Adaptive Friction Compensation in DC Motors

Canudas de Wit, Carlos

1985

Document Version:
Publisher's PDF, also known as Version of record

Link to publication

Citation for published version (APA):
Canudas de Wit, C. (1985). *Adaptive Friction Compensation in DC Motors*. (Technical Reports TFRT-7285). Department of Automatic Control, Lund Institute of Technology (LTH).

Total number of authors:
1

CODEN:LUTFD2/(TFRT-7285)/1-37/(1985)

ADAPTIVE FRICTION COMPENSATION IN DC MOTORS

CARLOS CANUDAS DE WIT

DEPARTMENT OF AUTOMATIC CONTROL
LUND INSTITUTE OF TECHNOLOGY
JULY 1985

LUND INSTITUTE OF TECHNOLOGY
DEPARTMENT OF AUTOMATIC CONTROL
Box 118
S 221 00 Lund Sweden

Document name
Report
Date of issue
850711
Document number
CODEN:LUTFD2/(TFRT-7285)/1-37/(1985)
Supervisor

Sponsoring organization

Author(s)

Carlos Canudas de Wit

Title and subtitle

Adaptive friction compensation in DC motors

Abstract

The problem of torque friction compensation in direct current motors is considered. A nonlinear discrete time polynomial representation is derived from the physical laws. The control linear design is then based on a linearized model and a nonlinear compensation. Through adaptive compensation the performance of the closed-loop system is improved over the non-adaptive case, where parameter uncertainties may be high. The control law resulting from this scheme is a combination of two sources: a fixed linear controller based on the linearized model and an adaptive contribution which compensates for the nonlinear effects and model parameters uncertainty. The stability and convergence of this scheme is studied. Some simulations exemplify the main ideas.

Key words

Classification system and/or index terms (if any)

Supplementary bibliographical information

ISSN and key title

ISBN

Language
English

Number of pages
37

Recipient's notes

Security classification

Distribution: The report may be ordered from the Department of Automatic Control or borrowed through the University Library 2, Box 1010, S-221 03 Lund, Sweden, Telex: 33248 lubbis lund.

ADAPTIVE FRICTION COMPENSATION

IN DC MOTORS

*Carlos Canudas de Wit

Department of Automatic Control
Lund Institute of Technology
Box 118, S-221 00 Lund
Sweden

ABSTRACT

The problem of torque friction compensation in direct current motors is considered. A nonlinear discrete time polynomial representation is derived from the physical laws. The control linear design is then based on a linearized model and a nonlinear compensation. Through adaptive compensation the performance of the closed-loop system is improved over the non-adaptive case, where parameter uncertainties may be high. The control law resulting from this scheme is a combination of two sources: a fixed linear controller based on the linearized model and an adaptive contribution which compensates for the nonlinear effects and model parameters uncertainty. The stability and convergence of this scheme is studied. Some simulations exemplify the main ideas.

* Laboratoire d'Automatique de Grenoble. B.P. 46, 38402 Saint Martin d'Heres, France. Member of the research group GRECO-Systemes adaptatifs. The Author work's is sponsored by CONACYT-MEXICO.

CONTENTS

INTRODUCTION

During the last years adaptive control theory has been growing and maturing. The strong assumptions under this theory was created have been progressively relaxed making its applications feasible. As a result of this work, several industrial products based on these ideas are actually available on the market, see Åström (1985).

Some of these assumptions are still necessary, this is the case of the hypothesis of linearity. This property may seem restrictive since it is well known that the most of the systems do not fulfill this condition. However in many instances the presence of nonlinear elements make a minor contribution to the system variance that their effects can be ignored for most purposes. In somes cases, where these characteristiques can not be ignored, it is possible to find simple solutions: i.e. try to split the system into distinct modes or regimes of operation over which it is effectively linear. Other possibilities consider the existence of transformations by which the input and the output of a nonlinear system can be reduced to a linear or stationary form. If any of these conditions is met, it will be necessary to adopt more complex methods.

As we mentioned above, the use of adaptive techniques in nonlinear systems, demands a particular solution which is proper to the process nature. In this work we consider the case of the electrical DC motors, where the friction torque is a piecewise nonlinear function of the angular velocity. By analysing the asymmetrical nonlinear structure we can obtain a nonlinear discrete time representation. This model isolate the friction torque effects in order to compensate them with an extra term in the control law.

The adaptive scheme that we will introduce here will try to use the maximum a priori information available from the system: the structure of the nonlinear block and some knowlege of the model parameters. It seems natural to use adaptive schemes (or explicit identification) which not destroys this a priori informations.

It should be also noted that we only attempt to estimate the part of the system which is related to the nonlinear nature and to the model parameter uncertainty. These estimates will serve to compensate the friction torque effects (it is called here "adaptive nonlinear compensation") and allow to carry out a linear control design based in a linearized model.

The resulting control structure can be viewed as a combination of a fixed linear controller and a feedback adaptive compensation. This control scheme is proved to be globally stable and globally convergent. To see previous work on similar nonlinear systems, refer to Kung,M.C. and B.F. Womack (1984).

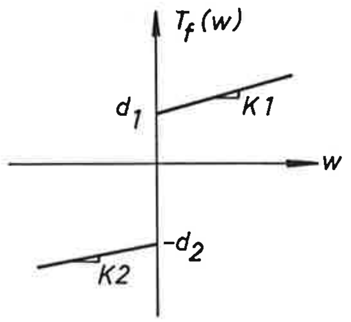

FIG.1 Asymmetrical nonlinearity between the angular velocity and the friction torque.

1.- PROBLEM STATEMENT

The set of equations below describe the behavior of a electrical D.C. motor driven by current. Equation (1) is the torque balance equation, (2) is the electrical balance equation and the set (3) related (1) and (2).

$$J \frac{dw(t)}{dt} = T_e(t) - T_f(t) \tag{1}$$

$$L_a \frac{di(t)}{dt} = -R_a\, i(t) - e_b(t) - e(t) \tag{2}$$

$$e_b(t) = K_b\, w(t) \quad ; \quad T_e(t) = K_i\, i(t) \tag{3}$$

$T_e(t)$ is the electrical torque, $T_f(t)$ is the nonlineal friction torque related to the angular velocity w(t) by a piecewise nonlinearity showed in Fig.1. e(t) is the input voltage which drives the output w(t). K_b, K_i, R_a, L_a and J are the electrical and mechanical constants .

For most purpose the inductance effects can be neglected (L_a=0). The reduced model can be rewriten into a discrete time version by using Euler expansion as:

$$y(t+1) = \begin{cases} (a+a_1)y(t)+b_0u(t)-\delta_1 & \text{if } y(t) > 0 \\ (a+a_2)y(t)+b_0u(t)+\delta_2 & \text{if } y(t) < 0 \end{cases} \tag{4}$$

This equation describes a nonlinear SISO process which is splite into two regions of operation over which a linear model is established. y(t) represent w(t) and the input e(t) is renamed u(t). The model parameters are :

$$a = 1 - (T_s K_b K_i) / JR_a \qquad b_0 = (T_s K_i) / JR_a$$
$$a_1 = (T_s K_1) / J \qquad \delta_1 = (T_s d_1) / J \tag{5}$$
$$a_2 = (T_s K_2) / J \qquad \delta_2 = (T_s d_2) / J$$

where T_s is the sampling time period, "a" is the part of the dynamic mode which is independent of the nonlinear characteristics and can be calculated from the motor constants as well as b_0. The parametes $a_1, a_2, \delta_1, \delta_2$ are related to the asymmetric piecewise nonlinear coefficients.

2.- LINEAR DESIGN

To improve the linear control design it is necessary to have a linearized version of the plant model. By reformulating equation (4) we can separate the terms related to the nonlinear feedback and compensate with an extra term in the control law. To do this we define the function h(t) as:

$$h(t) = \begin{cases} 1 & \text{if } y(t) > 0 \\ 0 & \text{if } y(t) < 0 \end{cases} \tag{6}$$

then, the model (4) can be rewriten as:

$$y(t+1) = a\, y(t) + b_0\, u(t) + g(t) \tag{7}$$

$$g(t) = (a_1 y(t) - \delta_1) h(t) + (a_2 y(t) + \delta_2)(1 - h(t)) \tag{8}$$

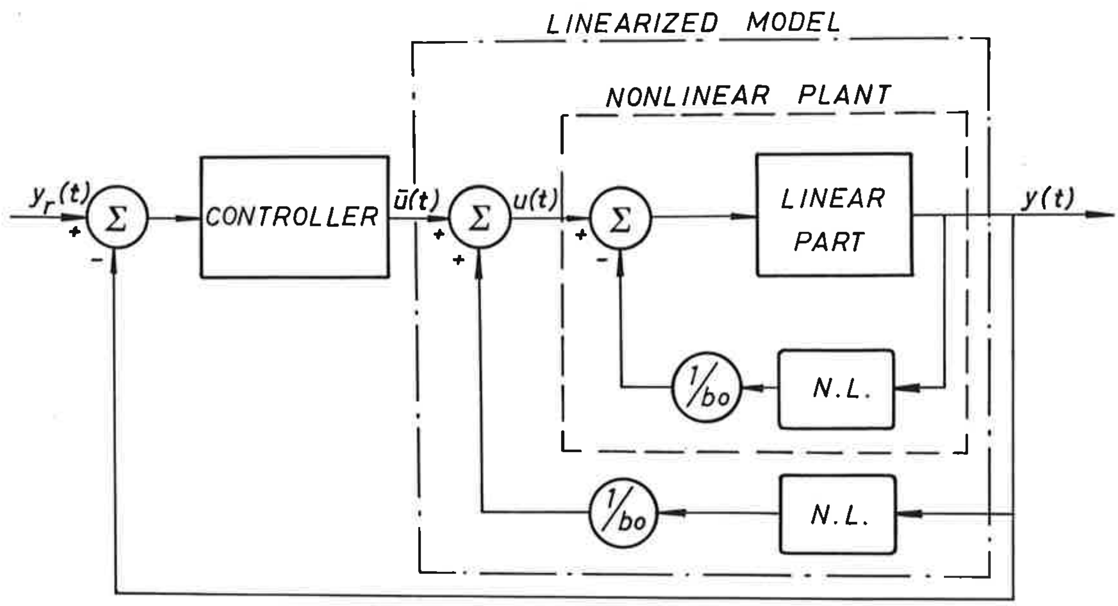

FIG.2 Control scheme with nonlinear compensation. Based on a linearized model.

Now, if we apply the control law $u(t) = \bar{u}(t) - g(t)/b0$, we obtain the next linearized model:

$$y(t+1) = a\ y(t) + b_0\ \bar{u}(t) \tag{9}$$

and we can find a control law $\bar{u}(t)$ based on linear control design and on the model (9). The Figure.2 shows this idea.

3.- ADAPTIVE DESIGN

The adaptive design will be carried out with the folowing idea in mind: we will try to use the maximum a priori information available from the process. Two differents types of a priori information can be distinguished in the DC motor case; structure of the nonlinear block and some knowlege of the model parameters.

Assume the following:

A1 : The mechanical and electrical constants are well known. Hence the parameters "a" and "b_0" are known exactly .

A2 : The structure of the piecewise nonlinearity is known but not the parameters d_1, d_2, K_1, K_2.

The key idea is to estimate only the part of the system which is related to the nonlinear nature and to the uncertainties in the model parameters (g(t) in equation (9)), and base the linear design on the known coefficients a and b_0. The final scheme will be a combination of a fixed controller and adaptive nonlinear compensation.

To obtain the adaptive version we proceed as follows:

the model (7) is rewriten as:

$$y(t+1) = \phi(t)^T\theta^* + T(t) \quad ; \quad T(t) = ay(t) + b_0u(t) \qquad (10)$$

where;

$$\phi(t)^T = [y(t)h(t), -h(t), y(t)(1-h(t)), (1-h(t))]$$

$$\theta^* = [\ a_1,\ \delta_1,\ a_2,\ \delta_2\ ]^T$$

and h(t) is the switch defined in (6). Note that the scalar product $\phi(t)^T\theta^*$ in equation (10) is equal to g(t) in (8). Next, the adaptive predictor in (10) will be given by:

$$\hat{y}(t+1|t) = \phi(t)^T \hat{\theta}(t) + T(t) \tag{11}$$

$$\hat{\theta}(t) = [\ \hat{a}_1 , \hat{\delta}_1 , \hat{a}_2 , \hat{\delta}_2\ ]^T$$

The prediction error is then :

$$e(t) = y(t) - \hat{y}(t|t-1) \tag{12}$$

$$e(t) = -\phi(t-1)\ \tilde{\theta}(t-1) \quad ; \quad \tilde{\theta}(t-1) = \hat{\theta}(t-1) - \theta^* \tag{13}$$

To estimate $\hat{\theta}(t)$ we use a RLS algorithm which is described by the next set of equations:

$$e(t) = y(t) - \phi(t-1)^T \hat{\theta}(t-1) - T(t-1)$$

$$\hat{\theta}(t) = \hat{\theta}(t-1) + P(t-1)\phi(t-1)e(t) \tag{14}$$

$$P(t) = P(t-1) - P(t-1)\phi(t)\phi(t)^T P(t-1)/[1+\phi(t)^T P(t-1)\phi(t)]$$

The control law, as we mentioned before, will have the form;

$$u(t) = \bar{u}(t) - \phi(t)^T \cdot \hat{\theta}(t)/b_0 \tag{15}$$

where $\bar{u}(t)$ is the contribution of the fixed controller and $\phi(t)^T\hat{\theta}(t)/b_0$ is the adaptive nonlinear compensation. Figure.3 shows this.

PROPERTIES OF THE ESTIMATION ALGORITHM

It is important to guarantee that the elementary properties of the RLS estimation algorithm will not change after the model manipulations. Basically we are estimating the function g(t) which has been expressed as a scalar product of the observation vector $\phi(t)$ and parameter vector $\hat{\theta}(t)$, where $\hat{\theta}(t)$ is linear in parameters and $\phi(t)$ is a nonlinear relation of the output y(t). The estimation error has been writen as a scalar product of $\phi(t)^T\tilde{\theta}(t)$. Hence we can guarantee that the basic properties of the RLS algorithm will not be modified. Nevertheless, it should be noted that if the assumption A1 does not hold, an extra term will appear in the estimation error and this will produce biased estimation (a further

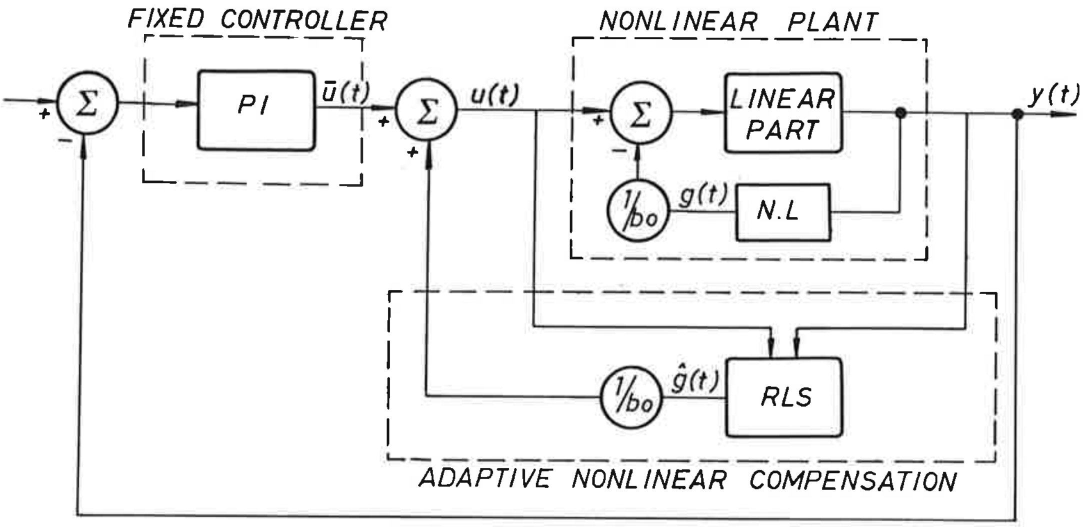

FIG.3 Control scheme with adaptive N.L. compensation.

discussion is given at the end of section 4).

The four parameters of the estimation vector demand the invertion of a 4 x 4 matrix P(t). The computation burden can be reduced by simplifying the internal structure of the RLS equations. In order to see this, the next definitions are made:

$$\phi(t)^T = [\ \phi_1(t)^T,\ \phi_2(t)^T\ ]$$

$$\hat{\theta}(T) = [\ \hat{\theta}_1(t)^T,\ \hat{\theta}_2(t)^T\ ]^T \qquad (16)$$

where

$\phi_1(t)^T=[y(t)h(t),-h(t)]$; $\hat{\theta}_1(t)=[\hat{a}_1(t),\hat{\delta}_1(t)]^T$

$\phi_1(t)^T=[y(t)(1-h(t)),(1-h(t))]$; $\hat{\theta}_2(t)=[\hat{a}_2(t),\hat{\delta}_2(t)]^T$

Note that

$$\phi(t)^T = \begin{cases} [\ \phi_1(t)^T,\ \underline{0}^T\ ] & \text{to} \quad y > 0 \\ [\ \underline{0}\ ,\ \phi_2(t)^T\ ] & \text{to} \quad y < 0 \end{cases} \tag{17}$$

Let $P(-1) = \alpha I$ ($\alpha >> 0$) be the initial value of the $P(t)$ matrix. The inverse of $P(t)$ is calculate by the recursive equation as:

$$P(t)^{-1} = P(t-1)^{-1} + \phi(t)\phi(t)^T$$

or

$$P(t)^{-1} = P(-1)^{-1} + \sum_{i=0}^{t-1}\phi(i)\phi(i)^T$$

using definitions (16,17), we rewrite this recursive equation as:

$$P(t)^{-1} = \frac{1}{\alpha} I + \sum_{i=0}^{t-1} \left[\begin{array}{c|c} \phi_1(i)\phi_1(i)^T & \phi_2(i)\phi_1(i)^T \\ \hline \phi_1(i)\phi_2(i)^T & \phi_2(i)\phi_2(i)^T \end{array} \right] \tag{18}$$

In accordance with the equation (17), all sub-matrices given by the vector product $\phi_k(i)\phi_j(i)^T$ $\forall$ $k \neq j$ will belong to the null space. Equation (18) can be written as follows:

$$\begin{bmatrix} P_1(t) & \underline{0} \\ \underline{0} & P_2(t) \end{bmatrix} = \begin{bmatrix} \frac{1}{\alpha} I + \sum_{i=0}^{t-1}\phi_1(i)\phi_1(i)^T & \underline{0} \\ \underline{0} & \frac{1}{\alpha} I + \sum_{i=0}^{t-1}\phi_2(i)\phi_2(i)^T \end{bmatrix} \tag{19}$$

Since $P_1(t)$ and $P_2(t)$ are independent, two 2 x 2 matrices can be inverted instead of one 4 x 4. This apply "mutatis mutandis" to the recursive invertion form:

$$P_1(t) = P_1(t-1)[\ I - \phi_1(t)\phi_1(t)^T P_1(t-1)/(\ 1 + \phi_1(t)^T P(t-1)\phi_1(t))\ ]$$

$$P_2(t) = P_2(t-1)[\ I - \phi_2(t)\phi_2(t)^T P_2(t-1)/(\ 1 + \phi_2(t)^T P(t-1)\phi_2(t))\ ] \tag{20}$$

Now using the equations (20) and (17), we can write the parameter estimation equations as two decoupled algorithms, where $\hat{\theta}(t)$ can be partioned into two disjoint sets of parameters each one associated with the $\hat{\theta}_j(t)$ and $\phi_j(t)$ vector $\forall$ j ε [1,2].

$$\hat{\theta}_1(t) = \hat{\theta}_1(t-1) - P_1(t-1)\phi_1(t-1)e(t)$$
$$\hat{\theta}_2(t) = \hat{\theta}_2(t-1) - P_2(t-1)\phi_2(t-1)e(t) \qquad (21)$$

where e(t) is the same as (12).

4.-EXAMPLES

The following examples are based on the control schema derived before and show the closed-loop performances obtained using adaptive compensation. The linear design is based on pole placement with zero cancellation. Integral action is included in order to improve high gain at low frequences (care should be taken to prevent windup phenomena).

The linear controller contribution is given by :

$$\bar{u}(t) = -\frac{R}{SD}y(t) + \frac{Cr(1)}{SD}y_r(t) \qquad (22)$$

D is the integral action, Cr is the polynomial which defines the closed-loop desired poles, yr(t) the reference signal and polynomials S and R are obtained from the solution of the Diophantine equation:

$$SDA + RB\,q^{-1} = BCr$$

A, B are the same as defined in (9). After adding the adaptive compensation the control law applied to the plant is given by:

$$u(t) = -\frac{R}{SD}\,y(t) + \frac{CR(1)}{SD}\,y_r(t) - \frac{1}{b_0}\,\hat{g}(t) \qquad (23)$$
$$\hat{g}(t) = \phi(t)^T\,\hat{\theta}(t)$$

The numerical values can be found in Appendix B.

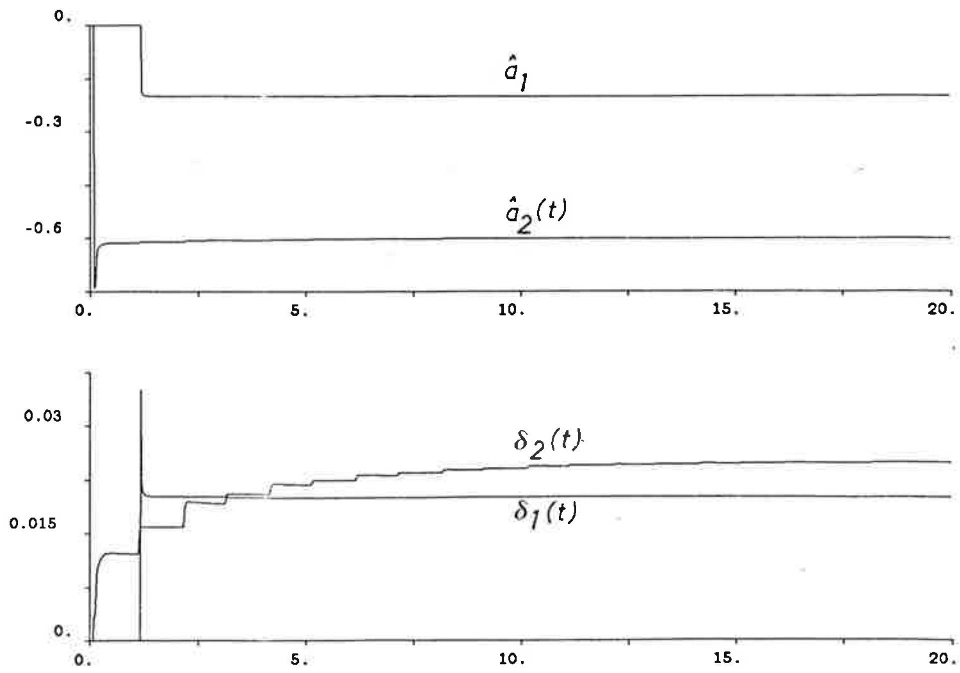

FIG 3. Time estimation of parameters $\hat{a}_1, \hat{a}_2$ and $\hat{\delta}_1$, $\hat{\delta}_2$.

EXAMPLE 1

In this example the assumptions A1 and A2 hold, the process is simulated by a discret time equations as is described in (4), the coefficients "a" and b_0 are exactly known (assumption A1). Hence perfect parameters estimation is obtained as shown the Fig.3 . Notable differences in the rate of convergence between the sets $\hat{a}_i$ and $\hat{\delta}_i$ are obtained. These are typical results when the process bias (δ) is estimated. Ameliorations in the rate of convergence of $\hat{\delta}$'s can be archived by using a different forgeting factor for each parameters set.

The performance obtained with the adaptive compensation are shown in Fig.4. Three curves are shown. Curve 1 shows the ideal case where the g(t) compensation is known exactly. Curve 2 shows the case where internal characteristics of the nonlinearity are approximated by a piecewise block having the same values of d_1 and d_2 but taking $k_1=k_2=.4$, this augment the gain when the output is positive and reduces it to negative outputs. Curve 3 shows the adaptive

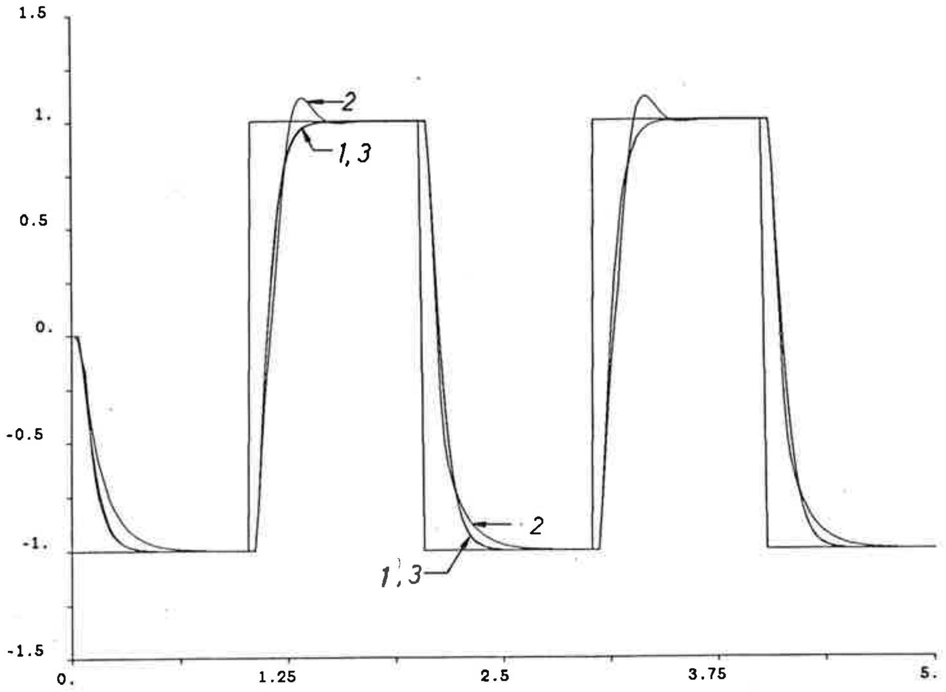

FIG 4. Closed-loop responses. Process output. Curve 1 Fixed compensation, ideal case. Curve 2 Fixed compensation, non ideal case. Curve 3 Adaptive compensation.

version, the output follows the ideal trajectory after some sampling periods.

EXAMPLE 2

In this example we analize a case when assumption A1 doesn't hold. This means we have uncertainty in the parameters a and b_0. This incertitude may come from two sources: 1) The physical coefficients of the process are inexact. 2)The time discrete representation obtained in (4) is an approximated version of the continuous time plant and hence, the model parameters derived from it will also be approximations.The pictures (5) and (6) show this. The plant is simulated by a continuous time system similar to the set (1-3).

Note that the uncertainty in the coefficient "a" will be assimilated by the estimation vector $\hat{\theta}(t)$ in $\hat{a}_1$, $\hat{a}_2$. This is not the case for b_0 , therefore an extra

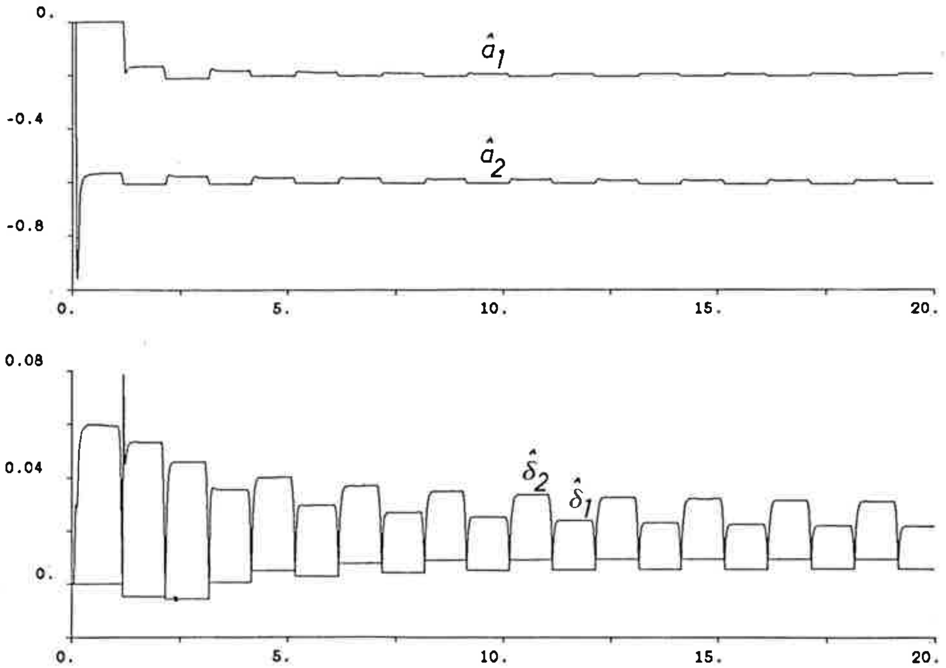

FIG 5. Time estimation of parameters $\hat{a}1$, $\hat{a}2$, $\hat{\delta}1$, $\hat{\delta}2$.

term will appear in the estimation error e(t) and the estimation vector will not follow the optimal trajectory, as shown in Fig.5. In picture (6) the dynamic response of the adaptive and ideal case are slightly deviated of the desired ones. This is produced by the reasons discussed before (approximation structure and coefficient of the model). Note that, even when b_0 is not exactly Known, the closed-loop performances (obtained with adaptive compensation) are improved over the non-adaptive case. Similar results, as in Example 1, were founded in simulations with a discrete time model when the value of b_0 ,in the equation (23) , was unexact.

It is possible to avoid the bias estimation problem by extending the size of the estimation vector and let place to estimate the incertitude on the coefficient b_0. A trade off will be then established between the quality of the estimations and the complexity of the estimation algorithm.

Until now, there is not a clear solution to avoid problems caused by unexact apriori process informations. Similar drawbacks appear in the "filter RLS " version of Sartry (1984), where the " Known plant dynamics " shall be exact in order to avoid estimation bias.

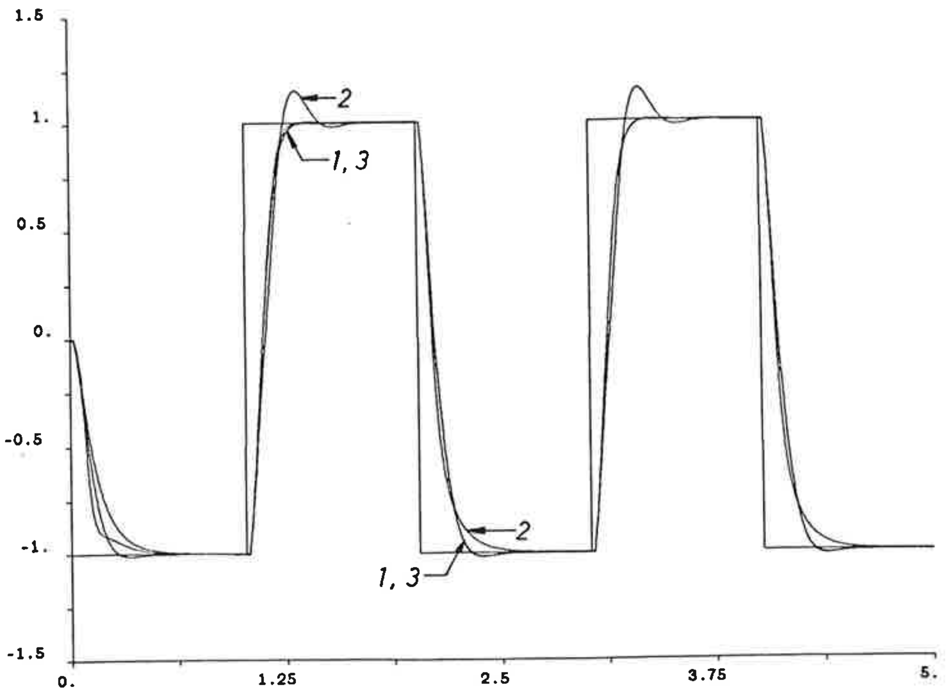

FIG.6 Closed-loop responses. System output. Curve 1 Ideal case.Fixed controller. Curve 2 non-ideal case.Fixed controller. Curve 3 Adaptive version.

5.- STABILITY ANALYSIS

The aim of this section is to show that the closed-loop system is stable and convergent in the deterministic case. In order to widen the analysis we will reformulate the model gived in the Section 1 to a more general one. Some properties of the closed-loop system will be given as result of the general linear controller structure. In a similar way, the closed-loop system properties of the adaptive version will be established in Lemma 2. This analysis will serve as a preparation to prove closed-loop system stability institute by Theorem 1.

GENERAL MODEL

Consider a nonlinear time invariant system having a representation of the form:

$$y(t+1)=\begin{cases} (A_1^*+\tilde{A}_1^*)y(t)+(B_1+\tilde{B}_1)u(t)+\delta_1 & \text{if } y(t)>0 \\ (A_2^*+\tilde{A}_2^*)y(t)+(B_2+\tilde{B}_2)u(t)+\delta_2 & \text{if } y(t)<0 \end{cases} \tag{24}$$

Where the polynomials $A1^*, B1^*, A2^*, B2^*$ (of order na1,nb1,na2,nb2 respectively) are the known-model part of the plant. $\tilde{A}1^*, \tilde{B}1^*, \tilde{A}2^*, \tilde{B}2^*$ (of order $n\tilde{a}1, n\tilde{b}1, n\tilde{a}2, n\tilde{b}2$ respectively) are the unknown-model part of the plant which is provided by the model incertitude and by the non-linear feed-back of the process.

The generality of this model is restrained by the type of applications in which we are involved. We prefer carry-out the following analysis with a general polynomial representation rather than a more particular form in order to show up the overall properties of the design.

Note that we can always let $A1^*=A2^*=A^*$ and B1=B2=B. The difference in each case will be absorbed by the corresponding polynomial uncertainly ($\tilde{A}1, \tilde{A}2$). Then equation (24) can be reduced to a more compact form:

$$y(t+1)=A^*y(t)+Bu(t)+g(t) \tag{25}$$

where:

$$g(t)=g_1(t)h(t)+g_2(t)(1-h(t))$$

$$g_1(t)=\tilde{A}_1{}^*y(t)+\tilde{B}_1u(t)+\delta_1$$

$$g_2(t)=\tilde{A}_2{}^*y(t)+\tilde{B}_2u(t)+\delta_2 \qquad (26)$$

and

$$h(t)=\begin{cases} 1 & \text{if} \quad y(t) > 0 \\ 0 & \text{if} \quad y(t) < 0 \end{cases} \qquad (27)$$

The nonlinear model (25) can be linearized if we apply a control law of the next form:

$$u(t) = \bar{u}(t) - \bar{g}(t) \qquad (28)$$

where $\bar{g}(t)$ is the filtered version $g(t)/B$ .Note that we can always choose B to have its roots inside of the unit circle. [The notation X^*indicate $X=(1-q^{-1}\ X^*)$; $X^*=x1\ +x2q^{-1}\ +\ ...+xnxq^{-nx+1}$]. Hence, after using the precedent control law, the equation (25) can be transformed to:

$$y(t+1)=A^*y(t)+B\bar{u}(t) \quad \text{or} \quad Ay(t)=q^{-1}B\bar{u}(t) \qquad (29)$$

This linear non-minimal phase system (B has been chosen as that) will be used for the linear design.

CLOSED-LOOP GENERAL LINEAR DESIGN

The most general linear controller is giving by the following:

$$R\bar{u}(t)=Ty_r(t)-Sy(t) \quad ; \quad y_r(t)=A_m(1)\bar{y}_r(t) \qquad (30)$$

where $\bar{y}r(t)$ is the reference signal, y(t) the process output and u(t) the input applied to the linearized system (29). Since the B polynomial can be chosen having all its roots inside of the unit circle, we are able to carry-out pole-zero cancellation linear design. Let Am be the polynomial which describe the desired closed-loop characteristics. The R,S,T polynomials will be finded by solving the

Diophantine equation:

$$RA + q^{-1}BS = TBA_m \qquad ; \; T = A_n A_o \quad ; \qquad R = DR' \qquad (31)$$

where Ao is the observer polynomial, An a notch filter and D the internal model (These polynomials can be included or not, the simplest case : Ao=An=D=1). Integral action can be included to improve robustness in the closed-loop system. This is achieved by putting D= $1-q^{-1}$.

The next lemma describes the closed-loop properties of the system (25) resulting from the feedback law (30).

LEMMA 1

Provided that the model (29) has been constructed keeping A and B prime and that the zeros of B are inside of the unit circle, the closed-loop system resulting from the feedback law (30,31) has the following properties:

(i) $A_m y(t) = q^{-1} y_r(t)$ (32)

(ii) $(B+\tilde{B}_1)A_m\, u_1(t) = (A-\tilde{A}_1{}^* q^{-1}) y_r(t) - A_m\, \delta_1$

$(B+\tilde{B}_2)A_m\, u_2(t) = (A-\tilde{A}_2{}^* q^{-1}) y_r(t) - A_m\, \delta_2$ (33)

(iii) $A_m[y(t) - y_r(t)] = (q^{-1} - A_m) y_r(t)$ (34)

(iv) $\lim_{t\to\infty}(y(t)- yr(t)) = 0$ provided that Am stable and yr(t) constant. (35)

(v) {u(t)} and {y(t)} are bounded provided that the polynomials $(B+\tilde{B}1)$, $(B+\tilde{B}2)$ and Am have theirs roots inside of the unit circle. (36)

The proof of Lemma 1 is refered to the Appendix A.

ADAPTIVE CONTROL ALGORITHM

We shall analyse the adaptive nonlinear compensation algorithm based on the same precedent linear design philosophy. We will first establish some definitions and the closed-loop relations. Afterward we will formulate the main theorem which guarantees closed-loop stability.

Let the adaptive algorithm be described by the next sets of equations:

ADAPTIVE PREDICTOR:

$$\hat{y}(t+1|t)=A^{*}y(t)+Bu(t)+\hat{g}(t) \tag{37}$$

where:

$$\hat{g}(t)= \phi(t)^{T}\hat{\theta}(t)=[\phi_{1}(t)^{T}h(t),\phi_{2}(t)(1-h(t))]\cdot\begin{bmatrix}\hat{\theta}_{1}(t)\\ \hat{\theta}_{2}(t)\end{bmatrix}$$

$$\begin{aligned}\phi_{1}(t)^{T}&=[y(t),\ \dots,y(t-\tilde{na}1),u(t),\ \dots,u(t-\tilde{nb}1),1]\\ \phi_{2}(t)^{T}&=[y(t),\ \dots,y(t-\tilde{na}2),u(t),\ \dots,u(t-\tilde{nb}2),1]\end{aligned} \tag{38}$$

$$\begin{aligned}\theta_{1}(t)^{T}&=[\tilde{a}_{1}^{1}(t),\ \dots,\tilde{a}_{\tilde{na}1}^{1}(t-\tilde{na}1),\tilde{b}_{0}^{1}(t),\ \dots,\tilde{b}_{\tilde{nb}1}^{1}(t-\tilde{nb}1),\delta_{1}]\\ \theta_{2}(t)^{T}&=[\tilde{a}_{1}^{2}(t),\ \dots,\tilde{a}_{\tilde{na}2}^{2}(t-\tilde{na}2),\tilde{b}_{0}^{2}(t),\ \dots,\tilde{b}_{\tilde{nb}2}^{2}(t-\tilde{nb}2),\delta_{2}]\end{aligned} \tag{39}$$

PREDICTION ERROR:

$$e(t) = y(t) - \hat{y}(t|t-1) = g(t-1) - \hat{g}(t-1) \tag{40}$$

PARAMETER ESTIMATION ALGORITHM: Use a RLS algorithm.

CONTROL LAW:

$$Ru(t) = Ty_{r}(t) - Sy(t) - R\hat{\bar{g}}(t) \quad ; \quad \hat{\bar{g}}(t) = \hat{g}(t)/B \tag{41}$$

ASSUMPTIONS:

A1:
The system model (24) has been transformed into the form (25), where A,B are prime polynomials, B; $|z| < 1$ and R,T,S polynomials are obtained from the solution of (31).

A2:
The polynomials $(B + \tilde{B}1)$, $(B + \tilde{B}2)$, T and Am are stable. If this assumption seems restrictive then the pole-zero cancellation design can be substituted by a pole assignment controller and the restriction on $(B + \tilde{B}1)$ and $(B + \tilde{B}2)$ will be removed.

Now we can establish the following lemma:

LEMMA 2

Let assumptions A1 and A2 hold for the system (24), the closed-loop system resulting from the adaptive non-linear compensation law (41) has the following properties :

(i) $$TBA_m y(t) = q^{-1}BTy_r(t) + Re(t) \tag{42}$$

(ii) $$TBA_m[B+\tilde{B}_1]u_1(t)=[A-q^{-1}\tilde{A}_1{}^*]BTy_r(t)+[A-q^{-1}\tilde{A}_1{}^*]Rq^{-1}e(t)-A_mT\delta_1$$

$$TBA_m[B+\tilde{B}_2]u_2(t)=[A-q^{-1}\tilde{A}_2{}^*]BTy_r(t)+[A-q^{-1}\tilde{A}_2{}^*]Rq^{-1}e(t)-A_mT\delta_2$$

(43)

where :

$$u(t) = u_1(t)h(t) + u_2(t)(1-h(t))$$

The proof is refered in Appendix A.

Now we can enunciate the next stability theorem based in the elementary properties of the RLS estimation algorithm and the lemmas established before.

THEOREM 1

Provided that assumptions A1 and A2 hold for the system (24) ; Use a recursive least square estimation algorithm and the adaptive nonlinear compensation law (41), then :

(i) $\lim_{t \to \infty} e(t) = 0$ (44)

(ii) $\{ \|\phi(t)\| \}$ is bounded (45)

(iii) $\lim_{t \to \infty} [A_m y(t) - q^{-1} y_r(t)] = 0$ (46)

The proof is refered in Appendix A.

CONCLUSIONS

This work deals with a specific application of the adaptive scheme on an electrical current motor. This type of process has a nonlinear components which degrades the linear control design and hence can not be ignored. By handling the continuous time differential equations we arrived at a partitioning of the overall system to two sets of independent discrete time linear equations (Section 1). The terms involved with the nonlinear effects and uncertainty on parameter model were concentrated in an explicit function (g(t)). The linear design was carried out (in Section 2) by using g(t) as a compensation term. An adaptive version was developed (in Section 3) . The overall scheme can be viewed as a linear fixed controller with a feedback adaptive compensation. Somes examples show the performance achieved with the adaptive compensation and the possible problems and solutions that may come up (when the main assumptions do not hold, Section 4). Section 5 gives a formal stability analysis that is carried out under a general framework. Details of simulations and proofs of lemmas are refered to Appendix B and A respectively.

ACKNOLEDGEMENTS

I would like to tank,

Profesor K.J. Åström and all the members of the departament for their hospitality and helful sugestions.

Profesor I.D. Landau my adviser at LAG-ENSIEG Fance.

The next organizations: CONACYT (Mexico), CEFI and GRECO-Systemes adaptatifs (France), which support the author's work.

REFERENCES

Äström, K.J. (1985). Adaptive Feedback Control. IFAC'85 York U.K. To apear.

Kung, M.C. and Womack B.F. (1984). Discrete Time Adaptive Control of Linear Systems with Preload Nonlinearity. Automatica. Vol.20. No.4. pp 477-479.

Clary, J.P. and Franklin, G.F. (1984). Self-tuning Control with A Priori Plant Knowledge. Proceedings of 23rd Conference on Decision and Control. Las Vegas, NV.

Goodwin, G.C. and Sin, K.S. (1984). Adaptive Filtering Prediction and Control. Prentice-Hall, INC.

Goodwin, G.C. , Ramage, P.J. and Caines, P.E. (1980). Discrete-time Multivariable Adaptive Control, IEEE Trans. Aut. Control, AC-25, 449.

This augmented system have the inputs $y_r(t)$, δ_1, δ_2 and the outputs $u_1(t)$, $u_2(t)$ and $y(t)$. In view that sequence $\{y_r(t)\}$ is bounded and δ_1, δ_2 are a constant term, the sequences $\{u_1(t)\}$, $\{u_2(t)\}$ and $\{y(t)\}$ will be bounded provided that the characteristic polynomial $(B+\tilde{B}1)(B+\tilde{B}2)A_m$ has all its roots inside of the unit circle (BIBO stability). The control law sequence $\{u(t)\}$ is the union of the disjoint sequence sets $\{u_1(t)\}$ and $\{u_2(t)\}$, hence if these sequences are bounded $\{u(t)\}$ will be also.

Remark

The polynomial B has been chosen in order to make possible the pole-zero cancellations design. However if the conditions (v) is not fulfilled, then we change to one pole positioning design (without zero cancellations) and the conditions over polynomials $(B+\tilde{B}_1)$ and $(B+\tilde{B}_2)$ will disappear.

∇∇∇

PROOF OF LEMMA 2

(i) Using the control law (41) in the process model (25) we get :

$$Ry(t+1)=RA^*y(t)+BTy_r(t)-SBy(t)+R(-\hat{g}(t)-g(t))$$

Grouping the polynomials in y(t), using the equality (31) and the definition (40) :

$$y(t+1)=[RA-q^{-1}SB]=BTy_r(t)+Re(t)$$

$$TBA_my(t)=q^{-1}BTy_r(t)+Re(t)$$

(ii) The proof is made as follows; Take the control law equations (41) and use the relation (42) in order to eliminate the output y(t), manipulate this equation until the control law can be expressed like a function of $y_r(t)$ and e(t) ;

$$Ru(t)=Ty_r(t)-Sy(t)-R\hat{\bar{g}}(t)$$

and using (42):

$$TBA_mRu(t)=TBA_mTy_r(t)-S[q^{-1}BTy_r(t)+Re(t)]-TBA_mR\hat{\bar{g}}(t)$$

$$TBA_mu(t)=TAy_r(t)-Se(t)-TA_m\hat{g}(t) \qquad \text{from (31)}$$

APPENDIX A

PROOF OF LEMMA 1

(i) From algebra manipulations of equations (25), (28), (30) and (31).

(ii) From algebra manipulations of equations (30), (32) and (31), we obtain:

$$A_m B u(t) = A y_r(t) - A_m g(t)$$

$$A_m B u(t) = A y_r(t) - A_m[g_1(t)h(t) + g_2(t)(1-h(t))]$$

$$A_m[B + \tilde{B}_1 h(t) + \tilde{B}_2(1-h(t))]u(t) =$$

using (26)

$$= A y_r(t) - A_m[(\tilde{A}_1^* y(t) + \delta_1)h(t) + (\tilde{A}_2^* y(t) + \delta_2)(1-h(t))]$$

using (32)

$$= [A - \tilde{A}_1{}^* q^{-1} h(t) - \tilde{A}_2{}^* q^{-1}(1-h(t))]y_r(t) - A_m \delta_1 h(t) - A_m \delta_2 (1-h(t))$$

(iii)

$$A_m[y(t) - y_r(t) + y_r(t)] = q^{-1} y_r(t)$$

$$A_m[y(t) - y_r(t)] = [-A_m + q^{-1}] y_r(t)$$

(iv)
For constant refence signals the tracking error ($y(t) - y_r(t)$) will converge exponentially to zero provided that A_m has its roots inside of the unit circle. This came from (iii).

(v)
The equation (33) can be rewritten like two disjoint sets of equations that united to equation (32) will give us the next augmented system:

$$\begin{bmatrix} (B+\tilde{B}_1)A_m & 0 & 0 \\ 0 & (B+\tilde{B}_2)A_m & 0 \\ 0 & 0 & A_m \end{bmatrix} \cdot \begin{Bmatrix} u_1(t) \\ u_2(t) \\ y(t) \end{Bmatrix} = \begin{bmatrix} (A-A_1{}^* q^{-1}) y_r(t) - A_m \delta_1 \\ (A-A_2{}^* q^{-1}) y_r(t) - A_m \delta_2 \\ q^{-1}\, y_r(t) \end{bmatrix}$$

adding and subtracting $TA_m g(t)$

$$TBA_m u(t)=TAy_r(t)-Se(t)+A_m Te(t+1)-TA_m g(t) \qquad \text{using (40)}$$

since $g_1(t)$ and $g_2(t)$ are disjoints sequences, we will manipulate on one set of these equations to simplify the analysis.

$$TBA_m u_1(t)=TAy_r(t)-Se(t)+TA_m e(t+1)-A_m T[\tilde{A}_1{}^* y(t)+\tilde{B}_1 u(t)+\delta_1]$$

multiply by B each side and use the diophantine equation :

$$TBA_m(B+\tilde{B}_1)u_1(t)=BTAy_r(t)+RAe(t+1)-TBA_m\tilde{A}_1{}^* y(t)-\delta_1 A_m T$$

using again (42):

$$TBA_m(B+\tilde{B}_1)u_1(t)=BTAy_r(t)+RAe(t+1)-\tilde{A}_1[q^{-1}BTy_r(t)+Re(t)]-\delta_1 A_m T$$

$$TBA_m(B+\tilde{B}_1)u_1(t)=[A-\tilde{A}_1{}^* q^{-1}]BTy_r(t)+[A-\tilde{A}_1{}^* q^{-1}]Rq^{-1}e(t)-\delta_1 A_m T$$

in similar way we can obtain for the set $g_2(t)$

$$TBA_m(B+\tilde{B}_2)u_2(t)=[A-\tilde{A}_2{}^* q^{-1}]BTy_r(t)+[A-\tilde{A}_2{}^* q^{-1}]Rq^{-1}e(t)-\delta_2 A_m T$$

and the control law is obtained by superposition of $u_1(t)$ and $u_2(t)$:

$$u(t) = u_1(t)h(t) + u_2(t)(1 - h(t))$$

∇∇∇

PROOF OF THEOREM 1

The proof of (44-46) are based on the key technical lemma given in (Goodwin, Ramadge and Caines, 1980), and using lemmas (Goodwin and Sin, 1984).

i)

To show that the prediction error e(t) goes to zero, we will first show that the sequences $\{u(t)\}$ and $\{y(t)\}$ grow not faster than linearly with e(t). Then using the equation given by the lemma 2, we can construct the next augmented system which has $e(t), y_r(t), \delta_1, \delta_2$ as inputs and $y(t)$, $u_1(t)$, $u_2(t)$ as outputs :

$$\begin{bmatrix} BTA_m & 0 & 0 \\ 0 & BTA_m(B+B_1) & 0 \\ 0 & 0 & BTA_m(B+B_2) \end{bmatrix} \cdot \begin{bmatrix} y(t) \\ u_1(t) \\ u_2(t) \end{bmatrix} =$$

$$\begin{bmatrix} q^{-1}BTy_r(t) + Re(t) \\ [A-\tilde{A}_1{}^{*}q^{-1}]BTy_r(t)+[A-\tilde{A}_1{}^{*}q^{-1}]Rq^{-1}e(t) \\ [A-\tilde{A}_2{}^{*}q^{-1}]BTyr(t)+[A-\tilde{A}_2{}^{*}q^{-1}]Rq^{-1}e(t) \end{bmatrix} - \begin{bmatrix} 0 \\ \delta_1 A_m \\ \delta_2 A_m T \end{bmatrix} \tag{A.1}$$

where $\{y_r(t)\}$ is a bounded sequence and δ_1, δ_2, are constant. This system has a characteristic polynomial equations given by $BTA_m(B+\tilde{B}_1)(B+\tilde{B}_2)$, en view of the assumptions A_1 and A_2 we can conclude that all the roots of this polynomial will be inside of the unit circle and hence, using lemmas (Goodwin and Sin 1984 (B.3.3)), there exist positive constants C1 and C2 that:

$$\|\phi(t)\| \le C_1 + C_2 \max_{1\le\tau\le N} |e(\tau)| \quad \forall \quad 1\le t\le N \tag{A.2}$$

Now using the properties of the RLS' algorithm ;

$$\lim_{t\to\infty} \frac{e(t)}{[1+K\phi(t-1)^T\phi(t-1)]^{1/2}} = 0 \quad ; \quad k=\lambda_{max}P(-1) \tag{A.3}$$

and the linear boundness of the regression vector by the estimation error, we can apply the technical lemma (Goodwin, Ramadge and Caines, 1980) and conclude (i).

(ii)

We argument as before and use the fact that e(t) --> 0 .Provided the sequences $\{y_r(t)\}, \{e(t)\}$ are bounded and the characteristic polynomial of (A.1) is a stable one, it follows that the sequence $\{y(t)\}$, $\{u1(t)\}$,$\{u2(t)\}$ and hence $\{\phi(t)\}$ will be bounded.

(iii)

Take the equation (42) and apply the limit to each side

$$\lim_{t\to\infty} TB(A_m y(t)-q^{-1}yr(t)=\lim_{t\to\infty} Re(t)=0 \qquad \text{using (44)}$$

hence

$$\lim_{t\to\infty} (Ay(t)-q^{-1}y_r(t))=0$$

and we obtain (iii).

APPENDIX B

SIMULATION USING SIMNON PACKAGE

Four main blocks of simulations have been implemented: the reference generator, the controller, the plant and a RLS estimation algorithm. Some flexibilities inside of each simulation blocks are available. The reference signal (periodic square wave) can be adjusted in period, amplitude (non symmetrical) and dc. component. Internal switches allow the choise of different controller schemes and/or process models. i.e. switch " sr " (switch regulator) selects between a pole placement design with or without zeros cancellation. Switch "sp" (s. process) selects between a first or second order continuous motor model. The nonlinear compensation g(t) can be calculated from fixed coefficient or estimated by the RLS algorithm, the selection is made with the switch "sa" (s. adaptation). Two sets of two macroinstructions are only needed in order to activate, simulate and display curves. "SIM" and "DIBUJA" simulate a discrete environment, "CSIM" and "CDIBUJA" use a continuous time process. The following picture describes these ideas.

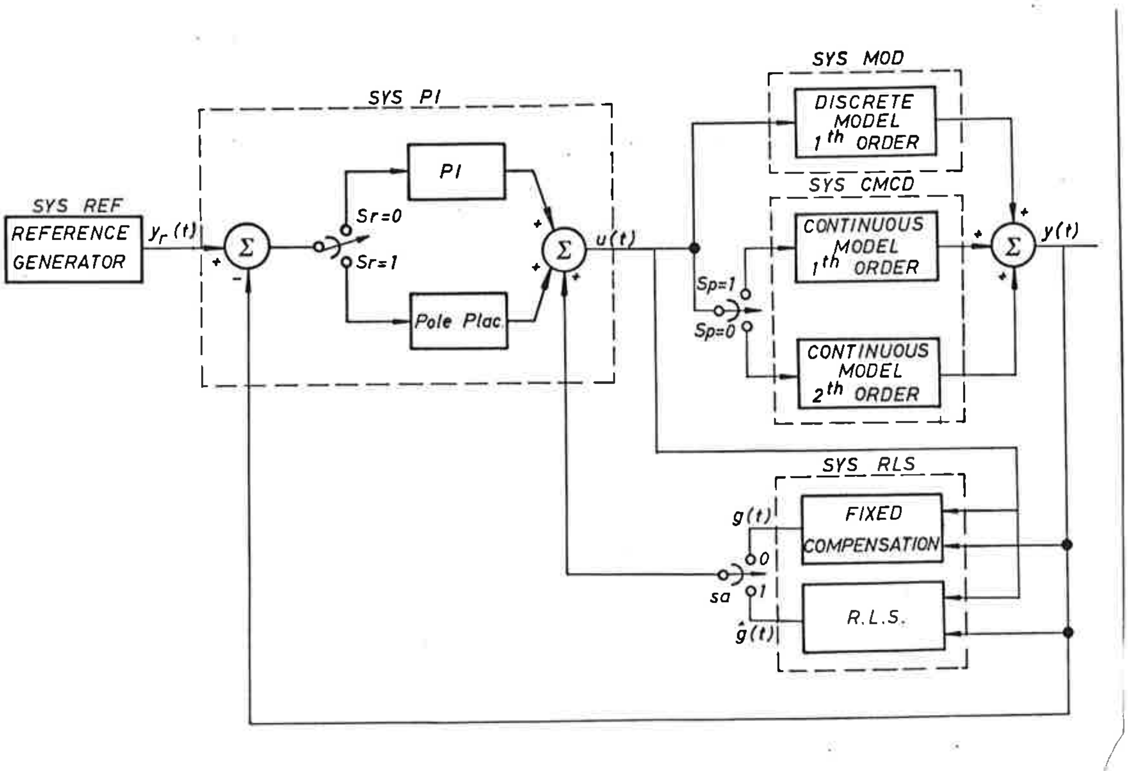

FIG. B1 Scheme of simulation blocks.

NUMERICAL VALUES USED IN THE EXAMPLES

	EXAMPLE 1	EXAMPLE 2
B	0.25	-same
A	$1-0.986q^{-1}$	-same
Cr	$(1-0.67q^{-1})^2$	-same
S	0.25	-same
R	$0.646-0.537q^{-1}$	-same
Ts	0.01	-same
d1	0.02	-same
d2	0.025	-same
k1	0.2	-same
k2	0.6	-same
J	0.02	-same

The rest of the numerical values are available in the simulation sheets.

```
continuous system proc
"Direct current motor with N.L. friction torque"
"sp= switch : 1= reduced model (1 order). 0=exact model (2 order).
""File is called Inproc"
"Author C.Canudas"

input v u
output y
state x1 x2 x3
der  dx1 dx2 dx3

k= if  y>0 then k1 else if  y<0 then k2 else 0
d= if  y>0 then d1 else if  y<0 then -d2 else 0



dx1=(-ra*x1 - kb*x2 +u)/la        "x1 = ceurrent
dx2=(ki*x1 -k*x2 - d + v)/j       "x2 = angular velocity
dx3=-(ki*kb+k*ra)*x3/(ra*j) + ki*u/(ra*j) -d/j
y = if sp> 0 then x3 else x2

j:.02
ra:.2
kb:.055
la:.001
ki:.05
k1:.2
k2:.6
d1:.02
d2:.025
sp:1

end
```

```
DISCRETE SYSTEM MOD
"D.C. motor with N.L. friction torque
"....Reduced model
"File called Dmodel

Input w u
Output y
State x
New nx
Time t
Tsamp ts

a1=1-h*(Kb*Ki+K1*ra)/(J*ra)
a2=1-h*(Kb*Ki+k2*ra)/(J*ra)
b0=Ki*h/(J*ra)
a=if y>0 then a1 else if y<0 then a2 else 0
d=if y>0 then -d1*h/J else if y<0 then d2*h/J else 0

nx=a*x+b0*u+d
y=x
ts=t+h

J:0.02
ra:0.2
Kb:0.055
Ki:0.05
K1:0.2
k2:0.6
d1:0.02
d2:0.025
h:0.01

End
```

```
DISCRETE SYSTEM RLS
"File called PAA
"Recursive least square.Two disjoint sets of parameters estimation

Input u y
Output g
State u1 y1
State f11 f12 f21 f22
State t11 t12 t21 t22
State p111 p112 p122
State p211 p212 p222
New nu1 ny1
New nf11 nf12 nf21 nf22
New nt11 nt12 nt21 nt22
New np111 np112 np122
New np211 np212 np222
Time t
Tsamp ts

"Switch gain and estimations error
h=if y>0 then 1 else 0
e=y - b0*u1 - a*y1 -t11*f11 -t12*f12 -t21*f21  -t22*f22

"Estimation gain
k11=p111*f11+p112*f12
k12=p112*f11+p122*f12
den1=lam+f11*k11+f12*k12

k21=p211*f21+p212*f22
k22=p212*f21+p222*f22
den2=lam+f21*k21+f22*k22

"Update estimates
nt11=t11+k11*e/den1
nt12=t12+k12*e/den1
nt21=t21+k21*e/den2
nt22=t22+k22*e/den2

"Compute output, N.L. compensation g=f*t
ga= ((nt11*h+nt21*(1-h))*y+(-nt12*h+nt22*(1-h)))/b0
gl=-((l11*h+l21*(1-h))*y+(l12 *h-l22 *(1-h)))/b0
g= if sa >0 then ga else gl

"Update covariance
np111=(p111-k11*k11/den1)*h/lam +p111*(1-h)
np112=p112-k11*k12*h/den1
np122=(p122-k12*k12/den1)*h/lam +p122*(1-h)

np211=(p211-k21*k21/den2)*(1-h)/lam +p211*h
np212=p212-k21*k22*(1-h)/den2
np222=(p222-k22*k22/den2)*(1-h)/lam +p222*h

"Update old data and f-vector
nu1=u
ny1=y
nf11=y*h
nf12=-h
nf21=y*(1-h)
nf22=1-h

ts=t+dt

"Parameters
dt:0.01
lam:.99
p111:1000
p122:1000
p211:1000
p222:1000
sa:1
a:0.9862     "Linear model coefficients
b0:.125
l11:.1       "Coefficients of fixed compensation g(t)
l21:.3
l12:.01
l22:.0125
```

```
DISCRETE SYSTEM PI
"PI controller with anti-windup and nonlinear compensation
"sr=0  PI controller with zeros added : TF=R(z)/Cr(z)
"sr=1  PI controller without zeros    : TF=  1 /Cr(z)
"File called PIREG

Input yr y g
Output u
State x1 x2
New  nx1 nx2
Time t
Tsamp ts

r0=(c1+a+1)
r1=(c2-a)
e=yr -y

"filtering the reference
nx1=-r1*x1/r0  +(1+c1+c2)*yr/r0
y1=x1
ef= if sr >0 then (y1-y) else (yr-y)

v=r0*ef/b0+x2
u1= if v < ulow then ulow else if v <uhigh then v else  uhigh
nx2=x2+(r0+r1)*ef/b0 +u1-v
u=u1-g
ts=t+h

h:.01
ulow:-10
uhigh:10
a:.9862
b0:.125
c1:-1.34     "Cr coefficients
c2:0.4489
sr:1

End
```

```
DISCRETE SYSTEM REF
"Symetric square wave generator with D.C. component
"File called GENREF

Output yr
Time t
Tsamp ts
w=h*per
s=sign(mod(t,w)-w/2)
amp = if s<0 then  amp2 else amp1
yr=amp+dc
ts=t+h

amp1:1    "high amplitude
amp2:-1   "low  amplitud
per:100   "number of samplin times by period
h:0.01    "samplin time
dc:0      "dc of the square wave

End
```

```
CONNECTING SYSTEM CSYS
"Connecting system for simulations PI+Adaptive N.L. compensation
"using a discrete model.
"File called CSYS

Time t
yr[PI]=yr[REF]
y[PI]=y[MOD]
g[PI]=g[RLS]
u[MOD]=u[PI]
y[RLS]=y[MOD]
u[RLS]=u[PI]
w[MOD]=0.0

End
```

```
MACRO csim
"Coopilation of the close loop system.
"Continuous time model.
"File called csim


syst PIREG PAA LNPROC GENREF CCSYS
End
```

```
MACRO CDIBUJA
"trace of curves
"Continuous time model.
"File called CDIBUJA

split 3 2
store y[proc] u[proc] yr[ref] t11 t12 t21 t22 e[rls]
simu 0 10
ashow y[proc] yr[ref]
ashow u[proc]
ashow e
ashow t11 t21
ashow t12 t22

End
```

```
CONNECTING SYSTEM ccsys
"Connecting system for simulations PI+Adaptative N.L. compensation
"Continuous time model
"File called ccsys

Time t
yr[PI]=yr[REF]
y[PI]=y[PROC]
g[PI]=g[RLS]
u[PROC]=u[PI]
y[RLS]=y[PROC]
u[RLS]=u[PI]
v[PROC]=0.0

End
```

```
MACRO sim
"Coopilation of the closed-loop system.
"File called sim


syst PIREG PAA DMODEL GENREF CSYS
End
```

```
MACRO dibuja
"discrete model case.
"trace of curves

split 3 2
store y[mod] u[mod] yr[ref] t11 t12 t21 t22  e[rls]
simu 0 10
ashow y[mod] yr[ref]
ashow u[mod]
ashow e
ashow t11 t21
ashow t12 t22

End
```