IFAC

International Federation of Automatic Control

SOFTWARE FOR COMPUTER CONTROL

Preprints of the 3rd IFAC / IFIP Symposium
Madrid, Spain
5 - 8 October 1982

Editors

G. Ferraté
Barcelona, Spain

E.A. Puente
Madrid, Spain.

PREPRINTS

CONTROL SOFTWARE ENGINEERING AND MANAGEMENT III

INDUSTRIAL APPLICATIONS I

INDUSTRIAL APPLICATIONS II

INDUSTRIAL APPLICATIONS III

INDUSTRIAL APPLICATIONS IV

ADDENDUM

STUDIES FOR THE APLICATION OF AN ADAPTIVE CONTROLLER TO HYDROTURBINE GENERATORS

F. R. Rubio, E.F. Camacho, J. Aracil, L.G. Franquelo and J.M.G. Provost

Department of Automatic and Electronics. E.T.S. Engineering Ind.
University of Sevilla. Spain.

Abstract. This paper describes some studies made towards the automatization of hydroturbine generators with microcomputers. The overall design will include an automata controlling the starting-up and shutting-down procedure as well as an self-tuning regulator for the speed control.

A self-tuning regulator based on the classical pole-assignment method is studied. The algorithm uses a fast procedure for solving the polynomial equation implicit to selfturner regulators. This procedure is very simple from a computational point of view as only applications of elementary transformations on a 2 x 2 polynomial matrix are needed.

The algorithm has been programmed on a Digital PDP 1103 computer and applied to some test problems.

Keywords. Adaptive control, microprocessors, process control.

INTRODUCTION

Self-tuning regulators are designed to operate in processes that exhibit either time varying or nonlinear dynamic or both. This is the case of hidroturbine speed regulators where non linearities in the valves and other systems parts are found. These systems also have gradual drifts and delay and what is more important, slow time changes in two of the main factors affecting the overall system dynamics, which are the water head (Elgerd,1982) and the Mw/Hz constant of the system (stiffness) depending on the governor and load characteristics (Sterling 1978). Classical analog regulators, mechanical and electronic, have been used to control the speed of hydroturbine generators. These regulators are tuned at installation and have to be retuned from time to time. Self-tuning regulators seem to be the answer to some of the problem mentioned.

Self-tuning regulators (Astrom ,1980) work basically like this: The process parameters are estimated on-line by a recursive least-squares estimation and these estimates are then used to calculate the regulator parameters.

Due to the parameter changes in the process, the estimation algorithm needs some adaptive mechanism to follow parameter changes. A way of introducing this mechanism is to use a forgetting factor in such a way that the algorithm gives more weight to the latest measurements and therefore allows it to follow parameter changes.

Choosing an appropiate forgetting factor is essential for the algorithm to work. If a large forgetting factor is chosen. The estimation algorithm has a very poor speed of adaptation to parameter changes in the model, on the other hand, if a small forgetting factor is chosen, the identification algorithm is extremely sensitive to disturbances and susceptible to numerical computational dificulties when "blow-up" problems in the covariance matrix arise.

The algorithm used in this paper works with an identification algorithm(Fortescue and coworkers,1981) with a variable forgetting factor that solves the problems mentioned before.

The adaptive algorithm used is described in next section, and is based on a control law equivalent to a linear feedback of the state vector and an observer. A method (Aracil,1974),based on elemental transformations on a 2 x 2 polynomial matrix is used to solve the polynomial equation found in the algorithm.

The amount of computation required for adaptive regulators, and therefore the sampling time, increases considerably with the order of the system considered. Working with a reduced model can sometimes improve the quality of the control as sampling time can be reduced, in this sense the paper presents some results obtained.

ADAPTIVE CONTROLLER.

The structure chosen for the adaptive controller is the one shown in fig. 1. This control structure corresponds to a linear state feedback and a Luenberger observer given certain conditions (Elliott and Wolovich,1979). As it is known this control law will arbitrarily assign all close-loop system poles preserving the open loop zeros.

Let us suppose that Td(z) is the desired close-loop transfer function. It can be seen from fig. 1 that

$$Td(z) = \frac{Y(z)}{V(z)} = \frac{N(z)\ G(z)}{D(z)G(z)+D(z)K(z)+N(z)H(z)} \qquad (1)$$

This equation can be transformed in

$$Td(z) = \frac{N(z)}{D(z) + F(z)} \qquad (2)$$

when the following equations hold

$$N(z)\ H(z) + D(z)\ K(z) = Y(z) \qquad (3)$$

$$Y(z) = F(z)\ G(z)$$

As degree G(z) = n-1, with n=degree D(z) , the degrees of K(z) and H(z) should be equal or smaller than n-1 in order to obtain a realisable controller.

To satisfy equation (3) polynomials K(z) and H(z) can be obtained applying the following algorithm (Aracil,1974).

1. Using Euclid's algorithm calculate P(z) and Q(z) such that

   $$N(z)\ P(z) + D(z)\ Q(z) = 1 \qquad (4)$$

2. Form the polynomial matrix,

   $$J(z) = \begin{bmatrix} j_{11} & j_{12} \\ j_{21} & j_{22} \end{bmatrix} = \begin{bmatrix} N(z) & -P(z)\ \psi\ (z) \\ D(z) & Q(z)\ \psi\ (z) \end{bmatrix} \qquad (5)$$

3. Reduce the degree of j_{12} and j_{22} adding column 1 multiplied by a constant to column 2 until it is no longer possible to decrease both or either of them.

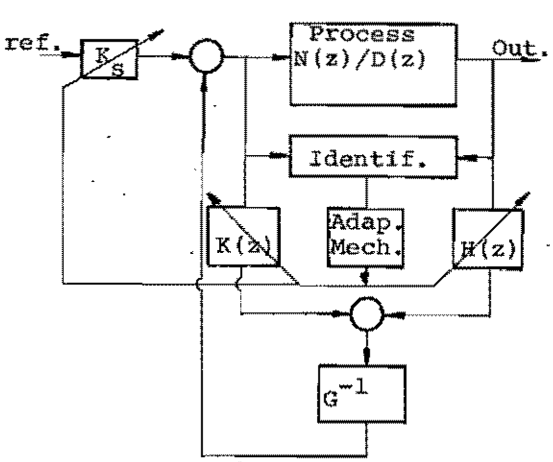

Figure 1. Structure of the Adaptive Controller.

Notice that 1 is always possible if N(z) and D(z) are prime. As transformations describedin 3 do not change the determinant of J(z) we have,

$$\det J(z) = N(z)j_{22}(z)-D(z)j_{11}(z)=\psi(z) \qquad (6)$$

as P(z) and Q(z) satisfy (4).

Therefore the polynomials K(z) and H(z) are respectively $j_{22}(z)$ an $-j_{11}(z)$.

IDENTIFICATION ALGORITHM

Most of the success of an adaptive controller depends on how well the identifier behaves. It is well known that the forgetting factor has great influence in the behaviour of the identifier and serious problems can be found when a constant forgetting factor is used, especially in steady-state regulation where old information is continually forgotten and very little dynamic information is coming in from the process which may in turn lead to an ex ponential growth of the covariance matrix making the identifier very sensitive to noise.

There are various ways of avoiding the problems mentioned before. Some of them act directly on the covariance matrix making sure that it is always bounded. Other methods consist of using a variable forgetting factor which in turn may be changed by different methods.

We have used a variable forgetting factor which is updated with an algorithm (Fortescue, 1981) which takes into account the following fact: the error between the output of the process and the one of the estimated system will tell us something about the behaviour of the identifier. A small error can be obtained if either: the process has not been excited; the parameter values are nearly correct or the identifier is sensitive enough to reduce this error. When we have a change in the process parameter, the process and identifier will give a different output and the error will be large.

In the first case a reasonable strategy would be to retain as much information as possible by choosing a forgetting factor close to one. In the second case the algorithm should forget the measurements corresponding to the "old" process. This can be accomplished by choosing a low forgetting factor. A measurement of the information content of the filter can be defined as the weighted sum of the squares of the errors, which can be expressed in a recursive way as:

$$\Sigma\ (t)=\lambda(t)\Sigma(t-1)+\ 1-\phi(t-k-1)^{T}K(t)\varepsilon(t)^{2}$$

where $\lambda(t)$ is the forgetting factor, $K(t)$ is the gain of the filter and $\phi^{T}(t)$ contains the last n+1 measurements of the input and output of the process.

The forgetting factor is then calculated in such a way that the estimation is based on the same amount of information. That is, keeping $\Sigma(t)$ constant.

It can be seen that this is accomplished by:

$$\lambda\ (t) = 1 - 1/N(t)$$

where

$$N(t) = \Sigma_{0}/\left[1-\phi(t-k-1)^{T}K(t)\right]\ \varepsilon(t)^{2}$$

the identification algorithm can be expressed as :

1. $\hat{y}(t) = \phi(t-k-1)^{T}\hat{\theta}(t-1)$
2. $\varepsilon(t) = y(t)-\hat{y}(t)$
3. $K(t)=P(t-1)\phi(t-k-1)/\left[1+\phi(t-k-1)^{T} P(t-1)\ \phi(t-k-1)\right]$
4. $\hat{\theta}(t) =\hat{\theta}(t-1) + K(t)\ \varepsilon(t)$
5. $\lambda(t) = 1-\left[1-\ \phi(t-k-1)^{T}\ K(t)\right]\varepsilon(t)^{2}/\ \Sigma_{0}$

   if $\lambda(t)<\lambda_{min}$ set $\lambda(t)=\lambda\,min$
6. $P(t) = \left[1-K(t)\ \phi(t-k-1)^{T}\right]P(t-1)/\ \lambda(t)$

IMPLEMENTATION OF THE ADAPTIVE CONTROLLER.

Two systems based on microprocessors, have been used for the implementation of the adaptive controller. The first one (center) is dedicated to the computation of the algorithm described in the previous sections; the second one (peripheral) is dedicated to actuate on the process with the set point received from the center and to scan the process output and send it to the center. This structure can be seen in fig. 2.

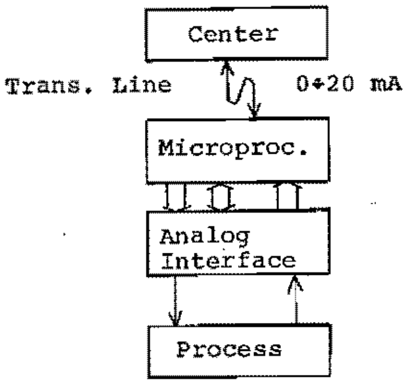

Figure 2. System Configuration

The main computer is liberated from the tasks of controlling the analog input output interface. This structure allows the system to be used for the control of a distributed process, connecting to the central computer various peripheral computers through series transmission lines.

The communication protocol is of the question from the center-answer from the peripheral type in such a way that the peripheral will be quiet until an order has been sent. A time out has been used in order to avoid that both computers remain simultaneously in a waiting state.

The peripheral station is normally waiting for a byte from the center. When the byte is received, it puts it through one of the output channels, reading the process output and sending it to the center.

An analog computer and an electromechanical system have been connected to the system to test the behaviour of the algorithm with various type of processes. The results obtained are described in next section.

APPLICATION TO SOME TEST PROBLEMS.

The adaptive controller described above has been applied to the control of some test problems. The first process chosen has been a second order system simulated in the analog computer with a transfer function

$$G(s) = \frac{4}{s^2+3s+12}$$

The system has been controlled, with the algorithm described in the previous sections, setting the poles of the desired transfer function to obtain a system without oscilations and a rising time of 1.5 seconds. The gain K_s of fig. 1 is calculated at each step in the algorithm to obtain a desired static gain.

Fig. 3 shows the behaviour of the system with a sampling time of 0.5 seconds and using a second order model for the identifier. It can be seen how the adaptation mechanism works in the starting up procedure and when a sudden change is produced in the process transfer function.

Table I shows the mean square error between the process output and the theoretical one for different values of the sampling time and using a first and a second order model for the controller.

TABLE 1. MEAN SQUARE ERROR (MSQ)

sampling time	1st order	2nd order
1.5 sec	0.255	0.314
1 sec	0.242	0.217
0.5 sec	0.192	0.139
0.25 sec	0.123	-

As it can be seen in table 1 the MSQ decreases with the sampling time and the results obtained using a first order model for the identifier are better than the ones obtained with a second order model if the sampling time is reduced. This can be helpful when implementing adaptive controllers with microprocessors, as the algorithm described is very simple in the case of a first order model.

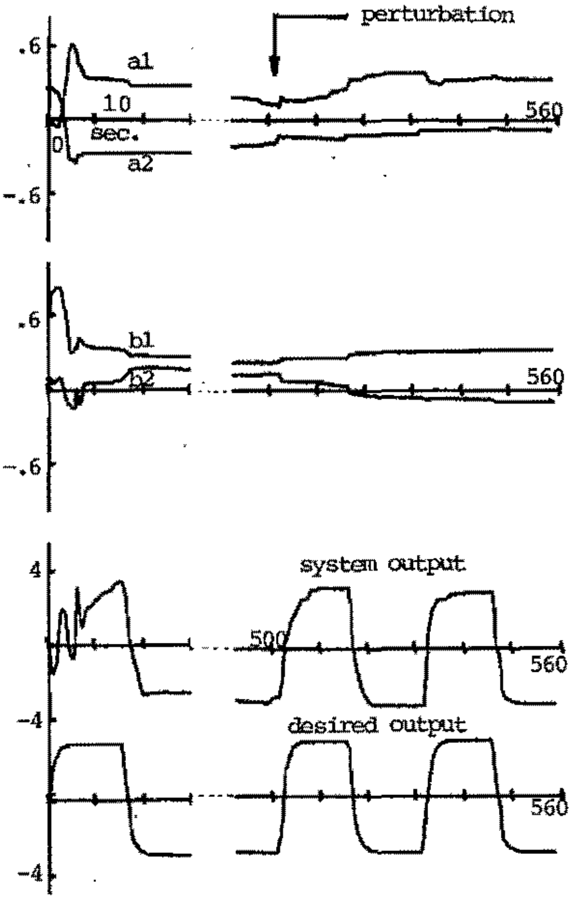

Figure 3. Response of a second order system.

Fig. 4 shows the process output for a first and second order model with the sampling time of 0.5 and 1 second respectively.

One of the main problems with adaptive controllers is the oscillatory behaviour during the starting-up period. This behaviour is due to the fact that in those moments the identifier is working with very little information and the process parameters, and therefore the control polynomials (K(z) and H(z)) are subject to sudden changes. One way of avoiding this problem is not allowing sudden changes of the parameter of the control polynomials. This can be accomplished by:

$$P_N = P_0 + \sigma (P_N - P_0)$$

where P_N is the new set of parameters, P_0 is the old set and σ is a constant that can take values between 0 (no adaptation) and 1. The value of σ can be calculated as a function of the covariance of the estimates.

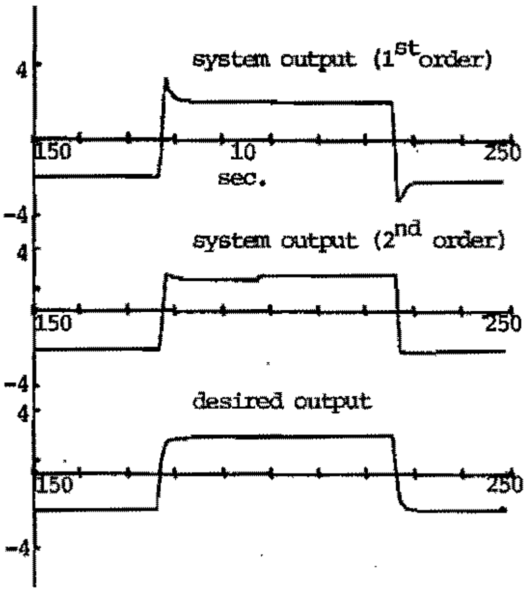

Figure 4. Comparison of a 1st and 2nd order controller

A suitable function can be:

$$\sigma = \exp(-\sigma_0\ \mathrm{trace}(\mathrm{covariance}))$$

Fig. 5 shows the output of the system with a sampling time of 0.5 seconds when this σ mechanism is introduced. As it can be seen, the starting up procedure is improved.

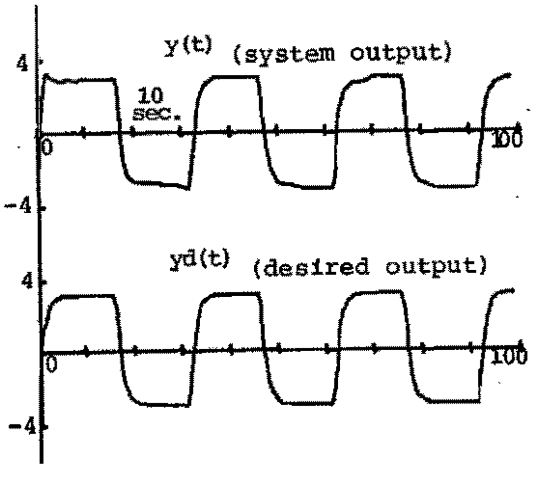

Figure 5. Modified adaptation mechanism

A process consisting of a DC motor, mechanical gear, servoamplifier and thaco has been chosen as a second example. The main characteristic of this system is its non linearities due to a dead zone produced by static frictions.

A second and first order model have been used to control the speed of this system. A square wave signal has been used as reference for the speed. The output of the system and some of the estimated parameters are shown in fig. 6. The sudden changes of the parameter correspond to the crossing of the dead zone by the process.

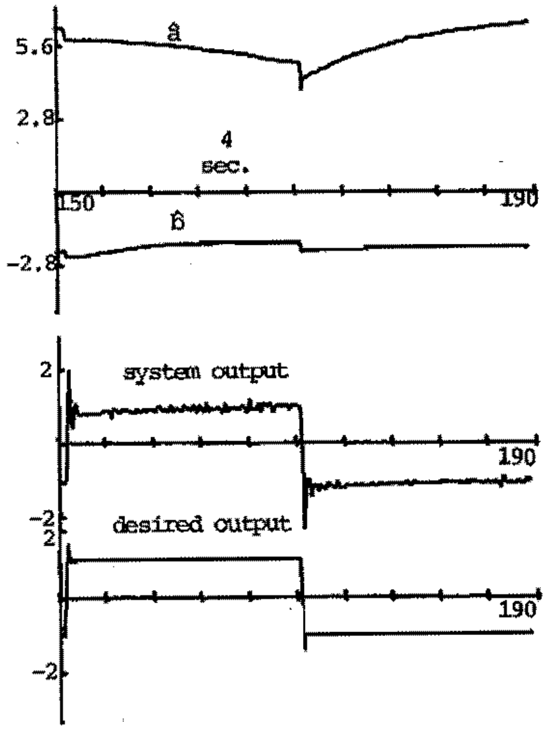

Figure 6. Speed control of a DC motor

As a third example we have chosen a model of a hydroturbine generator and the hydraulic amplifier. This process responds to a step change in the valve position with a momentary power decrease and can be modelled with a third order nonminimum phase system. The results obtained using a first and second order controller are shown in fig. 7.

As it can be seen the algorithm cannot control the system with a first order model. This is due to the fact that the identifier gets the parameters corresponding to the momentary drop of the output when a step change is produced in the input.

As the algorithm used does not change the zeros of the process, the overall system is a nonminimum phase and this behaviour can be seen in fig. 7. Note that the zero can not be cancelled as it is outside the unit circle. This problem is solved in practice by a double feedback of speed an valve position.

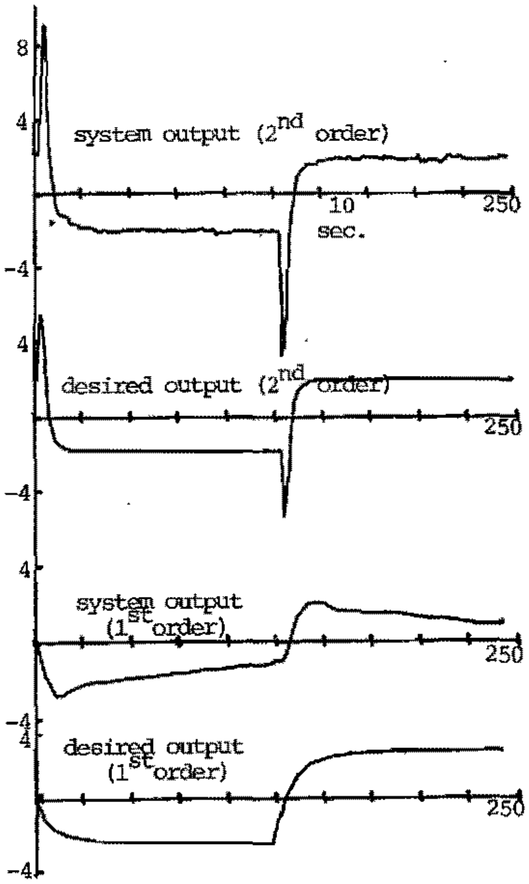

Figure 7. Adaptive control of a nonminimun phase system.

An automata for controlling the starting-up and shutting down procedure of this process is also being designed. This automata is being implemented with a Motorola 6800 microprocessor. More than a hundred digital input-output signals must be handled.

The automata should transfer control to the adaptive regulator once the turbine is started.

CONCLUSIONS

The paper presents some studies made with adaptive controllers. An algorithm using Euclid's algorithm and elementary transformation on a 2 x 2 polynomial matrix have been used to solve the polynomial equation implicit to pole placement adaptive controller. The algorithm has been tested on several systems using first and second order models. It can be seen in two of the examples chosen that the results obtained with a first order model with smaller sampling time are similar to those obtained with a second order model. This fact can be of interest when implementing adaptive controllers with microprocessors.

A smoother way of adaptation to solve some of the problems of the starting up period of an adaptive controller has been used and we feel that this heuristic aproach should work well in practice.

REFERENCES

Aracil, J. (1974). A Note on the Design of S.I.S.O. Systems. Revue R. A.I.R.O. N°Avril, pp. 119-125.

Astrom, K.J. (1980). Self-tuning Regulators Design Principles and Applications. Applications of Adaptive Control. Ed. Academic Press. Inc. pp. 2-68.

Eldger, O.I. (1982). Electric Energy Systems Teory. Mc Graw Hill.

Elliott,H. and W.A. Wolovich. (1979). Parameter Adaptive Identification and Control.IEEE trans. on Automatic Control. Vol AC-24 N°4. pp.592-599.

Fortescue, T.R.,L.S. Kershenbaum and B.E. Ydstie. (1981). Implementation of Self-tuning Regulators with Variable Forgetting Factors. Automatica. Vol 17. N°6. pp. 831-835.

Sterling, M.J.H. (1978). Power System Control. Ed. IEE.