# Self-Tuning Controllers Based on Pole Placement Design

Westerberg, Bo; Wittenmark, Björn; Åström, Karl Johan

1978

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

Link to publication

*Citation for published version (APA):*
Westerberg, B., Wittenmark, B., & Åström, K. J. (1978). *Self-Tuning Controllers Based on Pole Placement Design*. (Research Reports TFRT-3148). Department of Automatic Control, Lund Institute of Technology (LTH).

*Total number of authors:*
3



LUND UNIVERSITY

PO Box 117
221 00 Lund
+46 46-222 00 00

CODEN: LUTFD2/(TFRT-3148)/1-052/(1978)

# SELF-TUNING CONTROLLERS BASED ON POLE-PLACEMENT DESIGN

K.J ÅSTRÖM
B. WESTERBERG
B. WITTENMARK

Department of Automatic Control
Lund Institute of Technology

May 1978

Dokumentutgivare
Lund Institute of Technology
Handläggare Dept of Automatic Control
K J Åström
Författare
K J Åström
B Westerberg
B Wittenmark

Dokumentnamn
REPORT

Dokumentbeteckning
LUTFD2/(TFRT-3148)/1-52/(1978)

Utgivningsdatum
May 1978

Ärendebeteckning

Dokumenttitel och undertitel

Self-tuning Controllers Based on Pole-Placement Design

Referat (sammandrag)

Self-tuning regulators based on deterministic pole-placement design, and recursive parameter estimation are discussed in this paper. The differences between algorithms based on identification of explicit and implicit process models are discussed. The properties of the algorithms are illustrated using simulation.

Referat skrivet av
Authors

Förslag till ytterligare nyckelord
Adaptive control. Self-tuning controller. Pole-placement. Implicit and explicit algorithms.

Klassifikationssystem och -klass(er)

Indextermer (ange källa)

Omfång
52 pages

Övriga bibliografiska uppgifter

Språk
English

Sekretessuppgifter

ISSN

ISBN

Dokumentet kan erhållas från
Department of Automatic Control
Lund Institute of Technology
Box 725, S-220 07 LUND 7, Sweden

Mottagarens uppgifter

Pris

DOKUMENTDATABLAD enligt SIS 62 10 12

SIS-DB 1

Blankett LU 11:25 1976–07

TABLE OF CONTENTS

# 1. INTRODUCTION

It is in many cases advantageous to have regulators which are more complicated than ordinary PID-regulators. It is unfortunately a considerable problem to tune more complicated regulators because many parameters have to be adjusted. One reason to design self-tuning regulators is to overcome the tuning problem.

The basic self-tuning regulator described in Åström and Wittenmark (1973) was designed for a situation where the control problem could be characterized as a minimum variance control problem. This means that the criterion can be described as to minimize the variance of the output. The basic self-tuning regulator was designed based on a certainty-equivalence argument. The appropriate model of the process and its environment is thus estimated recursively. The control is determined as if the estimated model is equal to the true model. There are many problems which fit this problem formulation and the basic self-tuning regulator has also been shown to work very well in such cases. There are, however, also stochastic control problems where minimum variance control is not appropriate. One case is a non-minimum phase plant. Another case is when large control signals are required to achieve minimum variance. These cases can, however, be formulated as linear-quadratic-gaussian (LQG) control problems. A self-tuning regulator based on the LQG design technique was described in Åström and Wittenmark (1974). Other versions are given in Åström et al (1977). The self-tuning regulator based on the LQG formulation has the drawback of being more complicated than the basic self-tuning regulator. The reason for this is that the design calculations which are done in each step involve the solution of a steady-state Riccati equation or equivalently a spectral factorization. A simpler algorithm was proposed by Clarke and Gawthrop (1975). They proposed to use a LQG formulation with a *one-step* criterion only. This simplifies the algorithm considerably. The algorithm can be made to work well in many cases but it is not foolproof. Further discussions of the algorithm are given in Gawthrop (1977).

There are many problems which do not fit the stochastic control formulation. Encouraged by the success of the self-tuning regulators for stochastic control problems, it is tempting to try a similar approach in other cases. Using the certainty equivalence argument the design is straightforward. Start with a design method for systems with known parameters. Substitute the parameters of the known system model by estimates which are obtained recursively and recalculate the control parameters in each step. Self-tuning controllers of this type which are based on pole-placement design and least-squares estimation are discussed in this paper. The controllers obtained are useful in many situations. For instance they can be used to tune control loops when the parameters or the controlled system is unknown or slowly time--varying. It is assumed that the main source of disturbances are changes in the reference value or occasionally large disturbances that have to be eliminated. The self-tuning regulator based on minimum variance control is not well suited for this case. The new self-tuning controllers can be used to solve the servo problem and can thus be regarded as useful complements.

Self-tuning regulators based on pole-placement design have been discussed by several other authors. A digital adaptive pole shifting algorithm was discussed in a dissertation by Edmunds (1976). This algorithm is further discussed in Wellstead (1978) and Wellstead et al (1978). In these works the emphasis is, however, on the regulation problem and not on the servo problem. The use of feed-forward is not discussed. Wouters (1977) also proposes a stochastic pole-placement strategy. Wouters also focuses on the stochastic regulation properties of the algorithm. The self-tuning controller proposed by Clarke can also be discussed in a pole-placement framework. Our paper differs from the previous treatments by focusing entirely on the servo problem. In our formulation the links to a deterministic design procedure are also emphasized. This makes it possible to establish links to MRAS. See Egardt (1978). Another feature of this paper is that the notions of algorithms with implicit and explicit identification are introduced. Several of the algorithms proposed in this paper are also new.

The paper is organized as follows. Pole-placement design for systems with known parameters is reviewed in Section 2. The suitability of the pole-placement design as a basis for adaptive control is discussed in Section 3. It is shown that there are some difficulties which are inherent in the problem formulation. Adaptive pole-placement algorithms based on estimation of the parameters in an explicit process model are discussed in Section 4. This leads to the so called *explicit* schemes. In Section 5 it is shown that some simplification of the adaptive algorithms can be achieved by instead estimating parameters in a modified process model. This leads to the *implicit* schemes. Some simulations which illustrate the behaviour of adaptive algorithms based on the pole-placement design are given in Section 6.

## 2. POLE-PLACEMENT DESIGN

Since the self-tuning regulators are based on a deterministic design procedure for systems with known parameters a brief review of a pole-placement design procedure is first given.

The material is not new. Part of it is known from classic texts on sampled data systems like Ragazzini and Franklin (1958). More recent discussions on design of digital control systems based on pole-placement design are found in Andersson (1977), Wittenmark (1976), and Franklin (1977). Due to the algebraic nature of the problem there are strong similarities to the corresponding design procedure for continuous systems. See Åström (1976). The discussion given here is limited to single input systems. A treatment of the multivariable systems is given by Wolowich (1974), Kucera (1975), Bengtsson (1977), and Pernebo (1978).

### PROBLEM FORMULATION

Consider a process characterized by the rational transfer function

$$G_p = \frac{B}{A} \tag{2.1}$$

where A and B are polynomials in the forward shift operator. It is assumed that the polynomials A and B are relatively prime and that the transfer function $G_p$ is causal. It is desired to find a regulator such that the transfer function from the reference value $y_r$ to the process output y is given by the rational transfer function

$$G_M = \frac{Q}{P} \tag{2.2}$$

where Q and P are given polynomials. It is assumed that Q and P are relatively prime and that $G_M$ is causal.

DESIGN PROCEDURE

A general linear regulator which generates the control signal u from the command signal $y_r$ and the process output y can be represented as

$$Ru = Ty_r - Sy \tag{2.3}$$

where R, S, and T are polynomials in the forward shift operator. The regulator can be realized as a dynamical system of order deg R whose characteristic polynomial is R. A block diagram of the closed loop system is shown in Figure 2.1.

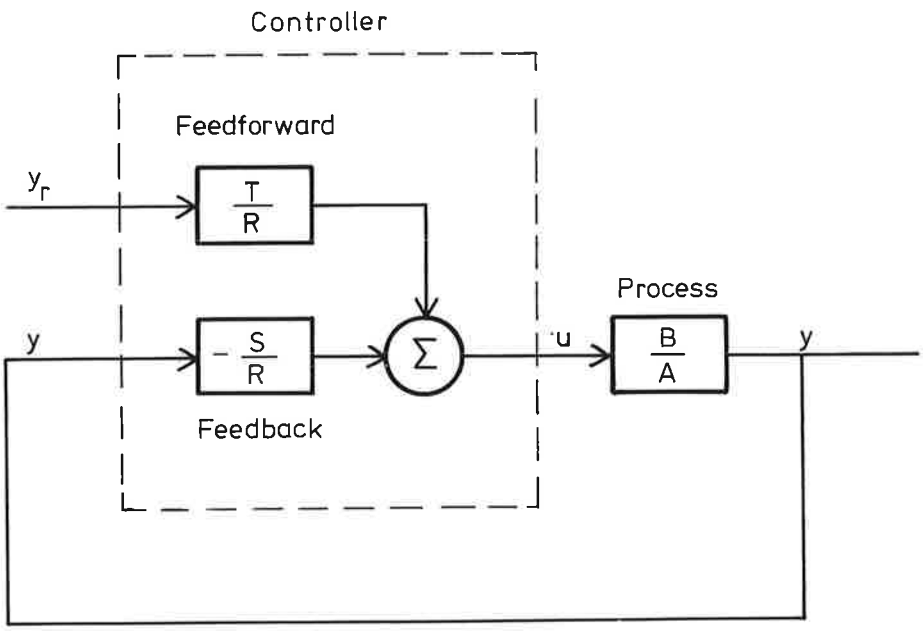


*Figure 2.1.* Block diagram of closed loop system.

The transfer function from the command signal $y_r$ to the output of the closed loop system is given by

$$G = \frac{TB}{AR + BS}.$$

The design problem is thus to find the polynomials R, S, and T such that the closed loop transfer function G is equal to the desired transfer function $G_M$. Hence

$$\frac{TB}{AR + BS} = \frac{Q}{P}. \tag{2.4}$$

Algebraically the design problem is thus to find three polynomials R, S, and T such that (2.4) holds. This algebraic problem does not have a unique solution. The lack of uniqueness can be exploited by the designer. To see how this can be done it is useful to have system-theoretic interpretations.

The degree of the polynomial $AR+BS$ is normally higher than the degree of the polynomial P. This means that the polynomials $AR+BS$ and TB have common factors. By a direct comparison with state space design composed of state feedback and an observer, it can be shown that the common factors of T and $AR+BS$ correspond to the observer poles. See e.g. Åström (1976). The polynomial T can thus be factored into two parts, one part corresponds to the desired observer poles and the other corresponds to the desired closed loop zeros which are not open loop zeros. Even if the zeros of T are specified there remains a constant factor which can be assigned arbitrarily. The polynomials R, S, and T can be multiplied by an arbitrary constant and the control law (2.3) and the condition (2.4) will still remain the same. In this paper this arbitrariness is eliminated by choosing a normalization of the polynomial T. Having obtained this insight, it is now easy to obtain a design procedure.

*DESIGN PROCEDURE 2.1 (General pole-placement)*

Data: Given polynomials A and B which characterize the process dynamics and polynomials P and Q which describe the desired closed loop transfer function.

Step 1: Find greatest common divisor $Q_2$ of B and Q. Factor Q and B as

$$Q = Q_1 Q_2$$
$$B = B_1 Q_2. \tag{2.5}$$

Step 2: Determine desired observer poles, specified by the polynomial $T_1$. Choose

$$T = T_1 Q_1. \tag{2.6}$$

Step 3: Solve the equation

$$AR + BS = PB_1T_1 \tag{2.7}$$

with respect to R and S. □

There is a solution to the pole-placement problem in the sense that a causal stabilizing regulator exists if and only if $B_1$ and $P_1$ are stable, and

$$\deg A - \deg B_1 \leqslant \deg P - \deg Q_1$$

If these conditions are satisfied and if deg B < deg A then the design procedure 2.1 gives a stable causal regulator if the observer polynomial $T_1$ is chosen stable with $\deg T_1 \geqslant \deg A - 1$ and if S has lower degree than A. Since A and B are relatively prime there is a unique solution to (2.7) with deg S < deg A. The choice $\deg T_1 = \deg A - 1$ corresponds to a Luenberg observer and the choice $\deg T_1 = \deg A$ corresponds to a Kalman observer. In some cases it may be desirable to choose observers of higher order.

It is straightforward to verify that the design procedure gives the desired results. Insertion of (2.5), (2.6), and (2.7) gives

$$G = \frac{TB}{AR+BS} = \frac{T_1Q_1B_1Q_2}{PT_1B_1} = \frac{Q}{P} = G_M.$$

This shows that the closed loop transfer function is equal to the specified transfer function.

## SPECIAL CASES

The calculations required for the design procedure 2.1 are simple enough for off-line calculations. It is sufficient to have procedures for finding greatest common divisors and for solving the equation (2.7). Such procedures can be obtained based on Euclid's algorithm. In the adaptive algorithms the design calculations have to be repeated in each iteration step. The computational burden may then be considerable and it is therefore of interest to see if there are special cases where the design calculations can be reduced. Some examples are given below.

*EXAMPLE 2.1 (All process zeros cancelled and $Q = 1$)*

Assume that the specifications are such that $Q = 1$. The specifications are normally such that the low frequency gain of the desired closed loop is unity. When $Q = 1$ the polynomial $P$ must then be normalized in such a way that $P(1) = 1$. It is assumed that this normalization is done. Equation (2.4) gives

$$AR + BS = PTB \tag{2.8}$$

or

$$AR = (PT - S)\,B.$$

Since $B$ does not divide $A$ it must thus divide $R$. Hence

$$R = BR_1.$$

Equation (2.8) can then be written as

$$AR_1 + S = PT. \tag{2.9}$$

It often happens that $A(z) = z^k A_1(z)$. The equation (2.9) is then particularly easy to solve. See Åström (1970). □

In this particular case the design procedure is thus to choose $P$ and $T$ and to solve (2.9) for $R_1$ and $S$. The regulator is then given by (2.3) with $R = BR_1$. A block diagram of the closed loop system is shown in Figure 2.2.

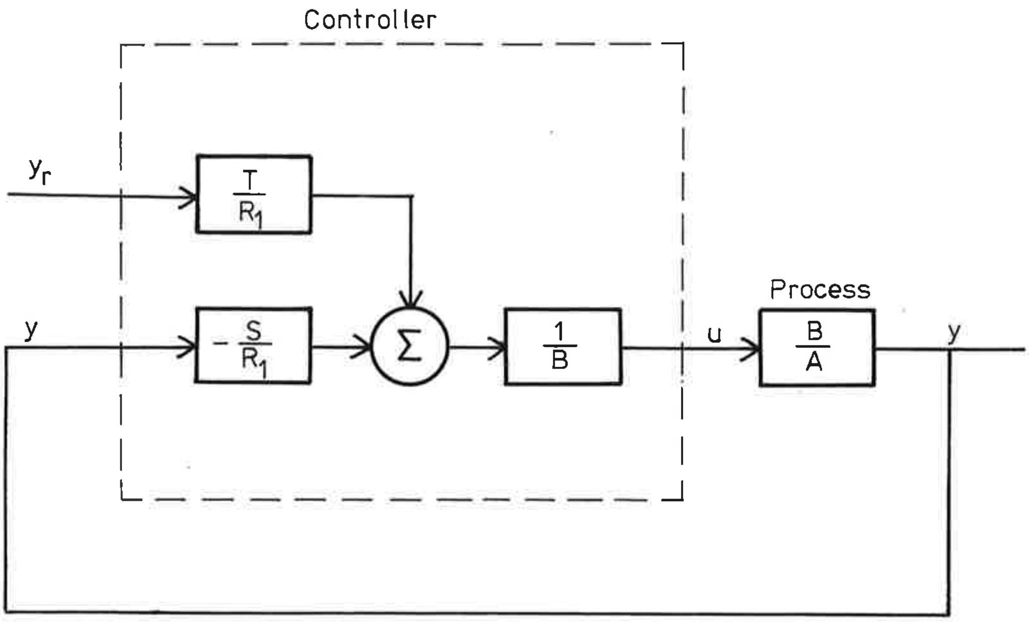


*Figure 2.2.* Block diagram of closed loop system obtained when all process zeros are cancelled. The regulator is implemented as a realization of $BR_1 u = Ty_r - Sy$.

It is clear from Fig. 2.2 that all process zeros are cancelled in this particular design. It then follows that the design procedure will give satisfactory results only when the cancelled polynomial is stable. The procedure will thus not work unless the process is minimum phase. In practice it is not sufficient to require that the zeros are inside the unit circle but it must be required that the zeros are inside a critical area like the one shown in Fig. 2.3.

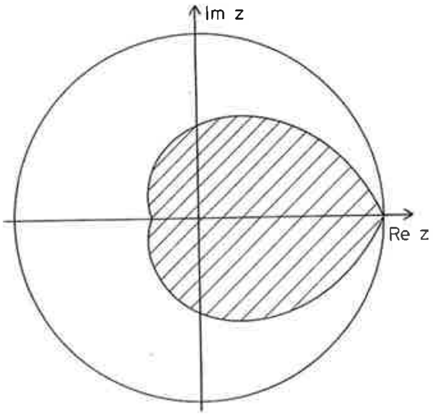


*Figure 2.3.* Critical area for process zeros.

□

The simplification obtained in the special case of Example 2.1 is that Step 1 in the general design procedure is avoided. Another special case is given below.

EXAMPLE 2.2 (*No process zeros are cancelled*)

Assume that the specifications are such that the desired closed loop zeros are equal to the process zeros, i.e. $Q = B$. The specifications are normally such that the low frequency gain of the desired closed loop is unity. Since $Q = B$ the constant factor of the polynomial $P$ is then chosen so that $P(1) = B(1)$. It is assumed that this normalization is made. Equation (2.4) then gives

$$AR + BS = PT. \tag{2.10}$$

The design procedure is thus again to choose the polynomials $P$ and $T$. Equation (2.10) is then solved with respect to $R$ and $S$ and the controller is then given by (2.3). □

The simplification obtained in the special case of Example 2.2 is that Step 1 in the general design procedure is simplified.

## 3. DISCUSSION ON THE ADAPTIVE POLE-PLACEMENT PROBLEM

Before presenting specific adaptive algorithms, it will be discussed whether the pole-placement design procedures are suitable for design of adaptive regulators. There are several problems to be considered.

First it may be too restrictive to specify all closed loop poles at least for high order systems. One possibility to avoid this difficulty for discrete time systems is to specify only the dominant poles and require that the remaining poles are close to the origin.

The pole-placement design procedure requires that the observer poles are specified. The observer poles are not critical. Their choice should, however, reflect the characteristics of the disturbances. If an estimation procedure which gives the disturbance dynamics is used e.g. in the form of a controlled ARMA model it is natural to choose the observer polynomial proportional to the polynomial which characterizes the moving average. In this paper this is not done and the observer polynomial thus has to be chosen arbitrarily.

Zeros of the process dynamics which can be cancelled present no problem. Zeros which are outside the unit circle or inside but close to the unit circle present problems. Since such zeros cannot be cancelled they must be included as zeros of the closed loop transfer function. This gives difficulties in the case of known parameters and even greater difficulties in self-tuning regulators. Even if the poles of the closed loop system are specified many properties of the closed loop system will depend critically on the closed loop zeros. Hence if the primary specifications are given in terms of properties like overshoot, bandwidth, static errors, etc. it is in general not possible to find closed loop poles that give the desired properties unless the closed loop zeros are known.

It follows from the discussion that self-tuning controllers based on pole-placement design can be reasonable for minimum phase systems and for non-minimum phase systems in those cases where the process zeros outside the unit disc only change moderately.

# 4. SELF-TUNING CONTROLLERS USING EXPLICIT IDENTIFICATION

The basic idea when using the separation principle to design self--tuning regulators can be expressed as follows. Start with a design procedure for systems with known parameters. When the parameters are not known they are estimated recursively and the regulator is redesigned in each step, using the estimated parameters. Since there is no recursive parameter estimator which is uniformly best there are many possibilities even if the design principle is fixed. There are also many details which can be done in different ways and consequently a large variety of possible adaptive algorithms. Some simple algorithms will be discussed in this section. Least squares, which is one of the simplest recursive estimation schemes will be used. This procedure will give biased estimates if there are stochastic disturbances which are coloured noise. Since the discussion is focused on the servo problem the major disturbances are, however, command inputs. This is also compatible with the pole-placement design procedure which is not suitable to handle trade-offs between measurement noise and process noise quantitatively.

Some self-tuning algorithms will now be described in detail. The parameter estimation is discussed first.

## LEAST SQUARES PARAMETER ESTIMATION

The parameters of the process model

$$Ay = Bu \tag{4.1}$$

are estimated by least squares. This is discussed in detail in Åström (1968). The model (4.1) can be explicitly written as

$$y(t)+a_1y(t-1)+\ldots+a_ny(t-n) = b_0u(t-k)+\ldots+b_mu(t-m-k). \tag{4.2}$$

Introduce a vector of parameter estimates

$$\theta = [\hat{a}_1 \; \ldots \; \hat{a}_n \;\; \hat{b}_0 \; \ldots \; \hat{b}_m]^T \tag{4.3}$$

and a vector of regressors

$$\varphi(t) = [-y(t-1) \ldots - y(t-n) \quad u(t-k) \ldots u(t-m-k)]^T \tag{4.4}$$

The recursive least squares estimate is then given by

$$\theta(t+1) = \theta(t) + P(t+1)\ \varphi(t+1)\ \varepsilon(t+1) \tag{4.5}$$

where

$$\varepsilon(t+1) = y(t+1) - \theta(t)^T \varphi(t+1) \tag{4.6}$$

and

$$P(t+1) = \left[P(t) - P(t)\varphi(t+1)[\sigma^2 + \varphi^T(t+1)P(t)\varphi(t+1)]^{-1}\varphi(t+1)^T P(t)\right] / \lambda. \tag{4.7}$$

There are also other possibilities to perform the least squares calculations. Square root algorithms are useful if the problem is poorly conditioned. See e.g. Peterka (1975). Fast algorithms can be used if computing time is critical. See e.g. Levinson (1947).

## CHOICE OF $\lambda$ AND MODIFICATIONS OF P-EQUATION

The factor $\lambda$ in equation (4.7) is introduced to discount past data when performing the least squares. For the regulation problem the estimator is excited by the process disturbances which normally are reasonably uniform in time. It has been found empirically that a value of $\lambda$ between 0.95 and 0.99 works well in such cases. For the servo problem the major excitation comes from the changes in the command signal. Such changes may be irregular and it has been found that there may be bursts in the process output if equation (4.7) is used with $\lambda$ less than one. The presence of bursts can be understood intuitively as follows. The negative term in (4.7) represents the reduction in parameter uncertainty due to the last measurement. When there are no changes in the set point the vector $P\varphi$ will be zero. There will not be any changes in the parameter estimate and the negative term in the right hand side of (4.7) will be zero. The equation (4.7) then reduces to

$$P(t+1) = \frac{1}{\lambda} P(t)$$

and the matrix P will thus grow exponentially if $\lambda < 1$. If there are no changes for a long time the matrix P may thus become very large. A change in the command signal may then lead to large changes in the parameter estimates and in the process output. The large values of the matrix P may also lead to numerical problems. Examples which illustrate this behaviour are found e.g. in Fortescue (1975) and Morris et al (1977).

There are many ways to eliminate bursts. Perturbation signals may be added to ensure that the process is properly excited. The estimation algorithm may be modified. One possibility is to stop the updating of the matrix $P(t)$ when the signal $P(t)\varphi(t)$ is smaller than a given value. Another possibility is to subtract a term like $\alpha P^2(t)$ from the right hand side of (4.7) to ensure that the matrix $P(t)$ stays bounded.

## SELF-TUNING POLE-PLACEMENT ALGORITHMS

An adaptive pole-placement algorithm can now be described as follows. The following steps are performed at each sampling period.

Step 1: Estimate the parameters of the model (4.1) by least squares.

Step 2: Apply the pole-placement procedure as described in Section 2, using the process model (4.1) with the estimated parameters obtained in Step 1.

An algorithm of this type is called an algorithm based on *estimation of process parameters* because the estimated parameters are the parameters of the process model in the standard form. In analogy with the terminology used for model reference adaptive systems (MRAS), the algorithm is also referred to as an algorithm using *explicit identification.*

Algorithms of this type require procedures for recursive parameter estimation and procedures for the pole-placement design calculations. The design calculations are normally more time-consuming than the parameter estimation and it is therefore of interest to look at the special cases of the design discussed in Examples 2.1 and 2.2, where the design calculations are simplified. A detailed description of the self-tuning algorithms obtained in these cases will now be given.

EXAMPLES

Assume that the process to be controlled has zeros well inside the unit disc, e.g. inside the critical area shown in Fig. 2.3. It is then reasonable to have a pole-placement design where all the process zeros are cancelled. Under this hypothesis the pole-placement procedure can be simplified as shown in Example 2.1. The corresponding self-tuning pole-placement algorithm then becomes

*ALGORITHM E1 (Explicit algorithm. All process zeros cancelled)*

Data: Given specifications in the form of the desired closed loop poles and desired observer poles specified by the polynomials P and T. It is assumed that P is normalized so that $P(1) = 1$. The polynomial T can be normalized arbitrarily. Compare with Example 2.1.

Step 1: Estimate the parameters of the model

$$Ay = Bu$$

by least squares.

Step 2: Determine the polynomials $R_1$ and S such that

$$AR_1 + S = PT.$$

Step 3: Use the control law

$$BR_1u = Ty_r - Sy.$$

The Steps 1, 2, and 3 are repeated for each sampling period. □

*REMARK*

Common factors of A and B should be eliminated after Step 1 to ensure that the equation in Step 2 has a solution.

This algorithm cannot be expected to work well unless the corresponding design procedure for systems with known parameters works well. Since all process zeros are cancelled the regulator will not be satisfactory for non-minimum phase systems. Such systems can, however, be handled using the design procedure in Example 2.2. The corresponding self-tuning control algorithm is given by

*ALGORITHM E2 (Explicit algorithm. No process zeros cancelled)*

Data: Given specifications in the form of the desired closed loop poles and the desired observer poles specified by the polynomials P and T, where P and T are normalized arbitrarily.

Step 1: Estimate the parameters of the model

$$Ay = Bu$$

by least squares.

Step 2: Normalize the polynomial P in such a way that $P(1) = B(1)$. Then determine the polynomials R and S such that

$$AR + BS = PT.$$

Step 3: Use the control law

$$Ru = Ty_r - Sy.$$

The Steps 1, 2, and 3 are repeated for each sampling period. □

*REMARK*

Possible common factors of A and B should be eliminated after the first step to ensure that the equation in Step 2 has a solution. Notice that the polynomial P cannot be normalized a priori because the normalization requires knowledge of the polynomial B in the process model.

Notice that with algorithm E2 the properties of the closed loop system will change even if P and T are fixed because the closed loop zeros will change if the process zeros change.

## THE GENERAL CASE

It is now straightforward to give a self-tuning control algorithm which corresponds to the general pole-placement design procedure 2.1.

*ALGORITHM E3 (General explicit algorithm)*

Data: Given the desired closed loop poles and zeros and the desired observer poles characterized by the polynomials P, Q, and $T_1$.

Step 1: Estimate the parameters of the process model

$$Ay = Bu$$

by least squares.

Step 2: Eliminate common factors of $A$ and $B$. Let A and B denote the polynomials obtained.

Step 3: Find the greatest common divisor $B_2$ of B and Q. Factor Q and B as

$$Q = Q_1 B_2$$

$$B = B_1 B_2.$$

Step 4: Solve the equation

$$AR + BS = PB_1 T_1$$

for R and S.

Step 5: Compute the control signal from

$$Ru = Ty_r - Sy,$$

where $T = T_1 Q_1$.

The Steps 1 through 5 are repeated at each sampling period. □

*REMARK*

To obtain a stable closed loop system it is necessary that the specifications are such that the unstable factors of $B_2$ are also factors of Q.

# 5. SELF-TUNING CONTROLLERS USING IMPLICIT IDENTIFICATION

The design calculations for the algorithms discussed in the previous section may be time-consuming. It is possible to obtain different algorithms where the design calculations are simplified considerably. The self-tuning regulator in Åström and Wittenmark (1973) is a prototype for algorithms of this type. The basic idea is to rewrite the process model in such a way that the design step is trivial. For minimum variance control the process model can be rewritten so that the parameters of the minimum variance regulator are the parameters of the rewritten model. By a proper choice of model structure the regulator parameters are thus updated directly and the design calculations are thus eliminated. Algorithms of this type are called algorithms based on *implicit identification* of a process model. In the simple cases the algorithms are also called algorithms based on *estimation of regulator parameters*. Some examples of such algorithms will now be given.

## EXAMPLES

First consider the problem discussed in Example 2.1. The process is thus described by

$$Ay = Bu \tag{5.1}$$

and it is desired to find a feedback such that the closed loop transfer function from the reference value to the output is $1/P$. It is assumed that the polynomial $P$ is normalized so that $P(1) = 1$. In the case of known parameters the design is based on the solution of the equation (2.9), i.e.

$$AR_1 + S = PT. \tag{5.2}$$

The equations (5.1) and (5.2) give

$$PTy = AR_1y + Sy = Sy + BR_1u = Sy + Bu$$

or

$$PTy = Sy + Ru. \tag{5.3}$$

The process can thus be represented either by (5.1) or by (5.3). The representation (5.3) has the advantage that the regulator polynomials R and S occur explicitly in the model. If the model (5.3) is available the pole-placement design is thus trivial because the regulator (2.3) is obtained from the model (5.3) by inspection. The following self--tuning control algorithm can now be obtained.

ALGORITHM II *(Implicit algorithm. All process zeros cancelled)*

Data: Given specifications in the form of the desired closed loop poles and the desired observer poles specified by the polynomials P and T. It is assumed that P is normalized in such a way that $P(1) = 1$. The polynomial T can be normalized arbitrarily.

Step 1: Estimate the parameters of the polynomials R and S in the model

$$PTy = Sy + Ru \tag{5.3}$$

by least squares.

Step 2: Use the control law

$$Ru = Ty_r - Sy. \tag{5.4}$$

The Steps 1 and 2 are repeated for each sampling period.

The least squares estimate is given by equations (4.5) and (4.7) where

$$\theta = [\hat{s}_0 \ldots \hat{s}_{n-1} \quad \hat{r}_0 \ldots \hat{r}_{m_1}]$$

$$\varphi(t) = [y(t-\ell+n-1) \ldots y(t-\ell) \quad u(t-\ell+m_1) \ldots u(t-\ell)]$$

$$\ell = \deg PT, \quad m_1 = \deg T, \quad n = \deg A$$

$$\varepsilon(t) = PTy(t-\ell) - \theta(t)^T\varphi(t+1).$$ □

Notice that the control law (5.4) is characterized by three polynomials R, S, and T. The polynomial T which represents the feedforward from the command signal is chosen arbitrarily. Its zeros correspond to the desired observer poles. The polynomials S and R whose quotient corresponds to the feedback transfer function are estimated directly. The algorithm I1 is thus clearly an algorithm where the regulator parameters are estimated directly.

A comparison with the corresponding explicit algorithm E1 shows that the implicit algorithm is a considerable simplification because the design calculations are avoided.

An implicit algorithm which corresponds to the problem in Example 2.2 will now be given. It is thus assumed that the process is governed by (5.1) and that it is desired to have a closed loop system with the transfer function B/P. For systems with known parameters the design is based on the equation (2.10), i.e.

$$AR + BS = PT.$$

Combining this equation with the process model (5.1) gives

$$PTy = ARy + BSy = BSy + BRu. \tag{5.5}$$

If the polynomials P and T are stable, the model (5.5) is equivalent to the model (5.1). The representation (5.5) has the advantage that the regulator polynomials R and S appear explicitly in the model (5.5). It is is thus easier to determine the control law from the model (5.5) than from (5.1). The self-tuning algorithm now becomes as follows.

*ALGORITHM I2 (Implicit algorithm. No process zeros cancelled)*

Data: Given specifications in the form of the closed loop poles and the observer poles specified by the polynomials P and $T$.

Step 1: Estimate the parameters of the model

$$PTy = \mathcal{S}y + \mathcal{R}u$$

by least squares.

Step 2: Find the largest common factor B of $\mathcal{R}$ and $\mathcal{S}$ and factors $\mathcal{R}$ and $\mathcal{S}$ as

$$\mathcal{R} = BR$$
$$\mathcal{S} = BS.$$

Step 3: Normalize $\mathcal{T}$ by multiplying it by the factor $P(1)/B(1)$ i.e.

$$T = \mathcal{T}\,P(1)\,/\,B(1).$$

The control law is then

$$Ru = Ty_r - Sy.$$

*REMARK*

The normalization of the polynomial P cannot be made a priori because the normalization requires knowledge of the polynomial B in the process model.

A comparison with the corresponding explicit algorithm E2 shows that less calculations are required if the implicit algorithm is used. Notice however that some calculations (elimination of common factors) are required to obtain the controller parameters from the estimated parameters. This calculation may be badly conditioned.

## THE GENERAL CASE

The general implicit algorithm is obtained from the general pole-placement design procedure 2.1. Consider the equation (2.7). It follows that $B_1$ divides R. Introduce

$$R = B_1R_1.$$

Division of (2.7) by $B_1$ gives

$$PT_1 = AR_1 + B_2S$$

where $B = B_1B_2$. Hence

$$PT_1y = AR_1y + B_2Sy. \qquad (5.6)$$

It follows from the process model (2.1) that

$$AR_1y = R_1Bu = B_2R_1B_1u = B_2Ru.$$

Equation (5.6) can thus be written as

$$PT_1y = B_2Sy + B_2Ru = \mathcal{S}y + \mathcal{R}u. \qquad (5.7)$$

This is the transformed system model which will be used to obtain the implicit algorithm. Notice that the factor $B_2$ which corresponds to the unstable zeros divides $\mathcal{R}$ and $\mathcal{S}$. The general implicit algorithm is now obtained as follows.

*ALGORITHM I3 (General implicit algorithm)*

Data: Given the desired closed loop transfer function $G_M = Q/P$ and the desired observer poles specified by the polynomial $T_1$.

Step 1: Estimate the parameters of the polynomial $\mathcal{R}$ and $\mathcal{S}$ in the model

$$PT_1y = \mathcal{R}y + \mathcal{S}u \tag{5.8}$$

by least squares.

Step 2: Find the largest common factor $B_2$ of $\mathcal{R}$ and $\mathcal{S}$ and factor $\mathcal{R}$ and $\mathcal{S}$ as

$$\mathcal{R} = B_2R$$
$$\mathcal{S} = B_2S.$$

Step 3: Factor Q as

$$Q = Q_1B_2.$$

Introduce

$$T = Q_1T_1.$$

The control law is then

$$Ru = Ty_r - Sy.$$

The Steps 1, 2, and 3 are repeated at each sampling instant. □

*REMARK*

The design procedure requires that all unstable and poorly damped process zeros are also zeros of Q. If an unstable factor of $B_2$ is found which is not a factor of Q then the specified Q must be modified to include this factor.

# 6. SIMULATIONS

Some of the properties of the algorithms are illustrated by simulation in this section. The simulations are made using the simulation program SIMNON, see Elmqvist (1975). The special SIMNON system, written in FORTRAN, for simulating a general adaptive controller described in Gustavsson (1978) was used. The parameters of this program and the special systems used are listed in the Appendix. Other simulations are found in Westerberg (1977).

## CHOICE OF PARAMETERS

There are several parameters which have to be selected in the algorithms. Unless otherwise stated the following parameters have been chosen. Initial values of all parameters are chosen as zero except for $r_0 = 2$ in the implicit algorithms and $b_m = 2$ in the explicit algorithms. The initial value of the covariance matrix is chosen as ten times the unit matrix.

## EXAMPLE 6.1 - Simple First Order System

The properties of the algorithms when controlling the system

$$y(t) - 0.9\, y(t-1) = u(t-1) + e(t) + ce(t-1), \tag{6.1}$$

where $\{e(t)\}$ is a sequence of independent normal $(0,\sigma)$ random variables, will first be explored. It is assumed that the desired behaviour of the closed loop system is characterized by the transfer function

$$G_M = \frac{Q}{P} = \frac{1}{z - 0.7} .$$

It is assumed that a first order observer with a pole in 0.5 is used. Hence

$$T = z - 0.5.$$

The process pole at $z = 0.9$ should thus be moved to $z = 0.7$. Since the process has no zeros the problems associated with cancellation of zeros do not occur. In the case of known parameters it follows from the analysis in Section 2 that the controller

$$u(t) = \frac{T}{R} y_r - \frac{S}{R} y$$

will satisfy the requirements. Solution of the equation (5.3) gives

$$R = z$$

$$S = -0.3\,z + 0.35.$$

The behaviour of the closed loop system for the implicit algorithm has been explored in three different cases. The forgetting factor $\lambda$ was 0.99 in all cases.

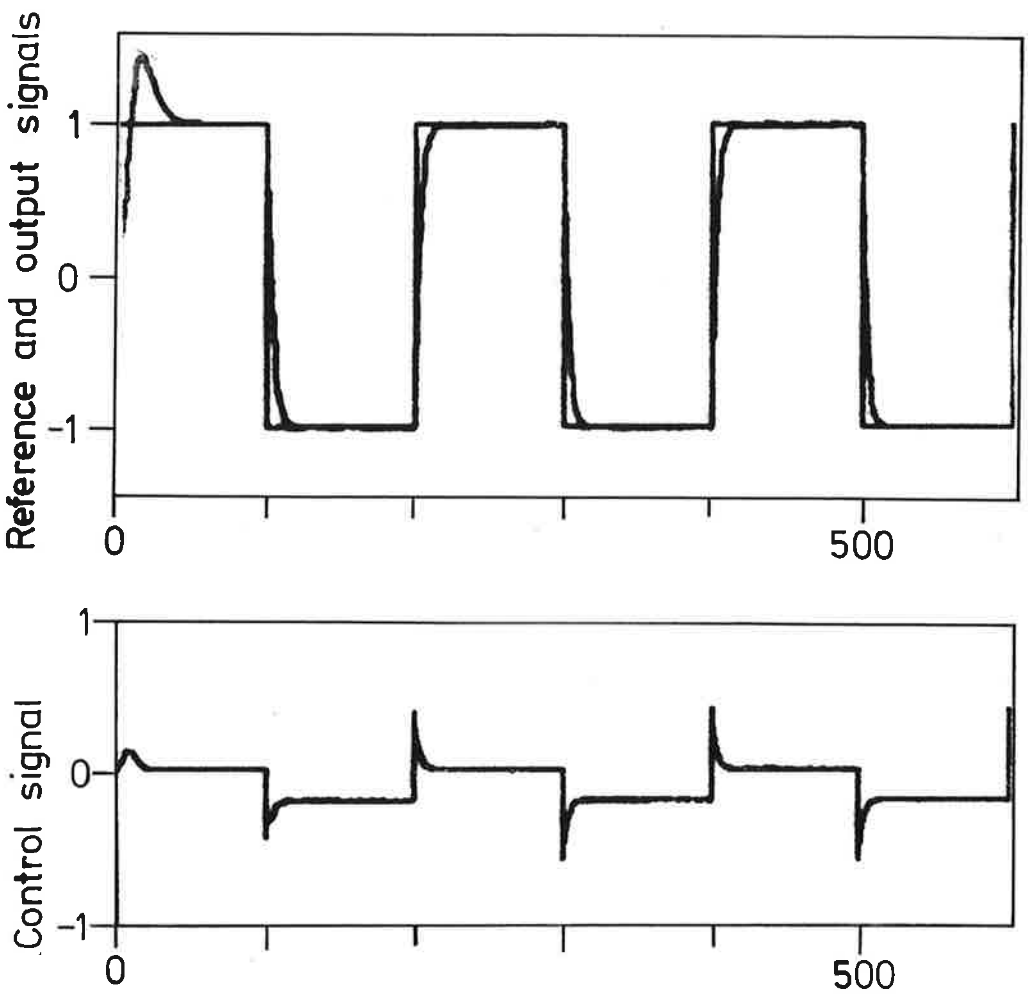


*Figure 6.1.* Output y, command $y_r$, and control signal u for the system (6.1) with $\sigma = 0$ controlled with implicit algorithm I1.

*CASE A (No random disturbances $\sigma = 0$)*

The behaviour of the closed loop system when following a square wave command is shown in Fig. 6.1. The parameter estimates are shown in Fig. 6.2. Notice that the parameter estimates converge quickly and that the closed loop response is very close to the desired response already at the second transient. Also notice that the major changes of the estimates occur at the step-changes in the command input.

The diagonal elements of the matrix P in the estimation algorithm are shown in Fig. 6.3. Notice the exponential increase of the matrix P(t) between the step-changes. This growth which depends on the forgetting factor may lead to bursts if the step-changes are far apart as was discussed in Section 4.

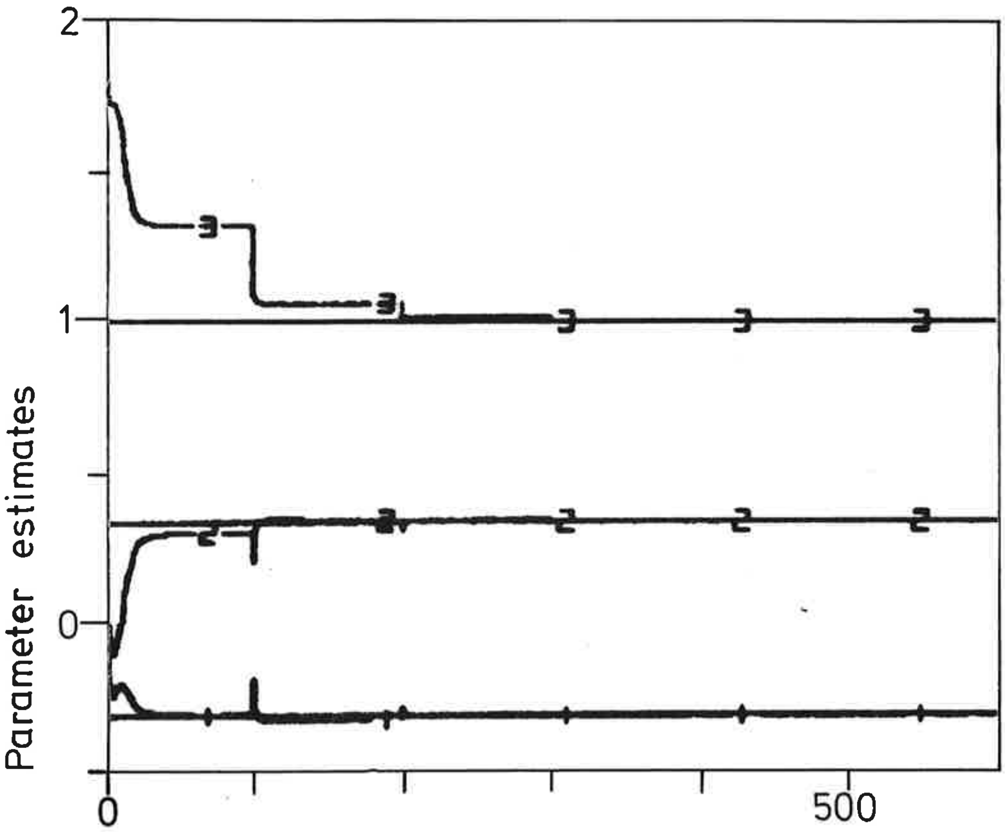


*Figure 6.2.* Estimated parameters for system (6.1) with $\sigma = 0$ controlled by the implicit algorithm I1.

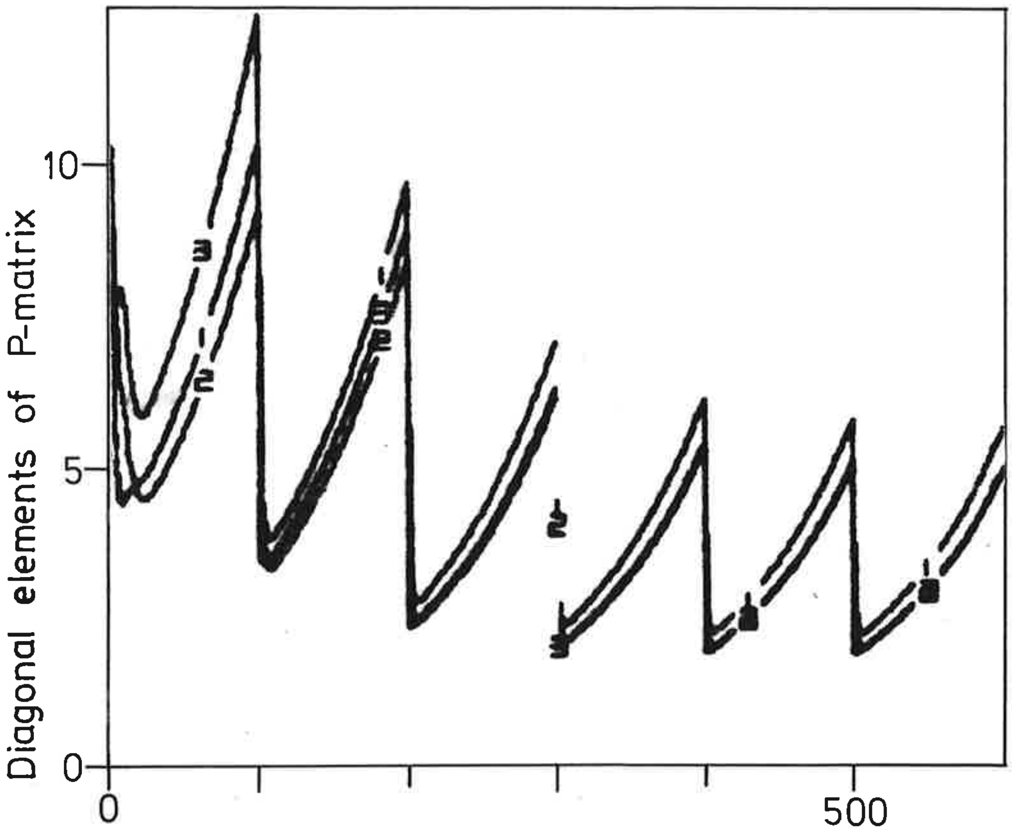


*Figure 6.3.* Diagonal elements of the matrix P of the estimation algorithm for the system (6.1) with $\sigma = 0$ controlled by the implicit algorithm I1.

*CASE B (White process noise $c = 0$, $\sigma = 0.1$)*

This case illustrates the behaviour of the system in the presence of moderate disturbances. The amplitude of the step command is one unit and the standard deviation of the disturbance is 0.1. The output and the control signals are shown in Fig. 6.4 and the parameter estimates are shown in Fig. 6.5. The results show that the algorithm is not particularly sensitive to a white noise disturbance. The diagonal elements of the matrix P of the parameter estimator are shown in Fig. 6.6. A comparison with the corresponding Fig. 6.3 for the noisefree case shows that the presence of the noise will limit the growth of some elements of the matrix P.

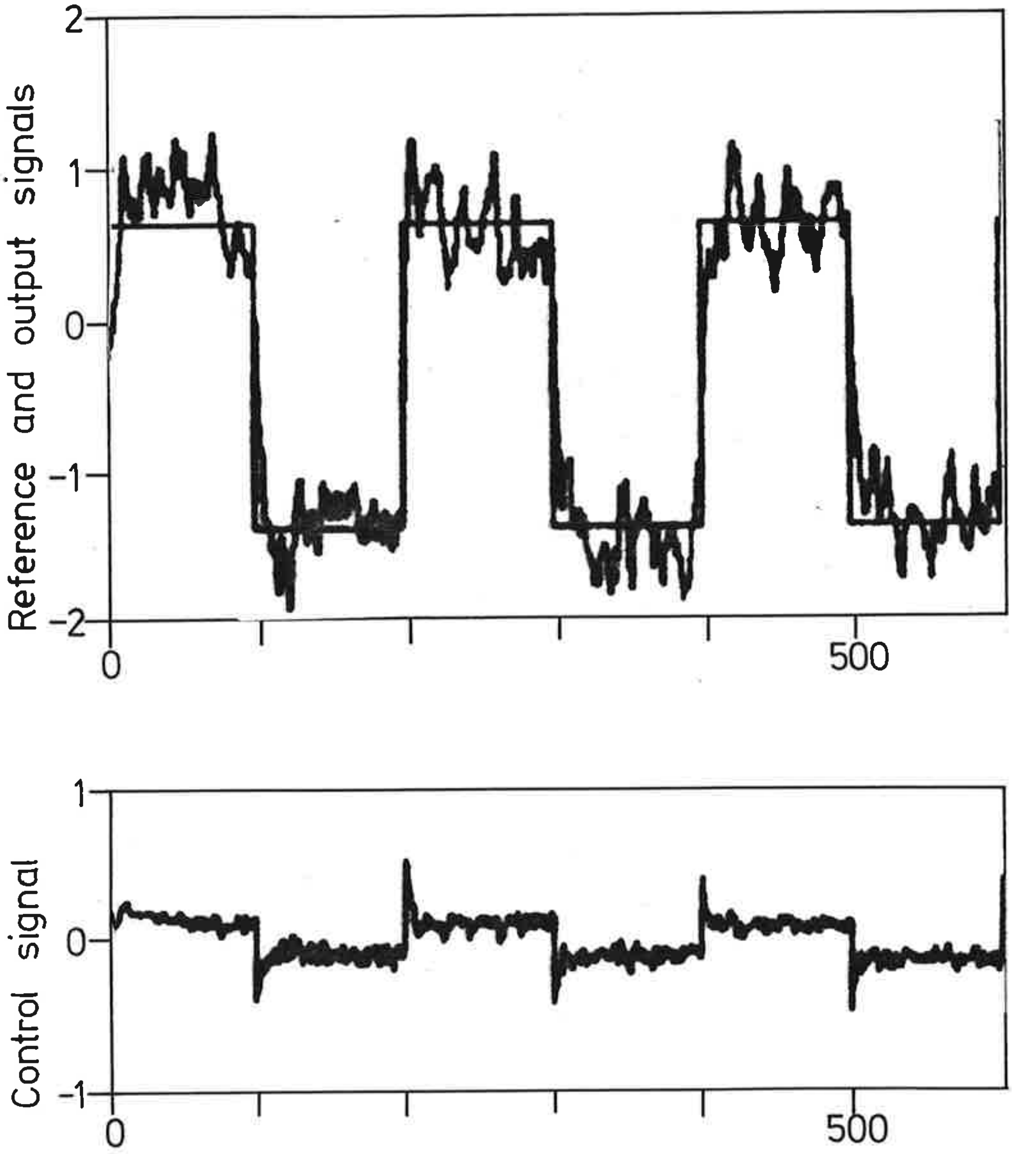


*Figure 6.4.* Command signal $y_r$, output signal $y$, and control signal $u$ for system (6.1) with $c = 0$ and $\sigma = 0.1$ controlled by the implicit algorithm I1.

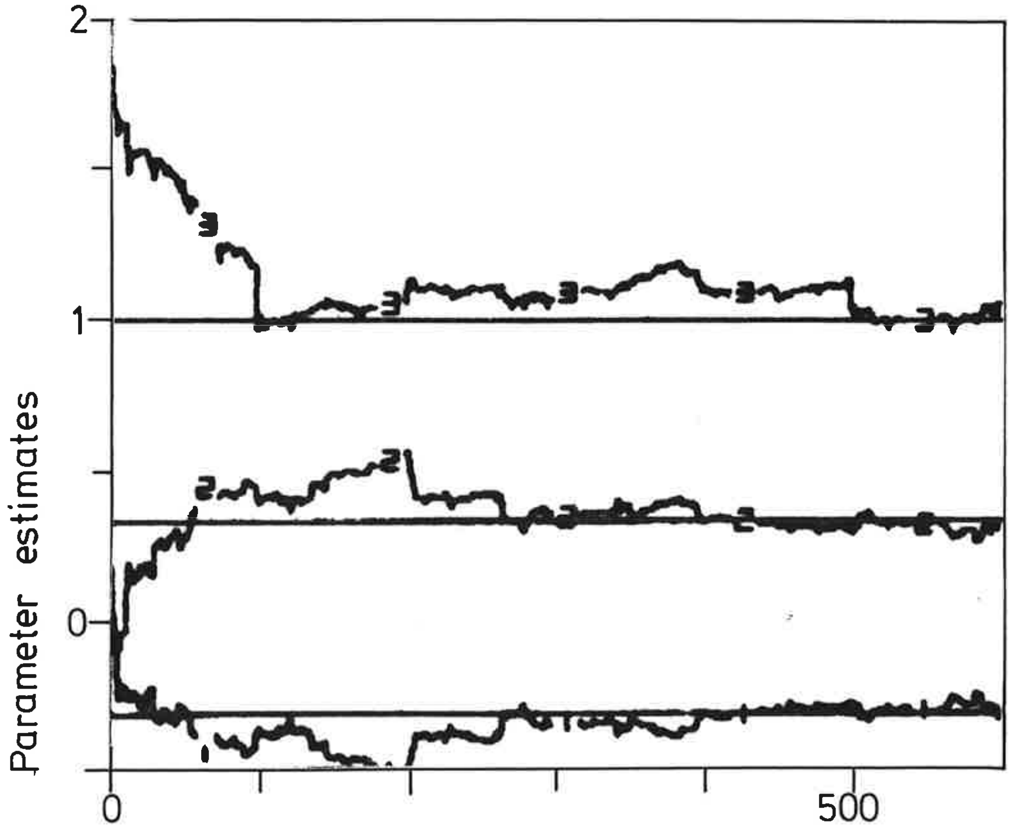


*Figure 6.5.* Parameter estimates for the system (6.1) with c = 0 and $\sigma$ = 0.1 controlled by the implicit algorithm I1.

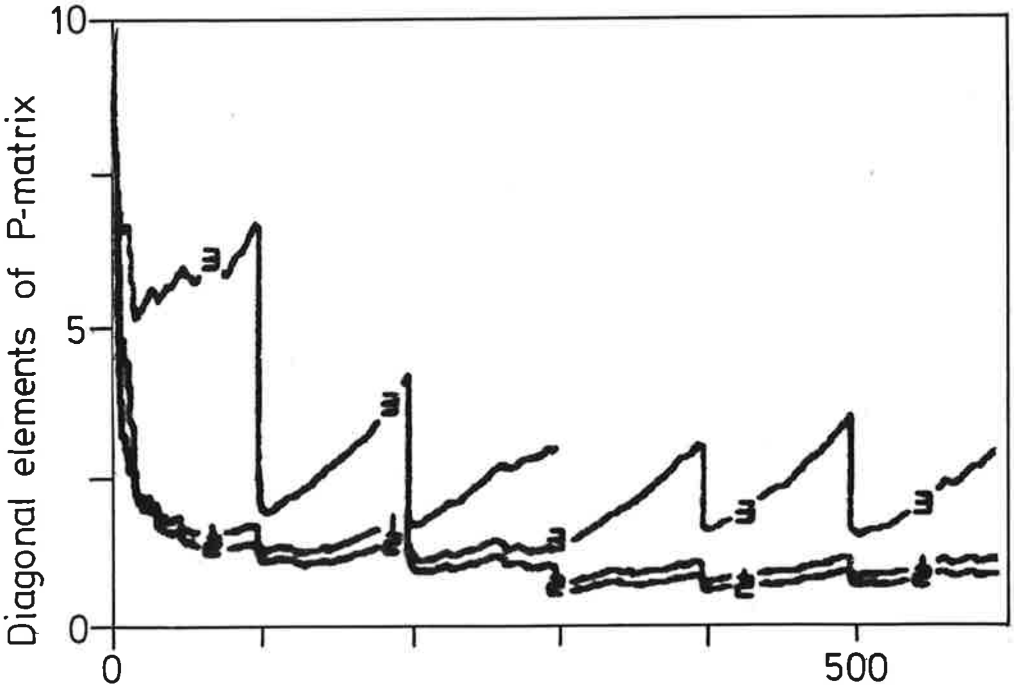


*Figure 6.6.* Diagonal elements of the matrix P of the estimation algorithm for the system (6.1) with c = 0 and $\sigma$ = 0.1 controlled by the implicit algorithm I1.

*CASE C (Coloured noise c = 0.5, σ = 0.1)*

This case illustrates the behaviour of the closed loop system when there are coloured disturbances. In this case the least squares estimates will be biased as is seen from Fig. 6.8. Since the disturbance level is low the performance of the regulator will still be adequate as is seen from Fig. 6.7 even if the estimation errors are substantial. The bias in the estimates can be eliminated by using another recursive estimator.

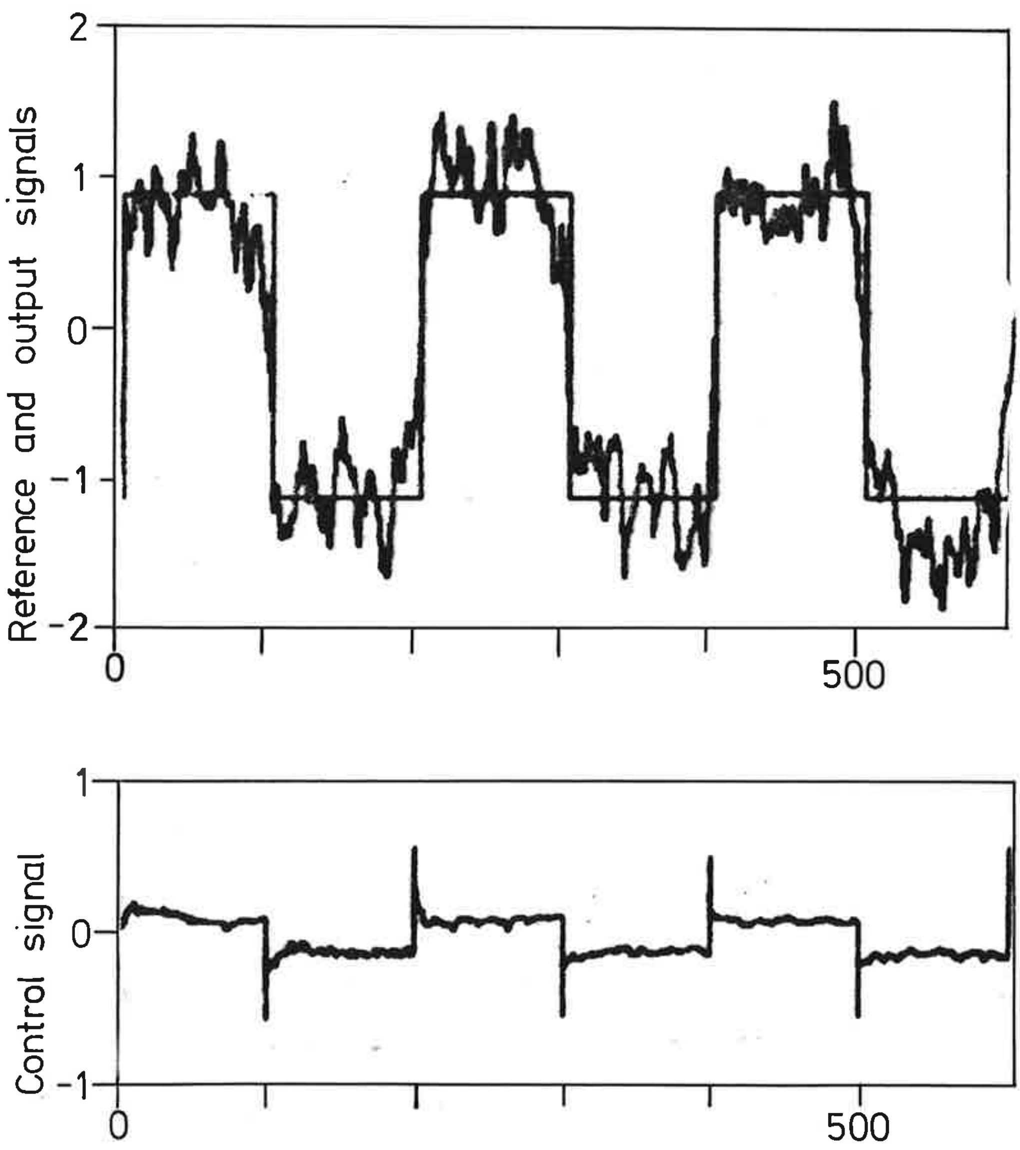


*Figure 6.7.* Command signal $y_r$, output signal $y$, and control signal $u$ for the system (6.1) with $c = 0.5$, $\sigma = 0.1$ controlled by the implicit algorithm I1.

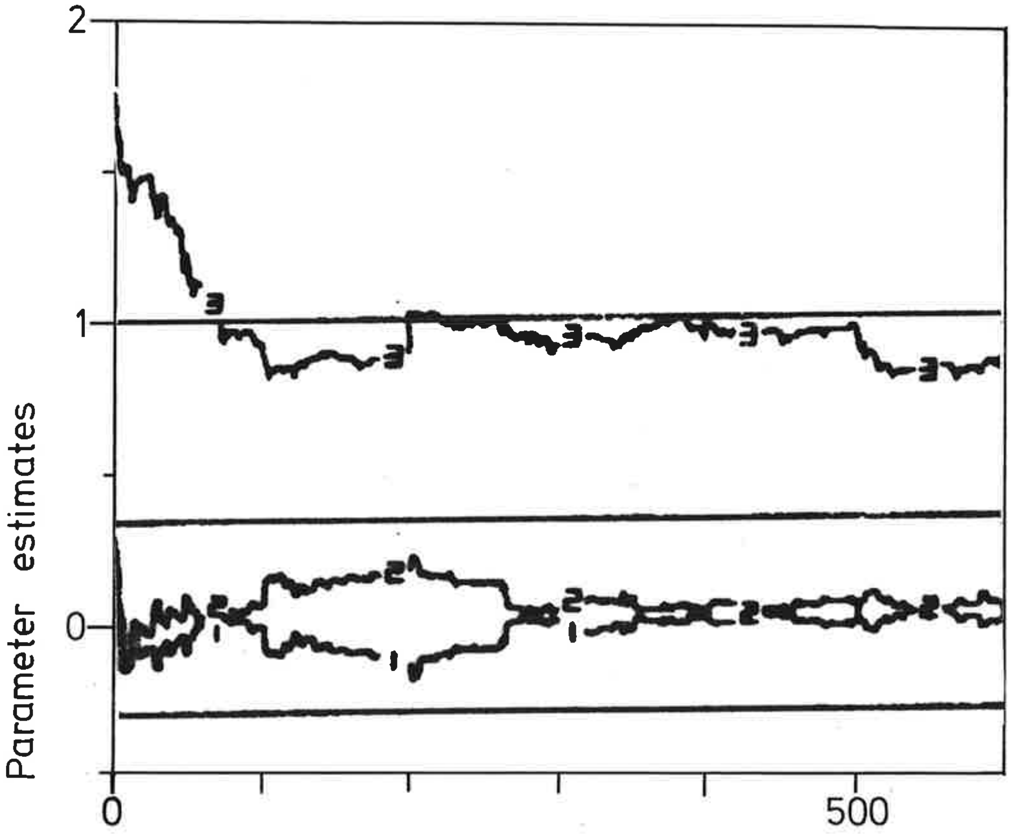


*Figure 6.8.* Parameter estimates for the system (6.1) with $c = 0.5$, $\sigma = 0.1$ controlled by the implicit algorithm I1. Notice that the estimates are biased.

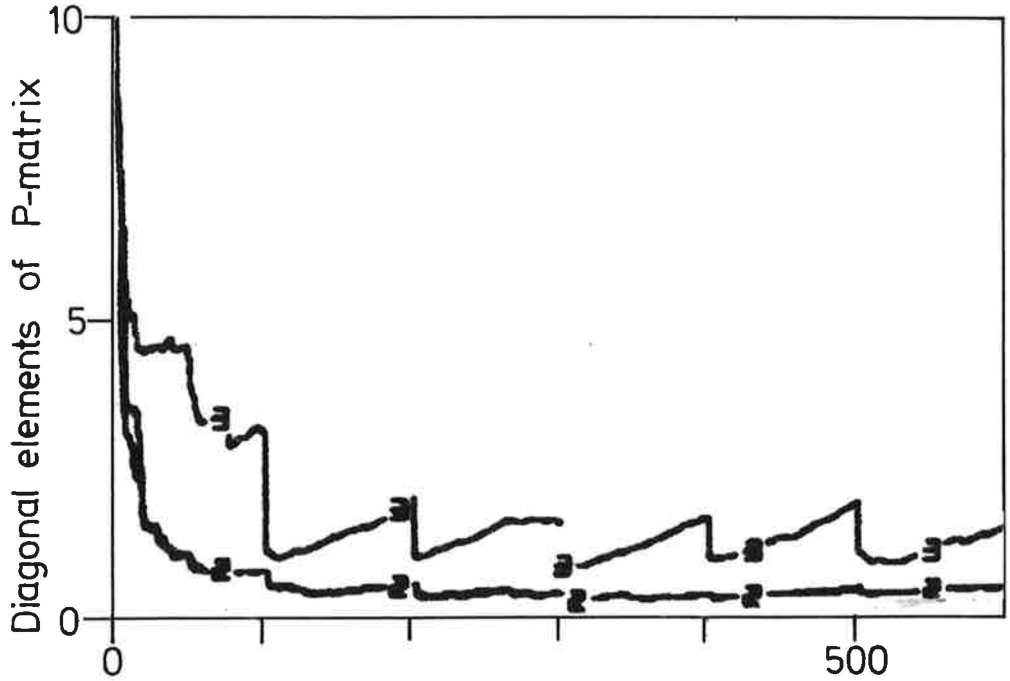


*Figure 6.9.* Diagonal elements of P-matrix in estimation algorithm for the system (6.1) with $c = 0.5$ and $\sigma = 0$ controlled by the implicit algorithm I1.

The system described in this example was also controlled by the explicit algorithm E1. The results were virtually the same as those obtained by the implicit algorithm I1 and the curves are therefore not shown.

EXAMPLE 6.2 - Non-minimum Phase System

Consider a system described by

$$y(t) - 1.5\, y(t-1) + 0.7\, u(t-2) = u(t-1) + 1.1\, u(t-2). \qquad (6.2)$$

The system is clearly non-minimum phase because its transfer function has the zero $z = -1.1$. Any design method which attempts to cancel the factor $z + 1.1$ will thus fail. Assume for example that it is desired to design a control law which gives a closed loop system with the transfer function

$$G_M = \frac{0.1225}{z^2 - 1.44z + 0.5625}.$$

The process (6.1) has the poles

$$0.75 \pm i\, 0.37$$

and the desired closed loop system has the poles

$$0.72 \pm i\, 0.21.$$

The specifications thus imply that the open-loop poles have to be shifted so that the damping is improved. Since the desired closed loop dynamics has no zeros, the factor $z + 1.1$ of the open-loop transfer function must be cancelled. This means that the control signal will increase without bound even in the case of known parameters. Any adaptive scheme based on a design procedure where the zero is cancelled will inherit this property. Fig. 6.10 shows what happens when the implicit algorithm I1 based on cancellation of the process zeros is used. The output appears very well-behaved after the first step. Because of the unstable cancelled factor the control variable will, however, grow exponentially as $(-1.1)^t$. Since the unstable mode $z = -1.1$ is cancelled by the process zero nothing is seen in the output for a while. At time 430 the input is, however, so large that round-off is noticable. The cancellation is no longer perfect and the unstable mode is seen in the output too.

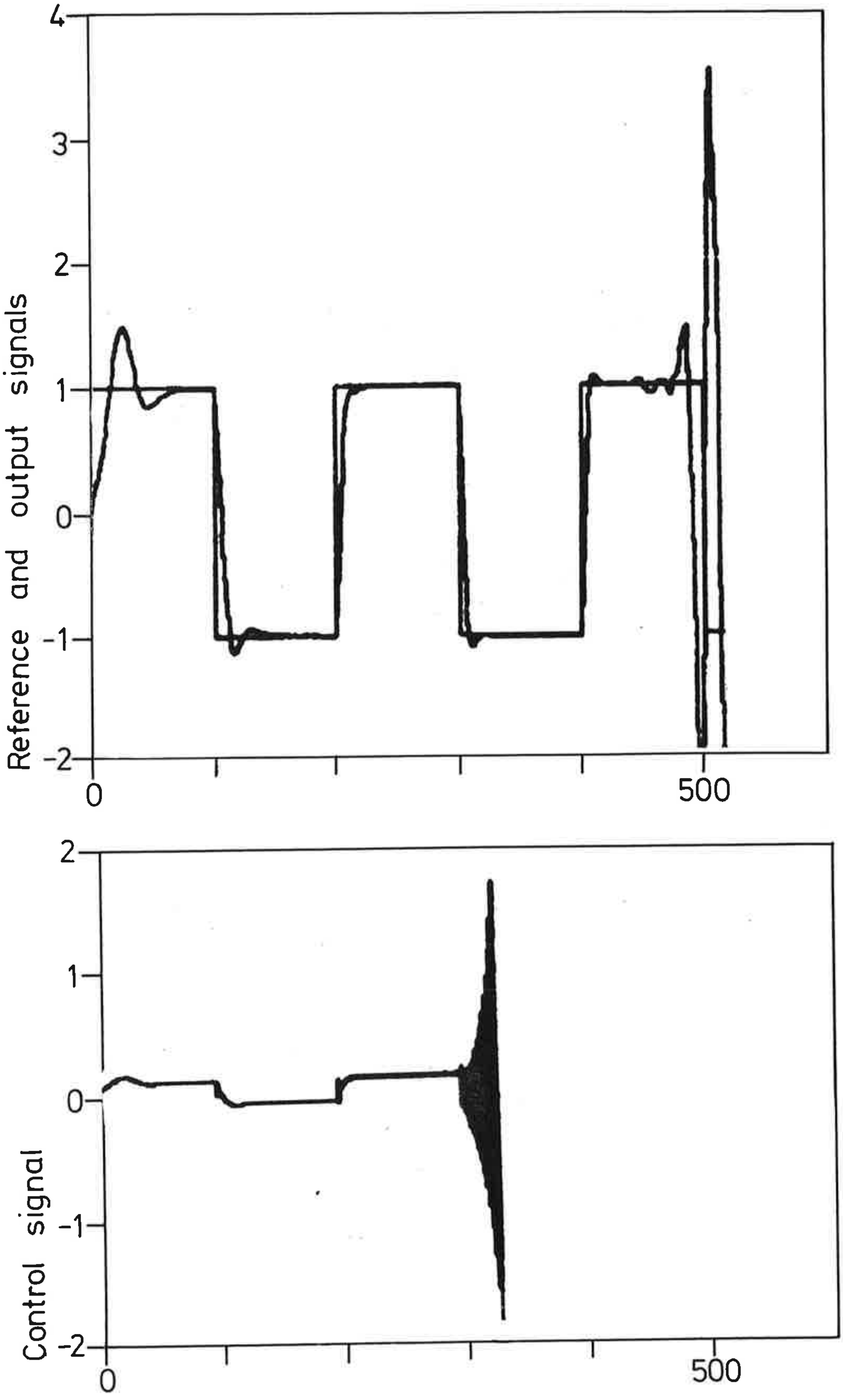


*Figure 6.10.* Command signal $y_r$, output $y$, and control signal $u$ when the non-minimum phase system (6.2) is controlled by the algorithm I1 based on cancellation of the process zeros.

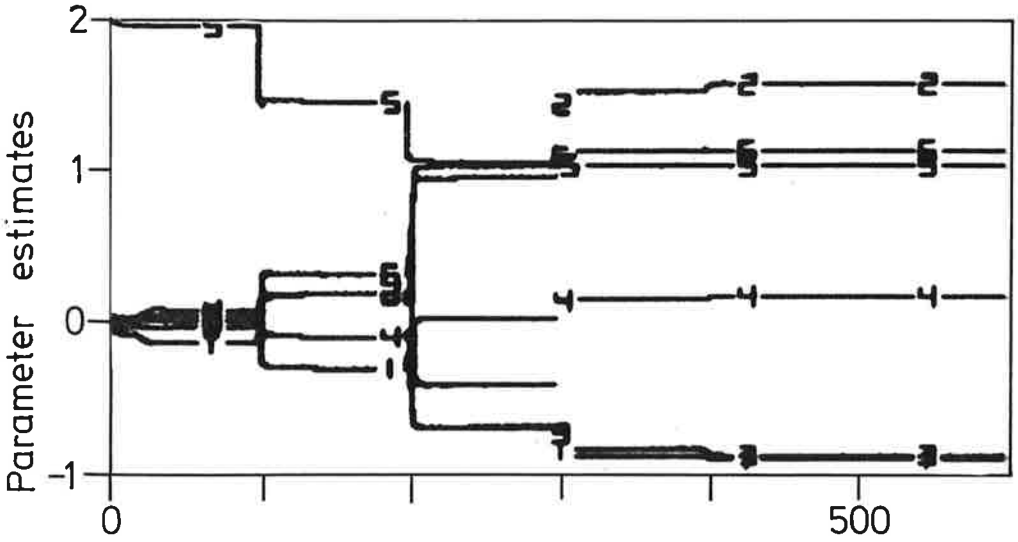


*Figure 6.11.* Parameter estimates obtained when the system (6.2) is controlled with the adaptive algorithm I1.

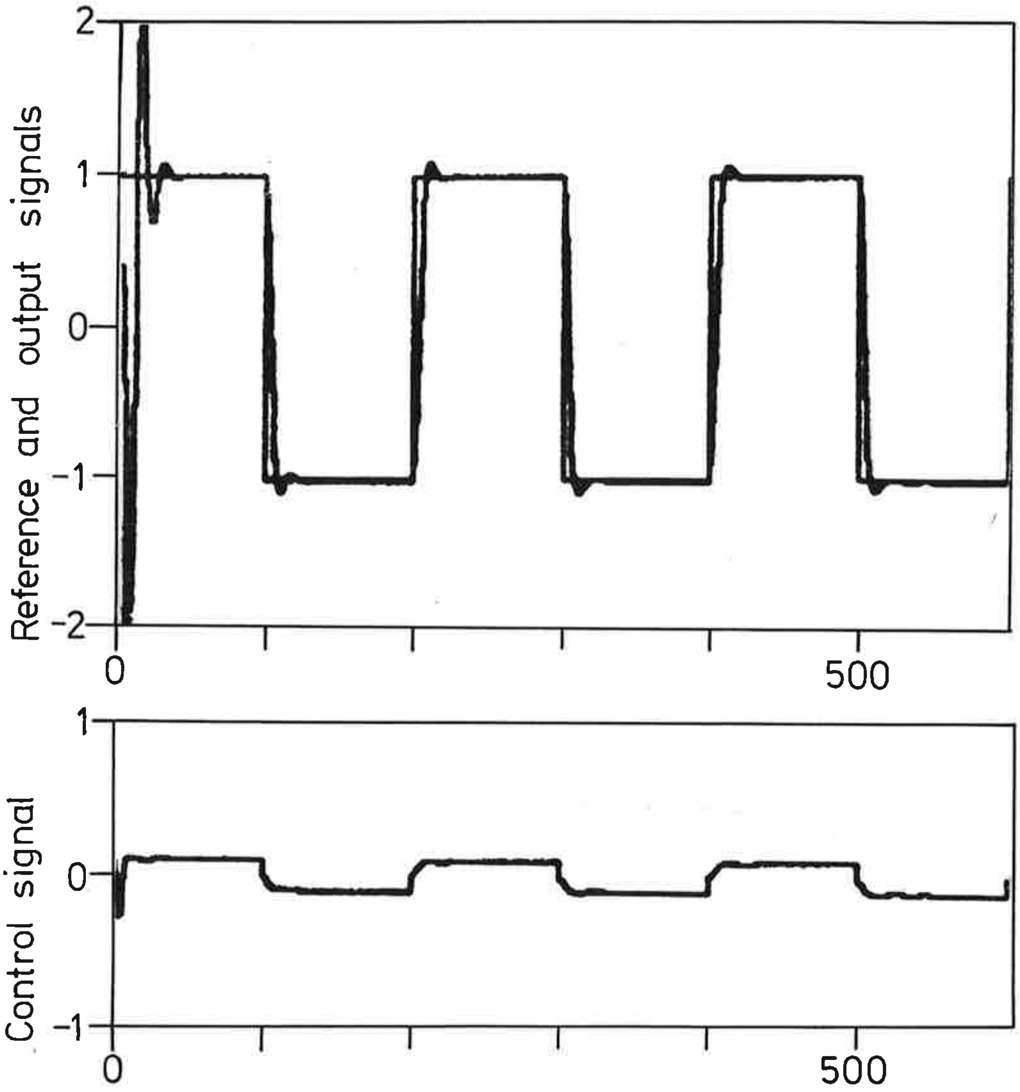


*Figure 6.12.* Command signal $y_r$, output signal $y$, and control signal $u$ when the system (6.2) is controlled by the algorithm E2.

Since the process zero can not be cancelled it is necessary that the specifications are changed so that the process zero $z = -1.1$ is also a zero of the desired closed loop transfer function. Hence assume that it is desired to have a closed loop system characterized by the transfer function

$$G_M = \frac{0.0583\ (z+1.1)}{z^2 - 1.44\,z + 0.5625} .$$

Assume in addition that the observer is required to have a double pole at $z = 0.5$. The behaviour of the closed loop system obtained when the explicit adaptive regulator E2, which retains all the process zeros, is used to control the plant is shown in Fig. 6.12 and Fig. 6.13. It is seen from the figures that the algorithm which does not attempt to cancel the process zeros works very well. Notice, however, that the closed loop system obtained will depend on the process zero.

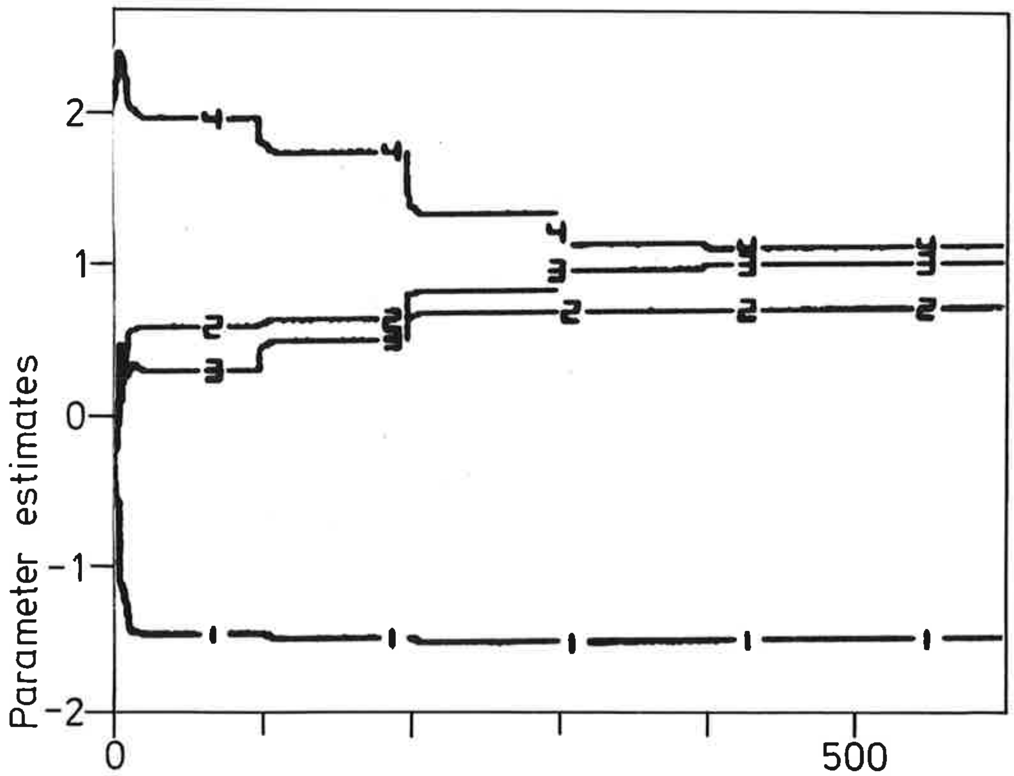


*Figure 6.13.* Parameter estimates obtained when the system (6.2) is controlled by the algorithm E2.

EXAMPLE 6.3 - Continuous Time Second Order System

This example illustrates the behaviour when the adaptive algorithms are used to control continuous time processes. It is assumed that the process is described by the transfer function

$$G(s) = \frac{0.864}{(s+0.36)(s+1.2)} \tag{6.3}$$

and that the sampling period of the digital regulator is 1 s. The sampled process has the pulse transfer function

$$H(z) = \frac{0.26\,z + 0.16}{z^2 - z + 0.21}. \tag{6.4}$$

This transfer function has the zero $z = -0.60$ which is inside the unit disc but poorly damped ($\zeta \approx 0.16$). The performance of the adaptive regulators will be explored.

It is first assumed that the specifications are such that the settling time of the system corresponds to a large number of sampling periods. To be specific it is assumed that the desired closed loop poles are

$$z = 0.72 \pm i\ 0.21 = 0.75\angle 16^{\circ}\ .$$

This means that the settling time of the system is about 20 sampling periods. The process zero at $z = -0.60$ corresponds to a mode with a period of 2 sampling periods. Since this mode is considerably faster than the desired closed loop dynamics it can be expected that the results obtained when the process zero is cancelled and when it is retained are similar. Even if the cancelled mode is poorly damped it will not be excited much because it is high frequent compared to the desired closed loop dynamics.

The characteristic polynomial of the desired closed loop system is

$$P = z^2 - 1.44z + 0.5625. \tag{6.5}$$

Assume that the observer is specified to have two poles at $z = 0.5$ i.e.

$$T = (z-0.5)^2. \tag{6.6}$$

The results obtained when using the implicit adaptive control algorithm I1 based on cancellation of the process zero are shown in Fig. 6.14 and Fig. 6.15.

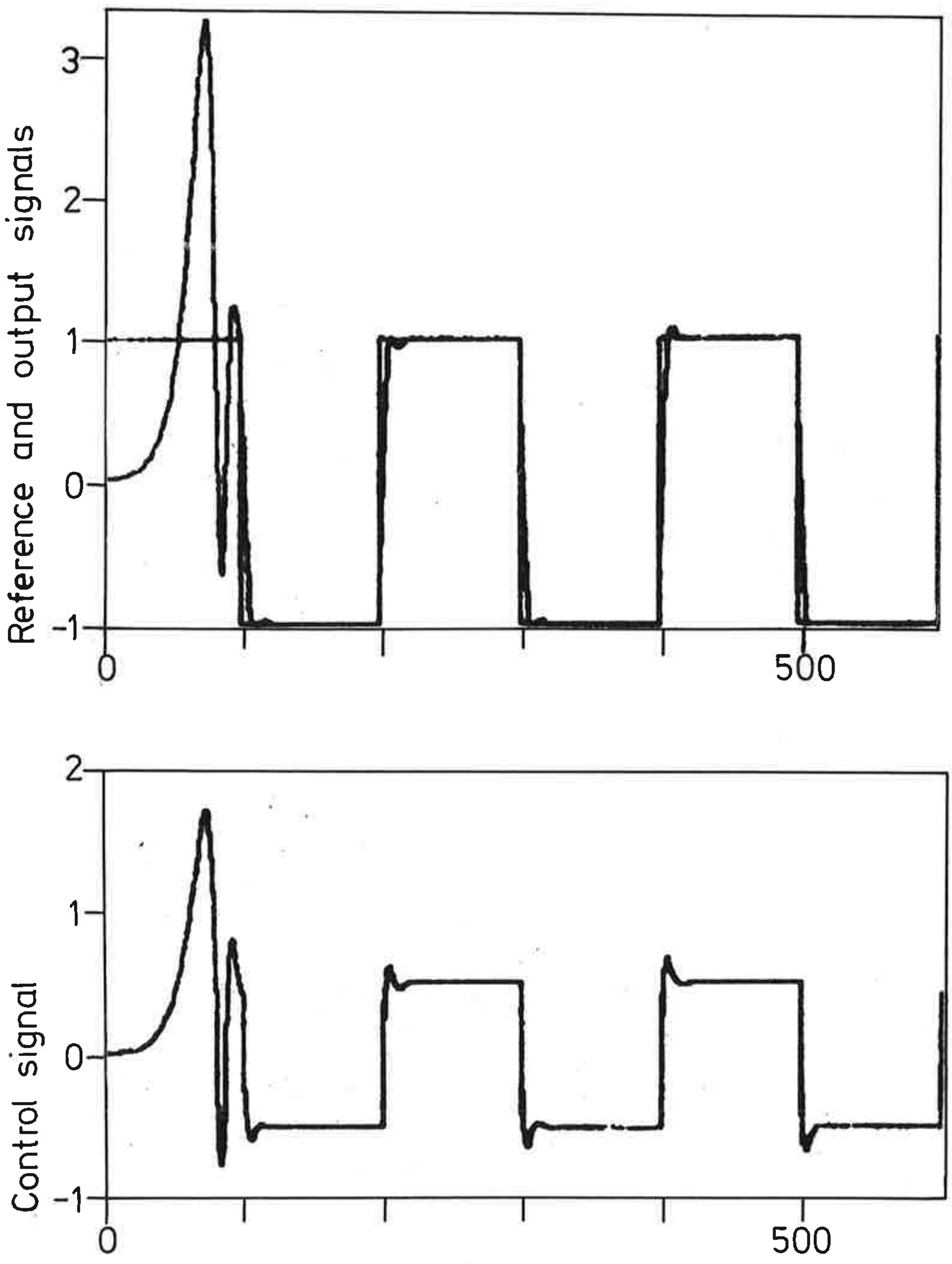


*Figure 6.14.* Command signal $y_r$, output signal $y$, and control signal $u$ when the system (6.3) is controlled by the algorithm I1 with P and T given by (6.5) and (6.6).

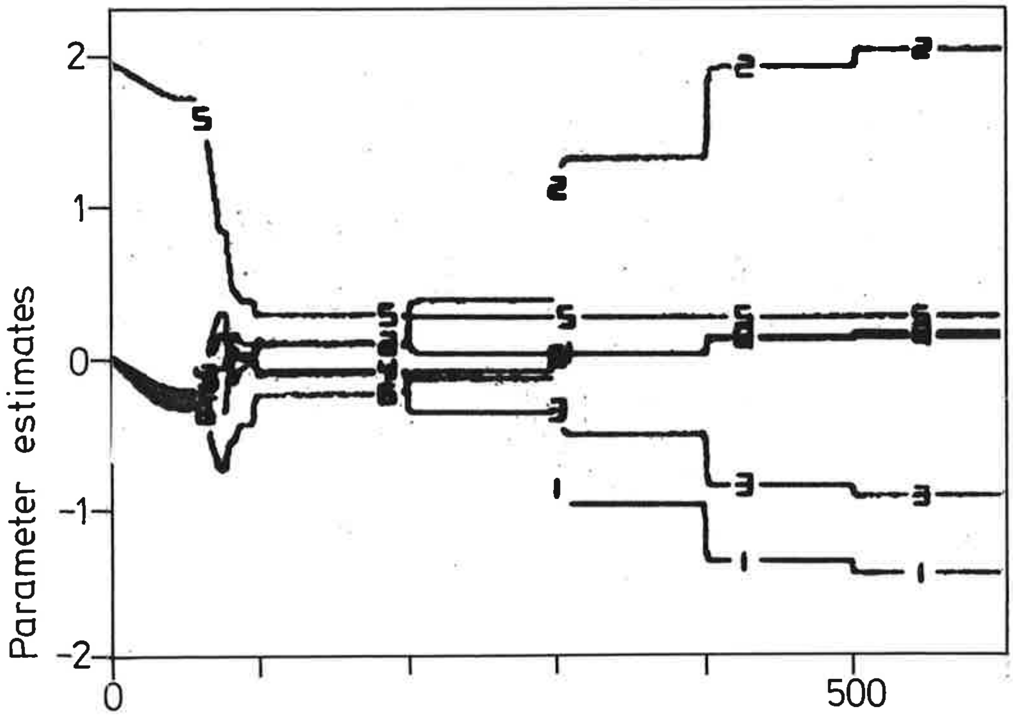


*Figure 6.15.* Parameter estimates obtained when the system (6.3) is controlled by the algorithm I1 with P and T given by (6.5) and (6.6).

The corresponding results obtained when using the explicit algorithm E2 based on a design where the process zero is retained are shown in Fig. 6.16 and Fig. 6.17. The behaviour obtained with these adaptive regulators is similar. The algorithm I1 gives somewhat larger excursions initially but the parameter estimates converge a little faster. Notice that the implicit algorithm is simpler but that more parameters have to be estimated in this algorithm. The limiting behaviour of the systems obtained is very similar as can be expected from the qualitative discussion made in the beginning of this section. This can be seen from Fig. 6.18 and Fig. 6.19 which show the responses of the systems in the time interval 400 to 450 s. With the resolution used in the graphs the output signals can not be distinguished. These poorly damped cancelled modes can, however, be traced in the control signal in Fig. 6.18.

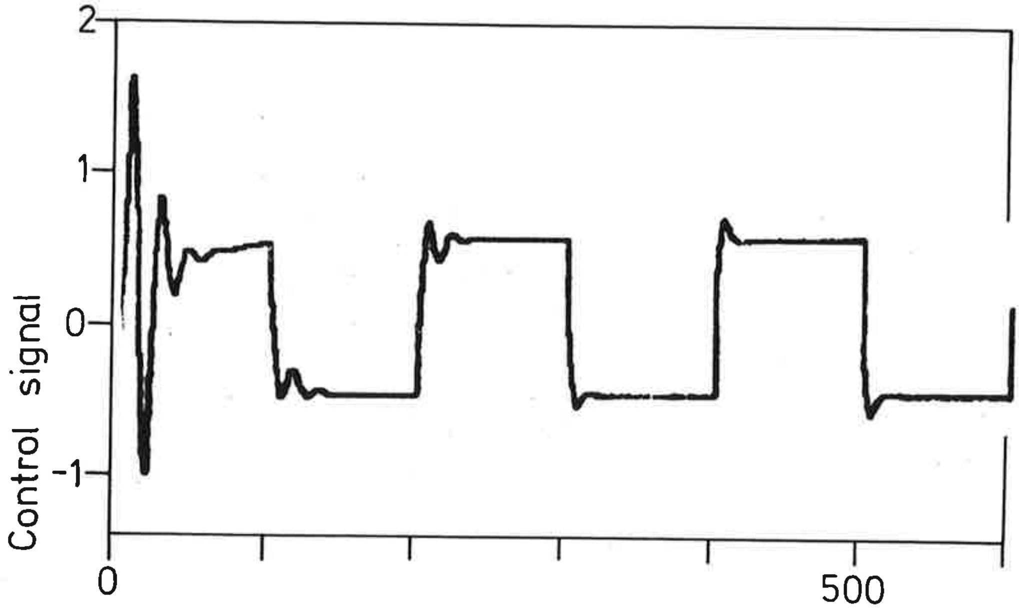



*Figure 6.16.* Command signal $y_r$, output signal $y$, and control signal u obtained when the system (6.3) is controlled by the algorithm E2 with P and T given by (6.5) and (6.6).

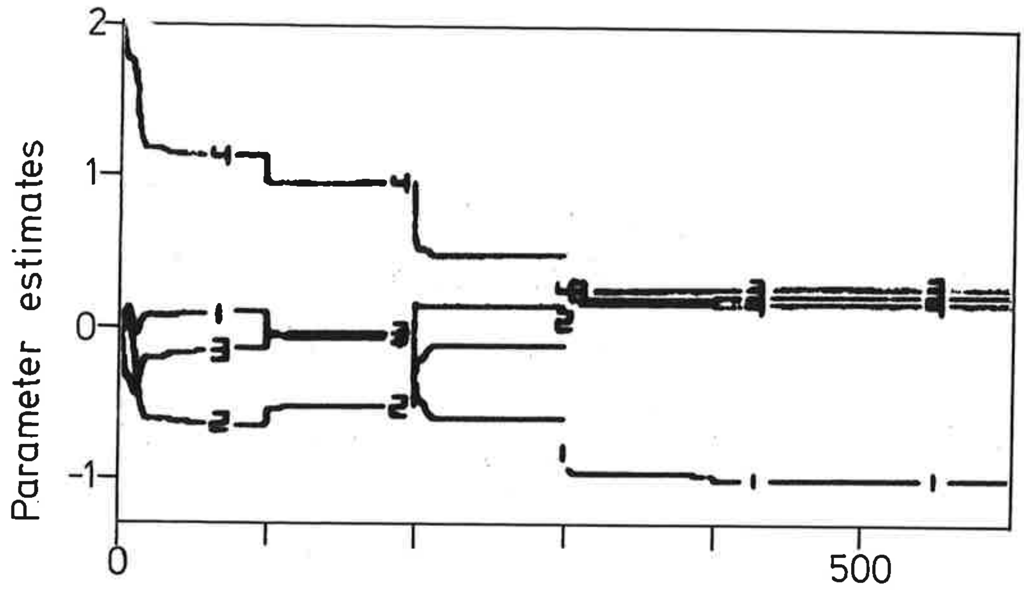


*Figure 6.17.* Parameter estimates obtained by the algorithm E2 with P and T given by (6.5) and (6.6).

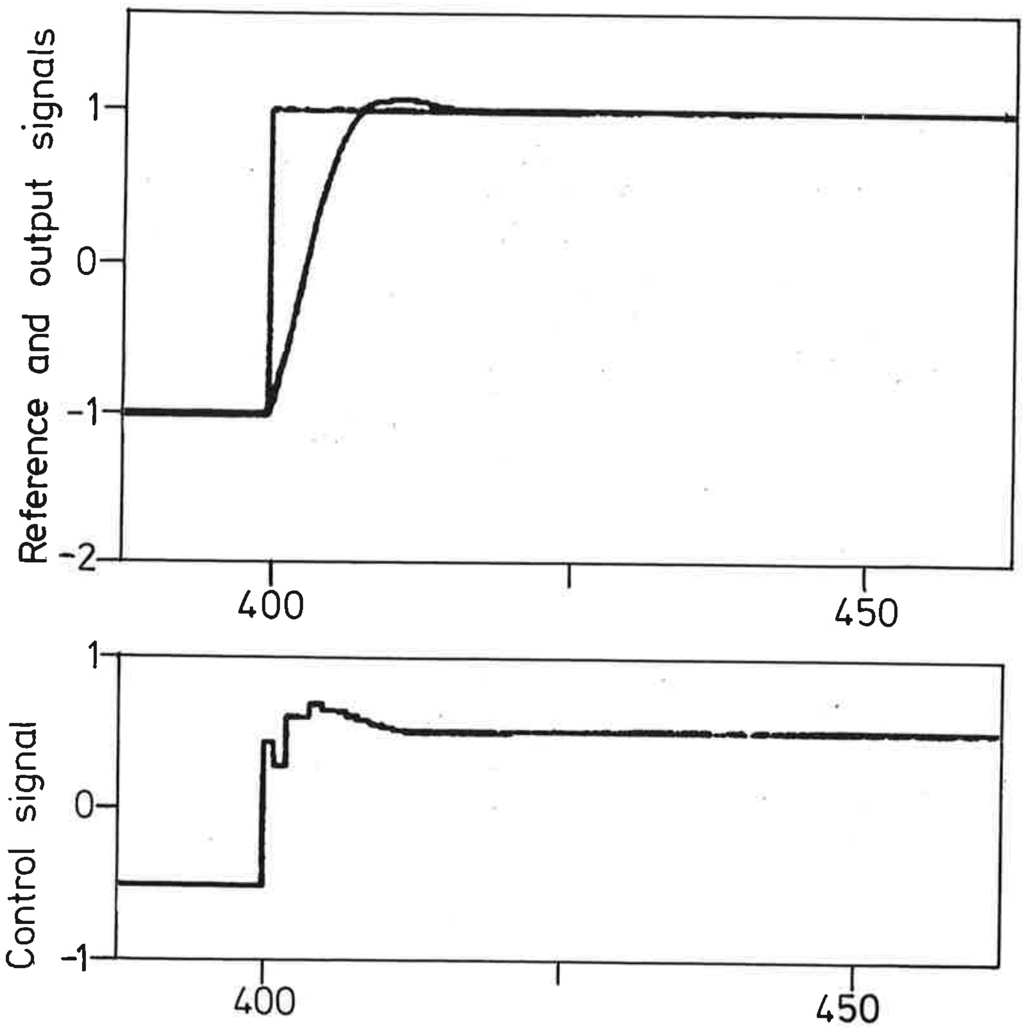


*Figure 6.18.* Command signal $y_r$, output signal y, and control signal u when the system (6.3) is controlled by the algorithm I1 with P and T given by (6.5) and (6.6).

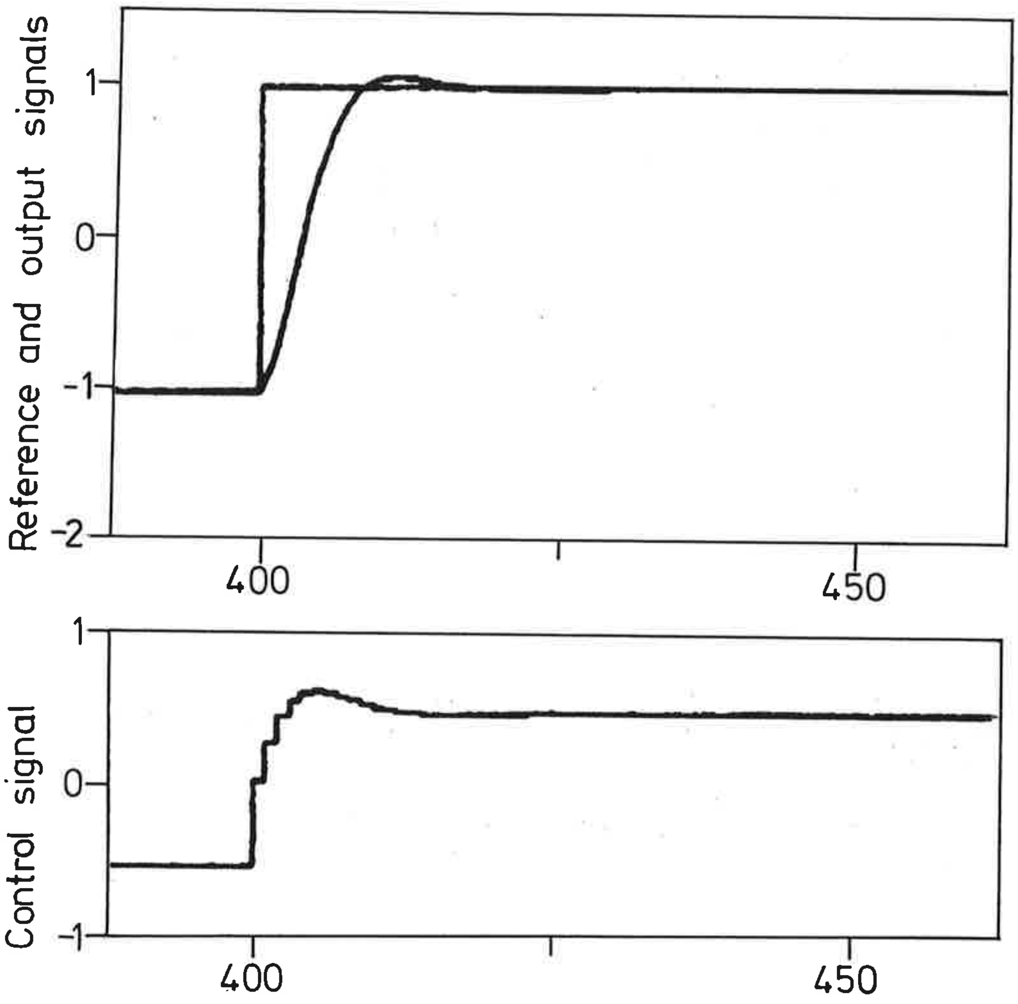


*Figure 6.19.* Command signal $y_r$, output signal y, and control signal u when the system (6.3) is controlled by the algorithm E2 with P and T given by (6.5) and (6.6).

EXAMPLE 6.4 - Same system as in Example 6.3 but different specifications

In this example the process discussed is the same as in Example 6.3. The sampled process thus has the pulse transfer function (6.4). It will, however, be assumed that the specifications require a much higher bandwidth of the desired closed loop system. To be specific the extreme case of dead-beat response is specified. It is thus required that the desired closed loop system has all its poles at the origin i.e.

$$P = z^2. \tag{6.7}$$

It is assumed as in Example 6.3 that the observer has two poles at $z = 0.5$ i.e.

$$T = (z - 0.5)^2. \tag{6.8}$$

The desired dynamics of the closed loop system now has a characteristic frequency corresponding to 2 sampling intervals (2 s). This frequency is the same as the frequency associated with the process zero at $z = -0.6$. It can thus be expected that there will be a substantial difference between the control designs obtained when the process zero is cancelled and when it is retained.

When the adaptive algorithms I1 and E2 are used the parameters will converge quickly in both cases. The behaviour obtained with the implicit algorithm I1 which is based on a design procedure where the process zero is cancelled is shown in Fig. 6.20. Notice that the output is equal to

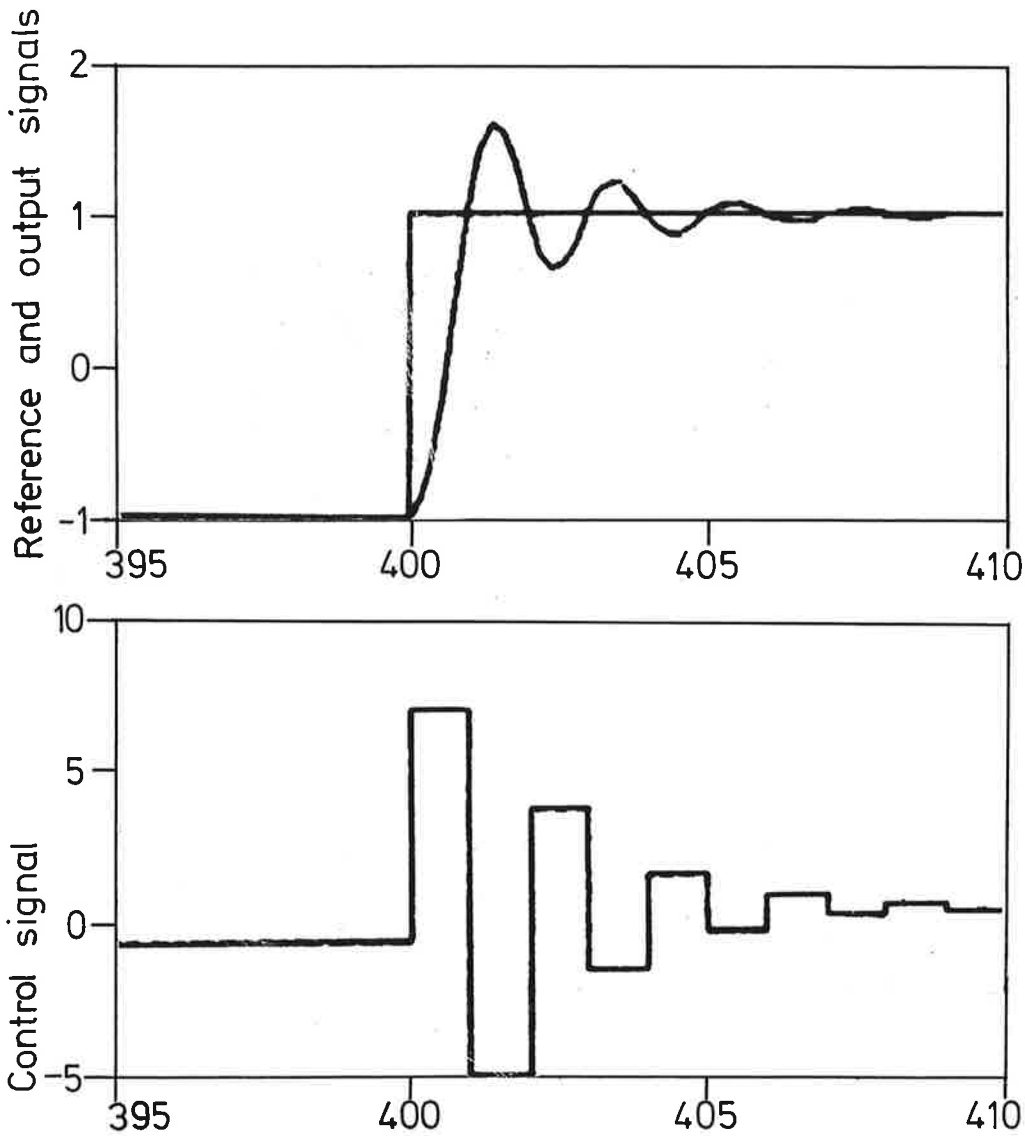


*Figure 6.20.* Command signal $y_r$, output signal y, and control signal u when the system (6.3) is controlled by by the algorithm I1 with P and T given by (6.7) and (6.8).

the desired output *one* sampling period after the step-change in the command signal. Also notice that the output is zero at the future sampling instants but that large deviations occur between the sampling instants due to the poorly damped cancelled factor. Also notice that the cancelled mode z = -0.6 is clearly noticable in the control signal.

The corresponding behaviour obtained with the explicit algorithm E2, which based on a design where the process zero is retained, is shown in Fig. 6.21. Notice that the process output reaches the desired value *two* sampling periods after the step in the command signal. The behaviour is otherwise much superior to the behaviour obtained with the algorithm I1 which was based on cancellation of the process zero at z = -0.6.

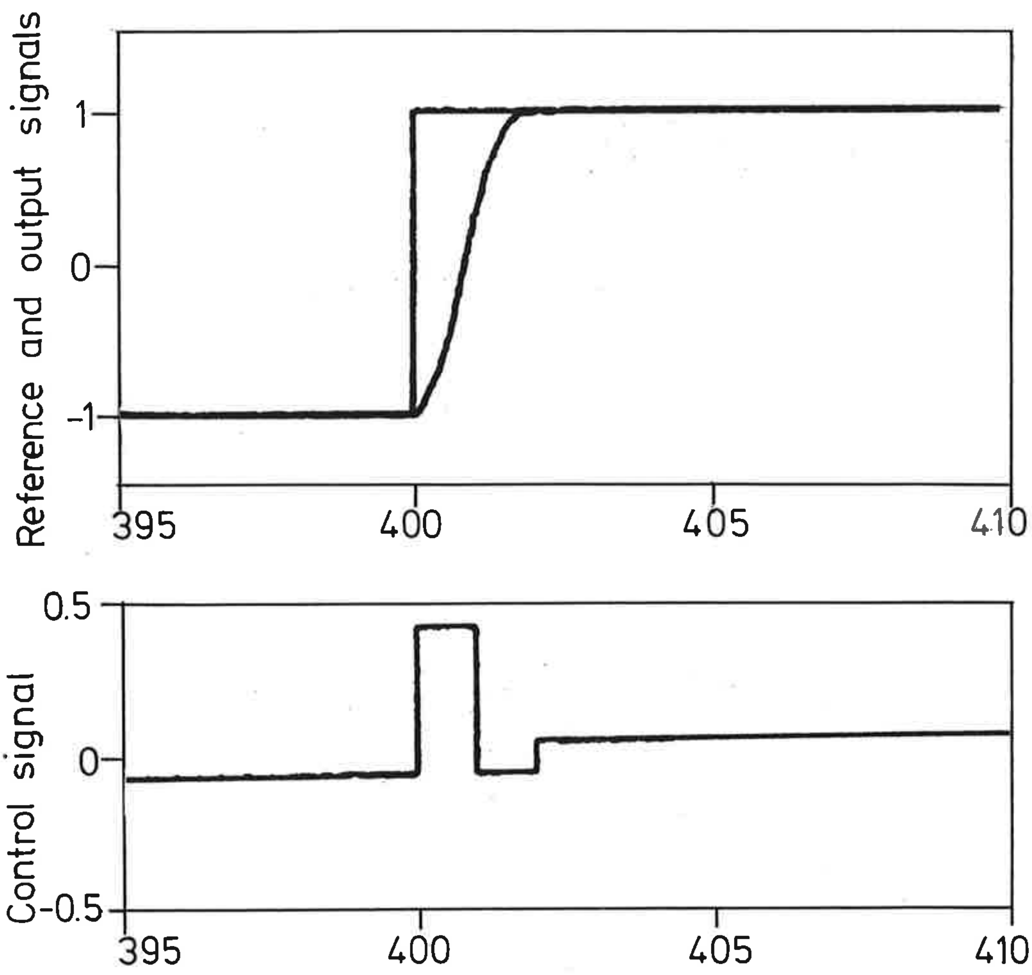


*Figure 6.21.* Command signal $y_r$, output signal y, and control signal u when the system (6.3) is controlled by the algorithm E2 with P and T given by (6.7) and (6.8).

## 7. ACKNOWLEDGEMENTS

This work has been partially supported by the Swedish Board of Technical Development under contract No. 76-3804.

## 8. REFERENCES

Anderson, L (1977): DISCO - An educational microcomputer controller. Preprints IFAC Symposium on Trends in Automatic Control Education, Barcelona, Spain, pp. 32-45.

Åström, K J (1968): Lectures on the identification problem - the least squares method. TFRT-3004, Dept of Automatic Control, Lund Institute of Technology, Sweden.

Åström, K J (1970): Introduction to stochastic control theory. Academic Press, New York.

Åström, K J (1974): A self-tuning regulator for non-minimum phase systems. TFRT-3113, Dept of Automatic Control, Lund Institute of Technology, Sweden.

Åström, K J (1976): Reglerteori. AWE/Gebers, Uppsala.

Åström, K J, and B Wittenmark (1973): On self-tuning regulators. Automatica 9, 185-189.

Åström, K J, and B Wittenmark (1974): Analysis of a self-tuning regulator for non-minimum phase systems. Preprints IFAC Symposium on Stochastic Control, Budapest, Hungary.

Åström, K J, U Borisson, L Ljung, and B Wittenmark (1977): Theory and applications of self-tuning regulators. Automatica 13, 457-476.

Bengtsson, G (1977): Output regulation and internal models - a frequency domain approach. Automatica 13, 333-345.

Clarke, D W, and P J Gawthrop (1975): Self-tuning controller. Proc IEE 122, 929-934.

Edmunds, J M (1976): Digital adaptive pole shifting regulators. PhD dissertation, Control Systems Centre, Institute of Science and Technology, The University of Manchester, Manchester, England.

Egardt, B (1978): A unified approach to model reference adaptive systems and self-tuning regulators. Dept of Automatic Control, Lund Institute of Technology, Sweden, CODEN: LUTFD2/(TFRT-7134)/ /1-67/(1978).

Elmqvist, H (1975): SIMNON - User's manual. TFRT-3091, Dept of Automatic Control, Lund Institute of Technology, Sweden.

Fortescue, T (1975). Private communication.

Franklin, G F (1977). Private communication.

Gawthrop, P J (1977): Some interpretations of the self-tuning controller. Proc IEE 124, 889-894.

Gustavsson, I (1978): User's guide for a program package for simulation of self-tuning regulators. Dept of Automatic Control, Lund Institute of Technology, Sweden.

Kucera, V (1975): Algebraic theory of discrete optimal control for multivariable systems. Supplement Kybernetica 11.

Levinson, N (1947): The Wiener RMS (root mean square) error in filter design and prediction. J Math Phys 28, 261-278.

Morris, A J, T P Fenton, and Y Nazer (1977): Application of self--tuning regulators to the control of chemical processes. In H R van Nauta Lemke and H B Vervruggen (Eds): Digital Computer Applications to Process Control, Preprints of the 5th IFAC/IFIP International Conference, The Hague, Netherlands, June 14-17.

Pernebo, L (1978): Algebraic control theory for linear multivariable systems. Dissertation, Dept of Automatic Control, Lund Institute of Technology, Sweden, CODEN: LUTFD2/(TFRT-1016)/1-307/(1978).

Peterka, V (1975): A square-root filter for real-time multivariable regression. Kybernetica 11, 53-67.

Ragazzini, J R, and G F Franklin (1958): Sampled-data control systems. McGraw-Hill, New York.

Westerberg, B (1977): Självinställande regulator baserad på polplacering. Dept of Automatic Control, Lund Institute of Technology, Sweden, CODEN: LUTFD2/(TFRT-5198)/1-65/(1977).

Wittenmark, B (1976): A design example of a sampled data system. TFRT-3130, Dept of Automatic Control, Lund Institute of Technology, Sweden.

Wellstead, P E (1978): On the self-tuning properties of pole/zero assignment regulators. Report 402, Control Systems Centre, The University of Manchester, Institute of Science and Technology, Manchester, England.

Wellstead, P E, P Zauber, and J M Edmunds (1978): Self-tuning pole/zero assignment regulators. Report 404, Control Systems Centre, The University of Manchester, Institute of Science and Technology, Manchester, England.

Wolovich, L A (1974): Linear multivariable systems. Springer, New York.

Wouters, W R E (1977): Adaptive pole placement for linear stochastic systems with unknown parameters. Proc IEEE Conference on Decision and Control, New Orleans, USA.

# APPENDIX

The examples in Chapter 6 were simulated using a program package for simulation of self-tuning regulators, Gustavsson (1978). This program package is found on disc No. 9. All the necessary commands for each simulated example are listed in this appendix.

## EXAMPLE 6.1

*Case (a)*

External systems required: REF, SCON3
Macro required: EX61I

```
LET IVR. = 3
LET ISA. = 3
LET ISB. = 2
LET IVS. = 1
SYST REG SYS1 REF SCON3
EX61I
PAR LAMB:0
```

*Case (b)*

External systems required: REF, SCON3
Macro required: EX61I

```
LET IVR. = 3
LET ISA. = 3
LET ISB. = 2
LET IVS. = 1
SYST REG SYS1 REF SCON3
EX61I
PAR LAMB:0.1
```

*Case (c)*

External systems required: REF, SCON3
Macro required: EX61I

```
LET IVR. = 3
LET ISA. = 3
LET ISB. = 2
LET IVS. = 1
SYST REG SYS1 REF SCON3
EX61I
PAR NSC:1
PAR LAMB:0.1
PAR C1:0.5
```

EXAMPLE 6.2

*Implicit algorithm I1*

External systems required: REF, SCON3
Macro required: EX62I

```
LET IVR. = 6
LET ISA. = 3
LET ISB. = 4
LET IVS. = 2
SYST REG SYS1 REF SCON3
EX62I
```

*Explicit algorithm E2*

External systems required: REF, SCON3
Macro required: EX62E

```
LET IVR. = 4
LET ISA. = 3
LET ISB. = 2
LET IVS. = 2
SYST REG SYS1 REF SCON3
EX62E
```

EXAMPLE 6.3

*Implicit algorithm I1*

External systems required: SYS2, REF, SCON4
Macro required: EX63I

```
ALGOR RK
LET IVR. = 6
LET ISA. = 3
LET ISB. = 4
LET IVS. = 2
SYST REG SYS2 REF SCON4
EX63I
```

*Explicit algorithm E2*

External systems required: SYS2, REF, SCON4
Macro required: EX63E

```
ALGOR RK
LET IVR. = 4
LET ISA. = 3
LET ISB. = 2
LET IVS. = 2
SYST REG SYS2 REF SCON4
EX63E
```

EXAMPLE 6.4

*Dead-beat regulation with the implicit algorithm I1*

External systems required:  SYS2, REF, SCON4
Macro required:  EX63I

```
ALGOR RK
LET IVR. = 6
LET ISA. = 3
LET ISB. = 4
LET IVS. = 2
SYST REG SYS2 REF SCON4
EX63I
PAR D2:0
PAR D3:0
```

*Dead-beat regulation with the explicit algorithm E2*

External systems required:  SYS2, REF, SCON4
Macro required:  EX63E

```
ALGOR RK
LET IVR. = 4
LET ISA. = 3
LET ISB. = 2
LET IVS. = 2
SYST REG SYS2 REF SCON4
EX63E
PAR D2:0
PAR D3:0
```

EXTERNAL SYSTEMS

```
CONTINUOUS SYSTEM SYS2
INPUT U
OUTPUT Y
STATE X1 X2
DER DX1 DX2
OUTPUT
Y=B1*X1+B2*X2
DYNAMICS
DX1=-A1*X1-A2*X2+U
DX2=X1
A1:1.56
A2:0.432
B1:0
B2:0.864
END
```

```
DISCRETE SYSTEM REF
TIME T
OUTPUT Y
TSAMP TS
OUTPUT
Y=IF MOD(T,PER)<(0.5*PER-EPS) THEN NIV1 ELSE NIV2
DYN[illegible]
TS=[illegible]
PER[illegible]
NI[illegible]1
NI[illegible]
EPS:0.00001
DT:1
END
```

```
CONNECTING SYSTEM SCON3
TIME T
U1[REG]=Y[SYS1]-Y[REF]
U3[REG]=Y[REF]
U[SYS1]=UR[REG]
U2[REG]=U[SYS1]
ULEV:0
END
```

```
CONNECTING SYSTEM SCON4
TIME T
U1[REG]=Y[SYS2]-Y[REF]
U3[REG]=Y[REF]
U[SYS2]=UR[REG]
U2[REG]=U[SYS2]
ULEV:0
END
```

```
MACRO EX62E
PAR REG:5
PAR N1:2
PAR N2:2
PAR TH04:2
PAR P01:10
PAR P02:10
PAR P03:10
PAR P04:10
PAR WTI:0.98
PAR REF:1
PAR ND:3
PAR NF:3
PAR D1:1
PAR D2:-1.44
PAR D3:0.5625
PAR F1:1
PAR F2:-1
PAR F3:0.25
PAR NSA:2
PAR NSB:2
PAR LAMB:0
PAR A1:-1.5
PAR A2:0.7
PAR B1:1
PAR B2:1.1
PAR PER:200
END
```

```
MACRO EX63I
PAR REG:4
PAR N1:4
PAR N2:2
PAR TH05:2
PAR P01:10
PAR P02:10
PAR P03:10
PAR P04:10
PAR P05:10
PAR P06:10
PAR WTI:0.98
PAR REF:1
PAR ND:3
PAR NF:3
PAR D1:1
PAR D2:-1.44
PAR D3:0.5625
PAR F1:1
PAR F2:-1
PAR F3:0.25
PAR PER:200
END
```

```
MACRO EX61I
PAR REG:4
PAR N1:2
PAR N2:1
PAR TH03:2
PAR P01:10
PAR P02:10
PAR P03:10
PAR WTI:0.99
PAR REF:1
PAR ND:2
PAR NF:2
PAR D1:1
PAR D2:-0.7
PAR F1:1
PAR F2:-0.5
PAR NSA:1
PAR NSB:1
PAR A1:-0.9
PAR B1:1
PAR PER:200
END
```

```
MACRO EX62I
PAR REG:4
PAR N1:4
PAR N2:2
PAR TH05:2
PAR P01:10
PAR P02:10
PAR P03:10
PAR P04:10
PAR P05:10
PAR P06:10
PAR WTI:0.98
PAR REF:1
PAR ND:3
PAR NF:3
PAR D1:1
PAR D2:-1.44
PAR D3:0.5625
PAR F1:1
PAR F2:-1
PAR F3:0.25
PAR NSA:2
PAR NSB:2
PAR LAMB:0
PAR A1:-1.5
PAR A2:0.7
PAR B1:1
PAR B2:1.1
PAR PER:200
END
```

```
MACRO EX63E
PAR REG:5
PAR N1:2
PAR N2:2
PAR TH04:2
PAR P01:10
PAR P02:10
PAR P03:10
PAR P04:10
PAR WTI:0.98
PAR REF:1
PAR ND:3
PAR NF:3
PAR D1:1
PAR D2:-1.44
PAR D3:0.5625
PAR F1:1
PAR F2:-1
PAR F3:0.25
PAR PER:200
END
```