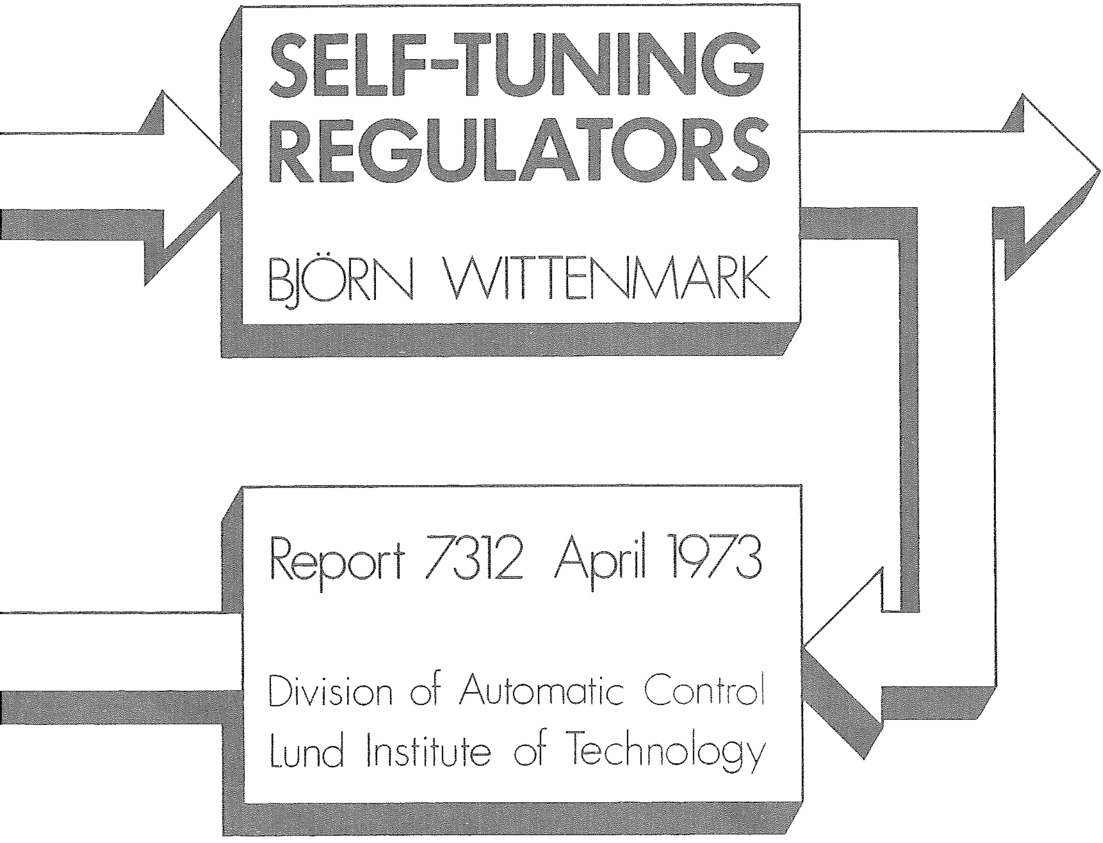
Self-Tuning Regulators

Wittenmark, Björn

1973



SELF-TUNING REGULATORS

BJÖRN WITTENMARK

Report 7312 April 1973

Division of Automatic Control
Lund Institute of Technology

To Karin,

with thanks for her encouragement and support.

SELF-TUNING REGULATORS.

During the industrial revolution clever engineers heuristically designed regulators which could bring the output of a system back to a reference value after a disturbance. These regulators which are nowadays called PI-regulators (Proportional and Integral regulators) can be described mathematically by the equation

$$u(t) = K\left(e(t) + \frac{1}{T}\int^{t} e(s)ds\right)$$

where u is the control signal and e is the difference between the desired and the actual output of the system. The PI-regulator has the property that the stationary error after a step disturbance is equal to zero irrespective of the system to be controlled if the closed loop system is stable. The first complete analysis of the PI-regulator was done by J.C. Maxwell [5] about 80 years after their construction. Maxwell explained mathematically the importance of the integral term. The PI-regulators are the main tools of control engineering practice. The regulators have been implemented in many different ways using analog technique and mechanical, pneumatic, hydraulic and electronic technology. In the last 10 - 15 years PI-regulators have also been implemented digitally on computers. The ways of implementing the regulators have been different but the basic idea has always been the same.

The PI-regulators are robust and are sufficiently good in many cases. It is also fairly easy to tune the regulator, i.e. to choose its parameters (K and T). The simple PI-regulator is, however, not suitable in all cases and more sophisticated regulators have been developed. More complex regulators are for instance needed if the process contains dead-time. The computer technology has given new possibilities to implement regulators. Much more complex regulators can be implemented on a computer than if analog technique is used. The growing complexity of the controllers has made the tuning more difficult. The tuning problem is one reason why complex regulator seldom are used.

The tuning problem has two aspects. The parameters of the regulator must be chosen at the start-up. The regulator must also be retunedd occasionally if the process dynamics are changing. This has led to the idea of adaptive controllers. The word adaptive implies that the regulator has the ability to change its parameters when the process dynamics or the characteristics of the disturbances are changing. Adaptive controllers have so far not been implemented to any greater extent.

If the parameters of the process are modelled as stochastic processes it is possible to formulate the adaptive control problem within the framework of stochastic optimal control theory. It is, however, difficult to get a practical solution to the problem since the computational requirements are such that only the simplest cases can be solved numerically even with the largest computers available today [1]. The solution has, however, a very interesting structure. Feldbaum [4] has pointed out that these control strategies have a twofold goal. The controller must obtain good estimates of the unknown parameters of the process, i.e. the controller must perform real-time identification. Secondly the controller must make as good control as possible based on the available information. These two goals are contradictory and there must be a compromize between the identification and control activities of the controller. Controllers with these properties are called dual controllers.

One interesting simplification of the adaptive control problem is obtained if it is assumed that the process to be controlled has constant but unknown parameters. Controllers for this type of processes will be called self-tuning regulators. This thesis treats one type of self-tuning regulators and compromise of the following three parts:

I. On Self-Tuning Regulators, Automatica, 9, No. 2, March, 1973, pp 185 - 199, Coauthor K.J. Åström,

II. A Self-Tuning Regulator, Report 7311, Division of Automatic Control, Lund Institute of Technology,

III. An Industrial Application of a Self-Tuning Regulator, Report 7310, Division of Automatic Control, Lund Institute of Technology, Coauthor U. Borisson.

The thesis covers the steps from analysis to practical implementation of a self-tuning regulator. The regulator is derived to control single-input single-output systems which are minimum phase. One early version was presented by Wieslander and Wittenmark at an IFAC Symposium in Prague 1970 [9]. At the same symposium Peterka presented a similar algorithm [6]. Part I of this thesis was presented at the IFAC World Congress in Paris 1972. Saridis and Dao [8] consider the self-tuning problem when the unknown parameters belong to a known discrete set of parameter values. Åström has in [2] given a self-tuning algorithm which can be used for nonmimimum phase systems. A more complex self-tuning regulator which can handle multiple-input multiple-output systems is given by Peterka and Åström [7].

Part I introduces the basic idea behind the considered class of self-tuning regulators based on the natural assumption of separation of identification and control. Some asymptotic results are given. For instance it is shown that if the algorithm converges then the controller under weak assumptions will converge to the minimum variance regulator that can be obtained if the parameters are known.

Part II is a detailed discussion of the properties of the self-tuning regulator. The results are partly obtained through analysis and partly through extensive simulations using an interactive computer program. The simulations have thus been used to gain insight into the properties of the algorithm and to generate hypotheses, some of which were later proved mathematically. The investigation in Part II shall thus be regarded as being theoretical as much as experimental.

Part III relates some of the results obtained from an implementation of a self-tuning regulator on an industrial process.

The self-tuning regulator discussed can be thought of as performing two tasks. The parameters in a model of the process are estimated using the method of least squares and a minimum variance regulator is computed based on the estimated parameters. The steps are repeated in each sampling interval. The computations to be done in each step are very small. It is interesting to compare the complexities of the PI-regulator and the self-tuning regulator. The number of operations (additions or multiplications) in a digital PI-regulator is about 4. The self-tuning regulator with two parameters requires about 34 operations. The number of operations for the self-tuning regulator is increasing with the square of the number of parameters. Experiments and simulations indicate that up to 6 parameters is a reasonable number of parameters in a self-tuning regulator. This means that the information processing is an order of magnitude larger when using a self-tuning regulator. A better performance of the system can, however, compensate for the more complex computations.

It is relatively easy to characterize the asymptotic properties of the self-tuning algorithm. If the parameter estimates converge then

$$\lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} y(t+\tau)y(t) = 0$$

and

$$\lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} y(t+\tau)u(t) = 0$$

for certain values of τ. This is true for a large class of processes. The self-tuning regulator can thus be compared with a PI-regulator. The PI-regulator will make the control error equal to zero irrespective of the properties of the process as long as the closed loop system is stable. Analogously the self-tuning regulator will make mean values of sums of squares of outputs and inputs equal to zero. The self-tuning regulator will converge to a generalized dead-beat regulator if the changes in the reference signal is dominating and to a minimum variance regulator if the noise is dominating.

The asymptotic properties discussed in Part I are true under the assumption that the parameter estimates really converge. The transient properties of the self-tuning algorithm are difficult to analyze. In Part II the convergence is discussed for some special cases. The algorithm does not converge for all linear single-input single-output systems with constant parameters. This can be shown by counter-examples.

Some parameters must be specified when using the self-tuning regulator. These are a scale factor for the control variables, the number of time-delays in the system, the number of parameters in the regulator, the initial values of the estimates and their covariances. The choice of these parameters is discussed in Part II. Experience has shown that it is fairly easy to make the proper choice in practice. These parameters are also much easier to choose than to directly determine the coefficients of a complex control law.

The self-tuning regulator presented has been used on different industrial processes. An example from the mining industry is given in [3]. Part III of this thesis gives experiences of one implementation in the paper industry. The process under consideration is moisture content control on a paper machine at the Gruvön mill of Billerud AB, Grums, Sweden. The experiences of the implementation are very good concerning the transient as well as the stationary behaviour of the regulator. The experiments at Gruvön show that it can be easy to implement a self-tuning regulator on a process which already is controlled by a process computer. The engineers at the mill could take over the operation of the self-tuning regulator after a short training period only.

The self-tuning regulator discussed has many attractive properties. It can be used for a large class of linear single-input single-output systems. The algorithm can be used for systems which have slowly time-varying parameters. It has turned out that even nonminimum phase systems can sometimes be handled. Finally the algorithm is easy to implement on a computer and to install on a real process. Among the drawbacks are the fact that the algorithm does not always converge. Further it is not possible to automatically handle nonminimum phase systems which is possible with the algorithm described in [2]. The presented algorithm can only handle single-input single-output systems. Multivariable systems can, however, be handled through cascade control. It is also believed that the results can be extended to certain multivariable systems.

It is the belief of the author of this thesis that the presented self-tuning regulator is a valuable tool for control engineers. It should be pointed out that the self-tuning regulator is not a substitute for but a good complement to the conventional PI-regulators.

The thesis has been possible to accomplish thanks to many persons in my environment. First of all I want to express my sincere thanks to my advisor Karl Johan Åström whose ideas, help and enthusiasm have guided and encouraged me throughout the work.

My collegues at the Division I want to thank for hours of discussions. In particular I want to thank Lennart Ljung who has helped me to understand the convergence properties of the self-tuning regulators and Ulf Borisson with whom I have shared hours of disbelief and success at the paper mill. Further I want to express my gratitude to Ivar Gustavsson and Per Hagander who always have time to discuss larger and minor problems.

The experiments with the self-tuning regulator at the paper mill have been possible to make thanks to financial support from STU, contract No 72-1026/U833 and to the kindness of Billerud AB. A special thank to Olle Alsholm, Leif Endresen and Odd Stavnes at the Gruvön Mill at Grums for their help and support during the experiments.

Thanks also to Gudrun Christensen and Kerstin Palmqvist who have typed the manuscripts and to Birgitta Tell and Bengt Lander who have made the drawings.

My girls Karin, Emma, Ida and Johanna have spent many evenings and week-ends without husband and father. I want to thank them for their encouragement, support and understanding.

REFERENCES.

[1] Åström, K.J.: Optimal Control of Markov Processes with Incomplete State Information, Journal of Mathematical Analysis and Applications, 10, No. 1, Febr., 1965, pp. 174 - 205.

[2] Åström, K.J.: A Self-Tuning Algorithm for Nonminimum Phase Systems, to appear, Division of Automatic Control, Lund Institute of Technology.

[3] Borisson, U.: Control of an Ore Crusher - A Feasibility Study of Self-Tuning Control and Remote Processing, to appear, Division of Automatic Control, Lund Institute of Technology.

[4] Feldbaum, A.A.: Dual Control Theory I-IV, Automation and Remote Control, 21, 1960, pp. 874 - 880, pp. 1033 - 1039, 22, 1961, pp. 1 - 12, pp. 109 - 121.

[5] Maxwell, J.C.: On Governors, Proceedings of the Royal Society of London, 16, 1868, pp. 270 - 283.

[6] Peterka, V.: Adaptive Digital Regulation of Noisy Systems, IFAC Symposium on Identification and Process Parameter Estimation, Prague, 1970, Paper No. 6.2.

[7] Peterka, V., Åström, K.J.: Control of Multivariable Systems with Unknown but Constant Parameters, Preprint Third IFAC Symposium on Identification and System Parameter Estimation, Hague, 1973.

[8] Saridis, G.N., Dao, T.K.: A Learning Approach to the Parameter Adaptive Self-Organizing Control Problem, Automatica, 8, 1972, pp. 589 - 597.

[9] Wieslander, J., Wittenmark, B.: An Approach to Adaptive Control Using Real Time Identification, Automatica, 7, 1971, pp. 211 - 217.

Automatica, Vol. 9, pp. 185–199. Pergamon Press, 1973. Printed in Great Britain.

On Self Tuning Regulators*

Sur les Régulateurs Auto-Syntonisants

Über selbsteinstellende Regler

О самонастраивающихся регуляторах

K. J. ÅSTRÖM and B. WITTENMARK

Control laws obtained by combining a least squares parameter estimator and a minimum variance strategy based on the estimated parameters have asymptotically optimal performance.

Summary—The problem of controlling a system with constant but unknown parameters is considered. The analysis is restricted to discrete time single-input single-output systems. An algorithm obtained by combining a least squares estimator with a minimum variance regulator computed from the estimated model is analysed. The main results are two theorems which characterize the closed loop system obtained under the assumption that the parameter estimates converge. The first theorem states that certain covariances of the output and certain cross-covariances of the control variable and the output will vanish under weak assumptions on the system to be controlled. In the second theorem it is assumed that the system to be controlled is a general linear stochastic *n*th order system. It is shown that if the parameter estimates converge the control law obtained is in fact the minimum variance control law that could be computed if the parameters of the system were known. This is somewhat surprising since the least squares estimate is biased. Some practical implications of the results are discussed. In particular it is shown that the algorithm can be feasibly implemented on a small process computer.

1. INTRODUCTION

IT HAS been shown in several cases that linear stochastic control theory can be used successfully to design regulators for the steady state control of industrial processes. See Ref. [1]. To use this theory it is necessary to have mathematical models of the system dynamics and of the disturbances. In practice it is thus necessary to go through the steps of plant experiments, parameter estimation, computation of control strategies and implementation. This procedure can be quite time consuming in particular if the computations are made off-line. It might also be necessary to repeat the procedure if the system dynamics or the characteristics of the disturbances are changing as is often the case for industrial processes.

From a practical point of view it is thus meaningful to consider the control of systems with constant but unknown parameters. Optimal control problems for such systems can be formulated and solved using non-linear stochastic control theory. The solutions obtained are extremely impractical since even very simple problems will require computations far exceeding the capabilities of todays computers. For systems with constant but unknown parameters it thus seems reasonable to look for strategies that will converge to the optimal strategies that could be derived if the system characteristics were known. Such algorithms will be called *self-tuning* or *self-adjusting* strategies. The word adaptive is not used since adaptive, although never rigorously defined, usually implies that the characteristics of the process are changing. The problem to be discussed is thus simpler than the adaptive problem in the sense that the system to be controlled is assumed to have constant parameters.

The purpose of the paper is to analyse one class of self-adjusting regulators. The analysis is restricted to single-input single-output systems. It is assumed that the disturbances can be characterized as filtered white noise. The criterion considered is the minimization of the variance of the output. The algorithms analysed are those obtained on the basis of a separation of identification and control. To obtain a simple algorithm the identification is simply a least squares parameter estimator.

The main result is a characterization of the closed loop systems obtained when the algorithm is applied to a general class of systems. It is shown in Theorem 5.1 that if the parameter estimates converge the closed loop loop system obtained will be such that certain covariances of the inputs and the outputs of the closed loop system are zero. This is shown under weak assumptions on the system to be

* Received 2 March 1972; revised 12 September 1972. The original version of this paper was presented at the 5th IFAC Congress which was held in Paris, France during June 1972. It was recommended for publication in revised form by Associate Editor A. Sage.

controlled. Moreover if it is assumed that the system to be controlled is a sampled finite dimensional linear stochastic system with a time delay in the control signal it is demonstrated in Theorem 5.2 that, if the parameter estimates converge, the corresponding regulator will actually converge to the minimum variance regulator. This is true, in spite of the fact that the least squares estimate is biased.

The major assumptions are that the system is minimum phase, that the time delay is known and that a bound can be given to the order of the system. The first two assumptions can be removed at the price of a more complicated algorithm.

The paper is organized as follows: sections 2 and 3 provide background and a motivation. The algorithm is given in section 4. Control strategies for systems with known parameters are given in section 2. Least squares parameter estimation is briefly reviewed in section 3. Some aspects on the notion of identifiability is also given in section 3. The algorithm presented in section 4 is obtained simply by fitting the parameters of a least squares structure as was described in section 3 and computing the corresponding minimum variance control strategy as was described in section 2. The possible difficulty with non-identifiability due to the feedback is avoided by fixing one parameter.

The main result is given as two theorems in section 5. We have not yet been able to prove that the algorithm converges in general. In section 6 it is, however, shown that a modified version of the algorithm converges for a first order system. The convergence properties of the algorithm are further illustrated by the examples in section 7. Some practical aspects of the algorithm as well as some problems which remain to be solved are given in section 8. In particular it is shown that the algorithm is easily implemented on a minicomputer.

2. MINIMUM VARIANCE CONTROL

This section gives the minimum variance strategy for a system with known parameters. Consider a system described by

$$\begin{aligned}&y(t)+a_1y(t-1)+\ldots+a_ny(t-n)=b_1u(t-k-1)\\&\quad+\ldots+b_nu(t-k-n)+\lambda[e(t)+c_1e(t-1)\\&\quad+\ldots+c_ne(t-n)],\\&t=0,\pm1,\pm2,\ldots\end{aligned}\tag{2.1}$$

where u is the control variable, y is the output and $\{e(t), t=0, \pm1, \pm2, \ldots\}$ is a sequence of independent normal (0, 1) random variables. If the forward shift operator q, defined by

$$qy(t)=y(t+1)$$

and the polynomials

$$A(z)=z^n+a_1z^{n-1}+\ldots+a_n$$

$$B(z)=b_1z^{n-1}+\ldots+b_n,\ b_1\neq0$$

$$C(z)=z^n+c_1z^{n-1}+\ldots+c_n$$

are introduced, the equation (2.1) describing the system can be written in the following compact form:

$$A(q)y(t)=B(q)u(t-k)+\lambda C(q)e(t).\tag{2.2}$$

It is well known that (2.1) or (2.2) is a canonical representation of a sampled finite dimensional single-input single-output dynamical system with time delays in the output and disturbances that are gaussian random processes with rational spectral densities.

The model (2.1) also admits a time delay τ in the system input which need not be a multiple of the sampling interval. The number k corresponds to the integral part of τ/h, where h is the sampling interval.

Let the criterion be

$$V_1=Ey^2(t)\tag{2.3}$$

or

$$V_2=E\frac{1}{N}\sum_1^N y^2(t).\tag{2.4}$$

The optimal strategy is then

$$u(t)=-\frac{q^kG(q)}{B(q)F(q)}y(t)\tag{2.5}$$

where F and G are polynomials

$$F(z)=z^k+f_1z^{k-1}+\ldots+f_k\tag{2.6}$$

$$G(z)=g_0z^{n-1}+g_1z^{n-2}+\ldots+g_{n-1}\tag{2.7}$$

determined from the identity

$$q^kC(q)=A(q)F(q)+G(q).\tag{2.8}$$

Proofs of these statements are given in [2]. The following conditions are necessary:

- —The polynomial B has all zeroes inside the unit circle. Thus the system (2.1) is minimum phase.
- —The polynomial C has all zeroes inside the unit circle.

These conditions are discussed at length in [1]. Let it suffice to mention here that if the system (2.1) is non-minimum phase the control strategy (2.5) will still be a minimum variance strategy. This strategy will, however, be so sensitive that the slightest variation in the parameters will create an unstable closed loop system. Suboptimal strategies which are less sensitive to parameter variations are also well known. This paper will, however, be limited to minimum phase systems.

3. PARAMETER ESTIMATION

For a system described by (2.1) it is thus straight forward to obtain the minimum variance regulator, if the parameters of the model are known. If the parameters are not known it might be a possibility to try to determine the parameters of (2.1) using some identification scheme and then use the control law (2.5) with the true parameters substituted by their estimates. A suitable identification algorıthm is the maximum likelihood method which will give unbiased estimates of the coefficients of the A, B and C polynomials. The maximum likelihood estimates of the parameters of (2.1) are, however, strongly non-linear functions of the inputs and the outputs. Since finite dimensional sufficient statistics are not known it is not possible to compute the maximum likelihood estimate of the parameters of (2.1) recursively as the process develops. Simpler identification schemes are therefore considered.

The least squares structure

The problem of determining the parameters of the model (2.1) is significantly simplified if it is assumed that $c_i=0$ for $i=1, 2, \dots, n$. The model is then given by

$$A(q)y(t)=B(q)u(t-k)+\lambda e(t+n). \qquad (3.1)$$

The parameters of this model can be determined simply by the least squares method [3]. The model (3.1) is therefore referred to as a *least squares model.*

The least squares estimate has several attractive properties. It can easily be evaluated recursively. The estimator can be modified to take different model structures, e.g. known parameters, into account. The least squares estimates will converge to the true parameters e.g. under the following conditions.

—The output $\{y(t)\}$ is actually generated from a model (3.1).

—The residuals $\{e(t)\}$ are independent.

—The input is persistently exciting, see Ref. [3].

—The input sequence $\{u(t)\}$ is independent of the disturbance sequence $\{e(t)\}$.

These conditions are important. If the residuals are correlated the least squares estimate will be biased. If the input sequence $\{u(t)\}$ depends on $\{e(t)\}$ it may not be possible to determine all parameters.

When the inputs are generated by a feedback they are correlated with the disturbances and it is not obvious that all the parameters of the model can be determined. Neither is it obvious that the input generated in this way is persistently exciting of sufficiently high order. A simple example illustrates the point.

Example 3.1

Consider the first order model

$$y(t)+ay(t-1)=bu(t-1)+e(t). \qquad (3.2)$$

Assume that a linear regulator with constant gain

$$u(t)=\alpha y(t) \qquad (3.3)$$

is used. If the parameters a and b are known the gain $\alpha=a/b$ would obviously correspond to a minimum variance regulator. If the parameters are not known the gain $\alpha=\hat{a}/\hat{b}$ where $\hat{a}$ and $\hat{b}$ are the least squares estimates of a and b could be attempted. The least squares parameter estimates are determined in such a way that the loss function

$$V(a, b)=\sum_1^N[y(t+1)+ay(t)-bu(t)]^2 \qquad (3.4)$$

is minimal with respect to a and b. If the feedback control (3.3) is used the inputs and outputs are linearly related through

$$u(t)-\alpha y(t)=0. \qquad (3.5)$$

Multiply (3.5) by $-\gamma$ and add to the expression within brackets in (3.4).

Hence

$$V(a, b)=\sum_1^N[y(t+1)+(a+\alpha\gamma)y(t)-(b+\gamma)u(t)]^2$$

$$=V(a+\alpha\gamma, b+\gamma).$$

The loss function will thus have the same value for all estimates a and b on a linear manifold. Thus the two parameters a and b of the model (3.2) are not identifiable when the feedback (3.3) is used.

The simple example shows that it is in general not possible to estimate all the parameters of the model (3.1) when the input is generated by a feedback. Notice, however, that all parameters can be estimated if the control law is changed. In the particular example it is possible to estimate both parameters of the model, if the control law (3.3) is replaced by

$$u(t)=\alpha y(t-1)$$

or

$$u(t)=\alpha_1 y(t)+\alpha_2 y(t-1)$$

or if a time varying gain is used.

4. THE ALGORITHM

In order to control a system with constant but unknown parameters the following procedure could be attempted. At each step of time determine the parameters of the system (3.1) using least squares estimation based on all previously observed inputs and outputs as was described in section 3. Then determine a control law by calculating the minimum variance strategy for the model obtained. To compute the control law the identity (2.8) must be resolved in each step. The problem of computing the minimum variance regulator is simplified if it is observed that by using the identity (2.8) the system (3.1) can be written as

$$\begin{aligned}&y(t+k+1)+\alpha_1 y(t)+\ldots+\alpha_{\mathrm{m}}(t-m+1)\\&\quad=\beta_0[u(t)+\beta_1 u(t-1)+\ldots+\beta_l u(t-l)]\\&\quad\quad+\varepsilon(t+k+1)\end{aligned}\tag{4.1}$$

where $m=n$ and $l=n+k-1$. The coefficients α_i and β_i are computed from the parameters a_i and b_i in (3.1) using the identity (2.8). The disturbance $\varepsilon(t)$ is a moving average of order k of the driving noise $e(t)$.

The minimum variance strategy is then simply

$$\begin{aligned}u(t)=&\frac{1}{\beta_0}[\alpha_1 y(t)+\ldots+\alpha_m y(t-m+1)]\\&-\beta_1 u(t-1)-\ldots-\beta_l u(t-l).\end{aligned}\tag{4.2}$$

In order to obtain simple computation of the control strategy it could thus be attempted to use the model structure (4.1) which also admits least squares estimation. The trade-off for the simple calculation of the control law is that k more parameters have to be estimated.

As was shown in Example 3.1 all parameters of the model (4.1) can not necessarily be determined from input–output observations if the input is generated by a feedback (4.2) with constant parameters. In order to avoid a possible difficulty it is therefore assumed that the parameter β_0 is given. It will be shown in section 6 that the choice of β_0 is not crucial.

Summing up, the algorithm can be described as follows.

Step 1 *parameter estimation*

At the sampling interval t determine the parameters $\alpha_1, \ldots \alpha_m, \beta_1, \ldots, \beta_l$ of the model

$$\begin{aligned}&y(t)+\alpha_1 y(t-k-1)+\ldots+\alpha_m y(t-k-m)\\&\quad=\beta_0[u(t-k-1)+\beta_1 u(t-k-2)\\&\quad\quad+\ldots+\beta_l u(t-k-l-1)]+\varepsilon(t)\end{aligned}\tag{4.1}$$

using least squares estimation based on all data available at time t, i.e.

$$\sum_{k=0}^{t}\varepsilon^2(k)$$

minimum. The parameter β_0 is assumed known.

Step 2 *control*

At each sampling interval determine the control variable from

$$\begin{aligned}u(t)=&\frac{1}{\beta_0}[\alpha_1 y(t)+\ldots+\alpha_m y(t-m+1)]\\&-\beta_1 u(t-1)-\ldots-\beta_l u(t-l)\end{aligned}\tag{4.2}$$

where the parameters α_i and β_i are those obtained in Step 1.

The control law (4.2) corresponds to

$$\begin{aligned}u(t)&=\frac{\alpha_1+\alpha_2 q^{-1}+\ldots+\alpha_m q^{-m+1}}{\beta_0[1+\beta_1 q^{-1}+\ldots+\beta_l q^{-l}]}y(t)\\&=\frac{q^{l-m+1}\mathscr{A}(q)}{\mathscr{B}(q)}y(t).\end{aligned}\tag{4.3}$$

Since the least squares estimate can be computed recursively the algorithm requires only moderate computations.

It should be emphasized that the algorithm is not optimal in the sense that it minimizes the criterion (2.3), or the criterion (2.4). It fails to minimize (2.3) because it is not taken into account that the parameter estimates are inaccurate and it fails to minimize (2.4) because it is not dual in FELDBAUM's sense [4]. These matters are discussed in [2]. It will, however, be shown in section 5 that the algorithm has nice asymptotic properties.

The idea of obtaining algorithms by a combination of least squares identification and control is old. An early reference is KALMAN [5]. The particular algorithm used here is in essence the same as the one presented by PETERKA [6]. A similar algorithm where the uncertainties of the parameters are also considered is given in WIESLANDER–WITTENMARK [7].

5. MAIN RESULTS

The properties of the algorithm given in the previous section will now be analysed. We have

Theorem 5.1

Assume that the parameter estimates $\alpha_i(t)$, $i=1,\ldots,m$, $\beta_i(t)$, $i=1,\ldots l$ converge as $t\to\infty$ and that the closed loop system is such that the output is *ergodic* (in the second moments). Then the closed loop system has the properties

$$Ey(t+\tau)y(t)=r_y(\tau)=0 \qquad \tau=k+1,\ldots k+m \tag{5.1}$$

$$Ey(t+\tau)u(t)=r_{yu}(\tau)=0 \quad \tau=k+1,\ldots,k+l+1. \tag{5.2}$$

Proof

The least squares estimates of the parameters $\alpha_1, \alpha_2, \ldots, \alpha_m, \beta_1, \beta_2, \ldots, \beta_l$ is given by the equations

$$\left[\begin{array}{cccc:ccc}
\Sigma y(t)^2 & \Sigma y(t)y(t-1) & \ldots & \Sigma y(t)y(t-m+1) & -\beta_0\Sigma y(t)u & \ldots & -\beta_0\Sigma y(t)u(t-l) \\
\Sigma y(t)y(t-1) & & & \Sigma y(t-1)y(t-m+1) & \cdot & & \cdot \\
\cdot & & & \cdot & \cdot & & \cdot \\
\cdot & & & \cdot & \cdot & & \cdot \\
\cdot & & & \cdot & & & \\
\Sigma y(t)y(t-m+1) & & & \Sigma y^2(t-m+1) & -\beta_0\Sigma y(t-m+1)u(t-1) & \ldots & -\beta_0\Sigma y(t-m+1)u(t-l) \\
\hdashline
-\beta_0\Sigma y(t)u(t-1) & & & & \beta_0^2\Sigma u^2(t-1) & \ldots & \beta_0^2\Sigma u(t-1)u(t-l) \\
\cdot & & & & & & \cdot \\
\cdot & & & & & & \cdot \\
\cdot & & & & & & \cdot \\
-\beta_0\Sigma y(t)u(t-l) & \ldots & & & \ldots & & \beta_0^2\Sigma u^2(t-l)
\end{array}\right] \cdot
\begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \cdot \\ \cdot \\ \cdot \\ \alpha_m \\ \beta_1 \\ \cdot \\ \cdot \\ \cdot \\ \beta_l \end{bmatrix}
=
\begin{bmatrix}
-\Sigma y(t+k+1)y(t)+\beta_0\Sigma u(t)y(t) \\
\Sigma y(t+k+1)y(t-1)+\beta_0\Sigma u(t)y(t-1) \\
\cdot \\ \cdot \\ \cdot \\
-\Sigma y(t+k+1)y(t-m+1)+\beta_0\Sigma u(t)y(t-m+1) \\
-\beta_0\Sigma y(t+k+1)u(t-1)-\beta_0^2\Sigma u(t)u(t-1) \\
\cdot \\ \cdot \\ \cdot \\
\beta_0\Sigma y(t+k+1)u(t-l)-\beta_0^2\Sigma u(t)u(t-l)
\end{bmatrix} \tag{5.3}$$

where the sums are taken over N_0 values. See Ref. [3].

Assume that the parameters converge. For sufficiently large N_0 the coefficients of control law (4.2) will then converge to constant values. Introduction of (4.2) into (5.3) gives

$$\begin{aligned}
&\Sigma y(t+k+1)y(t)=0 \\
&\Sigma y(t+k+1)y(t-1)=0 \\
&\cdot \\
&\cdot \\
&\cdot \\
&\Sigma y(t+k+1)y(t-m+1)=0 \\
&\Sigma y(t+k+1)u(t-1)=0 \\
&\cdot \\
&\cdot \\
&\cdot \\
&\Sigma y(t+k+1)u(t-l)=0.
\end{aligned}$$

Using the control law (4.2) it also follows that

$$\Sigma y(t+k+1)u(t)=0.$$

Under the ergodicity assumption the sums can furthermore be replaced by mathematical expectations and the theorem is proven.

Remark 1

Notice that the assumptions on the system to be controlled are very weak. In particular it is not necessary to assume that the system is governed by an equation like (2.1) or (3.1).

Remark 2

It is sufficient for ergodicity that the system to be controlled is governed by a difference equation of finite order, e.g. like (2.1), and that the closed loop system obtained by introducing the feedback law into (2.1) gives a stable closed loop system.

Remark 3

The self tuning algorithm can be compared with a PI-regulator. If the state variables of a deterministic system with a PI-regulator converge to steady state values, the control error must be zero irrespective of the properties of the system. Analogously theorem 5.1 implies that if the parameter estimates of the self tuning algorithm converge the covariances (5.1) and (5.2) are zero.

Remark 4

Theorem 5.1 holds even if the algorithm is modified in such a way that the parameter estimation (Step 1) is not done in every step.

If it is assumed that the system to be controlled is governed by an equation like (2.1) it is possible to show that the conditions (5.1) and (5.2) in essence imply that the self tuning regulator will converge to a minimum variance regulator. We have

Theorem 5.2

Let the system to be controlled be governed by the equation (2.1). Assume that the self tuning algorithm is used with $m=n$ and $l=n+k-1$. If the parameter estimates converge to values such that the corresponding polynomials $\mathscr{A}$ and $\mathscr{B}$ have no common factors, then the corresponding regulator (4.2) will converge to a minimum variance regulator.

Proof

Assume that the least squares parameter estimates converge. The regulator is then given by (4.3) i.e.

$$u(t)=\frac{q^{l-m+1}\mathscr{A}(q)}{\mathscr{B}(q)}y(t)=\frac{q^k\mathscr{A}(q)}{\mathscr{B}(q)}$$

where the coefficients of $\mathscr{A}$ and $\mathscr{B}$ are constant. Since the system to be controlled is governed by (2.1) the closed loop system becomes

$$[A(q)\mathscr{B}(q)-B(q)\mathscr{A}(q)]y(t)=\lambda C(q)\mathscr{B}(q)e(t). \qquad (5.4)$$

The closed loop system is of order $r=n+l$. Introduce the stochastic process $\{v(t)\}$ defined by

$$v(t)=\lambda\frac{q^lC(q)}{A(q)\mathscr{B}(q)-B(q)\mathscr{A}(q)}e(t). \qquad (5.5)$$

Then

$$y(t)=q^{-l}\mathscr{B}(q)v(t) \qquad (5.6)$$

and

$$u(t)=q^{-m+1}\mathscr{A}(q)v(t). \qquad (5.7)$$

Multiplying (5.6) and (5.7) by $y(t+\tau)$ and taking mathematical expectations gives

$$r_y(\tau)=r_{yv}(\tau)+\beta_1r_{yv}(\tau+1)\ldots+\beta_lr_{yv}(\tau+l) \qquad (5.8)$$

$$r_{yu}(\tau)=\alpha_1r_{yv}(\tau)+\alpha_2r_{yv}(\tau+1)$$

$$+\ldots+\alpha_mr_{yv}(\tau+m-1). \qquad (5.9)$$

Furthermore it follows from Theorem 5.1, equations (5.1) and (5.2), that the left member of (5.8) vanishes for $\tau=k+1,\ldots,k+m$ and that the left member of (5.9) vanishes for $\tau=k+1,\ldots,k+l+1$. We thus obtain the following equation for $r_{yv}(\tau)$.

$$\begin{bmatrix}
1 & \beta_1 & \cdots & & & \beta_l & 0 & \cdots & 0\\
0 & 1 & \beta_1 & & & & \ddots & & \vdots\\
\vdots & & \ddots & & & & & \ddots & 0\\
0 & \cdots & 0 & 1 & \beta_1 & & & & \beta_l\\
0 & \alpha_1 & \cdots & & & \alpha_m & 0 & \cdots & 0\\
\vdots & & \ddots & & & & \ddots & & \vdots\\
 & & & & & & & & 0\\
0 & \cdots & & & & \alpha_1 & & \cdots & \alpha_m
\end{bmatrix}
\begin{bmatrix}
r_{yv}(k+1)\\
r_{yv}(k+2)\\
\vdots\\
\\
\\
\\
\\
r_{yv}(k+l+m)
\end{bmatrix}
=
\begin{bmatrix}
0\\
0\\
\vdots\\
\\
\\
\\
\\
0
\end{bmatrix}. \qquad (5.10)$$

Since the polynomials $\mathscr{A}$ and $\mathscr{B}$ have no common factor it follows from an elementary result in the theory of equations [8, p. 145] that the $(l+m) \times (l+m)$-matrix of the left member of (5.10) is nonsingular.

Hence

$$r_{yv}(\tau)=0 \qquad \tau=k+1, \ldots, k+l+m. \tag{5.11}$$

Since v is the output of a dynamical system of order $r=n+l=m+l$ driven by white noise it follows from (5.11) and the Yule–Walker equation that

$$r_{yv}(\tau)=0 \qquad \tau \geqslant k+1. \tag{5.12}$$

The equation (5.8) then implies

$$r_y(\tau)=0 \qquad \tau \geqslant k+1. \tag{5.13}$$

The output of the closed loop system is thus a moving average of white noise of order k. Denote this moving average by

$$y(t)=q^{-k}F(q)e(t) \tag{5.14}$$

where F is a polynomial of degree k. It follows from (5.4) and (5.14) that

$$q^kC=FA-\frac{BF\mathscr{A}}{\mathscr{B}}.$$

Since q^kC and FA are polynomials $BF\mathscr{A}/\mathscr{B}$ must also be a polynomial.

Hence

$$q^kC=FA+G \tag{5.15}$$

where

$$G=-\frac{BF\mathscr{A}}{\mathscr{B}} \tag{5.16}$$

is of degree $n-1$ and F of degree k. A comparison with (2.8) shows, however, that (5.15) is the identity which is used to derive the minimum variance strategy. It thus follows from (2.5) that the minimum variance strategy is given by

$$u(t)=-\frac{q^kG}{BF}y(t).$$

The equation (5.16) then implies that

$$-\frac{q^kG}{BF}=\frac{q^k\mathscr{A}}{\mathscr{B}} \tag{5.17}$$

and it has thus been shown that (4.2) is a minimum variance strategy.

Remark

The conditions $m=n$ and $l=n+k-1$ mean that there are precisely the number of parameters in the model that are required in order to obtain the correct regulator for the process (2.1). Theorem 5.2 still holds if there are more parameters in the regulator in the following cases.

(i) Theorem 5.2 still holds if $m=n$ and $l \geqslant n+k-1$. In this case the order of the system is $r=n+l$ and since $m=n$ the equation (5.10) implies (5.11)–(5.13) and the equation (5.16) is changed to

$$G=-\frac{q^{l-k-m+1}BF\mathscr{A}}{\mathscr{B}}. \tag{5.16'}$$

The rest of the proof remains unchanged.

(ii) If $m \geqslant n$ and $l=n+k-1$ the theorem will also hold. The closed loop system is of order $r \leq m+l$ but the equation (5.10) will still imply (5.11) and (5.16) is changed to (5.16′). The rest of the proof remains unchanged.

(iii) If $m>n$ and $l>n+k-1$ the theorem does not hold because $\mathscr{A}$ and $\mathscr{B}$ must have a common factor if the parameter estimates converge. It can, however, be shown that if the algorithm is modified in such a way that common factors of $\mathscr{A}$ and $\mathscr{B}$ are eliminated before the control signal is computed, Theorem 5.2 will still hold for the modified algorithm.

6. CONVERGENCE OF THE ALGORITHM

It would be highly desirable to have general results giving conditions for convergence of the parameter estimates. Since the system (2.1) with the regulator (4.2) and the least squares estimator is described by a set of nonlinear time dependent stochastic difference equations the problem of a general convergence proof is difficult. So far we have not been able to obtain a general result. It has, however, been verified by extensive numerical simulations that the algorithm does in fact converge in many cases. The numerical simulations as well as analysis of simple examples have given insight into some of the conditions that must be imposed in order to ensure that the algorithm will converge.

A significant simplification of the analysis is obtained if the algorithm is modified in such a way that the parameter estimates are kept constant over long periods of time. To be specific a simple example is considered.

Example 6.1

Let the system be described by

$$y(t)+ay(t-1)=bu(t-1)+e(t)+ce(t-1) \quad |c|<1. \tag{6.1}$$

Assume that the control law

$$u(t)=\alpha_n y(t) \tag{6.2}$$

is used in the time interval $t_n<t<t_{n+1}$ where the parameter α_n is determined by fitting the least squares model

$$y(t+1)+\alpha y(t)=u(t)+\varepsilon(t+1) \tag{6.3}$$

to the data $\{u(t), y(t), t=t_{n-1}, \ldots, t_n-1)\}$.

The least squares estimate is given by

$$\alpha_n=-\frac{\sum\limits_{t=t_{n-1}}^{t_n-2} y(t)[y(t+1)-u(t)]}{\sum\limits_{t=t_{n-1}}^{t_n-2} y^2(t)}$$

$$=\alpha_{n-1}-\frac{\sum\limits_{t=t_{n-1}}^{t_n-2} y(t+1)y(t)}{\sum\limits_{t=t_{n-1}}^{t_n-2} y^2(t)} \tag{6.4}$$

where the last equality follows from (6.2). Assume that $t_n-t_{n-1}\to\infty$ and that

$$|a-b\alpha_{n-1}|<1 \tag{6.5}$$

which means that the closed loop system used during the time interval $t_{n-1}<t<t_n$ is stable then

$$\alpha_n=\alpha_{n-1}-\frac{r_y(1)}{r_y(0)} \tag{6.6}$$

where $r_y(\tau)$ is the covariance function of the stochastic process $\{y(t)\}$ defined by

$$y(t)+(a-b\alpha_{n-1})y(t-1)=e(t)+ce(t-1). \tag{6.7}$$

Straightforward algebraic manipulations now give

$$\alpha_n=\alpha_{n-1}-\frac{(c-a+b\alpha_{n-1})(1-ac+bc\alpha_{n-1})}{1+c^2-2ac+2bc\alpha_{n-1}}. \tag{6.8}$$

The problem is thus reduced to the analysis of the nonlinear differerence equation given by (6.8).

Introduce

$$x_n=\alpha_n-\frac{a-c}{b} \tag{6.9}$$

the equation (6.8) then becomes

$$x_{n+1}=g(x_n)=(1-b)x_n+\frac{b^2cx_n^2}{1-c^2+2bcx_n}. \tag{6.10}$$

The point $x=0$ is a fixed point of the mapping g which corresponds to the optimal value of gain of the feedback loop, i.e. $\alpha=(a-c)/b$. The problem is thus to determine if this fixed point is stable. Since the closed loop system is assumed to be stable it is sufficient to consider

$$\frac{c-1}{b}<x<\frac{c+1}{b}. \tag{6.11}$$

Three cases have to be investigated

1. $c=0$
2. $c>0$ or $c<0$, $0<b\leqslant 1$
3. $c>0$ or $c<0$, $1<b<2$.

For all cases $g(x)\approx(1-b)x$ if x is small. This implies that solutions close to 0 converge to $x=0$ if $0<b<2$.

Case 1

The equation (6.10) reduces to

$$x_{n+1}=(1-b)x_n$$

and the fixed point $x=0$ is stable if $|1-b|<1$ i.e. $0<b<2$.

Case 2

The pincipal behavior of $g(x)$ when $c>0$ and $0<b\leqslant 1$ is shown in Fig. 1. It is straightforward to verify that all initial values in the stability region $(c-1)/b<x<(c+1)/b$ will give solutions which converge to zero.

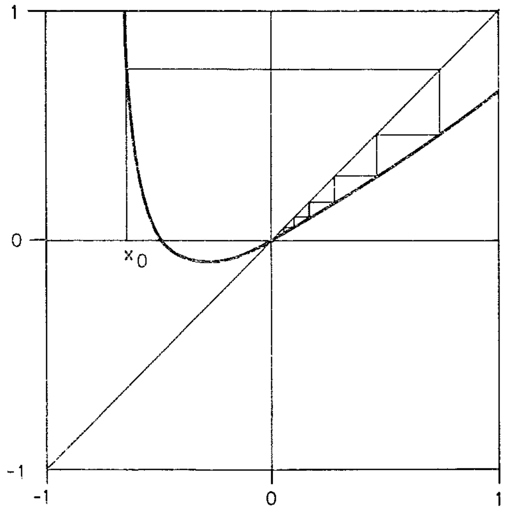


FIG. 1. Graph of the function g when $0<b\leqslant 1$. The figure is drawn with the parameter values $a=-0{\cdot}5$, $b=0{\cdot}5$ and $c=0{\cdot}7$.

Case 3

The function $g(x)$ for this case is shown in Fig. 2. It is not obvious that x will converge to zero, because there might exist a "limit cycle".

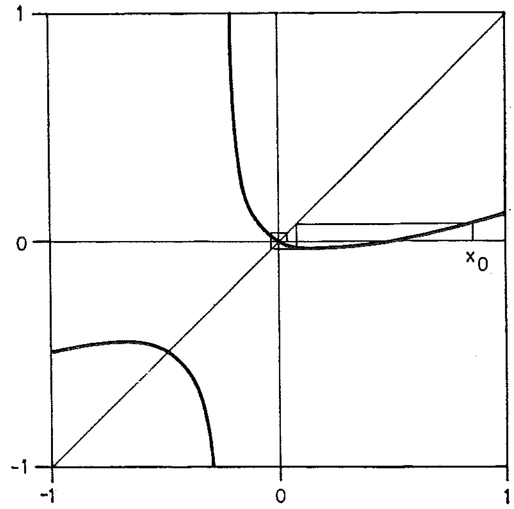


FIG. 2. Graph of the function g when $1<b<2$. The figure is drawn with the parameter values $a=-0{\cdot}5$, $b=1{\cdot}5$ and $c=0{\cdot}7$.

If $c>0$ and starting with $x_o>0$ then if $g(x_o)<0$ it can be shown that after two iterations the new value x_2 will satisfy $0<x_2<x_o$ i.e.

$$0<g(g(x))<x \qquad \text{if } x>0,\ g(x)<0.$$

If $(c-1)/b<x<0$ then $g(x)$ will be positive and can be taken as a new initial value for which the condition above will be satisfied. If $c<0$ then it can be shown that $g(g(x))>x$ if $x<0$ and $g(x)>0$.

Summary

From the analysis above we can conclude that $x=0$ is a stable fixed point if

$$-1<c<1$$

and

$$0<b<2. \qquad (6.12)$$

The example shows that under the condition (6.12) the version of the self-tuning algorithm where the parameters of the control law are kept constant over long intervals will in fact converge. In the analysis above $\beta_o=1$ was chosen. If $\beta_o\neq 1$ then the condition (6.12) is replaced by

$$0<b/\beta_o<2 \qquad (6.12')$$

or

$$0.5b<\beta_o<\infty. \qquad (6.13)$$

The condition (6.13) implies that it is necessary to pick the parameter β_o in a correct manner. The algorithm will always converge if β_o is greater than b. Under-estimation may be serious and the value $\beta_o<0{\cdot}5b$ gives an unstable algorithm.

The analysis presented in the simple example can be extended to give stability conditions for the modified algorithm in more complex cases. The analysis required is tedious.

7. SIMULATIONS

The results in section 5 are given under the assumption that the least squares estimator really converges, but yet we have not been able to give general conditions for convergence. But simulation of numerous examples have shown that the algorithm has nice convergence properties.

This section presents a number of simulated examples which illustrate the properties of the self-tuning algorithm.

Example 7.1

Let the system be

$$y(t)+ay(t-1)=bu(t-1)+e(t)+ce(t-1) \qquad (7.1)$$

with $a=-0{\cdot}5$, $b=3$ and $c=0{\cdot}7$. The minimum variance regulator for the system is

$$u(t)=\frac{a-c}{b}y(t)=-0{\cdot}4y(t). \qquad (7.2)$$

A regulator with this structure can be obtained by using the self-tuning algorithm based on the model

$$y(t+1)+\alpha y(t)=\beta_o u(t)+\varepsilon(t+1). \qquad (7.3)$$

Figure 3 shows for the case $\beta_o=1$ how the parameter estimate converges to the value $\alpha=-0{\cdot}4$ which corresponds to the minimum variance strategy (7.2).

In Figure 4 is shown the expected variance of the output if the current value of α should be used for all future steps of time. Notice that the algorithm has practically adjusted over 50 steps.

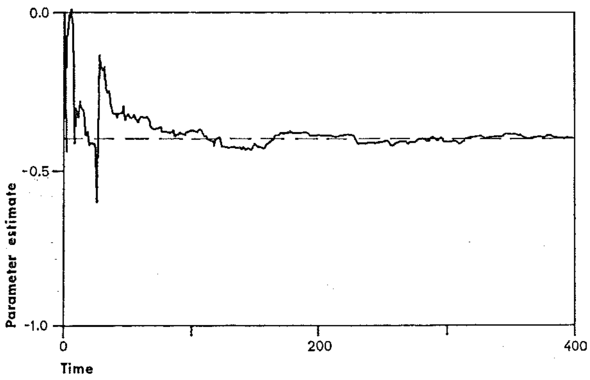


FIG. 3. Parameter estimate $a(t)$ obtained when the self tuning algorithm based on the model (7.3) is applied to the system given by (7.1). The minimum variance regulator corresponds to $a=-0{\cdot}4$ and is indicated by the dashed line.

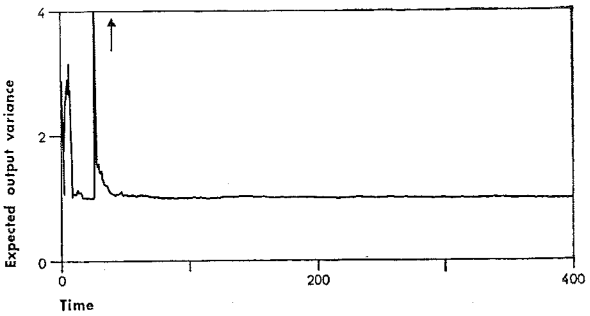


FIG. 4. Expected variance of the output of Example 7.1 if the control law obtained at time t is kept constant for all future times. Notice that the estimate at time $t=26$ would give an unstable system.

The analysis of Example 6.1 shows that, since $b>2$, and $\beta_o=1$ the modified self-tuning algorithm obtained when the parameters of the controller are kept constant over long intervals is unstable. The simulation in Example 7.1 shows that at least in the special case a conservative estimate of the convergence region is obtained by analysing the modified algorithm. If the value of b is increased further it has been shown that the algorithm is unstable. Unstable realizations have been found for $b=5$. In such cases it is of course easy to obtain a stable algorithm by increasing β_o. This requires, however, a knowledge of the magnitude of b.

The system of Example 7.1 is very simple. For instance, if no control is used the variance will still be reasonably small. The next example is more realistic in this aspect.

Example 7.2

Consider the system

$$y(t)-1.9y(t-1)+0{\cdot}9y(t-2)=u(t-2)$$
$$+u(t-3)+e(t)-0{\cdot}5e(t-1). \qquad (7.4)$$

If no control is used the variance of the output is infinite. Also notice that $B(z)=z-1$. The assumption that B has all zeroes inside the unit circle is thus violated. The minimum variance strategy for the system is

$$u(t)=-1{\cdot}76y(t)+1{\cdot}26y(t-1)-0{\cdot}4u(t-1)$$
$$+1{\cdot}4u(t-2). \qquad (7.5)$$

A regulator with this structure is obtained by using the self-tuning algorithm with the model

$$y(t+2)+\alpha_1y(t)+\alpha_2y(t-1)$$
$$=u(t)+\beta_1u(t-1)+\beta_2u(t-2)+\varepsilon(t+2). \qquad (7.6)$$

The convergence of the parameters is shown in Fig. 5. Figure 6 shows the accumulated losses when using the self-tuning algorithm and when using the optimal minimum variance regulator (7.5).

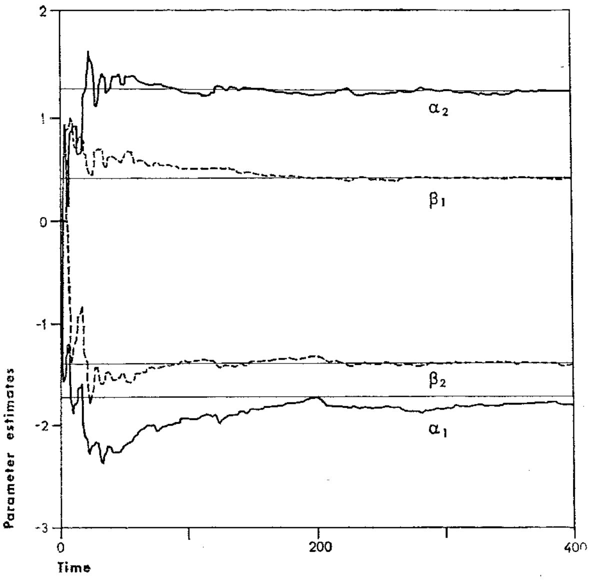


FIG. 5. Parameter estimates α_1, α_2, β_1 and β_2 obtained when the self tuning algorithm based on (7.6) is applied to the system given by (7.4). The thin lines indicate the parameter values of the minimum variance strategy.

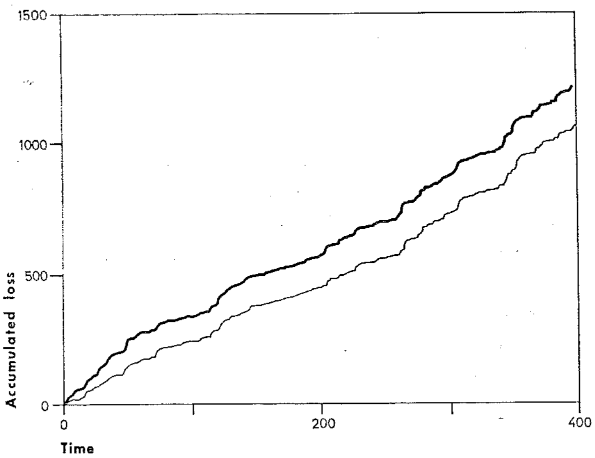


FIG. 6. Accumulated loss

$$\sum_{s=1}^{t} y^2(s)$$

for a simulation of the system (7.4) when using the self tuning algorithm (thick line) and when using the optimal minimum variance regulator (7.5) (thin line).

In both examples above, the models in the self-tuning algorithm have had enough parameters so it could converge to the optimal minimum variance regulator. The next example shows what happens when the regulator has not enough parameters.

Example 7.3

Consider the system

$$y(t)-1{\cdot}60y(t-1)+1{\cdot}61y(t-2)-0{\cdot}776y(t-3)$$
$$=1{\cdot}2u(t-1)-0{\cdot}95u(t-2)+0{\cdot}2u(t-3)+e(t)$$
$$+0{\cdot}1e(t-1)+0{\cdot}25e(t-2)+0{\cdot}87e(t-3). \qquad (7.7)$$

The polynomial $A(z)$ has two complex zeroes near the unit circle $(+0{\cdot}4\pm 0{\cdot}9i)$ and one real zero equal to 0·8.

If a self-tuning regulator is determined based on a model with $m=3$ and $l=2$ it will converge to the minimum variance regulator as expected. Figure 7(a) shows a small sample of the output together with the sample covariance of the output, $\hat{r}_y(\tau)$.

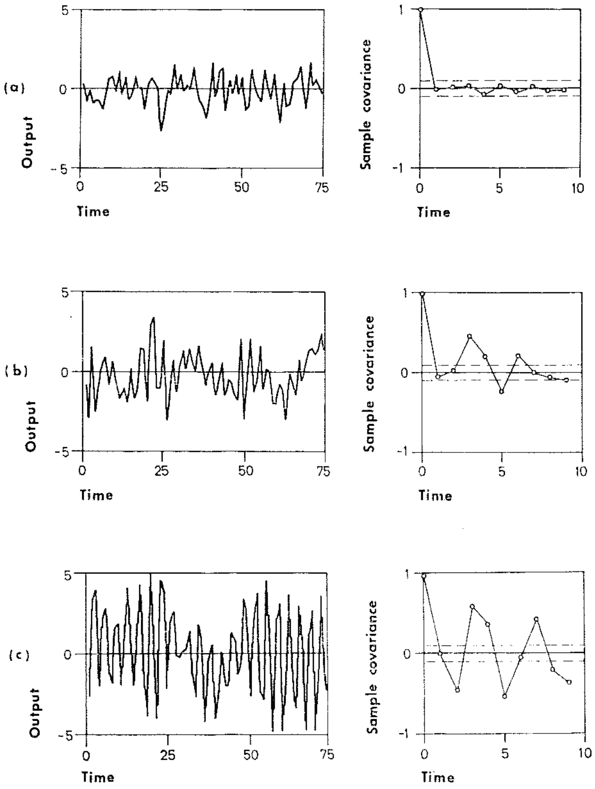


FIG. 7. Output of the system (7.7) and sample covariance of the output $\hat{r}_y(\tau)$ when controlling with self tuning regulator is having different number of parameters

(a) $m=3$, $l=2$ (b) $m=2$, $l=1$ (c) $m=1$, $l=0$.

The dashed lines show the 5 per cent confidence intervals for $\tau\neq 0$.

If the self-tuning algorithm instead is based on a model with $m=2$ and $l=1$ it is no longer possible to obtain the minimum variance regulator for the system since there are not parameters enough in the self-tuning regulator. Theorem 5.1 indicates, however, that if the self-tuning regulator converges, its parameters will be such that the covariances $r_y(1)$, $r_y(2)$, $r_{yu}(1)$ and $r_{yu}(2)$ are all zero. The simulation shows that the algorithm does in fact converge with $\beta_0=1{\cdot}0$. The covariance function of the output is shown in Fig. 7(b). It is seen that the sample covariances $\hat{r}_y(1)$ and $\hat{r}_y(2)$ are within the 5 per cent confidence interval while $\hat{r}_y(3)$ is not as would be expected from Theorem 5.1.

If a self-tuning algorithm is designed based on a model with $m=1$, $l=0$ then Theorem 5.1 indicates that $r_y(1)$ should vanish. Again the simulation shows that the algorithm does in fact converge and that the sample covariance $\hat{r}_y(1)$ does not differ significantly from zero. See Fig. 7(c).

When using regulators of lower order than the optimal minimum variance regulator, the parameters in the controller will not converge to values which for the given structure gives minimum variance of the output. In Table 1 is shown the variance of the output for the system above when using different regulators.

The loss when using the self-adjusting regulator is obtained through simulations. The optimal regulator is found by minimizing $r_y(0)$ with respect to the parameters in the controller.

TABLE 1

		Loss $\frac{1}{N}\sum_{t=1}^{N} \mathrm{H}y_2(t)$	
m	l	Self-adjusting	Optimal
3	2	1·0	1·0
2	1	2·5	1·9
1	0	4·8	3·4

The previous examples are all designed to illustrate various properties of the algorithm. The following example is a summary of a feasibility study which indicates the practicality of the algorithm for application to basis weight control of a paper machine.

Example 7.4

The applicability of minimum variance strategies to basis weight control on a paper machine was demonstrated in [9]. In this application the control loop is a feedback from a wet basis weight signal to thick stock flow. The models used in [9] were obtained by estimating the parameters of (2.1) using the maximum likelihood method. In one particular case the following model was obtained.

$$y(t)=\frac{b_1q^{-1}+b_2q^{-2}}{1+a_1q^{-1}+a_2q^{-2}}u(t-2)+v(t) \qquad (7.8)$$

where the output y is basis weight in g/m^2 and the control variable is thick stock flow in g/m^2. The disturbance $\{v(t)\}$ was a drifting stochastic process which could be modelled as

$$v(t)=\lambda\frac{1+c_1q^{-1}+c_2q^{-2}}{(1+a_1q^{-1}+a_2q^{-2})(1-q^{-1})}e(t) \qquad (7.9)$$

where $\{e(t)\}$ is white noise. The sampling interval was 36 sec and the numerical values of the parameters obtained through identification were as follows

$$a_1 = -1{\cdot}283$$

$$a_2 = 0{\cdot}495$$

$$b_1 = 2{\cdot}307$$

$$b_2 = -2{\cdot}025$$

$$c_1 = -1{\cdot}438$$

$$c_2 = 0{\cdot}550$$

$$\lambda = 0{\cdot}382.$$

To investigate the feasability of the self-tuning algorithm for basis weight control, the algorithm was simulated using the model (7.8) where the disturbance v was the actual realization obtained from measurements on the paper machine. The parameters of the regulator were chosen as $k=1$, $l=3$, $m=4$ and $\beta_o=2{\cdot}5$ and the initial estimates were set to zero. The algorithm is thus tuning 7 parameters.

The results of the simulation are shown in Figs. 8–10. Figure 8 compares the output obtained when using the self-tuning algorithm with the result obtained when using the minimum variance regulator computed from the process model (7.8) with the disturbance given by (7.9). The reference value was 70 g/m^2. In the worst case the self-tuning regulator gives a control error which is about 1 g/m^2 greater than the minimum variance regulator. This happens only at two sampling intervals.

After about 75 sampling intervals (45 min) the output of the system is very close to the output obtained with the minimum variance regulator.

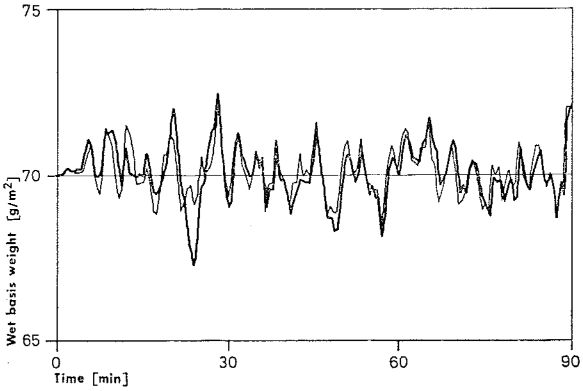


FIG. 8. Wet basis weight when using the self tuning regulator (thick line) and when using the minimum variance regulator based on maximum likelihood identification (thin line). The reference value for the controller was 70g/m².

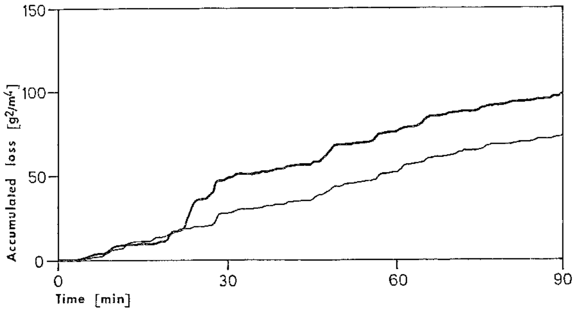


FIG. 9. Accumulated loss for Example 7.4 when using the self tuning regulator (thick line) and when using the minimum variance regulator (thin line).

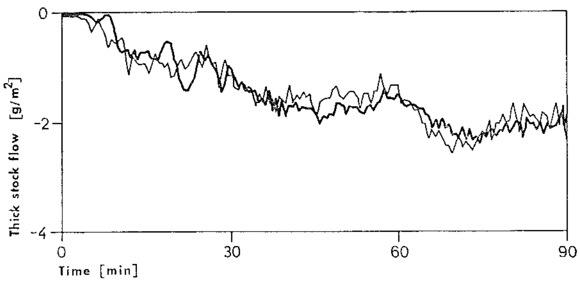


FIG. 10. The control signal in g/m² for Example 7.4 when using the self tuning regulator (thick line) and when using the minimum variance regulator (thin line).

Figure 9 compares the accumulated losses

$$V(t) = \sum_{n=0}^{t} y^2(n)$$

obtained with the minimum variance regulator and the self-tuning regulator. Notice that in the time interval (21, 24) minutes there is a rapid increase in the accumulated loss of the self-tuning regulator of about 17 units. The largest control error during this interval is 2·7 g/m^2 while the largest error of the minimum variance regulator is 1 g/m^2. The accumulated losses over the last hour is 60 units for the self-tuning regulator and 59 units for the minimum variance regulator.

The control signal generated by the self-tuning algorithm is compared with that of the minimum variance regulator in Fig. 10. There are differences in the generated control signals. The minimum variance regulator generates an output which has more rapid variations than the output of the self-tuning regulator.

The parameter estimates obtained have not converged in 100 sampling intervals. In spite of this the regulator obtained will have good performance as has just been illustrated. The example thus indicates that the self-tuning algorithm could be feasible as a basis weight regulator.

8. PRACTICAL ASPECTS

A few practical aspects on the algorithm given in section 4 are presented in this section which also covers some possible extensions of the results

A priori knowledge

The only parameters that must be known à *priori* are k, l, m and β_o. If the algorithm converges it is easy to find out if the à *priori* guesses of the parameters are correct simply by analyzing the sample covariance of the output. Compare Example 7.3. The parameter β_o should be an estimate of the corresponding parameter of the system to be controlled. The choice of β_o is not critical as was shown in the Examples 6.1 and 7.1. In the special cases studied in the examples an under-estimate led to a diverging algorithm while an over-estimate was safe.

Implementation on process computers

It is our belief that the self-tuning algorithm can be conveniently used in process control applications. There are many possibilities. The algorithm can be used as a tool to tune regulators when they are installed. It can be installed among the systems programs and cycled through different control loops repetitively to ensure that the regulators are always properly tuned. For critical loops where the parameters are changing it is also possible to use a dedicated version which allows slowly drifting parameters.

A general self-tuning algorithm requires about 40 FORTRAN statements. When compiled using the standard PDP 15/30 FORTRAN compiler the code consists of 450 memory locations. The number of memory locations required to store the data is $(l-1+m)^2+3(l-1+m)+2k+4$. Execution times on a typical process computer (PDP 15) without floating point hardware are given in the table below. The major part of the computing is to update the least squares estimate.

Number of parameters $l+m$	Execution time ms
1	5
3	16
5	34
8	69

Improved convergence rates

The results of this paper only shows that if the parameters converge the regulator obtained will tend to a minimum variance regulator. Nothing is said about convergence rates, which of course is of great interest from practical as well as theoretical points of view. There are in fact many algorithms that have the correct asymptotic properties. Apart from the algorithm given in section 4 we have the algorithm which minimizes (2.4). But that algorithm is impossible to use due to the computational requirements. It is of interest to investigate if other possible algorithms have better convergence rates than the algorithm of section 4. No complete answer to this problem is yet known. A few possibilities will be indicated. It could be attempted to take into account that the parameter estimates are uncertain. See Refs. [2, 7 and 10]. The least squares identifier can be improved upon by introducing exponential weighting of past data. This has in some cases shown to be advantageous in simulations. Algorithms of this type have in simulations been shown to handle slowly drifting parameters.

Another possibility is to assume that the parameters are Wiener processes, which also can be incorporated fairly easily [2, 7]. It has been verified by simulation that the region of convergence can be improved by introducing a bound on the control signal.

Feed forward

In many industrial applications the control can be improved considerable if feed forward is used. The self tuning regulators in this paper can include feed forward control by changing the process model (4.1) to

$$\begin{aligned} y(t+k+1)&+\alpha_1 y(t)+\dots+\alpha_m y(t-m+1) \\ &=\beta_0[u(t)+\beta_1 u(t-1)+\dots+\beta_l u(t-l)]+\gamma_1 s(t) \\ &\quad+\dots+\gamma_p s(t-p+1)+\varepsilon(t+k+1) \qquad (8.1) \end{aligned}$$

where $s(t)$ is a known disturbance.

The parameters α_i, β_i and γ_i can be identified as before and the control law (4.2) will be changed to

$$\begin{aligned} u(t)=\frac{1}{\beta_0}[&\alpha_1 y(t)+\dots+\alpha_m y(t-m+1)-\gamma_1 s(t) \\ &-\dots-\gamma_p s(t-p+1)]-\beta_1 u(t-1) \\ &-\dots-\beta_l u(t-l). \qquad (8.2) \end{aligned}$$

Nonminium phase systems

Difficulties have been found by a straightforward application of the algorithm to nonminimum phase systems, i.e. systems where the polynomial B has zeroes outside the unit circle.

Several ways to get around the difficulty have been found. By using a model with $B(z)=\beta_o$ it has in many cases been possible to obtain stable algorithms at the sacrifice of variance.

It is well-known that the minimum variance regulators are extremely sensitive to parameter variations for nonminimum phase systems [1]. This is usually overcome by using suboptimal strategies which are less sensitive [1]. The same idea can be used for the self-tuning algorithms as well. The drawback is that the computations increase because the polynomials F and G of an identity similar to (2.8) must be determined at each step of the iteration. An alternative is to solve a Riccati-equation at each step.

Multivariable and nonlinear systems

It is possible to construct algorithms that are similar to the one described in section 4 for multivariable and nonlinear systems as long as a model structure which is linear in the parameters [3, p. 131] is chosen. For multivariable systems the structure given in equation (3.2) of Ref. [3] can thus be attempted. Analyses of the properties of the algorithm obtained when applied to a multivariable or a nonlinear system are not yet done.

9. CONCLUSIONS

The paper has been concerned with control of systems with constant but unknown parameters. The analysis has been limited to single-input single-output systems with disturbances in terms of filtered white noise. A control algorithm based on least squares parameter estimation and a minimum variance regulator computed from the estimated parameters has been analysed. Assuming that the parameter estimates converge the closed loop system has been analysed. A characterization of the closed loop system has been given under weak assumption on the system to be controlled. Under stronger assumptions on the system to be controlled it has been shown that the regulator obtained will actually converge to the minimum variance regulator if the estimates converge.

Since the closed loop system is characterized as a nonlinear stochastic system it is very difficult to give general conditions that guarantee that the estimates converge. The convergence has only been treated for simple examples and under further assumptions as in section 6. But simulations of numerous examples indicate that the algorithm has nice convergence properties.

The simplicity of the algroritm in combination with its asymptotic properties indicate that it can be useful for industrial process control. The feasibility has also been demonstrated by experiments on real processes in the paper and mining industries.

10. REFERENCES

[1] K. J. ÅSTRÖM: *Introduction to Stochastic Control Theory.* Academic Press, New York (1970).

[2] K. J. ÅSTRÖM and B. WITTENMARK: Problems of identification and control. *J. Math. Analysis Applic.* **34,** 90–113 (1971).

[3] K. J. ÅSTRÖM and P. EYKHOFF: System identification—A survey. *Automatica* **7,** 123–162 (1971).

[4] A. A. FELDBAUM: Dual control theory I–IV. *Aut. Remote Control* **21,** 874–880, 1033–1039 (1961); **22,** 1–12, 109–121 (1962).

[5] R. E. KALMAN: Design of a self optimizing control System. *Trans. ASME* **80,** 468–478 (1958).

[6] V. PETERKA: Adaptive Digital Regulation of Noisy Systems, 2nd Prague IFAC Symposium on Identification and Process Parameter Estimation (1970).

[7] J. WIESLANDER and B. WITTENMARK: An approach to adaptive control using real time identification. *Automatica* **7,** 211–217 (1971).

[8] L. E. DICKSON: *First Course in the Theory of Equations.* Wiley, New York (1922).

[9] K. J. ÅSTRÖM: Computer control of a paper machine—An application of linear stochastic control theory. *IBM J. Res. Development* **11,** 389–405 (1967).

[10] J. B. FARISON *et al.*: Identification and control of linear discrete systems. *IEEE Trans. Aut. Control* **12,** 438–442 (1967).

Résumé—On considère le problème du contrôle d'un système avec paramètres constants mais inconnus. L'analyse se limite aux systèmes discrets à entrée unique et sortie unique. Un algorithme obtenu en combinant un estimateur à carrés minimum avec un régulateur à variance minimum calculée du modèle estimé, est analysé. Les résultats principaux sont deux théorêmes qui caractérisent le système à boucle fermée obtenu en supposant que les estimations des paramètres convergent. Le premier théorême dit que certaines covariances de la sortie et certaines covariances transverses de la variable de contrôle et de la sortie disparaitront avec des suppositions faibles du système contrôlé. Dans le second théorême il est supposé que le système à contrôler est un système général linéaire stochastique du nème ordre. Il est montré que si les estimations des paramètres convergent, la loi de contrôle obtenue est en fait la loi de contrôle de variance minimale qui pourrait être calculée si les paramètres du système étaient connus. Ceci est quelque peu surprenant car l'estimation des carrés minimum est partiale. On discute certaines des implications pratiques des résultats. Il est montré en particulier qu'il est possible d'appliquer l'algorithme à un petit ordinateur.

Zusammenfassung—Betrachtet wird das Problem der Steuerung eines Systems mit konstanten, aber unbekannten Parametern. Die Analyse wird auf Systeme mit Diskretzeit und einem Eingang bzw. einem Ausgang beschränkt. Ein durch Kombination einer Schätzeinrichtung und der Methode der kleinsten Quadrate mit einer aus dem Schätzmodell berechneten Regeleinrichtung erhaltener Algorithmus wird analysiert. Die Hauptergebnisse sind zwei Theoreme, die unter der Annahme, daß die Parameterschätzungen konvergieren, den erhaltenen geschlossenen Regelkreis charakterisieren. Das erste Theorem konstatiert, daß bestimmte Kovarianzen des Ausganges und bestimmte Kreuz-Kovarianzen der Steuervariablen und des Ausgangs unter schwachen Annahmen über das zu regelnde System verschwinden. Im zweiten Theorem wird angenommen, daß das zu regelnde System ein allgemeines lineares stochastisches System n-ter Ordnung ist. Gezeigt wird, daß bei Konvergenz der Parameterschätzung des erhaltenen Steuergesetzes in der Tat das Steuergesetz bei minimaler Varianz ist, das berechnet werden kann, wenn die Parameter des Systems bekannt waren. Das ist etwa überraschend, weil die Schätzung nach den kleinsten Quadraten angesteuert wird. Einige praktische Folgerungen aus den Ergebnissen werden diskutiert. Speziell wird gezeigt, daß der Algorithmus auf einem kleinen Prozeßrechner leicht verwirklicht werden kann.

Резюме—Рассматривается проблема регулирования систем при помощи постоянных но неизвестных параметров. Анализ ограничивается системами дискретного времени с одним вводом и с одним выводом. Анализируется алгоритм, полученный путем комбинирования оценок наименьших квадратов с регулировкой минимальных колебаний, вычисленных из расчетной модели. Основными результатами являются две теоремы, характеризующие систему замкнутого контура, полученного в предположении, что оценки параметров сходятся. Первая теорема утверждает, что некоторая ковариантность вывода и некоторая пересекающая ковариантность контрольных переменных вывода исчезнут при небольших допущениях в регулируемой системе. Во второй теореме предполагается, что контролируемая система является общей линейной стохастической системой порядка n:th. Показано, что, если оценки параметров сходятся, то полученный закон регулировкя ивляется фактически законом регулировки минимального колебания, которое могло бы быть вычислено при известных параметрах системы. Это до некоторой степени неожиданно, так как оценка наименьших квадратов явияется смещенной. Обсуждаются некоторые практические выводы из результатов. В частности, показано, что этот алгоритм может быть применен на небольшой счетно-вычислительной машине для процессов.

REPORT 7311
APRIL 1973

A Self-tuning Regulator

BJÖRN WITTENMARK

LTH

Division of Automatic Control · Lund Institute of Technology

A SELF-TUNING REGULATOR.

B. Wittenmark

ABSTRACT.

The control of constant but unknown single-input single-output systems is considered. Controllers for this type of systems are called self-tuning regulators. The proposed algorithm can be divided into two steps. Firstly, the parameters in a model of the process is estimated using the method of least squares. Secondly, a minimum variance controller is determined based on the estimated parameters. The basic algorithm discussed in the report has several attractive properties. It can, for instance, be shown that if the estimation converges, then under weak assumptions the regulator will converge to the optimal controller that could be obtained if the parameters of the system were known. The behaviour of the system has been investigated theoretically as well as experimentally. Questions concerning convergence and limitations are discussed. The algorithm is easy to implement on a small computer which makes the regulator very attractive for use on industrial processes.

TABLE OF CONTENTS

1. INTRODUCTION.

One difficulty when designing regulators is to determine a good model of the process to be controlled. The dynamics of the process as well as the characteristics of the disturbances must in many cases be known. Identification methods of different kinds have been developed to meet this demand [6]. In many cases it is necessary to make the computations of the control strategy off-line. This has led to the idea of adaptive controllers. An adaptive controller has the ability that it can change its parameters depending on the environment. An adaptive regulator will thus assure that the regulator loop is properly tuned despite changing dynamics and characteristics of the disturbances. It is, however, very difficult to analyze a system controlled by an adaptive controller. One interesting special case is when the process has constant but unknown parameters. Controllers for this type of processes will be called self-tuning regulators. This report thoroughly discusses a self-tuning regulator described in [8]. A similar algorithm is given in [24]. Other approaches are discussed for instance in [26] and [27].

The basic algorithm described in [8] is derived to control minimum phase single-input single-output systems. The algorithm is based on the assumption that it is possible to separate the estimation of the parameters in the process from the determination of the control law. The algorithm can thus be divided into two parts. Firstly, the parameters in a model of the process are estimated using the method of least squares [6]. Secondly, the minimum variance regulator [1] is computed, as if the values of the estimated parameters were the true ones. Thus it is not taken into consideration that the values of the estimated parameters are uncertain. The self-tuning regulator contains a few parameters that must be specified. For instance it is necessary to determine the time-delay in the process and the number of parameters in the regulator. The main features of the basic algorithm were examplified in [8]. Under weak assumptions it can be shown that if the algorithm converges then the regulator will converge to the minimum variance regulator that could be obtained if the parameters in the process were known. The properties of the algorithm will be further analyzed and discussed in this report. Various types of modifications and extensions will also be given. The self-tuning regulator has successfully been used on processes in the paper and mining industries. Experiences of moisture control on a paper machine are given in [12].

It is not trivial to analyze the properties obtained when a self-tuning regulator is used to control a stochastic system since the closed loop system is nonlinear, stochastic and time-varying. Certain asymptotic results can be established mathematically. Many interesting practical questions like convergence, convergence rate etc. have been investigated through simulations. One serious drawback with simulation is, however, that it is difficult to find out if an algorithm converges. The simulations have been performed through the use of an interactive computer program which makes it possible to enter system and regulator descriptions from a teletype. The results of the simulations can be presented on a display or a line-printer. Simulation of a second order system with four parameters in the regulator takes about 25 seconds for 1000 time steps on a PDP--15. The simulations have served as a rich source of empirical results. Some of these results could later be proved mathematically. The investigation of the self-tuning regulators given in this report should thus be considered as an experimental as much as a theoretical study. It is, of course, not possible to present all the results of the simulations, only a small fraction is given in this report.

The report is organized as follows: In Section 2 the basic algorithm and the main results given in [8] are listed for easy reference. The selection of the different parameters in the algorithm is discussed in Section 3. Theoretical results as well as practical rules of thumb are given. The convergence of the algorithm can be proved in some special cases. It is possible to give examples which show that the algorithm does not converge in general. Convergence is discussed in Section 4.

Modifications of the basic algorithm are given in Section 5. It is e.g. shown how feed forward compensation can be included in the self-tuning algorithm. Although the algorithm in most cases has a very good performance there are certain limitations of the algorithm. One limitation discussed in Section 6 is that the parameters of the process are assumed to be constant. If the parameters are slowly varying it is possible to modify the algorithm in such a way that the regulator can follow the parameter variations. But if the variations are fast the algorithm will have an unsatisfactory behaviour. Nonminimum phase systems can also be handled by making some minor modifications.

In order to get a smooth start-up the algorithm can be modified to take into consideration that the parameter estimates are uncertain. This is discussed in Section 7. In Section 8 it is shown how self-tuning regulators can be used in connection with the classical servo-mechanism problem, i.e. to design regulators which make it possible to reproduce different types of reference signals. References are given in Section 9.

I want to express my gratitude to my advisor Karl Johan Åström who has found time to discuss and to give valuable criticism from the first ideas until the final manuscript. Also I want to thank Lennart Ljung who has helped me with the part concerning the convergence properties of the self-tuning regulators.

2. THE BASIC ALGORITHM.

The basic self-tuning algorithm is described in [8]. The algorithm is derived to control linear single-input single-output systems which can be described by the model:

$$A(q)y(t) = B(q)u(t-k) + C(q)e(t) \tag{2.1}$$

where q is the forward shift operator, {e(t)} is a sequence of independent N $(0,\sigma)$ random variables and

$$A(q) = q^n + a_1q^{n-1} + \dots + a_n \tag{2.2}$$

$$B(q) = b_1q^{n-1} + \dots + b_n \tag{2.3}$$

$$C(q) = q^n + c_1q^{n-1} + \dots c_n \tag{2.4}$$

It is assumed that the parameters in the polynomials (2.2)-(2.4) are constant but unknown. Further it is assumed that the B-polynomial has all roots inside or on the unit circle, i.e. the system is minimum phase.

If the parameters in (2.1) are known the output variance is minimized by using the control law [1]

$$u(t) = - \frac{q^kG(q)}{B(q)F(q)} y(t)$$

F and G are polynomials of degree k and n-1 respectively given by the identity

$$q^kC(q) = A(q)F(q) + G(q) \tag{2.5}$$

If $C(q) = q^n$ it is possible to rewrite (2.1) as

$$y(t) + \alpha_1 y(t-k-1) + \ldots + \alpha_m y(t-k-m) =$$

$$= \beta_0 [u(t-k-1) + \beta_1 u(t-k-2) + \ldots + \beta_\ell u(t-k-\ell-1)] +$$

$$+ \varepsilon(t) \tag{2.6}$$

where $m = n$ and $\ell = n+k-1$. The parameters α_i and β_i can be computed from the parameters in (2.1) using the identity (2.5). The disturbance $\varepsilon(t)$ is a moving average of order k of the noise e(t). The minimum variance regulator for (2.6) is

$$u(t) = \frac{1}{\beta_0}[\alpha_1 y(t) + \ldots + \alpha_m y(t-m+1)] -$$

$$- \beta_1 u(t-1) - \ldots - \beta_\ell u(t-\ell) \tag{2.7}$$

The idea is now to estimate the parameters in the model (2.6) and to use the estimated parameter values instead of the true parameter values in the controller (2.7).

The algorithm can thus be described as follows:

Step 1: Parameter estimation. At each sampling interval t determine the parameters $\alpha_1, \ldots, \alpha_m, \beta_1, \ldots, \beta_\ell$ of the model (2.6) based on data available at time t using the method of least squares, i.e. minimize

$$\sum_{i=0}^{t} \varepsilon(i)^2$$

[3]. The parameter β_0 is assumed known.

Step 2: Control. At each sampling interval determine the control variable from the control law (2.7) where the parameters α_i and β_i are those obtained in Step 1.

Using the forward shift operator the control law (2.7) can be written as

$$u(t) = \frac{\alpha_1 + \alpha_2 q^{-1} + \ldots + \alpha_m q^{-m+1}}{\beta_0\left[1 + \beta_1 q^{-1} + \ldots + \beta_\ell q^{-\ell}\right]}\, y(t) =$$

$$= \frac{q^{\ell-m+1}\mathcal{A}(q)}{\mathcal{B}(q)}\, y(t) \qquad (2.8)$$

where

$$\mathcal{A}(q) = \frac{1}{\beta_0}\left[\alpha_1 q^{m-1} + \alpha_2 q^{m-2} + \ldots + \alpha_m\right] \qquad (2.9)$$

$$\mathcal{B}(q) = q^{\ell} + \beta_1 q^{\ell-1} + \ldots + \beta_\ell \qquad (2.10)$$

The idea to estimate the parameters in a model as (2.6) using the method of least squares is for instance found in [19]. The least squares estimates can easily be computed recursively, see [3], [30].

Introduce the vectors φ and θ defined by

$$\varphi(t) = [-y(t) \quad -y(t-1) \ \ldots \ -y(t-m+1) \quad \beta_0 u(t-1) \ \ldots \ \beta_0 u(t-\ell)]$$

$$\theta^T = [\alpha_1 \quad \alpha_2 \ \ldots \ \alpha_m \quad \beta_1 \quad \beta_2 \ \ldots \ \beta_\ell]$$

The equation (2.6) can then be written as

$$y(t) = \beta_0 u(t-k-1) + \varphi(t-k-1)\theta + \varepsilon(t)$$

and the recursive equations of the estimates are given by

$$\hat{\theta}(t+1) = \hat{\theta}(t) + K(t)\left(y(t) - \beta_0 u(t-k-1) - \varphi(t-k-1)\hat{\theta}(t)\right) \quad (2.11)$$

$$K(t) = P(t)\varphi(t-k-1)^T\left(1 + \varphi(t-k-1)P(t)\varphi(t-k-1)^T\right)^{-1} \quad (2.12)$$

$$P(t+1) = P(t) - K(t)\left(1 + \varphi(t-k-1)P(t)\varphi(t-k-1)^T\right)K(t)^T \quad (2.13)$$

To update the estimates the equations (2.11) - (2.13) have to be iterated one step. The algorithm thus requires a moderate amount of computation. Computation times for the basic self-tuning algorithm are given in [8].

If the estimates obtained in Step 1 converge, it is possible to characterize the closed loop system. In [8] the following two theorems are stated and proved.

Theorem 2.1. Assume that the estimates of the parameters $\alpha_i(t)$, $i = 1, \ldots, m$, $\beta_i(t)$, $i = 1, \ldots, \ell$, in the model (2.6) converge as $t \to \infty$ and that the closed loop system is such that the output is ergodic (in the second moments) when using the control law (2.7). Then the closed loop system has the properties

$$r_y(\tau) = Ey(t+\tau)y(t) = 0 \qquad \tau = k+1, \ldots, k+m \quad (2.14)$$

$$r_{yu}(\tau) = Ey(t+\tau)u(t) = 0 \qquad \tau = k+1, \ldots, k+\ell+1 \quad (2.15)$$

■

If the system to be controlled is governed by an equation like (2.1) it is possible to show that the conditions (2.14) and (2.15) in essence imply that the self-tuning regulator will converge to a minimum variance regulator.

Theorem 2.2. Let the system to be controlled be governed by the equation (2.1). Assume that the self-tuning algorithm is used with $m = n$ and $\ell = n+k-1$. If the parameter estimates converge to values such that the corresponding polynomials $\mathcal{A}$ and $\mathcal{B}$ have no common factor then the corresponding regulator (2.8) converges to a minimum variance regulator. ■

If the noise is white, i.e. $C(q) = q^n$ it is not surprising that the regulator can converge to the optimal regulator. The self-tuning regulator can, however still converge to the optimal regulator for a general C-polynomial. This is unexpected since if $C(q) \neq q^n$ the least squares estimates will be biased [3]. Theorem 2.2 now states that the estimates will be biased in such a way that the regulator will be optimal. The class of systems for which the basic self-tuning algorithm can be used is thus larger than could be expected at first.

It should be pointed out that the main object of the algorithm is not to minimize the variance of the output but to make certain covariances and cross-covariances equal to zero. The minimum variance regulator that can be obtained in some cases is a concequence of the conditions (2.14) and (2.15).

It is not necessary that the disturbance e(t) is a stochastic process. Let the disturbance be a deterministic signal and replace the ergodicity assumption in Theorem 2.1 with the assumption that the closed loop system is stable. If the estimates converge the conditions (2.14) and (2.15) will be changed to

$$\lim_{N\to\infty} \frac{1}{N} \sum^{N} y(t+\tau)y(t) = 0 \qquad \tau = k+1, \ldots, k+m$$

$$\lim_{N\to\infty} \frac{1}{N} \sum^{N} y(t+\tau)u(t) = 0 \qquad \tau = k+1, \ldots, k+\ell+1$$

The regulator parameters will, however, converge to different values depending on the properties of the deterministic signal. Deterministic disturbances will be discussed in Section 8.

3. SELECTION OF THE PARAMETERS IN THE ALGORITHM.

The performance of the basic self-tuning algorithm is influenced by the choice of some parameters. This choice has theoretical as well as practical implications. The theoretical aspects mainly concern the relationship between the structure of the regulator and the model of the process to be controlled. From a practical point of view the main task is to make the regulator work well on a real process. This implies that the regulator must operate satisfactorily in spite of minor nonlinearities in the process, changes in the environment etc. For a known process it is easy to determine the parameters of the algorithm. It is, however, also necessary that the parameters are easy to determine in practice when the regulator is used on a real process. Furthermore it is desirable that the choice of the parameters is not crucial.

The basic self-tuning regulator is specified by the following parameters:

- exponential forgetting factor, λ,
- initial values for the least squares estimator,
- allowed maximum value of the control signal,
- scale factor in the regulator, β_0,
- number of regulator parameters, m, ℓ,
- number of pure time-delays in the model, k,
- sampling time.

The transient and the asymptotic properties of the system are influenced by the choice of these parameters. Theoretical as well as empirical rules of thumb will be given which can facilitate the practical use of the self-tuning regulators. Experience has indicated that it is much easier to select the parameters given above than to select the parameters θ of the control law directly.

3.1. Exponential Forgetting.

The estimates may sometimes converge very slowly, see e.g. Fig. 3.2a below. The magnitude of the changes of the estimates in each step depends on the gain in the least squares estimation algorithm, K(t), see (2.12). The gain K(t) decreases with time. It may happen that the gain becomes too small before the estimates are close to the final values. This is e.g. the case if the input signals are large in the first steps. One obvious remedy is to prevent K to become too small. This can be done by introducing an exponential weighting of past data in the least squares estimation [30]. The ordinary least squares estimator minimizes the loss function $\Sigma\varepsilon(t)^2$. The new loss function

$$V(t) = \sum_{n=0}^{t} \lambda^{t-n}\varepsilon(n)^2 \qquad \lambda \leq 1 \qquad (3.1)$$

gives more weight to recent data and less to past. The estimator which minimizes (3.1) is described by the equations (2.11), (2.12) and

$$P(t+1) = \frac{1}{\lambda}\Big[P(t) - K(t)\big(1 + \varphi(t-k-1)P(t)\varphi(t-k-1)^T\big)K(t)^T\Big] \qquad (3.2)$$

The weighting given by (3.1) can be interpreted as sending $\varepsilon(t)^2$ through a first order filter

$$V(t) = \lambda V(t-1) + \varepsilon(t)^2$$

while $\lambda = 1$ (ordinary least squares) corresponds to an integrator.

Figure 3.1 shows how old data are weighted for different values on λ. The weight is λ^n after n steps. The weight has decreased to 0.1 for

$$n = \ln 0.1/\ln \lambda \approx 2.3/(1-\lambda)$$

This can be used as a rule of thumb to determine how many values should be remembered. About 200 old values are remembered for $\lambda = 0.99$, while when only 40 are remembered for $\lambda = 0.95$.

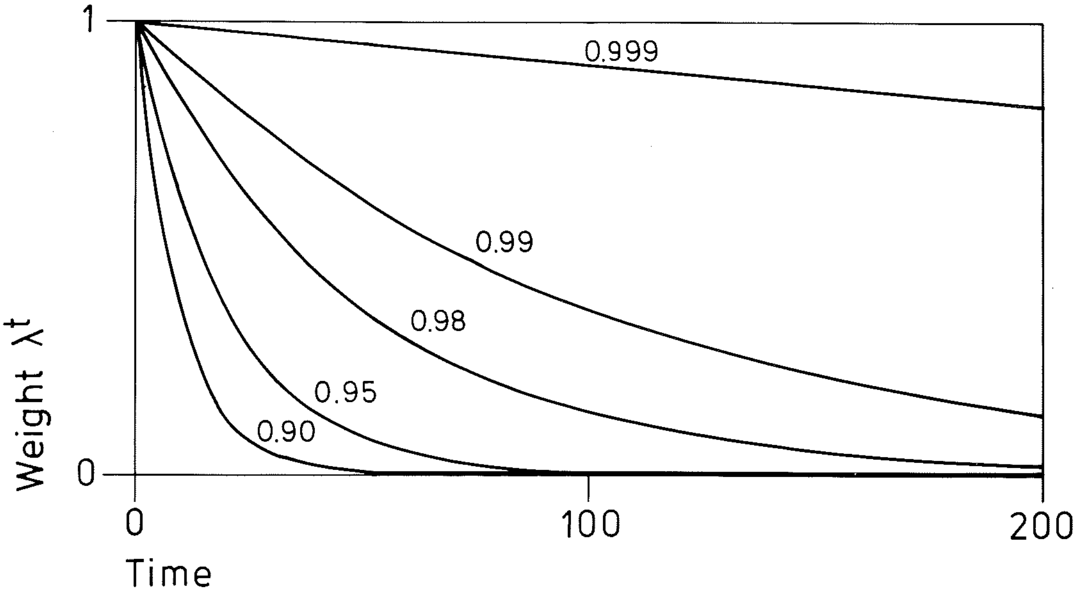


Fig. 3.1 - The weighting function λ^t for different values on λ.

The value of λ will influence both the transient and the stationary behaviour of the estimates. This is illustrated by the following example.

Example 3.1. Consider the system

$$y(t) - 0.9y(t-1) = 0.5u(t-1) + e(t)$$

The minimum variance regulator is given by

$$u(t) = - 1.8y(t)$$

The system was simulated with $\beta_0 = 1$ using different values of the exponential weighting factor λ. Figure 3.2 shows the estimated parameter in the model

$$y(t+1) + \alpha y(t) = u(t) + \varepsilon(t+1)$$

for $\lambda = 1.00$, 0.99 and 0.95 respectively. By decreasing λ the estimate converges faster to a neighbourhood of the optimal value. But the estimate will also become more "noisy" for smaller values on λ. This will influence the control in the long run. Table 3.1 shows the average losses for the first 200 steps and for the interval t = = 801 - 1000 when using different values on λ.

λ	$\frac{1}{200}\sum_{1}^{200} y(t)^2$	$\frac{1}{200}\sum_{801}^{1000} y(t)^2$
1.00	1.18	1.02
0.99	1.15	1.01
0.95	1.10	1.03
0.90	1.10	1.05
0.80	1.13	1.10
Min. variance controller	1.04	1.01
No control	3.70	5.27

Table 3.1 - Average loss for the system in Example 3.1 using different values of the exponential forgetting factor λ.

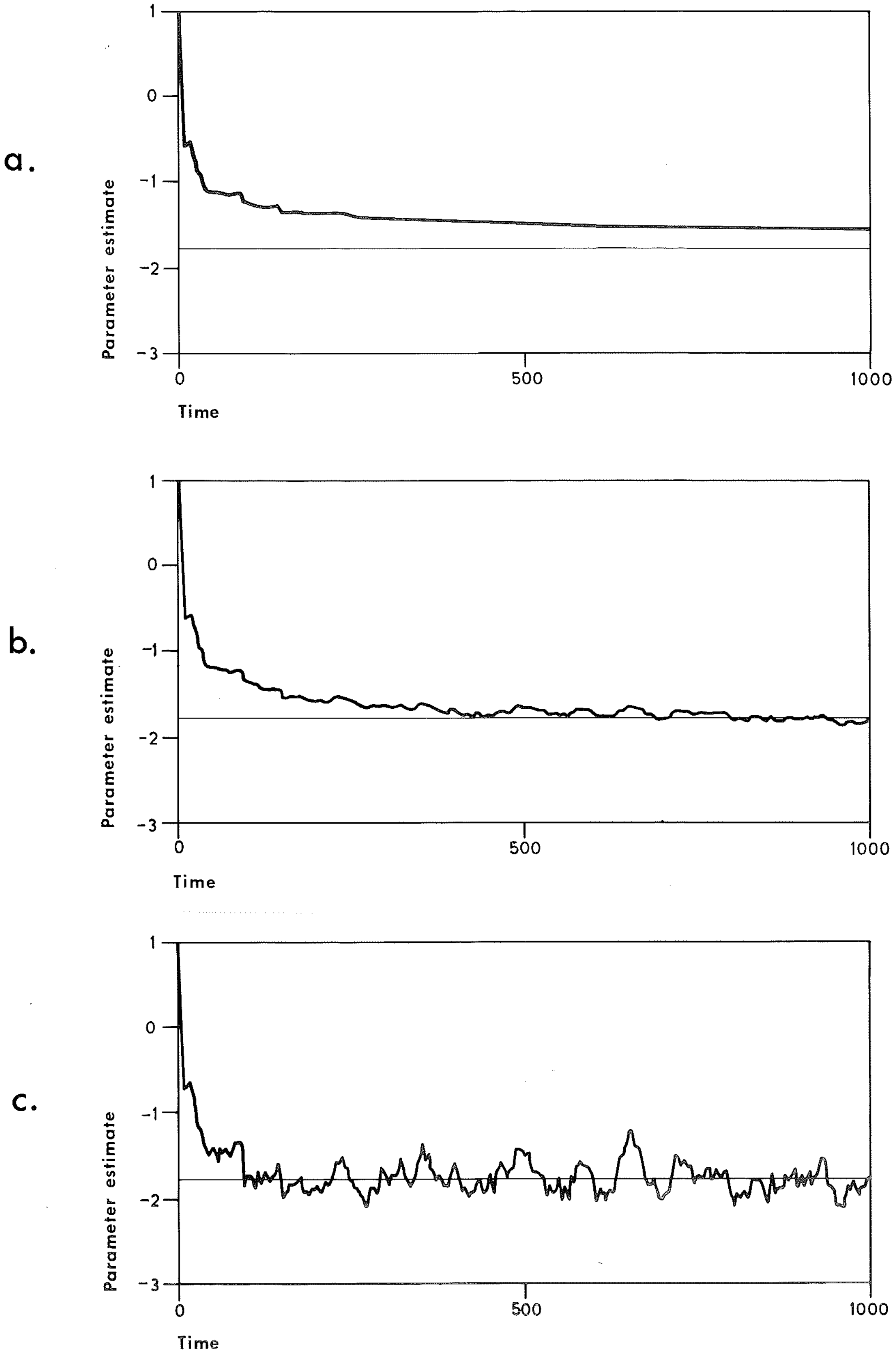


Fig. 3.2 - Estimated parameter when controlling the system in Example 3.1 using different values of the exponential forgetting factor λ.

a. $\lambda = 1.0$ b. $\lambda = 0.99$ c. $\lambda = 0.95$ ■

The experience based on many examples is that a λ-value of about 0.90 - 0.95 gives the best transient performance, while a value close to one gives the best stationary performance.

Example 3.1 shows that the choice of λ must be a compromise between the desired rate of convergence and the control in the long run, a typical dilemma of adaptive controllers.

In a practical case it must be taken into account that the characteristics of the process and the disturbances may change with time. Using an exponential weighting factor which is less than one it is possible to follow slowly time-varying parameters. The value of λ must then be adapted to the rate of change in the parameters. Time-varying systems are further discussed in Section 6.

The parameter λ is best chosen by inspecting the variations in the parameter estimates. Too much fluctuations implies either that too many parameters are used in the regulator or that a too small a value of λ is used. If the parameter estimates are almost constant it is advisable to make a small decrease in the value of λ occasionally. This should be done in order to make sure that the estimates really have converged and not only are slowly changing, compare Fig. 3.2a.

When implementing self-tuning regulators it is desirable that the parameter λ can be changed. The change of λ can be done manually by studying the estimated parameters. The parameter λ can also be changed automatically according to a time schedule. To make this time schedule requires, however, good knowledge about the process.

3.2. Initial Values.

Initial values of the parameters α_i and β_i and the covariance matrix, P(t), must be given when starting up the least squares estimator. The matrix P(0) shall reflect the confidence in the initial values of the parameters α_i and β_i. The units by which the inputs and the outputs are measured will, of course, influence the values of the parameters. A good practice is to choose the units in such a way that the inputs and the outputs have approximately the same magnitude.

If the parameters are completely unknown a standard choice can be to put them equal to zero. In this case the covariance matrix shall be given a rather large value, say $10 - 100 \times I$, where I is a unit matrix. If P(t) is large then the estimator is allowed to take large steps and the estimator can find fairly good estimates already after a few steps. The choice of P(0) is not crucial in this case.

If the parameters in the system are approximately known these values can be used as initial values for α_i and β_i. P(0) can then be given a small value which means that the estimator shall have a large confidence in the initial parameter values. Typically P(0) can be chosen as 10^{-1} - - 10^{-4} times a unit matrix.

The exponential forgetting factor gives a smallest value to the gain K(t) in the estimator. This can be used to increase the convergence rate. Using $\lambda < 1$ has, however, the drawback that the asymptotic properties are influenced. The estimates will no longer converge but will vary around the stationary values that would be obtained if $\lambda = 1$. Another way to increase the convergence rate is to restart the estimation using the obtained parameter estimates as initial values. Through the initialization the P-matrix can be given a larger value than before and the estimates of the parameters can again take larger steps. This method has the advantage that $\lambda = 1$ can be used. Also notice that it follows from (3.2) that a sudden reduction of λ has the same effect as a sudden increase of P. A very low value of λ during the first steps thus has the same effect as to use a large value of P(0).

To summarize the choice of initial values for the estimator is not crucial. A too small value on P(0) can, however, give rise to a slow rate of convergence. In order to avoid this the exponential forgetting factor can be given a value somewhat less than one as discussed above.

3.3. Limitation of Control Signal.

Many control systems have nonlinearities. There may be saturation, static friction and dead zones in the actuators. Experience has shown that the performance of the self-tuning regulator in the presence of such nonlinearities can be improved by exploiting the information about these effects. No detailed analysis will be given let a heuristic argument suffice. The idea is simply that the self-tuning regulator must know the signal that actually is acting on the process. The variable u(t) used in the estimation and in the computation of the control signal should be the process input but is actually the actuator signal. In the presence of actuator nonlinearities there may be a considerable difference. This can to some extent be eliminated if the process input is measured. If this is not done an alternative is to provide a simple static model for the nonlinearities. This has also the advantage that the dynamics of the actuator is taken into consideration when estimating the parameters in the model of the process. For example, if the actuator saturates it is often a good idea to limit the control signal before it is sent to the actuator. There are also other good reasons to limit the output signal. The control signals or the rate of change in the control signals are in many cases limited for security.

It has been shown in simulations that a limitation of the control signal in many cases has good influence on the transient behaviour of the self-tuning regulators. This is the case if the initial estimates of the parameters are poor. The regulator might then send large signals into the process which can cause large errors initially. The control signal can thus be limited to get a smooth start-up. The limit can be small in the beginning and later increased when better parameter estimates are obtained. Another way is to have a fixed limit which should not be reached under normal control of the process. If the maximum value is reached often it is not possible for the regulator to converge to the minimum variance regulator.

There are, however, two situations when the control signal should not be limited too hard. If the system is unstable it may be difficult to get good parameter estimates before the system has drifted far away from the reference value. A limit on the control signal may then make it impossible to stabilize the system. To handle this situation the process can be stabilized by a fixed regulator. The self-tuning regulator can then be used to tune the parameters in a regulator for the stabilized system.

If the noise is drifting large input signals may be needed to compensate for the drift. In this case the increment of the process input, $\nabla u(t) = u(t) - u(t-1)$, can be chosen as the control signal. The differences can now be limited.

Example 3.2. The effect of limiting the control signal will be shown using the following system:

$$y(t) - 0.9y(t-1) = 0.25u(t-2) + e(t)$$

The minimum variance controller is given by

$$u(t) = - \frac{3.24}{1 + 0.9q^{-1}} y(t) \qquad (3.3)$$

If $e(t) \in N(0,1)$ the minimum average loss is 1.81 per step and the variance of the optimal control signal is 10.5.

The parameters $\beta_0 = 1$, $m = \ell = 1$ and $\lambda = 0.99$ were used, when simulating with the self-tuning regulator. Curve a in Fig. 3.3 shows the accumulated loss, $\Sigma y(t)^2$, when $u(t)$ is not limited. The value used for β_0 is too large compared with the first coefficient in the B-polynomial of the system. This explains the large loss in the first 25 steps. After about 500 - 1000 steps the estimator has obtained good parameter estimates and the control is good.

Curve c shows the loss when $|u(t)|$ is limited to 5. The control signal reaches the limit approximately 40 times over the first 1000 steps. The loss function when using (3.3) and with $|u(t)|$ limited to 5 is shown by curve d. If the control signal is limited harder the controller has not "authority" enough to make as good control as before. Curve b in Fig. 3.3 shows the loss for the self-tuning regulator when the control signal is limited to 1. In this case the control signal is on the limit in almost every step.

Table 3.2 shows the average loss over the first 100 steps and over the interval 501 - 1000 when different limits are used for the control signal.

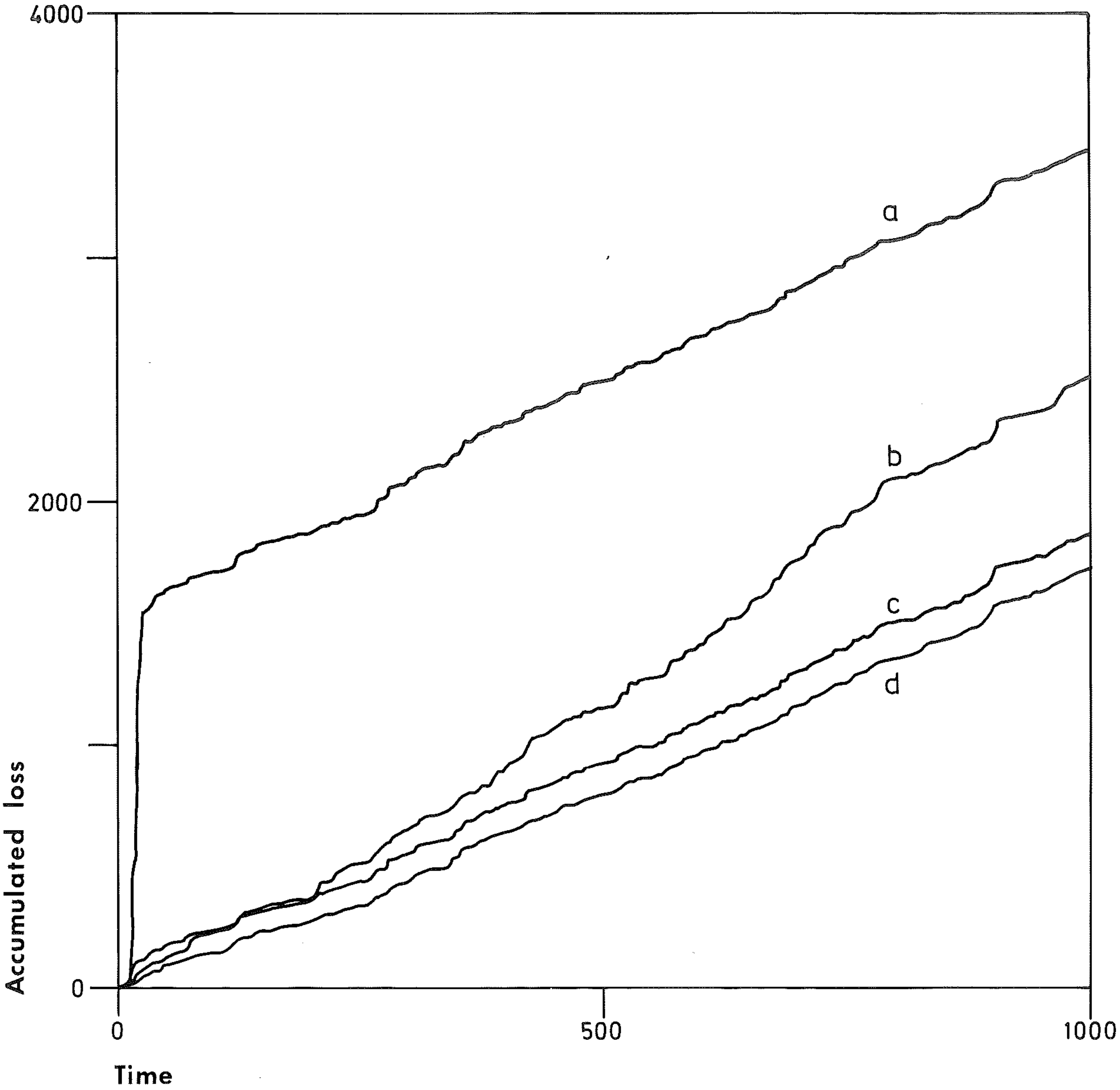


Fig. 3.3 - Accumulated loss, $\Sigma y(t)^2$, for different regulators and limitations of the input signal.

a.	Self-tuning regulator	no limitation
b.	- " - - " -	$|u| \leq 1$
c.	- " - - " -	$|u| \leq 5$
d.	Controller (3.3)	$|u| \leq 5$

Limit of control signal	$\frac{1}{100}\sum_{1}^{100} y(t)^2$	$\frac{1}{500}\sum_{501}^{1000} y(t)^2$
∞	17.07	1.88
5	2.41	1.87
2	2.00	2.20
1	2.28	2.71
0.5	2.73	3.42
Control law (3.3) with $\lvert u(t)\rvert \leq 5$	1.52	1.86

Table 3.2 - The average loss during different periods of time when controlling the system in Example 3.2 and using different limits of the control signal. ■

3.4. Scale Factor.

The parameter β_0 should ideally be an estimate of the parameter b_1, the first coefficient in the B-polynomial of the system (2.1). In [8] it is shown that the choice of β_0 is crucial for the convergence of the self-tuning algorithm when a long time horizon is used for the identification. For a first order system it is shown that the ratio β_0/b_1 must then be within the interval

$$0.5 < \beta_0/b_1 < \infty$$

Since the choice of β_0 is crucial, why not estimate β_0 too? In [8] it is shown that if a fixed control law is used then it might happen that the system is not identifiable [10]. But if the control law is time-varying the system can be identifiable. This is illustrated by the following example:

Example 3.3. Let the system be

$$y(t) - 2y(t-1) = u(t-1) + e(t)$$

Estimate the parameters α and β of the model

$$y(t+1) + \alpha y(t) = \beta u(t) + \varepsilon(t+1) \tag{3.4}$$

using the method of least squares and use the control law

$$u(t) = \frac{\alpha(t)}{\beta(t)} y(t)$$

If the control law is fixed it is not possible to identify both α and β. Simulations have, however, shown that it is possible to estimate both α and β if a time-varying control law is used. The convergence may in some cases be very slow.

The estimates of α and β from one simulation are shown in Fig. 3.4. The controller gain α/β is shown in Fig. 3.5. In this case the noise is white and α and β converge to the true values of a and b. If the noise is coloured the estimates will be biased but the controller will still converge to the minimum variance regulator. ■

The example above confirms that sometimes it is possible to identify β_0. The number of parameters that need to be estimated is, however, increased. The next example shows that the choice of β_0 is less crucial if the control signal is limited.

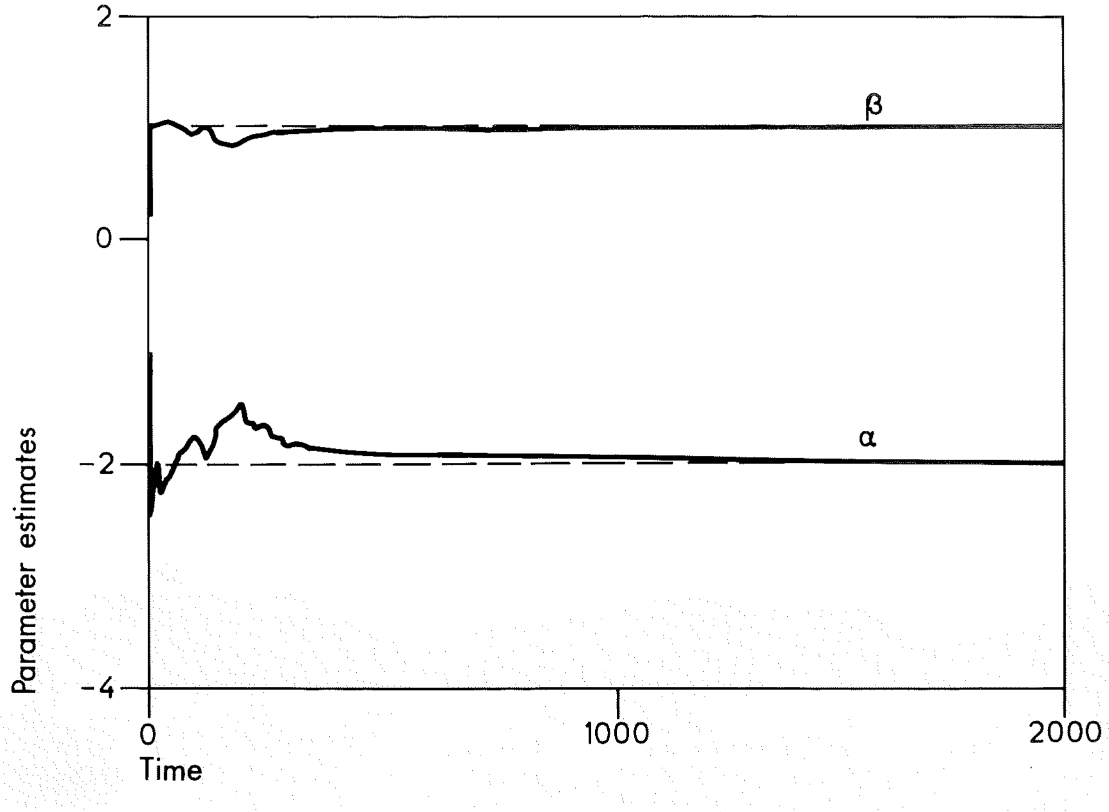


Fig. 3.4 - Parameter estimates of the model (3.4). The true values are shown by the dashed lines.

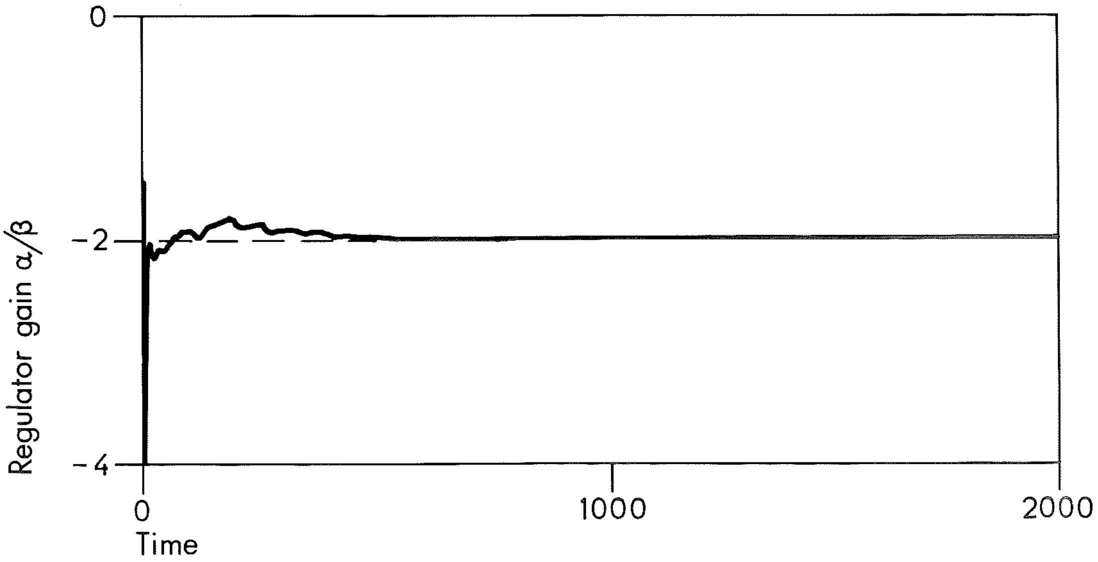


Fig. 3.5 - Regulator gain, α/β, for the model (3.4). The value of the optimal regulator is indicated by the dashed line.

Example 3.4. The system

$$y(t) - 1.5y(t-1) + 0.54y(t-2) =$$

$$= 2u(t-1) - 1.8u(t-2) + e(t) + 0.2e(t-1) - 0.48e(t-2)$$

is simulated for $m = 2$, $\ell = 1$ and different values of the ratio β_0/b_1. The initial values of the parameters are equal to zero except for $\beta_0/b_1 = 25$ and the control signal is limited to ±5. In Table 3.3 the average loss is shown for different values of β_0/b_1.

β_0/b_1	$\frac{1}{1000}\sum_{1}^{1000} y(t)^2$	$\frac{1}{500}\sum_{501}^{1000} y(t)^2$	Numb. of times u(t) is limited during the interval (0-50)	λ
0.05	2.71	1.05	23	1.00
0.2	1.44	1.04	4	1.00
1	1.11	1.04	1	1.00
5	1.12	1.04	0	0.95
10	1.39	1.05	0	0.95
25	1.58	1.14	0	0.95
Minimum variance controller	0.98	1.02	0	

Table 3.3 - Average loss for different values on the ratio β_0/b_1. For $\beta_0/b_1 = 25$ the initial values were $\alpha_1 = -20$, $\alpha_2 = 15$ and $\beta_1 = -0.5$, in all other cases the initial values were zero. The different values of the exponential weighting factor, λ, used are also shown in the table.

The ratio β_0/b_1 is thus varied from 0.05 to 25 and the system still has acceptable behaviour. If β_0/b_1 is further decreased the system becomes more difficult to control since the α_i parameters in the controller will be very small and the system will be more sensitive to variations in the α_i parameters. If β_0/b_1 is large the α_i parameters will be large too. This explains why initial values not equal to zero were chosen for $\beta_0/b_1 = 25$. ■

To summarize if β_0 is assumed known its value is not crucial for the convergence if the control signal is limited. The transient behaviour seems to be best if β_0 and b_1 are of the same magnitude.

After an experiment it is possible to judge if a sensible value of β_0 has been used. Suppose that the control signal is limited in such a way that the limit is seldom reached after the parameters have converged. The number of times the control signal hits the limit can be used to determine if a good value of β_0 has been used. If β_0 is too small, then the control signal use to reach the limit more often, than if too large a β_0 is used, compare Table 3.3. Too large a β_0 will on the other hand give very small input signals in the beginning. Further the α_i parameters use to be large if β_0 is too large.

3.5. Number of Regulator Parameters.

Theorems 2.1 and 2.2 show how the number of parameters in the regulator influences the asymptotic properties of the closed loop system. If $m = n$, $\ell = n+k-1$ and the correct k is used in the identification then if the estimation converges the obtained controller will be the minimum variance controller. What might happen if k does not have the correct value is discussed in Section 3.6.

If m or ℓ is too large then Theorem 2.2 is still valid. In that case simulations have shown that the extra parameters will converge to zero.

If m $>$ n and ℓ $>$ n+k-1 then Theorem 2.2 is not valid since if the estimates converge then the regulator will contain a common factor. Simulation with too large values on m and ℓ have shown that the algorithm still seems to converge, at least in the sense that the parameter estimates are practically constant. The control law will then contain a common factor. It is not possible to decide if the estimates and thus the common factor will slowly change, but the remaining part of the regulator is constant. The performance of the system when $\lambda < 1$ indicates that this may be the case. Using $\lambda < 1$ can result in large fluctuations in the estimates but the closed loop system may still have a near optimal performance.

It is possible to investigate if the number of regulator parameters is sufficient by computing the covariance function of the output, $\hat{r}_y(\tau)$. If $\hat{r}_y(\tau)$ is equal to zero for all $\tau \geq k+1$ then m and ℓ are sufficiently large. If some $\hat{r}_y(\tau)$:s for $\tau > k+m$ differ from zero the number of parameters has to be increased.

The covariance function has to be computed from a realisation of the process and it cannot be assumed that $\hat{r}_y(\tau)$ is exactly equal to zero for $\tau \geq k+1$. It is, however, possible to make a hypothesis test. If $\{x(t),\ t = 1,2,\ldots,N\}$ is a sequence of independent stochastic variables then for $\tau \neq 0$ the normalized computed covariance $\hat{r}_x(\tau)$ with probability 0.95 is within the interval $\pm 1.96/\sqrt{N}$ for large N. Now the outputs, y(t), are generally not independent, but the test can still be used for $\tau \geq k+1$ if y(t) is a moving average of order k of independent stochastic variables.

Sometimes the purpose of the control can be to make as good a control as possible using a regulator which is as simple as possible. In those cases Theorem 2.1 can be used to determine which covariances and crosscovariances that are equal to zero. The theorem gives, however, no indication of how much the output variance is increased if the number of regulator parameters is decreased. With the right structure of the regulator the algorithm will converge to the regulator giving the minimum output variance. But this is no longer the case if the number of regulator parameters is decreased. Let the number of parameters in the numerator and the denominator of the regulator be specified. It is then possible to use a minimization routine to determine the values of the parameters that minimize the output variance of the closed loop system. These parameter values will not be the same as those obtained when using a self-tuning regulator of the same structure. This was shown by an example in [8].

3.6. Number of Time-Delays.

The number of pure time-delays in the model, k, is the most crucial parameter to choose. Too large a value of k can give an output variance which is larger than necessary. Sometimes it is also difficult to get the estimates to converge. Too small a k can make it very difficult to get good control and even make it impossible to stabilize the system. This is intuitively clear. Assume that the estimator uses a time-delay, k, which is smaller than the true value, k_0. The controller then tries to make the predicted output k+1 steps ahead equal to zero using the control signal u(t). But u(t) will influence the output first at time $t+k_0+1$. Thus to make the predicted output at time t+k+1 equal to zero the controller will use large input signals, since the effect of u(t) on y(t+k+1) is small. If the control signal is limited then u(t) will reach the limit in almost every step. If $k < k_0$ and the regulator is started with initial values of the regulator parameters near the values of the minimum variance regulator then the control may be good for some period of time. The estimates may, however, drift away from the optimal values and the system can become unstable.

The following example illustrates the effect of choosing the wrong time-delay.

Example 3.5. Consider the system

$$y(t) - 1.6y(t-1) + 0.8y(t-2) = u(t-2) + 0.5u(t-3) + e(t) \quad (3.5)$$

The expected minimum loss is 3.56 per step if a minimum variance regulator is used. Without any control the loss is about 14 per step. The system (3.5) is controlled using the self-tuning regulator with k = 1, 2, 3 and 4 and with $m = \ell = 2$. The sample covariance functions $\hat{r}_y(\tau)$, when $u(t) \equiv 0$ and for k = 1, 2 and 4 respectively have been computed over 5000 steps of time and are shown in Fig. 3.6. From Theorem 2.1 it is expected that $\hat{r}_y(\tau)$ is equal to zero for $\tau = k+1$ and $k+2$. It turns out that $\hat{r}_y(\tau)$ is within the 5% test limit for $\tau \geq k+1$. The output covariances in the different cases are

k	$\hat{r}_y(0)$
1	3.59
2	3.97
4	3.62

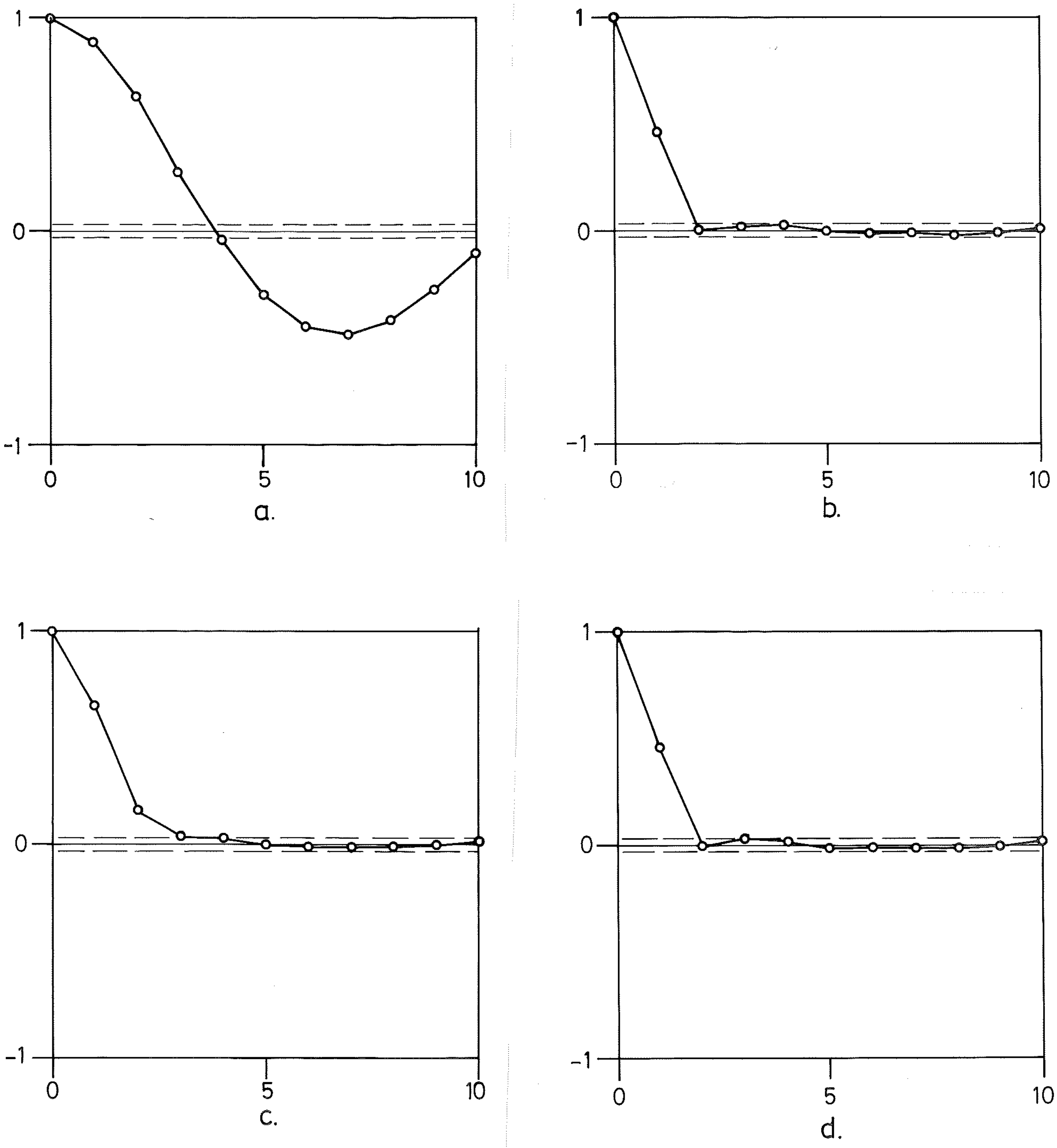


Fig. 3.6 - Estimated covariance function of the output of the system in Example 3.5. The dashed lines show the 5% limits to test if $\hat{r}_y(\tau) = 0$.

a. Without control b. k = 1 c. k = 2 d. k = 4

The value $\beta_0 = 1$ was used for k = 1. For k = 2 and 4 it was necessary to use $\beta_0 = 5$ in order to get convergence. For k = 3 it was impossible to get convergence of the parameters in spite of many attempts of selecting β_0, λ and the initial values. A typical behaviour of the parameter estimates for k = 3 is shown in Fig. 3.7. The dashed lines correspond to the parameter values of the minimum variance regulator. It seems as if the estimator tries to reach the values of the minimum variance regulator. When the estimates approach these values they are forced away to small values. The parameter values corresponding to the minimum variance regulator have also been used as initial values, but the estimates drifted away from this set of parameters and started to behave as in Fig. 3.7. Even if the estimates did not converge for k = 3 the control was fairly good. Over a period of 7500 steps the average loss was 5.44 which can be compared with 14 without any control and 3.56 for minimum variance control.

As expected the controller converges to the minimum variance regulator for k = 1. In that case the output is a moving average of first order. For k = 2 the controller seems to converge to a regulator making the output to a moving average of second order. The control law after 6000 time-steps was

$$u(t) = \frac{-1.16 + 0.94q^{-1}}{1 + 1.57q^{-1} + 0.60q^{-2}} y(t)$$

This regulator is practically the same as the suboptimal minimum variance regulator described in [1] which is obtained from the following identity

$$1 = (1-1.6q^{-1}+0.8q^{-2})(1+f_1q^{-1}+f_2q^{-2}) +$$

$$+ q^{-2}(g_0+g_1q^{-1})(1+0.5q^{-1})$$

This gives the controller

$$u(t) = \frac{-1.176 + 0.934q^{-1}}{1 + 1.6q^{-1} + 0.584q^{-2}} y(t)$$

and the closed loop system

$$y(t) = (1+1.6q^{-1}+0.584q^{-2})e(t)$$

The covariance function for this system is in agreement with the covariance function obtained in the simulation with k = 2.

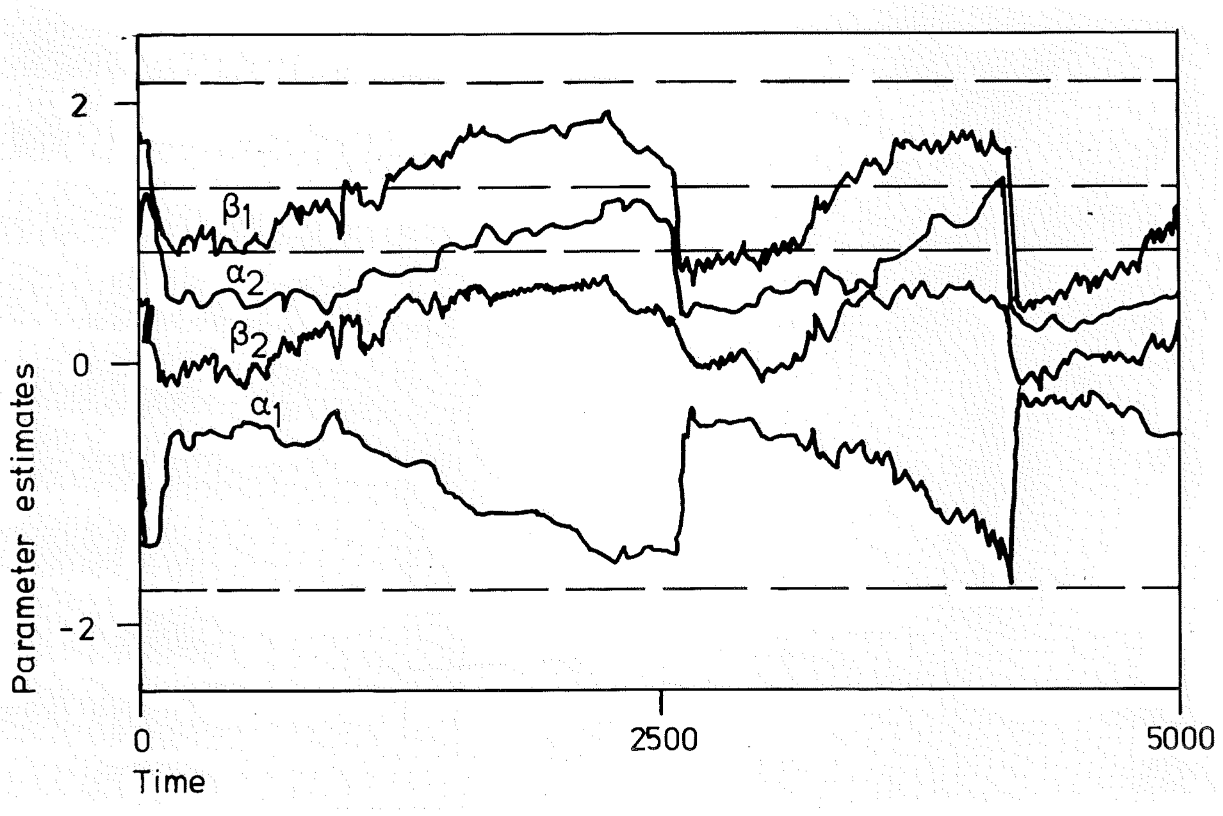


Fig. 3.7 - Parameter estimates of the system in Example 3.5. k = 3 is used in the identification and λ = 0.995. The dashed lines show the parameter values, corresponding to the minimum variance regulator.

For $k = 4$ the controller again converges to parameter values which give a moving average of first order, i.e. the controller again converges to the minimum variance regulator.

The example shows that the algorithm can converge to different controllers depending on the number of time-delays that is chosen in the parameter estimator. Figure 3.8 shows what might happen if too small a value on k is used. For $k = 0$ the controller is started with parameter values which give the minimum variance controller, $\alpha_1 = -1.76$, $\alpha_2 = 1.28$, $\beta_1 = 2.1$ and $\beta_2 = 0.8$. The starting value of the covariance matrix was 0.005 times a unit matrix. The exponential forgetting factor was $\lambda = 1$ and the control signal was limited to 10. The parameters move away from the initial values and after about 500 steps the closed loop system becomes unstable. The same initial values but with k changed to 2 gives the loss shown by curve b. For comparison the loss when using the minimum variance controller is also shown, curve c. ■

In the example above the closed loop system had the property that if the parameter estimates converged then $\hat{r}_y(\tau) = 0$ for $\tau \geq k+1$. This experimental observation can be formalized in the following theorem which is a slight modification of Theorem 2.2.

Theorem 3.1. Let the system to be controlled be governed by the equation

$$A(q)y(t) = B(q)u(t-k_0) + C(q)e(t)$$

where A, B and C are polynomials of degree n, n-1 and n respectively. Assume that the self-tuning algorithm is

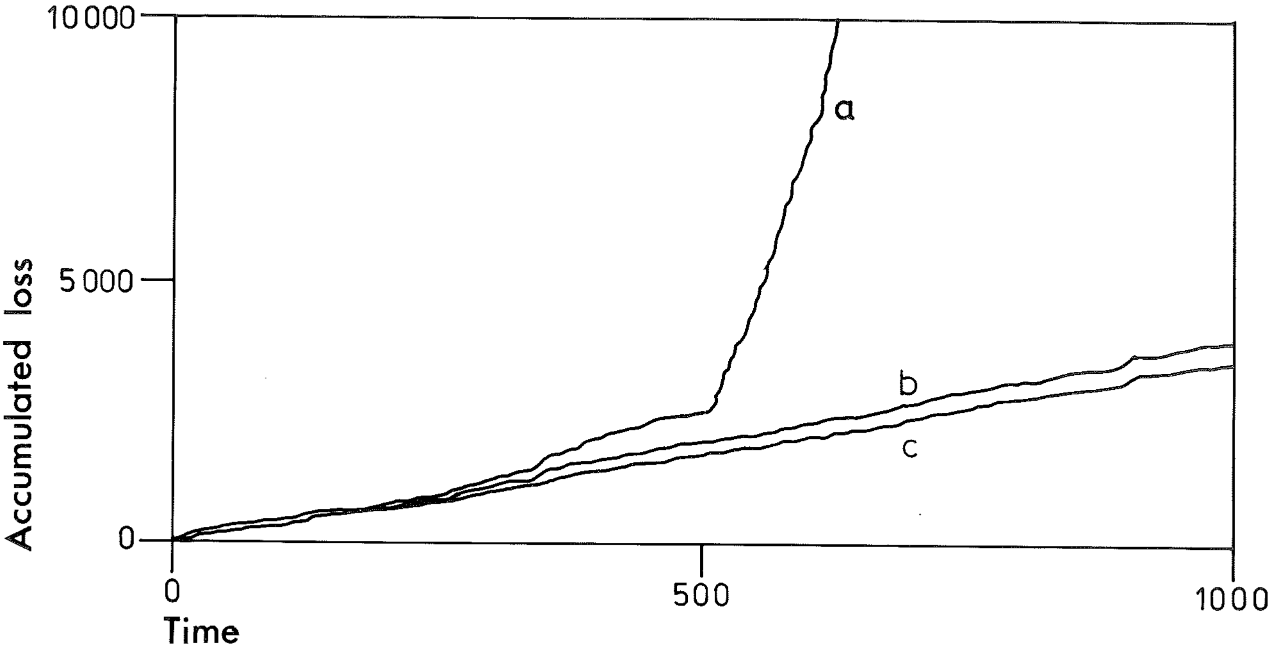


Fig. 3.8 - The accumulated loss, $\Sigma y(t)^2$, when controlling the system (3.5) with different regulators. Curves a and b show the losses when using the self-tuning regulator with k = 0 and k = 2 respectively. Curve c shows the loss when using the minimum variance regulator.

used with m = n and $\ell = n+k_0-1$. If the parameter estimates converge to values such that the corresponding polynomials $\mathcal{A}$ and $\mathcal{B}$ have no common factor, then the output of the closed loop system has the property that

$$r_y(\tau) = 0 \qquad \tau \geq k+1$$

where $k \geq k_0$ is the number of pure time-delays used for the identification of the regulator parameters.

Outline of the proof. The proof of Theorem 2.2, given in [8], can be copied step by step. Theorem 2.1 shows that $r_y(\tau) = 0$ for $\tau = k+1, \ldots, k+m$. Using the same arguments as in [8] it can be shown that $r_y(\tau) = 0$ for $\tau \geqslant k+1$, i.e. y(t) is a moving average of maximum order k. ■

Remark 1. Theorem 3.1 is valid under the assumption that the estimates actually converge. Example 3.5 shows that there may be difficulties with the convergence for some values of k. The convergence of the algorithm is further discussed in Section 4.

For different values on k the closed loop system can be a moving average of different order. It is not necessary that a larger k will imply that the moving average will be of higher order.

To summarize it is critical if k is chosen too small, but less critical if k is one or two sampling intervals too large. Estimates of k can be obtained through experiments.

3.7. Sampling Time.

The choice of sampling rate is crucial when implementing the self-tuning regulators. The sampling interval influences the number of time-delays as well as the parameters in the discrete time model of the process. The choice of sampling interval can also change other properties of the system. For instance a continuous time system which is minimum phase can have a discrete time representation that is nonminimum phase [4]. This may be serious when using dead-beat and minimum variance regulators.

There are many aspects that must be considered when determining the sampling interval, e.g.

- the dynamics of the process
- the dead time in the process
- the complexity of the regulator
- computation times

How the choice of sampling interval influences the behaviour of the self-tuning regulators has not yet been investigated. The problem must, however, be carefully considered when making an implementation of a self-tuning regulator.

4. CONVERGENCE PROPERTIES OF THE SELF-TUNING REGULATORS.

It is difficult to analyze a system controlled by an adaptive regulator. The difficulties depend on the nonlinearities, the stochastic disturbances and the time-dependence of the closed loop system. The tools that can be used to prove convergence are for instance Lyapunov theory [23] and hyperstability concepts [18]. It should, however, be pointed out that most algorithms for which the convergence have been proved utilize the unknown parameters when the regulator is designed. It is the sensitivity derivatives of the system that are usually used which depend on the unknown parameters of the system. For the practical use of the algorithm it will probably not matter that the regulator does not use the true values of the sensitivity derivatives. But the convergence can only be proved under the assumption that the regulator uses the correct values of the sensitivity derivatives.

From a mathematical point of view it is necessary to assume that the parameters of the system are constant, otherwise the problem is too difficult to handle. The empirical result of simulations with the basic self-tuning regulator is that the algorithm converges in most cases from a practical point of view. It is impossible to determine if and when the algorithm really converges from simulation. It is, however, possible to prove convergence for some special cases. For instance it can be shown that the algorithm converges when the noise is white, i.e. $C(q) = q^n$. It is also possible to give examples which show that the basic algorithm does not converge in general. These examples were possible to construct only by a thoroughly analyse of the linearized estimator equations.

It is possible to derive a related set of differential equations from the difference equations of the estimator [20]. To prove convergence of the algorithm will then be the same as to prove stability of the differential equations. The differential equations are strongly nonlinear but the advantage is that the stochastic part of the problem is eliminated. To make the analysis easier a simplified algorithm will be introduced. The new algorithm is based on the idea of stochastic approximation. The behaviour of the basic and the simplified algorithms are compared. The solutions of the differential equations related to the simplified algorithm can be interpreted as the expected trajectories of the parameters in the regulator. The differential equations make it possible to show convergence for a first order system with one regulator parameter and to analyze how the number of time-delays used in the identification will influence the system. The differential equations for the regulator parameters have also been simulated for a system with two regulator parameters. That example gives valuable insight into the convergence properties of the self-tuning algorithm.

Convergence and asymptotic properties of the self-tuning regulators are further discussed in [20] and [21].

4.1. Convergence of the Basic Self-Tuning Algorithm.

The basic algorithm is derived heurically under the assumptions that identification and control can be separated and that the least squares method can be used for identification. The least squares estimate is unbiased only if the noise acting on the process is white, i.e. $C(q) = q^n$. Simulations have shown, however, that the algorithm in most cases also converges to the minimum variance regulator also when $C(q) \neq q^n$. When the noise is white it is straightforward to prove the convergence of the algorithm when the system is of arbitrary order and has an arbitrary but known time-delay.

Assume that the system to be controlled can be written as

$$A(q)y(t) = B(q)u(t-k) + e(t+n) \tag{4.1}$$

Using the identity

$$q^{k+n} = A(q)F(q) + G(q)$$

it is possible to rewrite (4.1) as

$$\begin{aligned} y(t+k+1) - q^{-(n-1)}G(q)y(t) &= \\ &= q^{-(n+k-1)}B(q)F(q)u(t) + \varepsilon(t+k+1) \end{aligned} \tag{4.2}$$

where

$$\varepsilon(t+k+1) = F(q)e(t+1)$$

When using the self-tuning algorithm the parameters in the model (2.6) are estimated using the method of least squares. The model (2.6) is of the same structure as (4.2) if $m = n$ and $\ell = n+k-1$. For systems of the form (4.2) the following theorem [21] holds.

Theorem 4.1. Consider the system

$$y(t+k+1) + \alpha_1 y(t) + \ldots + \alpha_m y(t-m+1) =$$

$$= \beta_0[u(t) + \beta_1 u(t-1) + \ldots + \beta_\ell u(t-\ell)] + \varepsilon(t+k+1) \quad (4.3)$$

where $\varepsilon(t+k+1)$ is a sequence of random variables with uniformly bounded fourth moments. Let $\varepsilon(t+k+1)$ be independent of $y(t)$, $y(t-1)$, ..., $u(t)$, $u(t-1)$, ..., $\varepsilon(t)$, $\varepsilon(t-1)$, ... Assume that β_0 is known. Let the system be controlled by the control law

$$u(t) = \frac{1}{\beta_0}[\hat{\alpha}_1 y(t) + \ldots + \hat{\alpha}_m y(t-m+1)] - \hat{\beta}_1 u(t-1) - \ldots \quad (4.4)$$

$$- \hat{\beta}_\ell \; u(t-\ell)$$

where $\hat{\alpha}_i$, $i = 1,..m$, $\hat{\beta}_i$, $i = 1,\ldots, \ell$ are the least squares estimates of the parameters in (4.3). Suppose that the closed loop system is such that

$$|y(t)| < C'$$
$$\quad (4.6)$$
$$|u(t)| < C''$$

C' and C" may depend on the realization. Then with probability one

$$\hat{\alpha}_i \to \alpha_i \qquad i = 1,\ldots, m$$

$$\hat{\beta}_i \to \beta_i \qquad i = 1,\ldots, \ell \quad \blacksquare$$

The theorem is proved in [21]. Let if suffice to make some remarks. The conditions (4.6) that $|y(t)|$ and $|u(t)|$ are bounded are physically relevant although these conditions are not satisfied if $\varepsilon(t)$ is normal distributed. The conditions (4.6) are stated in order to use the iterated logarithms law for martingales. There are mainly two reasons which make the proof somewhat difficult. Firstly, the output and input processes will in general be nonstationary thus the ergodic theorems usually relied upon when proving convergence are not applicable. Secondly the question of identifiability must be considered which is no easy problem for a timevarying feedback. In [21] it is shown that the system (4.3) is identifiable when using the control law (4.4). In a few words it can be said that the system is identifiable if the control law is sufficiantly complex.

The parameters in the model (2.6) will thus converge to the minimum variance regulator if ℓ, m, k and β_0 in the model (4.3) are known. The parameter β_0 can, however, be difficult to determine. The parameter β_0 in the basic algorithm was chosen to be fixed in order to avoid possible difficulties with identifiability. If β_0 is identified then there exists a linear manifold of parameter values that gives the optimal regulator. In [21] it is shown that the regulator will converge to the optimal regulator. Compare Example 3.3.

The difficulties to determine β_0 can be avoided by fixing another parameter in the system. One parameter of a system that is fairly easy to determine experimentally is the steady state gain. The steady state gain can be used as the parameter that is assumed to be known. This can be done by writing the regulator as a polynomial in $(1-q^{-1})$ instead as a polynomial in q^{-1}. This method is proposed in [29]. Since the choice of β_0 in practice does not seem to be crucial this idea is not stressed further.

In [20] and [21] it is shown that there exist ordinary differential equations associated with the difference equations of the estimator. The equations are nonlinear and difficult to analyze in a general case. Through linearization it has been possible to determine systems which make the linearized equations unstable. One example is given below.

Example 4.1. Consider the following system

$$y(t) - 1.6y(t-1) + 0.75y(t-2) =$$

$$= u(t-1) + u(t-2) + 0.9u(t-3) + e(t) + 1.5e(t-1) +$$

$$+ 0.75e(t-2)$$

The C-polynomial has negative real part for an interval on the unit circle. The A- and B-polynomials have been chosen in such a way that (A-C)/BC has a large gain for the frequencies when the C-polynomial has a negative real part.

The system has been controlled using the basic self-tuning algorithm with $m = 1$ and $\ell = 2$. Using $\beta_0 = 1$ it has not been possible to get convergence. Figure 4.1 shows the parameter estimates during 200 steps. The initial values were $\alpha_1 = -3.1$, $\beta_1 = 1$ and $\beta_2 = 0.9$ which corresponds to minimum variance regulator. The covariance matrix was

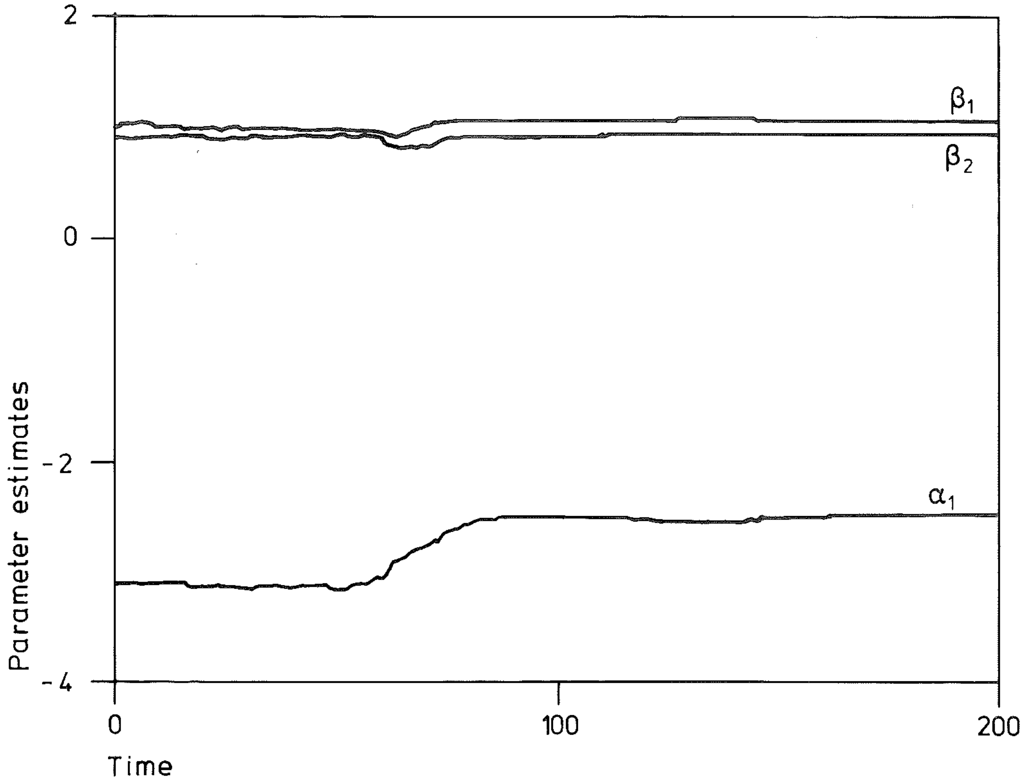


Fig. 4.1 - Parameter estimates when the system in Example 4.1 is control by the basic self-tuning regulator.

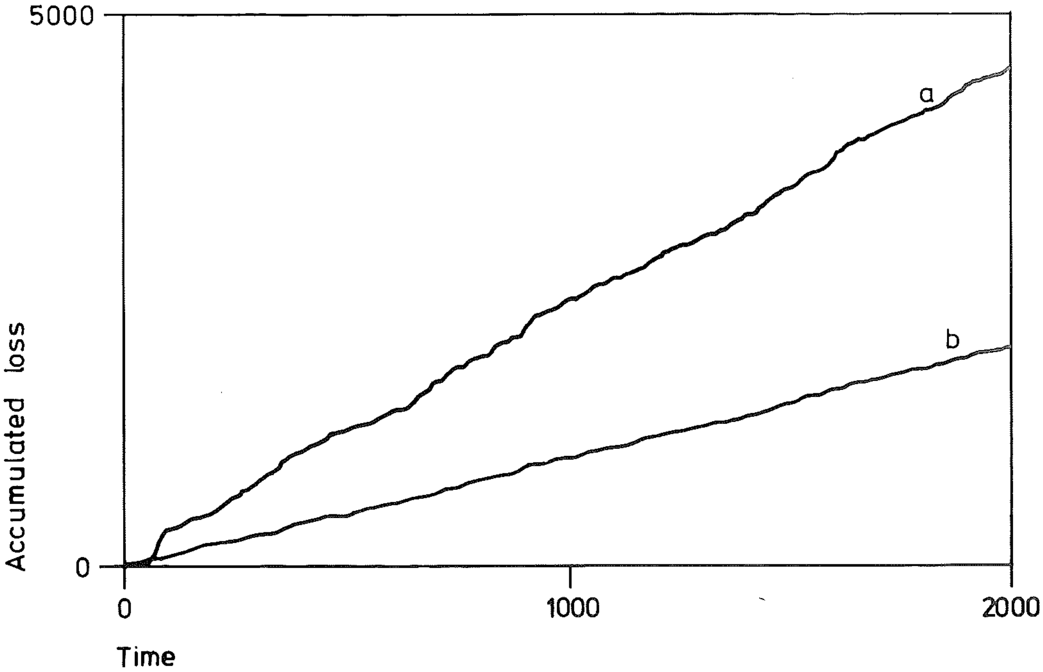


Fig. 4.2 - Accumulated losses of the system in Example 4.1 when using:

a. basic self-tuning regulator

b. minimum variance regulator.

$P(0) = 0.01 \times I$ and $\lambda = 1$. The estimates are forced away from the values corresponding to the minimum variance regulator. Since the parameters cannot converge to any other set of parameters they will vary continuously. The accumulated losses when using the basic algorithm and when using the optimal regulator are shown in Figure 4.2. ■

4.2. A Simplified Self-Tuning Algorithm.

A simplified self-tuning algorithm will now be introduced. The estimator is simplified by replacing the least squares estimate by an estimate based on stochastic approximation. For a survey of stochastic approximation methods see e.g. [28].

Consider the model

$$y(t+k+1) = \beta_0 u(t) + \varphi(t)\theta + \varepsilon(t+k+1) \qquad (4.6)$$

where $\varepsilon(t+k+1)$ is assumed independent of $y(t)$, $y(t-1)$, ..., $u(t)$, $u(t-1)$, ..., and where

$$\varphi(t) = [-y(t) \quad -y(t-1) \quad \ldots \quad -y(t-m+1) \quad \beta_0 u(t-1) \quad \ldots \quad \beta_0 u(t-\ell)]$$

$$\theta^T = [\alpha_1 \quad \alpha_2 \quad \ldots \quad \alpha_m \quad \beta_1 \quad \beta_2 \quad \ldots \quad \beta_\ell]$$

If β_0 is assumed known the parameter vector θ can be estimated using the following scheme

$$\hat{\theta}(t) = \hat{\theta}(t-1) + K(t)\left[y(t) - \hat{y}(t|t-k-1)\right] \tag{4.7}$$

where $\hat{\theta}(t)$ is the estimate of θ given data up to and including time t. $\hat{y}(t|t-k-1)$ is the prediction of y(t), based on data up to and including time t-k-1. Due to the assumption on $\varepsilon(t)$ the prediction is simply

$$\hat{y}(t|t-k-1) = \beta_0 u(t-k-1) + \varphi(t-k-1)\hat{\theta}(t-k-1) \tag{4.8}$$

The purpose of the control is to minimize the output variance and the following control law is used

$$u(t) = \frac{-1}{\beta_0}\varphi(t)\hat{\theta}(t) \tag{4.9}$$

Introduce (4.8) and (4.9) into (4.7), then

$$\hat{\theta}(t) = \hat{\theta}(t-1) + K(t)y(t)$$

The problem is how to choose the vector K(t). When using Kalman filtering or least squares estimation the estimator has the same structure as (4.7), and in those cases K(t) is proportional to $\varphi^T(t-k-1)$. But K(t) is also decreasing inversely proportional with time. The idea of stochastic approximation also gives an estimator, having the same structure. In that case the gain K(t) can be chosen to be proportional to 1/t. Hence choose

$$K(t) = \frac{\rho}{t}\varphi(t-k-1)^T$$

The parameter ρ can be interpreted as a step length.

The self-tuning algorithm will thus be to estimate the parameters in (4.6) using

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \frac{\rho}{t}\,\varphi(t-k-1)^T y(t) \tag{4.10}$$

and to use the control law (4.9).

For the simplified algorithm it is possible to state a theorem that is analogous to Theorem 2.1.

Theorem 4.2. Assume that the parameter estimates $\hat{\theta}(t)$ in (4.10) converge as $t \to \infty$ and that the closed loop system is such that the output is ergodic (in the second moments) when the controller (4.9) is used. The closed loop system then has the properties

$$r_y(\tau) = Ey(t+\tau)y(t) = 0 \qquad \tau = k+1,\ \ldots,\ k+m$$

$$r_{yu}(\tau) = Ey(t+\tau)u(t) = 0 \qquad \tau = k+1,\ \ldots,\ k+\ell+1$$

Proof. The estimate at time N can be written as

$$\hat{\theta}(N) = \hat{\theta}(0) + \rho \sum_{t=1}^{N} \frac{1}{t}\varphi(t-k-1)^T y(t)$$

Since the estimates are assumed to converge when $N \to \infty$ we get

$$\sum_{1}^{\infty} - \varphi(t-k-1)^T y(t) < \infty$$

Using Kronecker's lemma, see e.g. [16, p. 117] it follows that

$$\lim_{N\to\infty} \frac{1}{N} \sum_{1}^{N} \varphi(t-k-1)^T y(t) = 0$$

Then from the ergodic assumption

$$r_y(\tau) = 0 \qquad \tau = k+1, \ldots, k+m$$

$$r_{yu}(\tau) = 0 \qquad \tau = k+2, \ldots, k+\ell+1$$

Using the control law (4.9) it also follows that $r_{yu}(\tau) =$ $= 0$ for $\tau = k+1$ and the theorem is proven. ■

Using Theorem 4.2 and Theorem 2.2 it follows that the simplified algorithm converges to the minimum variance regulator if the estimation converges and if the number of parameters in the model is sufficient. The simplified algorithm thus has the same asymptotic properties as the basic algorithm. The simplified algorithm has in some cases a slower convergence rate than the basic algorithm, as can be expected. Variations in the step length, ρ, influence the convergence rate. Too small a ρ gives slow convergence while too large a ρ can make the system unstable if the control signal is not limited.

Example 4.2. Example 7.2 from [8] is used to compare the behaviour of the basic and the simplified algorithms. The system is influenced by drifting noise and is given by

$$y(t) - 1.9y(t-1) + 0.9y(t-2) = u(t-2) - u(t-3) + e(t) -$$
$$- 0.5e(t-1)$$

In the simulation the gain vector K(t) is normalized with the estimated output variance, i.e.

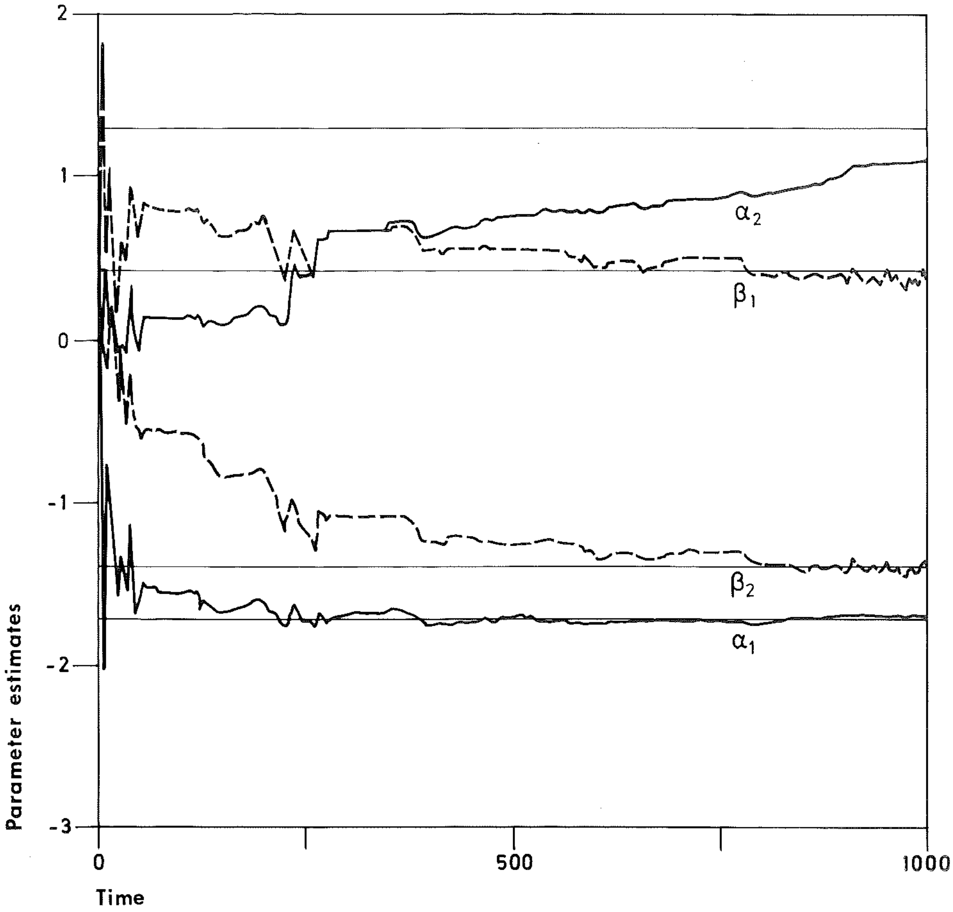


Fig. 4.3 - Parameter estimates when the system in Example 4.2 is controlled using the simplified algorithm (compare Figure 5 of [8]).

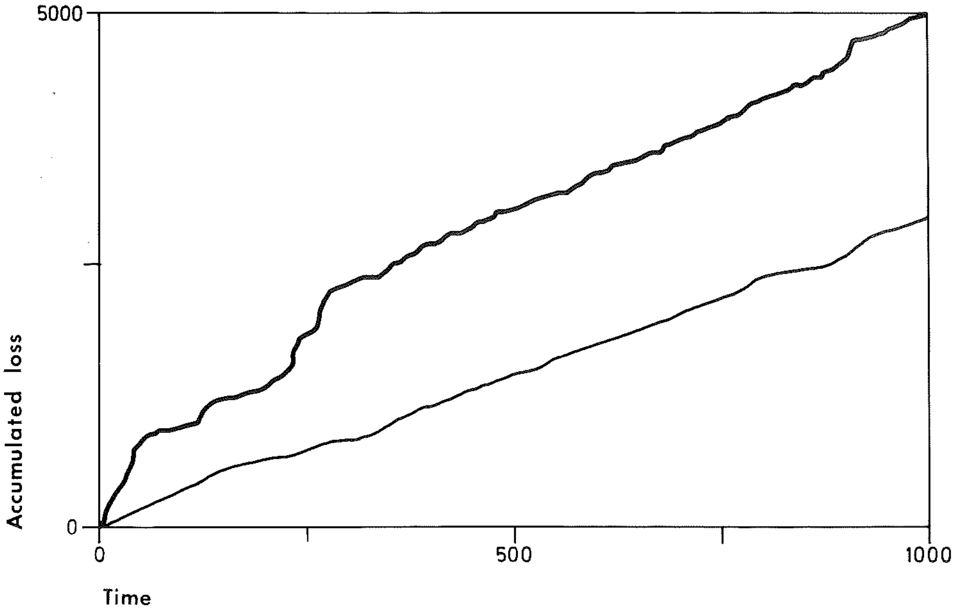


Fig. 4.4 - Accumulated loss for the system in Example 4.2 when using the simplified algorithm (thick line) compared with the basic algorithm (thin line).

$$K(t) = \frac{\rho}{\sum_{1}^{t} y(s)^2} \varphi(t-k-1)^T$$

The following parameter values were used $\rho = 3$, $\beta_0 = 1$, $m = \ell = 2$ and the initial values for the parameters were equal to zero. The estimates for the simplified algorithm is shown in Figure 4.3. When using the basic algorithm the parameters converge after about 200 steps while for the simplified algorithm all the parameters have not converged after 1000 steps. Figure 4.4 shows the accumulated losses when the basic and the simplified algorithm is used. ■

4.3. Convergence of the Simplified Algorithm.

The algorithm introduced in Section 4.2 has in principle the same properties as the basic algorithm. For instance the linearized estimator equations mentioned in Section 4.1 are the same as those for the basic algorithm. It is, however, easier to analyze some simple systems if the simplified algorithm is used. The main tool in this section is a theorem given in [21]. This theorem gives the differential equations for the expected trajectories of the parameter estimates and thus makes it possible to eliminate the stochastic part of the problem. In [8] a convergence proof is given for a first order system when using a long time horizon in the identification. For this simple system it is now possible to prove the convergence when the simplified as well as the basic self-tuning algorithm is used.

Example 4.3. Consider the system

$$y(t) + ay(t-1) = bu(t-1) + e(t) + ce(t-1) \qquad b > 0$$

The minimum variance regulator is given by

$$u(t) = \frac{a - c}{b} y(t)$$

The system is controlled using the simplified algorithm. The regulator parameter is estimated using the model

$$y(t+1) + \alpha y(t) = u(t) + \varepsilon(t+1)$$

i.e. it is assumed that $\beta_0 = 1$. The differential equation for the expected trajectories of the parameter α is according to [21] given by

$$\dot{\alpha} = - r_y(1) \tag{4.11}$$

where

$$r_y(1) = \frac{(c-a+\alpha b)\big(1 - c(a-\alpha b)\big)}{1 - (a-\alpha b)^2}$$

The right hand side of (4.11) has only one stable stationary point, $\alpha = (a-c)/b$ within the interval where $r_y(1)$ is defined. This point corresponds to the minimum variance regulator. Now

$$r_y(1) \begin{cases} > 0 & \text{if} \quad (a-c)/b < \alpha < (a+1)/b \\ < 0 & \qquad (a-1)/b < \alpha < (a-c)/b \end{cases}$$

Thus if the closed loop system is stable then

$$\alpha \to \frac{a - c}{b}$$

Notice that in this case it is not necessary to assume that $\beta_0 = b$. From (4.11) it can be seen that it is essential that β_0 and b have the same sign. Simulations have shown that it is not necessary to assume that the closed loop system is stable if the control signal is limited.

For the basic algorithm the differential equation corresponding to (4.11) will be

$$\dot{\alpha} = -f(t) r_y(1)$$

where $f > 0$ and depends on the actual realization of the estimate. The variable f(t) can be regarded as a time-varying time scale factor for the differential equation. As $f > 0$ it will not influence the stability properties of the differential equation. This implies that the basic algorithm also converges to the minimum variance regulator.

It is also possible to investigate how the number of time-delays used in the identification, k, will influence the behaviour of the closed loop system. For a general k the differential equation (4.11) for α will be changed to

$$\dot{\alpha} = - r_y(k+1) = - \frac{(\alpha b - a)^k (c - a + \alpha b)\left(1 - c(a - \alpha b)\right)}{1 - (a - \alpha b)^2} \qquad (4.12)$$

For $c > 0$ the principal shape of $r_y(k+1)$ is shown in Fig. 4.5 for k = 0, 1, 2 and 3 when a = -0.99, b = 1 and c = = 0.7. The differential equation now has two stationary

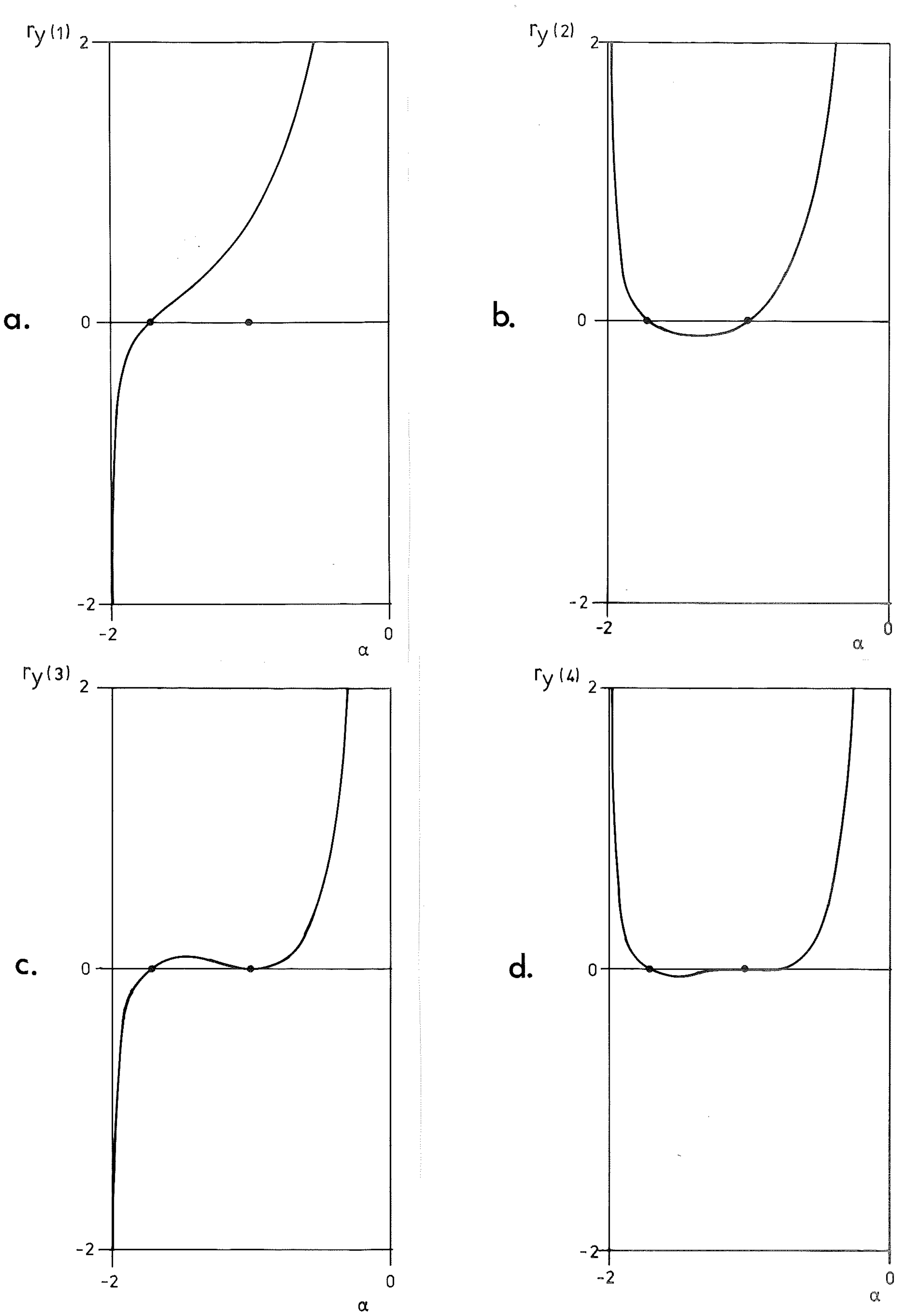


Fig. 4.5 - The covariance function

$$r_y(k+1) = \frac{(\alpha b-a)^k(c-a+\alpha b)\big(1 - c(a-\alpha b)\big)}{1 - (a-\alpha b)^2}$$

for different values on k and when a = -0.99, b = 1, c = 0.7.

a) k = 0 b) k = 1 c) k = 2 d) k = 3

points, one corresponding to the minimum variance regulator and one corresponding to the dead-beat regulator

$$\alpha = \frac{a}{b}$$

When $c > 0$ the parameter value $\alpha = (a-c)/b$ will be the stable solution to (4.12) if k is an even number. If k is odd then the algorithm will converge to the dead-beat regulator. If $c \leq 0$ then the minimum variance regulator all the time will be the stable solution of (4.12).

If for instance $k = 2,4,\ldots$ and $c > 0$ and the differential equation is started with a value $a/b < \alpha < (a+1)/b$ then α will converge to a/b, but a slightest perturbation of α in that point in the direction of decreasing α will give convergence to the stable point $\alpha = (a-c)/b$. When using the self-tuning algorithm the noise will all the time perturb the system. This will make the system converge to the minimum variance regulator. Simulations have shown that the estimate can for long time be in the surrounding of the dead-beat regulator. Simulations with basic algorithm have for this example given the same result as when the simplified algorithm has been used.

For the simple system in Example 4.3 it was possible to show convergence when not having white noise. Further it was possible to analyze what happens if the wrong number of time-delays is used in the identification.

The differential equations for the parameter estimates are difficult to analyze when the number of regulator parameters is increased. The differential equations for the expected estimates can, however, also be used to investigate how the number of pure time-delays influence the convergence properties when the regulator has two parameters.

Example 4.4. Consider the system

$$y(t) + ay(t-1) = u(t-1) + bu(t-2) + e(t) + ce(t-1)$$

The minimum variance regulator for this system has the structure

$$u(t) = \frac{\alpha}{1 + \beta q^{-1}} y(t)$$

The differential equations for the expected trajectories of α and β are given by [21]

$$\dot{\alpha} = - r_y(k+1)$$

$$\dot{\beta} = r_{yu}(k+2)$$

where k is the number of time-delays used in the identification.

Which are the stationary points of these equations? Theorem 3.1 shows that the only values of α and β that need to be considered are those making the closed loop system a moving average. The closed loop system is

$$y(t) = \frac{(1+cq^{-1})(1+\beta q^{-1})}{(1+aq^{-1})(1+\beta q^{-1}) - \alpha q^{-1}(1+bq^{-1})} e(t) \qquad (4.13)$$

Straightforward calculations give that there are four regulators making (4.13) a moving average, see Table 4.1.

The first regulator corresponds to the minimum variance regulator. Regulator number two is the dead-beat regulator. The third regulator is the suboptimal minimum variance regulator discussed in [1]. This regulator can be obtained by using the identity

	α	β	Order of the moving average
1	$a - c$	b	0
2	a	b	1
3	$\frac{c - a}{b - a}\, a$	$\frac{c - a}{b - a}\, b$	1
4	$-\frac{a^2}{b - a}$	$-\frac{a}{b - a}\, b$	2

Table 4.1 - The values of the regulator parameters which make (4.13) a moving average of different order.

$$(1+cq^{-1}) = (1+aq^{-1})(1+f_1q^{-1}) + q^{-1}g_0(1+bq^{-1})$$

and the control strategy

$$u(t) = \frac{-g_0}{1 + f_1q^{-1}}\, y(t)$$

This regulator can be used for nonminimum phase systems and is not sensitive to variations in the parameter values.

The fourth regulator is obtained by using $c = 0$ in the identity above. This can be referred to as the suboptimal dead-beat regulator.

Figure 4.6 shows trajectories with different starting values when $a = -0.99$, $b = 0.5$ and $c = -0.7$ and when $k = $ $= 0$, 1 and 2. The trajectories are obtained by using an

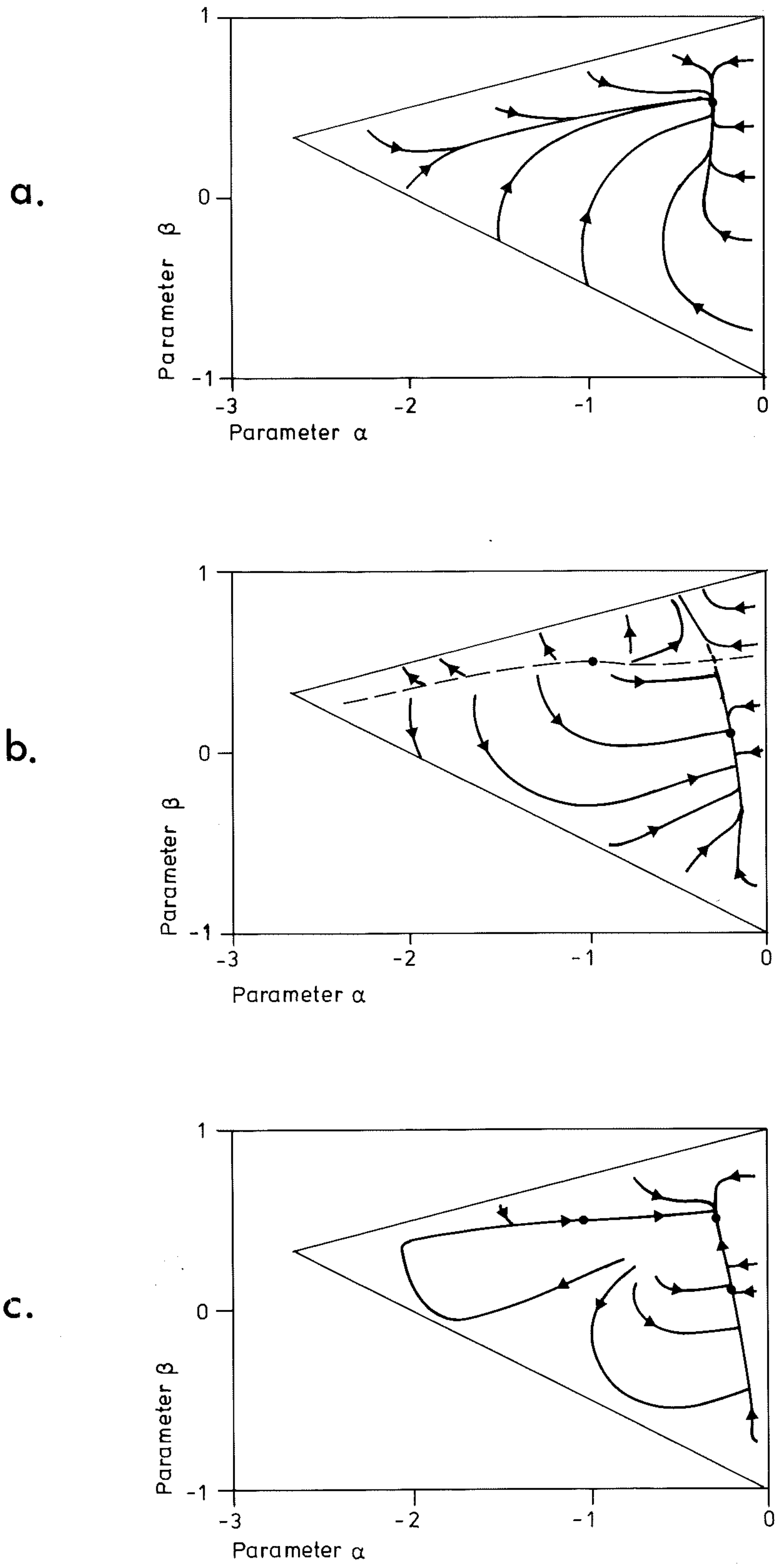


Fig. 4.6 - Trajectories given by

$\dot{\alpha} = - r_y(k+1)$

$\dot{\beta} = r_{yu}(k+2)$

when a = -0.99, b = 0.5, c = -0.7 and for
a) k = 0   b) k = 1   c) k = 2.
The triangles show the stability region of the closed loop systems.

interactive program for simulation of nonlinear differential equations, SIMNON [14], available at the Division of Automatic Control, Lund Institute of Technology. For k = = 0 the stable point corresponds to the minimum variance regulator. The stability region of the differential equations is the same as the stability region for the closed loop system which is indicated by the triangles in Fig. 4.6. For k = 1 the stable point corresponds to the suboptimal regulator. The stability area for the differential equations is no longer the same as the stability region of the closed loop system. When k is further increased to 2 then again the stable point corresponds to the minimum variance regulator. In this case the points corresponding to the dead-beat and the suboptimal regulator are stable points for trajectories coming from some directions. But small disturbances can make the system converge to the minimum variance regulator (compare Example 4.3). ■

Thus by using different k in the identification the algorithm will converge to regulators making the closed loop system a moving average of different orders, compare Example 3.5. The results are verified by simulations with both the simple as well as the basic algorithms. Also for the basic algorithm it is possible to derive differential equations from which the stability can be investigated. The number of equations will be larger than the number of regulator parameters for the basic algorithm [20]. This implies that the expected parameter trajectories are not uniquely determined by the actual values of the parameters but also depends on the past values, i.e. the expected parameter values are in this case not a state vector for the system.

5. MODIFIED MODEL STRUCTURES.

The model structure (2.6) used for the basic algorithm was chosen in order to make it easy to compute the control signal. It is also possible to apply self-tuning to other structures. This is discussed in this section.

5.1. Time-Delay in the Regulator.

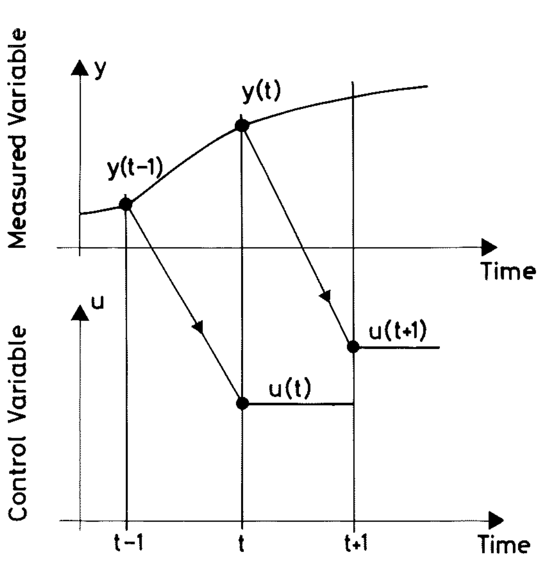
There are essentially two ways to implement control algorithms using a computer [17].

Case A: The measured variables, y(t),are read at time t and the control variables to be set at time t+1, u(t+1), are computed from y(t) during the time interval (t,t+1), see Figure 5.1.

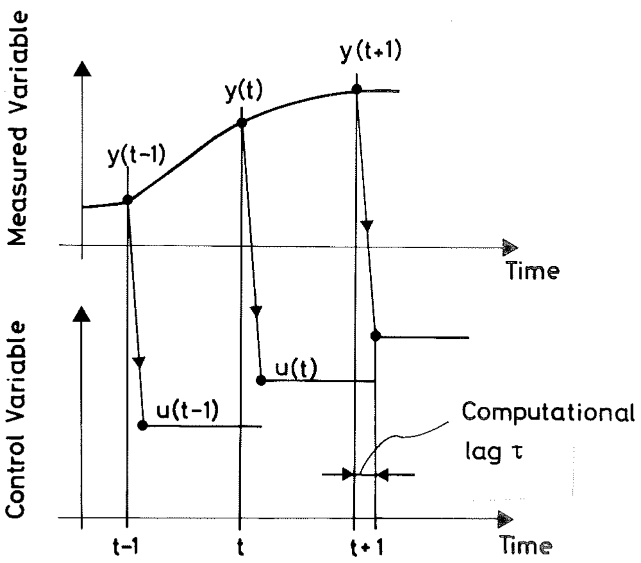
Case B: The measured variables, y(t), are read at time t and the control variables are evaluated as quickly as possible and set at time $t+\tau$, where τ is the smallest time required to do the computation, see Figure 5.2.

In Case A the time-delay introduced in the controller is distinct and known, but may be undesirably long. In Case B the introduced time-delay is as small as possible, but it may vary in an unknown way depending on the work load on the computer.

Case B is assumed when using the basic self-tuning algorithm and it is further assumed that the computation time is short compared with the sampling time. In many cases the computational time cannot be neglected and Case A must thus be considered.



<u>Fig. 5.1</u> - Illustration of inputs and outputs in Case A. (From [17].)



<u>Fig. 5.2</u> - Illustration of synchronization of inputs and outputs in Case B. (From [17].)

For Case A it is straightforward to derive the minimum variance regulator for known systems. The effect of having one time-delay in the regulator is the same as having one extra time-delay in the system. The minimum variance regulator when having one extra time-delay in the regulator has the structure

$$u(t) = \frac{1}{\beta_0} \sum_{i=1}^{m} \alpha_i y(t-i) - \sum_{i=1}^{\ell} \beta_i u(t-i) \qquad (5.1)$$

To obtain a self-tuning regulator it is thus possible to estimate the parameters in a model with the structure

$$y(t+k+1) + \alpha_1 y(t-1) + \dots + \alpha_m y(t-m) =$$

$$= \beta_0 [u(t) + \beta_1 u(t-1) + \dots + \beta_\ell u(t-\ell)] +$$

$$+ \varepsilon(t+k+1) \qquad (5.2)$$

As before the parameter β_0 is assumed known and $\alpha_1, \dots, \alpha_m, \beta_1, \dots, \beta_\ell$ are estimated using the least squares method. Equation (5.1) is then used to compute the control signal. The control law has the desired property that u(t+1) can be computed in the time interval (t,t+1) using data obtained up to and including time t.

It is straightforward to extend Theorems 2.1 and 2.2 to the case when there is a time-delay of one unit in the regulator. In Theorem 2.1 equation (2.14) is changed to

$$r_y(\tau) = Ey(t+\tau)y(t) = 0 \qquad \tau = k+2, \dots, k+m+1$$

In Theorem 2.2 the only change is that the number of parameters in the denominator of the regulator must be increased by one, i.e. $\ell = n+k$. Thus if $m = n$ and $\ell = n+k$ and if the algorithm converges then the output will be a moving average of order k+1. The time-delay in the regulator has the same effect as an additional time-delay in the system.

5.2. Identification Structures.

In the basic algorithm the identification is done using a very special structure of the model selected in such a way that the parameters of the regulator are equal to the estimated parameters. Is it possible to use other structures of the model and still get the same results as before?

Consider the system

$$A(q)y(t) = B(q)u(t-k) + e(t+n) \tag{5.3}$$

Identify the parameters in (5.3) using the least squares method and compute the control law from

$$u(t) = - \frac{q^k G(q)}{\hat{B}(q)F(q)} y(t) \tag{5.4}$$

where F(q) and G(q) are polynomials obtained using the identity

$$q^{k+n} = \hat{A}F + G \tag{5.5}$$

If the closed loop system is identifiable then

$$\hat{A} \rightarrow A$$
$$\hat{B} \rightarrow B$$

The problem of identifiability when the control law is fixed is for instance discussed in [10]. In Example 3.3 it was shown that the conditions for identifiability might be relaxed if the control law is time-varying.

Thus if $C(q) = q^n$ and if the algorithm converges then the controller will converge to the minimum variance regulator. From the arguments given above it could be tempting to conclude that the control law (5.4) will converge to the optimal controller also when $C(q) \neq q^n$. A counter example will be given which shows that this is not always true if the identified model is of the same order as the system.

Example 5.1. Consider the system

$$y(t) + ay(t-1) = bu(t-3) + e(t) + ce(t-1)$$

The minimum variance regulator is given by

$$u(t) = \frac{a^2(a - c)/b}{1 + (c-a)q^{-1} + a(a-c)q^{-2}} y(t)$$

Identify the system using the model

$$y(t) + \hat{a}y(t-1) = \hat{b}u(t-3) + \varepsilon(t) \qquad (5.6)$$

From equations (5.4) and (5.5) the control law is given by

$$u(t) = \frac{\hat{a}^3/\hat{b}}{1 - \hat{a}q^{-1} + \hat{a}^2q^{-2}} y(t) \qquad (5.7)$$

The control law (5.7) is optimal only if

$$\hat{a}^3/\hat{b} = a^2(a-c)/b$$

$$\hat{a} = a - c$$

$$\hat{a}^2 = a(a-c)$$

These relations can be satisfied only if $c = 0$ or $c = a$. Thus it is in general not possible to get the optimal controller in the assumed way.

The analysis will be illustrated by simulations when $a = -1.5$, $b = 1$ and $c = 0$ or -0.9. When $c = 0$ the parameters $\hat{a}$ and $\hat{b}$ will rapidly converge to -1.5 and 1 respectively and the loss is near optimal as can be seen in Figure 5.3. Figure 5.4 shows the parameters and the accumulated loss when the basic self-tuning algorithm is used.

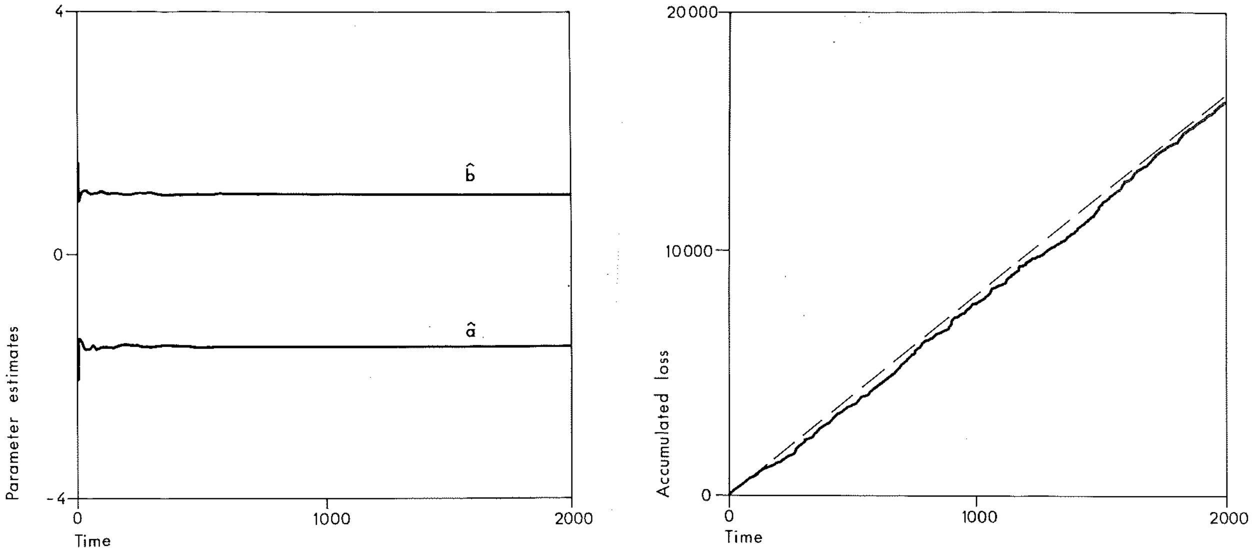


Fig. 5.3 - Parameter estimates and accumulated loss for the system in Example 5.1 when c = 0 and when the parameters are identified using the model (5.6) and the control law is according to (5.7). The dashed line indicates the expected minimal loss.

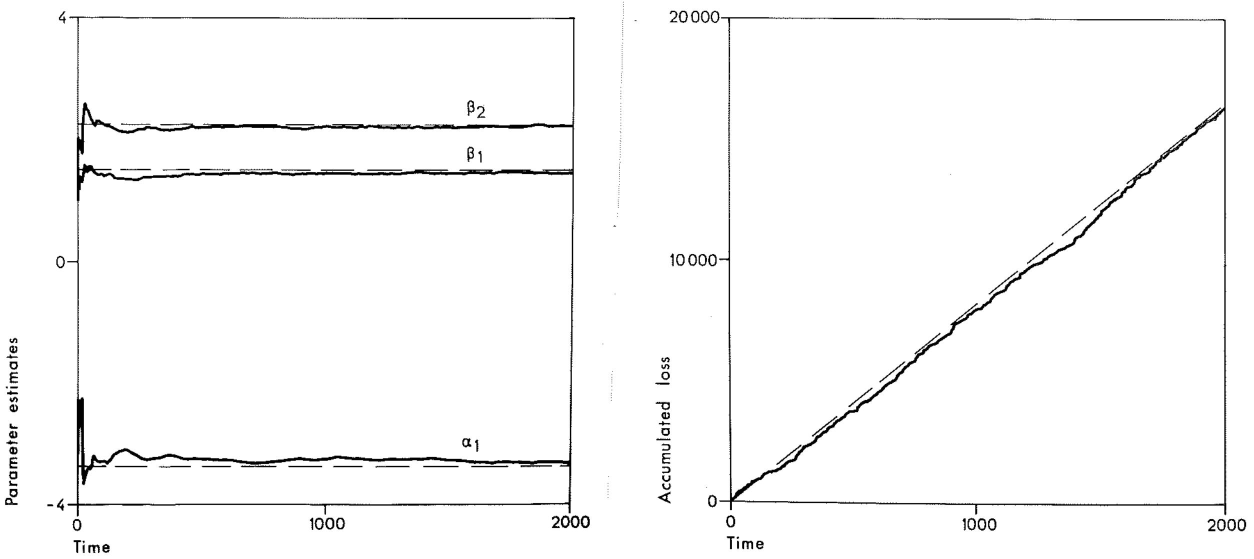


Fig. 5.4 - The parameter estimates and accumulated loss when c = 0 and when the basic self-tuning algorithm is used. The dashed lines indicate the optimal parameter values and the expected minimal loss respectively.

If $c = -0.9$ and if $\hat{a}$ and $\hat{b}$ are estimated using the model (5.6) the accumulated loss will be approximately 50% greater than the optimal loss, see Figure 5.5. The loss is in average 3.33 per step computed over 5000 steps while the minimal expected loss is 2.17 per step. The basic self-tuning algorithm will behave similar to the expected behaviour of the optimal controller, Figure 5.6.

When identifying the parameters according to the model (5.6) using the correct value on the order of the system and when $C(q) \neq q^n$ then the class of controllers possible to generate does not contain the optimal controller. But if the order of the model is increased then the controller gets more degrees of freedom and it can be possible to get a better controller.

Let the model be changed from (5.6) to

$$y(t) + \hat{a}_1 y(t-1) + \hat{a}_2 y(t-2) = \hat{b}_1 u(t-3) + \hat{b}_2 u(t-4) + \varepsilon(t) \qquad (5.8)$$

The algorithm now converges to a controller having a common factor

$$u(t) = \frac{-1.36(1+0.56q^{-1})}{(1+0.61q^{-1}+0.91q^{-2})(1+0.56q^{-1})}\, y(t)$$

The control law is based on parameter values obtained after 10000 steps. When the common factor is removed the rest is very close to the minimum variance controller, which is

$$u(t) = \frac{-1.35}{1 + 0.6q^{-1} + 0.9q^{-2}}\, y(t)$$

The behaviour in this case is shown in Figure 5.7.

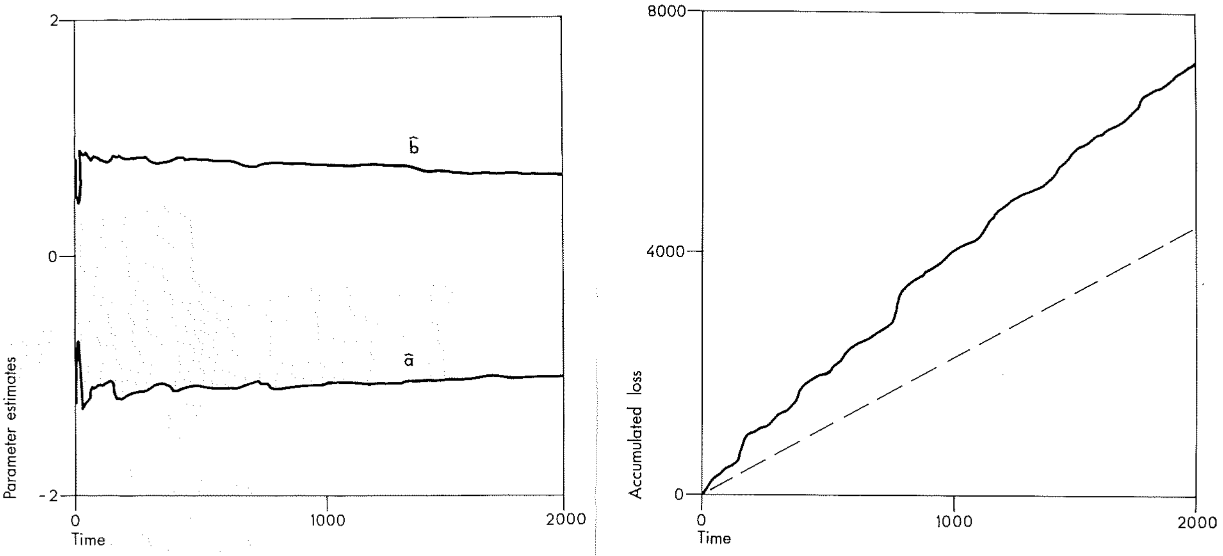


Fig. 5.5 - Estimates and loss when c = -0.9 and the parameters are estimated according to the model (5.6) and when the control law (5.7) is used. The dashed line indicates the expected minimal loss.

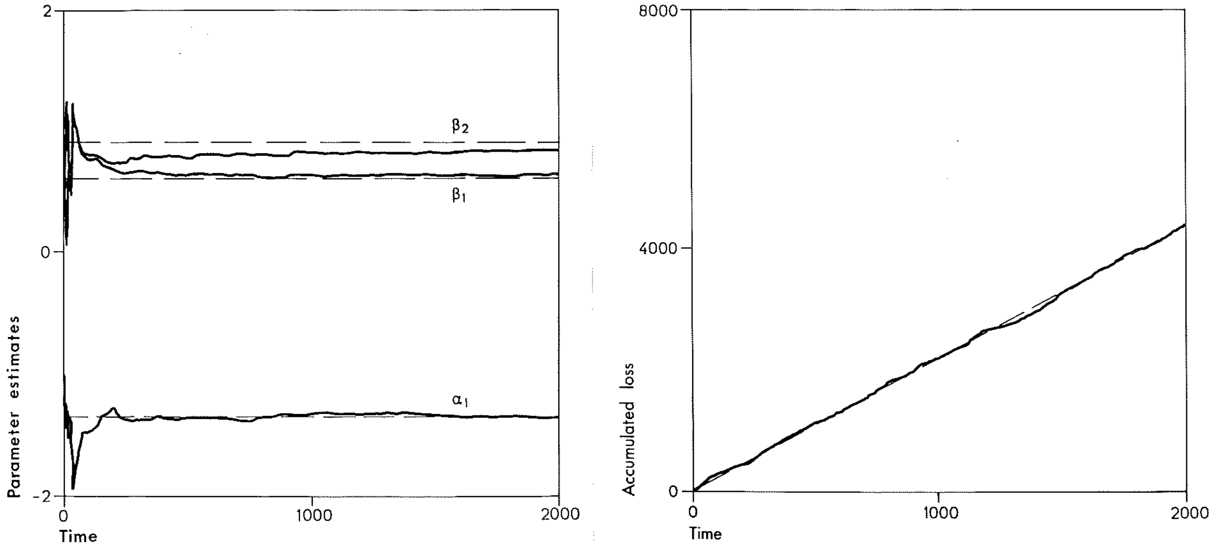


Fig. 5.6 - Estimates and loss when c = -0.9 and when the basic self-tuning algorithm is used. The dashed lines indicate the optimal parameter values and the expected minimal loss respectively.

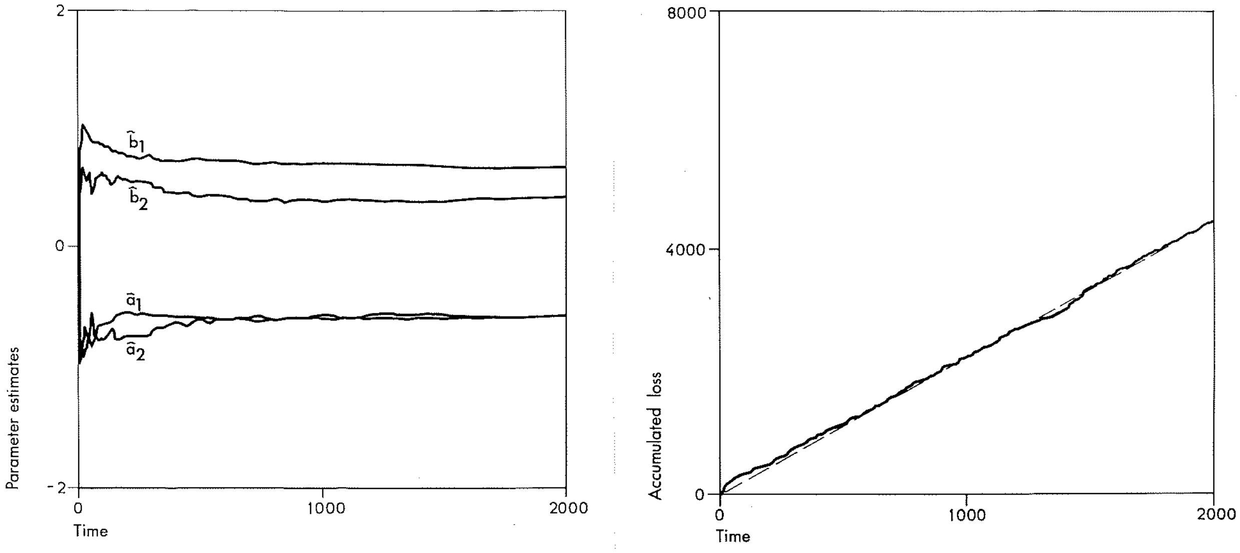


Fig. 5.7 - Estimates and loss when c = -0.9 and when the parameters are estimated according to the model (5.8) and the control law is computed from (5.4) and (5.5). The control law will contain a common factor. ■

Simulations indicate that it is possible to identify the parameters using the model (5.3) and compute the control signal from (5.4) using the identity (5.5). One drawback compared with the basic self-tuning algorithm is that the computation of the control signal is more complicated since the identity (5.5) has to be solved in each step of time. Further it is sometimes necessary to use a controller of higher order than if the basic algorithm is used.

5.3. Feedforward Control.

In many applications it is possible to measure some of the disturbances acting on the process. It is then possible to use these measurements to reduce the influence of the known disturbances using feedforward compensation.

The structure of a regulator with feedforward compensation is first given in the case of a system with known parameters. Let the system be

$$A(q)y(t) = B(q)u(t-k) + C(q)e(t) + D(q)v(t-k) \tag{5.9}$$

where A, B, C and D are known polynomials of degree n, n-1, n and n-1 respectively. Further v(t) is known at time t and v(t) is independent of e(t). Using the identity

$$q^kC(q) = A(q)F(q) + G(q)$$

the equation (5.9) can be written as

$$y(t+k+1) = \frac{B}{A}\, u(t+1) + \frac{C}{A}\, e(t+k+1) + \frac{D}{A}\, v(t+1) =$$

$$= Fe(t+1) + \frac{1}{C}[Gy(t+1) + BFu(t-k+1) + DFv(t-k+1)]$$

The variance of y(t+k+1) is minimized if the control signal is chosen as

$$u(t) = - \frac{q^kG}{BF}\, y(t) - \frac{D}{B}\, v(t) \tag{5.10}$$

The control law (5.10) eliminates the disturbance v(t) completely. But if v enters in equation (5.9) with the time argument t-k' where k' < k it is not possible to get a total elimination of v(t) with a causal regulator. Then in equation (5.10) v(t) has to be replaced by $\hat{v}(t+k-k'|t)$ which is the prediction of v(t+k-k') based on data available up to time t.

From (5.10) it is seen that the feedforward terms appear in the controller in the same way as the measurements. To make a self-tuning regulator which includes feedforward compensation the following algorithm can be attempted:

Step 1: Estimation. Determine the parameters in the model

$$y(t+k+1) + \alpha_1 y(t) + \ldots + \alpha_m y(t-m+1) =$$

$$= \beta_0 [u(t) + \beta_1 u(t-1) + \ldots + \beta_\ell u(t-\ell)] +$$

$$+ \gamma_1 v(t) + \ldots + \gamma_p v(t-p+1) + \varepsilon(t+k+1) \qquad (5.11)$$

using least squares estimation. The parameter β_0 is assumed known.

Step 2: Control. At each sampling interval determine the control variable from

$$u(t) = \frac{1}{\beta_0} \sum_{i=1}^{m} \alpha_i y(t-i+1) - \sum_{i=1}^{\ell} \beta_i u(t-i) -$$

$$- \frac{1}{\beta_0} \sum_{i=1}^{p} \gamma_i v(t-i+1) \qquad (5.12)$$

Using the forward shift operator, q, (5.12) can be written as

$$u(t) = \frac{q^{\ell-m+1}\mathcal{A}(q)}{\mathcal{B}(q)}\, y(t) + \frac{q^{\ell-p+1}\mathcal{C}(q)}{\mathcal{B}(q)}\, v(t)$$

where $\mathcal{A}$ and $\mathcal{B}$ are given by (2.9) and (2.10) respectively and

$$\mathcal{C}(q) = \gamma_1 q^{p-1} + \gamma_2 q^{p-2} + \ldots + \gamma_p$$

If the estimation in Step 1 converges it is possible to characterize the closed loop system in the same way as when the basic self-tuning algorithm is used.

Theorem 5.1. Assume that the system is controlled by the control law (5.12). Further assume that the signal v(t) is persistently exciting [6] and that the parameter estimates $\alpha_i(t)$, $i = 1, \ldots, m$, $\beta_i(t)$, $i = 1,\ldots, \ell$, and $\gamma_i(t)$, $i = 1,\ldots, p$, of the model (5.11) converge and that the closed loop system is such that the output is ergodic (in the second moments). Then the closed loop system has the properties

$$r_y(\tau) = 0 \qquad \tau = k+1, \ldots, k+m \tag{5.13}$$

$$r_{yu}(\tau) = 0 \qquad \tau = k+1, \ldots, k+\ell+1 \tag{5.14}$$

$$r_{yv}(\tau) = 0 \qquad \tau = k+1, \ldots, k+p \tag{5.15}$$

Proof. The least squares estimates are given by the system of equations (5.16).

$$\left[\begin{array}{ccc|ccc|ccc}
\Sigma y(t)^2 & \cdots & \Sigma y(t)y(t-m+1) & -\beta_0\Sigma y(t)u(t-1) & \cdots & -\beta_0\Sigma y(t)u(t-\ell) & -\Sigma y(t)v(t) & \cdots & -\Sigma y(t)v(t-p+1) \\
 & & \Sigma y(t-m+1)^2 & & & & & & \\
\hline
 & & & \beta_0^2\Sigma u(t-1)^2 & & & & & \\
 & & & & & \beta_0^2\Sigma u(t-\ell)^2 & & & \\
\hline
-\Sigma y(t)v(t) & & & & & & \Sigma v(t)^2 & & \\
-\Sigma y(t)v(t-p+1) & & & & & & & & \Sigma v(t-p+1)^2
\end{array}\right]
\left[\begin{array}{c}
\alpha_1 \\ \vdots \\ \alpha_m \\ \beta_1 \\ \vdots \\ \beta_\ell \\ \gamma_1 \\ \vdots \\ \gamma_p
\end{array}\right] =$$

$$= \left[\begin{array}{c}
-\Sigma y(t+k+1)y(t) + \beta_0\Sigma u(t)y(t) \\
\vdots \\
-\Sigma y(t+k+1)y(t-m+1) + \beta_0\Sigma u(t)y(t-m+1) \\
\hline
\beta_0\Sigma y(t+k+1)u(t-1) - \beta_0^2\Sigma u(t)u(t-1) \\
\vdots \\
\beta_0\Sigma y(t+k+1)u(t-\ell) - \beta_0^2\Sigma u(t)u(t-\ell) \\
\hline
\Sigma y(t+k+1)v(t) - \beta_0\Sigma u(t)v(t) \\
\vdots \\
\Sigma y(t+k+1)v(t-p+1) - \beta_0\Sigma u(t)v(t-p+1)
\end{array}\right] \qquad (5.16)$$

where the sums are taken over N_0 values.

Since the parameters converge the coefficients of the control law (5.12) will converge to constant values for sufficiently large N_0. Introduce (5.12) into (5.16). Hence

$$\lim_{N\to\infty} \frac{1}{N} \Sigma y(t+k+1)y(t) = 0$$

$$\vdots$$

$$\lim_{N\to\infty} \frac{1}{N} \Sigma y(t+k+1)y(t-m+1) = 0$$

$$\lim_{N\to\infty} \frac{1}{N} \Sigma y(t+k+1)u(t-1) = 0$$

$$\vdots$$

$$\lim_{N\to\infty} \frac{1}{N} \Sigma y(t+k+1)u(t-\ell) = 0$$

$$\lim_{N\to\infty} \frac{1}{N} \Sigma y(t+k+1)v(t) = 0$$

$$\vdots$$

$$\lim_{N\to\infty} \frac{1}{N} \Sigma y(t+k+1)v(t-p+1) = 0$$

It also follows from the control law (5.12) that

$$\lim_{N\to\infty} \frac{1}{N} \Sigma y(t+k+1)u(t) = 0$$

Under the ergodic assumption the sums can be replaced by mathematical expectations and the theorem is proven. ■

Remark. If v(t) is a deterministic signal the condition (5.15) shall be interpreted as

$$\lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} y(t+\tau)v(t) = 0 \qquad \tau = k+1, \ldots, k+p$$

Consequently it is possible to state a theorem that is analogous to Theorem 2.2.

Theorem 5.2. Let the system to be controlled be governed by the n:th order system

$$A(q)y(t) = B(q)u(t-k) + C(q)e(t) + D(q)v(t-k)$$

where A, B, C and D are polynomials of degree n, n-1, n and n-1 respectively. e(t) is a sequence of independent stochastic variables. v(t) is a known disturbance which is persistently exciting when the parameters are tuned. Further it is assumed that v(t) is independent of e(t).

Assume that the self-tuning regulator is used with $m = n$, $\ell = n+k-1$ and $p = n+k$. If the parameter estimates converge, independently of the realization of v(t), to values such that the controller polynomials $\mathcal{A}$ and $\mathcal{B}$ do not contain any common factors, then the corresponding regulator (5.12) will converge to the minimum variance regulator (5.10).

Proof. Assume that the least squares estimates have converged. The control law is then given by

$$u(t) = \frac{q^{\ell-m+1}\mathcal{A}(q)}{\mathcal{B}(q)}\, y(t) + \frac{q^{\ell-p+1}\mathcal{C}(q)}{\mathcal{B}(q)}\, v(t) =$$

$$= \frac{q^{k}\mathcal{A}(q)}{\mathcal{B}(q)}\, y(t) + \frac{\mathcal{C}(q)}{\mathcal{B}(q)}\, v(t)$$

The closed loop system becomes

$$(A\mathcal{B}-B\mathcal{A})y(t) = C\mathcal{B}e(t) + (B\mathcal{C}+D\mathcal{B})v(t-k)$$

or

$$y(t) = \frac{C\mathcal{B}}{A\mathcal{B} - B\mathcal{A}}\, e(t) + \frac{B\mathcal{C} + D\mathcal{B}}{A\mathcal{B} - B\mathcal{A}}\, v(t-k) \qquad (5.17)$$

$\mathcal{A}$, $\mathcal{B}$ and $\mathcal{C}$ are constant and assume that v(t) is equal to zero over a long period of time then the closed loop system is given by

$$y(t) = \frac{C\mathcal{B}}{A\mathcal{B} - B\mathcal{A}} e(t)$$

The conditions (5.13) and (5.14) are still valid. Use Theorem 2.2 which now implies that

$$u(t) = \frac{q^k\mathcal{A}}{\mathcal{B}} y(t) = \frac{q^k G}{BF} y(t)$$

where

$$q^k C = AF + G$$

and that

$$A\mathcal{B} - \mathcal{A}B = q^k BC$$

$$\mathcal{B} = BF$$

Use these expressions in (5.17) then for $v(t) \not\equiv 0$

$$y(t) = q^{-k}F(q)e(t) + \frac{\mathcal{C} + DF}{q^k F} v(t-k)$$

The transfer function

$$H = \frac{\mathcal{C} + DF}{q^k F}$$

is of order n+k. The crosscovariance function between y(t) and v(t) will now be

$$r_{yv}(\tau) = H(q)r_v(\tau-k)$$

Condition (5.15) now implies that $H(q) \equiv 0$ implying that

$$\mathcal{G} = -DF$$

Thus

$$u(t) = \frac{q^k\mathcal{A}}{\mathcal{B}}\,y(t) + \frac{\mathcal{G}}{\mathcal{B}}\,v(t) = \frac{q^kG}{BF}\,y(t) - \frac{D}{B}\,v(t)$$

The controller has thus converged to the minimum variance regulator which gives the closed loop system

$$y(t) = q^{-k}F(q)e(t)$$

The disturbance v(t) is totally eliminated. ■

6. LIMITATIONS OF THE BASIC SELF-TUNING ALGORITHM.

The self-tuning algorithm described in [8] and in this report has good transient and asymptotic properties although the algorithm does not converge for all systems. The algorithm can handle systems having different orders, many time-delays, coloured noise, feedforward terms etc. Another advantage is that only few parameters are needed to specify the algorithm. There are, however, some limitations of the algorithm. Analysing the assumptions done when deriving the algorithm it can be found that some types of systems are excluded.

Firstly it is assumed that the exciting noise is stationary. An example will show what might happen if the variance of the noise is time-varying.

Secondly when deriving the minimum variance regulator for known systems it is assumed that the system is minimum phase. If the system is nonminimum phase special care must be taken since the ordinary minimum variance regulator is extremely sensitive for variations in the values of the parameters in the regulator. It should, however, be desirable if nonminimum phase systems could also be handled by the basic self-tuning algorithm.

Thirdly it is assumed that the parameters in the system are constant. It will be shown by examples what may happen if the parameters are time-varying.

It turns out that the performance of the self-tuning algorithm will deteriorate in the cases discussed above, but the performance will in many cases still be acceptable. In essence the basic self-tuning algorithm seems to be able to handle most types of systems. It is, however, necessary to use the algorithm with care when attempting to solve more difficult problems.

6.1. Nonstationary Noise.

When the minimum variance controller is derived for systems with known parameters it is assumed that the noise is stationary. In many practical situations this is not the case. It is thus desirable that the self-tuning regulators can work satisfactorily even if the noise is nonstationary.

The self-tuning regulators are mainly characterized by Theorems 2.1 and 2.2. In Theorem 2.1 nothing is assumed concerning the noise, but if the estimation converges then certain covariances are equal to zero. Since no general convergence proofs are available it is only possible to investigate the behaviour of the algorithm using simulations. In Section 8 the case when the noise is a deterministic signal will be discussed. In this section will be demonstrated what may happen if the stochastic variables e(t) are not equally distributed.

In many practical cases the disturbances can be described as stationary processes, but suddenly larger disturbances may occur in the process. From simulations it can be concluded that a time-varying standard deviation of the noise is not especially serious for the self-tuning regulators.

Example 6.1. Consider the system

$$y(t) - 1.9y(t-1) + 0.9y(t-2) =$$

$$= u(t-2) - u(t-3) + e(t) - 0.5e(t-1)$$

The noise e(t) is gaussian distributed, $N(0,\sigma)$ with a time-varying standard deviation, σ, given by

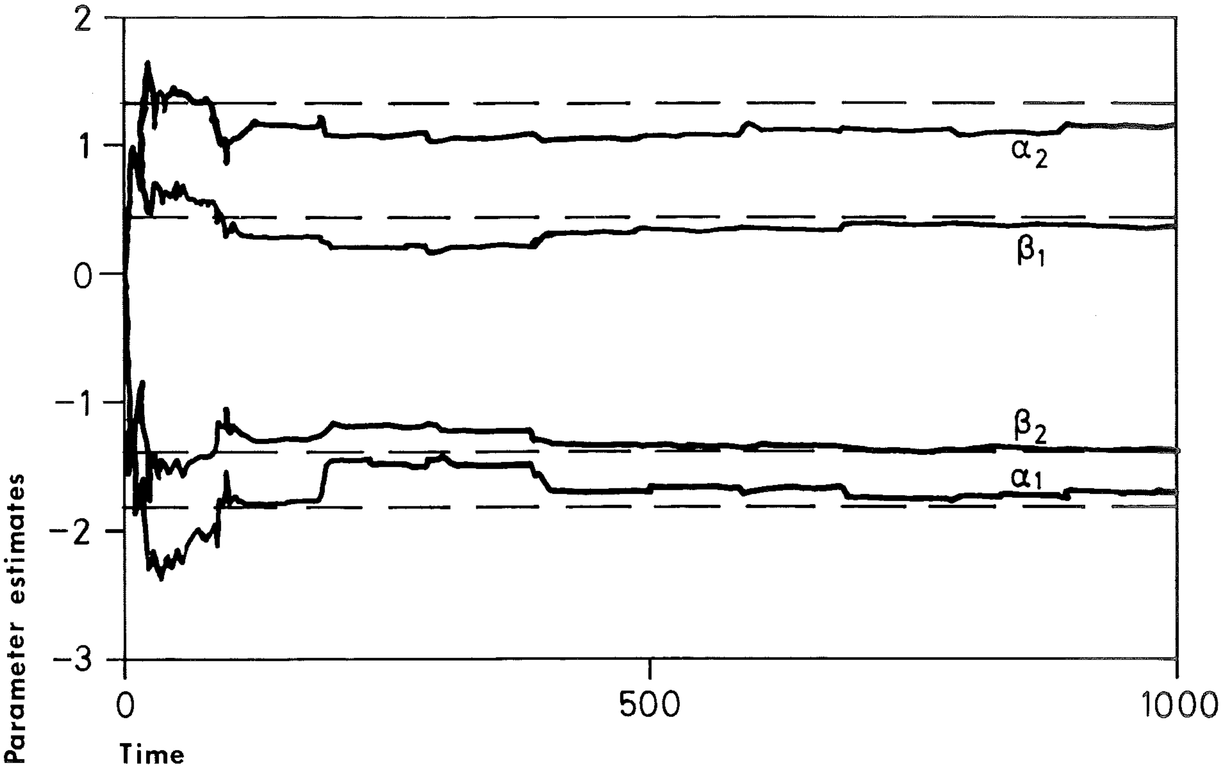


Fig. 6.1 - The parameter estimates for the system in Example 6.1. The jumps in the estimates occur when the noise amplitude is increased.

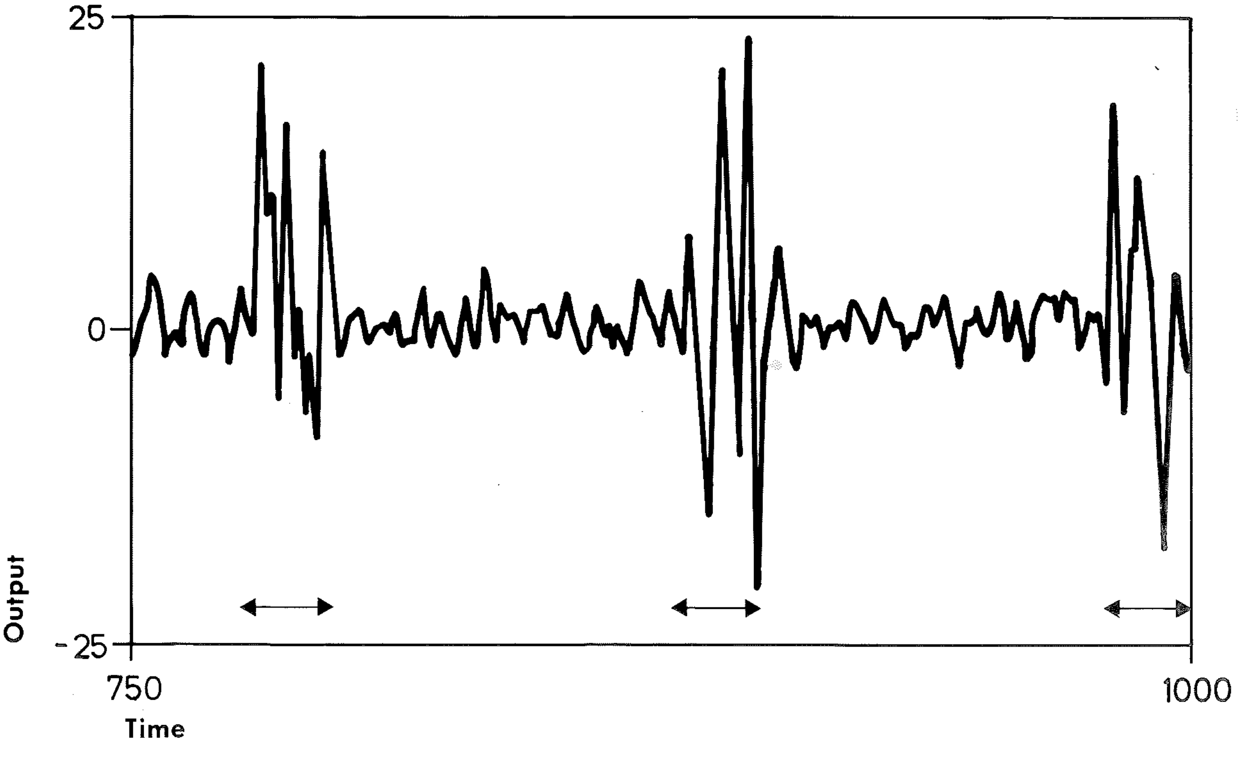


Fig. 6.2 - The output from the system in Example 6.1. The arrows show the intervals when the variance of the noise is increased.

$$\sigma(t) = \begin{cases} 1 & t \pmod{100} < 80 \\ 5 & t \pmod{100} \geq 80 \end{cases}$$

The minimum variance regulator is given by

$$u(t) = \frac{-1.76 + 1.26q^{-1}}{1 + 0.4q^{-1} - 1.4q^{-2}}\, y(t)$$

The system is controlled by a self-tuning regulator with $m = \ell = 2$, $P(0) = 10 \times I$ and $\lambda = 1$. The parameter estimates are shown in Figure 6.1. The convergence is not as fast as when the noise is stationary (compare Example 7.2 in [8]). The regulator will, however, still converge to the minimum variance regulator. If σ is constant equal to 1 the estimates will converge in about 400 steps of time. The average loss for the self-tuning regulator is 17.4 per step calculated over 1000 steps while the optimal regulator would give a loss of 17.2 per step. One part of the output is shown in Figure 6.2. ■

The noise entering the system will also be nonstationary if the parameters in the A or C polynomials in (2.1) are time-varying. In that case the optimal regulator will be time-varying and not fixed as in Example 6.1. This case will be discussed in Section 6.3.

6.2. Nonminimum Phase Systems.

Let the system to be controlled be described by

$$A(q)y(t) = B(q)u(t-k) + C(q)e(t) \tag{6.1}$$

If the parameters in (6.1) are known the minimum variance controller is given by [1]

$$u(t) = - \frac{q^k G(q)}{B(q)F(q)} y(t) \tag{6.2}$$

where G(q) and F(q) are polynomials defined by

$$q^k C(q) = A(q)F(q) + G(q) \tag{6.3}$$

If the system is nonminimum phase, then the closed loop system will be extremely sensitive for variations in the regulator parameters. A slightest error in the parameters will make the system unstable. This since the closed loop system has uncontrollable modes outside the unit circle which are cancelled if the parameters are exact.

One way to get around the problem is to use a suboptimal control law which is discussed in [1]. Introduce $B_1(q)$ and $B_2(q)$ such that

$$B(q) = B_1(q)B_2(q)$$

and where $B_1(q)$ has all zeroes inside the unit circle and $B_2(q)$ has all zeroes outside the unit circle. Determine F and G from the identity

$$q^{k+n_2} C = A(q)F(q) + B_2(q)G(q) \tag{6.4}$$

where n_2 is the order of the B_2 polynomial and use the control law

$$u(t) = - \frac{q^k G}{B_1 F} y(t)$$

The closed loop system will now have the characteristic polynomial

$$q^{k+n_2} B_1 C$$

i.e. the modes corresponding to B_2 are moved to the origin. This suboptimal regulator will make the closed loop system to a moving average of order $k+n_2$.

Another way to get around the problem with nonminimum phase systems is to reformulate the control problem. Assume as before that the output variance shall be minimized, but now under the constraint that the closed loop system has all poles inside the unit circle. This problem can be solved using linear quadratic control theory. If the system is minimum phase the solution will be the same as when the identity (6.3) is used. If the system is nonminimum phase the reformulated control problem will give the closed loop system

$$q^k B_1 \tilde{B}_2 C$$

where the zeroes of $\tilde{B}_2$ have the reciprocal values of the zeroes of B_2 . The closed loop system will no longer be a moving average. The problem can be solved either by solving an identity similar to (6.4) [25] or by iterating a Riccati equation. From a computational point of view it seems to be more favourable to solve the problem by iterating the Riccati equation than by solving a system of equations.

One way to make a self-tuning regulator is now to estimate the parameters in the model

$$A(q)y(t) = B(q)u(t-k) + e(t)$$

using the method of least squares. The control law can then be obtained by iterating a Riccati equation based on the estimated parameters. This approach was proposed in [5]. It is thus possible to make self-tuning regulators that automatically can handle nonminimum phase systems. These routines are, however, more complicated and have a much longer computation time than the basic self-tuning algorithm. It would thus be attractive to use the basic self-tuning algorithm also for nonminimum phase systems. Examples in Sections 3 and 4 showed that the basic self-tuning regulator can converge to the suboptimal regulator obtained by using the identity (6.4). This can happen if the number of time-delays used in the identification, k, is greater than the number of pure time-delays in the system. Guided by this observation the following example is considered:

Example 6.2. Let the system be

$$y(t) - 0.99y(t-1) = - u(t-1) + 2u(t-2) + e(t) - 0.7e(t-1)$$

The suboptimal controller obtained by using the identity (6.4) is

$$u(t) = - \frac{0.28}{1 + 0.57q^{-1}} y(t)$$

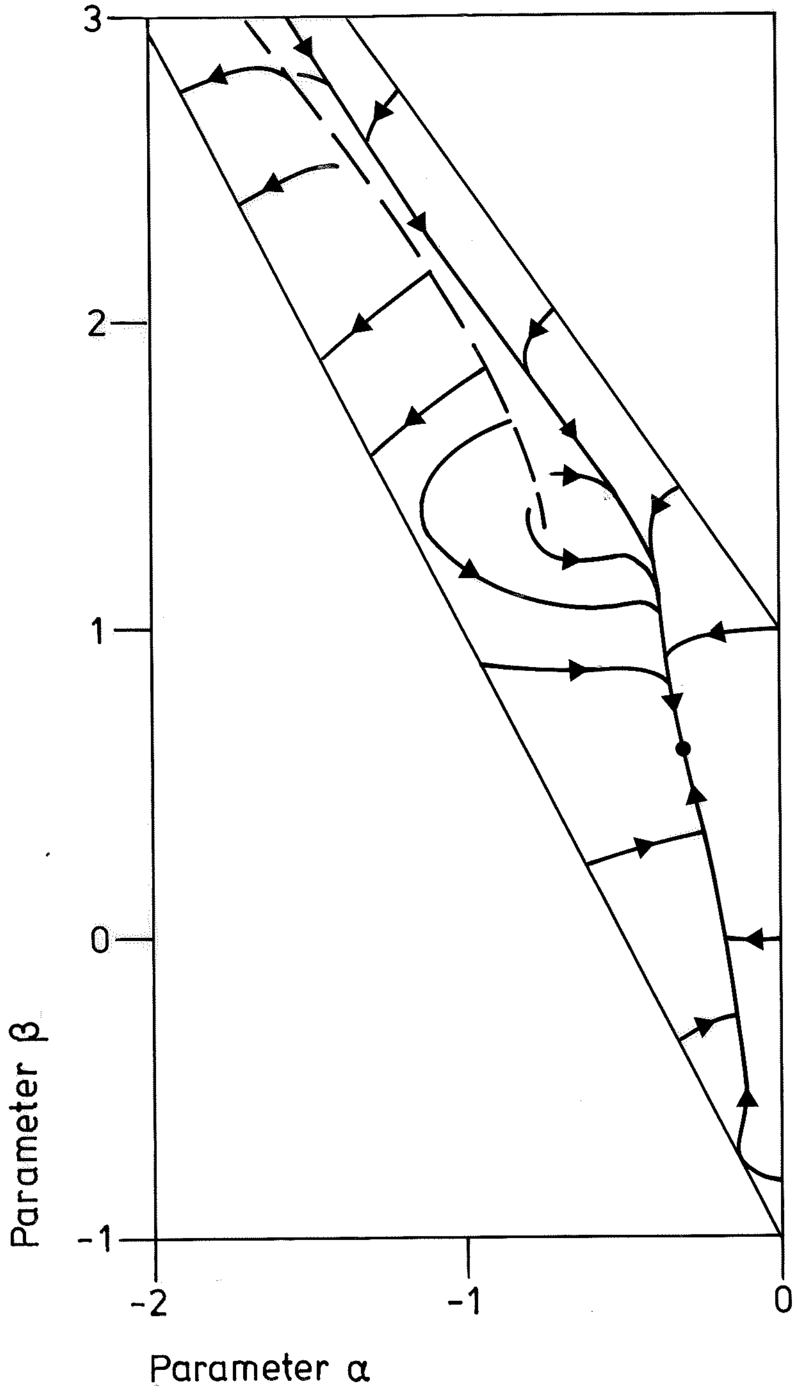


Fig. 6.3 - Expected trajectories of the parameters α and β when the simplified self-tuning algorithm is used to control the nonminimum phase system in Example 6.2. It is assumed that k = 3 in the identification.

which gives a loss of 1.33 per step. The solution to the linear quadratic problem results in a controller giving a loss of 1.25 per step.

To apply the basic self-tuning algorithm the following model is used:

$$y(t+k+1) + \alpha y(t) = \beta_0[u(t) + \beta u(t-1)] + \varepsilon(t+k+1)$$

According to the rule of thumb given in Section 3.4 β_0 should in this case be chosen equal to -1. Since the static gain in the system is negative β_0 must be given a positive sign in order to get a stable system. β_0 is thus given the value 1.

For k equal to 0, 1 and 2 it has not been possible to obtain convergence. But for k = 3 the algorithm will converge to the suboptimal regulator. Using the same arguments as in Example 4.3 it is possible to derive differential equations for the expected trajectories of the parameters when the simplified self-tuning algorithm is used. The trajectories for some starting values are shown in Figure 6.3 when k = 3. In this case simulations show that the simplified and the basic algorithm have the same gross features. Even if the regulator only contains two parameters it has not been possible to tell which values on k that will give convergence to the suboptimal regulator. There seems, however, to be some regularity since it has been possible to get convergence to the suboptimal controller for k = 3, 5, but the system has not been stable for k = 0, 1, 2, 4. ■

In the example above the suboptimal controller gives only a small increase in the loss compared with the loss obtained when solving the linear quadratic control problem. The difference between the two controllers is larger in the next example.

Example 6.3. Consider the continuous time system

$$Y(s) = \frac{1 - s}{1 + s} U(s)$$

The system is sampled with a sampling time $T = -\ln 0.7 \approx \approx 0.35$ seconds and a delay of two sampling intervals is introduced. The sampled system is then described by the difference equation

$$y(t) = \frac{-q^{-2} + 1.3q^{-3}}{1 - 0.7q^{-1}} u(t)$$

Drifting noise characterized by

$$v(t) = \frac{1}{(1-0.7q^{-1})(1-q^{-1})} e(t)$$

is added to y(t) and the following system is finally obtained

$$y(t) - 1.7y(t-1) + 0.7y(t-2) = - \nabla u(t-2) + 1.3\nabla u(t-3) + + e(t) \tag{6.5}$$

where

$$\nabla u(t) = u(t) - u(t-1)$$

The basic self-tuning regulator is used to control the system with $m = \ell = 2$. For $k = 1$ the closed loop system will be unstable. If k is increased to 2 the algorithm will converge to the suboptimal controller

$$\nabla u(t) = \frac{-9.77 + 6.44q^{-1}}{1 + 1.7q^{-1} + 11.96q^{-2}} y(t)$$

This controller gives an expected loss of 147 per step. The controller obtained by solving the linear quadratic control problem gives an expected loss of 62 per step. The two step head prediction error for the system described by (6.5) corresponds to a variance of 3.89. The nonminimum phase property will thus to a large extent deteriorate the behaviour of the system.

The algorithm appears to diverge if using $k = 6$ in the identification. The loss will, however, be smaller than if the suboptimal controller is used. Figure 6.4 shows the accumulated losses when using the complex self-tuning regulator based on linear quadratic control and when using the basic algorithm with $k = 2$ and 6. The difference between the used controllers can also be seen in the covariance functions, Figure 6.5. The output will be a moving average of second order when the basic self-tuning algorithm is used with $k = 2$. The complex self-tuning algorithm gives a covariance function which in this case is exponentially decreasing. It is very difficult to describe what happens when the basic algorithm is used with $k = 6$. The algorithm tries to make $r_y(7)$ and $r_y(8)$ equal to zero. Theorem 3.1 shows that if the estimation converges then the output will be a moving average. Analyzing the closed loop system it can be shown that in this case there are only two possibilities to obtain a moving average. The first case corresponds to the minimum variance regulator, which is very sensitive to variations in the parameters. The second case corresponds to the suboptimal regulator, which gives a moving average of second order. One heuristic explanation of the behaviour when $k = 6$ can be the following: The algorithm tries to converge to the suboptimal regulator by making $r_y(7)$ and $r_y(8)$ equal to zero. Since the algorithm only can influence the covariance function for $\tau = 7$ and 8 the behaviour of the closed loop system can be sensitive for variations in the parameter estimates. This seems to cause a smoother control which results in the lower loss when $k = 6$.

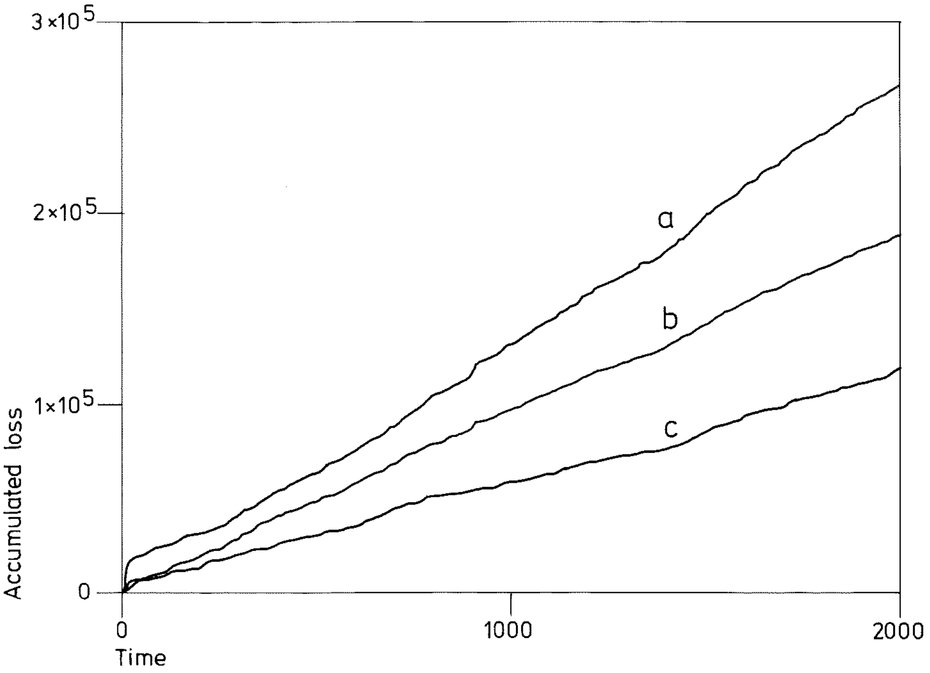


Fig. 6.4 - The accumulated losses when the system (6.5) is controlled by different regulators.
a. Basic self-tuning algorithm k = 2.
b. Basic self-tuning algorithm k = 6.
c. Complex self-tuning algorithm.

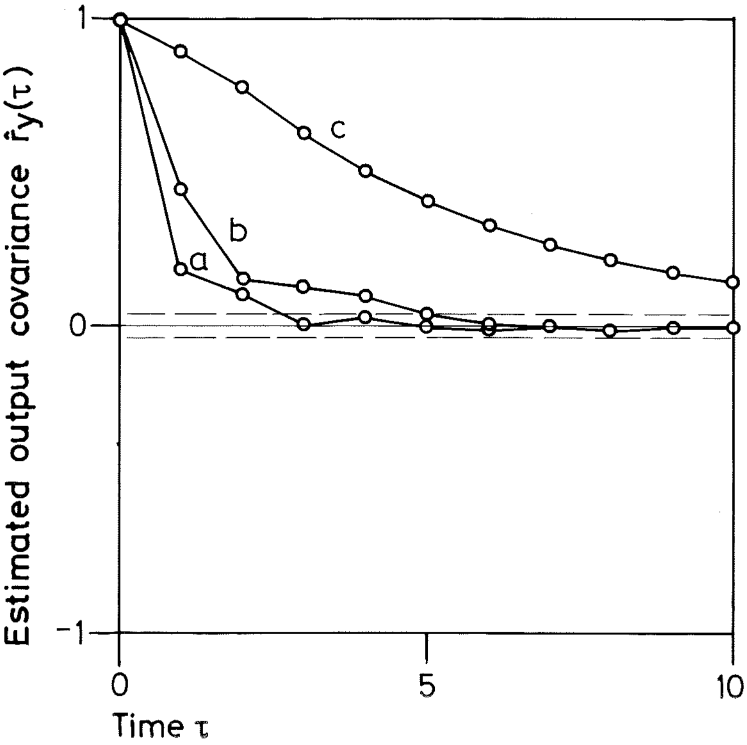


Fig. 6.5 - Estimated covariance function when the system (6.5) is controlled by different regulators.
a. Basic self-tuning algorithm k = 2.
b. Basic self-tuning algorithm k = 6.
c. Complex self-tuning algorithm. ■

The examples above show that the basic self-tuning algorithm can in fact be used to control nonminimum phase systems. The number of time-delays used in the identification must, however, be increased. General rules for how much k ought to be increased have not been found. Further the parameter β_0 must be chosen in such a way that it has the same sign as the steady state gain of the system.

If the performance of the system is not satisfactory when the basic self-tuning algorithm is used it is possible to use the more complex routine based on linear quadratic theory.

6.3. Time-Varying Parameters.

In the analysis of almost all adaptive controllers it is assumed that the parameters of the process are constant. Otherwise it is very difficult to make any statements about stability and convergence. Having obtained results for the case of constant parameters the results may then sometimes be extended to systems with slowly varying parameters. The self-tuning algorithms are derived to control systems with constant but unknown parameters. The self-tuning regulators can also be used for systems with slowly time-varying parameters. In Section 3.1 the exponential forgetting factor, λ, was discussed. By introducing a $\lambda < 1$ it is possible to follow slowly varying parameters. This is done at the price of fluctuations in the estimates in case the parameters are constant. It is difficult to say anything in general about how fast the parameters are allowed to vary. Some examples are given in the following.

Example 6.4. Let the system be

$$y(t) + a_1(t)y(t-1) + a_2(t)y(t-2) = u(t) - 0.5u(t-2) + e(t)$$

where the parameters a_1 and a_2 are time-varying

$$a_1(t) = -1 - 0.001t$$

$$a_2(t) = 0.09 + 0.002t$$

The open loop system will be unstable when t is greater than 455. This system has been controlled by the basic self-tuning regulator using $m = 2$, $\ell = 1$, $\beta_0 = 1$, $P(0) = 10 \times I$ and zeroes as initial values for the parameters in the controller. The average loss over the interval 101 - 2000 is shown in Table 6.2 for different values on λ. If the system is controlled by the minimum variance regulator then the expected loss is 1.00 per step.

λ	$\frac{1}{1900}\sum_{t=101}^{2000} y(t)^2$
1.00	6.26
0.999	2.86
0.995	1.18
0.99	1.07
0.95	1.09
0.90	1.22

Table 6.1 - Average loss for different values on λ when the system in Example 6.4 is controlled by the basic self-tuning regulator.

By using an appropriate value on λ it is in this case possible to get a quite good control of the system in spite

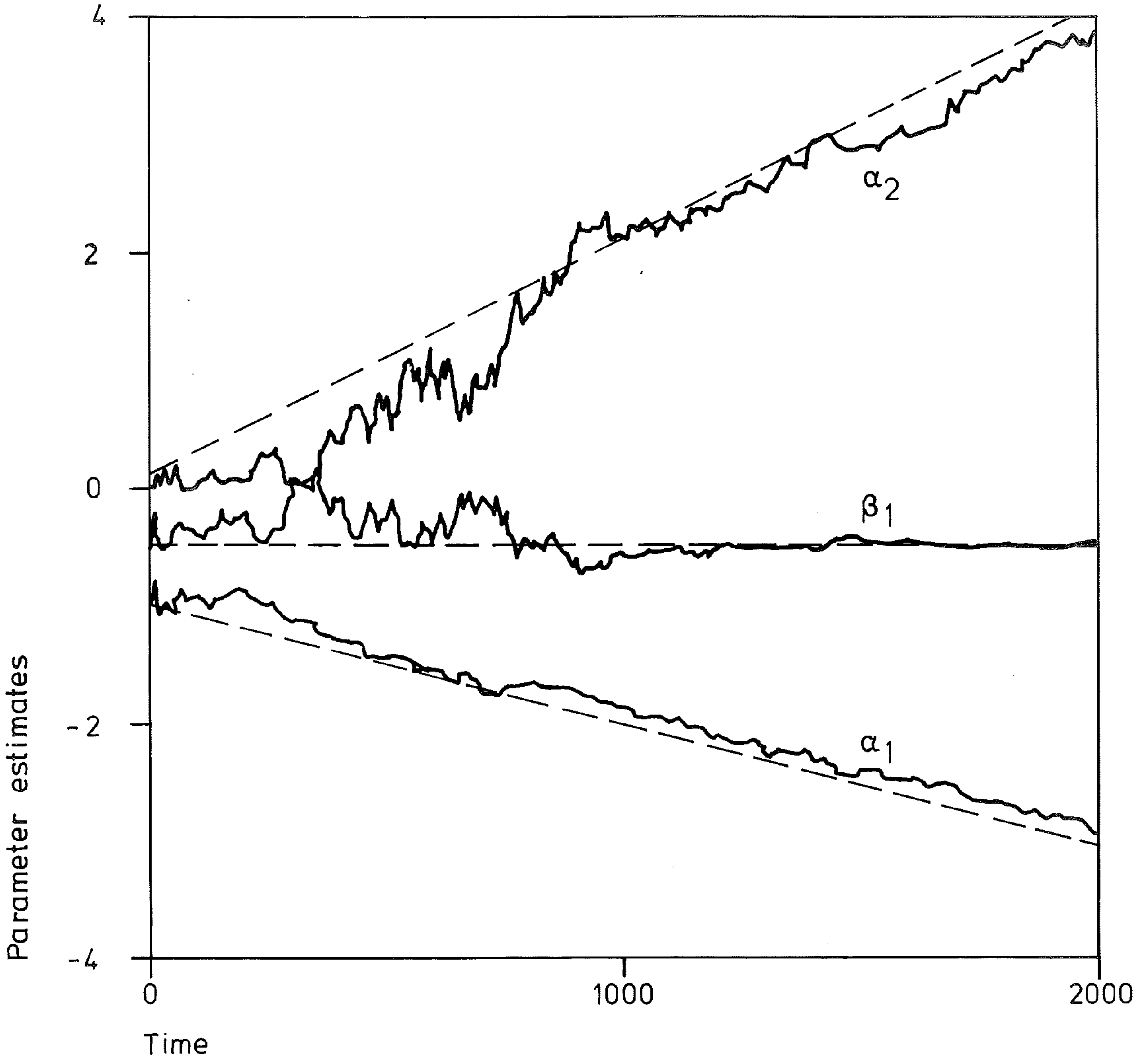


Fig. 6.6 - The parameter estimates for the system in Example 6.4. The parameter values of the optimal controller are indicated by the dashed lines. The value of λ is 0.99.

of the time-varying parameters. For λ less than about 0.97 the estimates of the parameters are very noisy. The estimated parameters when $\lambda = 0.99$ is shown in Figure 6.6.

It is possible to increase the rate of change in the parameters without encountering any difficulties. If the rate of change is increased by a factor 10 then the smallest average loss was 1.25 per step which was obtained for $\lambda = 0.92$. ■

In the following example the zero of the system is time-varying in such a way that the system becomes nonminimum phase.

Example 6.5. Consider the following system

$$y(t) - 0.99y(t-1) = u(t-2) + b(t)u(t-3) + e(t)$$

where

$$b(t) = 0.1 + 0.001t$$

For $t \geq 900$ the system is nonminimum phase. The system has the property that the steady state gain is positive for all t. This means that the same sign of β_0 can be used in the basic self-tuning algorithm.

If the number of time-delays in the identification is $k = 1$ then it is possible to use the basic algorithm up to $t \approx 1000$. After that time the system becomes unstable and the control signal will all the time be limited. If k is increased to 3 then the loss will be about 25% larger over the first 1000 steps. It is, however, possible to control the system also after it has become nonminimum phase. Some difficulties to get a stable system were encountered. It was necessary to use $\beta_0 = 10$ and $\lambda = 0.97$.

It is in this case easier to use the regulator based onlinear quadratic control. Thisalgorithm tries to place the poles of the closed loop system inside a circle with radius r_0. For the system under consideration it is favourable to choose r_0 somewhat less than one. The estimate of b(t) lags the true value somewhat and the system will thus be nonminimum phase some time before this is discovered by the estimator. A radious of 0.9 will give good performance of the system in this particular case. The loss functions when using the basic routine with k = = 1 and 3 and when using the more complex routine is shown in Figure 6.7. The control signal will start to hit the limitation at t ≈ 1000 when the basic algorithm is used with k = 1.

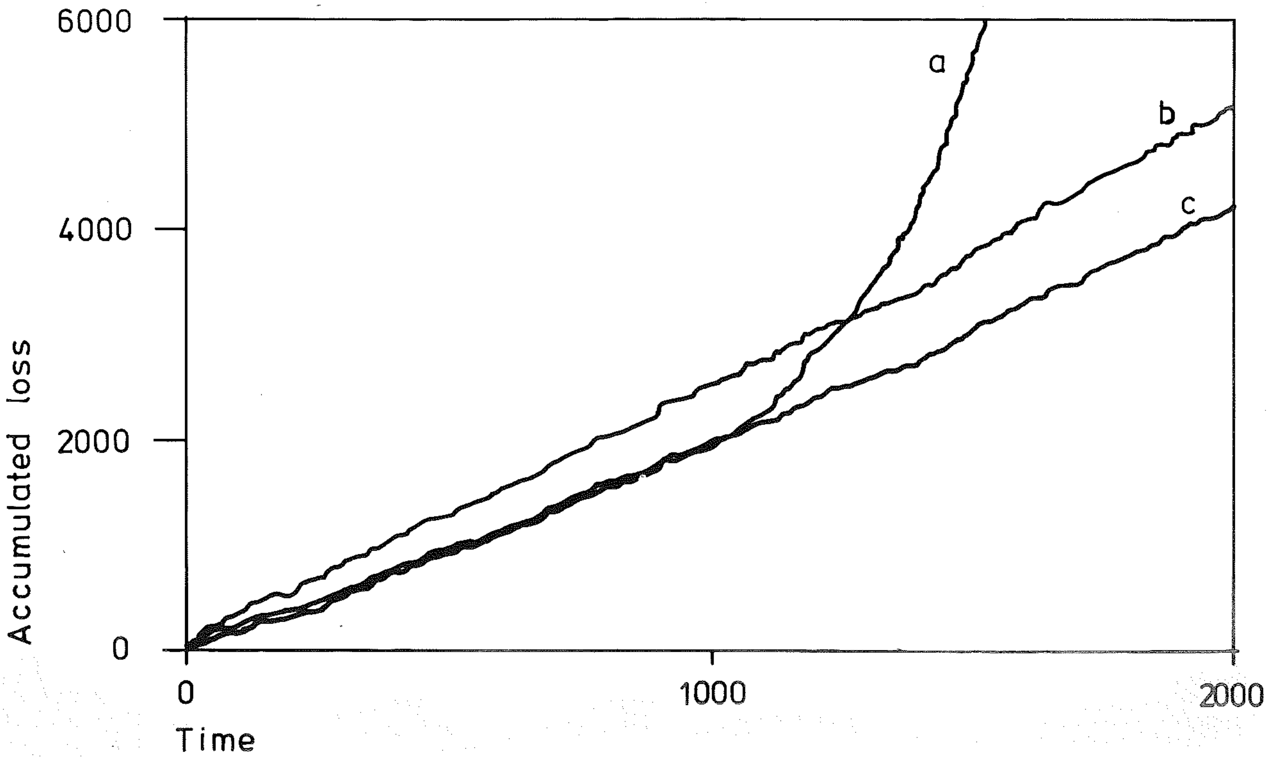


Fig. 6.7 - Accumulated loss when controlling the system in Example 6.5. $|u(t)| \leq 10$.
a. Basic self-tuning algorithm k = 1.
b. Basic self-tuning algorithm k = 3.
c. Complex self-tuning algorithm based onlinear quadratic control theory. ■

The examples discussed show that it is possible to use the basic self-tuning algorithm even if the systems have time-varying parameters. In the first example the open loop system became unstable and then it seems to be quite easy to follow parameters that are varying relatively fast in comparison with the system dynamics. This depends probably on the fact that the estimator is very sensitive to errors in the parameters if the system is unstable. It may be difficult to follow parameter variations if b_1 is increasing. In that case the α_i-parameters can become very small and if a too small λ is used then the α_i-parameters can get wrong sign because of the "noisy" estimates. This can, however, be circumvented by increasing β_0 which will increase the absolute value of the α_i-parameters.

Using an exponential factor there is no possibility to use a priori knowledge that some parameters are constant while others are time-varying. This can, however, be done by making a minor change in the estimator.

Assume that the parameter vector $\theta(t)$ can be described as a stochastic process

$$\theta(t+1) = \theta(t) + v(t)$$

where $Ev(t)v(t)^T = R_1$. The parameter vector can now be estimated using a Kalman filter [30]. The equations (2.11) and (2.12) in the estimator will be the same as before but (2.13) is changed to

$$P(t+1) = P(t) + R_1 - K(t)\left(1 + \varphi(t-k-1)P(t)\varphi(t-k-1)^T\right)K(t)^T$$

It is convenient to choose R_1 as a diagonal matrix. The elements in the diagonal shall reflect the rate of change in the parameters. If a parameter is assumed to be constant then the corresponding diagonal element in R_1 shall be zero or very small.

7. ALGORITHMS USING PARAMETER UNCERTAINTIES.

The algorithm for the self-tuning regulators is derived heuristically based on the assumption that identification and control can be separated. Further the control law is the deterministic control law with the true parameter values substituted by the estimated ones. The fact that the estimates are inaccurate are thus not considered when deriving the control law. In [7] and [31] an alternative was proposed where the uncertainties of the estimates were included. In this section these control strategies are compared with the basic self-tuning algorithm. The analysis follows [7] the notation is, however, slightly different in order to make a direct comparison with the basic algorithm possible.

Consider the minimization of

$$Ey(t+k+1)^2 \tag{7.1}$$

Using a fundamental lemma from stochastic control theory [1, p. 261] it is possible to show that the minimization of (7.1) is equivalent to the minimization of

$$E\left(y(t+k+1)^2 \,|\mathcal{Y}_t\right) \tag{7.2}$$

with respect to u(t) where

$$\mathcal{Y}_t = [y(t),\ y(t-1),\ \ldots,\ u(t-1),\ u(t-2),\ \ldots]$$

i.e. a notation for all available information up to and including time t.

Let the system be described by the model

$$y(t+k+1) + \alpha_1 y(t) + \ldots + \alpha_m y(t-m+1) =$$

$$= \beta_0 u(t) + \beta_1 u(t-1) + \ldots + \beta_\ell u(t-\ell) + \varepsilon(t+k+1) \quad (7.3)$$

Introduce

$$\hat{\alpha}_i(t+k+1) = E\left(\alpha_i(t+k+1) \,|\mathcal{Y}_t\right)$$

$$\hat{\beta}_i(t+k+1) = E\left(\beta_i(t+k+1) \,|\mathcal{Y}_t\right)$$

$$P_{\alpha_i\beta_i}(t+k+1) = E\Big[\left(\alpha_i(t+k+1) - \hat{\alpha}_i(t+k+1)\right) \cdot$$

$$\cdot \left(\beta_j(t+k+1) - \hat{\beta}_j(t+k+1)\right) |\mathcal{Y}_t\Big]$$

$$P_{\beta_i\beta_j}(t+k+1) = E\Big[\left(\beta_i(t+k+1) - \hat{\beta}_i(t+k+1)\right) \cdot$$

$$\cdot \left(\beta_j(t+k+1) - \hat{\beta}_j(t+k+1)\right) |\mathcal{Y}_t\Big]$$

i.e. the estimates and the variance of the estimation errors of the parameters at time t+k+1 in the model (7.3) based on data up to and including time t. The following theorem can now be stated. Compare [7, p. 100].

Theorem 7.1. Let the system be described by (7.3) where $\varepsilon(t+k+1)$ is independent of $\mathcal{Y}_t$ and $u(t)$. The loss function (7.1) is then minimized by the control strategy

$$u(t) = \frac{1}{\hat{\beta}_0(t+k+1)^2 + P_{\beta_0\beta_0}(t+k+1)} \cdot \Big[\sum_{i=1}^{m} \left(\hat{\alpha}_i(t+k+1)\hat{\beta}_0(t+k+1) + \right.$$

$$\left. + P_{\alpha_i\beta_0}(t+k+1)\right) y(t-i+1) - \sum_{i=1}^{\ell} \left(\hat{\beta}_0(t+k+1)\beta_i(t+k+1) + \right.$$

$$\left. + P_{\beta_0\beta_i}(t+k+1)\right) u(t-i)\Big] \quad (7.4)$$

Proof. Using basic properties for gaussian stochastic processes the conditional expectation (7.2) can be written as

$$
\begin{aligned}
E\left(y(t+k+1)^2 \mid \mathcal{Y}_t\right) = {} & \hat{\beta}_0^2 u(t)^2 + p_{\beta_0\beta_0} u(t)^2 - \\
& - 2u(t) \sum_{i=1}^{m} \hat{\alpha}_i \hat{\beta}_0 y(t-i+1) - \\
& - 2u(t) \sum_{i=1}^{m} p_{\alpha_i\beta_0} y(t-i+1) + \\
& + 2u(t) \sum_{i=1}^{\ell} \hat{\beta}_i \hat{\beta}_0 u(t-i) + \\
& + 2u(t) \sum_{i=1}^{\ell} p_{\beta_0\beta_i} u(t-i) + \\
& + \text{terms independent of } u(t)
\end{aligned}
$$

For simplicity the time arguments for $\hat{\alpha}_i$ and $\hat{\beta}_i$ are omitted. Take the derivative with respect to u(t) and set it equal to zero then

$$
\begin{aligned}
(\hat{\beta}_0^2 + p_{\beta_0\beta_0}) u(t) = {} & \sum_{i=1}^{m} (\hat{\alpha}_i \hat{\beta}_0 + p_{\alpha_i\beta_0}) y(t-i+1) - \\
& - \sum_{i=1}^{\ell} (\hat{\beta}_i \hat{\beta}_0 + p_{\beta_0\beta_i}) u(t-i)
\end{aligned}
$$

Divide by $\hat{\beta}_0^2 + p_{\beta_0\beta_0}$ and the theorem is proven. ■

Remark 1. If the variance matrix is equal to a zero matrix the control law (7.4) will be the same as the basic self-tuning algorithm when also β_0 is estimated.

Remark 2. If the parameter β_0 is not estimated then all covariances $p_{\alpha_i\beta_0}$ and $p_{\beta_0\beta_i}$ are equal to zero and the control law reduces exactly to the control law for the basic self-tuning algorithm.

In order to use the control law (7.4) the expressions $\hat{\alpha}_i$, $\hat{\beta}_i$, $p_{\alpha_i\beta_0}$ and $p_{\beta_0\beta_i}$ have to be evaluated. This can be done in special cases only. Let the system be described by an n:th order model

$$A(q)y(t) = B(q)u(t-k) + C(q)e(t) \tag{7.5}$$

Assume that $C(q) = q^n$ and $k = 0$, the model (7.5) can then be transformed to a model described by (7.3) where $\varepsilon(t+$ $+k+1)$ is white noise. The problem of evaluating $\hat{\alpha}_i$, $\hat{\beta}_i$ etc. now falls within the framework of Kalman filtering and the parameter estimates and the covariance matrix are given by (2.11), (2.12) and (2.13). The details are given in [7].

For $C(q) = q^n$ and a general k the model (7.5) can be transformed to a model (7.3) where $\varepsilon(t+k+1)$ is a moving average of order k. If the coefficients of the moving average were known it would be possible to use Kalman formalism to estimate $\hat{\alpha}_i$, $\hat{\beta}_i$ etc. But since the A and B polynomials are unknown the coefficients of the moving average will be unknown.

Since it is difficult to obtain the conditional distributions of the parameters the following algorithm could be attempted:

Step 1: Parameter estimation. Estimate the parameters α_1, ..., α_m, β_0, ..., β_ℓ in the model (7.3) using the least squares method described by the equations (2.11), (2.12) and (2.13).

Step 2: Control. Choose the control signal according to equation (7.4) where the estimates and covariance matrix are those obtained in Step 1.

The algorithm obtained will minimize the loss function (7.1) in the special case $C(q) = q^n$ and $k = 0$ only. The properties of the modified algorithm are now illustrated by an example.

Example 7.1. Consider the system

$$y(t) - 1.6y(t-1) + 0.8y(t-2) = 5u(t-1) + 2.5u(t-2) + e(t)$$

The parameters in the model

$$y(t+1) + \alpha_1 y(t) + \alpha_2 y(t-1) = \beta_0 u(t) + \beta_1 u(t-1) + \varepsilon(t+1) \quad (7.6)$$

are identified and the system is controlled with the regulator (7.4) using $P(0) \doteq 2 \times I$, $\lambda = 1$ and $|u| \leq 1.5$. The output of the system for $t = 0-50$ is shown in Figure 7.1a where also the output when using the optimal minimum variance regulator is shown.

If the parameters in (7.6) are identified and the control law

$$u(t) = \frac{\alpha_1 + \alpha_2 q^{-1}}{\beta_0 + \beta_1 q^{-1}} y(t) \quad (7.7)$$

is used then the output will be as shown in Figure 7.1b.

The output when using the basic self-tuning algorithm is shown in Figure 7.1c and d when β_0 is 2.5 and 5 respectively, i.e. the control law is

$$u(t) = \frac{\alpha_1 + \alpha_2 q^{-1}}{\beta_0(1+\beta_1 q^{-1})} y(t) \tag{7.8}$$

where β_0 is fixed.

The behaviour when using the different controllers is summarized in Table 7.1, where the accumulated losses at t = = 50 are given together with the average losses over the time interval t = 101 - 1100.

Regulator	$\sum_{1}^{50} y(t)^2$	$\frac{1}{1000} \sum_{101}^{1100} y(t)^2$	Number of times u(t) reaches the limit for t=0-50
Uncertainties considered (7.4)	68	0.99	0
Basic self-tuning algorithm with estimation of β_0 (7.7)	296	0.99	3
Basic self-tuning algorithm β_0 = 2.5	259	0.99	4
Basic self-tuning algorithm β_0 = 5	97	1.00	0
Basic self-tuning algorithm β_0 = 7.5	161	1.08	0
Optimal	48	0.99	0

Table 7.1 - The accumulated loss at t = 50 and the average loss over the period t = 101 - 1100 when the system in Example 7.1 is controlled by different regulators.

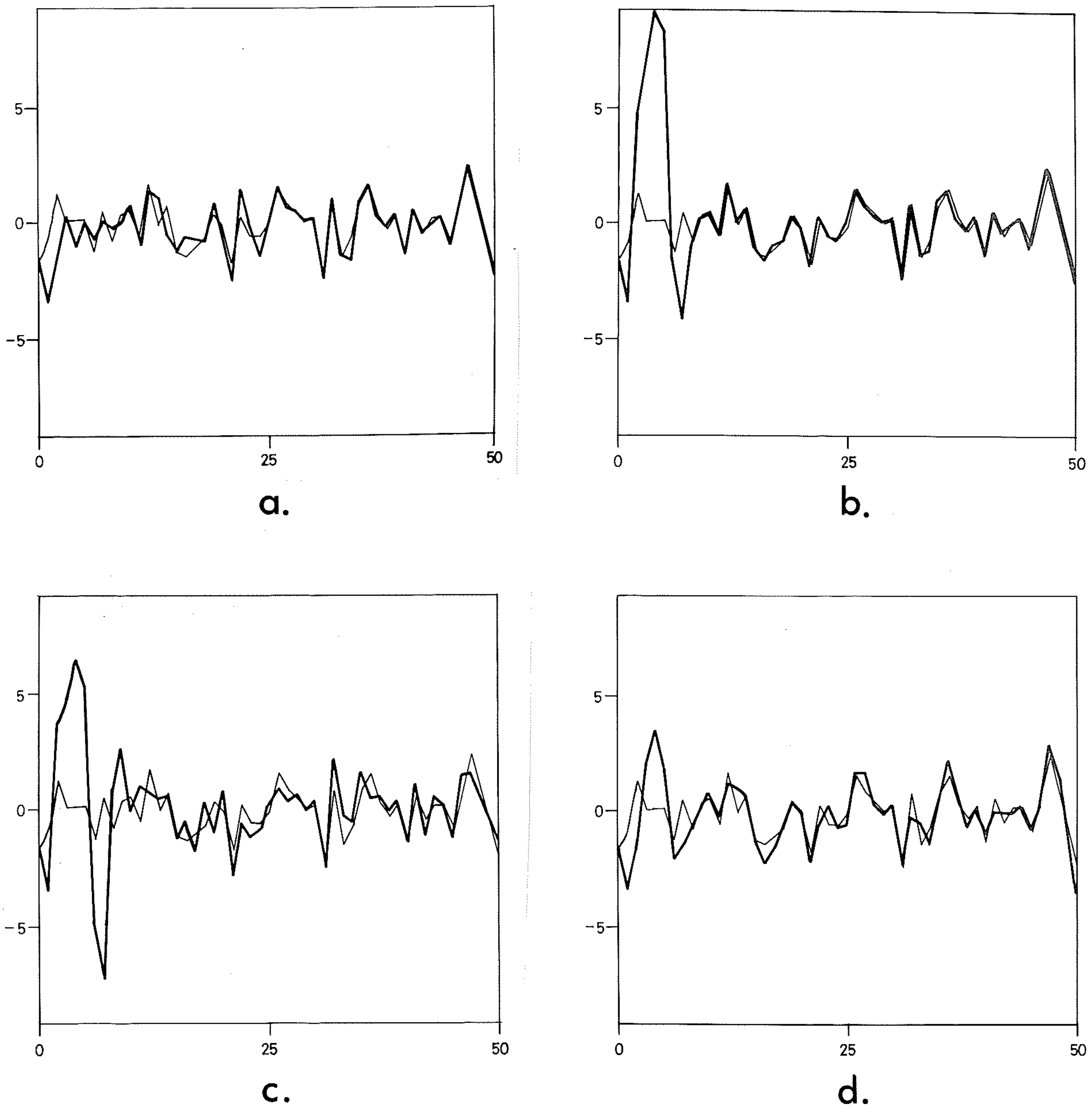


Fig. 7.1 - The output when the system in Example 7.1 is controlled by different regulators ($|u| \leqslant 1.5$).

a. Control law using parameter uncertainties β_0 estimated Eq (7.4)

b. Self-tuning regulator with β_0 estimated (7.7)

c. Basic self-tuning regulator, $\beta_0 = 2.5$ Eq. (7.8)

d. Basic self-tuning regulator, $\beta_0 = 5$ Eq. (7.8)

The thin line shows the output when the optimal regulator is used.

The table shows that the accumulated loss after 50 steps is lower for the modified algorithm (7.4) in comparison with the basic algorithm with the correct value of β_0. The table also shows that the accumulated loss at t = 50 is fairly sensitive to a selection of β_0 and that an attempt to estimate β_0, regulator (7.7) is worse than a fixed value in the range $0.5b_1 < \beta_0 < 2b_1$. The table also shows that after 1000 steps there are very small differences between the algorithms with exception of the basic self-tuning algorithm with $\beta_0 = 7.5$ which has a slightly higher loss.

It can also be seen from Figure 7.1 that the controller (7.7), which attempts to estimate β_0 has the largest loss at t = 50, converges quicker than the others to the optimal regulator. This indicates that the large control signals obtained initially helps the regulator to obtain good estimates. ■

The example above shows that the parameter uncertainties mainly influence the transient behaviour of the system. After a short period the effect of the uncertainties cannot be seen.

Simulations show that also for $k \neq 0$ the control law (7.4) can give a smaller loss in the beginning than the basic algorithm. If the control signal is limited harder there will be smaller differences between the algorithms. In a practical implementation precautions should be taken to prevent the regulator from introducing large disturbances before the parameters of the regulator have obtained reasonable values. This means that an identification period can be used when the algorithm only performs the estimation and when the control is done manually or by a fixed regulator. When reasonable parameter estimates have been obtained the fixed regulator can be disconnected and the self-tuning regulator takes over the control. Having this in mind it should not be necessary to use algorithms which take the parameter uncertainties into consideration, at least not when the parameters of the process are constant. If the process has rapidly time-varying parameters then it is probably more important to use parameter uncertainties in the control law, but the self-tuning regulator is, of course,not the proper tool to handle such processes.

In [7] and [31] it is shown that the algorithms using parameter uncertainties may behave in a strange manner. The control can for longer or shorter periods of time be turned off unintentionally. This so called "turn-off" phenomenon is analyzed and explained in [32] for a simple case. The turn-off has only occured when the parameters have been time-varying. The turn-off phenomenon has never occured when using the basic self-tuning algorithm or its modification which includes parameter uncertainties on systems with constant parameters in the extensive simulations that have been performed so far. If the system has rapidly varying parameters it may happen that the control can be turned off when controlling with algorithms which include parameter uncertainties.

The way to prevent turn-off is to use dual control strategies [15]. Dual control strategies for some simple examples are discussed in [7] and [11]. One simpler way to prevent turn-off is to introduce a perturbation signal [11], [31].

8. CLASSICAL DESIGN AND SELF-TUNING REGULATORS.

The basic self-tuning algorithm was introduced for the regulator problem. It will now be shown that the self-tuning algorithm can be modified to handle the servo problem. The problem of following command signals can be handled in many different ways. A simple approach will be discussed in this section. The point of departure is to incorporate a self-tuning regulator into a standard design procedure. A simple servo-loop will thus be designed where integrators are introduced in the usual way and a suitable self-tuning regulator will be constructed. The self-tuning regulator will be used as an ordinary compensator, see Figure 8.1. Notice that from the regulator's point of view the reference signal enters the system in exactly the same way as the disturbances. This basic idea can, of course, be applied in many different ways for many different design procedures.

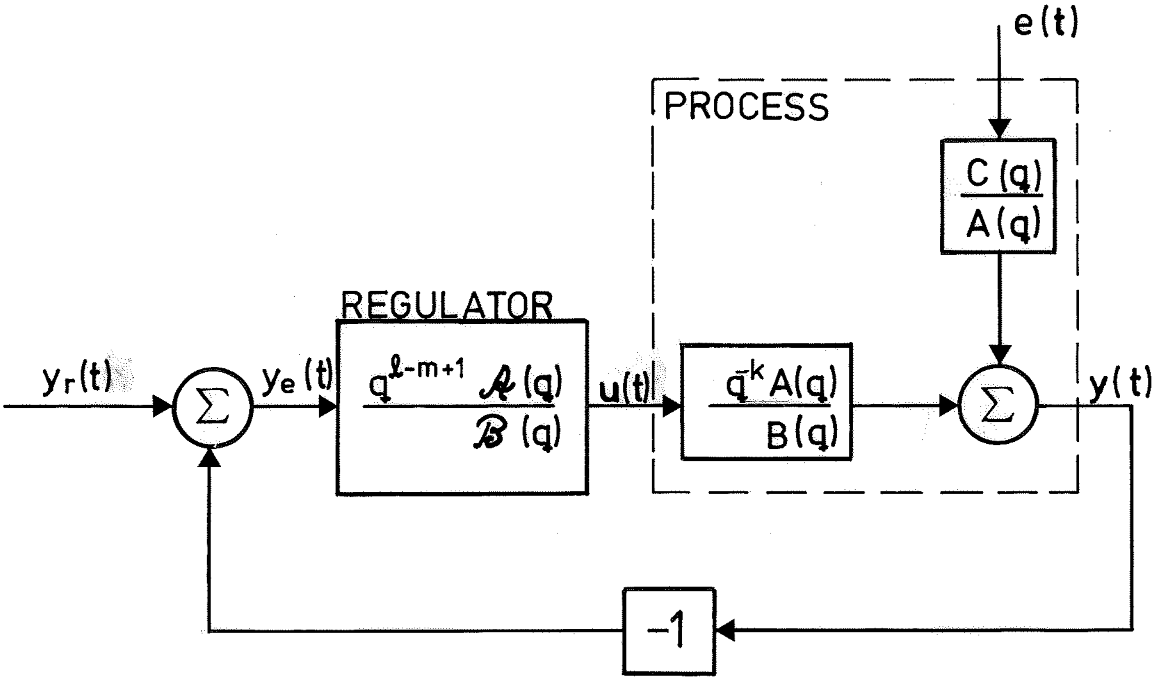


Fig. 8.1 - Block diagram for the systems considered in Section 8.

8.1. Constant Reference Values.

The minimum variance regulator will first be modified to handle constant reference values. This problem is discussed in [2]. When this problem is solved it is easy to see how self-tuning can be incorporated in the system.

Consider the system

$$A(q)y(t) = B(q)u(t-k) + C(q)e(t) \tag{8.1}$$

where the polynomials A, B and C are assumed known. It is further assumed that the system is minimum phase. Let the purpose of the control be to minimize the loss function

$$E\Sigma\left(y_r(t) - y(t)\right)^2 \tag{8.2}$$

where $y_r(t)$ is a known time-varying reference value.

Introduce the reference input, $u_r(t)$, defined as

$$u_r(t) = \frac{A(q)}{B(q)} y_r(t+k) \tag{8.3}$$

Equation (8.1) can now be written as

$$A(q)\left(y(t) - y_r(t)\right) = B(q)\left(u(t-k) - u_r(t-k)\right) + C(q)e(t)$$

The loss function (8.2) is minimized by the control strategy

$$u(t) - u_r(t) = -\frac{q^kG}{BF}\left(y(t) - y_r(t)\right) \tag{8.4}$$

where F and G are polynomials given by the identity

$$q^k C(q) = A(q)F(q) + G(q)$$

The control law (8.4) can be written as

$$u(t) = -\frac{q^k G}{BF}\, y(t) + \frac{q^{k-1} C}{BF}\, y_r(t+k+1)$$

To compute the optimal control law it is thus necessary to know the reference value in at least k+1 steps ahead.

In steady state control it is reasonable to assume that the reference value is constant for long periods of time. This will simplify the regulator given above. For a constant reference value (8.3) is changed to

$$u_r = \frac{A(1)}{B(1)}\, y_r$$

If there is an integrator in the system, i.e. A(z) has at least one zero more than B(z) in z = 1, then $u_r = 0$. This implies that (8.4) can be written as

$$u(t) = -\frac{q^k G}{BF}\,\big(y(t) - y_r\big) \tag{8.5}$$

If there is no integrator in the system it is possible to introduce one by changing the controlled variable from u(t) to

$$\nabla u(t) = u(t) - u(t-1)$$

i.e. the control signal is selected as the increment of the control variable. The equation describing the system will be changed to

$$A(q)(q-1)y(t) = B(q) \cdot q \cdot \nabla u(t-k) + C(q)(q-1)e(t)$$

and the optimal control law is given by

$$\nabla u(t) = -\frac{q^{k-1}(q-1)G}{BF}\left(y(t) - y_r\right)$$

where F and G are the same polynomials as before. The minimum variance is not influenced through the introduction of the integrator.

The regulator (8.5) for known systems has the same structure as the regulator (2.7) which is used in the basic self-tuning regulator. The only difference is that the regulator (8.5) uses the signal $y(t) - y_r(t)$ instead of the process output $y(t)$ only.

To extend the self-tuning regulators to the case with constant reference values it is assumed that the system contains an integrator or that one is introduced by using $\nabla u(t)$ as control signal. The following algorithm can now be proposed:

Step 1: Estimation. Estimate the parameters $\alpha_1, \ldots, \alpha_m$, $\beta_1, \ldots, \beta_\ell$ in the model

$$y(t+k+1) - y_r + \alpha_1\left(y(t) - y_r\right) + \ldots + \alpha_m\left(y(t-m+1) - y_r\right) =$$

$$= \beta_0[\nabla u(t) + \beta_1\nabla u(t-1) + \ldots + \beta_\ell\nabla u(t-\ell)] + \varepsilon(t+k+1)$$

using the least squares method. The parameter β_0 is assumed known.

Step 2: Control. Determine the control signal from

$$\nabla u(t) = \frac{1}{\beta_0} \sum_{i=1}^{m} \alpha_i \left(y(t-i+1) - y_r\right) - \sum_{i=1}^{\ell} \beta_i \nabla u(t-i)$$

where α_i and β_i are the least squares estimates obtained in Step 1.

Notice that this algorithm is identical to the basic self-tuning algorithm if y is replaced by $y(t) - y_r$ and $u(t)$ by $\nabla u(t)$. The properties of the algorithm will now be demonstrated.

Example 8.1. Let the system be

$$y(t) - 0.9y(t-1) = u(t-1) + 0.2\left(e(t) - 0.5e(t-1)\right)$$

The system does not contain any integrator and the control signal is therefore changed to $\nabla u(t)$. The minimum variance regulator is

$$\nabla u(t) = - 0.4\left(y(t) - y_r\right) + 0.4\left(y(t-1) - y_r\right)$$

Part of the output (t = 400 - 600) is shown in Figure 8.2 when controlling with $m = 2$, $\ell = 0$, $\beta_0 = 1$, $\alpha_1(0) = -0.5$ and $\alpha_2(0) = 0.5$. At time t = 500 the reference value is changed from 2 to 4. The accumulated loss is shown for 1000 steps of time in Figure 8.3.

It is clear from the figures that the regulator has an acceptable performance. The step increase in the loss at t = 500 is, of course, unavoidable due to the structure of the problem.

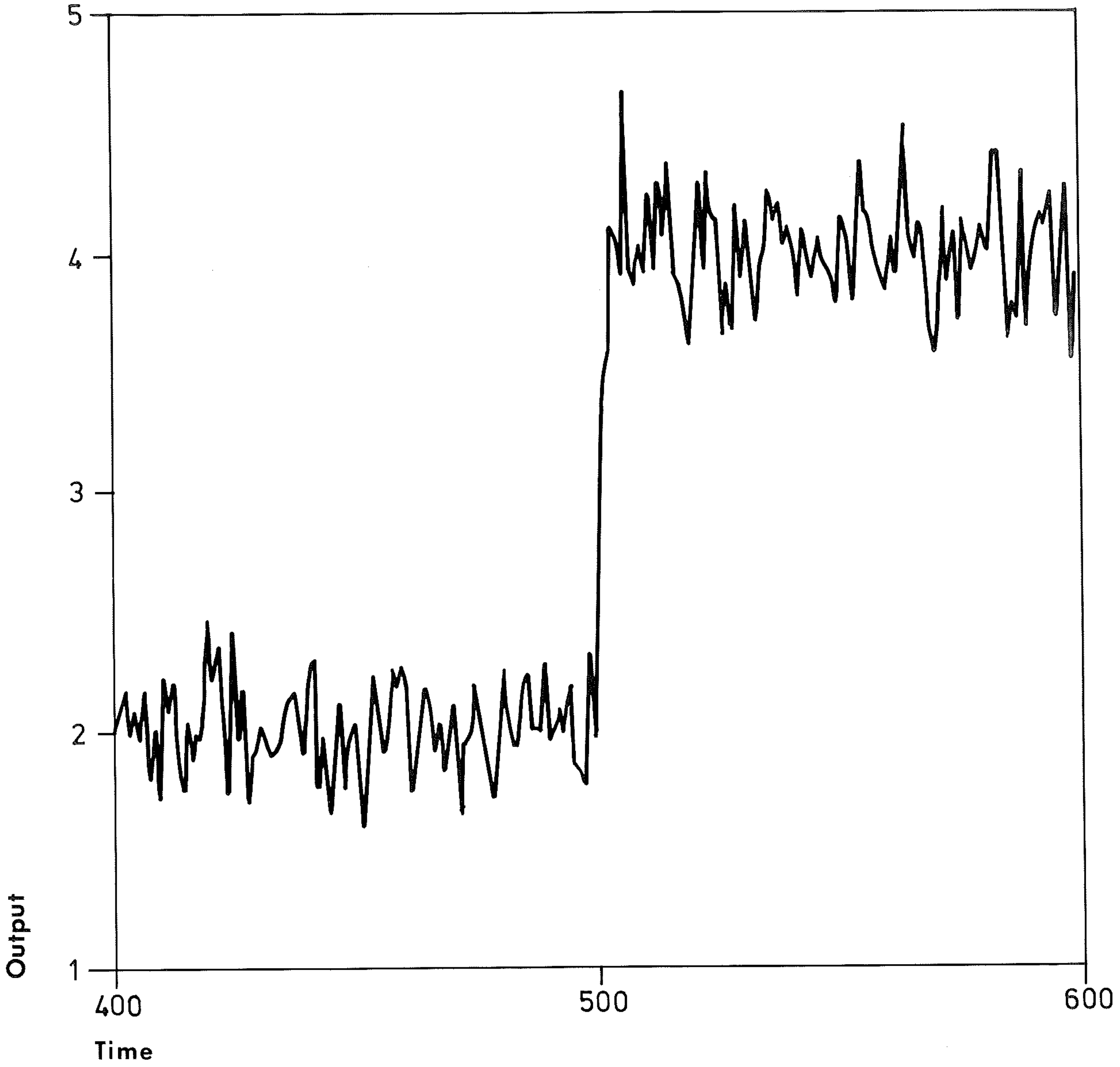


Fig. 8.2 - The output of the system in Example 8.1 for t = 400 - 600 when the reference value is changed from 2 to 4 at time t = 500.

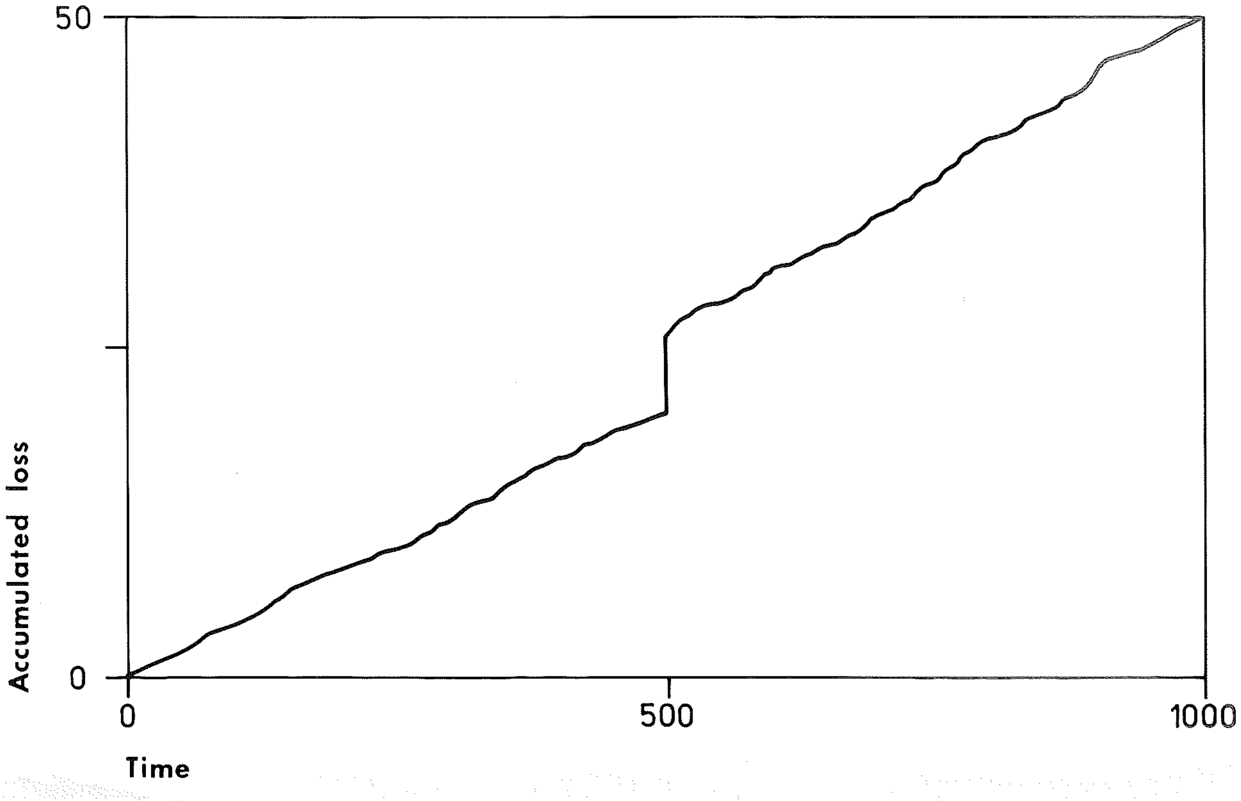


Fig. 8.3 - Accumulated loss for the system in Example 8.1. ■

In the example above the parameter estimates were not much influenced by the single change in the reference value. How much changes in the reference value influence the parameter estimates depends on how often the changes occur and how well the estimates have converged.

8.2. Time-Varying Reference Signals.

The servo problem will now be considered. To simplify it is first assumed that there are no disturbances. Integrators are introduced in the system to make it possible for the output to follow reference signals of different types in the usual way. The conventional regulator is then replaced with a self-tuning regulator in analogy with the case of constant reference values.

One class of regulators for sampled data system is the one which makes the error zero as quickly as possible. The regulators which achieve this will depend on the reference signals. For impulse changes the regulators are dead-beat regulators. There seems to be no accepted name for this class of regulators for other reference signals. These regulators will here be called generalized dead-beat regulators.

Dead-beat regulators are the deterministic correspondence to the minimum variance regulators. When there are no disturbances and the reference signal is time-varying it can thus be expected that the self-tuning regulator converges to a generalized dead-beat regulator. That this assumption may be true can be seen from the following example.

Example 8.2. Consider the system

$$(1-1.5q^{-1}+0.7q^{-2})(1-q^{-1})^2y(t) = (1+0.5q^{-1})u(t) \tag{8.6}$$

which contains two integrators. Let the reference value be a triangular wave with amplitude 4 and a period of 50 steps. The system is controlled by a self-tuning regulator with $m = 4$ and $\ell = 1$. The initial covariance matrix was $P(0) = 10 \times I$ and the exponential forgetting factor

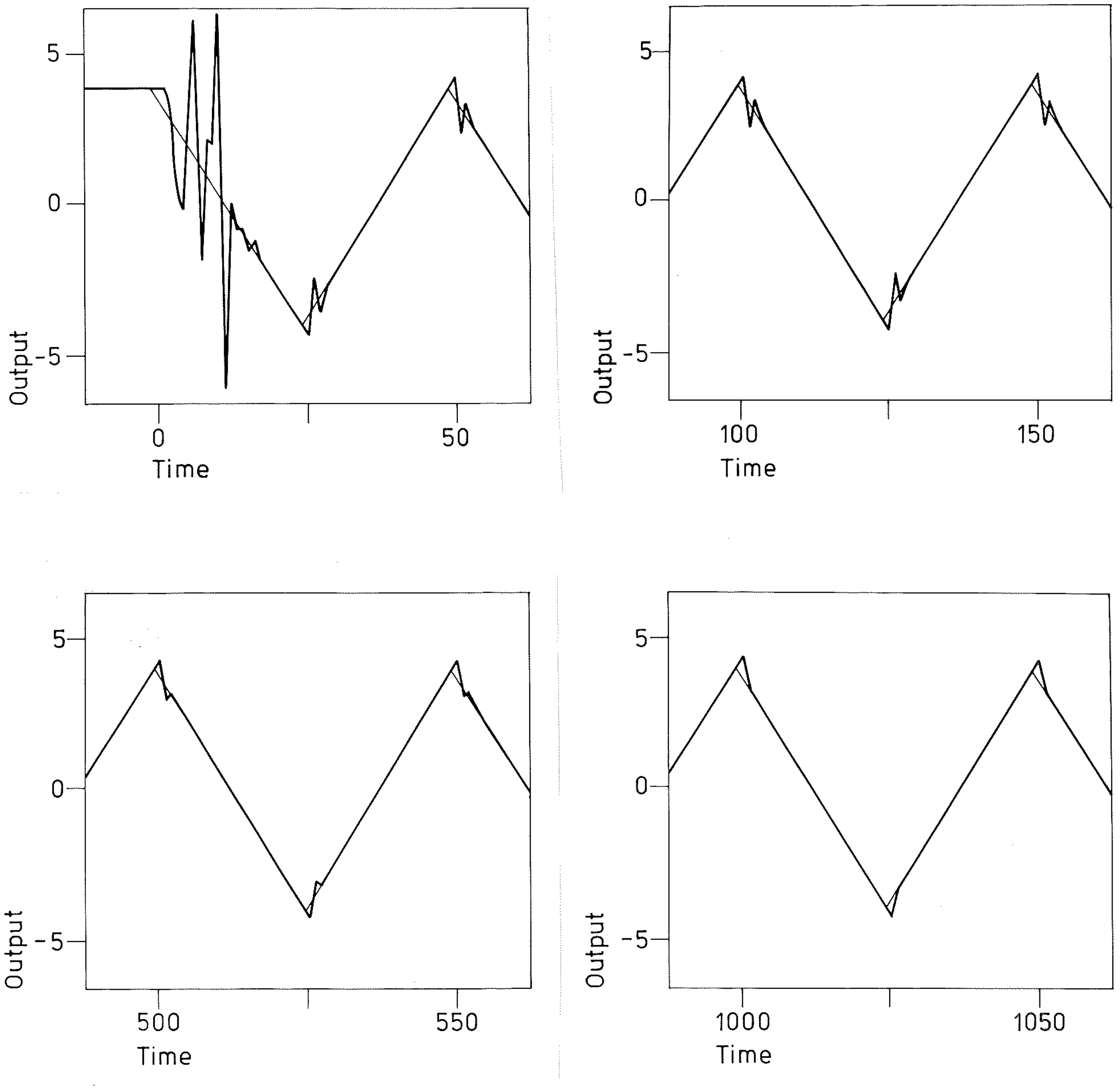


Fig. 8.4 - Output of the system in Example 8.2. The reference signal is a triangular wave and is indicated by the thin line.

was $\lambda = 0.99$. The output from the system at different periods of time is shown in Figure 8.4. From the figure it is seen that the self-tuning regulator converges to a regulator with the property that the error is zero after one step after a change in the reference value has occured. The regulator obtained after 2000 steps (80 changes in the reference signal) was

$$u(t) = \frac{2.00 - 4.00q^{-1} + 2.89q^{-2} - 0.70q^{-3}}{1 + 0.50q^{-1}} y_e(t)$$

This is almost the same as the generalized dead-beat regulator of the system (8.6) when the reference signal is a triangular wave. The generalized dead-beat regulator is in this case

$$u(t) = \frac{(1-1.5q^{-1}+0.7q^{-2})(2-q^{-1})}{1 + 0.5q^{-1}} y_e(t)$$

which gives the closed loop system

$$y(t) = (2-q^{-1})y_r(t-1)$$

For the system (8.6) the self-tuning algorithm converges to the same controller independent of the amplitude and period of the triangular reference signal as long as the system has time to settle between the changes. ■

Simulations have shown that the self-tuning algorithm converges to a generalized dead-beat regulator when the noise is equal to zero and the reference signal is a square or a triangular wave. The period of the reference signal must, however, not be too short. It has not yet been possible to give a rigorous proof, but simulations clearly indicates that the self-tuning really converges to a generalized dead-beat regulator which depends on the properties of the reference signal.

Let the system to be controlled be described by the model

$$(q-1)A(q)y(t) = B(q)u(t-k)$$

If the reference signal is a square wave the generalized dead-beat regulator is

$$u(t) = \frac{q^k A(q)}{B(q)(q^k+q^{k-1}+\ldots+1)} y_e(t)$$

The closed loop system will then be

$$y(t) = y_r(t-k-1)$$

If the system to be controlled contains two integrators, i.e. it is described by

$$(q-1)^2 A(q) = B(q)u(t-k)$$

and if the reference signal is a triangular wave then the generalized dead-beat regulator is

$$u(t) = \frac{q^k A(q)\left[(k+2)q - (k+1)\right]}{B(q)(q^k+2q^{k-1}+\ldots+kq+k+1)} y_e(t)$$

The closed loop system will be

$$y(t) = \left[k + 2 - (k+1)q^{-1}\right]y_r(t-k-1)$$

Simulations have shown that it is not necessary that the system to be controlled contains any integrators. If the self-tuning regulator has enough parameter the integrators will appear in the $\mathfrak{B}$-polynomial. The convergence rate is, however, very much slower compared to the case when integrators are introduced. From a practical point of view it thus seems desirable to include a priori knowledge of the gross features of the reference signals by introducing integrators.

It is well-known that dead-beat regulators as well as minimum variance regulators can be very sensitive to changes in the regulator parameters if the system is nonminimum phase. To investigate the sensitivity of the regulator obtained when using the self-tuning algorithm it is for simplicity assumed that the reference signal is a square wave. Let the system be

$$(q-1)A^{o}(q)y(t) = B^{o}(q)u(t-k)$$

Assume that the regulator is

$$u(t) = \frac{q^{k}A(q)}{B(q)(q^{k}+q^{k-1}+...+1)}\left(y_{r}(t) - y(t)\right)$$

The closed loop system will be

$$\left(A^{o}Bq^{k+1} + (AB^{o}-A^{o}B)\right)y(t) = AB^{o}y_{r}(t)$$

If $A = A^{o}$ and $B = B^{o}$ the characteristic polynomial reduces to $q^{k+1} A^{o}B^{o}$ and the factor $A^{o}B^{o}$ cancels. For small perturbations in the parameters the poles of the closed loop system are close to the zeroes of $q^{k+1} A^{o}B^{o}$. If $A^{o}B^{o}$ has any zero outside the unit circle the system will be very sensitive for variations in the parameters of the controller, compare [1, p. 181]. The generalized dead-beat regulator will thus be sensitive if the system is nonminimum phase and also if the system is unstable. This is also verified by simulations with the self-tuning regulator. The difficulties with nonminimum phase systems can be handled in the same way as in Section 6, i.e. increase the number of time-delays used in the identification. Unstable systems can be stabilized using a fix regulator. The self-tuning algorithm can then be used to control the stabilized system.

It is difficult to analyze the servo problem when the reference value varies with time and disturbances are included. Since the algorithm converges to the minimum variance regulator when the reference value is constant and to a generalized dead-beat regulator when the noise is zero it can be conjectured that the algorithm will converge to different controllers depending on the relative magnitude of the noise and the variation in the reference signal. This is illustrated in the following example.

Example 8.3. Consider the system

$$y(t) - 1.9y(t-1) + 0.9y(t-2) = u(t-1) + \sigma e(t)$$

Let the reference signal be a square wave with amplitude 4 and period 400 steps. The system is simulated for different values of the standard deviation of the noise, σ, and with $m = 2$ and $\ell = 0$. The results of simulations are shown in Table 8.1.

The regulator changes from the dead-beat regulator ($\alpha_1 = -1.00$, $\alpha_2 = 0.90$) to the minimum variance regulator ($\alpha_1 = -1.90$, $\alpha_2 = 0.90$) when the intensity of the noise is increased.

The example shows that different types of exciting signals will make the algorithm converge to different regulators.

σ	α_1	α_2	Loss per step divided by σ^2 when the ref. value is constant	Loss for one unit change in the ref. value when the noise is zero
0	-1.00	0.90	5.26	1.00
0.1	-1.09	0.89	2.85	1.03
0.2	-1.20	0.88	1.89	1.09
0.5	-1.51	0.90	1.18	1.31
1.0	-1.71	0.90	1.04	1.52
2.0	-1.83	0.91	1.00	1.69
4.0	-1.87	0.91	1.00	1.76

Table 8.1 - Parameter estimates for the system in Example 8.3 when having different noise amplitudes. The parameter values α_1 = -1.00, α_2 = 0.90 correspond to the generalized dead-beat and α_1 = -1.90 and α_2 = 0.90 correspond to the minimal variance regulator. The table also shows the loss that would be obtained for the different regulators when only having noise or a time-varying reference value is acting on the system. ■

Simulations show that self-tuning regulators can be used to tune controllers both for the regulator and the servo problem. The simple structure that is used here to take care of the reference signals has, however, some drawbacks for the servo problem. One drawback is that it is difficult to influence the transient behaviour of the closed loop system. Consider for instance the case when the reference signal is a square wave. The self-tuning regulator used makes the output equal to the reference signal delayed as many steps as there are time-delays in the system. It can, however, be desirable that the step response is a more smooth signal. This can for instance be arranged by using a model which has the desired step response. The difference between the output of the model and the output of the system can then be used as the input signal to the self-tuning controller. This solution has the drawback that the controller only gets information about changes in the reference signal via the model. A better performance can be obtained if the controller also gets direct information about changes in the reference signal. This can be taken care of by a feedforward signal from the reference value.

The self-tuning algorithm has the property that it adapts its parameters depending on the environment. The regulator will converge to a generalized dead-beat regulator if the changes in the reference signal are dominating and to a minimum variance regulator if the noise is dominating. This adaption in the parameters can also be regarded as an adaption in the performance criterion in the sense that the controller changes from a controller for transient control to a controller for steady state control.

9. REFERENCES.

[1] Åström, K.J.: Introduction to Stochastic Control Theory, Academic Press, New York, 1970.

[2] Åström, K.J.: Notes on the Regulation Problem, Internal Report CT 211, Aug., 1965, IBM Nordic Laboratory.

[3] Åström, K.J.: Lectures on the Identification Problem - The Least Squares Method, Report 6806, Sept., 1968, Division of Automatic Control, Lund Institute of Technology.

[4] Åström, K.J.: Linear Samples System with Time-Delay which is a Fraction of the Sampling Period, Report 7217(B), Division of Automatic Control, Lund Institute of Technology.

[5] Åström, K.J.: A Self-tuning Regulator for Non-minimum Phase Systems, to appear, Division of Automatic Control, Lund Institute of Technology.

[6] Åström, K.J., Eykhoff, P.: System Identification - A Survey, Automatica 7, p. 123 - 162, 1971.

[7] Åström, K.J., Wittenmark, B.: Problems of Identification and Control, Journal of Mathematical Analysis and Applications, 34, p. 90 - 113, 1971.

[8] Åström, K.J., Wittenmark, B.: On Self-Tuning Regulators, Automatica, 9, p. 185 - 199, 1973.

[9] Bar-Shalom, Y., Sivan, R.: On the Optimal Control of Discrete Time Linear Systems with Random Parameters, IEEE T-AC, 14, p. 3 - 8, Febr., 1969.

[10] Bohlin, T.: On the Problem of Ambiguities in Maximum Likelihood Identification, Automatica, 7, p. 199 - - 210, 1971.

[11] Bohlin, T.: Optimal Dual Control of a Simple Process with Unknown Gain, TP 18.196, Febr., 1969, IBM Nordic Laboratory.

[12] Borisson, U., Wittenmark, B.: An Industrial Application of a Self-Tuning Regulator, Report 7310, Division of Automatic Control, Lund Institute of Technology.

[13] Dickson, L.E.: First Course in the Theory of Equations, Wiley, New York, 1922.

[14] Elmqvist, H.: SIMNON - Ett interaktivt simuleringsprogram för olinjära system (in Swedish), RE-113, Division of Automatic Control, Lund Institute of Technology.

[15] Feldbaum, A.A.: Dual Control Theory I-IV, Automation and Remote Control, 21, p. 874 - 880, p. 1033 - 1039, 1960, 22, p. 1 - 12, p. 109 - 121, 1961.

[16] Kai Lai Chung.: A Course in Probability Theory, Harcourt, Brace & World Inc., 1968.

[17] Källström, C., Åström, K.J.: Real Time Computing III: Implementing Linear Filtering and Control Algorithms, Report 7122, Division of Automatic Control, Lund Institute of Technology, 1971.

[18] Landau, I.D.: A Hyperstability Criterion for Model Reference Adaptive Control, IEEE T-AC, 14, p. 552 - 555, Oct., 1969.

[19] Lee, R.C.K.: Optimal Estimation, Identification and Control, MIT Press, Cambridge, Mass., 1964.

[20] Ljung, L.: Convergence Concepts for Adaptive Structures, Report 7218(B), Division of Automatic Control, Lund Institute of Technology.

[21] Ljung, L., Wittenmark, B.: Asymptotic Properties of Self-Tuning Algorithms, to appear, Division of Automatic Control, Lund Institute of Technology.

[22] Maršik, J.: A Simple Adaptive Controller, IFAC Symposium on Identification and Process Parameter Estimation, Prague, 1970, Paper No. 6.6.

[23] Parks, P.C.: Liapunov Redesign of Model Reference Adaptive Control Systems, IEEE T-AC, 11, p. 362 - - 367, July, 1966.

[24] Peterka, V.: Adaptive Digital Regulation of Noisy Systems, IFAC Symposium on Identification and Process Parameter Estimation, Prague, 1970, Paper No. 6.2.

[25] Peterka, V.: On Steady State Minimum Variance Control Strategy, Kybernetika, 8, (1972), No 3, p. 219 - 231.

[26] Saridis, G.N., Dao, T.K.: A Learning Approach to the Parameter Adaptive Self-Organizing Control Problem, Automatica, 8, p. 589 - 597 (1972).

[27] Tse, E., Athans, M.: Adaptive Stochastic Control for a Class of Linear Systems, IEEE T-AC, 17, No. 1, Febr., 1972, p. 38 - 51.

[28] Tsypkin, Ya.Z.: Adaption and Learning in Automatic Systems, Academic Press, New York (1971).

[29] Wade, H.L., Schoeffler, J.D.: A Technique for Adaptive Feeforward Control, 63rd Annual Meeting American Institute of Chemical Engineers, Chicago, Illinois, 1970.

[30] Wieslander, J.: Real Time Identification Part I, Report 6908, Division of Automatic Control, Lund Institute of Technology.

[31] Wieslander, J., Wittenmark, B.: An Approach to Adaptive Control Using Real Time Identification, Automatica, 7, p. 211 - 217, 1971.

[32] Wittenmark, B.: On the Turn-Off Phenomenon in Adaptive Control, Report 7105, Division of Automatic Control, Lund Institute of Technology.

AN INDUSTRIAL APPLICATION OF A SELF-TUNING REGULATOR

U. BORISSON

B. WITTENMARK

Report 7310 April 1973
Lund Institute of Technology
Division of Automatic Control

AN INDUSTRIAL APPLICATION OF A SELF-TUNING REGULATOR.

U. Borisson B. Wittenmark

ABSTRACT.

An implementation of a self-tuning regulator is described. The regulator algorithm consists of two steps. Firstly the parameters of a model of the process are estimated in real time using the method of least squares. Secondly a minimum variance regulator based on the estimated parameters is computed at each sampling interval. Implementation of the algorithm on a process computer is discussed. Questions concerning program structure and storage requirements are considered. Results from the moisture content control on a paper machine are presented. The experiences are very good concerning the start-up behaviour as well as the stationary behaviour of the self-tuning regulator.

TABLE OF CONTENTS

1. INTRODUCTION.

More and more industrial processes are controlled by process computers. The computers make it possible to use more complex regulators than the conventional PID-regulators. The administration of the data handling and the computation of control signals are generally done by a special program package (DDC package). The program packages consist of a collection of standardized regulators and data handling routines that can be used for different control loops. Often it is also possible for the user to include his own subroutines. The regulators are mostly discrete time versions of PID-regulators, sometimes with addition of dead-time compensation based on the idea originally proposed by O.J.M. Smith. These standardized regulators have simple structures which are characterized by few parameters. In many applications it is, however, desirable to have more complex regulators, but these seldom are implemented since they are difficult to tune. A badly tuned complex regulator can make a poorer control than a well-tuned simple regulator. It is thus desirable to have some kind of automatic tuning of the regulator parameters.

One way is to use an adaptive controller that automatically can adjust its parameters depending on the changing environment in which the regulator works. One special type of adaptive controllers are the self-tuning regulators, which are discussed in [2], [3] and [6]. The regulator tunes its parameters on-line. It is assumed that the processes to be controlled have constant but unknown parameters. Further, it is assumed that the processes have one single input and one single output and that they are minimum phase. The algorithm of the self-tuning regulator can be divided into two steps, identification and control. The algorithm performs a real time identification of the parameters in a model of the process using the method of least squares. The control law is a minimum variance regulator based on the estimated parameters. The regulator can be used to tune parameters in feedback as well as feedforward loops. Other types of self-tuning regulators are discussed e.g. in [4] and [5].

The theoretical aspects of the self-tuning regulators, considered in this report, are thoroughly discussed in [2] and [6]. The main result is that if the parameter estimates converge, certain covariances of the output and cross-covariances between the output and the input are equal to zero. Further, if the number of parameters of the regulator is large enough and the process to be controlled can be described by a linear finite dimensional model the regulator will converge to the minimum variance regulator that could be obtained if the parameters of the process were known. In practice, the algorithm has shown good convergence properties. Experiments on laboratory as well as industrial processes have shown that self-tuning regulators are well suited for industrial applications.

In this report it is discussed how the self-tuning regulators can be implemented on process computers and results are given from control experiments on a paper machine. The algorithm which is quite general can be applied to many types of processes.

In Section 2 it is discussed how the self-tuning regulators can be used and how program structure and data bases can be organized. Moisture content control on a paper machine is described in Section 3. Results from experiments on a paper machine at the Gruvön Mill of Billerud Company in Sweden are given in Section 4. Experiences of the experiments are summarized in Section 5.

The authors want to express their gratitude to Professor K.J. Åström for his support and encouragement in the work with the self-tuning regulators.

The experiments have been possible to make thanks to financial support from STU (The Swedish Board for Technical Development) and to the kindness of Billerud Company. A special thank to O. Alsholm, L. Endresen and O. Stavnes at the Gruvön Mill at Grums for their help and support during the experiments.

2. IMPLEMENTATION OF SELF-TUNING REGULATORS ON PROCESS COMPUTERS.

2.1. The Use of Self-Tuning Regulators.

Sometimes it is necessary to use a regulator with many parameters. This is for instance the case when:

- the process has several time-delays,
- feedforward compensation is used.

In these cases it can be favourable to use a self-tuning regulator, because it is difficult to do the tuning manually.

The theoretical results concerning the self-tuning regulators are obtained under the assumption that the parameters of the process controlled are constant but unknown. In [6] it is discussed how the algorithm can be modified to follow slowly time-varying parameters by introducing a weighting factor, λ, less than one.

The self-tuning regulators can be used in different ways:

- A self-tuning regulator can be used at the installation or retuning of a regulator loop. It can be removed when a proper parameter set is obtained.

- The self-tuning regulator can be installed among the other system programs in the computer and periodically serve different control loops.

- If the process has time-varying parameters, it may be desirable to have the self-tuning regulator connected to the regulator loop all the time.

2.2. Program Structure.

When implementing the algorithm on a process computer it is sometimes advantageous to divide the algorithm into two parts, one for the estimation and one for the control. If the algorithm is implemented on a computer having a DDC package the control part in many cases can be implemented using the standard set of regulators defined in the DDC package. The estimation or tuning part then delivers the regulator parameters to the data base used by the DDC package. If a regulator structure is used that is not available among the standard routines it is necessary to write a special routine for the control part.

The tuning part must be specially written and included in the system programs. This routine can be used for many different loops if special care is taken concerning the storage of data.

If the routines are written straightforwardly using standard FORTRAN, the program length will be about 31 statements for the tuning part and about 28 statements for the control part. When compiled on an IBM 1800 with floating point hard-ware the storage requirement will be about 448 memory cells for the tuning part and about 310 memory cells for the control part. The storage requirements can be considerably reduced if the structure of the estimator equations is utilized.

The data needed for the self-tuning algorithm can be divided into three types: state variables, dummy variables and constants. The state variables are input-output signals, parameter estimates and their covariances. The dummy variables are temporary variables used at only one interval of time. The constants define the regulator structure.

Let the number of parameters of the regulator be r and assume that there are k steps of time-delay in the process. The data storage of the two routines together will be

State variables	$0.5r^2 + 2.5r + 3k + 6$
Dummy variables	r
Constants	5

It has then been utilized that the covariance matrix is symmetric. The control routine needs 4 constants and 2r state variables. If the tuning algorithm serves only one loop at a time it can use the same data area for all loops.

2.3. Selection of Parameters and Start-Up Procedure.

Simulations and experiments have shown that the choice of parameters in the algorithm is not crucial. The only parameter that must be chosen with some care is the number of time-delays, k. An under-estimation of k can be serious, while an over-estimation of k with one or two sampling intervals is not serious. The choice of parameters is discussed thoroughly in [6].

The start-up procedure can be made in different ways. It is always suitable to limit the control signal. Such a limit test is included in most DDC packages. The initial values of the parameters can be arbitrary. It is suitable, however, to limit the control signal comparatively much in the beginning. After a short period of time the parameter estimates are good enough to give successful control, and the limit of the control signal can be eased.

Another way to start up is to use good initial values of the parameters. For instance, these values can be obtained by letting the control routine work as a simple PI-regulator for a short period of time with fixed parameters, while the tuning routine is running. The estimates of the regulator parameters and the value of the covariance matrix obtained in this way generally work very well as initial values when the real tuning starts.

3. MOISTURE CONTROL ON PAPER MACHINES.

One of the control problems when producing paper is to keep the moisture content of the paper on a desired level. The moisture content is influenced by many variables e.g. the basis weight, the degree of refining and the quality of the pulp.

The moisture content control loop at the Gruvön Mill will now be described. The experiments in the next section were made on a machine producing fluting. The production is approximately 130 000 ton/year. The basis weight range of the fluting is 112 - 150 g/m^2. The paper machine as well as the refiners and the digesters are controlled from an IBM 1800 computer. A DDC package, PPCP (Process and Production Control Package), is available on the computer [1].

The moisture content of the paper is controlled primarily with a feedback loop from a capacitive moisture gauge, to the steam pressures of the drying cylinders of the last two drying sections. See Figure 3.1. The moisture gauge is kept in a fixed position. No averaging of the moisture content over the machine width is done.

The moisture content is influenced by several process variables and it is advantageous to use feedforward compensation. The thick stock flow, which influences the basis weight, or the reference value of the refiners are signals that can be used for feedforward control of the moisture. On the paper machine used in the experiments a feedforward signal from the couch vacuum was used. This has the advantage that disturbances from the refiners as well as basis weight fluctuations originating from the head box and the wire are taken into account.

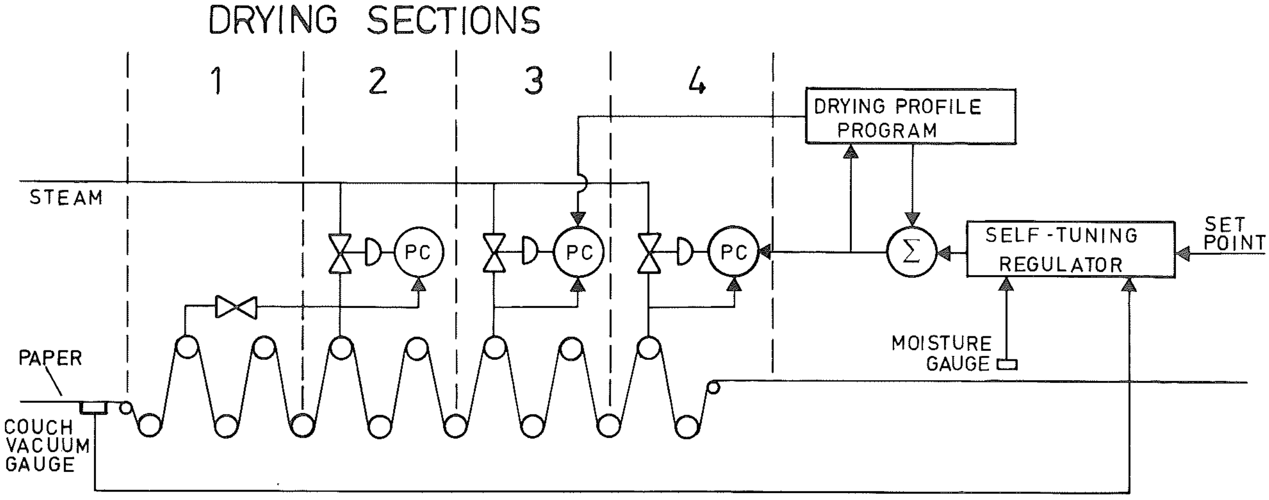


Fig. 3.1 - Schematic figure of the drying sections and the moisture content regulation at the Gruvön Mill when the self-tuning regulator was used.

It is desirable to have a certain drying profile, i.e. to have a certain relationship between the pressures of the different drying sections. Therefore, a special program which is able to adjust the drying profile is included in the system. The pressure of the fourth section is allowed to vary in the interval 0 - 1 kp/cm^2 below the pressure of the third drying section. The drying profile program changes the pressure of the third section if a pressure outside the limits of the fourth drying section is required.

The reference value of the fourth drying section thus can be influenced by the moisture and couch vacuum signals and the drying profile program.

When the self-tuning regulator was used to control the moisture content the sampling interval was 16 seconds. The moisture and couch vacuum signals were measured and filtered every 8th second. The filter used for the moisture signal was part of the DDC package and had a short time constant.

4. EXPERIMENTS AT GRUVÖN.

Several experiments with the self-tuning algorithm were made using different regulator structures. The number of regulator parameters and the number of time-delays were varied. The regulator was tested under different operating conditions including great process disturbances. Basis weight changes and machine speed adjustments were also investigated.

Two examples will be discussed, where the self-tuning regulator controlled the moisture content of the paper machine. The structure of the regulator was different in the two cases.

Example 1. The number of regulator parameters was 6. Four parameters were used in the feedback loop and two in the feedforward loop. The regulator had the structure

$$\nabla u(t) = \alpha_1 y(t) + \alpha_2 y(t-1) - \beta_1 \nabla u(t-1) - \beta_2 \nabla u(t-2) +$$

$$+ \gamma_1 \nabla v(t) + \gamma_2 \nabla v(t-1)$$

where

∇u - incremental control signal
y - moisture content deviation
∇v - increment in coach vacuum signal.

The time-delay in the process, k, was assumed to be four sampling intervals.

The following process variables were registered during the experiment:

- moisture content (the set point is indicated by a line) [%]
- incremental control signal (∇u) [engineering units]
- steam pressure of the fourth drying section (u) [kp/cm^2]
- steam pressure of the third drying section (reference value) [kp/cm^2]
- couch vacuum signal [m H_2O]

The experiment started by letting the tuning algorithm run for about ten minutes having a PI-regulator with fixed parameters to compute the control signal. The estimated parameters obtained were used as initial values when the self-tuning regulator started to control. The results obtained are shown in Fig. 4.1 - 4.8.

Changes in refiner energy often are possible to observe in the couch vacuum registration.

At 21.42 the refiner energy was decreased by the process operator. This adjustment can be observed in the moisture content and couch vacuum registrations (Fig. 4.1 and 4.5). The self-tuning regulator compensated for the change by decreasing the steam pressure, which best can be seen in Fig. 4.4.

At 22.28 the process operator increased the refiner energy. After that the couch vacuum variations were rather small until about 24.00. Then some disturbances influenced the process, and the variations in the moisture content were increased.

The self-tuning regulator was able to change the steam pressure fast enough to maintain good control. The estimated standard deviation of the moisture content in Fig.

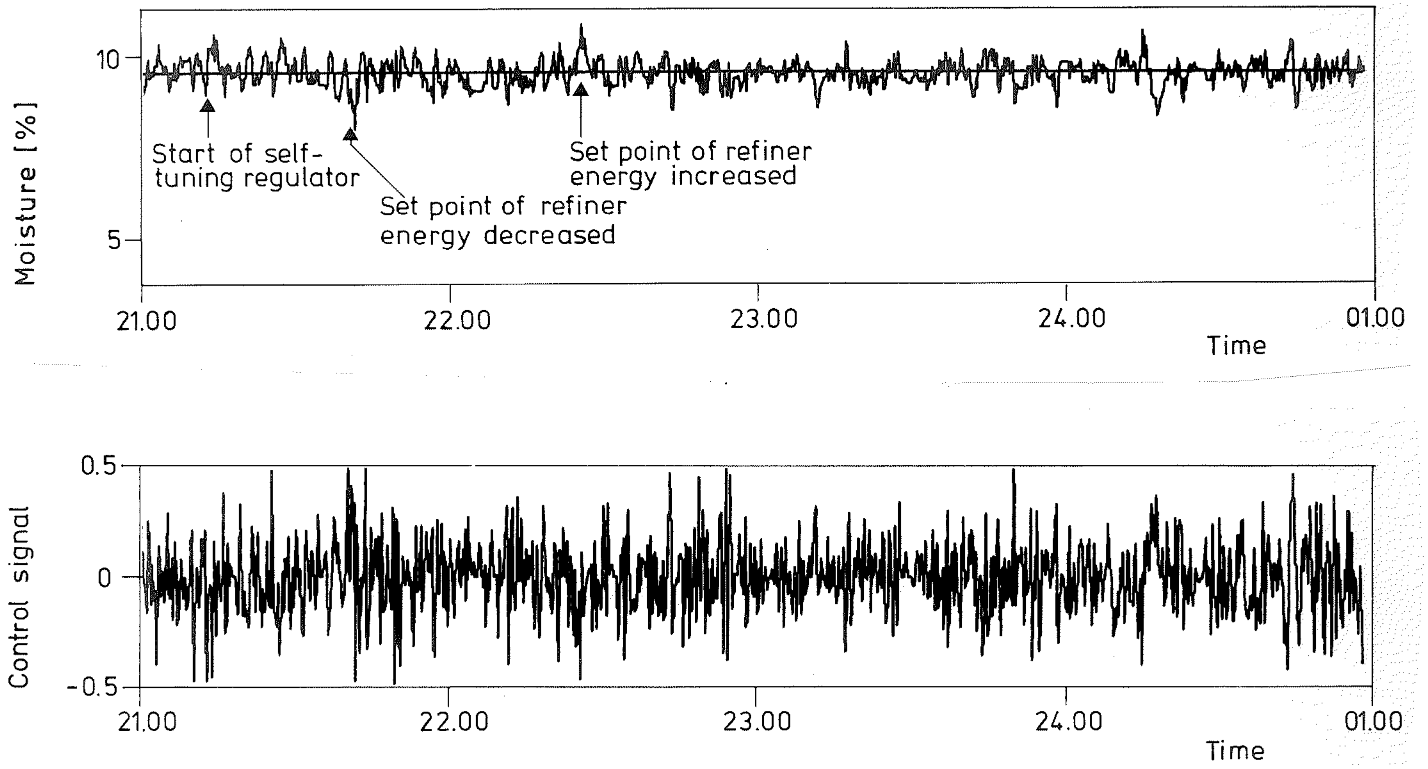


Fig. 4.1 - 4.2 - Moisture content signal and control signal (u) in Example 1. The regulator contains 6 parameters.

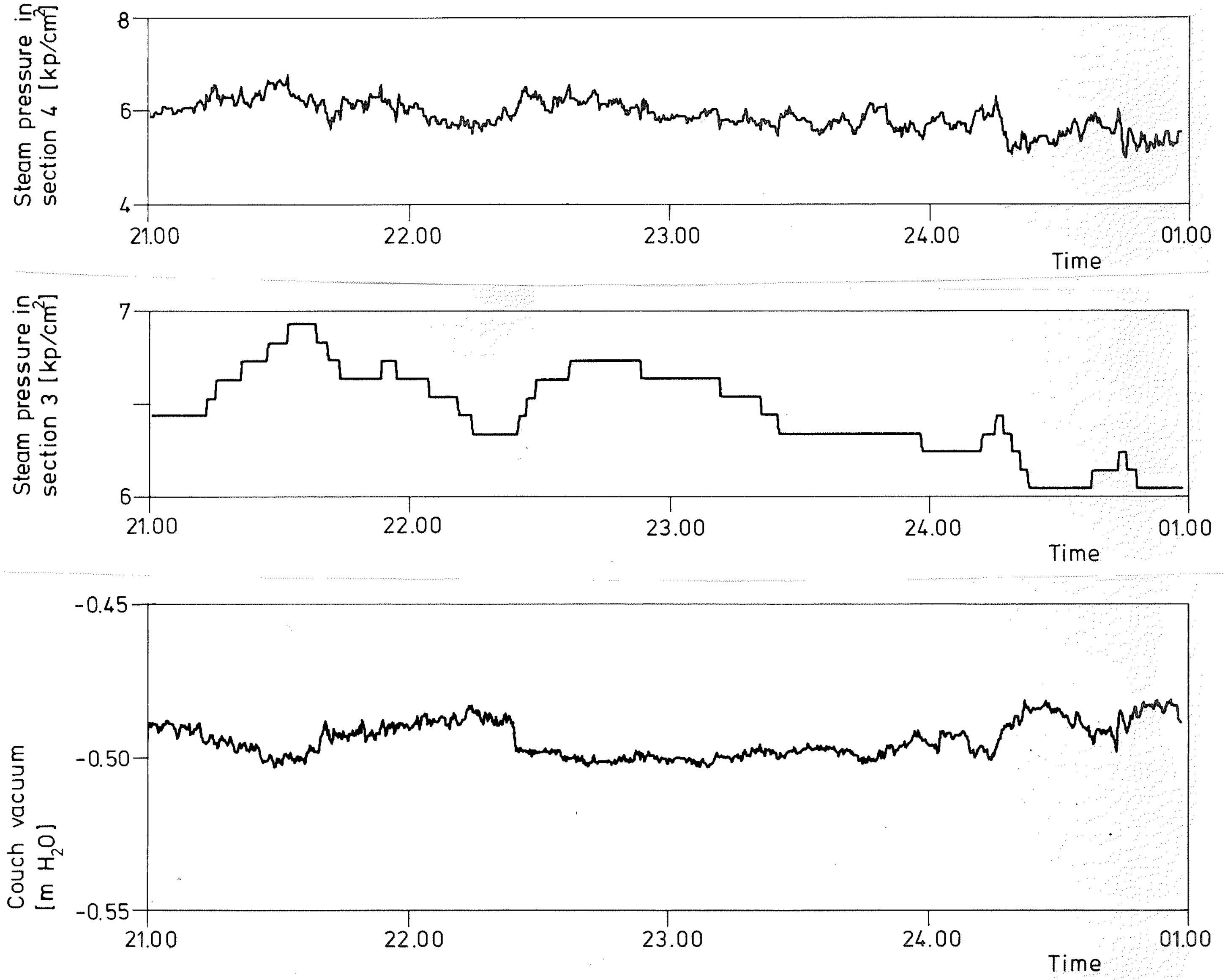


Fig. 4.3 - 4.5 - Steam pressure in the last drying section, reference value of steam pressure in the third drying section and couch vacuum from the experiment in Example 1.

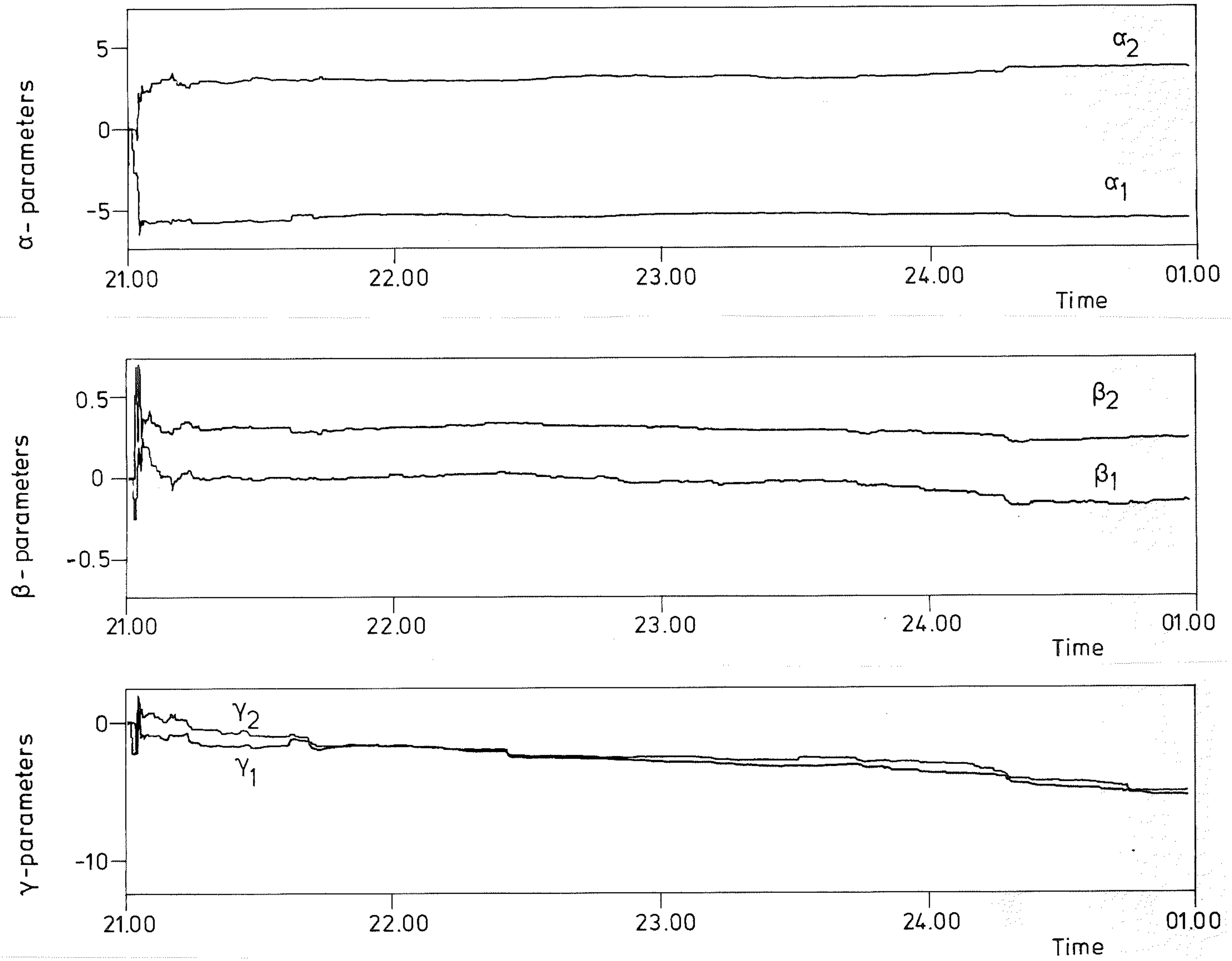


Fig. 4.6 - 4.8 - Parameter estimates from the experiment in Example 1.

4.1 is about 0.37%. This result is quite good when using a moisture gauge kept in a fixed position and without averaging over the machine width. In the Figures 4.6 - - 4.8 the regulator parameters are shown.

When the parameters have converged, and if the structure of the regulator is the same as of the minimum variance regulator of the system, then [2]

$$r_y(\tau) = 0 \qquad \tau \geq k+1$$
$$r_{yu}(\tau) = 0 \qquad \tau \geq k+1$$

where k is the number of time-delays. In Fig. 4.9 the estimated covariances $\hat{r}_y(\tau)$ and $\hat{r}_{yu}(\tau)$ have been plotted. The dashed lines indicate a 95% confidence interval for $\tau \neq 0$.

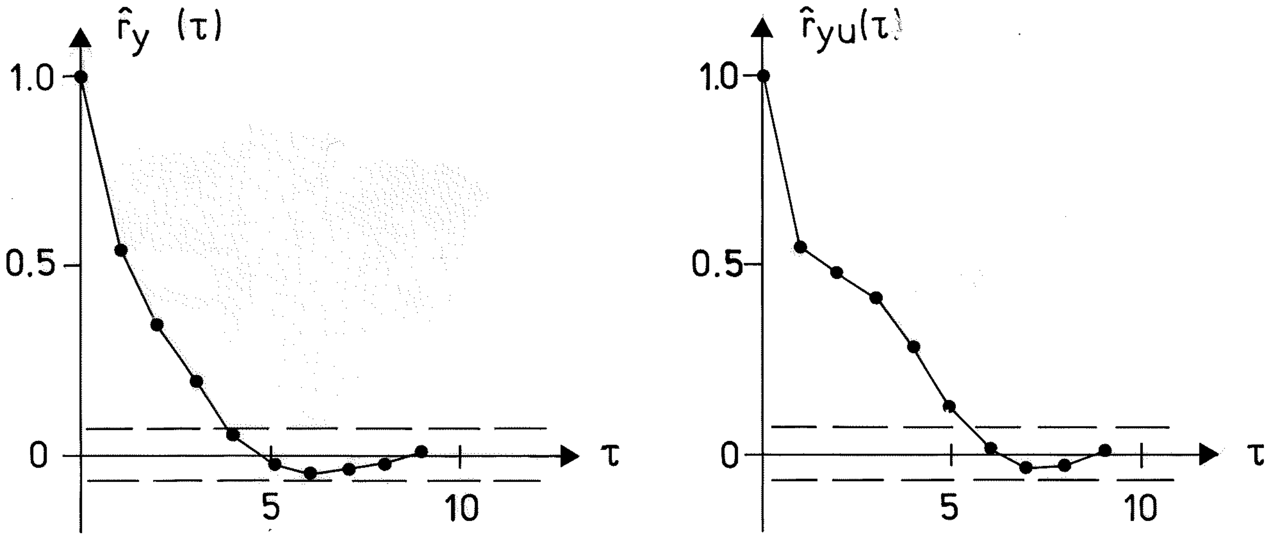


Fig. 4.9 - The estimated covariances of the data in Fig. 4.1 and 4.2. The covariances have been normalized.

In this example the covariances are expected to be zero for $\tau \geq 5$. In Fig. 4.9 there is only one point, $\hat{r}_{yu}(5)$, that is outside the 95% confidence interval.

Example 2. In this case the regulator contained nine parameters. The regulator had the structure

$$\begin{aligned}\nabla u(t) = {} & \alpha_1 y(t) + \alpha_2 y(t-1) + \alpha_3 y(t-2) - \\ & - \beta_1 \nabla u(t-1) - \beta_2 \nabla u(t-2) - \beta_3 \nabla u(t-3) - \beta_4 \nabla u(t-4) + \\ & + \gamma_1 \nabla v(t) + \gamma_2 \nabla v(t-1)\end{aligned}$$

The parameter k was given the value 3.

In Fig. 4.10 there is a registration of the moisture content for about 20 hours. The set point was 8.5% until 08.30. Then it was increased to 8.7%. The self-tuning regulator managed to keep the moisture content near the set point all the time. The estimated standard deviation of the first four hours is 0.39%. Later in the experiment it was even smaller.

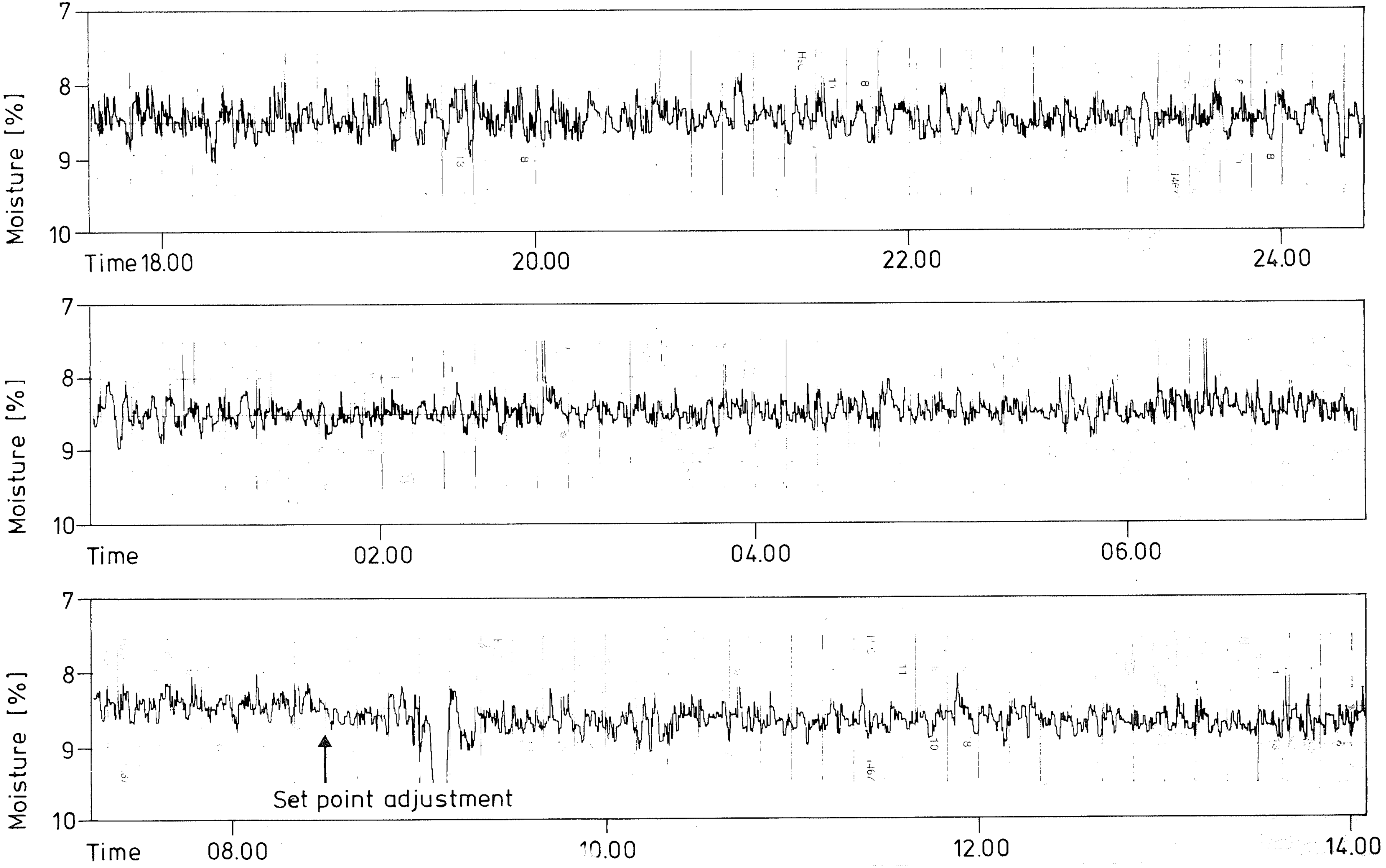


Fig. 4.10 - Moisture content from the experiment in Example 2. The length of the experiment is about 20 hours.

5. EXPERIENCES.

The self-tuning regulator has been used in experiments of different lengths to control the moisture content of paper. The experiences are very good both concerning the start-up phase and the steady state control.

In the experiments on the paper machine it turned out that the choice of parameters in the regulator was not crucial. The number of parameters in the regulator could be chosen in different ways without influencing the steady state performance drastically.

When the disturbances acting on the process were small, the experimental results were in accordance with the theoretical results. See for instance the estimated covariances in Figure 4.9. At paper breaks the self-tuning routine was stopped and a special program ~~started to dec~~ decrease the steam pressure of the drying sections. When the process was started up again the same program increased the steam pressure. Then the self-tuning regulator took over the control using the same parameters as before the paper break. Generally the parameters were not influenced very much by the paper breaks.

In most experiments a start-up procedure with an identification phase was used as described in Section 2. In the registrations of the parameters it can be seen that the transient period of the parameters has been about 10 minutes, which corresponds to about 40 sampling intervals. There may be a slow trend after the first transient before the stationary values are reached. Since it is desired that the stationary values should be reached as quickly as possible it can be convenient to use an exponential weighting factor, λ, which is time-varying. The weighting factor also makes it possible to follow slowly varying parameters. When the value of λ is determined there is a conflict between the demands on the ability of parameter tracking and the quality of steady state control. This is the dilemma that always exists when designing adaptive controllers.

From the registrations of the moisture content and the couch vacuum it can be seen that the disturbances do not always seem to be stationary. The disturbances can be small for a long period of time but suddenly large disturbances may occur in the process. However, the self-tuning regulator does not seem to have any difficulty to handle this type of disturbances in practice.

The self-tuning regulator had the same structure as the moisture controller used on the paper machine, i.e. a feedback loop from the moisture gauge and a feedforward loop from the couch vacuum gauge. The ordinary controller sometimes had difficulties to compensate for drift in the couch vacuum which originated mainly from variations in pulp quality and the degree of refining. Using the self-tuning regulator it was possible to follow the reference value better. The balance between the feedback and feedforward loops was improved by the self-tuning regulator. The performance of the ordinary controller was then improved by adjusting the parameters in accordance with the estimates obtained from the self-tuning regulator.

6. REFERENCES

[1] Alsholm O, Hempel A, Morsk B, Sangregorio G: Process and Production Control Package, 2nd IFAC Conference on Instrumentation and Automation in the Paper, Rubber and Plastics Industries, Brussels, May, 1971.

[2] Åström K J, Wittenmark B: On Self-Tuning Regulators, Automatica, Vol. 9, No. 2, 185 - 199, (1973).

[3] Peterka V: Adaptive Digital Regulation of Noisy Systems, IFAC Symposium on Identification and Process Parameter Estimation, Prague, 1970, Paper No. 6.2.

[4] Saridis G N, Dao T K: A Learning Approach to the Parameter-Adaptive Self-Organizing Control Problem, Automatica, Vol. 8, pp 589 - 597 (1972).

[5] Tse E, Athans M: Adaptive Stochastic Control for a Class of Linear Systems, IEEE T-AC, Vol. 17, No. 1, Feb., 1972, pp 38 - 51.

[6] Wittenmark B: A Self-Tuning Regulator, Report 7311, Division of Automatic Control, Lund Institute of Technology.