (32) can be found from a General Eigenvalue Problem (GEVP) defined as follows:

$$\Phi = \min_{\mathbf{P}} \gamma$$

s.t.

$$\begin{bmatrix} \mathbf{E}_k(\boldsymbol{\delta}_i)^T \mathbf{P} E_k(\boldsymbol{\delta}_i) - \mathbf{P} & \mathbf{E}_k(\boldsymbol{\delta}_i)^T \mathbf{P} F & \mathbf{G}^T \\ \mathbf{F}^T \mathbf{P} \mathbf{E}_k(\boldsymbol{\delta}_i) & \mathbf{F}^T \mathbf{P} \mathbf{F} - \gamma^2 \mathbf{I} & \mathbf{H}^T \\ \mathbf{G} & \mathbf{H} & -\mathbf{I} \end{bmatrix} < 0 \tag{36}$$

This GEVP can be solved by using the LMI toolbox of Matlab. Then, an approximated solution to the problem given by equation (9) can be obtained from:

$$\min_{\Theta} \Phi \tag{37}$$

Where, Φ is calculated from (36) and Θ is a finite combination of tuning parameters defined by (24) and Δ is the uncertainty set defined by equation (29). The minimization in equation (37) is done using the function *fmin* in Matlab.

After initializing the values $\hat{\mathbf{A}}_k$ and $\hat{\mathbf{B}}_k$ and control action vector $\mathbf{U}_k^{old}$, the tuning procedure includes the following steps at each time interval k:

1. Calculate the uncertainty bounds in the uncertainty set Δ using available data up to the current time..
2. Update the parameter values according to equations (25) for each one of the tuning parameters combinations in the set Θ defined by equation (24).
3. Find the best tuning parameter combination in the set Θ by solving the optimization problem stated in equation (37).
4. Implement into the process the control action that corresponds to the best set of tuning parameters found in step 3 (It should be noticed that simultaneous calculations for all the combinations in the set Θ are carried on at each interval k but only

one control action corresponding to the best combination is actually implemented.

5. Go to step 1.

4. EXAMPLE

To illustrate the tuning method, a first order system was investigated as described by equations (38). First order filtered white noise disturbance d and square wave input $y_{ref,k}$ are considered and step changes in the parameters are assumed to occur at k=100 and 150 respectively as described in (38). The tuning parameters in the set Θ defined by (32) are limited to all the combinations of the values [0.4, 1.6, 3]. For example:

$$
\begin{aligned}
&\Theta = \{K_A, K_B, K_D\} = \{\{1.6,0.4,3\},\{1.6,1.6,3\},\{3,3,0.4\}....etc\}\\
&x_{k+1} = a_k x_k + b_k u_k\\
&y_k = x_k + \eta_k\\
&where\\
&k \in \{1,\cdots,200\}\\
&a_k = \begin{cases} 1.05 & k < 100 \\ 0.95 & 100 \le k \le 200 \end{cases}\\
&b_k = \begin{cases} 0.5 & k < 150 \\ 0.6 & 150 \le k \le 200 \end{cases}\\
&\eta_k = (1-\beta)d_k + \beta\eta_{k-1},\ \beta = 0.75, d \in N(0,0.005)\\
&y_{sp,k+1} = (1-\alpha)y_{sp,k} + \alpha y_{ref,k}, \alpha = 0.65,\\
&y_{ref,k} : squarewave\ (amplitude = 0.5, period = 20)
\end{aligned} \tag{38}
$$

Figure 2 shows the evolution of the tuning parameters as a function of time following the solution of the optimization given by (38) at each time interval in the neighbourhood of the parameter step change. This figure shows that the tuning parameters have to change frequently in time both due to the change in parameters and the oscillating setpoint. Table 1 shows the normalized sum of square errors along the simulation. For comparison, the sum of squared errors obtained from a simulation for an arbitrarily tuned adaptive controller with fixed in time tuning parameters (Ka=1,Kb=1 and Kd=1) is shown in Table 1. The sum of errors is significantly larger for the arbitrary tuning as compared to the LMI method illustrating the need for proper tuning. Finally an adaptive controller based on an RLS estimator is simulated. The resulting normalized error, significantly larger than the error obtained with the LMI based method, is tabulated in Table 1. As expected the RLS based controller does not perform as well as the LMI controller especially after the step change in the parameters. The reason is that before this step change occurs, at k=100, the covariance matrix P converges almost to zero and consequently the RLS estimator responds very slowly to the sudden change in the model parameters as compared to the proposed estimator tuned according to equation (38). This is clearly shown in Figure 3 where the adaptation of parameter a is shown for both the proposed tuning method and for the RLS based method.

CONCLUSIONS

A method is proposed to tune an adaptive controller based on a gradient estimator. The method uses a robust control approach to minimize a cost function in the presence of model uncertainty and disturbances. The controller based on the proposed tuning method is shown to be superior to a controller based on an RLS estimator during step changes in parameters.

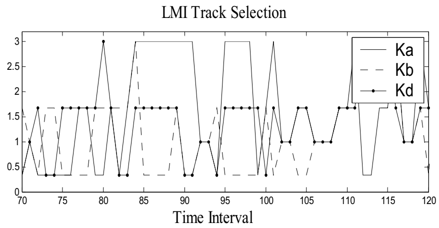

Fig. 2: tuning parameters as a function of time resulting from the proposed optimization (eq. (37))

Tuning	Ka=1,Kb=1 KD=1	Proposed method	RLS
$\Vert y_{sp}-y\Vert/\Vert y_{sp}\Vert$	0.0084	0.0074	0.0109

Table 1: Comparison of the normalized sum of squared errors.

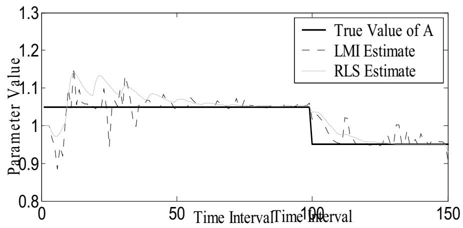

Fig. 3: adaptation of parameter a for the proposed estimator (eq.(37)) and for RLS.

REFERENCES

Liu, R. W. (1968) Convergent Systems, IEEE Trans. On Aut. Control, 13, 4, 384-391.

Huzmezan, M.,G.A.Dumont,W.A.Gough and S. Kovak (2003) Adaptive Control of Integrating Time Delay Systems: A PVC Batch Reactor, IEEE Trans. On Contr. Syst. Tech., 11, 3, 390-398.

Perrier, M. and D. Dochain (1993) Evaluation of Control Strategies for Anaerobic Digestion Processes. International J. of Adaptive and Signal Processing, 7, 309-321.

Sanner, R.M. and J.J. Slotine, (1992), Gaussian Networks for Direct Adaptive Control, IEEE Trans. On Neural Networks, 3, 837-862.

Wittenmark, B. (1995) Adaptive dual control methods: An overview. In 5th IFAC Symposium on Adaptive Systems in Control and Signal Processing, 67-72, Budapest, Hungary.

Zhang, T. and and M. Guay, (2002) Adaptive nonlinear observers of microbial growth processes, Journal of Process Control, 12, 633-643.