Adaptive Prediction and Recursive Estimation

Holst, Jan

1977

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

Jan Holst

Adaptive Prediction and Recursive Estimation

Lund 1977

Dokumentutgivare
Lund Institute of Technology
Handläggare Dept of Automatic Control
Karl Johan Åström
Författare
Jan Holst

Dokumentnamn
REPORT
Utgivningsdatum
Sept 1977

Dokumentbeteckning
LUTFD2/(TFRT-1013)/1-206/(1977)
Ärendebeteckning

Dokumenttitel och undertitel

Adaptive Prediction and Recursive Estimation

Referat (sammandrag)

In the first part of the thesis, algorithms for adaptive prediction of ARMA processes are studied. Their mutual relations and convergence properties are investigated.

The algorithms are applied to power load prediction in part II. They are compared with other prediction algorithms proposed in the literature by computer simulations on authentical load data.

In part III local convergence properties of recursive estimation algorithms are considered. The general results are applied to some earlier proposed recursive algorithms amongst them the Extended Least Squares method and the self-tuning regulator. A new algorithm for estimation of the parameters in an ARMA process is given.

Referat skrivet av
Author

Förslag till ytterligare nyckelord

Klassifikationssystem och -klass(er)

Indextermer (ange källa)
Adaptive System, Automatic Control, Forecasting, Time Series Analysis, Electric Power Demand, Algorithms, Convergence, Least Squares Method. (Thesaurus of Engineering and Scientific Terms, Engineers Joint Council, N.Y., USA)

Omfång
206 pages
Övriga bibliografiska uppgifter
Språk
English

Sekretessuppgifter
ISSN
ISBN

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

Pris

TABLE OF CONTENTS

Adaptive Prediction and Recursive Estimation

The prediction of timeseries is a significant problem in many technical, economical or social situations. It is important for efficient planning. Reordering of supplies to a storage is e.g. based on prediction of future sales. The prediction of future power load is important for power load scheduling. Prediction is also needed in many other applications e.g. in EEG analysis, geophysics or speech communication.

The modeling of the timeseries to be predicted is crucial for the prediction result. A priori knowledge about the physical background of the timeseries can be represented as a deterministic component, e.g. a trend or a period in the model. In this manner one aspect of the nonstationarity of the time series is treated. The discrepancies between this function and the observed data are regarded as random disturbancies. The modeling and prediction thereof are in many cases important parts of the total prediction problem.

The variation with time of the parameters and of the structure of the process description is another aspect of the lack of stationarity. This is illustrated in Figure 1, where the hourly load during a week on a power network is shown for different parts of the year. The periodic structure of the timeseries as well as the changing pattern is clearly illustrated.

In order to predict a nonstationary process one has to adjust the parameters in the process description according to the obtained data. The general exponential smoothing approach to this problem is presented in Brown (1963). In this method a model of the process is given ad hoc.

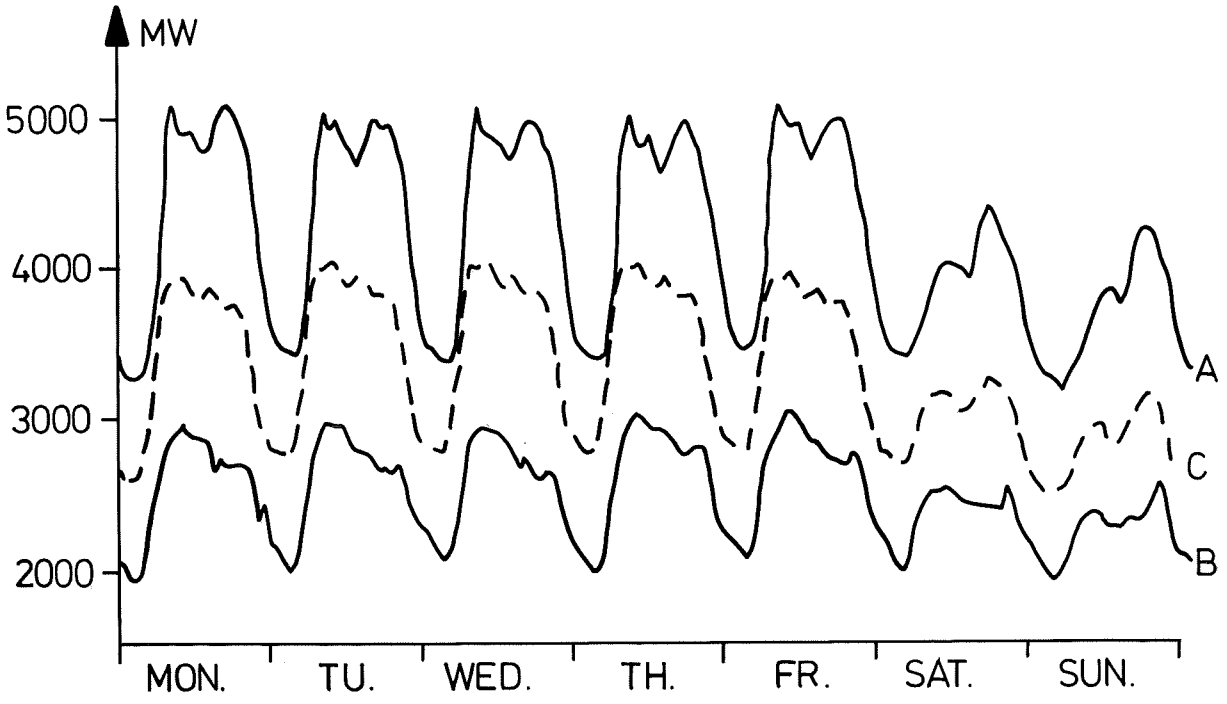


Figure 1 - Hourly power load. A - during a winter week, B - during a summer week, C - mean value over a year.

It contains seasonal and trend components represented with trigonometrical functions and polynomials respectively. The parameters in the model are adjusted at every time step according to a weighted mean value of old prediction errors. A predecessor to this method is the method of exponentially weighted moving averages which is discussed for example in Holt et al (1963) and Coutie (1964).

In Box and Jenkins (1970) a method to handle the trend and the periodic components of the timeseries is proposed. The nonstationary elements in the level description are eliminated through difference calculation, leaving an autoregressive moving average (ARMA) model for the noise component. This latter treatment of nonstationary processes originates from Yaglom, see Yaglom (1955).

The adaptive filtering approach to the problem is discussed e.g. in Mehra (1971). He uses the same technique as Box and Jenkins to the nonstationarities in the process level. The prediction method in Harrison and Stevens (1971) is also based on Kalman filtering but applied on a simple trend model. They use Bayesian principles in order to treat sudden changes in the process description e.g. changes in trend or slope.

A different method for solving the prediction problem is the Group Method of Data Handling (GMDH) proposed by Ivakhnenko and his co-workers, see e.g. Ivakhnenko (1970, 1971). In this approach the unknown process is modeled with a general nonlinear function which is expressed as a Kolmogorov-Gabor polynomial

$$y(t) = f(x_1,\ldots,x_n) = a_0 + \sum_i a_i x_i + \sum_{i,j} a_{ij} x_i x_j + \ldots$$

where $x_i = x_i(t), i=1,\ldots,n$ are input signals. The high dimensionality of the problem due to the large amount of parameters is circumvented by a multilayer regression approach. The adequate relationsships are selected using e.g. a minimum mean square prediction error criterion. The method is applied for prediction in e.g. Sawaragi and Ikeda (1976).

OUTLINE OF THE WORK

In the present report an additive input-output model is used to describe the data sequence $\{y(t)\}$

$$y(t) = n(t) + s(t) \qquad (1)$$

The stochastic term $n(t)$ is supposed to be an ARMA process

$$A(q^{-1})\, n(t) = C(q^{-1})\, w(t) \qquad (2)$$

where q^{-1} is the backward shift operator, $\{w(t)\}$ is a white noise sequence, $C(q^{-1})$ is a stable polynomial and $A(q^{-1})$ and $C(q^{-1})$ are relatively prime. The trend and periodic components are collected in the term $s(t)$.

The thesis consists of three parts:

I. Adaptive Prediction of ARMA Processes

II. Adaptive Short-Term Prediction of Power Load

III. Local Convergence of Some Recursive Estimation Algorithms

The first part treats prediction of an ARMA process when the parameters in the process description are constant but unknown. In the second part some of the methods discussed are extended to cover prediction of the process in (1). They are applied on short-term prediction of power load. In the third part the interest is focussed on the estimation algorithms per se. Conditions for local convergence for a class of estimation algorithms are given.

PART I - ADAPTIVE PREDICTION OF ARMA PROCESSES

The problem of minimum mean square error prediction of an ARMA process with constant but unknown parameters have many features in common with the self-tuning regulator problem. It can be approached by making an estimation of the parameters in a process model. These parameter estimates

are then used to calculate the prediction. Another method is to directly estimate the parameters in a predictor as in the self-tuning regulator case, see e.g. Åström et al (1977). Different predictor representations may be used, leading to different adaptive prediction algorithms. One of these is discussed in Wittenmark (1974) and another in Holst (1974). A similar algorithm is applied to river flow prediction by Kashyap and Rao (1973).

It is shown that if the Extended Least Squares method is used to estimate the parameters in (2), the estimates produced by all the considered algorithms when used for one-step prediction are linearly related. Hence the one-step predictions are the same irrespective of which prediction algorithm is used. When used for k-step prediction there is no such relation between all of the predictor representations. In this case simulation studies show that the method where the process parameters are estimated and the two methods by Wittenmark and by Holst where the parameters in the predictor description are estimated have a similar loss function performance.

The convergence properties of the adaptive predictors are discussed. Conditions for local convergence to the parameters giving minimum mean square error prediction are established. The key condition is expressed in terms of the parameters in the process description and is independent of the prediction horizon.

In certain applications it is necessary to simultaneously predict 1,...,N steps ahead. If an adaptive predictor for each of these predictions is to be used the calculations might be rather time consuming. Simplifications using only the parameters estimated from the one-step predictor are discussed.

PART II - ADAPTIVE SHORT-TERM PREDICTION OF POWER LOAD

The self-tuning predictor has been applied to real data for short-term prediction of the hourly load on the power network. The results are presented in the second part of this thesis. Preliminary results were given in Holst (1974). It has also been applied to prediction of urban sewer flows, see Beck (1977).

In both applications the algorithm has to be extended to handle a timevarying process description and a periodic component in the data series. The timevarying parameters are handled with exponential weighting of the prediction errors in the criterion function. The periodic component is represented by a vector with as many elements as the length of the period. It may be fixed or exponentially updated with the obtained data.

In the power load prediction application different adaptive prediction algorithms have been studied. All the algorithms involve ARMA process prediction as was treated in part I. The results of the predictions by the adaptive k-step predictor compare favourably both with other published methods studied in this part of the thesis and with published prediction results. The obtained prediction results imply that the adaptive prediction algorithms studied are useful and well suited for practical use.

Part III - LOCAL CONVERGENCE OF SOME RECURSIVE ESTIMATION ALGORITHMS

This part is devoted to the convergence problem for estimation algorithms. Results concerning local convergence are obtained. They are applied to some specific algorithms namely the Extended Least Squares method and a modification thereof (see Young (1976)), an algorithm

given in Landau (1976) and the self-tuning regulator.

The analysis is based on Ljung's result concerning convergence of recursive stochastic algorithms, Ljung (1975, 1976). It is shown in these papers that the asymptotic behaviour of the algorithm is described by an ordinary differential equation and that only stable stationary points to it are possible convergence points to the algorithm. The key result in this part of the thesis is the calculation of explicit expressions for the eigenvalues of a matrix occuring when linearizing this differential equation.

As an example of the result, consider the Extended Least Squares method applied on an ARMA process (2). The eigenvalues of the matrix in the differential equation linearized around the true values of the parameters are then

$$-1 \qquad \text{multiplicity } n_c$$

$$-1/C(\alpha_i) \qquad i = 1,\ldots,n_a$$

where n_a and n_c are the orders of the A and C polynomials in (2). α_i, $i=1,\ldots,n_a$ are the solutions to the equation

$$z^{n_a} A(z^{-1}) = 0$$

Thus if any of the numbers $-1/C(\alpha_i)$, $i=1,\ldots,n_a$ has a positive real part the true value of the parameter vector is not a possible convergence point.

Two of the considered algorithms,i.e. the basic and the modified Extended Least Squares method are apt for estimation of parameters in a timeseries modeled as an ARMA process. Through simple modifications in the algorithms and the corresponding differential equations two other algorithms can be derived. One of these is the Recursive Maximum Likelihood method (RML), see e.g. Ljung, Söderström and Gustavsson (1975). Both RML and the second method,

which appears new, have the desirable property that the true value of the parameter vector is always a possible convergence point of the estimation algorithm.

ACKNOWLEDGEMENTS

It is a pleasure for me to express my sincere gratitude to my supervisor Professor Karl Johan Åström. His guidance and many stimulating suggestions have been invaluable.

I also want to thank my colleagues at the Department of Automatic Control and at the Department of Mathematical Statistics for many provoking discussions and valuable comments. In particular, I want to thank Professor Lennart Ljung for the many enlightening discussions on the convergence subject, and Teknologie licentiat Sture Lindahl for his assistance with the power load prediction.

I am most grateful to Civilingenjör Lennart Tyrén at the Swedish State Power Board, who supplied the data used in the power load application.

The manuscripts were excellently typed by Gudrun Christensen, Eva Dagnegård and Dagmar Måhlén, and the figures were skillfully prepared by Britt-Marie Carlsson.

Part of the work was supported by the Swedish Board for Technical Development under contract 74-3476.

REFERENCES

Åström, K J, U Borisson, L Ljung, and B Wittenmark (1977): Theory and Application of Self-Tuning Regulators. To be published in the September issue of Automatica 13. This is an expanded version of a paper given at the 6th IFAC World Congress in Boston, Mass. 1975.

Beck, B (1977): The Identification and Adaptive Prediction of Urban Sewer Flows. Int J Control 25, 425-440.

Box, G E P and G M Jenkins (1970): Time Series Analysis: Forecasting and Control. Holden Day, San Francisco.

Brown, R G (1963): Smoothing, Forecasting and Prediction. Prentice Hall, Englewood Cliffs, N J.

Coutie, G A (1964): Short Term Forecasting. ICI Monograph No. 2, Oliver & Boyd, Edinburgh.

Harrison, P J and C F Stevens (1971): A Bayesian Approach to Short-term Forecasting. Opl Res Quarterly 22, 341-362.

Holst, J (1974): On the Use of Self-tuning Predictors for the Prediction of Power Load (in Swedish). TFRT-3119, Dept of Automatic Control, Lund Institute of Technology, Lund, Sweden.

Holt, C C, F Modigliani, J F Muth, and H A Simon (1963): Planning Production, Inventories and Work Force. Prentice Hall, Englewood Cliffs, N J.

Ivakhnenko, A G (1970): Heuristic Self-Organization in Problems of Engineering Cybernetics. Automatica 6, 207-219.

Ivakhnenko, A G (1971): Polynomial Theory of Complex Systems. IEEE Tr-SMC 1, 364-378.

Kashyap, R L and A R Rao (1973): Real Time Recursive Prediction of River Flows. Automatica 9, 175-183.

Landau, I D (1976): Unbiased Recursive Identification Using Model Reference Adaptive Techniques. IEEE Tr-AC 21, 194-202.

Ljung, L (1975): Theorems for the Asymptotic Analysis of Recursive Stochastic Algorithms. TFRT-3096, Dept of Automatic Control, Lund Institute of Technology, Lund, Sweden.

Ljung, L (1976): Analysis of Recursive Stochastic Algorithms. TFRT-7097, Dept of Automatic Control, Lund Institute of Technology, Lund, Sweden. To be published in IEEE Tr-AC 22, Aug 1977.

Ljung, L ,T Söderström and I Gustavsson (1975): Counter-examples to General Convergence of a Commonly Used Recursive Identification Method. IEEE Tr-AC 20, 643-652.

Mehra, R K (1971): On-Line Identification of Linear Dynamic Systems with Applications to Kalman Filtering. IEEE Tr-AC 16, 12-21.

Sawaragi, Y and S Ikeda (1976): Identification Methods in Environmental Pollution Problems. Paper S-7, Preprints from IV IFAC Symposium. Identification and System Parameter Estimation, Tbilisi, USSR.

Wittenmark, B (1974): A Self-Tuning Predictor. IEEE Tr-AC 19, 848-851.

Yaglom, A M (1955): Correlation Theory of Processes with Random Stationary nth Increments. Mat Sb 37, 141-196.

Young, P C (1976): Some Observations on Instrumental Variable Methods of Time-Series Analysis. Int J Control 23, 593-612.

Part I - Adaptive Prediction of ARMA Processes

ABSTRACT

The adaptive prediction of ARMA processes with constant but unknown parameters is studied. Different algorithms are considered and their mutual relations investigated. It is shown that when the parameter estimation converges the algorithms asymptotically give minimum mean square error prediction of the process. A necessary condition for convergence to the corresponding parameter values is established. It is expressed in terms of the process description and is independent of the prediction horizon.

1. Introduction

The linear minimum mean square error prediction of an ARMA-process is a problem with a wellknown solution when the parameters of the process are known. The predictor can be written in a number of ways using different representations of the history of the process. A number of different predictors can then be generated. Since the parameters of the process are known the resulting prediction, however, will be the same independently of the way the predictor is written.

If the parameters of the process are unknown an adaptive predictor may be used. The prediction can be calculated either via estimation of the parameters in the process model or via estimation of the parameters in the predictor. In the latter case the structure of the predictor can be chosen as one of the predictor versions derived for known parameters.

In this part of the report different adaptive prediction algorithms are analysed and compared. Transient as well as asymptotical properties of the algorithms are discussed. All the proposed algorithms are based on a certainty equivalence principle (see e.g. Wittenmark (1975)), i.e. the unknown parameters are estimated and then used in the predictor as if the estimates were the true values of the parameters. The Least Squares method is used in the identification.

This approach to adaptive prediction has been taken also by Wittenmark (1974). A similar type of algorithm is applied to river flow prediction by Kashyap and Rao (1973). The prediction algorithms are closely related to the algorithms used for self-tuning control, compare for example Åström et al (1977). The adaptive prediction problem is discussed also in Bohlin (1976).

Some of the algorithms which are discussed here are presented elsewhere, for example in Wittenmark (1974) and Holst (1974) resp. for two of the k-step prediction algorithms and in Åström (1974b) and Young (1970) for one of the algorithms in the one-step prediction case.

This part of the report is organized in the following way. In Chapter 2 the minimum mean square error prediction of an ARMA process with known parameters is discussed and different versions of the predictor are presented. The corresponding adaptive prediction algorithms are given in Chapter 3 together with an algorithm where the parameters in the process model are estimated.

In Chapter 4 it is shown that all these predictors, when used for one-step prediction, are related by a change of variables. For three of the algorithms this is so also in the k-step prediction case.

The asymptotical properties of the algorithms are discussed in Chapter 5. The analysis in this chapter is based on the results in Ljung (1975, 1976a). The main idea there is to associate the parameter estimation algorithm with a differential equation that contains all relevant information about the asymptotic behaviour of the algorithm. The connection between this differential equation and the corresponding algorithm that is of interest here is that only stable stationary points to the differential equation are possible convergence points for the algorithm. It is shown that when the parameter estimation converges and the order of the process is known, the unique stationary point corresponds to the minimum mean square error predictor of the process. Furthermore, it is shown that this stationary point to the differential equation is unstable for certain systems. These systems are the same for the considered algorithms and the stability is independent of the number of steps to predict.

In Chapter 6 the prediction of an ARMA process will be

discussed from a slightly different point of view. The aim is to design an adaptive prediction algorithm which predicts 1,...,N steps ahead. Such a predictor is termed an adaptive multistep predictor and it is useful for example in connection with prediction of parts of a period or a whole period in a process which contains periodic elements. In the given predictor the parameters for one--step prediction are estimated and the predictions are then calculated recursively in the number of steps to predict. Three different alternative representations of this predictor are considered. They are based on the adaptive one-step predictors from the previous chapters. Hence they are equivalent for k-step prediction and the convergence results from Chapter 5 apply directly.

Chapter 7 contains numerical examples, where some of the predictors are compared. Finally, in Chapter 8 the algorithms and their properties are summarized and discussed.

In the following the proposed algorithms will be distinguished by an index 0,...,5. The optimal values of e.g. the prediction when the parameters are known will be denoted by the subscript M for minimum. Parameter and polynomial estimates will be denoted by a time argument, such as for example $a_1(t)$ as an estimate at time t of the parameter a_1.

Unless otherwise stated it will be assumed that the order of the process to predict is known.

2. PRELIMINARIES - K-STEP PREDICTORS

In this chapter the linear minimum mean square error prediction of an ARMA process with known parameters will be discussed. The solution of the prediction problem and the properties of the resulting predictor are well known, see e.g. Åström (1970) or Box and Jenkins (1970), and will be briefly reviewed. The main purpose of the chapter is to present different possibilities to express the predictor. Since the parameters in the process are known, the different expressions only give different possibilities to calculate the same prediction value. However, all the predictor versions may be used in an adaptive context when the process parameters are unknown and recursively estimated as will be discussed in the following chapters.

Statement of the problem. Version 1

Consider a stationary stochastic process $\{y(t)\}$ described by the ARMA model

$$A(q^{-1})\, y(t) = C(q^{-1})\, e(t) \tag{2.1}$$

where

$$A(q^{-1}) = 1 + a_1 q^{-1} + \ldots + a_n q^{-n}$$

and similarly for $C(q^{-1})$. The polynomials are supposed to be asymptotically stable and relatively prime. Introduce the reciprocal polynomials $A^f(q)$ and $C^f(q)$ where

$$A^f(q) = q^n A(q^{-1}) = q^n + a_1 q^{n-1} + \ldots + a_n$$

$C^f(q)$ is analogously defined. $\{e(t)\}$ is a sequence of independent random variables with mean value zero and variance σ^2.

If the vectors

$$\varphi_{0,M}(t) = [-y(t-1),\ldots,-y(t-n), e(t-1),\ldots,e(t-n)]^T \quad (2.2a)$$

$$\theta_0 = [a_1,\ldots,a_n,\ c_1,\ldots,c_n]^T \quad (2.2b)$$

are used, (2.1) can be written

$$y(t) = \varphi^T_{0,M}(t)\ \theta_0 + e(t) \quad (2.3)$$

The problem at hand at time t is to find a prediction $\hat{y}_M(t+k|t)$ of $y(t+k)$ based on the available information about $\{y(t)\}$ at time t, i.e. $y(t), y(t-1),\ldots$ such that the criterion

$$V_0 = E[\,\varepsilon^2(t+k)\,] \quad (2.4)$$

is minimized. $\varepsilon(t+k)$ is the k-step prediction error

$$\varepsilon(t+k) = y(t+k) - \hat{y}(t+k|t) \quad (2.5)$$

and $\hat{y}(t+k|t)$ is any linear prediction of $y(t+k)$ based on data up to t.

Denote the smallest σ-field generated by the values of the process $\{y(t)\}$ up to and including the time t by $\mathcal{Y}_t$ = $= F(y(t), y(t-1),\ldots)$ (cf. e.g. Breiman (1968)). The solution of the prediction problem is then the conditional expectation of $y(t+k)$ given $\mathcal{Y}_t$. If this predictor is denoted $\hat{y}_M(t+k|t)$ we have $\hat{y}_M(t+k|t) = E(y(t+k)|\mathcal{Y}_t)$ (cf. Box and Jenkins (1970) or Gikhman and Skorokhod (1969)).

Using the identity (Åström (1970))

$$C(q^{-1}) = A(q^{-1})\ F(q^{-1}) + q^{-k}\ G(q^{-1}) \quad (2.6)$$

where

$$F(q^{-1}) = 1 + f_1 q^{-1} + \ldots + f_{k-1} q^{-k+1}$$

$$G(q^{-1}) = g_0 + g_1 q^{-1} + \ldots + g_{n-1} q^{-n+1}$$

the process model (2.1) can be written

$$y(t+k) = \frac{C(q^{-1})}{A(q^{-1})} e(t+k) = F(q^{-1})\, e(t+k) + q^{-k} \frac{G(q^{-1})}{A(q^{-1})} e(t+k)$$

Here $(e(t+1),\ldots,e(t+k))$ do not belong to the σ-field y_t but $\frac{G}{A}e(t)$ does since it is composed of $y(t), y(t-1),\ldots$. Thus the optimal predictor is

$$\hat{y}_M(t+k|t) = \frac{G(q^{-1})}{A(q^{-1})} e(t) \qquad \text{or}$$

$$\hat{y}_M(t+k|t) = (1-A(q^{-1}))\, \hat{y}_M(t+k|t) + G(q^{-1})\, e(t) \qquad (2.7)$$

This version of the predictor will be referred to as version 1. The predictions are based on the innovations of the ARMA process, which can be calculated from the available measurements by inversion of (2.1). The optimal prediction error is

$$\varepsilon_M(t+k) = F(q^{-1})\, e(t+k) \qquad (2.8)$$

with the variance

$$E\, \varepsilon_M^2(t+k) = \sigma^2 (1 + f_1^2 + \ldots + f_{k-1}^2).$$

The other predictor versions are derived below by applying (2.1), (2.5), (2.6), and (2.8) on the predictor (2.7).

In these predictors just two of the polynomials $A(q^{-1})$, $C(q^{-1})$, $G(q^{-1})$, and $A(q^{-1})F(q^{-1})$ are needed for the k-step prediction. This means that only $2n$, or if the AF-polynomial is used $2n+k-1$, old values of $\hat{y}_M(t+k|t)$, $y(t)$, $\varepsilon_M(t)$ or $e(t)$ must be stored.

Remark. If the independence assumption on $\{e(t)\}$ is replaced by the assumptions that the process is uncorrelated and the predictor linear, the predictor (2.7) remains optimal. The predictor (2.7) is also optimal for unstable polynomials $A(q^{-1})$. □

Version 2

Using the system equation (2.1) the predictor can be expressed as

$$\hat{y}_M(t+k|t) = \frac{G(q^{-1})}{C(q^{-1})} y(t) \quad \text{or}$$

$$\hat{y}_M(t+k|t) = (1-C(q^{-1}))\, \hat{y}_M(t+k|t) + G(q^{-1})\, y(t) \qquad (2.9)$$

This expression for the predictor is given in Åström (1970). It is used in an adaptive context in Holst (1974). The input to the predictor system is in this case the measured values of the given process.

Version 3

Denote the polynomial $A(q^{-1})F(q^{-1})$ by $H(q^{-1})$ with coefficients $h_1,\ldots,h_{n+k-1}$. If the prediction error, (2.5), is used the predictor can be written

$$\hat{y}_M(t+k|t) = \frac{G(q^{-1})}{H(q^{-1})} \varepsilon_M(t) \quad \text{or}$$

$$\hat{y}_M(t+k|t) = (1-H(q^{-1}))\, \hat{y}_M(t+k|t) + G(q^{-1})\, \varepsilon_M(t) \qquad (2.10)$$

In this version of the predictor the prediction error is used as input to the prediction system. It is discussed in Wittenmark (1974). The number of parameters is $2n+k-1$.

Version 4

In version 3 some of the parameters in the $H(q^{-1})$ and $G(q^{-1})$ polynomials are equal due to the identity (2.6). If this is taken into consideration version 4 appears.

In the discussion it is favourable to separate the cases $n \geq k$ and $n < k$. When $n \geq k$ the polynomials H and G in version 3 are partitioned according to

$$H(q^{-1}) = H_1(q^{-1}) + H_2(q^{-1}) = (1+h_1q^{-1} + \ldots + h_nq^{-n}) +$$

$$+ (h_{n+1}\, q^{-n-1} + \ldots + h_{n+k-1}\, q^{-n-k+1})$$

$$G(q^{-1}) = G_1(q^{-1}) + G_2(q^{-1}) = (g_0 + g_1q^{-1} + \ldots + g_{n-k}q^{-n+k}) +$$

$$+ (g_{n-k+1}\, q^{-n+k-1} + \ldots + g_{n-1}\, q^{-n+1}).$$

When $k = 1$, $H_2 = G_2 = 0$. The identity (2.6) gives

$$H_2(q^{-1}) = -q^{-k}\, G_2(q^{-1})$$

i.e. $h_i = -g_{i-k}$; $i = n+1, \ldots, n+k-1$. If this is introduced into (2.10) the following expression for the optimal predictor is obtained:

$$\hat{y}_M(t+k|t) = (1-H)\, \hat{y}_M(t+k|t) + G\, \varepsilon_M(t) =$$

$$= (1-H_1)\, \hat{y}_M(t+k|t) + G_1\, \varepsilon_M(t) + G_2\, y(t) \quad (2.11)$$

Thus only $2n$ parameters are used.

If $n < k$ the polynomial G in version 3 need not to be partitioned and the polynomial H is partitioned into

$$H(q^{-1}) = H_1'(q^{-1}) + H_2'(q^{-1}) + H_3'(q^{-1}) =$$

$$= (1 + h_1 q^{-1} + \ldots + h_n q^{-n}) + (h_{n+1} q^{-n-1} + \ldots + h_{k-1} q^{-k+1}) +$$

$$+ (h_k q^{-k} + \ldots + h_{k+n-1} q^{-n-k+1}) \qquad (2.12)$$

When $n = k - 1$ the polynomial H_2' is not part of the partition. The identity (2.6) gives

$$H_1'(q^{-1}) = C(q^{-1})$$

$$H_2'(q^{-1}) = 0$$

$$H_3'(q^{-1}) = -G(q^{-1}) \cdot q^{-k}$$

i.e.

$$h_i = \begin{cases} c_i & i = 1, \ldots, n \\ 0 & i = n+1, \ldots . k-1 \\ -g_{i-k} & i = k, \ldots, k+n-1 \end{cases}$$

If this is introduced into (2.10) the resulting predictor is

$$\hat{y}_M(t+k|t) = (1-H)\ \hat{y}_M(t+k|t) + G\ \varepsilon_M(t) =$$

$$= (1-C)\ \hat{y}_M(t+k|t) + G\ y(t)$$

Thus when $n < k$ the predictor 3 is transformed into predictor 2. The predictor 4 will thus be discussed only for $k \leq n$, when it can be regarded as a $2n$ parameter variant of predictor 3.

Version 5

A fifth way of expressing the predictor can be derived from (2.9) with the aid of the identity (2.6) and the prediction error:

$$\hat{y}_M(t+k|t) = (1-C)\ \hat{y}_M(t+k|t) + G\ y(t) =$$

$$= (1-H)\ y(t+k) + (C-1)\ \varepsilon_M(t+k) \qquad (2.13)$$

Here the predictor seems to be depending on values of the process and prediction error that are not at hand at the time of prediction. This is however not true as, according to the identity (2.6) the k-1 first coefficients in the polynomial H are equal to the k-1 first coefficients in the polynomial C. To exploit this the polynomials H and C are partitioned. As in the discussion of version 4 it is favourable to separate the cases $n \geq k$ and $n < k$.

Thus when $n \geq k$ the polynomials H and C are partitioned as

$$H(q^{-1}) = H_1''(q^{-1}) + H_2''(q^{-1}) = (1 + h_1 q^{-1} + \ldots + h_{k-1} q^{-k+1}) +$$

$$+ (h_k q^{-k} + \ldots + h_{n+k-1} q^{-n-k+1})$$

and

$$C(q^{-1}) = C_1(q^{-1}) + C_2(q^{-1}) = (1 + c_1 q^{-1} + \ldots + c_{k-1} q^{-k+1}) +$$

$$+ (c_k q^{-k} + \ldots + c_n q^{-n})$$

the predictor (2.13) is equal to

$$\hat{y}_M(t+k|t) = (1-H)\ y(t+k) + (C-1)\ \varepsilon_M(t+k) =$$

$$= -H_2''\ y(t+k) - (C_1-1)\ \hat{y}_M(t+k|t) + C_2\ \varepsilon_M(t+k)$$

(2.14)

If $k = 1$, $H_1'' = C_1 = 1$.
This can be regarded as a generalization to k-step prediction of the one-step predictor used e.g. in Åström (1974b) for parameter identification.

When $n < k$ the polynomial C need not be partitioned and the polynomial H can be partitioned as in version 4 which gives

$$\hat{y}_M(t+k|t) = (1-C)\ \hat{y}_M(t+k|t) + G\ y(t)$$

i.e. version 2 of the predictor. Version 5 of the predictor is thus used only when $k \leq n$.

Summary

All the versions of the predictor can be expressed as

$$\hat{y}_M(t+k|t) = \varphi_{i,M}^T(t+k)\ \theta_i \qquad i = 1,\ldots,5 \qquad (2.15)$$

where θ_i and $\varphi_{i,M}$ are two columnvectors with parameters and data respectively. They are given in Table 1. There are 2n elements in the vectors except for version 3, where the old prediction errors are used in the calculations. In that case the vectors contain $2n+k-1$ elements. For version 4 and 5 only the case $k \leq n$ is considered. When $k = 1$ there are no elements of y in the datavector $\varphi_{4,M}(t+k)$ and no elements of $\hat{y}_M$ in $\varphi_{5,M}(t+k)$.

Using the expression (2.15) for the predictor, (2.5) and (2.8) give

$$\varepsilon_M(t+k) = F\ e(t+k) = y(t+k) - \varphi_{i,M}^T(t+k)\,\theta_i \quad i=1,\ldots,5 \quad (2.16)$$

Table 1 - The Data and Parameter Vectors

Equation	Version	
(2.7)	1	$\theta_1 = (a_1,\ldots,a_n,g_0,\ldots,g_{n-1})^T$
		$\varphi_{1,M}(t+k) = (-\hat{y}_M(t+k-1\|t-1),\ldots,-\hat{y}_M(t+k-n\|t-n),e(t),..$
		$..,e(t-n+1))^T$
(2.9)	2	$\theta_2 = (c_1,\ldots,c_n,g_0,\ldots,g_{n-1})^T$
		$\varphi_{2,M}(t+k) = (-\hat{y}_M(t+k-1\|t-1),\ldots,-\hat{y}_M(t+k-n\|t-n),y(t),..$
		$..,y(t-n+1))^T$
(2.10)	3	$\theta_3 = (h_1,\ldots,h_{n+k-1},g_0,\ldots,g_{n-1})^T$
		$\varphi_{3,M}(t+k) = (-\hat{y}_M(t+k-1\|t-1),\ldots,\hat{y}_M(t-n+1\|t-n-k+1),$
		$\varepsilon_M(t),\ldots,\varepsilon_M(t-n+1))^T$
(2.11)	4	$\theta_4 = (h_1,\ldots,h_n,g_0,\ldots,g_{n-k},g_{n-k+1},\ldots,g_{n-1})^T$
		$\varphi_{4,M}(t+k) = (-\hat{y}_M(t+k-1\|t-1),\ldots,-\hat{y}_M(t+k-n\|t-n),\varepsilon_M(t),..$
		$..,\varepsilon_M(t-n+k),y(t-n+k-1),\ldots,y(t-n+1))^T$
(2.14)	5	$\theta_5 = (h_k,\ldots,h_{k+n-1},c_1,\ldots,c_{k-1},c_k,\ldots,c_n)^T$
		$\varphi_{5,M}(t+k) = (-y(t),\ldots,-y(t-n+1),-\hat{y}_M(t+k-1\|t-1),..$
		$..,-\hat{y}_M(t+1\|t-k+1),\varepsilon_M(t),\ldots,\varepsilon_M(t+k-n))^T$

Multistep Predictors

The predictor versions discussed are the only ones where two polynomials and the values of $\hat{y}_M(t+k|t)$, $y(t)$, $\varepsilon_M(t)$ or $e(t)$ are used. Another possibility to do the k-step prediction is to use the expression (2.1) with time argument $t+k$ or any one-step predictor of $y(t+k)$ and make a conditioning with respect to y_t. If this is done on (2.1) the resulting predictor for $k \leq n$ is (cf. e.g. Akaike (1974) or Box and Jenkins (1970))

$$\hat{y}_M(t+k|t) + \ldots + a_{k-1}\, \hat{y}_M(t+1|t) + a_k\, y(t) + \ldots + a_n\, y(t+k-n) =$$

$$= c_k\, e(t) + \ldots + c_n\, e(t+k-n) \qquad (2.17)$$

This is termed multistep prediction and is treated in Chapter 6. If this predictor is used in order to calculate the k-step predictor also the $k-1, k-2, \ldots, 1$ step predictors must be calculated. If these prediction values are not used, the calculation effort might be unnecessarily large. But if there is a need for prediction $1, \ldots, N$ steps ahead (2.17) might be an efficient way of doing the calculations. The application to power load prediction, where there is a need for profile prediction, is discussed in part II of this work.

3. ADAPTIVE K-STEP PREDICTORS

One way of solving the prediction problem when the parameters of the process (2.1) are unknown is to estimate the parameters in the A and C polynomials and use the estimates in the calculation of the predictor. If this is done recursively it means that the predictor polynomials must be calculated at every time step, which might be a rather heavy computational burden.

Another possibility is to look for adaptive schemes corresponding to the predictor versions discussed above, i.e. to estimate the parameters in the predictor instead of in the process model. In the below proposed adaptive prediction algorithms of this kind the Least Squares (LS) method is used for parameter identification. An important reason for this is that it needs a comparatively small amount of computations. Hence the adaptive predictor with the LS method will be well suited for practical real time use.

3.1 ESTIMATION OF THE PARAMETERS IN THE PROCESS MODEL

The LS method for parameter identification can not be used directly for identification of the parameters in the process model (2.1) because of the MA part of the model. A straightforward application would give biased parameter estimates, see e.g. Åström and Eykhoff (1971). In this reference a variety of parameter estimation schemes are presented which circumvent this problem. One of these is the Extended Least Squares method (Panuska (1969), Young (1970), Åström (1974b)), which will be discussed here. When this method is used the noise sequence $e(t), e(t-1), \ldots$ is estimated as a sequence of one-step ahead prediction errors $e_0(t), e_0(t-1), \ldots$ where $e_0(t)$ is calculated from the equation

$$e_0(t) = y(t) - \varphi_0^T(t)\ \theta_0(t-1) \tag{3.1}$$

$\varphi_0(t)$ and $\theta_0(t)$ are the data and parameter estimate vectors at time t. These are defined by

$$\theta_0(t) = \left(a_1(t),\ldots,a_n(t), c_1(t),\ldots,c_n(t)\right)^T$$

$$\varphi_0(t) = \left(-y(t-1),\ldots,-y(t-n), e_0(t-1),\ldots, e_0(t-n)\right)^T \tag{3.2}$$

The method is also discussed in Ljung, Söderström and Gustavsson (1975). The obtained parameter estimates are used to calculate the predictor parameters from the identity (2.6).

3.2 ESTIMATION OF THE PARAMETERS IN THE PREDICTOR

When the parameters in the predictor are unknown adaptive schemes based on predictor versions 1,...,5 can be constructed. The parameters are recursively estimated and the current estimates are used for prediction. The algorithms are given the same numbers as the corresponding versions of the predictor.

Consider as an example the adaptive prediction algorithm corresponding to version 2 of the predictor, where the prediction is based upon the measured data. The discussion may be duplicated for the other versions after appropriate reindexing.

The data and parameter vectors in algorithm 2 are

$$\varphi_2(t+k) = \left(-\hat{y}(t+k-1|t-1),\ldots,-\hat{y}(t+k-n|t-n), y(t),\ldots,y(t-n+1)\right)^T$$

$$\theta_2(t) = \left(c_1(t),\ldots,c_n(t), g_0(t),\ldots, g_{n-1}(t)\right)^T$$

Consider the process model (2.1) and the identity (2.6)

$$y(t) = \frac{C}{A}\, e(t) = F\, e(t) + \frac{G}{C}\, y(t-k) \tag{3.3}$$

Introduce the notation

$$w(t) = F\ e(t)$$

Equation (3.3) then may be written as

$$\varepsilon(t) = w(t) + \frac{1}{C}\left(\varphi_2^T(t)\ \theta_2 - \hat{y}(t|t-k)\right) \tag{3.4}$$

where $\{\hat{y}(t|t-k)\}$ could be any sequence of predictions and $\{\varepsilon(t)\}$ the corresponding prediction errors. When the parameters of the process are known, i.e. θ_2 is known, (3.4) is reduced to (2.8) if $\hat{y}(t|t-k)$ is chosen as the minimum mean square error predictor, i.e. $\hat{y}(t|t-k) = \hat{y}_M(t|t-k)$. The elements of $\varphi_2(t) = \varphi_{2,M}(t)$ are then by construction uncorrelated with the prediction error.

Now suppose that the parameters are unknown. Consider first pure AR processes, i.e. $C \equiv 1$. Then (3.4) reads (after appropriate redefinition of φ_2 and θ_2)

$$\varepsilon(t) = w(t) + \left(\varphi_2^T(t)\ \theta_2 - \hat{y}(t|t-k)\right)$$

or

$$y(t) = \varphi_2^T(t)\ \theta_2 + w(t) \tag{3.5}$$

Since $w(t)$ is uncorrelated with the elements of $\varphi_2(t)$ the application of the LS method for estimation of the parameters in θ_2 gives asymptotically consistent estimates. Next, consider processes with $C \neq 1$. If the parameters are estimated in the model (3.5) with full φ_2 and θ_2 vectors, then (3.4), which describes the process in this case, indicates that the estimates of the parameters would be biased since the elements in φ_2 are correlated with the equation error in (3.4). However, if the parameter estimates converge and if $\hat{y}(t+k|t)$ is chosen as

$$\hat{y}(t+k|t) = \varphi_2^T(t+k)\ \theta_2(t) \tag{3.6}$$

then one possible convergence point in the parameter estimation is the minimum mean square error predictor corresponding to the absolute minimum of the loss function $V(\eta) = \Sigma\left(y(t) - \varphi_2^T(t)\eta\right)^2$ (which is used in the LS determination of the parameters). The correlation between φ_2 and the prediction error will consequently asymptotically be reduced to zero when the parameters converge to this point. It will be shown in Chapter 5 that it in fact is the only possible convergence point.

Algorithms

The discussion, valid for the adaptive prediction algorithms where the parameters in the predictor are estimated, thus gives the following algorithm performed at time t.

- Estimate the parameters θ_i $i = 1,\ldots,5$ in the model

$$y(t) - \varphi_i^T(t)\ \theta_i = \varepsilon(t) + \hat{y}(t|t-k) - \varphi_i^T(t)\ \theta_i = w_i(t) \qquad (3.7)$$

  giving the estimates $\theta_i(t)$, $i = 1,\ldots,5$.

- Use the obtained parameter estimates to calculate the predicted value of the process at time $t+k$.

$$\hat{y}(t+k|t) = \varphi_i^T(t+k)\ \theta_i(t) \qquad (3.8)$$

Note that $\varphi_i(t)$ $i=1,\ldots,5$ is a function of all previous estimates since the predictions and the prediction errors depend on all these estimates.

Since the LS method is used, the estimates at time t are given by the normal equations

$$\sum^N \varphi_i(t)\left(y(t) - \varphi_i^T(t)\ \theta_i\right) = 0 \qquad i = 1,\ldots,5 \qquad (3.9)$$

They satisfy the following recursive equations (Åström (1968), Söderström, Ljung and Gustavsson (1974)). These equations also describe the parameter estimation in the adaptive predictor where the parameters in the process model are estimated, i.e. in algorithm 0. In that case $\hat{w}_0(t) = e_0(t)$ (3.1).

$$\begin{cases} \theta_i(t) = \theta_i(t-1) + K_i(t)\hat{w}_i(t) = \theta_i(t-1) + P_i(t)\varphi_i(t)\hat{w}_i(t) \\ \hat{w}_i(t) = y(t) - \varphi_i^T(t)\theta_i(t-1) \\ K_i(t) = \dfrac{P_i(t-1)\varphi_i(t)}{1+\varphi_i^T(t)P_i(t-1)\varphi_i(t)} \qquad i=0,\ldots,5 \\ P_i(t) = P_i(t-1) - \dfrac{P_i(t-1)\varphi_i(t)\varphi_i^T(t)P_i(t-1)}{1+\varphi_i^T(t)P_i(t-1)\varphi_i(t)} \end{cases} \tag{3.10}$$

where

$$P_i(t) = \left[\sum_{t_0}^{t} \varphi_i(s)\varphi_i^T(s) \right]^{-1} \qquad i=0,\ldots,5 \tag{3.11}$$

For part of the analysis in Chapter 5 an alternative description of the algorithm may be given (cf. Söderström, Ljung and Gustavsson (1974)). Introduce the matrices

$$R_i(t) = 1/t\ P_i(t)^{-1}$$

$$\tilde{K}_i(t) = t\ K_i(t)$$

Then the algorithm (3.10) becomes for $i = 0,\ldots,5$

$$
\begin{cases}
\theta_i(t) = \theta_i(t-1) + \frac{1}{t}\,\tilde{K}_i(t)\hat{w}_i(t) = \theta_i(t-1) + \frac{1}{t}\,R_i^{-1}(t)\varphi_i(t)\hat{w}_i(t) \\
\hat{w}_i(t) = y(t) - \varphi_i^T(t)\theta_i(t-1) \\
\tilde{K}_i(t) = \dfrac{R_i^{-1}(t-1)\varphi_i(t)}{1 + \frac{1}{t}\left(\varphi_i^T(t)R_i^{-1}(t-1)\varphi_i(t) - 1\right)} \\
R_i(t) = R_i(t-1) + \frac{1}{t}\left(\varphi_i(t)\varphi_i^T(t) - R_i(t-1)\right)
\end{cases}
\tag{3.12}
$$

In algorithm 1, where the corresponding predictor with known parameters is based on the innovations of the process, an estimate $e_1(t)$ of the element $e(t)$ in the noise sequence is needed. It is calculated at time t from the equation

$$
e_1(t) = \frac{A(t-1;q^{-1})}{C(t-1;q^{-1})}\; y(t)
$$

In this calculation the earlier obtained values of $e_1(t-1)$, $e_1(t-2),\ldots$ are used as initial values. $A(t-1;q^{-1})$ is the estimate at time $t-1$ of the polynomial A. $C(t-1;q^{-1})$ is an estimate of the polynomial C calculated from the identity (2.6) applied to the polynomial estimates $A(t-1;q^{-1})$ and $G(t-1;q^{-1})$ i.e. from the system of equations

$$
C(t-1;q^{-1}) = A(t-1;q^{-1})F(t-1;q^{-1}) + q^{-k}\,G(t-1;q^{-1}) \tag{3.13}
$$

For $k=1$ $F(t-1;q^{-1})$ is equal to 1 and $C(t-1;q^{-1})$ is determined simply by adding $A(t-1;q^{-1})$ and $q^{-1}G(t-1;q^{-1})$. However, when $k > 1$ it is easily seen that the $C(t-1;q^{-1})$ and $F(t-1;q^{-1})$ polynomials are uniquely determined from this system of equations if and only if $a_n(t-1) \neq 0$. This implies that the algorithm might be numerically ill conditioned when $a_n(t-1)$ is small.

The data and parameter vectors for the algorithms are collected in Table 2.

Table 2 - Data and Parameter Vectors Used in the Adaptive Prediction Algorithms

Algorithm	Vectors
0	$\theta_0(t) = \left(a_1(t),\ldots,a_n(t), c_1(t),\ldots,c_n(t)\right)^T$
	$\varphi_0(t) = \left(-y(t-1),\ldots,-y(t-n), e_0(t-1),\ldots,e_0(t-n)\right)^T$
1	$\theta_1(t) = \left(a_1(t),\ldots,a_n(t), g_0(t),\ldots,g_{n-1}(t)\right)^T$
	$\varphi_1(t+k) = \left(-\hat{y}(t+k-1\|t-1),\ldots,-\hat{y}(t+k-n\|t-n), e_1(t),\ldots,e_1(t-n+1)\right)^T$
2	$\theta_2(t) = \left(c_1(t),\ldots,c_n(t), g_0(t),\ldots,g_{n-1}(t)\right)^T$
	$\varphi_2(t+k) = \left(-\hat{y}(t+k-1\|t-1),\ldots,-\hat{y}(t+k-n\|t-n), y(t),\ldots,y(t-n+1)\right)^T$
3	$\theta_3(t) = \left(h_1(t),\ldots,h_{n+k-1}(t), g_0(t),\ldots,g_{n-1}(t)\right)^T$
	$\varphi_3(t+k) = \left(-\hat{y}(t+k-1\|t-1),\ldots,-\hat{y}(t-n+1\|t-n-k+1), \varepsilon(t),\ldots,\varepsilon(t-n+1)\right)^T$
4	$\theta_4(t) = \left(h_1(t),\ldots,h_n(t), g_0(t),\ldots,g_{n-k}(t), g_{n-k+1}(t),\ldots,g_{n-1}(t)\right)^T$
	$\varphi_4(t+k) = \left(-\hat{y}(t+k-1\|t-1),\ldots,-\hat{y}(t+k-n\|t-n), \varepsilon(t),\ldots,\varepsilon(t+k-n),\right.$ $\left. y(t+k-n-1),\ldots,y(t-n+1)\right)^T$
5	$\theta_5(t) = \left(h_k(t),\ldots,h_{n+k-1}(t), c_1(t),\ldots,c_{k-1}(t), c_k(t),\ldots,c_n(t)\right)^T$
	$\varphi_5(t+k) = \left(-y(t),\ldots,-y(t-n+1), -\hat{y}(t+k-1\|t-1),\ldots,-\hat{y}(t+1\|t-k+1),\right.$ $\left.\varepsilon(t),\ldots,\varepsilon(t+k-n)\right)^T$

3.3 SOME FURTHER COMMENTS ON THE ALGORITHMS

It has been shown in Doncarli (1977) that the transient properties of the ELS algorithm might be considerably affected by the age of the parameter estimates entering in the calculation of the residual. From this point of view it would be preferable e.g. to calculate the $\{e_1(t)\}$ sequence in the algorithm 1 from

$$e_1(t) = \frac{A(t;q^{-1})}{C(t;q^{-1})} \quad y(t)$$

i.e. to use the obtained estimates at time t and the earlier obtained values $e_1(t-1)$ etc. in the calculations. The asymptotical properties of such variants of the algorithms will however not differ from the asymptotical properties of the given algorithms, since the age of the parameter estimates entering into the calculations is unimportant in the asymptotical analysis, cf. Chapter 4.

4. RELATIONS BETWEEN THE ALGORITHMS

In this chapter some relations between the proposed algorithms for solving the adaptive prediction problem are established. It is shown in Section 4.1 that the algorithms 0, 1, 2 and 3 in fact are equivalent when used for one-step prediction. In Section 4.2, in which k-step prediction is discussed, the algorithms 2, 4 and 5 are shown to be equivalent for all k. It is then finally shown that it is not possible to extend the equivalence for k=1 between the algorithms 0, 1, 2 and 3 to general k.

4.1. ONE-STEP PREDICTION

Consider the algorithms presented in Chapter 3 with $k = 1$.

Theorem 1. Consider one-step prediction of an ARMA process described by the model (2.1) using any of the proposed algorithms 0,1,2,3. There exist constant matrices S_{ij} such that if

$$\theta_i(t_0) = S_{ij}\theta_j(t_0) \qquad i = 0,1,2,3;\ j = 0,1,2,3$$

and

$$P_i(t_0) = S_{ij}P_j(t_0)S_{ij}^T \qquad i = 0,1,2,3;\ j = 0,1,2,3$$

for some t_0 and the initial values of the processes $\{y(t)\}$ and $\{\varepsilon(t)\}$ are the same for the considered algorithms, then

$$\theta_i(t) = S_{ij}\theta_j(t) \quad t \geq t_0 \qquad i = 0,1,2,3;\ j = 0,1,2,3$$

when the same realization of $\{y(t), t \geq t_0\}$ is used in the algorithms.

Proof: The proof is based on direct comparisons of data and parameter vectors for the different algorithms. It is found in Appendix A.

□

Remark 1. Since this theorem states that these four algorithms are equivalent when k=1 also the convergence properties of the algorithms are the same. It is shown in Ljung, Söderström and Gustavsson (1975) and in Ljung and Wittenmark (1974) that the algorithms 0 and 3 respectively do not converge for all systems. This result is now possible to apply on algorithms 1 and 2 too. This means that the systems for which the algorithms do not give converging parameter estimates are the same for all the algorithms. The convergence properties are further discussed in Chapter 5.

□

Remark 2. The proof of Theorem 1 thus shows that the algorithms for one step ahead adaptive prediction are related via a change of variables.

□

4.2. K-STEP PREDICTION

For k > 1 the comparison between the algorithms gets more complicated. The reason for this is that the data vectors used in the determination of the parameter estimates and in the prediction are not in all cases linearly related as was the case when the algorithms were used for one step ahead prediction. It is, however, possible to establish a linear connection between the algorithms 2, 4, 5 in the same manner as in Section 4.1.

Theorem 2. Consider k-step prediction of an ARMA process described by the model (2.1) using any of the proposed algorithms 2, 4 or 5. There exist constant matrices Q_{ij} such that if

$$\theta_i(t_0) = Q_{ij}\theta_j(t_0) \qquad i = 2,4,5;\ j = 2,4,5$$

$$P_i(t_0) = Q_{ij}P_j(t_0)Q_{ij}^T \qquad i = 2,4,5;\ j = 2,4,5$$

for some t_0, and the initial values of the processes $\{y(t)\}$ and $\{\varepsilon(t)\}$ are the same for the considered algorithms, then

$$\theta_i(t) = Q_{ij}\theta_j(t) \qquad t \geq t_0 \qquad i = 2,4,5;\ j = 2,4,5$$

when the same realization of $\{y(t), t \geq t_0\}$ is used in the algorithms.

Proof: This theorem is proven in the same manner as Theorem 1. The proof is given in Appendix A.

□

No linear relationship can be established between algorithm 0 and any of the other algorithms 1, 2 or 3 since the data and parameter vectors are not linearly related. Similarly it is seen that no linear transformation exists between algorithm 1 and any of the other two algorithms 2 or 3. Finally, consider the algorithms 2 and 3 or equivalently, due to Theorem 2, algorithms 3 and 4. As was remarked in Chapter 2 algorithm 4 could be regarded as a 2n parameter version of algorithm 3 when the parameters in the process (2.1) were known. When the parameters are unknown it is not possible to find a linear regular transformation between the two data and parameter vectors since the number of parameters to estimate differs. The problem is then if there exists a linear transformation, which is not bijective and does not depend on data or time, of the data and parameter vectors from algorithm 3

such that the algorithm for the transformed parameters coincides with the algorithm for the parameters from algorithm 4.

Theorem 3. Consider k-step prediction of an ARMA process described by the model (2.1) using any of the algorithms 2, 4 or 5. Then there exists no linear data- and timeinvariant transformation between the parameter estimates obtained from these algorithms and the algorithm 3.

Proof: The proof is given in Appendix A. □

Hence, for $k > 1$ the adaptive k-step prediction algorithms 0, 1, 2 and 3 are not linearly related.

5. CONVERGENCE PROPERTIES OF THE ADAPTIVE PREDICTION ALGORITHMS

In this chapter the convergence problem for the adaptive predictors will be treated. Since it was shown in Chapter 4 that algorithms 2, 4 and 5 are equivalent only algorithms 0, 1, 2 and 3 are considered.

The convergence problem will be separated into two subproblems. Firstly the possible convergence points for the algorithms will be derived and secondly local convergence to these points will be discussed. The parameter estimates given by any of the four algorithms do, however, not always converge. This is reported in Ljung, Söderström and Gustavsson (1975) for the ELS algorithm, i.e. algorithm 0 and in Ljung and Wittenmark (1974) for the algorithm 3. For the remaining two algorithms it follows from Chapter 4.

The convergence results are based on the method for analysis of recursive stochastic algorithms derived by Ljung, cf Ljung (1975, 1976a). This theoretical background is shortly reviewed in Section 5.1 and it is shown that the theory can indeed be applied to the adaptive prediction algorithms.

The question of possible convergence points is treated in Section 5.2. It is shown in Ljung, Söderström and Gustavsson (1975) that when the ELS method is applied to an ARMA process there is one possible convergence point for the parameter estimates, the true values of the process parameters. This means that the corresponding predictor converges to the minimum mean square error predictor if the parameter estimates converge. Hence, in the following discussion on the first subproblem only the adaptive prediction algorithms where the parameters in the predictor are estimated will be treated.

In Section 5.3 the local convergence to these points is fur-

ther discussed. A necessary condition valid for the algorithms 0, 2 and 3 is derived. This result is based on a theorem given in part III of this report.

5.1. THEORETICAL BACKGROUND

In the theory by Ljung the main idea is to associate the parameter estimation algorithm with a differential equation that contains the information on the asymptotic behaviour of the stochastic algorithm that is relevant to the current problem. The aim of this section is to calculate the differential equations, which correspond to the prediction algorithms.

The parameters are estimated using the equation (3.12), i.e. for $i = 0,\ldots,3$

$$\begin{cases} \theta_i(t) = \theta_i(t-1) + \frac{1}{t} \cdot \frac{1}{1 + \frac{1}{t}\left(\varphi_i^T(t) R_i^{-1}(t-1)\varphi_i(t) - 1\right)} \cdot \\ \qquad \cdot R_i^{-1}(t-1)\varphi_i(t)\hat{w}_i(t) \\ \hat{w}_i(t) = y(t) - \varphi_i^T(t)\theta_i(t-1) = \\ \qquad = \varepsilon(t) + \varphi_i^T(t)\left[\theta_i(t-k) - \theta_i(t-1)\right] \\ R_i(t) = R_i(t-1) + \frac{1}{t}\left[\varphi_i(t)\varphi_i^T(t) - R_i(t-1)\right] \end{cases}$$

Following the lines in Ljung (1975, 1976a), introduce the stationary stochastic processes

$$\hat{y}(t|t-k;\bar{\theta}_i), \quad \varepsilon(t,\bar{\theta}_i), \quad e_i(t,\bar{\theta}_i), \quad \varphi_i(t,\bar{\theta}_i); \quad i = 0,\ldots,3 \qquad (5.1)$$

which are defined as above with $\bar{\theta}_i$ a constant value of the parameter vector. For example

$$\varphi_0(t,\bar{\theta}_0) = [-y(t-1),\ldots,-y(t-n),$$

$$e_0(t-1,\bar{\theta}_0),\ldots,e_0(t-n,\bar{\theta}_0)]^T$$

where

$$e_0(t,\bar{\theta}_0) = y(t) - \varphi_0^T(t,\bar{\theta}_0)\bar{\theta}_0$$

Note that these processes (5.1) are defined only for such parameter values that guarantee the defining relations to be asymptotically stable. Denote the set of all such $\bar{\theta}_i$ values $D_{S,i}$. For example, in the algorithms 0 (i.e. the ELS algorithm) and 2 this set is

$$D_{S,i} = \{\bar{\theta}_i \,|\, z^n + \bar{c}_1 z^{n-1} + \ldots + \bar{c}_n = 0 \Rightarrow |z| < 1\}$$

$$i = 0,\ 2 \qquad (5.2)$$

Introduce the functions

$$f_i(\bar{\theta}_i) = E\ \varphi_i(t,\bar{\theta}_i)\varepsilon(t,\bar{\theta}_i); \quad i = 0,\ldots,3 \qquad (5.3a)$$

$$G_i(\bar{\theta}_i) = E\ \varphi_i(t,\bar{\theta}_i)\varphi_i^T(t,\bar{\theta}_i); \quad i = 0,\ldots,3 \qquad (5.3b)$$

where the expectation is taken over the distribution of $\{e(t)\}$. The set of differential equations associated with the algorithm (3.12) is then

$$\frac{d}{d\tau}\,\theta_{i,D}(\tau) = R_{i,D}^{-1}(\tau) f_i\big(\theta_{i,D}(\tau)\big); \quad i = 0,\ldots,3 \qquad (5.4a)$$

$$\frac{d}{d\tau}\,R_{i,D}(\tau) = G_i\big(\theta_{i,D}(\tau)\big) - R_{i,D}(\tau); i = 0,\ldots,3 \qquad (5.4b)$$

where the solution to the differential equation is denoted

by subscript D. Then the connection between these differential equations and the algorithm (3.12) which will be exploited in the following is given in the following theorem. The convergence point is denoted by (θ_i^*, R_i^*) $i = 0, \ldots, 3$. The ELS case is covered in Ljung (1976a).

Theorem 4. Given an ARMA process according to (2.1) where the noise $\{e(t)\}$ is supposed to be a stationary sequence of independent stochastic variables with $E|e(t)|^p$ existing for each $p > 1$. Suppose that $(\theta_i(t), R_i(t))$ is generated by the algorithm (3.12) and that $(\theta_i(t), R_i(t))$ tends to (θ_i^*, R_i^*) with a strictly positive probability, $i = 1, \ldots, 3$. Then

$$f_i(\theta_i^*) = 0; \quad R_i^* = G_i(\theta_i^*) \qquad i = 1, \ldots, 3 \tag{5.5}$$

and all eigenvalues of

$$G_i^{-1}(\theta_i^*) \frac{d}{d\theta_i} f_i(\theta_i)\Big|_{\theta_i=\theta_i^*} \qquad i = 1, \ldots, 3 \tag{5.6}$$

where $G_i(\theta_i^*)$ is supposed to be regular, have nonpositive real parts.

Proof: The proof of this theorem consists of an application a of a theorem by Ljung (Ljung (1975)). It is given in Appendix B.

□

Remark 1. The first part of the theorem thus states that the only possible convergence points of the algorithm are the stationary points to the associated differential equation. The second part of the theorem says that only such stationary points to the differential equation that are associated with a stable linearization are possible convergence points of the algorithm.

□

Remark 2. This theorem thus shows that when using the adaptive prediction algorithm $\varepsilon(t,\theta^*)$ and $\varphi_i(t,\theta^*)$ are orthogonal, as is also the case when the minimum mean square error predictor is used with known process parameters. This common property of the adaptive algorithm and the minimum mean square error predictor will be of importance when showing that the adaptive predictor in fact converges to the minimum mean square error predictor.

□

Remark 3. With $\varepsilon^*(t) = \varepsilon(t;\theta_i^*)$, $\hat{y}^*(t|t-k) = \hat{y}(t|t-k;\theta_i^*)$ and $e_1^*(t) = e_1(t;\theta_1^*)$ the interpretation of this theorem for the algorithms 1 to 3 is given below.

1: $E\ \varepsilon^*(t+\tau)\hat{y}^*(t+k|t) = r_{\varepsilon^*\hat{y}^*}(\tau) = 0$

$\tau = k+1,\ldots,k+n$

$E\ \varepsilon^*(t+\tau)e_1^*(t) = r_{\varepsilon^* e_1^*}(\tau) = 0$

$\tau = k,\ldots,k+n-1$

2: $E\ \varepsilon^*(t+\tau)\hat{\Delta}\hat{y}^*(t+k|t) = r_{\varepsilon^*\hat{y}^*}(\tau) = 0$

$\tau = k+1,\ldots,k+n$

$E\ \varepsilon^*(t+\tau)y(t) = r_{\varepsilon^* y}(\tau) = 0$

$\tau = k,\ldots,k+n-1$

3: $E\ \varepsilon^*(t+\tau)\hat{y}^*(t+k|t) = r_{\varepsilon^*\hat{y}^*}(\tau) = 0$

$\tau = k+1,\ldots,2k+n-1$

$E\ \varepsilon^*(t+\tau)\varepsilon^*(t) = r_{\varepsilon^*}(\tau) = 0$

$\tau = k,\ldots,k+n-1$

From the predictor equation it then follows for all of the proposed structures that

$$r_{\varepsilon^* \hat{y}^*}(k) = 0$$

□

Remark 4. The result is valid even if the order of the process (2.1) is unknown.

□

Remark 5. Note the similarities between the normal equations (3.9) and equation (5.5).

□

5.2. CONVERGENCE POINTS

In this section the application of the first part of Theorem 4 to the algorithms where the parameters in the predictor are estimated will be discussed. The fact that the data vector $\varphi_i(t)$, $i = 1,\ldots,3$ and the prediction error $\varepsilon(t)$ are asymptotically uncorrelated for certain lags when the parameter estimates converge will be used to show that the only convergence point when the order of the system is known is the minimum mean square error predictor. This result strongly resembles the discussion in Åström and Wittenmark (1973) concerning the asymptotical properties of the self-tuning regulator. The result has also been derived in Wittenmark (1974) for algorithm 3.

Lemma 5. Given an ARMA process according to (2.1) with known order n. Suppose that it is predicted by any of the adaptive predictors 1, 2 or 3 with the constant predictor parameters θ_i^*, $i = 1,\ldots,3$, where

$$f_i(\theta_i^*) = 0; \quad i = 1,\ldots,3 \qquad \text{(cf. (5.5))}$$

Assume that the polynomials A^* and G^* for the algorithm 1, C^* and G^* for the algorithm 2 and G^* and H^* for the algorithm 3 have no factors in common. These polynomials are formed from the elements in θ_i^* as in Chapter 2.

The prediction process is then the minimum mean square error prediction of (2.1). Moreover, if one of the algorithms 2 or 3 is used the parameters in the predictor are the parameters in the minimum mean square error predictor of (2.1).

Proof: The proof is given in Appendix C. □

Remark. Although the predictions given by algorithm 1 with $\theta_1 = \theta_1^*$ are minimum mean square error predictions, the parameters in the predictors might be different. There might be factors which cancel in the polynomials C and G and in C^* and G^* but these factors do not have to be the same. This implies that $C^* \neq C$ and $G^* \neq G$. However, if C and G have no factors in common then it follows from the proof of the Lemma that $C^* = C$ and $G^* = G$. Still it is only possible to prove that $H^* = H$, i.e. $A^*F^* = AF$.

An exception from this discussion is the one-step prediction. Then $F = F^* = 1$ which gives $e_1(t) = e(t)$. (C.4) gives

$$1 - \frac{G^*}{C^*} q^{-1} = \frac{A}{C}$$

The algorithm gives

$$1 - \frac{G^*}{C^*} q^{-1} = \frac{A^*}{C^*}$$

Thus

$$\frac{A}{C} = \frac{A^*}{C^*}$$

and since A and C have no common factors $A = A^*$; $C = C^*$ giving $G^* = G$.

□

The results from Theorem 4 and Lemma 5 will now be combined to a theorem on the convergence points of the adaptive prediction algorithms.

Theorem 6. Given an ARMA process according to (2.1) where the noise $\{e(t)\}$ is supposed to be a stationary sequence of independent random variables with $E|e(t)|^p$ existing for each $p>1$.

Suppose that the process is predicted with an adaptive prediction algorithm where the parameter estimates $\theta_i(t)$ $i = 1, 2$ or 3 are generated by the algorithm (3.12).

Further suppose that $(\theta_i(t), R_i(t))$ tends to (θ_i^*, R_i^*) with a strictly positive probability and that the polynomials A^* and G^* for algorithm 1, C^* and G^* for algorithm 2, and G^* and H^* for algorithm 3 have no factors in common.

The asymptotic prediction process is then the minimum mean square error prediction process for (2.1). Moreover, if one of the predictors 2 or 3 is used the predictor is the unique minimum mean square error predictor for (2.1).

Proof. The result follows directly from Theorem 4 and Lemma 5.

□

Remark. Note that the algorithms 2 and 3 give an estimate of the polynomial H and not of A. It is not possible to find A from the minimum variance identity (2.6) without knowing F. One, trivial, such case is when k = 1, then F = 1 and A = H.

□

5.3. LOCAL CONVERGENCE PROPERTIES

In this section the second part of Theorem 4 is to be applied to the algorithms 2 and 3. The eigenvalues to the matrix

$$K_i(\theta_i^*) \triangleq G_i^{-1}(\theta_i^*)\frac{d}{d\bar{\theta}_i} f_i(\bar{\theta}_i)\Big|_{\bar{\theta}_i=\theta_i^*} ; \quad i = 2,3$$

in the theorem will be determined, which gives a local convergence condition. According to Lemma 5 the stationary point θ_i^* contains the minimum mean square error predictor parameters θ_i given in Chapter 2.

The ELS algorithm, i.e. algorithm 0, is treated in part III of this report. There it is shown that the eigenvalues to the K_0-matrix in θ_0 are

$$\begin{cases} -1 \text{ of multiplicity } n \\ -1/C(\alpha_i) \text{ where } A^f(\alpha_i) = 0; \quad i = 1,\ldots,n \end{cases}$$

These are also the eigenvalues in the linearization of the algorithms 2 and 3 when used for one-step prediction due to Theorem 1. Below is to be shown that these numbers in fact are the eigenvalues to the K_i-matrix in the linearization of the k-step prediction algorithms irrespective of k.

Study the functions $f_i(\bar{\theta}_i) = E\varphi_i(t,\bar{\theta}_i)\varepsilon(t,\bar{\theta}_i)$; i = 2, 3 i.e. for the algorithms based on estimation of the parame-

ters in the predictor. In this function

$$\varepsilon(t,\bar{\theta}_i) = y(t) - \hat{y}(t|t-k,\bar{\theta}_i) =$$

$$= \hat{y}_M(t|t-k) + \varepsilon_M(t) - \hat{y}(t|t-k,\bar{\theta}_i) =$$

$$= \varphi^T_{i,M}(t)\theta_i - \varphi^T_i(t,\bar{\theta}_i)\bar{\theta}_i + \varepsilon_M(t) =$$

$$= [\varphi_{i,M}(t) - \varphi_i(t,\bar{\theta}_i)]^T\theta_i +$$

$$+ \varphi_i(t,\bar{\theta}_i)[\theta_i-\bar{\theta}_i] + \varepsilon_M(t)$$

For both of the algorithms considered

$$[\varphi_{i,M}(t) - \varphi_i(t,\bar{\theta}_i)]^T\theta_i = [C(q^{-1}) - 1][\varepsilon_M(t) - \varepsilon(t,\bar{\theta}_i)]$$

When i = 3 the polynomial identity (2.6)

$$C(q^{-1}) = H(q^{-1}) + q^{-k}G(q^{-1})$$

have to be used. Thus

$$\varepsilon(t,\bar{\theta}_i) = [C(q^{-1}) - 1][\varepsilon_M(t) - \varepsilon(t,\bar{\theta}_i)] +$$

$$+ \varphi^T_i(t,\bar{\theta}_i)(\theta_i-\bar{\theta}_i) + \varepsilon_M(t)$$

$$\varepsilon(t,\bar{\theta}_i) = \varepsilon_M(t) + \frac{1}{C(q^{-1})}\varphi^T_i(t,\bar{\theta}_i)(\theta_i-\bar{\theta}_i)$$

Introduce the vector

$$\tilde{\varphi}_i(t,\bar{\theta}_i) = \frac{1}{C(q^{-1})}\varphi_i(t,\bar{\theta}_i) \qquad (5.7)$$

Then, since $\varepsilon_M(t)$ is a moving average of the k latest innovations $e(s)$, $s = t-k+1,\ldots,t$ it is uncorrelated with the

elements in $\varphi_i(t,\bar{\theta}_i)$. Thus we have

$$f_i(\bar{\theta}_i) = E\ \varphi_i(t,\bar{\theta}_i)\varepsilon(t,\bar{\theta}_i) =$$

$$= E\ \varphi_i(t,\bar{\theta}_i)\tilde{\varphi}_i^T(t,\bar{\theta}_i)(\theta_i-\bar{\theta}_i)$$

$$= \tilde{G}_i(\bar{\theta}_i)(\theta_i-\bar{\theta}_i) \qquad (5.8)$$

with

$$\tilde{G}_i(\bar{\theta}_i) = E\ \varphi_i(t,\bar{\theta}_i)\tilde{\varphi}_i^T(t,\bar{\theta}_i) \qquad (5.9)$$

Hence the matrix in the linearization around θ_i is

$$K_i(\theta_i) = -\ G_i^{-1}(\theta_i)\tilde{G}_i(\theta_i) =$$

$$= -\ [E\ \varphi_i(t,\theta_i)\varphi_i^T(t,\theta_i)]^{-1} E\ \varphi_i(t,\theta_i)\tilde{\varphi}_i^T(t,\theta_i) \qquad (5.10)$$

The eigenvalues of this matrix and their implication on the convergence of the algorithm are then given in the following theorem.

Theorem 7. Consider the adaptive prediction algorithms 2 and 3. If the parameter estimates converge to $\theta_i^* = \theta_i$ then the eigenvalues to $K_i(\theta_i^*)$, $i = 2,3$

$$-\frac{1}{C(\alpha_k)} \quad \text{where } A^f(\alpha_k) = 0 \quad k = 1,\ldots,n$$

have negative real parts.

Proof: The proof is an application of the results in part III of this report. It is given in Appendix D.

□

Remark. In the previous chapter the adaptive prediction algorithms were compared. It was, however, not possible to find

any time- and datainvariant relationship between the algorithms when used for k-step prediction, $k > 1$. The theorem shows, however, that the algorithms are equal in one important aspect. They are locally convergent for the same systems, i.e. only the data generating system, not the algorithm nor the number of steps to predict affects the local convergence of the predictor.

□

The condition for local convergence given in the theorem is discussed in part III of this report. There also examples are given where it is not fulfilled, i.e. the parameter estimation diverges.

6. MULTISTEP PREDICTION

Let the precess be described by (2.1). Consider the problem to design an algorithm which at time t simultaneously predicts the outcomes of the process $\{y(t)\}$ at times $t+1,\ldots,t+k,\ldots,t+N$ recursively in k. This prediction scheme is called a multistep predictor. A similar algorithm is presented in Bohlin (1976).

6.1 REPRESENTATIONS OF THE MULTISTEP PREDICTOR

Let $y(t+k)$ be predicted by a minimum mean square error predictor for each k, $k = 1,\ldots,N$. Then the mean square prediction error of any linear function

$$\sum_{1}^{N} \alpha_i y(t+i)$$

of future observations is minimized by

$$\sum_{1}^{N} \alpha_i \hat{y}_M(t+i|t)$$

Hence, a multistep prediction scheme can be seen as a simplification of the calculations in the prediction of such a linear combination of observations, cf. Box and Jenkins (1970), Jazwinski (1970).

In the following parts of this section some different ways of computing this predictor will be presented.

Consider first (2.1) at time t+k

$$y(t+k) + a_1 y(t+k-1) + \ldots + a_n y(t+k-n) =$$
$$= e(t+k) + \ldots + c_n e(t+k-n)$$

The conditional expectation given y_t is

$$E\{y(t+k)|y_t\} + a_1 E\{y(t+k-1)|y_t\} + \ldots +$$

$$+ a_n E\{y(t+k-n)|y_t\} =$$

$$= E\{e(t+k)|y_t\} + \ldots + c_n E\{e(t+k-n)|y_t\}$$

$$E\{y(t+\tau)|y_t\} = \begin{cases} \hat{y}_M(t+\tau|t) & \tau > 0 \\ y(t+\tau) & \tau \leq 0 \end{cases}$$

and

$$E\{e(t+\tau)|y_t\} = \begin{cases} 0 & \tau > 0 \\ e(t+\tau) & \tau \leq 0 \end{cases}$$

since $F\{y(t), y(t-1), \ldots\}$ also contains the noise sequence up to and including t.

As $C(q^{-1})$ is asymptotically stable, $e(t)$ may be calculated at time t

$$e(t) = \frac{A(q^{-1})}{C(q^{-1})} y(t)$$

The calculation of the prediction can be done recursively for $k = 1, \ldots, N$ from the equation

$$\hat{y}_M(t+k|t) + \ldots + a_{k-1}\hat{y}_M(t+1|t) + a_k y(t) + \ldots + a_n y(t+k-n) =$$
$$= c_k e(t) + \ldots + c_n e(t+k-n) \qquad (6.1a)$$

for $k \leq n$ and from

$$\hat{y}_M(t+k|t) + a_1\hat{y}_M(t+k-1|t) + \ldots + a_n\hat{y}_M(t+k-n|t) = 0 \qquad (6.1b)$$

for $k > n$ (compare e.g. Akaike (1974)).

Another multistep predictor which is recursive in the number of steps to predict,starts from a one-step predictor of the values of the process at time t+k. Equation (2.9) gives

$$\hat{y}_M(t+k|t+k-1) + \ldots + c_n\hat{y}_M(t+k-n|t+k-n-1) =$$

$$= g_0 y(t+k-1) + \ldots + g_{n-1}y(t+k-n)$$

or

$$E\left(y(t+k)|\,y_{t+k-1}\right) + c_1E\left(y(t+k-1)|\,y_{t+k-2}\right) + \ldots +$$

$$+ c_nE\left(y(t+k-n)|\,y_{t+k-n-1}\right) =$$

$$= g_0 y(t+k-1) + \ldots + g_{n-1}y(t+k-n)$$

Now take a conditional expectation of this equation given y_t. The result is

$$E\left[E\left(y(t+k)|\,y_{t+k-1}\right)|y_t\right] + c_1E\left[E\left(y(t+k-1)|\,y_{t+k-2}\right)|\,y_t\right] +$$

$$+ \ldots + c_nE\left[E\left(y(t+k-n)|\,y_{t+k-n-1}\right)|\,y_t\right] =$$

$$= g_0E\left(y(t+k-1)|\,y_t\right) + \ldots + g_{n-1}E\left(y(t+k-n)|\,y_t\right)$$

But the conditional expectations are

$$E\left[E\left(y(t+\tau)|\,y_{t+\tau-1}\right)|\,y_t\right] = \begin{cases} E\left(y(t+\tau)|\,y_{t+\tau-1}\right) & \tau \leq 0 \\ E\left(y(t+\tau)|\,y_t\right) & \tau > 0 \end{cases}$$

$$= \begin{cases} \hat{y}_M(t+\tau|t+\tau-1) & \tau \leq 0 \\ \hat{y}_M(t+\tau|t) & \tau > 0 \end{cases}$$

(compare e.g. Chung (1968)), which gives the following recursive equation for the prediction of the process at time t+k

$$\hat{y}_M(t+k|t) + c_1\hat{y}_M(t+k-1|t) + \ldots + c_{k-1}\hat{y}_M(t+1|t) +$$

$$+ c_k\hat{y}_M(t|t-1) + \ldots + c_n\hat{y}_M(t+k-n|t+k-n-1) =$$

$$= g_0\hat{y}_M(t+k-1|t) + \ldots + g_{k-2}\hat{y}_M(t+1|t) +$$

$$+ g_{k-1}y(t) + \ldots + g_{n-1}y(t+k-n) \qquad (6.2a)$$

for $k \leq n$ and

$$\hat{y}_M(t+k|t) + \ldots + c_n\hat{y}_M(t+k-n|t) =$$

$$= g_0\hat{y}_M(t+k-1|t) + \ldots + g_{n-1}\hat{y}_M(t+k-n|t) \qquad (6.2b)$$

$k > n$. The identity (2.6) with $k = 1$ applied on (6.1) gives another possibility to derive this representation of the predictor.

Finally, the system (2.1), i.e.

$$y(t) = C(q^{-1})/A(q^{-1})e(t) = e(t) + \left(C(q^{-1}) - A(q^{-1})\right)/$$

$$/A(q^{-1})e(t) \qquad (6.3)$$

can be represented on state space form

$$\begin{cases} x(t+1) = F\,x(t) + G\,e(t) \\ y(t) = H\,x(t) + e(t) \end{cases}$$

This gives the optimal prediction of $y(t+k)$ given y_t as

$$\begin{cases} \hat{y}_M(t+k|t) = H\,\hat{x}(t+k|t) \\ \hat{x}(t+k|t) = F\,\hat{x}(t+k-1|t) = \ldots = F^{k-1}\hat{x}(t+1|t) \end{cases}$$

where $\hat{x}(t+1|t)$ is given by an ordinary Kalman filter expression (cf Åström (1970)).

As only stationary predictors of (6.3) are considered the interesting solution to the Riccati equation is P = 0 giving K (the Kalman gain) = G, provided that F - GH is stable, i.e. $C(q^{-1})$ is stable. P = 0 is a solution since the state at time t can be considered as composed of noise elements $e(\tau)$ $\tau \leq t-1$, cf (6.3). The one-step prediction error variance of y(t+1) given y_t is then σ^2, the minimum value of this error variance. Thus a representation of the multistep predictor will be

$$\begin{cases} \hat{x}(t+1|t) = F\,\hat{x}(t|t-1) + G\,\varepsilon(t) & (6.4a) \\ y(t) = H\,\hat{x}(t|t-1) + \varepsilon(t) & (6.4b) \end{cases}$$

$$\hat{y}_M(t+k|t) = H\,\hat{x}(t+k|t) \quad (6.4c)$$

$$\hat{x}(t+k|t) = F\,\hat{x}(t+k-1|t) \quad (6.4d)$$

Thus, by starting either at the process equation (2.1) or at the predictor equation (2.9) with k = 1, the multistep predictor of $y(t+k)$, $k = 1,\ldots,N$ easily can be recursively calculated from any of the representations (6.1), (6.2) or (6.4).

6.2. ADAPTIVE MULTISTEP PREDICTION

Suppose that the process is described by the equation (2.1) but the parameters in that equation are unknown. The design of an adaptive multistep predictor may then, as here, be based on the certainty equivalence principle, cf e.g. Wittenmark (1975) . First the unknown parameters either in the process equation (2.1) or in the one-step predictor (2.9 with k = 1) are estimated with some parameter estimation method. The obtained estimates are then used as if they were

the correct values of the corresponding parameters.

The representations of the multistep predictor with known parameters discussed in Section 6.1 suggest three possible algorithms for adaptive multistep prediction.

Algorithm A is based on the process equation (2.1) and realizes the prediction via (6.1), i.e.

A1/ Calculate the estimate of e(t) from the equation $\hat{e}(t) = y(t) - \hat{y}(t|t-1)$

A2/ Estimate the parameters in the polynomials A and C giving the polynomials A(t) and C(t)

A3/ Use the equation (6.1) with A(t), C(t) and $\{\hat{e}(t)\}$ instead of A, C and $\{e(t)\}$ to determine the desired predictions recursively in k.

In algorithm B the starting point is the one-step predictor ((2.9) with k = 1), i.e. the steps in the algorithm are

B1/ Estimate the parameters in the G and C polynomials to get the polynomials G(t) and C(t)

B2/ Use the equation (6.2) to determine the desired predictions recursively in k.

Finally, in algorithm C the predictions are calculated from the state space representation of the process. Thus the steps in this algorithm are

C1/ Estimate the parameters in the A and C polynomials and use the result together with a reconstructed state z(t) in a state space representation of the process, cf (6.4a, b)

$$\begin{cases} \hat{z}(t+1|t) = F(t)z(t) + G(t)\varepsilon(t) \\ y(t) = H(t)z(t) + \varepsilon(t) \end{cases}$$

Here $F(t)$ is the estimate at time t of the matrix F, etc.

C2/ Use the equations (6.4c, d) to determine the desired predictions recursively in k.

Consider first the algorithms A and B. Suppose that the ELS method is used in the estimation part in algorithm A and suppose that the estimates of the parameters in the G and C polynomials in algorithm B are obtained from an adaptive one-step predictor as described in Chapter 3. It then follows from Theorem 1 that these algorithms are algebraically equivalent giving identical one step ahead predictions, if they are started with proper relations between the initial values. Since the value of the k step ahead prediction is calculated recursively this means that the two algorithms A and B are algebraically equivalent for <u>all</u> values of k. The asymptotical adaptive multistep predictors A and B will thus produce minimum mean square error prediction of the process for all values of k if the parameter estimates converge (cf. Chapter 5).

Suppose that the ELS method is used for parameter identification also in algorithm C. Then the values of the parameters used in the prediction coincide with the values used in the other two algorithms. If also a new reconstruction of the state $z(t)$ is done at every time step to account for the latest values of the parameter estimates, the resulting predictions $\hat{y}(t+k|t)$ $k = 1,\ldots,N$ will be identical to the predictions obtained from the other two algorithms. However, this algorithm will have higher computational demand due to the need of reconstruction. A fast method for doing this when the system is represented on observable canonical form is

presented in Åström (1974a). As is shown in the following example it is also possible to do without the explicit reconstruction. The amount of computations needed is then of the same order of magnitude. The resulting predictions are, of course, identical.

Example. If a third order system (2.1) is represented on observable canonical form the reconstructed state at time t+1 given information up to time t+1 is

$$z(t+1) = \begin{bmatrix} y(t+1) - \varepsilon(t+1) \\ -a_2(t+1)y(t) - a_3(t+1)y(t-1) + c_2(t+1)\varepsilon(t) + c_3(t+1)\varepsilon(t-1) \\ -a_3(t+1)y(t) + c_3(t+1)\varepsilon(t) \end{bmatrix}$$

where the estimate at time t of the parameter c_i is denoted $c_i(t)$ etc.

Note that

$$y(t+1) - \varepsilon(t+1) = \hat{y}(t+1|t) =$$

$$= -a_1(t)y(t) - a_2(t)y(t-1) - a_3(t)y(t-2) +$$

$$+ c_1(t)\varepsilon(t) + c_2(t)\varepsilon(t-1) + c_3(t)\varepsilon(t-2)$$

The predicted state at time t+1 given information about the process up to t is

$$\hat{z}(t+1|t) = \begin{bmatrix} -a_1(t)y(t) - a_2(t)y(t-1) - a_3(t)y(t-2) + \\ \quad + c_1(t)\varepsilon(t) + c_2(t)\varepsilon(t-1) + c_3(t)\varepsilon(t-2) \\ -a_2(t)y(t) - a_3(t)y(t-1) + c_2(t)\varepsilon(t) + \\ \quad + c_3(t)\varepsilon(t-1) \\ -a_3(t)y(t) + c_3(t)\varepsilon(t) \end{bmatrix}$$

Thus if we like to find a vector $r(t+1)$ such that

$$z(t+1) = r(t+1) + \hat{z}(t+1|t)$$

then

$$r(t+1) = \begin{bmatrix} 0 & 0 \\ -a_2(t+1) + a_2(t) & -a_3(t+1) + a_3(t) \\ -a_3(t+1) + a_3(t) & 0 \end{bmatrix} \begin{bmatrix} y(t) \\ y(t-1) \end{bmatrix} +$$

$$+ \begin{bmatrix} 0 & 0 \\ c_2(t+1) - c_2(t) & c_3(t+1) - c_3(t) \\ c_3(t+1) - c_3(t) & 0 \end{bmatrix} \begin{bmatrix} \varepsilon(t) \\ \varepsilon(t-1) \end{bmatrix} \triangleq$$

$$\triangleq N_y'(t+1)s_y(t+1) + N_\varepsilon'(t+1)s_\varepsilon(t+1)$$

Since

$$F(t+1)N_y'(t+1) = \begin{bmatrix} -a_2(t+1)+a_2(t) & -a_3(t+1)+a_3(t) \\ -a_3(t+1)+a_3(t) & 0 \\ 0 & 0 \end{bmatrix} \triangleq N_y(t+1)$$

and similarly for $F(t+1)N_\varepsilon'(t+1) \triangleq N_\varepsilon(t+1)$ the algorithm is

$$\begin{cases} \hat{z}(t+2|t+1) = F(t+1)\hat{z}(t+1|t) + N_y(t+1)s_y(t+1) \\ \qquad\qquad + N_\varepsilon(t+1)s_\varepsilon(t+1) + G(t+1)\varepsilon(t+1) \\ y(t+1) = H(t+1)z(t+1) + \varepsilon(t+1) \end{cases}$$

The example is easily generalized to arbitrary n.

The algorithm closely resembles an algorithm given in Landau (1976).

□

7. NUMERICAL EXAMPLES

The different methods for adaptive prediction have also been simulated. The aim has then mainly been to find out whether any of the proposed algorithms is in any respect clearly superior or clearly inferior to the others. However, only the algorithms 0, 2 and 3 are included in the comparison, due to the following reasons:

- Algorithms 2, 4 and 5 are equivalent according to Theorem 2, therefore only algorithm 2 is represented.
- In algorithm 1 an estimate of the C polynomial is calculated from (3.13). This calculation has however shown to be numerically ill conditioned when $a_n(t)$ is small. For this reason algorithm 1 is considered significantly inferior to the others and ruled out from further comparisons.
- Since Theorem 1 states that the algorithms 0, 2 and 3 are equivalent for $k = 1$ only $k > 1$ is considered.

The methods are compared on ARMA processes and the order of the model is the same as the order of the generating system. The noise $\{e(t)\}$ is generated by a random number generator, producing a normally distributed random variable with zero mean and unit variance.

To measure the goodness of the prediction some loss function akin to the function used in Chapter 2 for known parameters, i.e. (2.4)

$$V_0 = E\ \varepsilon(t)^2$$

should be used. One possible such loss function, which is

used here, is the time average over the observed prediction errors, i.e.

$$V = V(n_0,n) = \frac{1}{n-n_0} \sum_{n_0+1}^{n} \varepsilon^2(t) \qquad (7.1)$$

There are of course other possible loss functions with resemblence of V_0 above. The choise amongst them depends on the application, e.g. in Söderström, Ljung and Gustavsson (1974), where parameter estimation algorithms are compared, the loss function

$$\tilde{V}(\theta) = E\ \varepsilon(t,\theta)^2$$

is employed. The loss function V (7.1) is used for example in Wittenmark (1974), Åström and Wittenmark (1973), Borisson (1975), and Clarke and Gawthrop (1975) in connection with discussions on self-tuning algorithms. It is used since some of the influence of the noise realization on the parameter estimates is supposed to be averaged out.

The initial values of the parameters have been zero, and the P matrix in the algorithm has initially been a scaled identity matrix.

At each step in the algorithms 0 and 2 the stability of the estimated C polynomial was tested and the estimates were modified to give stability (cf. Gustavsson (1969)).

A commonly used method for prediction of timeseries is the exponential smoothing algorithm, see Brown (1963). When applied to a pure ARMA process, this algorithm reads

$$\begin{aligned} s(t) &= \lambda y(t) + (1-\lambda)\ s(t-1) \\ \hat{y}(t+k|t) &= s(t) \end{aligned} \qquad (7.2)$$

This algorithm has also been included in the comparisons and in the simulations of it, different values of λ have been tested. In the following examples it is used with that value of λ that minimizes the loss (7.1) in each particular case.

Example 1. Consider the process

$$(1 - 1.5q^{-1} + 0.7q^{-2})y(t) = (1 + 0.4q^{-1} - 0.21q^{-2})e(t) \quad (7.3)$$

where $\sigma^2 = 1$, subject to two step ahead prediction. The optimal predictor, i.e. the minimum mean square error predictor with known parameters, is given by

$$\hat{y}_M(t+2|t) = \frac{1.94 - 1.33q^{-1}}{1 + 0.4q^{-1} - 0.21q^{-2}}\, y(t) = \frac{1.94\ (1 - 0.69q^{-1})}{(1 - 0.3q^{-1})(1 + 0.7q^{-1})} \cdot$$

$$\cdot\, y(t) = \frac{1.94 - 1.33q^{-1}}{1 + 0.4q^{-1} - 2.15q^{-2} + 1.33q^{-3}}\, \varepsilon_M(t) \quad (7.4)$$

and the minimal loss is $V_0 = 4.61$. The simulations lasted over 2000 steps and the initial value of the P matrix was $0.1 \cdot I$, where I is an identity matrix. The accumulated loss V (7.1) was calculated for the three adaptive algorithms for 10 different realizations of the noise. The mean value and standard deviation of V during the last 1000 and last 500 steps are shown in Table 3.

Table 3 - Sample mean and standard deviation for ten realizations of the loss when the process (7.3) is predicted 2 steps ahead.

Algorithm	V(1000,2000)	V(1500,2000)
0	4.716 ± 0.287	4.773 ± 0.270
2	4.712 ± 0.287	4.769 ± 0.269
3	4.726 ± 0.290	4.780 ± 0.269
Exp.smoothing	7.957 ± 0.543	8.063 ± 0.538

The calculated estimation error in the mean value of the loss for the adaptive algorithms thus is 0.09.

The accumulated loss function from one of the runs is shown in Figure 1. It shows that all the adaptive predictors give about the same incremental loss as the optimal predictor after only a few steps. The parameter estimates for algorithm 2 from the same noise realization are shown in Figure 2.

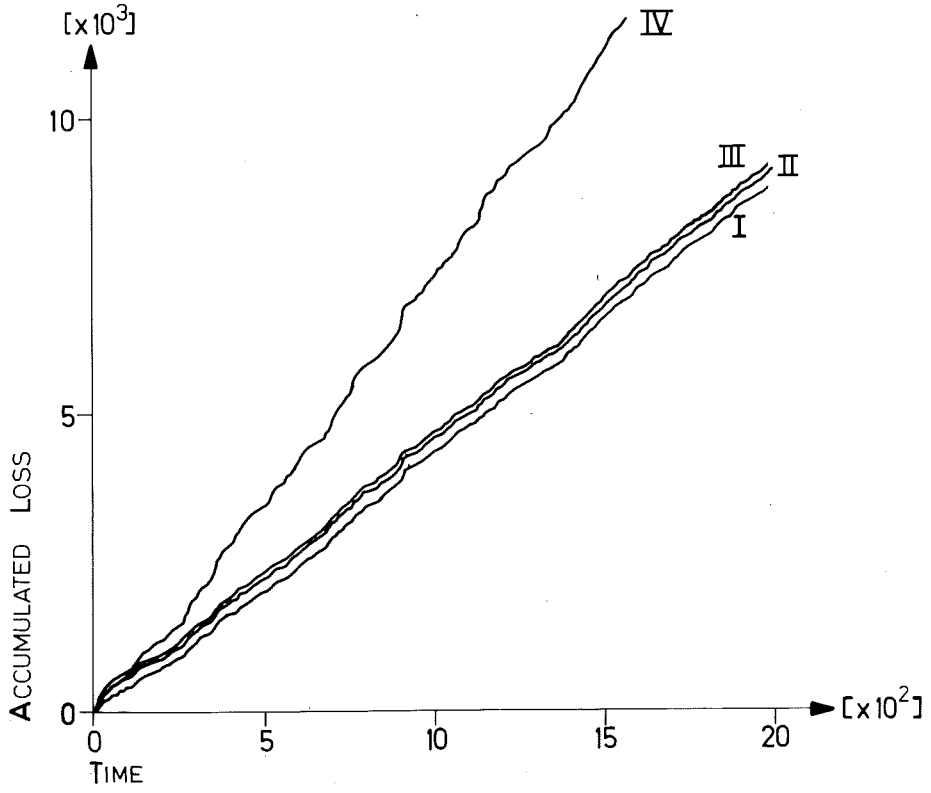


Figure 1 - Accumulated loss function for the adaptive predictors, the optimal predictor, i.e. the minimum mean square error predictor for the process with known parameters, and the best single exponential smoothing algorithm when applied to the process in Example 1. The noise realization is the same in all five cases.
I: Optimal predictor
II: Adaptive prediction algorithms 0 and 2
III: Adaptive prediction algorithm 3
IV: Exponential smoothing with $\lambda = 1.9$

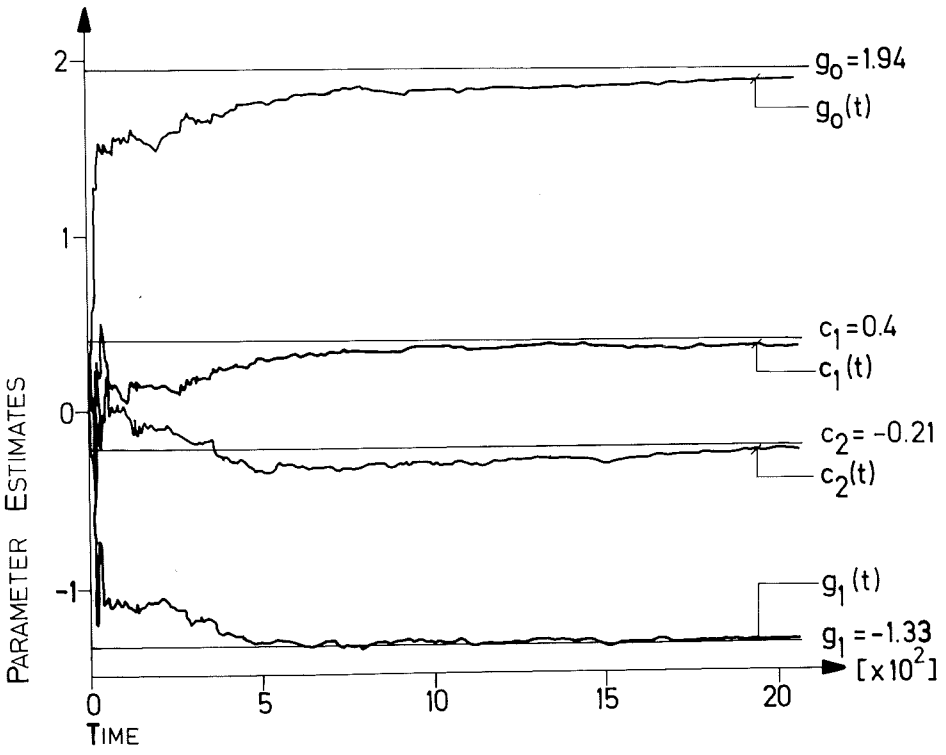


Figure 2 - Estimates of the C and G parameters in algorithm 2 applied to Example 1. The same noise realization as in Figure 1 is used.

The parameter estimates need considerably more time to reach their final values. Thus the adaptive algorithm has a good performance even when the parameters are not very precisely known. This is a typical behaviour of the algorithms. It is reported also in Wittenmark (1974) and Åström and Wittenmark (1973). The discussion is applicable also to the other adaptive algorithms.

The predictions are shown in Figure 3 for the optimal and the adaptive predictors in the initial part of the prediction. After about 40 steps the prediction values from the optimal and the different adaptive predictors are fairly close to each other, cf. Figure 1. The initial behaviour of the different adaptive predictors are similar.

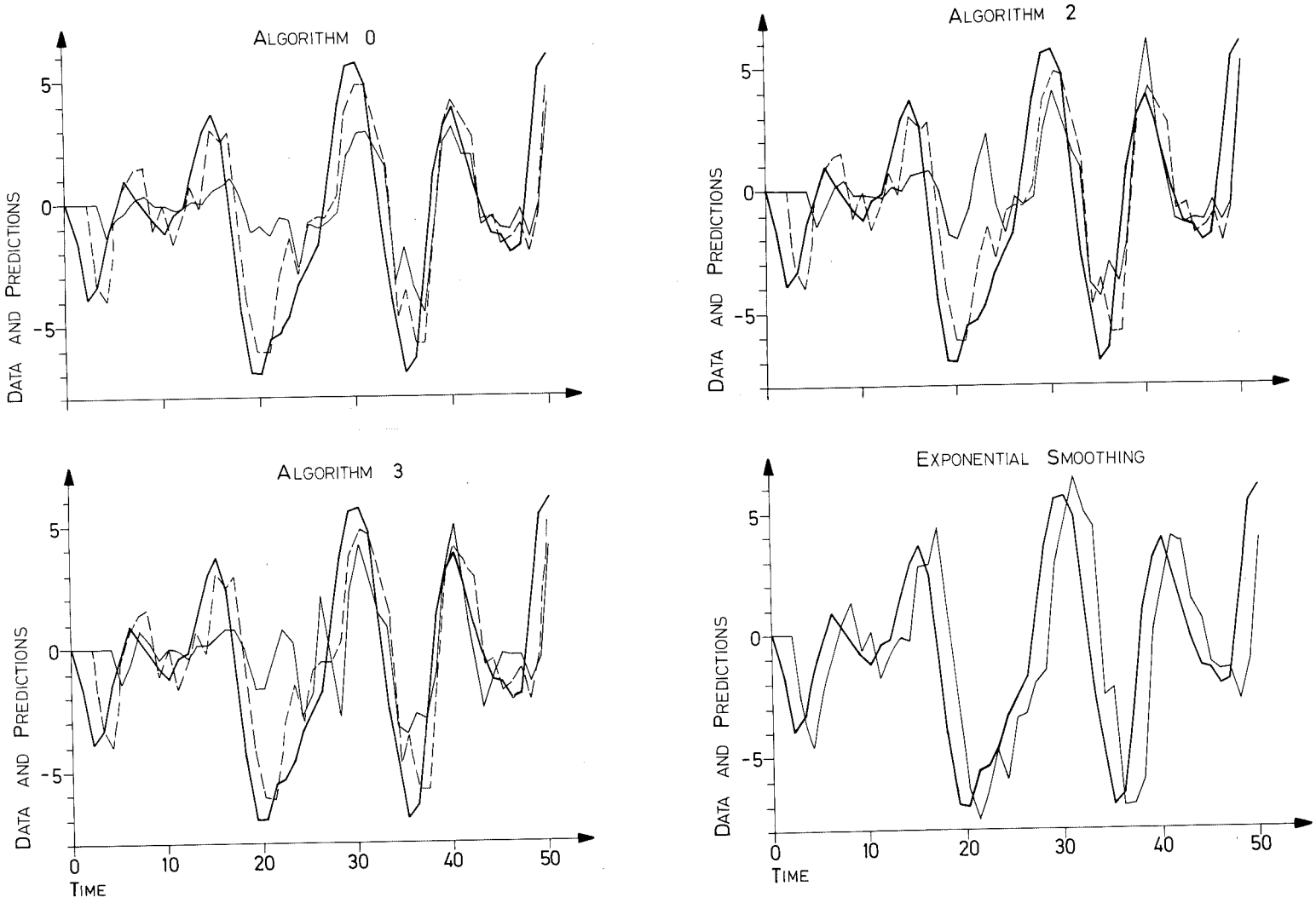


Figure 3 - Data and prediction values from the adaptive predictors, the optimal predictor and the exponential smoothing algorithm when applied to Example 1. The noise realization is the same as in Figures 1 and 2. ——— Data; — — — Optimal predictor; ——— Adaptive predictor or Exponential smoothing algorithm.

It is thus not possible from this example to point out any of the adaptive algorithms as being significantly different from the others.

The resulting average loss when the exponential smoothing algorithm (7.2) was applied to the process is also shown in Table 3. In Figure 1 one of the realizations of the loss function is plotted. Hence, in this example the adaptive predictors give significantly better predictions. Figure 3 shows that the exponential smoothing algorithm in contrast to the adaptive predictors lags the data with two time steps.

The best value of λ was $1.9 > 1$. The filter in (7.2) thus has a pole in -0.9, compare with the pole-zero constellation for the optimal predictor (7.4). Thus a zero to the C polynomial in the left half of the unit disc might indicate that the best value of the parameter in the exponential smoothing algorithm will be greater than 1.

□

Example 2. Consider 5 step prediction of the process

$$(1 - 1.6q^{-1} + 0.63q^{-2})y(t) = (1 - 1.6083q^{-1} + 0.9875q^{-2})e(t) \tag{7.5}$$

with $\sigma^2 = 1$. The optimal predictor is

$$\hat{y}_M(t+5|t) = \frac{0.7261 - 0.4238q^{-1}}{1 - 1.6083q^{-1} + 0.9875q^{-2}}\, y(t) =$$

$$= \frac{0.7261 - 0.4238\, q^{-1}}{1 - 1.6083q^{-1} + 0.9875q^{-2} - 0.7261q^{-5} - 0.4238q^{-6}}\, \varepsilon_M(t)$$

which gives the miminal loss $V_0 = 1.88$.

The adaptive predictors have been tested on one realization of the noise. The test lasted over 10000 steps and the

initial value of the matrix P was 0.1 times an identity matrix. When the adaptive predictor based on prediction errors, i.e. algorithm 3, was used eight different parameters had to be estimated. In the two other predictors considered only four parameters had to be determined.

In this example where the zeroes of the polynomial C lie close to the unit circle the parameter estimates approach their corresponding true values slowly compared to the process in Example 1, see Figure 4. It is especially evident for algorithm 3. In this algorithm the parameters in the polynomial C have been calculated from the estimated polynomial H. A condition number of the matrix P (3.10) measured as the quotient between the largest and the smallest eigenvalue, is $4.1 \cdot 10^2$ for algorithm 0, $1.8 \cdot 10^2$ for algorithm 2, and $2.5 \cdot 10^3$ for algorithm 3 after 5000 steps.

The loss function (7.1) is shown in Table 4. It shows that the incremental loss for the adaptive predictors just slowly approaches the values for the minimum mean square error predictor of the process with known parameters. In this realization of the noise the algorithms 2 and 3 are slightly better in the initial and final parts of the prediction respectively.

Table 4 - Loss functions for the predictors applied to Example 2 when the process is predicted 5 steps ahead.

Algorithm	V(1000,2000)	V(5000,10000)	V(9000,10000)
0	2.60	2.12	2.26
2	2.43	2.02	2.14
3	2.02	1.99	2.21
Optimal	1.82	1.88	2.03
Exp. smoothing	2.93	2.98	3.40

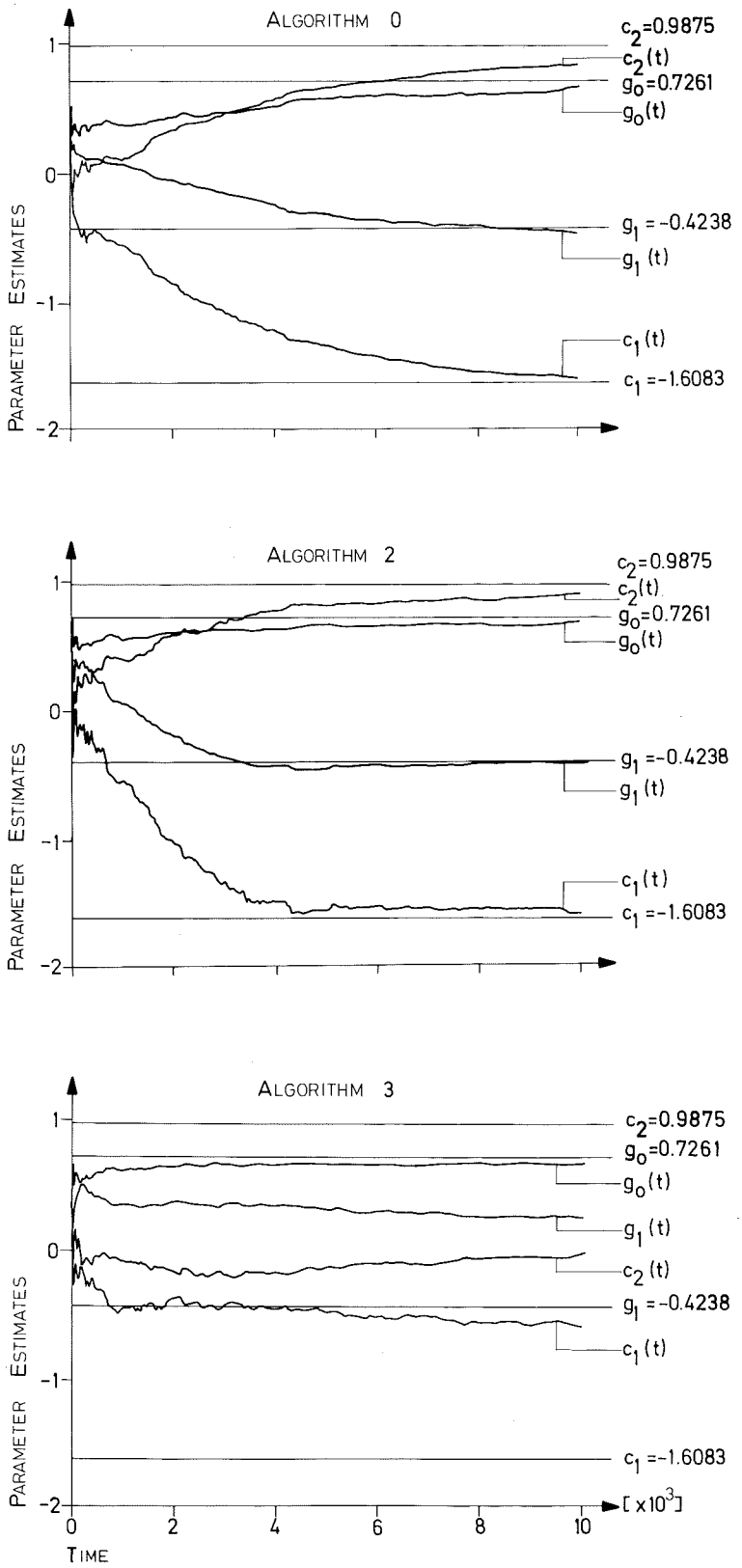
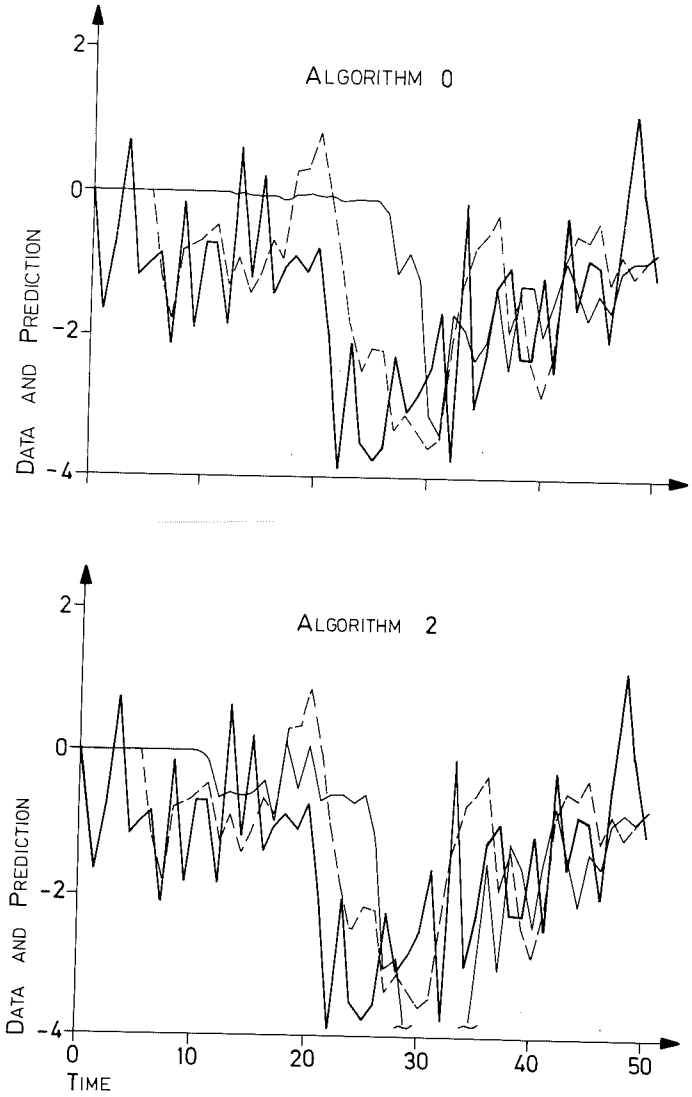


Figure 4 - Parameter estimates from the adaptive predictors applied to Example 2.



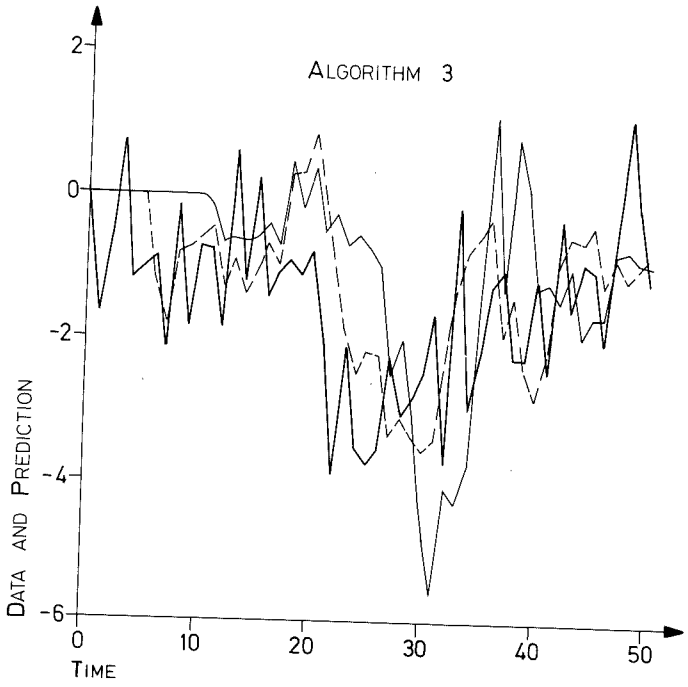


Figure 5 - Data and prediction values during the first 50 steps in Example 2 for the adaptive algorithms, and the optimal predictor. The noise realization is the same as in Figure 4.

——— Data

– – – – – Optimal predictor

——— Adaptive predictor

The predictions are shown in Figures 5 and 6 in the initial part of the prediction and after 9500 steps respectively. During the initial part the predicted values are very far from the predictions from the optimal predictor. Even after more than 9500 steps the predictions from the adaptive predictors differ significantly from the predicted values from the minimum mean square error predictor when the parameters are known, cf. Example 1. This is caused both by the parameters being incorrect and the zeros of the polynomial C being very close to the unit circle.

The exponential smoothing algorithm (7.2) was also simulated on this example. The loss is given in Table 4 for $\lambda = 0.73$. The minimum of the loss (7.1) with respect to λ was very flat, giving almost the same loss for $0.5 < \lambda < 1$.

□

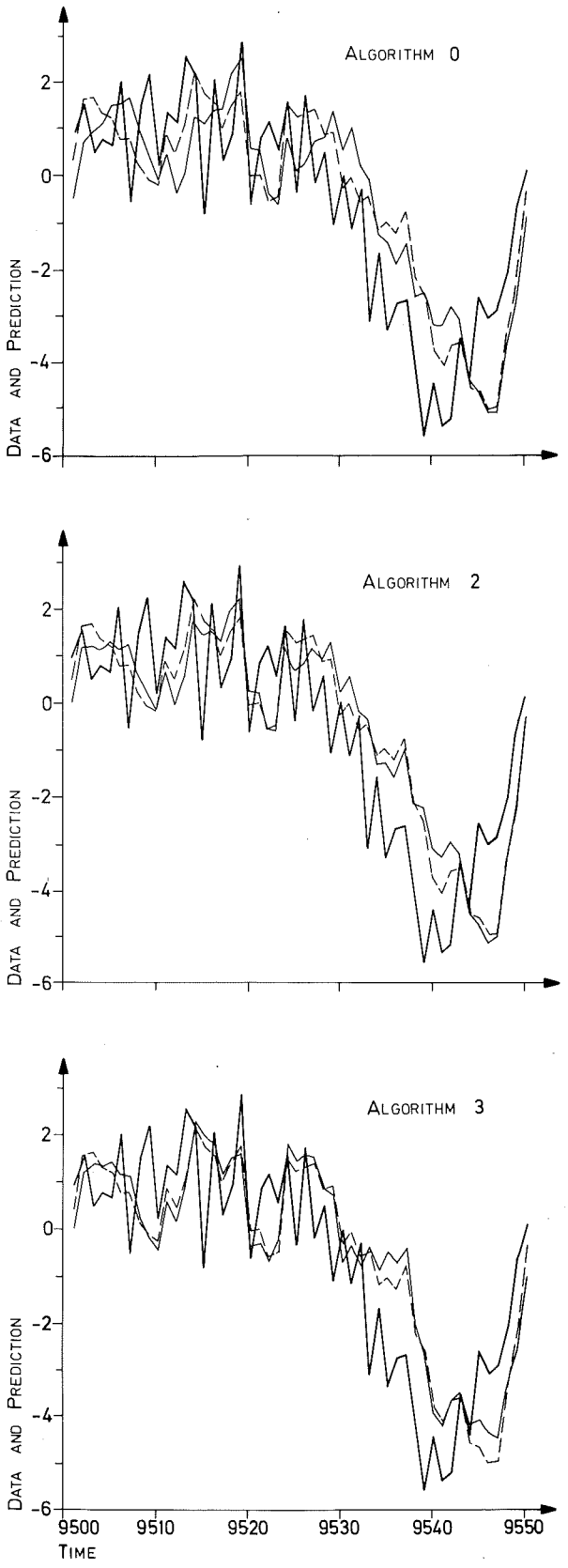


Figure 6 - Data and prediction values after 9500 steps in Example 2 for the adaptive algorithms and the optimal predictor. The noise realization is the same as in Figure 4.

——— Data

— — — — Optimal predictor

——— Adaptive predictor

8. Summary and discussion

In this report the problem of linear minimum mean square error prediction of a stochastic ARMA process with unknown but constant parameters have been discussed. Six methods with different structures for solving the problem have been considered. All these methods consist of two parts, an identification part where the parameters are estimated and a prediction part where the obtained parameter estimates are used for prediction. One of the methods, algorithm 0, is based on estimation of the parameters in the ARMA model and calculation of the predictor. In the remaining five algorithms the parameters in the predictor are estimated. The Least Squares method has been used for the parameter estimation.

It has been shown that all the methods are equivalent for one-step prediction, thus having identical transient and asymptotical properties. This result can be extended to k-step prediction for general k for the methods 2, 4 and 5 but not for the others. Thus the methods 4 and 5 can be considered just as equivalent variants of method 2.

The convergence properties of the algorithms have been analysed. Since the Extended Least Squares method is used, the algorithms do not give converging parameter estimations for all systems. However, it has been shown that if the parameter estimates converge, they converge to the minimum mean square error k-step predictor. Moreover, it has been shown that the methods 0, 2 and 3 give a nonconverging parameter estimation if any of the numbers

$$-\frac{1}{C(\alpha_k)} \quad \text{where} \quad A^f(\alpha_k) = 0; \quad k = 1,\ldots,n$$

have a positive real part. Thus the same necessary condition for local convergence is applicable to these three methods

irrespective of the number of steps to predict. There are other parameter estimation methods which do not have this divergence drawback. One such is the Recursive Maximum Likelihood method (cf. Söderström, Ljung and Gustavsson (1974)). When it is used for identification of the parameters in the process model the parameter estimates converge for all systems.

For the algorithm 0, where the parameters in the process model are identified, a linear system of equations must be solved in every time step. Also for algorithm 1 more computations are needed for k-step prediction than in any other of the algorithms 2,...,5 since a linear system of equations must be solved in every time step. This system of equations will be ill conditioned if the estimate of the coefficient for q^{-n} in the polynomial A is small. For the algorithm 3 k-1 more parameters are estimated than in any other structure. Thus the equivalent algorithms 2, 4 and 5 need less computations per prediction than the others.

In algorithms 0, 2 and 5 the characteristic polynomial of the transfer function from data to prediction or prediction error is estimated. This gives a possibility at every time step to maintain the stability of the prediction, which might be desirable.

In Tables 5 and 6 the execution time and storage requirement for some of the algorithms are given. The figures refer to a PDP-15/35 computer with floating point hardware. In the least squares identification routine double precision arithmetic is used. The difference in execution time between the algorithms 0 and 2 is caused by the need to solve the identity (2.6) in algorithm 0. The storage requirement in Table 6 does not include data areas or library routines.

Table 5 - Execution time for some of the algorithms.

n	k	Algorithm 0		Algorithm 2		Algorithm 3	
		# par	Time (ms)	# par	Time (ms)	# par	Time (ms)
2	2	4	32	4	22	5	30
	5	4	34	4	22	8	60
	10	4	39	4	22	13	134
5	2	10	97	10	85	11	101
	5	10	101	10	85	14	154
	10	10	108	10	85	19	266
10	2	20	308	20	292	21	320
	5	20	313	20	292	24	410
	10	20	322	20	292	29	586

n	Stability test Time (ms)
2	3.2
5	8.2
10	19.6

Table 6 - Storage requirement for some of the algorithms.

Algorithm 0 with LS identification and solution of identity (2.6)	1 375
Algorithm 2 with LS identification	755
Algorithm 3 with LS identification	764
Stability test	533
LS identification	445
Solution of identity (2.6)	409

The algorithms have also been subject to simulations for comparison between the predictors 0, 2 and 3. It is a difficult task to make an always valid statement with just simulation results at hand. There are however often no other practical method available. The rather extensive simulations performed, of which two examples are given, show that the algorithm 3 often give slightly inferior predictions when the accumulated prediction error is used as a goodness measure. It should however be underlined that the differences mostly are very small. It has not been possible to distinguish between the algorithms 0 and 2 from the simulation studies. All the adaptive algorithms have however a significantly better performance than ordinary exponential smoothing.

In this part of the report also some different ways of designing an adaptive multistep predictor, i.e. an algorithm that gives a prediction of the ARMA process 1,...,N steps ahead, have been presented. These multistep predictors are all recursive in the number of steps to predict. Two of them are based on a difference equation representation of the prediction, versions 0 and 2 in the discussion of the k-step predictor, and the third on a state space representation.

It has been demonstrated that when the Extended Least Squares method is used for parameter estimation, these predictors are algebraically equivalent giving identical predictions when started with proper relations between the initial values. This especially means that if the parameter estimates converge these adaptive multistep predictors asymptotically give minimum mean square error k-step prediction for $k = 1,\ldots,N$.

An algorithm for adaptive multistep prediction of $y(t+k)$, $k = 1,\ldots,N$ based on y_t could of course also be designed

as N adaptive predictors. Such a multistep predictor could then be realized either via estimation of the parameters in the ARMA model followed by N systems of equations in order to calculate the parameters in the predictors or via estimation of the parameters in the predictors directly. In the first of these approaches the solution of the N systems of equations can be simplified,cf Holst (1977). Only one parameter estimation is needed, i.e. 2n parameters must be calculated if the system is of n:th order. In the second of these approaches it is necessary to estimate the parameters in N adaptive predictors. Whichever of these alternatives is used the old prediction values for N predictors must be stored in the computer.

In the predictors presented here considerably fewer data elements have to be stored. Estimation of 2n parameters in an n:th order system is needed. Thus, this way of solving the adaptive multistep prediction problem is more efficient both regarding computer time and computer storage. However, since there are errors in the parameter estimates these errors might propagate in an increasing manner into the predictions. This could cause a deterioration of the prediction result when the number of steps to predict increases. Furthermore, when the process to be predicted is just approximately described by an ARMA process the one-step predictor parameters could be irrelevant for the k-step prediction. In both cases it might be worthwhile to use the adaptive k-step predictors.

A further comparison of the prediction algorithms together with examples of multistep prediction is given in part II of the report. This part contains an application on short term prediction of power load.

9. References

Akaike, H. (1974): Markovian Representation of Stochastic Processes and its Application to the Analysis of Autoregressive Moving Average Processes. The Institute of Statistical Mathematics, Tokyo.

Albert, A. (1972): Regression and the Moore-Penrose Pseudoinverse. Academic Press, New York.

Åström, K.J. (1968): Lectures on the Identification Problem - The Least Squares Method. TFRT-3004, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden.

Åström, K.J. (1970): Introduction to Stochastic Control Theory. Academic Press, New York.

Åström, K.J. (1974a): A Self-Tuning Regulator for Nonminimum Phase Systems. TFRT-3113, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden.

Åström, K.J. (1974b): A Self-Tuning Parameter Estimator. TFRT--3114, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden.

Åström, K.J., U. Borisson, L. Ljung and B. Wittenmark (1977): Theory and Application of Self-Tuning Regulators. To be published in Automatica 13. This is an expanded version of a paper given at the 6th IFAC World Congress 1975 in Boston, Mass.

Åström, K.J. and P. Eykhoff (1971): System Identification - A Survey. Automatica 7, 123-162.

Åström, K.J. and T. Söderström (1974): Uniqueness of the Maximum Likelihood Estimates of the Parameters in an ARMA Model. IEEE Tr-AC 19, 769-774.

Åström, K.J. and B. Wittenmark (1973): On Self Tuning Regulators. Automatica 9, 185-199.

Bohlin, T. (1976): Four Cases of Identification of Changing Systems. In R.K. Mehra and D.G. Lainiotis: System Identification: Advances and Case Studies. Academic Press, New York.

Borisson, U. (1975): Self-Tuning Regulators - Industrial Application and Multivariable Theory. TFRT-1010, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden.

Box, G.E.P. and G.M. Jenkins (1970): Time Series Analysis: Forecasting and Control. Holden Day, San Francisco.

Breiman, L. (1968): Probability. Addison-Wesley, Reading.

Brown, R.G. (1963): Smoothing, Forecasting and Prediction of Discrete Time Series. Prentice Hall, Englewood Cliffs, NJ

Chung, K.L. (1968): A Course in Probability Theory. Harcourt, Brace & World.

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

Doncarli, C. (1977): Sur L'Identification des Processus Stochastiques Multivariables. Thèse de Docteur es Sciences, Ecole National Superieure de Mechanique, L'Universite de Nantes, France.

Gikhman, I.I. and A.V. Skorokhod (1969): Introduction to the Theory of Random Processes. Saunders, Philadelphia.

Gustavsson, I. (1969): Parametric Identification on Multiple Input, Single Output Linear Dynamic Systems. TFRT-3012, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden.

Holst, J. (1974): On the Use of Self-Tuning Predictors for the Prediction of Power Load (in Swedish). TFRT-3119, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden.

Holst, J. (1977): Multistep Prediction - One Possible Simplification of the Calculations. In preparation.

Jazwinski, A.H. (1970): Stochastic Processes and Filtering Theory. Academic Press, New York.

Kashyap, R.L. and A.R. Rao (1973): Real Time Recursive Prediction of River Flows. Automatica 9, 175-183.

Landau, I.D. (1976): Lectures in Lund.

Ljung, L. (1975): Theorems for the Asymptotic Analysis of Recursive Stochastic Algorithms. TFRT-3096, Dept of Automatic Control, Lund Institute of Technology, Lund, Sweden.

Ljung, L. (1976a): Analysis of Recursive Stochastic Algorithms. TFRT-7097, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden. To be published in IEEE Tr-AC 22, Aug 1977.

Ljung, L. (1976b): On Positive Real Transfer Functions and the Convergence of Some Recursive Schemes. TFRT-3138, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden. To be published in IEEE Tr-AC 22, Aug 1977.

Ljung, L., T. Söderström and I. Gustavsson (1975): Counterexample to General Convergence of a Commonly Used Recursive Identification Method. IEEE Tr-AC 20, 643-652.

Ljung, L. and B. Wittenmark (1974): Asymptotic Properties of Self-Tuning Regulators. TFRT-3071, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden.

Panuska, V. (1969): An Adaptive Recursive Least-Squares Identification Algorithm. Proc. 8th IEEE Symposium on Adaptive Processes.

Söderström, T., L. Ljung and I. Gustavsson (1974): A Comparative Study of Recursive Identification Methods. TFRT-3085, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden.

van der Waerden, B.L. (1966): Algebra - Erster Teil. Springer, Berlin.

Wittenmark, B. (1974): A Self-Tuning Predictor. IEEE Tr-AC 19, 848-851.

Wittenmark, B. (1975): Stochastic Adaptive Control Methods - A Survey. Int. J. Control 21, 705-730.

Young, P. (1970): An Extension to the Instrumental Variable Method for Identification of a Noisy Dynamic Process. Report CN/70/1, Department of Engineering, University of Cambridge, Cambridge, Great Britain.

Young, P. (1976): Some Observations on Instrumental Variable Methods of Time-Series Analysis. Int. J. Control 23, 593--612.

APPENDIX A

PROOFS OF THEOREMS 1, 2 and 3

Proof of Theorem 1 (p. 39). The time argument in the parameter estimates is omitted.

In the proof the parameter and data vectors for the four different algorithms are mutually compared. First it will be shown that the algorithms 1 and 3 in fact are identical, i.e. the S_{13} matrix is an identity matrix.

Algorithms 1 and 3. In the algorithms 1 and 3 the data and parameter vectors are given by (cf. Table 2):

$$\theta_1 = [a_1,\ldots,a_n,\ g_0,\ldots,g_{n-1}]^T$$

$$\varphi_1(t+1) = [-\hat{y}(t|t-1),\ldots,-\hat{y}(t-n+1|\,t-n),$$

$$e_1(t),\ldots,e_1(t-n+1)]^T$$

$$\theta_3 = [h_1,\ldots,h_n,\ g_0,\ldots,g_{n-1}]^T$$

$$\varphi_3(t+1) = [-\hat{y}(t|\,t-1),\ldots,-\hat{y}(t-n+1|\,t-n),$$

$$\varepsilon(t),\ldots,\varepsilon(t-n+1)]^T$$

The polynomial H in algorithm 3 is equal to the polynomial A (cf Chapter 2) which means that the parameters to estimate are the same in the algorithms 1 and 3. The estimate $\{e_1(t)\}$ of $\{e(t)\}$ is calculated at time t as

$$e_1(t) = \frac{A(t-1;q^{-1})}{C(t-1;q^{-1})}\,y(t)$$

where the parameter estimates from time $t-1$ are used. The polynomial C is simply the sum of the estimated polynomials

A and G. Thus

$$e_1(t) = \frac{A(t-1;q^{-1})}{C(t-1;q^{-1})}\, y(t) = y(t) - \frac{G(t-1;q^{-1})}{A(t-1;q^{-1})}\, e_1(t-1)$$

However, the prediction $\hat{y}(t|t-1)$ is

$$\hat{y}(t|t-1) = \frac{G(t-1;q^{-1})}{A(t-1;q^{-1})}\, e_1(t-1)$$

which gives

$$e_1(t) = y(t) - \hat{y}(t|t-1) = \varepsilon(t)$$

The sequence $\{e_1(t)\}$ is thus a sequence of prediction errors. Hence also the data vectors in the algorithms 1 and 3 are the same, thus giving identical algorithms.

Thus only the relationships between the algorithms 0, 2 and 3 remain to be discussed. The parameter and data vectors are:

$$\theta_0 = [a_1^0,\ldots,a_n^0,c_1^0,\ldots,c_n^0]^T$$

$$\varphi_0(t+1) = [-y(t),\ldots,-y(t-n+1),$$
$$\varepsilon^0(t),\ldots,\varepsilon^0(t-n+1)]^T$$

$$\theta_2 = [c_1^2,\ldots,c_n^2,g_0^2,\ldots,g_{n-1}^2]^T$$

$$\varphi_2(t+1) = [-\hat{y}^2(t|t-1),\ldots,-\hat{y}^2(t-n+1|t-n),$$
$$y(t),\ldots,y(t-n+1)]^T$$

and

$$\theta_3 = [a_1^3,\ldots,a_n^3,g_0^3,\ldots,g_{n-1}^3]^T$$

$$\varphi_3(t+1) = [-\hat{y}^3(t|t-1),\ldots,-\hat{y}^3(t-n+1|t-n),$$
$$\varepsilon^3(t),\ldots,\varepsilon^3(t-n+1)]^T$$

where also the components in the vectors are indexed.

Algorithms 0 and 2. Introduce for the following discussion of the connections between the algorithms 0 and 2 the data and parameter vectors

$$\theta_{2,0} = [a_1^2,\ldots,a_n^2,c_1^2,\ldots,c_n^2]^T$$

$$\varphi_{2,0}(t+1) = [-y(t),\ldots,-y(t-n+1),$$
$$\varepsilon^2(t),\ldots,\varepsilon^2(t-n+1)]^T$$

i.e. the vectors with the same parameters and data elements as used in the algorithm 0 but calculated from the algorithm 2 using the identity (2.6) on the estimated parameters. It is then easily seen that

$$\varphi_2(t) = \begin{bmatrix} I_n & I_n \\ -I_n & 0_n \end{bmatrix} \varphi_{2,0} = Q\,\varphi_{2,0} \qquad (A.1)$$

where I_n is an nxn identity matrix and 0_n is an nxn zero matrix. The identity (2.6) gives

$$\theta_2 = \begin{bmatrix} 0_n & I_n \\ -I_n & I_n \end{bmatrix} \theta_{2,0} = [Q^T]^{-1}\theta_{2,0} \qquad (A.2)$$

The equation (3.11) for the P matrix in the least squares algorithm is

$$P_2(t) = \left[\sum_{t_0}^{t} \varphi_2(s)\varphi_2^T(s)\right]^{-1} = \left[\sum_{t_0}^{t} Q\,\varphi_{2,0}(s)\varphi_{2,0}^T(s)Q^T\right]^{-1} =$$

$$= [Q^T]^{-1}P_{2,0}(t)Q^{-1}$$

which for the parameter gain (3.10) means

$$K_2(t) = \frac{P_2(t-1)\varphi_2(t)}{1 + \varphi_2^T(t)P_2(t-1)\varphi_2(t)} = [Q^T]^{-1}K_{2,0}(t)$$

where

$$K_{2,0}(t) = \frac{P_{2,0}(t-1)\varphi_{2,0}(t)}{1 + \varphi_{2,0}^T(t)P_{2,0}(t-1)\varphi_{2,0}(t)}$$

and

$$P_{2,0}(t) = \left[\sum_{t_0}^{t} \varphi_{2,0}(i)\varphi_{2,0}^T(i)\right]^{-1}$$

This gives an algorithm for $\theta_{2,0}(t)$ via the algorithm (3.10) which is used for $\theta_2(t)$

$$\theta_{2,0}(t) = Q^T\theta_2(t) = Q^T[\theta_2(t-1) + K_2(t)\{y(t)-\varphi_2^T(t)\theta_2(t-1)\}]$$

$$= \theta_{2,0}(t-1) + K_{2,0}(t)[y(t) - \varphi_{2,0}^T(t)\theta_{2,0}(t-1)]$$

This is the equation for the parameters in the algorithm 0. Thus if $\theta_{2,0}(t_0) = \theta_0(t_0), P_{2,0}(t_0) = P_0(t_0)$ and the initial values of the y- and ε-processes are the same in the two algorithms then for the same realization of the process $\{y(t)\}$

$$\theta_{2,0}(t) = \theta_0(t) \qquad t \geq t_0$$

which means that

$$\theta_2(t) = [Q^T]^{-1}\theta_0(t) \qquad t \geq t_0; S_{02} = Q^T$$

Algorithms 0 and 3. If the parameter and data vectors

$$\theta_{3,0} = [a_1^3,\ldots,a_n^3,c_1^3,\ldots,c_n^3]^T$$

$$\varphi_{3,0}(t+1) = [-y(t),\ldots,-y(t-n+1),$$
$$\varepsilon^3(t),\ldots,\varepsilon^3(t-n+1)]^T$$

are introduced it is seen that

$$\theta_3(t) = [P^T]^{-1}\theta_0(t) \qquad t \geq t_0; \qquad S_{03} = P^T$$

where

$$P = \begin{bmatrix} I_n & I_n \\ 0_n & I_n \end{bmatrix}, [P^T]^{-1} = \begin{bmatrix} I_n & 0_n \\ -I_n & I_n \end{bmatrix}$$

The transformation matrix between the algorithms 2 and 3 is easily found from the transformation matrices between the algorithms 0 and 2 and the algorithms 0 and 3. All the considered structures are thus knitted together.

□

PROOF OF THEOREM 2 (p.41)

This theorem is proven in the same manner as Theorem 1. A change of variables in one of the algorithms is done in order to get a sequence of estimates which is identical to the sequence of estimates achieved from another algorithm.

If $n < k$ the discussion in Chapter 2 showed that these three algorithms actually were equal. It is thus here assumed that $n \geq k$.

If the transformed data and parameter vectors $\theta_{4,2}$, $\varphi_{4,2}(t)$, $\theta_{5,2}$ and $\varphi_{5,2}$ are introduced in a manner similar to that in the proof of Theorem 1, it is seen that

$$\theta_5 = P\ \theta_{5,2}$$

$$\varphi_5(t) = [P^T]^{-1} \varphi_{5,2}(t)$$

with

$$P = \left[\begin{array}{cc|c} 0_{n-k+1,k-1} & I_{n-k+1} & \\ 0_{k-1} & 0_{k-1,n-k+1} & -I_n \\ \hline & I_n & 0_n \end{array}\right]$$

$$[P^T]^{-1} = \left[\begin{array}{c|cc} 0_n & & -I_n \\ \hline & 0_{k-1,n-k+1} & 0_{k-1} \\ I_n & I_{n-k+1} & 0_{n-k+1,k-1} \end{array}\right]$$

where I_p is a pxp identity matrix, 0_p a pxp zero matrix and $0_{p,m}$ a pxm zero matrix. It is also seen that

$$\theta_4 = T\ \theta_{4,2}$$

$$\varphi_4(t) = [T^T]^{-1} \varphi_{4,2}(t)$$

where

$$T = \left[\begin{array}{c|cc} & 0_{k-1,n-k+1} & 0_{k-1} \\ I_n & -I_{n-k+1} & 0_{n-k+1,k-1} \\ \hline 0_n & & I_n \end{array}\right]$$

$$[T^T]^{-1} = \left[\begin{array}{cc|c} & I_n & 0_n \\ \hline 0_{n-k+1,k-1} & I_{n-k+1} & \\ 0_{k-1} & 0_{k-1,n-k+1} & I_n \end{array}\right]$$

In the same way as in Theorem 1 it is then found that $Q_{52} = P$ and $Q_{42} = T$.

□

PROOF OF THEOREM 3 (p.42)

The proof is similar in structure to the proof of Theorem 1. It aims at showing that the parameter vector in algorithm 4 and the corresponding parameter vector calculated from the estimates obtained in algorithm 3 are not based on the same information. The time argument on the parameter estimates is omitted.

The data and parameter vectors for algorithms 3 and 4 are

$$\theta_3 = [h_1^3,\ldots,h_{n+k-1}^3,g_0^3,\ldots,g_{n-1}^3]^T$$

$$\varphi_3(t+k) = [-\hat{y}^3(t+k-1|t-1),\ldots,-\hat{y}^3(t-n+1|t-n-k+1),$$
$$\varepsilon^3(t),\ldots,\varepsilon^3(t-n+1)]^T$$

$$\theta_4 = [h_1^4,\ldots,h_n^4,g_0^4,\ldots,g_{n-1}^4]^T$$

$$\varphi_4(t+k) = [-\hat{y}^4(t+k-1|t-1),\ldots,-\hat{y}^4(t+k-n|t-n),$$
$$\varepsilon^4(t),\ldots,\varepsilon^4(t-n+k),$$
$$y(t-n+k-1),\ldots,y(t-n+1)]^T$$

Introduce the data vector

$$\tilde{\varphi}_3(t+k) = [-\hat{y}^3(t+k-1|t-1),\ldots,-\hat{y}^3(t+k-n|t-n),$$
$$\varepsilon^3(t),\ldots,\varepsilon^3(t+k-n),$$
$$y(t-n+k-1),\ldots,y(t-n+1),0,\ldots,0]^T$$

with k-1 zeroes in the bottom of the vector, and the first 2n components of the vector the same as the 2n components of φ_4. It is then easily seen that

$$\tilde{\varphi}_3(t) = \left[\begin{array}{c|cc|c} I_n & 0_n & & 0_{n,k-1} \\ \hline 0_n & 0_{n-k+1,k-1} & I_{n-k+1} & 0_{n-k+1,k-1} \\ & -I_{k-1} & 0_{k-1,n-k+1} & I_{k-1} \\ \hline & 0_{k-1,2n+k-1} & & \end{array}\right] \varphi_3(t) = T\,\varphi_3(t)$$

Introduce also the parameter vector

$$\tilde{\theta}_3 = [h_1^3,\ldots,h_n^3,g_0^3,\ldots,g_{n-1}^3,0,\ldots,0]^T$$

When studying the connections between the $\tilde{\theta}_3$ and θ_3 parameters the identity (2.6) is used. This means that there is a certain amount of arbitraryness in the choice of transformation of the last k-1 parameters in θ_3. The transformation is

$$\tilde{\theta}_3 = \left[\begin{array}{c|cc|c} I_n & 0_{n,n+k-1} & & \\ \hline 0_n & 0_{n-k+1,k-1} & I_{n-k+1} & 0_{n-k+1,k-1} \\ & A & 0_{k-1,n-k+1} & B \\ \hline & 0_{k-1,2n+k-1} & & \end{array}\right] \theta_3 = \tilde{P}\,\theta_3$$

where A and B are two $(k-1)\times(k-1)$ matrices that could be chosen

$$A = I_{k-1}(\alpha-1)$$
$$B = I_{k-1}\alpha$$

Now choose α so that P is the Moore-Penrose pseudo-inverse of T^T (refer to Albert (1972) on pseudo-inverses). This means that $\alpha = 1/2$ and

$$P = \tilde{P}_{\alpha=\frac{1}{2}} = \left[\begin{array}{c|cc|c} I_n & 0_{n,n+k-1} & & \\ \hline 0_n & 0_{n-k+1,k-1} & I_{n-k+1} & 0_{n-k+1,k-1} \\ & -\frac{1}{2}I_{k-1} & 0_{k-1,n-k+1} & \frac{1}{2}I_{k-1} \\ \hline & 0_{k-1,2n+k-1} & & \end{array}\right]$$

and also that

$$\theta_3 = T^T\tilde{\theta}_3 = P^+\tilde{\theta}_3$$

where A^+ denotes the pseudo-inverse of A. With this choice of P the T matrix can be seen as

$$T = T\ P^T T$$

where $P^T T$ is an orthogonal projection on the image of T^T. This P matrix also gives

$$\hat{y}^3(t+k|t) = \varphi_3^T(t+k)\theta_3 = \tilde{\varphi}_3^T(t+k)[T^T]^+\theta_3 + x^T[I - T^T[T^T]^+]\theta_3 =$$

$$= \tilde{\varphi}_3^T(t+k)\tilde{\theta}_3 = \tilde{\hat{y}}(t+k|t) \qquad (A.3)$$

where the equation

$$\varphi_3(t) = T^+\tilde{\varphi}_3(t) + [I-T^+T]x$$

for some x have been used. This equation thus partitions the data used in algorithm 3 into the data used in algorithm 4 and a rest which is nonzero. This latter part contains

$$\varepsilon(t-n+k-1),\ldots\varepsilon(t-n+1),\hat{y}(t-n+k-1|t-n-1),\ldots,$$
$$\hat{y}(t-n+1|t-n-k+1)$$

The equality (A.3) between the two predictions implies equality between the prediction errors. Now consider the algorithm for $\tilde{\theta}_3$

$$\tilde{\theta}_3(t) = P\theta_3(t) = P\theta_3(t-1) + PK_3(t)[y(t) - \varphi_3^T(t)\theta_3(t-1)]$$

$$= \tilde{\theta}_3(t-1) + PK_3(t)[y(t) - \tilde{\varphi}_3^T(t)\tilde{\theta}_3(t-1)]$$

What is left to discuss is, cf (3.10)

$$PK_3(t) = P\ P_3(t)\varphi_3(t) =$$

$$= P\ P_3(t)T^+\tilde{\varphi}_3(t) + P\ P_3(t)[I-T^+T]x =$$

$$= P\ P_3(t)P^T\tilde{\varphi}_3(t) + P\ P_3(t)[I-P^TT]x$$

It is easily seen that

$$P\ P_3(t)[I-P^TT]x \neq 0$$

i.e. the data in algorithm 3 which are not used in algorithm 4 are also contained in $PK_3(t)$. The algorithm for $\tilde{\theta}_3$ is thus not similar in the meaning discussed above to the algorithm for θ_4. Thus there exists no data- and time-invariant relationship between the algorithm 3 and any of the algorithms 2, 4 or 5.

□

APPENDIX B

PROOF OF THEOREM 4

The theorem by Ljung (Ljung (1975)) which is used in the proof of Theorem 4 is quoted for reference. D_R is an open connected subset of $D_{s,i}$ (5.2).

Theorem (Ljung). Consider the algorithm

$$x(t) = x(t-1) + \gamma(t)Q\big(t,x(t-1),\psi(t)\big)$$

where the observation $\psi(t)$ is obtained from a linear dynamical system

$$\psi(t) = A\big(x(t-1)\big)\psi(t-1) + B\big(x(t-1)\big)\, e(t)$$

under the following assumptions $\big(N(\bar{z},\alpha) = \{z \mid |z-\bar{z}| < \alpha\}\big)$

1) $\{e(t)\}$ is a sequence of independent random variables (not necessarily stationary or with zero means).

2) $E|e(t)|^p$ exists and is bounded in t for each $p > 1$.

3) The function $Q(t,x,\psi)$ is Lipschitz continuous in x and ψ :
$|Q(t,x_1,\psi_1) - Q(t,x_2,\psi_2)| < K_1(\bar{x},\bar{\psi},\rho,v)\{|x_1-x_2| + |\psi_1-\psi_2|\}$
for $x_i \in N(\bar{x},\rho)$ for some $\rho = \rho(\bar{x}) > 0$ and all $\bar{x} \in D_R$. $\psi_i \in N(\bar{\psi},v)$.

4) $|K_1(\bar{x},\psi_1,\rho,v_1) - K_1(\bar{x},\psi_2,\rho,v_2)| < K_2(\bar{x},\bar{\psi},\rho,\bar{v},w) \cdot \{|\psi_1-\psi_2| + |v_1-v_2|\}$
for $\psi_i \in N(\bar{\psi},w)$ and $v_i \in N(\bar{v},w)$.

5) The matrix functions $A(.)$ and $B(.)$ are Lipschitz continuous in D_R.

6) $\lim_{t\to\infty} E\, Q(t,\bar{x},\psi(t,\bar{x}))$ exists for $\bar{x} \in D_R$ and is denoted by $h(\bar{x})$. The expectation is over $\{e(t)\}$.

7) For $\bar{x} \in D_R$ the random variables $Q(t,\bar{x},\psi(t,\bar{x}))$, $K_1(\bar{x},\psi(t,\bar{x}),\rho(\bar{x}),v(t,\bar{x}))$ and $K_2(\bar{x},\psi(t,\bar{x}),\rho(\bar{x}),v(t,\bar{x}),v(t,\bar{x}))$ have bounded p-moments for all $p > 1$.

8) $\sum_1^\infty \gamma(t) = \infty$.

9) $\sum_1^\infty \gamma(t)^p < \infty$ for some $p > 1$.

10) $\{\gamma(t)\}$ is a decreasing sequence.

11) $\lim_{t\to\infty} \sup(1/\gamma(t) - 1/\gamma(t-1)) < \infty$.

Assume that $x^* \in D_R$ has the property

$$P(x(t) \to N(x^*,\rho)) > 0 \quad \text{for all} \quad \rho > 0.$$

Then

$$h(x^*) = 0$$

Further suppose that

$Q(t,x^*,\psi(t,x^*))$ has a covariance matrix that is bounded from below by a strictly positive definite matrix (B.1)

and that

$E\, Q(t,x,\psi(t,x))$ is continuously differentiable w.r.t. x in a neighbourhood of x^* and the derivatives converge uniformly in this neighbourhood as t tends to infinity. (B.2)

Then

$$H(x^*) = \frac{d}{dx} h(x)\Big|_{x=x^*} \text{ has all eigenvalues in the left half plane (including the imaginary axis).} \tag{B.3}$$

Proof of Theorem 4 (p. 46) Consider first the algorithm 2 and omit the subscript. The estimates are collected in a vector

$$x(t) = \left(\theta(t)^T, \operatorname{col}\left(R(t)\right)^T\right)^T = \left(\hat{\theta}^T, \operatorname{col}\left(R(t)\right)^T\right)^T$$

where $\operatorname{col}\left(R(t)\right)$ is a column vector containing the elements in $R(t)$. As this matrix is symmetric it is sufficient to include equal elements once. The algorithm (3.12) is then described by

$$x(t) = x(t-1) + \gamma(t)\, Q\left(t, x(t-1), \psi(t)\right)$$

with

$$Q\left(t, x(t-1), \psi(t)\right) \triangleq \begin{pmatrix} Q_\theta \\ Q_R \end{pmatrix} = \begin{pmatrix} \dfrac{R^{-1}(t-1)\varphi(t)\hat{w}(t)}{1+\frac{1}{t}\left(\varphi^T(t)R^{-1}(t-1)\varphi(t)-1\right)} \\ \operatorname{col}\left(\varphi(t)\varphi^T(t) - R(t-1)\right) \end{pmatrix} \tag{B.4}$$

where

$$\hat{w}(t) = \varepsilon(t) + \varphi^T(t)\left(\theta(t-k) - \theta(t-1)\right)$$

The observation vector $\psi(t)$ contains in addition to the elements of $\varphi(t)$ also the state vectors in the data and prediction generating systems. Shifted values of the elements in $\varphi(t)$ are also included. The generation of $\psi(t)$ is given by

$$\psi(t) = A(x(t-1))\, \psi(t-1) + B\, e(t) \tag{B.5}$$

In the matrix $A(x(t-1))$ the parameter estimates enter either linearly or bilinearly. The poles of this system are in the zeroes of the polynomial A, in the zeros of the estimated polynomial C and in the origin.

The conditions 1 and 2 are fulfilled due to the noise assumptions. It is shown in Ljung (1976a) that the conditions 3 and 4 are satisfied for this function Q (B.4) in the open area $D_R = \{x | R>0\}$ e.g. with

$$K_1(x,\psi,\rho,v) = (|\theta| + \rho)(1 + |\psi| + v)^2 / (1 - \rho|R^{-1}|)^2$$

and

$$K_2(x,\psi,\rho,v,w) = (|\theta| + \rho)(|\psi| + 2w + v) / (1 - \rho|R^{-1}|)^2$$

for $\rho = \rho(x) < 1 / |R^{-1}|$. It should be noted that these regularity conditions are not fulfilled if $G(\theta^*) = E\,\varphi(t,\theta^*)\,\varphi^T(t,\theta^*)$ has any eigenvalue equal to zero since then Q_θ might increase without bound. This is discussed in Ljung (1975). The matrix A in (B.5) is clearly Lipschitz continuous as needed in condition 5.

To handle condition 6 define for fixed $\bar{x} \in D_R$

$$\bar{\psi} = \psi(t,\bar{x}) = A(\bar{x})\ \psi(t,\bar{x}) + B\ e(t)$$

Since $e(t)$ is stationary $\psi(t,\bar{x})$ will approach stationarity exponentially. The expectations $E\,\varphi(t,\bar{\theta})\varepsilon(t,\bar{\theta})$ and $E\,\varphi(t,\bar{\theta})\varphi^T(t,\bar{\theta})$ thus asymptotically tend to the functions

$$f(\bar{\theta}) = \lim_{t\to\infty} E\,\varphi(t,\bar{\theta})\ \varepsilon(t,\bar{\theta})$$

$$G(\bar{\theta}) = \lim_{t\to\infty} E\,\varphi(t,\bar{\theta})\ \varphi^T(t,\bar{\theta})$$

where the elements of $\varphi(t,\bar{\theta})$ and $\varepsilon(t,\bar{\theta})$ are asymptotically stationary since $\psi(t,\bar{\theta})$ is.

The function $h(\bar{x})$ which is required in condition 6 is thus defined as

$$h(\bar{x}) = \lim_{t\to\infty} EQ(t,\bar{x},\psi(t,\bar{x})) = \begin{pmatrix} \bar{R}^{-1} f(\bar{\theta}) \\ \text{col } (G(\bar{\theta}) - \bar{R}) \end{pmatrix}$$

The ψ-vector and thus also the proposed scalars K_1 and K_2 have bounded moments for all $p > 1$ since $\psi(t)$ is generated by $\{e(t)\}$. This is required in condition 7. As $\gamma(t)$ in the algorithm is $1/t$ the conditions 8, 9, 10 and 11 are all fulfilled.

Thus the conditions for the first part of the theorem by Ljung are fulfilled and $h(x^*) = 0$, i.e.

$$f(\theta^*) = 0$$

$$R^* = G(\theta^*)$$

It remains to be shown that among the stationary points to the ordinary differential equation (5.4) only those which are connected to a stable linearization are possible convergence points.

If the equation (5.4) is linearized around the stationary point $x^* = \left(\theta^{*T}, \text{col}^T(G(\theta^*))\right)^T$ the resulting differential equation is

$$\frac{d}{d\tau}(\theta-\theta^*) = G(\theta^*)^{-1} f_\theta(\theta^*)(\theta-\theta^*)$$

$$\frac{d}{d\tau}\text{col}(R-G(\theta^*)) = -\text{col}(R-G(\theta^*)) + \frac{d}{d\theta_i}\{\text{col } G(\theta)\}\Big|_{\theta-\theta^*} (\theta-\theta^*)$$

(B.6)

where

$$f_\theta(\theta^*) = \frac{d}{d\theta} f(\theta)\Big|_{\theta=\theta^*}$$

It is thus clear that the stability properties of the linearized differential equation (B.6) are determined by

$$G(\theta^*)^{-1} f_\theta(\theta^*)$$

since the system matrix of the linearized system is block triangular with a negative unit matrix in the lower diagonal block. In the following thus only the Q_θ part of the Q vector is discussed. As only stable generation of $\psi(t,\bar{x})$ in (B.5) is studied, possible transients in $\psi(t,\bar{x})$ are neglected and henceforth it is assumed to be stationary. Then

$$Q_\theta(t,\bar{x},\psi(t,\bar{x})) = \bar{R}^{-1}\,\bar{\varphi}\,\bar{\varepsilon} - \frac{1}{t}\,\frac{[\bar{\varphi}^T\,\bar{R}^{-1}\,\bar{\varphi} - 1]\,\bar{R}^{-1}\,\bar{\varphi}\,\bar{\varepsilon}}{1 + \frac{1}{t}[\bar{\varphi}^T\,\bar{R}^{-1}\,\bar{\varphi} - 1]} =$$

$$\triangleq \bar{R}^{-1}\,\bar{\varphi}\,\bar{\varepsilon} - S(t,\bar{x},\psi(t,\bar{x})) \qquad \text{(B.7)}$$

where

$$\bar{\varphi} = \varphi(t,\bar{x}) \quad \text{and} \quad \bar{\varepsilon} = \varepsilon(t,\bar{x})$$

The covariance of Q_θ is

$$\text{Cov } Q_\theta = E[\bar{R}^{-1}\,\bar{\varphi}\,\bar{\varepsilon}]\,[\bar{R}^{-1}\,\bar{\varphi}\,\bar{\varepsilon}]^T + Z(t,\bar{x},\psi(t,\bar{x}))$$

Z is neglected since

$$Z(t,x^*,\psi(t,x^*)) = O(1/t) \qquad t \to \infty.$$

Thus

$$\text{Cov } Q_\theta(t,x^*,\psi(t,x^*)) =$$

$$E(R^{*-1}\varphi(t,x^*)\varepsilon(t,x^*))(R^{*-1}\varphi(t,x^*)\varepsilon(t,x^*))^T \geq T$$

where T is positive definite since $G(\theta^*)$ is supposed to be regular.

Finally, look at the differentiation of

$$E\, Q_\theta(t,x,\psi(t,x)) - \lim_{t\to\infty} E\, Q_\theta(t,x,\psi(t,x)) = -E\, S(t,x,\psi(t,x))$$

where S is given in (B.7). If the order of differentiation and integration are changed, then

$$\frac{d}{dx_i} E\, S(t,x,\psi(t,x)) = E\, \frac{1}{t} \frac{\bar{\varphi}^T R^{-1} \bar{\varphi} - 1}{1 + \frac{1}{t}[\bar{\varphi}^T R^{-1} \bar{\varphi} - 1]} \cdot \frac{d}{dx_i} \cdot R^{-1} \bar{\varphi} \bar{\varepsilon}$$

$$+ E\, \frac{1}{t} \frac{1}{[1 + \frac{1}{t}(\bar{\varphi} R^{-1} \bar{\varphi} - 1)]^2} \cdot R^{-1} \bar{\varphi} \bar{\varepsilon} \cdot \frac{d}{dx_i}[\bar{\varphi}^T R^{-1} \bar{\varphi} - 1].$$

When x_i is an element in $\mathrm{col}(R)$ the derivative clearly is continuous and it converges uniformly to zero as $t \to \infty$. When x_i is an element in θ the derivative also is continuous since $\frac{d}{d\theta_i}\varphi(t,x)$ depends continuously on x. As the moments of $\psi(t,x)$ are bounded according to the discussion above it follows that

$$\frac{d}{d\theta_i} E\, S(t,x,\psi(t,x))$$

converges uniformly to zero as $t \to \infty$.

The conditions for the second part of the theorem by Ljung are therefore fulfilled and the eigenvalues of

$$G(\theta^*)^{-1} f_\theta(\theta^*)$$

have nonpositive real parts. The theorem is thus proven for algorithm 2. The other algorithms are treated quite analogously.

□

APPENDIX C

Proof of Lemma 5 (p. 48)

Start with algorithm 2. The predictor is

$$\hat{y}(t+k|t) = \varphi^T(t+k)\ \theta$$

where the stars at ε, $\hat{y}$, φ, and θ are omitted as well as the subscript on φ and θ. If the polynomials

$$C(q^{-1}) = c_1^* q^{-1} + \ldots + c_n^* q^{-n}$$

and

$$G(q^{-1}) = g_0^* + \ldots + g_{n-1}^* q^{-n+1}$$

are introduced, the predictor could be written (cf. (2.1))

$$\hat{y}(t+k|t) = \frac{G}{1+C} y(t) = \frac{G}{1+C} \cdot \frac{C}{A} e(t)$$

$$\varepsilon(t) = y(t) - \hat{y}(t|t-k) = y(t) - \frac{q^{-k} G}{1+C} y(t) =$$

$$= \frac{1 + C - q^{-k} G}{1 + C} y(t) = \frac{1 + C - q^{-k} G}{1 + C} \cdot \frac{C}{A} e(t) \qquad (C.1)$$

Introduce the stochastic process

$$v(t) = \frac{1}{1+C} \cdot \frac{C}{A} e(t)$$

which is an ARMA (2n,n) process. Then

$$\begin{cases} \hat{y}(t+k|t) = Gv(t) \\ y(t) = (1+C)v(t) \\ \varepsilon(t) = (1 + C - q^{-k} G)\ v(t) \end{cases} \qquad (C.2)$$

Multiplying both the $\hat{y}$ and y equations with $\varepsilon(t+\tau)$ and taking mathematical expectations gives

$$r_{\varepsilon\hat{y}}(\tau) = E(\varepsilon(t+\tau)\hat{y}(t+k|t)) = E(\varepsilon(t+\tau)\ G\nu(t))$$

$$= g_0^*\ r_{\varepsilon\nu}(\tau) + \ldots + g_{n-1}^*\ r_{\varepsilon\nu}(\tau+n-1)$$

$$r_{\varepsilon y}(\tau) = E(\varepsilon(t+\tau)y(t)) = E(\varepsilon(t+\tau)(1+C)\nu(t))$$

$$= r_{\varepsilon\nu}(\tau) + c_1^*\ r_{\varepsilon\nu}(\tau+1) + \ldots + c_n^*\ r_{\varepsilon\nu}(\tau+n)$$

Since this parameter vector is a stationary point to the differential equation (5.4)

$$\begin{pmatrix} r_{\varepsilon y}(k) \\ \vdots \\ r_{\varepsilon y}(k+n-1) \\ r_{\varepsilon\hat{y}}(k+1) \\ \vdots \\ r_{\varepsilon\hat{y}}(k+n) \end{pmatrix} = \begin{pmatrix} 1 & c_1^* & \cdots & c_n^* & 0 & 0 \\ & & & & & 0 \\ 0 & & 1 & c_1^* & \cdots & c_n^* \\ 0 & g_0^* & \cdots & g_{n-1}^* & 0 & 0 \\ & & & & & 0 \\ 0 & & & g_0^* & \cdots & g_{n-1}^* \end{pmatrix} \begin{pmatrix} r_{\varepsilon\nu}(k) \\ \vdots \\ r_{\varepsilon\nu}(k+n-1) \\ r_{\varepsilon\nu}(k+n) \\ \vdots \\ r_{\varepsilon\nu}(k+2n-1) \end{pmatrix} = 0$$

Thus since the G and $1+C$ polynomials have no factors in common the matrix is regular (van der Waerden (1966)) giving

$$r_{\varepsilon\nu}(\tau) = 0 \qquad \tau = k,\ldots,k+2n-1 \qquad (C.3)$$

Multiply (C.1) with $\nu(t-\tau)$ and take mathematical expectation. Denote the coefficients in the $(1+C)A$ polynomial α_i $i = 1,\ldots,2n$ and in the $[1+C-q^{-k}G]C$ polynomial β_i $i = 1,\ldots,2n+k-1$. Then

$$r_{\varepsilon\nu}(\tau) + \alpha_1\ r_{\varepsilon\nu}(\tau-1) + \ldots + \alpha_{2n}\ r_{\varepsilon\nu}(\tau-2n) =$$

$$= r_{e\nu}(\tau) + \beta_1\ r_{e\nu}(\tau-1) + \ldots + \beta_{2n+k-1}\ r_{e\nu}(\tau-2n-k+1)$$

Thus with $\tau \geq 2n+k$ this equation together with (C.3) gives

$$r_{\varepsilon\nu}(\tau) = 0 \qquad \tau \geq k$$

The equation (C.2) then gives

$$r_{\varepsilon}(\tau) = 0 \qquad \tau \geq k$$

The prediction error process $\{\varepsilon(t)\}$ is thus a moving average process of order $k-1$. Denote it by

$$\varepsilon(t) = e(t) + f_1 e(t-1) + \ldots + f_{k-1} e(t-k+1) = F(q^{-1})\, e(t)$$

and (C.1) gives

$$\frac{1 + C - q^{-k} G}{1 + C} \cdot \frac{C}{A} = F$$

or

$$C = AF + q^{-k} \frac{G}{1+C} \cdot C \qquad (C.4)$$

Thus since $\frac{G\,C}{1+C}$ is a polynomial of degree $n-1$

$$\frac{G}{1+C} \cdot C = G$$

and (C.4) is the minimum variance identity (2.6). This gives

$$\hat{y}(t+k|t) = \frac{G}{1+C}\, y(t) = \frac{G}{C}\, y(t)$$

i.e. the minimum mean square error prediction process. When $i = 1$ or 3 the discussion is quite analogous. To show that the adaptive predictor in fact is the minimum mean square error predictor, consider again the structure 2. From (C.4) it follows that since G and $1+C$ do not have

any factors in common and $\frac{G}{1+C} \cdot C$ is a polynomial of degree n-1

$$C = 1 + C$$

and

$$G = G$$

Thus in this predictor also the resulting polynomials are the polynomials that would be obtained in the minimum mean square error predictor of the process if the system polynomials were known. An analog discussion can be performed for algorithm 3, which concludes the proof.

□

APPENDIX D

Proof of Theorem 7 (p. 53)

First calculate the eigenvalues to the matrices $K_2(\theta_2)$ and $K_3(\theta_3)$. Consider algorithm 2 with the characteristic equation

$$0 = \det\ (\lambda E\varphi_{2,M}\ \varphi_{2,M}^T + E\varphi_{2,M}\ \tilde{\varphi}_{2,M}^T) \qquad (D.1)$$

where $\varphi_{2,M}$ is given in Table 1 in Chapter 2. There exists a constant matrix B_2 such that $\varphi_{2,M}$ can be transformed as

$$\varphi_{2,M} = \left[-\hat{y}_M(t-1|t-k-1),\ldots,-\hat{y}_M(t-n|t-k-n),y(t-k),\ldots,y(t-k-n+1)\right]^T$$

$$= B_2\left[-\hat{y}_M(t-k|t-k-n),\ldots,-\hat{y}_M(t-k-n+1|t-k-n),e(t-k),\ldots,e(t-k-n+1)\right]^T$$

$$\hat{=}\ B_2\ \eta_{2,M}$$

$$\tilde{\varphi}_{2,M} \hat{=}\ B_2\ \tilde{\eta}_{2,M}$$

since, for example $(i=1,\ldots,n)$

$$\hat{y}_M(t-i|t-k-i) = \hat{y}_M(t-i|t-k-n) + f_k\ e(t-k-i) + \ldots + f_{k+n-1-i}\ e(t-k-n+1)$$

where f_j is the $(j+1)$:th coefficient in the impulse response

$$y(t) = \sum_{i=0}^{\infty} f_i\ e(t-i), \qquad f_0 = 1$$

and $\hat{y}_M(t-i|t-k-i)$, $i=1,\ldots,n$ as well as $y(t-i)$, $i=k,\ldots,k+n-1$ are linear combinations of the elements in $\eta_{2,M}$ (cf. Chapter 6). Thus (D.1) is rewritten as

$$0 = \det B_2 \cdot \det B_2^T \det(\lambda E\eta_{2,M}\, \eta_{2,M}^T + E\eta_{2,M}\, \tilde{\eta}_{2,M}^T)$$

where the elements in $\eta_{2,M}$ may be partitioned into two mutually uncorrelated parts. The application of Lemma 1 and the corollary to Lemma 2 in part III of this report gives the eigenvalues to $K_2(\theta_2)$

$$\begin{cases} -1 \quad \text{of multiplicity} \quad n \\ -\dfrac{1}{C(\alpha_i)} \quad \text{where} \quad A^f(\alpha_i) = 0; \quad i=1,\ldots,n \end{cases}$$

since the filter $H(q^{-1})$ that is required in Lemma 2 is

$$H(q^{-1}) = \frac{1}{C(q^{-1})}$$

as in the ELS case.

Now consider algorithm 3. Theorem 1 in part III of the report is directly applicable. The filter involved is also in this case

$$H(q^{-1}) = \frac{1}{C(q^{-1})}$$

as above. Thus the eigenvalues are

$$\begin{cases} -1 \quad \text{of multiplicity} \quad n+k-1 \\ -\dfrac{1}{C(\alpha_i)} \quad \text{where} \quad A^f(\alpha_i) = 0; \quad i=1,\ldots,n \end{cases}$$

The convergence condition then follows by application of Theorem 4.

□

Part II - Adaptive Short-Term Prediction of Power Load

ABSTRACT

The application of the adaptive k-step prediction methods to short-term prediction of the hourly load on a power network is considered. The load is partitioned into a residual and a nominal part. The residual load is modeled as an ARMA process. Different possibilities to represent and to model the nominal load are considered as well as a variety of possible prediction algorithms. The results of the predictions with the proposed adaptive k-step predictor compare favourably both with the prediction results from other published methods studied and with published prediction results.

1. Introduction

Prediction of power load is a part of planning and operation of power systems. The ultimate aim is to produce and to distribute power to the consumers reliably and efficiently. The prediction is needed for a variety of lead times, ranging from years down to fractions of an hour. The adaptive prediction methods presented in the first part of this report may be used, if necessary after some modifications, in any of these ranges. The current application, however, is on short-term load forecasting where the lead time ranges from one hour up to some days.

An accurate prediction in this time interval is needed for control and scheduling of the power plants, e.g. concerning smooth start up and shut down procedures. Prediction with these lead times is also needed as one of the inputs to load flow determinations, contingency tests etc.

The power load can be regarded as a nonstationary random process. It has a noticable seasonal pattern and a periodic structure where the main period is one week. It is influenced by e.g. industrial activity and long or short term weather conditions. The prediction method therefore preferably should be adaptive in order to take changes of this kind into account.

The inclusion of weather variables in the load prediction is questionable. There are some easily noticed problems. The weather predictions are often inprecise. The choice of relevant weather variables is difficult as well as the modeling of their influence on the power consumption. Furthermore, the response in power load to changes in weather conditions might be rather slow and therefore recognizable in past load data. It can then be handled by the adaptive mechanisms in the prediction algorithm.

On the other hand there is an obvious correlation between weather and power load, especially in areas where a substantial part of the load is domestic. Hence the inclusion of weather information in the prediction algorithm could be beneficial for the prediction result when the changes in the weather variables are large and faster then the speed of adaption of the algorithm.

Weather variables are used as the main input signal for prediction of power load in the papers by Dryar (1944) and Davies (1959). In the method proposed by Davies weather-load relationships are developed with a nonlinear regression method. Recent data are however not included in forming the functional relationships and the method can therefore not adapt to changing load patterns.

In the prediction method by Farmer, see Farmer (1963), Farmer and Potton (1966,1968) and Matthewman and Nicholson (1968), the load is decomposed in a nominal and a residual part. The latter is then expressed as a weighted sum of eigenfunctions from a Karhunen-Loeve expansion. The method is adaptive since the weighting of the eigenfuctions is done in real time. Weather data are not included in Farmer's own work. However, in Lijesen and Rosing (1971), where the residual part of the load is decomposed according to Farmer's method, the weighting coefficients are related to average weather conditions.

The periodic structure of the load is emphasized in the work by Christiaanse (1971). He uses a Fourier expansion with nine frequencies of the weekly load and updates the parameters using general exponential smoothing (Brown (1963)). Weather data are not included. This method has the same structure as a stationary Kalman filter with the gain matrix chosen ad hoc. If this matrix is determined by usual Kalman filtering technique from Christiaanse's original load model the prediction method by Sharma and

Mahalanabis (1974) results. Another Kalman filtering approach to the problem is taken by Toyoda, Chen and Inoue (1970).

Two predictions, one hourly prediction based on historical load data only and one peak load prediction based on weather data, are weighted together in the method by Gupta and Yamada (1972). The exponential smoothing technique is used for updating the elements in the nominal part of the load data prediction. The Least Squares method is used in the calculation of the weather-load model. The weighting coefficients are essentially inversly proportional to the estimated variance of the corresponding prediction error which means that the precision of the weather data may be less critical for the final prediction result.

In Galiana (1971) the load is partitioned into a nominal and a residual part. The nominal load is modeled as above, i.e. as a sum of trigonometrical functions. The residual part is represented as an ARMA process on canonical state space form with a nonlinear function of the deviation between the actual and a normal temperature as an input signal. The parameters in the model, including the steady state Kalman gain, are determined on historical data with a minimization technique. The model obtained is then used for prediction during typically a week.

The representation of the residual load as an AR or an ARMA process with or without explicitly included weather variables is common to the following methods. In Bohlin (1976) and Bohlin and Kamjou (1977) the nominal load is represented as a state vector, containing four profiles. It is updated on line using Kalman filtering technique. The parameters in the model of the residual load are also updated on line with a Least Squares algorithm. In Bohlin and Kamjou (1977) also weather variables are included. Their influence is found to be small and in general negative to the prediction result. In Keyhani and El-Abiad (1975) only one-step ahead prediction is considered.

The total load is modeled as an ARMA process with an additive constant and without weather information. The parameters may be updated in every time step.

Also in the current prediction method the load $\{y(t)\}$ is partitioned as

$$y(t) = y_r(t) + y_n(t)$$

where the residual load $\{y_r(t)\}$ is represented as an ARMA process and the parameters in the predictor are updated with the methods from part I of the report. Preliminary results are given in Holst (1974). The period of the nominal load $\{y_n(t)\}$ is a week. The timevarying character of the parameters is handled by introducing a weighting factor in the parameter estimation. Weather information is not included. If available, such signals however easily can be incorporated in the model structure.

The data used in the prediction experiments are authentical load data from 1973 obtained from the Swedish State Power Board. They are given in Holst (1977). Data from 1971 and 1972 are used in the calculation of one of the nominal load representations. Data are labelled "The Compensated Internal Consumption on the Swedish State Power Board Network" ("Statens Vattenfallsverks Korrigerade Egenförbrukning"). These data are also used in a study of some other prediction methods by Tyrén (1974).

In Chapter 2 different ways to handle the nominal load are discussed. Methods used to measure the goodness of the prediction are also treated. Chapter 3 contains an exposition of the algorithms used. In Chapter 4 the prediction results are given and in Chapter 5 this part of the report is summarized.

2. PRELIMINARIES

In this chapter the nominal load representation and the measuring of the goodness of the prediction are discussed.

2.1 NOMINAL LOAD REPRESENTATION

Data have an obvious periodic pattern, cf. Figure 1 where some weekly data profiles are shown. The load is quite different on weekdays compared to weekends but there are also differencies between the various weekday loads and between the Saturday and Sunday loads. Therefore the period of the nominal load is chosen as a week and not as a day which also might be possible.

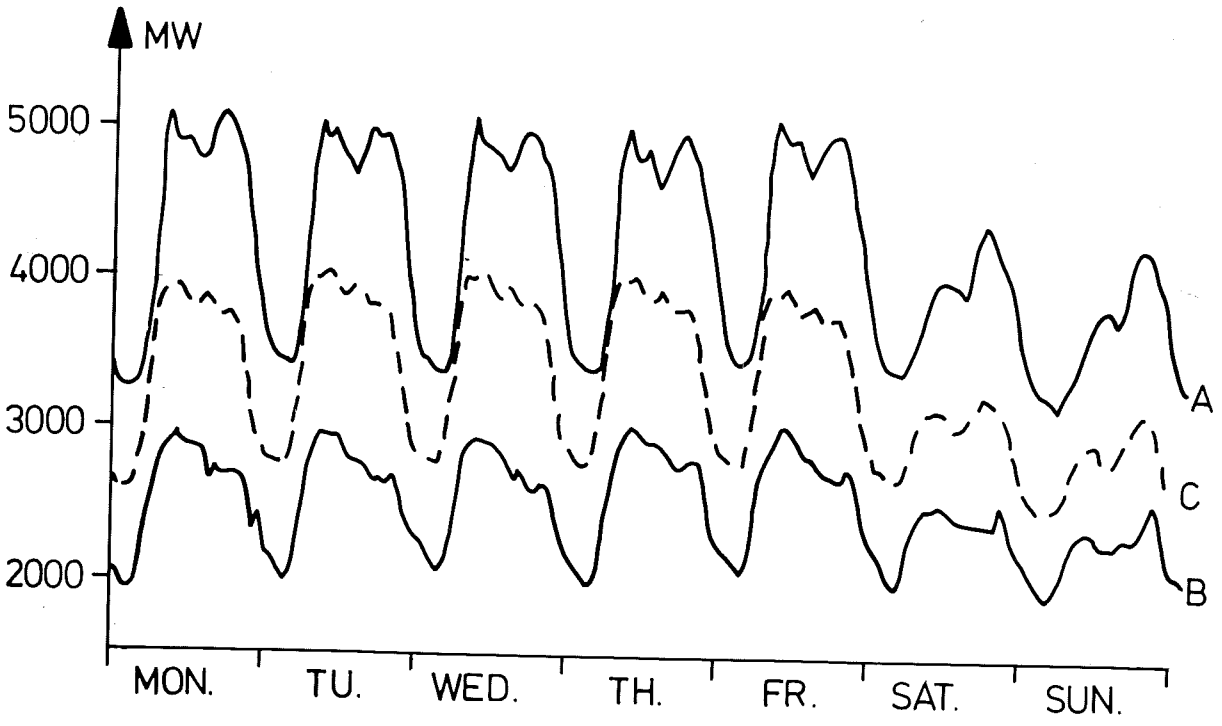


Figure 1 - Hourly power load. A - during a winter week, B - during a summer week, and C - mean value over 1973.

The variations of the weekly profile during a year are comparatively slow. They are handled by the profile updating. The nominal load is thus based on a vector containing 168 elements representing the load at each hour of the week.

In the simulations two different representations of the weekly profile have been used. The first of these is based on a mean value over the data from 1971 and 1972. This signal is shown in Figure 2. It was used in the experiments without updating. Random variations in the load are filtered greatly. Systematic variations of the weekly profile over the year are however also supressed.

The second representation of the profile is formed from the load at the corresponding time a week ago. Using this profile representation systematic variations as well as random disturbancies are contained in the signal. The updating of this weekly profile is somewhat more cautiously done if just a fraction of the measured

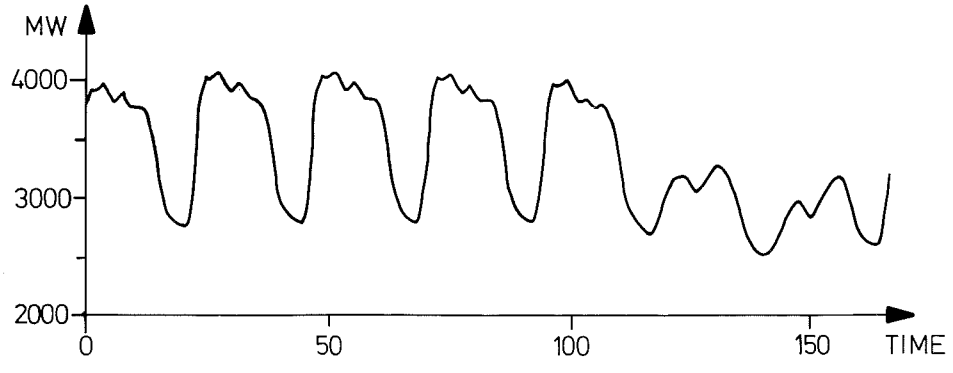


Figure 2 - The mean value of the weekly profile in load data. It is based on all data from 1971 and 1972.

load is used in the profile adjustment. A very simple exponential updating of the profile $\{v(t)\}$ has been used, i.e.

$$v(t) = \alpha\, y(t) + (1-\alpha)\, v(t-168) \tag{2.1}$$

The smaller value of α the more supression of both the systematic and random variations in the load. With $\alpha = 0.75$ the influence of a profile value is reduced to about 1% after three weeks. With $\alpha = 0.60$ the same reduction is acheived after five weeks.

In the simulations the nominal load has been modeled as

$$y_n(t) = Q(q^{-1})\, w(t)$$

where $\{w(t)\}$ is taken from either the fixed mean value profile from 1971 and 1972 or the value of the possibly exponentially updated profile a week ago. $Q(q^{-1})$ is a transfer function operator. It is discussed in the following chapter.

2.2 PERFORMANCE MEASURING

The goodness of the prediction could be expressed in relative as well as in absolute terms. When a relative measure is chosen the prediction error can be related either to peak load, as in e.g. Galiana (1971) or Bohlin (1976), to average load during the studied period, as in e.g. Christiaanse (1971) or to the load at the same time instant as when the error occurs as in e.g. Tyrén (1974).

In the current experiments one absolute and two relative measures have been used. The loss functions are

$$V_{MW} = \sum_{n_0+1}^{n} \varepsilon^2(t) \tag{2.2}$$

and

$$V_{\%} = \sum_{n_0+1}^{n} \left(\frac{\varepsilon(t)}{y(t)} \cdot 100 \right)^2 \tag{2.3}$$

where $\varepsilon(t)$ is the prediction error and $y(t)$ the load at time t. The goodness of the prediction is measured with either of the following values. The absolute measure is

$$s_{MW} = \sqrt{\frac{V_{MW}}{n-n_0}} \tag{2.4}$$

i.e. the estimated standard deviation of the error values. The first relative measure is the corresponding coefficient of variation, i.e.

$$s_{rel} = \frac{100}{y_{av}} \cdot s_{MW} = \sqrt{\frac{1}{n-n_0} \sum_{n_0+1}^{n} \left(\frac{\varepsilon(t)}{y_{av}} \cdot 100 \right)^2} \tag{2.5}$$

where y_{av} is the average value of the load for $t \in [n_0+1, n]$. The second relative measure is

$$s_{\%} = \sqrt{\frac{V_{\%}}{n-n_0}} \tag{2.6}$$

i.e. the estimated standard deviation of the instantaneous relative prediction errors. When using $s_{\%}$ or s_{rel} it is possible to compare prediction methods applied to different data series. However the impact on $s_{\%}$ of a certain prediction error will depend on when it occurred. This may be a drawback with the measure and it is illustrated in the prediction experiments.

3. PREDICTION ALGORITHMS

In the prediction algorithm the nominal and the residual loads are to be treated and the model parameters estimated. Modeling of the nominal load can be approached in some different manners which are studied below. In addition to the nominal load extension of the algorithms in Part I of the report, the current algorithms also must contain some facility to handle the timevarying parameters. This is treated below too. Finally, a simple prediction method, closely related to the scaling method is discussed. This latter method is used today at the Swedish State Power Board and it is described in Farmer and Potton (1968) and in Tyrén (1974).

3.1 TOTAL LOAD MODELING

3.1.1 K-step prediction algorithms

The total load is partitioned as

$$y(t) = y_r(t) + y_n(t) = \frac{C_1(q^{-1})}{A_1(q^{-1})} e(t) + y_n(t) \qquad (3.1)$$

where $A_1(q^{-1})$ and $C_1(q^{-1})$ are polynomials in the backward shift operator q^{-1}. They are supposed to be relatively prime, asymptotically stable and of order n_1. $\{e(t)\}$ is supposed to be a sequence of independent random variables with zero mean value and variance σ^2. Depending on the model of the nominal load $y_n(t)$ different versions of the adaptive prediction algorithm result.

Let $y_n(t)$ be modeled as

$$y_n(t) = \frac{B_2(q^{-1})}{A_2(q^{-1})} w(t) \tag{3.2}$$

where $A_2(q^{-1})$ and $B_2(q^{-1})$ are two relatively prime polynomials in the backward shift operator of order n_2. The constant term in $B_2(q^{-1})$ is not necessarily equal to one. $A_2(q^{-1})$ is asymptotically stable. $\{w(t)\}$ is any of the weekly profiles in the nominal load representation that was studied in the previous chapter. Thus the total load is

$$A(q^{-1})y(t) = C(q^{-1})e(t) + B(q^{-1})w(t) \tag{3.3}$$

where $A(q^{-1}) = A_1(q^{-1})A_2(q^{-1})$; $B(q^{-1}) = B_2(q^{-1})A_1(q^{-1})$; and $C(q^{-1}) = C_1(q^{-1})A_2(q^{-1})$, cf. Åström (1970) for the same kind of process modeling. The polynomials in (3.3) are all of order n.

The minimum mean square error k-step predictor $\hat{y}_M(t+k|t)$ of $y(t+k)$ is

$$\hat{y}_M(t+k|t) = (1-C(q^{-1}))\,\hat{y}_M(t+k|t) + G(q^{-1})\,y(t) + B(q^{-1})\,F(q^{-1})\,w(t+k) \tag{3.4}$$

or

$$\hat{y}_M(t+k|t) = (1-A(q^{-1})F(q^{-1}))\,\hat{y}_M(t+k|t) + G(q^{-1})\,\varepsilon_M(t) + B(q^{-1})\,F(q^{-1})\,w(t+k) \tag{3.5}$$

cf. Part I in the report or Åström (1970). The $F(q^{-1})$ and $G(q^{-1})$ polynomials are determined by the identity

$$C(q^{-1}) = A(q^{-1})\,F(q^{-1}) + q^{-k}\,G(q^{-1}) \tag{3.6}$$

$\{\varepsilon_M(t)\}$ is the sequence of prediction errors that occur when the optimal predictor is used. $\{w(t)\}$ is known

k steps ahead. Denote in the following the polynomials $A(q^{-1})F(q^{-1})$ and $B(q^{-1})F(q^{-1})$ by $H(q^{-1})$ and $K(q^{-1})$ with elements h_i; $i = 1,\ldots,n+k-1$ and k_j; $j = 0,\ldots,n+k-1$ respectively.

When the parameters in the $A(q^{-1})$, $B(q^{-1})$, and $C(q^{-1})$ polynomials are unknown any of the equations (3.3), (3.4), or (3.5) can serve as a starting point for an adaptive predictor. When the Least Squares method is used for the parameter estimation the resulting algorithms correspond

Table 1 - Parameter and data vectors used in the adaptive load prediction algorithms.

Algorithm	
0	$\theta_0(t) = [a_1(t),\ldots,a_n(t),c_1(t),\ldots,c_n(t),b_0(t),\ldots,b_n(t)]^T$
	$\varphi_0(t) = [-y(t-1),\ldots,-y(t-n),e_0(t-1),\ldots,e_0(t-n),w(t),\ldots,w(t-n)]^T$
2	$\theta_2(t) = [c_1(t),\ldots,c_n(t),g_0(t),\ldots,g_{n-1}(t),k_0(t),\ldots,k_{n+k-1}(t)]^T$
	$\varphi_2(t+k) = [-\hat{y}(t+k-1\|t-1),\ldots,-\hat{y}(t+k-n\|t-n),y(t),\ldots,y(t-n+1),w(t+k),\ldots,w(t-n+1)]^T$
3	$\theta_3(t) = [h_1(t),\ldots,h_{n+k-1}(t),g_0(t),\ldots,g_{n-1}(t),k_0(t),\ldots,k_{n+k-1}(t)]^T$
	$\varphi_3(t+k) = [-\hat{y}(t+k-1\|t-1),\ldots,-\hat{y}(t-n+1\|t-k-n+1),\varepsilon(t),\ldots,\varepsilon(t-n+1),w(t+k),\ldots,w(t-n+1)]^T$

to the algorithms 0, 2, and 3 in Part I of the report. Therefore they are denoted by 0, 2, and 3 also here. The data and the parameter estimates are collected in vectors $\varphi_i(t)$ and $\theta_i(t)$; $i = 0,2,3$ that are given in Table 1. The estimate of an element in $H(q^{-1})$ is called $h_i(t)$ and correspondingly for the other polynomials. $\{e_0(t)\}$ is the sequence of residuals obtained when the Extended Least Squares method is applied for estimation of the parameters in (3.3), i.e.

$$e_0(t) = y(t) - \varphi_0^T(t)\ \theta(t-1) \tag{3.7}$$

$\{\hat{y}(t+k|t)\}$ and $\{\varepsilon(t)\}$ are the prediction and prediction error sequences.

Algorithms. The algorithms consist of an estimation step and a prediction step.

Estimation: Estimate the parameters in the model, $i = 0,2,3$

$$y(t) - \varphi_i^T(t)\ \theta_i = r_i(t)$$

with a Least Squares algorithm, i.e.

$$\begin{cases} \theta_i(t) = \theta_i(t-1) + K_i(t)\left(y(t) - \varphi_i^T(t)\,\theta_i(t-1)\right) \\ K_i(t) = \dfrac{P_i(t-1)\ \varphi_i(t)}{1 + \varphi_i^T(t) P_i(t-1) \varphi_i(t)} \\ P_i(t) = P_i(t-1) - \dfrac{P_i(t-1)\varphi_i(t)\varphi_i^T(t)P_i(t-1)}{1 + \varphi_i^T(t) P_i(t-1) \varphi_i(t)} \end{cases} \tag{3.8}$$

$r_i(t)$ is the model error.

Prediction: Calculate in algorithm 0 estimates of the $F(q^{-1})$ and $G(q^{-1})$ polynomials in the system of equations

(3.6) applied to the estimated $A(q^{-1})$ and $C(q^{-1})$ polynomials. Use all the obtained polynomial estimates to compute the k-step prediction e.g. by (3.4) or (3.5).

Use in algorithm 2 or 3 the obtained parameter estimates, $\theta_i(t)$, to compute the prediction of $y(t+k)$

$$\hat{y}(t+k|t) = \varphi_i^T(t+k)\ \theta_i(t)$$

□

In the algorithms 2 and 3 the number of parameters to be determined in principle grows linearly with the number of steps to predict since the elements in $H(q^{-1})$ or $K(q^{-1})$ are estimated. Furthermore, the construction of the total load model (3.3) implies that the parameters in $A_1(q^{-1})$ and $A_2(q^{-1})$ influence at least two of the polynomials that are to be estimated. This might result in bad condition of the P matrix in (3.8). In particular, if

$$A_2(q^{-1}) = 1; \quad B_2(q^{-1}) = \gamma$$

then

$$K(q^{-1}) = \gamma\ H(q^{-1}).$$

The simulations have indicated that the total load model

$$y(t) = \frac{C_1(q^{-1})}{A_1(q^{-1})}\ e(t) + \gamma\ w(t) \qquad (3.9)$$

indeed is appropriate in this application. The algorithm is therefore modified in order to avoid the drawbacks mentioned. A similar algorithmic problem is treated in Wieslander (1969).

Since then $A(q^{-1}) = A_1(q^{-1})$ and $C(q^{-1}) = C_1(q^{-1})$, (3.9) can be rewritten as

$$y(t) = \gamma\, w(t) + (1-A)(y(t) - \gamma\, w(t)) + C\, e(t) \qquad (3.10)$$

The minimum mean square error k-step predictor of (3.10) is

$$\hat{y}_M(t+k|t) = \gamma\, w(t+k) + (1-C)(\hat{y}_M(t+k|t) - \gamma\, w(t+k)) + \\ + G(y(t) - \gamma\, w(t)) \qquad (3.11)$$

or

$$\hat{y}_M(t+k|t) = \gamma\, w(t+k) + (1-AF)(\hat{y}_M(t+k|t) - \gamma\, w(t+k)) + \\ + G\, \varepsilon_M(t) \qquad (3.12)$$

The polynomials $F(q^{-1})$ and $G(q^{-1})$ are calculated by (3.6).

Table 2 - Parameter and data vectors used in the modified adaptive load prediction algorithms.

Algorithm	
0	$\theta_0(t) = [a_1(t), \ldots, a_n(t), c_1(t), \ldots, c_n(t), \gamma]^T$
	$\varphi_0(t) = [-(y(t-1) - \gamma w(t-1)), \ldots, -(y(t-n) - \gamma w(t-n)), e_0(t-1), \ldots, e_0(t-n), w(t)]^T$
2	$\theta_2(t) = [c_1(t), \ldots, c_n(t), g_0(t), \ldots, g_{n-1}(t), \gamma]^T$
	$\varphi_2(t+k) = [-(\hat{y}(t+k-1\|t-1) - \gamma w(t+k-1)), \ldots, -(\hat{y}(t+k-n\|t-n) - \gamma w(t+k-n)), (y(t) - \gamma w(t)), \ldots, (y(t-n+1) - \gamma w(t-n+1)), w(t+k)]^T$
3	$\theta_3(t) = [h_1(t), \ldots, h_{n+k-1}(t), g_0(t), \ldots, g_{n-1}(t), \gamma]^T$
	$\varphi(t+k) = [-(\hat{y}(t+k-1\|t-1) - \gamma w(t+k-1)), \ldots, -(\hat{y}(t-n+1\|t-k-n+1) - \gamma w(t-n+1)), \varepsilon(t), \ldots, \varepsilon(t-n+1), w(t+k)]^T$

When the parameters are unknown an adaptive prediction algorithm can be based on any of the equations (3.10), (3.11), or (3.12). If the Least Squares method is used for the parameter estimation, the algorithms described above applies also in this case. The data and parameter vectors $\varphi_i(t)$ and $\theta_i(t)$ which are used in the modified algorithms are given in Table 2. Note that the number of parameters to estimate in algorithm 0 or 2 does not depend on the number of steps to predict.

3.1.2 Relations between the algorithms

The equivalence for one-step prediction between the adaptive prediction algorithms that was stated in Chapter 4 of Part I of the report will now be extended to the given algorithms.

Theorem 1. Consider one-step adaptive prediction of the process (3.3) using any of the algorithms given above. There exist constant matrices S_{ij} such that if

$$\theta_i(t_0) = S_{ij}\,\theta_j(t_0)$$

$$P_i(t_0) = S_{ij}\,P_j(t_0)\,S_{ij}^T \qquad i = 0,2,3;\quad j = 0,2,3$$

for some t_0 and the initial values of the $\{y(t)\}$ and $\{\varepsilon(t)\}$ processes are the same for the considered algorithms, then

$$\theta_i(t) = S_{ij}\,\theta_j(t) \qquad t \geq t_0 \qquad i = 0,2,3;\quad j = 0,2,3$$

when the same realization of $\{y(t),\ t \geq t_0\}$ is used in the algorithms.

Proof. The proof of this theorem is quite similar to the proof of Theorem 1 in Part I of the report, why it is omitted. The matrices are

$$S_{02} = \begin{pmatrix} I_n & -I_n & O_{n,n+1} \\ I_n & O_{n,n} & O_{n,n+1} \\ O_{n+1,n} & O_{n+1,n} & I_{n+1} \end{pmatrix};$$

$$S_{03} = \begin{pmatrix} I_n & O_{n,n} & O_{n,n+1} \\ I_n & I_n & O_{n,n+1} \\ O_{n+1,n} & O_{n+1,n} & I_{n+1} \end{pmatrix};$$

$$S_{32} = \begin{pmatrix} I_n & -I_n & O_{n,n+1} \\ O_{n,n} & I_n & O_{n,n+1} \\ O_{n+1,n} & O_{n+1,n} & I_{n+1} \end{pmatrix},$$

where I_n is an $n \times n$ identity matrix and $O_{n,m}$ is an $n \times m$ zero matrix.

□

Corollary. Under the same conditions, the result from Theorem 1 holds also for the modified adaptive prediction algorithms applied to the process (3.10).

Proof. The proof is identical to the proof of Theorem 1. The S_{ij} matrices given there are applicable also to the modified algorithms if the subscript $n+1$ in the rightmost column and in the bottom row is replaced by 1.

□

3.1.3 Multistep predictors

All of the algorithms proposed in Section 3.1.1 can be transformed into multistep prediction algorithms in the same manner as was done for the ARMA process prediction algorithms. Only one of these possible predictors has been tested, the multistep version of the modified algorithm 2.

When applied to one-step prediction of $y(t+k)$ in (3.10) with known parameters the modified algorithm 2 is

$$\begin{aligned}\hat{y}_M(t+k|t+k-1) = {} & \gamma w(t+k) - c_1(\hat{y}_M(t+k-1|t+k-2) - \gamma w(t+k-1)) - \ldots \\ & - c_n(\hat{y}_M(t+k-n|t+k-n-1) - \gamma w(t+k-n)) + \\ & + g_0(y(t+k-1) - \gamma w(t+k-1)) + \ldots + \\ & + g_{n-1}(y(t+k-n) - \gamma w(t+k-n)).\end{aligned}$$

$\hat{y}_M(t|s)$ is the mean of $y(t)$ conditioned on $F(y(s), y(s-1), \ldots)$, i.e. conditioned on the smallest σ-field containing the elements of $\{y(t), t \leq s\}$, cf. e.g. Box and Jenkins (1970). Hence the multistep prediction of $y(t+k)$ is

$$\begin{aligned}\hat{y}_M(t+k|t) = {} & \gamma w(t+k) - c_1(\hat{y}_M(t+k-1|t) - \gamma w(t+k-1)) - \ldots \\ & - c_{k-1}(\hat{y}_M(t+1|t) - \gamma w(t+1)) - c_k(\hat{y}_M(t|t-1) - \\ & - \gamma w(t)) - \ldots - c_n(\hat{y}_M(t+k-n|t+k-n-1) - \gamma w(t+k-n)) + \\ & + g_0(\hat{y}_M(t+k-1|t) - \gamma w(t+k-1)) + \ldots + g_{k-2} \\ & (\hat{y}_M(t+1|t) - \gamma w(t+1)) + g_{k-1}(y(t) - \gamma w(t)) + \ldots \\ & \ldots + g_{n-1}(y(t+k-n) - \gamma w(t+k-n)) \qquad (3.13a)\end{aligned}$$

for $k \leq n$. When $k > n$ the algorithm is

$$\begin{aligned}\hat{y}_M(t+k|t) = {} & \gamma w(t+k) - c_1(\hat{y}_M(t+k-1|t) - \gamma w(t+k-1)) - \ldots - \\ & - c_n(\hat{y}_M(t+k-n|t) - \gamma w(t+k-n)) + g_0(\hat{y}_M(t+k-1|t) - \\ & - \gamma w(t+k-1)) + \ldots + g_{n-1}(\hat{y}_M(t+k-n|t) - \\ & - \gamma w(t+k-n)) \qquad (3.13b)\end{aligned}$$

In the unknown parameter case the algorithm is based on (3.13). The two steps in the predictor are given below, cf. algorithm B in the ARMA process case.

- Estimate the $G(q^{-1})$ and $C(q^{-1})$ polynomials in (3.11) applied to one-step prediction.

- Use (3.13) with the estimated parameters to determine the desired predictions recursively in k.

3.2 TIME VARYING PARAMETERS

The tracking of time variable parameters in real time applications can be accomplished in the algorithms after just minor modifications, cf. Söderström, Ljung and Gustavsson (1974). Two different modifications are usually seen. In both of these the gain in the parameter estimation algorithm does not tend to zero.

By introducing a discounting factor λ in the loss function for the parameter calculation the influence of old errors on the estimates is reduced. This approach to real time estimation is described in e.g. Wieslander (1969) and it is used in the self-tuning algorithms, cf. Åström et al (1977). The algorithm (3.8) is modified to

$$\begin{cases} \theta(t) = \theta(t-1) + K(t)\left(y(t) - \varphi^T(t)\theta(t-1)\right) \\ K(t) = \dfrac{P(t-1)\,\varphi(t)}{\lambda + \varphi^T(t)P(t-1)\varphi(t)} \\ P(t) = \left[P(t-1) - \dfrac{P(t-1)\varphi(t)\varphi^T(t)P(t-1)}{\lambda + \varphi^T(t)P(t-1)\varphi(t)}\right] \Big/ \lambda \end{cases} \tag{3.14}$$

where the subscripts are omitted.

Another way, used e.g. in Bohlin (1976), is based on the Kalman filter interpretation of the Least Squares algorithm. Introduction of state noise, i.e. parameter noise, R leads to the modified P equation

$$P(t) = P(t-1) - \frac{P(t-1)\varphi(t)\varphi^T(t)P(t-1)}{1+\varphi(t)^T P(t-1)\varphi(t)} + R \qquad (3.15)$$

R is positive semidefinite.

3.3 A SIMPLE PREDICTION METHOD

A simple adaptive load predictor may be based on the updated weekly profile (2.1). The prediction is simply this signal multiplied by a scaling factor to account for the actual load.

The basic model is

$$y(t) = \gamma\, w(t)$$

where $w(t)$ is taken from the exponentially updated weekly profile $\{v(t)\}$, cf. Chapter 2. The coefficient γ is calculated in the algorithm (3.14), i.e. the algorithm is given by the following two steps:

- Estimate the scaling factor by

$$\begin{cases} \gamma(t) = \gamma(t-1) + P(t)w(t)\big(y(t) - \gamma(t-1)w(t)\big) \\ P(t) = \dfrac{P(t-1)}{\lambda + w^2(t)P(t-1)} \end{cases} \qquad (3.16)$$

- Use the estimate $\gamma(t)$ to calculate the predictions

$$\hat{y}(t+k|t) = \gamma(t)w(t+k).$$

λ is supposed to have a small value which means that old values of the errors are discounted at a fast rate in the scaling factor calculation. If λ is negligible in the denominator expression in (3.16), the scaling factor is

$$\gamma(t) = \frac{y(t)}{w(t)}$$

The prediction method using this value of $\gamma(t)$ is given by Farmer and Potton (1968). A similar scaling algorithm is described by Tyrén (1974). That algorithm can be considered as a stochastic approximation variant of (3.16), cf. e.g. Ljung and Wittenmark (1974).

4. PREDICTION RESULTS

In this chapter the results from a series of prediction experiments will be given. A threedigit number xx/y will be used for time notation. xx is the week number and y is the day in the week (1 = monday, 2 = tuesday etc.).

Three different data sets from 1973 have been used, 8000 hours - almost the whole year, 1000 hours in the beginning of the year and 1500 hours in the autumn. In all the data sets the first load measurement is taken at 7 o'clock a.m.

The winter data set as well as the data set covering the whole year starts in 02/2. The winter set contains roughly six weeks of load measurements, i.e. the weeks 02,...,07. The starting point for the autumn data set is 33/1. This includes roughly 9 weeks, i.e. the weeks 33,...,41. These two sets of data have slightly different characteristics. The average load is lower in the autumn data set than in the winter set, 3.59 GW compared to 4.26 GW. The influence of random disturbances on the load is more pronounced in the autumn load. Moreover, winter data has a more stable load pattern. These differences taken together render the prediction of autumn data more difficult than prediction of winter data.

The results from the predictions over the whole year are given in Section 4.1. The rest of the results are presented in Section 4.2.

4.1 PREDICTION OVER THE WHOLE YEAR

The predictions over the whole year were performed with the modified algorithm 2, cf Table 2. Just one-step predictions

are considered. The results of a sensitivity study of the constants in the algorithm are given in Section 4.2. Here the number of parameters in the estimates of the polynomials $C(q^{-1})$ and $G(q^{-1})$ were three and one respectively, cf (3.11). γ in (3.11) was fixed to 1. The discounting factor λ in (3.14) was 0.995.

In the predictions no attempt was made to treat the holidays in a special manner. This will have at least two negative implications on the result.

1/ The increase in the lossfunctions V_{MW} (2.2) and $V_{\%}$ (2.3) will be significant during the holiday since the nominal load is irrelevant.
2/ The large prediction errors will imply large changes in the parameters. This in turn might deteriorate the performance of the algorithm immediately after the holiday.

Hence, results from these simulations can be regarded as an upper limit for the obtainable results with this algorithm applied on these data.

In Figures 3 and 4 the lossfunctions V_{MW} and $V_{\%}$ as functions of time are shown. The corresponding standard deviations are given in Table 3.

Table 3 - Prediction results for one-step predictions over 8000 hours starting in 02/2 with the modified algorithm 2.

Nominal Load	s_{MW}(MW)	s_{rel}(%)	$s_{\%}$(%)
Fixed profile	69.5	1.89	1.98
Previous week α=1 in (2.1)	58.8	1.60	1.75
Previous week α=0.75 in (2.1)	55.8	1.52	1.66

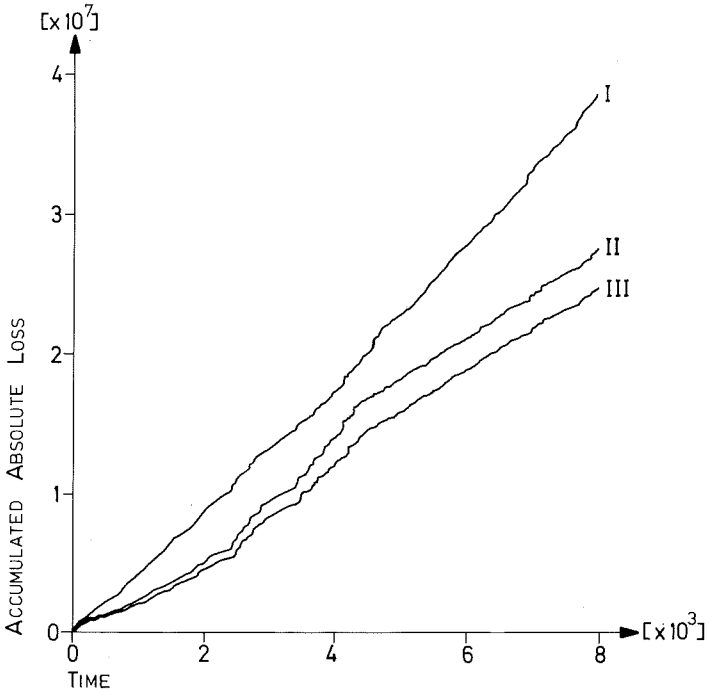


Figure 3 - The lossfunction V_{MW} (2.2) obtained from one-step prediction over 8000 hours starting in 02/2 with the modified algorithm 2. The nominal load is represented by I/ a fixed profile II/ the previous week, α=1 in (2.1) and III/ the previous week, α=0.75 in (2.1).

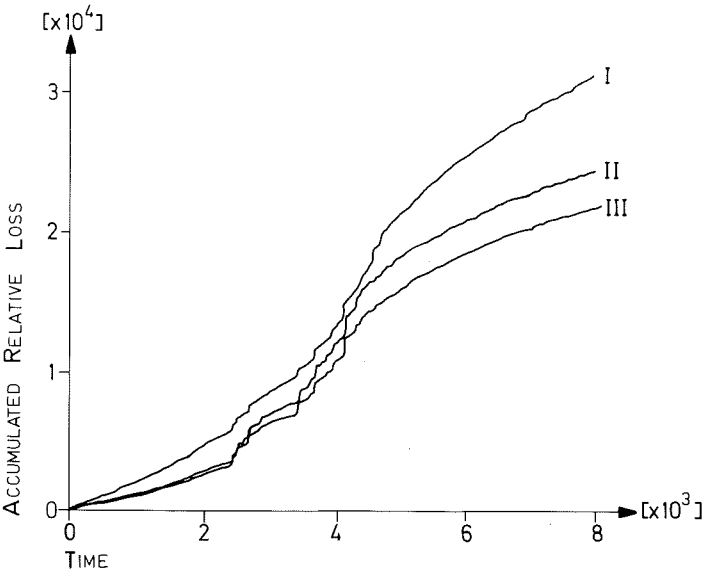


Figure 4 - The lossfunction $V_{\%}$ (2.3) obtained from one-step prediction over 8000 hours starting in 02/2 with the modified algorithm 2. The nominal load is represented by I/ a fixed profile II/ the previous week, α=1 in (2.1) and III/ the previous week, α=0.75 in (2.1).

Evidently, the inclusion of seasonal characteristics in the nominal load representation is preferable to the fixed profile. Furthermore, updating of the nominal load according to (2.1) with $\alpha<1$ have a positive influence on the prediction result.

It is also obvious that an implementation for online use need a special facility to treat the holidays. The late spring and summer part of the curves in Figure 3 shows that the prediction error then has a larger value than in the other parts of the year. This is emphasized in Figure 4 where the large increase in $V_{\%}$ during the middle of the year in addition depends on the relatively low value of the load. Hence s_{MW} or s_{rel} are more relevant measures of the power which has to be put on line or removed in order to compensate for the erroneous prediction.

4.2 PREDICTION OVER PARTS OF THE YEAR

In this section the performance of the various algorithms as well as the influence of the parameters in the algorithms that are at user's disposal will be investigated.

Nominal load representation

An example of prediction of autumn load is shown in Figures 5 and 6. The modified algorithm 2 is applied to one-step prediction of the data in week 41. In Figure 5 the nominal load is represented by the fixed profile and in Figure 6 it is represented by the load in the previous week updated as in (2.1) with $\alpha=0.75$. In both of the experiments the

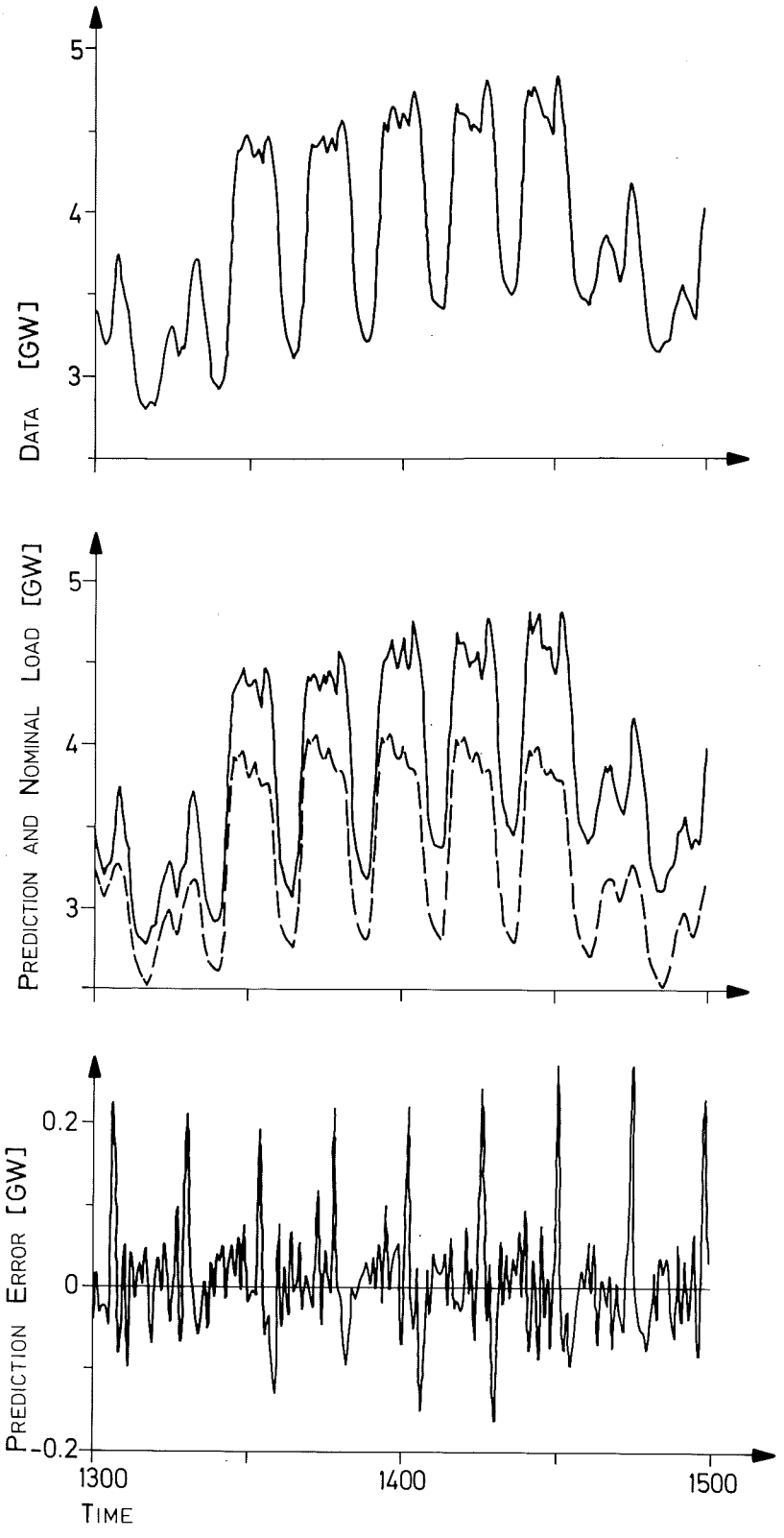


Figure 5 - One-step prediction of data in week 41. Prediction started in 33/1. The nominal load is represented by the fixed profile.

In the middle figure the solid line shows the prediction.

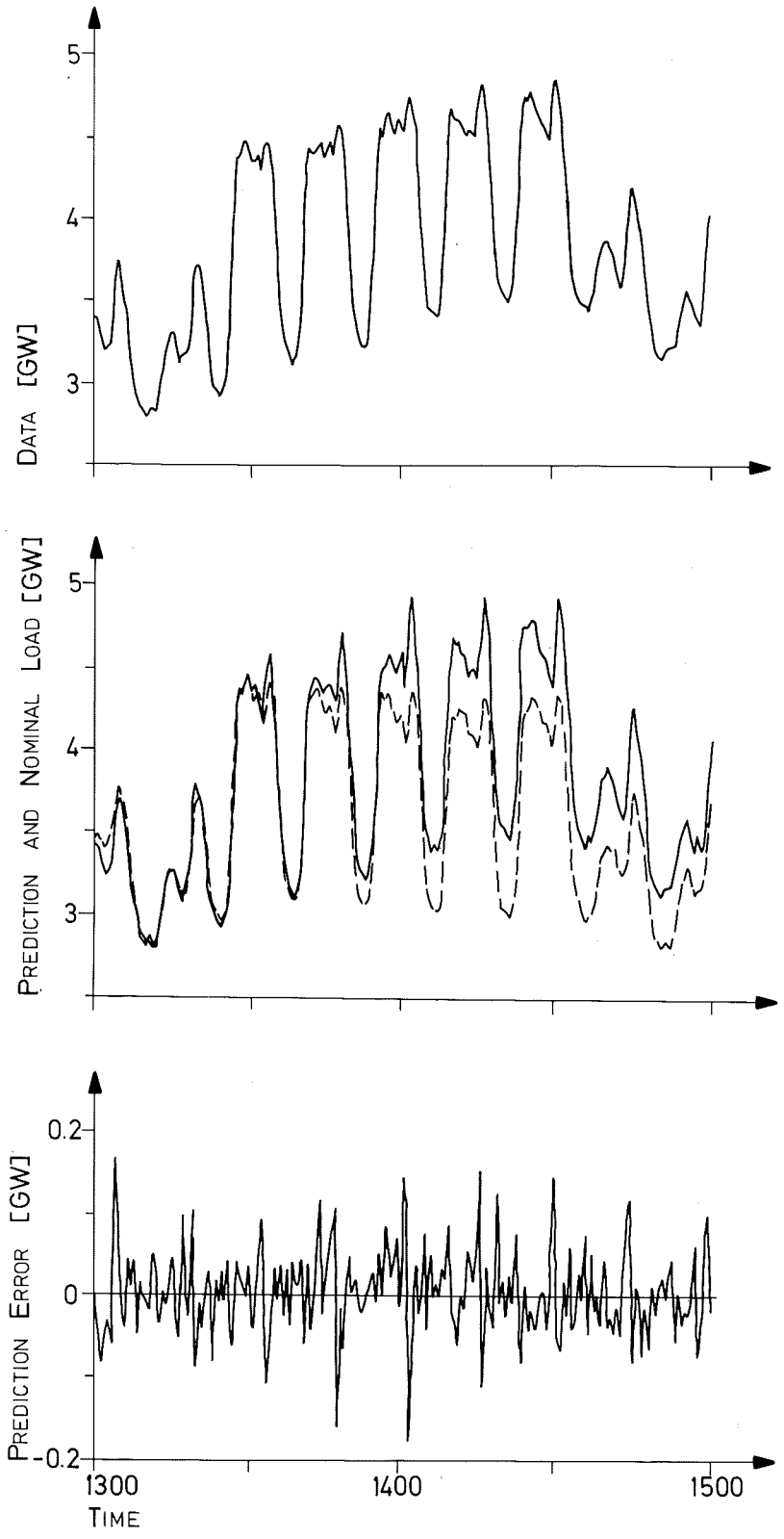


Figure 6 - One-step prediction of data in week 41. Prediction started in 33/1. The nominal load is represented by filtered data from the previous week, $\alpha=0.75$ in (2.1).

In the middle figure the solid line shows the prediction.

number of parameters in the $C(q^{-1})$ and $G(q^{-1})$ polynomial estimates were three and one respectively, cf. (3.11). γ in (3.11) was fixed to 1. The discounting factor λ in (3.14) was 0.995.

Evidently, the prediction algorithm can adapt its performance to the variation in the mean level in data. This implies that changes from day to day, as e.g. most weather induced load variations can be handled.

The differencies between the two nominal load representations are illustrated. As in the prediction over the whole year the inclusion of seasonal characteristics in the profile is beneficial to the prediction results. In this case, the regular high prediction errors when the fixed profile is used depend on the missing afternoon peak in that nominal load representation.

In the predictions with the fixed profile $s_{\%}$=1.743 % and s_{rel}=1.813 %. When the filtered load from the previous week is used in the nominal load representation the corresponding results are 1.354 % and 1.410 % respectively.

In the following, only the filtered previous week load is used in the nominal load representation.

Nominal load model

Depending on the complexity of the nominal load model, two types of algorithm were given in Chapter 3. When the nominal load is modeled as in (3.2) the number of parameter estimates used in the adaptive k-step prediction algorithms 2 and 3 depends linearly on k, cf. Table 1. It is in accordance with the model to estimate only the first n coefficients in the polynomial $H(q^{-1})$,

cf. algorithm 4 in Part I of this report. However, the number of parameters needed in the $K(q^{-1})$ polynomial estimate is not a priori reducible.

In the modified algorithms, based on the load model (3.9) no $K(q^{-1})$ polynomial has to be estimated. Hence, the number of parameters to estimate does not depend on k, except possibly in the modified algorithm 3.

The advantages of the modified algorithm are also illustrated in the following application of algorithm 3 to one-step prediction of winter data with $n = 2$ and $\lambda = 0.995$. The parameter and data vectors are given in Table 1. In Figures 7A and B the h_1 and k_2 and the h_2 and k_3 estimates respectively are shown. These trajectories show that there is a large covariation in the two parameter estimates. Furthermore, the condition number of the matrix P in algorithm (3.8), measured as the quotient between the largest and the smallest eigenvalue, is roughly 10^6. This is an indication of overparametrization, and it also induces numerical difficulties in the propagation of the Riccati equation (3.8). In the modified algorithms no such covariation is present and the condition number of the matrix P is typically 10^2.

Because of these essential advantages of the modified algorithms, they are considered significantly superiour to the algorithms based on the more general load model (3.3). Hence, in the following only results from applications of the modified algorithms are shown. The simulations performed with the model (3.3) give similar or inferior prediction results.

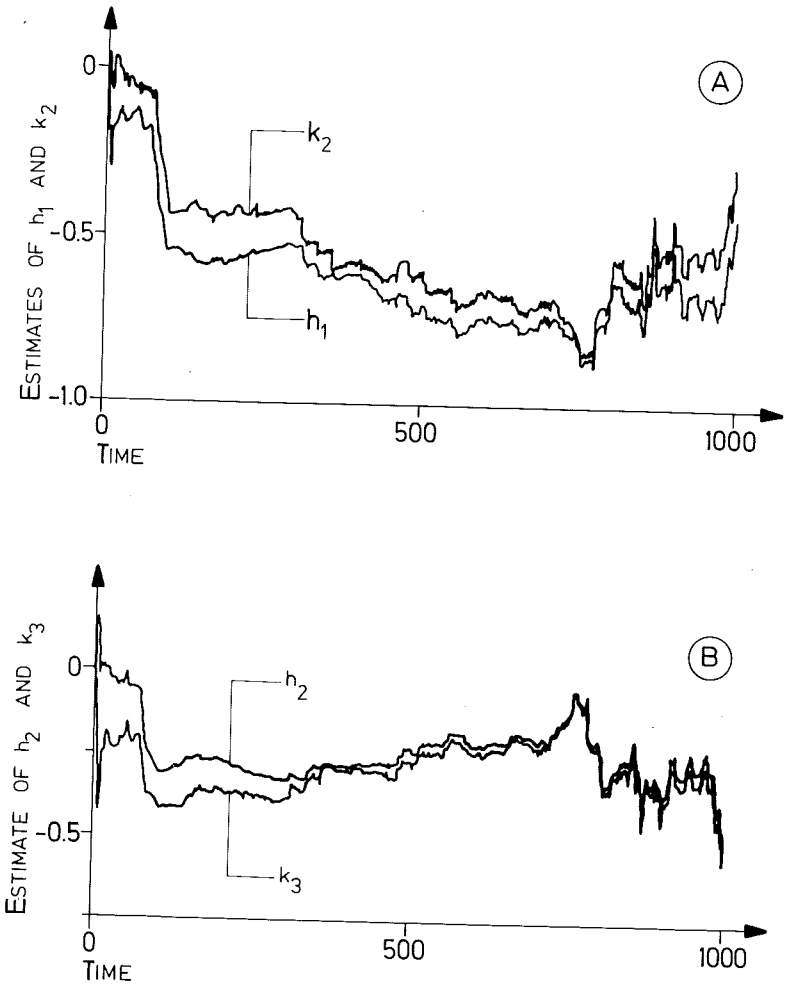


Figure 7 - One-step prediction of winter data with algorithm 3.

A: h_1 and k_2 parameter estimates

B: h_2 and k_3 parameter estimates

Constants in the algorithm

When applying the adaptive predictor some constants in the estimation algorithm have to be postulated. In order to find reasonable values of these constants a series of one-step predictions using the modified algorithm 2 has been performed. The constants are: the number of parameters, the discounting factor λ in (3.14) and the parameter noise

R in (3.15). Furthermore, the filtering constant α in the profile updating equation (2.1) must be determined.

These simulations showed that γ in (3.9) preferably should be fixed to 1. The timevariation of the parameters was found to be easily handled by the discounting factor λ in (3.14). When three estimates in the polynomial C and one estimate in the polynomial G were used, the results varied within 1% as long as $\lambda < 0.99$. In the simulations α in (2.1) was 1.0, 0.75, and 0.67. The best results were obtained for $0.995 \leq \lambda \leq 0.999$. The inclusion of the parameter noise R in (3.15) did not improve the prediction result.

In Table 4 the results from predictions where the number of parameters was varied are presented. Autumn data are used in the simulations. The last 200 data are plotted in Figures 5 and 6.

Table 4 - Influence of the number of parameters n_c and n_g in the estimated C and G polynomials on the prediction result. λ=0.999 in (3.14) and α=1.0 in (2.1). Autumn data. γ in (3.9) was fixed to 1.

		Last 1350 data		Last 200 data	
n_c	n_g	$s_{\%}$	s_{rel}	$s_{\%}$	s_{rel}
0	1	1.545	1.552		
1	1	1.538	1.548		
1	0	5.55	5.79		
0	2	1.541	1.552		
1	2	1.538	1.548		
2	2	1.534	1.543		
2	1	1.534	1.542		
2	0	5.55	5.79		
3	3	1.533	1.542	1.453	1.508
3	2	1.533	1.542	1.456	1.510
3	1	1.532	1.540	1.456	1.510
4	1	1.530	1.539	1.458	1.512

Clearly, the one-step prediction result is not very sensitive to the number of parameters in the $C(q^{-1})$ and $G(q^{-1})$ polynomial estimates as long as $n_g > 0$.

Finally, the influence of the filtering constant α in the profile updating equation (2.1) on the prediction result has been studied. Winter as well as autumn data have been used in the prediction experiments. The results are given i Table 5.

Table 5 - Influence of the filter constant α in the profile updating equation (2.1) on the one-step prediction result. Three C-parameters and one G-parameter were estimated. $\lambda = 0.995$ in (3.14). γ in (3.9) was fixed to 1.

	Winter Last 850 data	Autumn Last 1350 data
α	$s_{\%}$	$s_{\%}$
1.00	1.014	1.535
0.90	0.9801	1.508
0.80	0.9577	1.496
0.75	0.9509	1.496
0.70	0.9473	1.500
0.65	0.9470	1.508
0.60	0.9503	1.520
0.50	0.9694	1.559
0.40	1.009	1.621

These results show that the inclusion of filtering operations on the profile improves the prediction. They are rather robust relative to variations in the filtering constant. k-step prediction results are however somewhat more sensitive to changes in α, especially for large values of k. It is an important parameter in the k-step prediction algorithm.

Version of the algorithm

A discussion of the different versions of the adaptive prediction algorithm must be based on results from k-step predictions, since they have identical one-step prediction

performance, cf. Chapter 3.

Version 0 of the algorithm, cf. Table 2, has not as good a performance as version 2 when applied to k-step prediction with large values of k. This is illustrated in Table 6 for k=24.

Table 6 - The modified algorithms 0 and 2 applied to 24-step prediction. λ=0.999 in (3.14) and α=0.60 in (2.1)

	Winter Last 850 data		Autumn Last 1350 data		Last 200 data	
	$s_{\%}$	s_{rel}	$s_{\%}$	s_{rel}	$s_{\%}$	s_{rel}
Version 0	1.930	1.907	3.440	3.526	4.161	4.393
Version 2	1.864	1.837	2.919	2.894	3.140	3.192

This discrepancy between the prediction results is caused by the approximations done when the model (3.9) is applied to data. In the 24-step predictor in algorithm 2 the C- and G-parameters are tuned to fit the 24-step prediction problem. In algorithm 0 the parameters in the one-step predictor are estimated. The estimates are then used in the calculation of the predictor as if the model was an exact description of the process which not is the case.

The prediction results from algorithm 3 are similar to the results from algorithm 2. However, in algorithm 3 the $H(q^{-1})$ polynomial, with n+k-1 parameters have to be estimated. The model structure can be exploited as in Part I of the report to yield an algorithm with fewer parameters. This algorithm is identical to algorithm 2, which is shown in the same manner as in Chapter 4 in Part I.

k-step prediction

Finally, the k-step and the multistep versions of the modified algorithm 2 together with the scaling method with $\lambda = 0.01$ in (3.16) have been used in a series of k-step prediction experiments. The results from the prediction of winter data are shown in Figure 8. s_{MW}, s_{rel}, and $s_{\%}$ are evaluated over the last 850 hours. In Figure 9 the corresponding results from the autumn data set are shown. The goodness measures are evaluated during the last 1350 and the last 200 hours. In all the simulations $\lambda = 0.999$ in (3.14) and $\alpha = 0.75$ in (2.1). The number of parameters in the $C(q^{-1})$ and $G(q^{-1})$ polynomial estimates was three and one respectively and γ in (3.9) was fixed to 1.

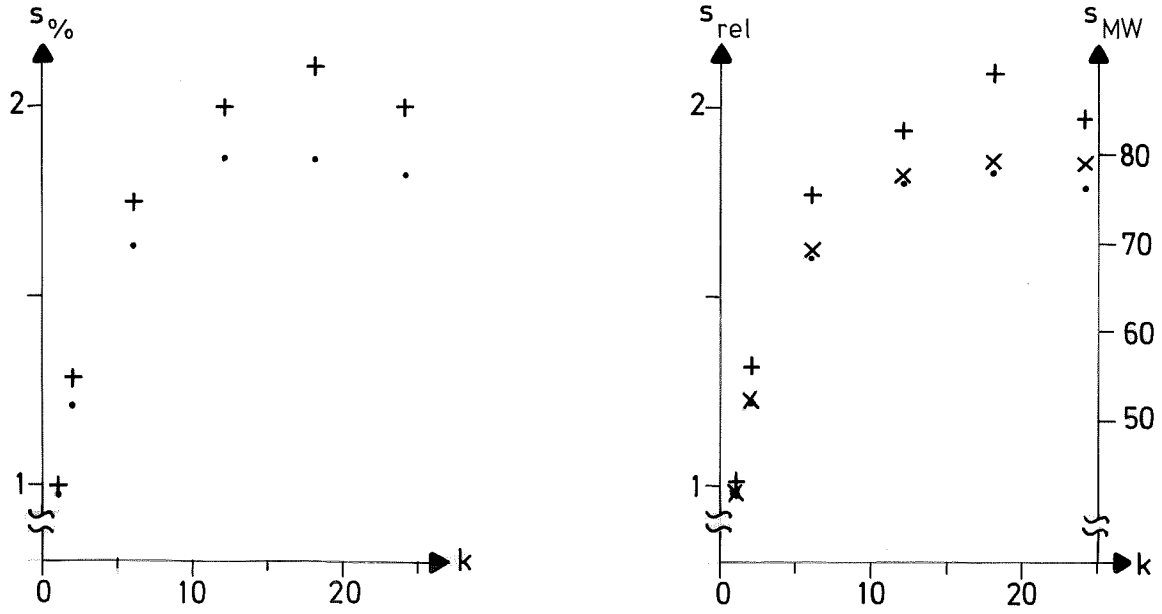


Figure 8 - Results of k-step prediction of winter data. The results are evaluated over the last 850 hours in the data set.

- • Adaptive k-step predictor
- x Adaptive multistep predictor
- \+ Scaling method

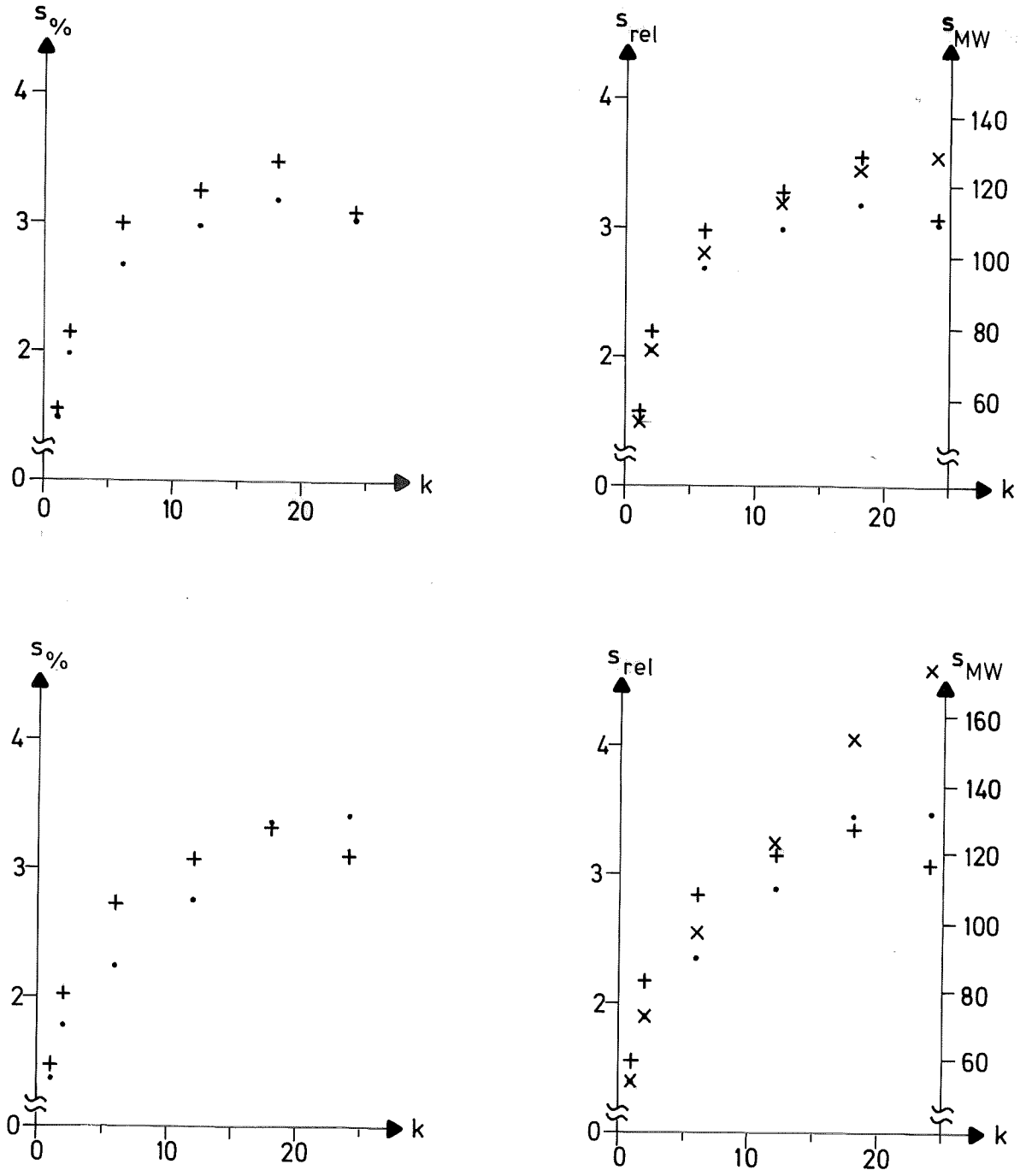


Figure 9 - Results of k-step prediction of autumn data. The results are evaluated over the last 1350 hours in the upper figures and over the last 200 hours in the lower figures.

- • Adaptive k-step predictor
- x Adaptive multistep predictor
- \+ Scaling method

The adaptive k-step predictor has for all values of k a better performance than the multistep predictor. The difference is small for small values of k but gets significant when k increases. As in version 0 of the algorithm the multistep predictor is based on the one-step predictor parameters. The discussion above of the discrepancies between version 0 and 2 is thus applicable also here.

The essential property of the scaling method is that changes in the process in relation to the nominal load are reflected in the parameter just one step after their occurance. It gives in almost all cases not as good prediction results as the adaptive k-step algorithm. This is mainly caused by the better modeling of the process in the latter of these methods. However, in week 41 the scaling method has a slightly better performance for 18- and 24-step prediction. The large variations in the load, cf. Figures 5 and 6, are quickly reflected in the value of the scaling parameter. In the adaptive k-step predictor however, the parameter adjustment is based approximatively on the prediction error. Especially in the 24-step prediction case this lag is too large to allow for compensation of the fast and large changes in the load.

5. Summary

The short-term prediction of power load is a wellknown and early recognized problem. The presented approach is a new variant of attacking it. The given adaptive k-step predictor is not believed to be the final solution. However, based on the given criteria this algorithm gives similar or in most cases superior prediction results compared to other methods presented.

A variety of algorithms has been discussed. They are all but one based on a separation of the load into a nominal and a residual part. The residual load is modeled as an ARMA process and the nominal load representation has been based either on a fixed profile or on the obtained data filtered by a first order filter.

The solution to the representation and modeling problem for the nominal load is crucial for the final prediction result. Different approaches to this problem have been considered. It has been demonstrated that the nominal load should contain seasonal characteristics of the load and also that the current load should be filtered when updating the weekly profile. Moreover, with a nominal load representation with these properties, the best results are obtained with a simple load model where the deviations between the current load and the nominal load is modeled as an ARMA process.

The prediction methods developed in the first part of the report have been extended to fit the load prediction problem. Four different adaptive k-step prediction algorithms and a simple variant, the scaling method, have been considered. Two of these, version 0 of the basic algorithm and the multistep predictor are based on the one-step predictor parameters. They give not as good

prediction results for large values of k as the other two adaptive k-step alternatives. The reason for this is the inherent approximations done in the basic modeling. The two remaining adaptive prediction methods, versions 2 and 3 of the algorithm give similar prediction results. Of these, version 2 is preferred since the number of parameters therein does not depend on k. The scaling method gives in general significantly worse prediction results than this algorithm.

The multistep predictor is apt for efficient profile prediction, i.e. prediction of the hourly load during e.g. a day. The obtained k-step prediction results show, however, that it is more accurate to use one adaptive k-step predictor for each value of k. The amount of computer time and computer storage needed is of course larger.

In the algorithms, some constants have to be chosen. The values used of these constants have proved to give good results. They are, however, not claimed to be overall optimal, why tuning to a specific prediction situation might improve the result.

A future development of the algorithm should be directed towards an improved nominal load handling. Such an improvement should manage the influence of fast and large variations in the load on the predictions for large values of k. It may e.g. be based on weather information, which in such cases can be beneficial for the prediction result.

In an online implementation the current algorithm has to be extended with a facility to treat the holiday loads. It should also include a possibility to do manual adjustments of the predictions to compensate for large, occasional but aforeseen and known changes in the load such as e.g. changes caused by very popular TV programs.

6. REFERENCES

Åström, K J (1970): Introduction to Stochastic Control Theory. Academic Press, New York.

Åström, K J, U Borisson, L Ljung and B Wittenmark (1977): Theory and Application of Self-Tuning Regulators. To be published in the September issue of Automatica 13. This is an expanded version of a paper given at the 6th IFAC World Congress in Boston, Mass. 1975.

Bohlin, T (1976): Four Cases of Identification of Changing Systems. In: R Mehra and D G Lainiotis (Editors): System Identification: Advances and Case Studies. Academic Press, New York.

Bohlin, T and P Kamjou (1977): A Batch Program for Forecasting Electric Consumption. TRITA-REG-7701, Department of Automatic Control, The Royal Institute of Technology, Stockholm, Sweden.

Box, G E P and G M Jenkins (1970): Time Series Analysis: Forecasting and Control. Holden Day, San Fransisco.

Brown, R D (1963): Smoothing, Forecasting and Prediction of Discrete Time Series. Prentice Hall, Englewood Cliffs,N.J.

Christiaanse, W R (1971): Short-Term Load Forecasting Using General Exponential Smoothing. IEEE Tr-PAS 90, 900-910.

Davies, M (1959): The Relationship between Weather and Electricity Demand. Proc. IEE 106 C, 27-37.

Dryar, H A (1944): The Effect of Weather on the System Load. Tr-AIEE 63, 1006-1013.

Farmer, E D (1963): A Method of Prediction of Nonstationary Processes and its Application to the Problem of Load Estimation. Proc. 2nd IFAC Congress, Basel, Switzerland.

Farmer, E D and M J Potton (1966): The Prediction of Load on a Power System. Proc. 3rd IFAC Congress, London.

Farmer, E D and M J Potton (1968): Development of Online Load-Prediction Techniques with Results from Trials in the South-West Region of the CEGB. Proc. IEE 115, 1549-1558.

Galiana, F D (1971): An Application of System Identification and State Prediction to Electric Load Modeling and Forecasting. PhD Thesis, MIT.

Gupta, P C and K Yamada (1972): Adaptive Short-Term Forecasting of Hourly Load Using Weather Information. IEEE Tr-PAS 91, 2085-2095.

Holst, J (1974): On the Use of Self-Tuning Predictors for the Prediction of Power Load.(In Swedish). TFRT-3119, Dept of Automatic Control, Lund Institute of Technology, Lund, Sweden.

Holst, J (1977): Adaptive Short-Term Prediction of Power Load. Load Data. In preparation.

Keyhani, A and A H El-Abiad (1975): One-Step-Ahead Load Forecasting for On-line Applications. Paper C75027-8, 1975 Winter Power Meeting, New York.

Lijesen, D P and J Rosing (1971): Adaptive Forecasting of Hourly Loads Based on Load Measurement and Weather Information. IEEE Tr-PAS 90, 1757-1767.

Ljung, L and B Wittenmark (1974): Asymptotic Properties of Self-Tuning Regulators, TFRT-3071, Dept of Automatic Control, Lund Institute of Technology, Lund, Sweden.

Matthewman, P D and H Nicholson (1968): Techniques for Load Prediction in the Electricity-Supply Industry. Proc. IEE 115, 1451-1457.

Sharma, K L S and A K Mahalanabis (1974): Recursive Short-Term Load-Forecasting Algorithm. Proc. IEE 121, 59-62.

Söderström, T, L Ljung and I Gustavsson (1974): A Comparative Study of Recursive Identification Methods. TFRT-3085, Dept of Automatic Control, Lund Institute of Technology, Lund, Sweden.

Toyoda, J, M Chen and Y Inoue (1970): An Application of State Estimation to Short-Term Load Forecasting. Part I - Forecasting Modeling, Part II - Implementation. IEEE Tr-PAS 89, 1678-1688.

Tyrén, L (1974): Some Experiments with Short-Term Prediction of the Load on the Power Network of the Swedish State Power Board. (In Swedish). Report, Swedish State Power Board.

Wieslander, J (1969): Real Time Identification - Part I. TFRT-3013, Dept of Automatic Control, Lund Institute of Technology, Lund, Sweden.

Part III - Local Convergence of Some Recursive Estimation Algorithms

ABSTRACT

The local convergence properties of recursive estimation algorithms are considered. The key result is the calculation of explicit expressions for the eigenvalues in the linearization of a differential equation describing the algorithm. The general result is applied to the Extended Least Squares method and a modification thereof, an algorithm by Landau and the self-tuning regulator. The analysis gives useful insight into the properties of recursive estimation algorithms. It leads to the construction of a new algorithm for estimation of the parameters in an ARMA process.

1. Background

Some applications of control theory require determination of process parameters using system identification. It is often desirable or even necessary to calculate the estimates on line. Recursive parameter estimation algorithms are therefore of both practical and theoretical interest.

Recursive Least Squares is the generic recursive method. It gives, however, biased estimates if the equation error is not white noise. Many algorithms have been proposed to overcome this difficulty, see e.g. the survey by Åström and Eykhoff (1971) or the book by Eykhoff (1974). Different algorithms have been compared on various test cases, cf. Söderström, Ljung and Gustavsson (1974), Saridis (1974) or Isermann et al (1974). For all of these methods the consistency question is crucial.

One approach to the convergence analysis is based on Lyapunov and hyperstability methods. These are often used in connection with model reference adaptive schemes, cf. Landau (1974). They have been adopted on a recursive stochastic parameter estimation algorithm by Landau (1976), where he shows asymptotic unbiasedness of the parameter estimates. In Ledwich and Moore (1976) stability results from deterministic considerations based on Lyapunov theory are used together with martingale theory to show convergence w.p. 1 of the parameter estimates for a class of stochastic parameter estimation algorithms.

Ljung has in a series of papers, see e.g. Ljung (1975, 1976a) developed a different method for convergence analysis of recursive stochastic algorithms. In Ljung (1976b, c) the positive realness concept, which is intimitely linked to hyperstability theory via the Kalman-Yakubovich lemma, is connected to his analysis.

The analysis in the present report is based on the theory in Ljung (1975, 1976a). It aims at establishing conditions for local convergence of some parameter estimation algorithms to appropriately chosen convergence points. The estimation algorithms are applied to linear, timeinvariant, single input-

single output stochastic systems described by difference equations. The main idea in this theory is to associate the parameter estimation algorithm with a differential equation that contains all relevant information about the asymptotic behaviour of the algorithm. Here a brief review of the results needed in the current analysis will be given.

Let the algorithm be given by

$$\begin{cases} x(t) = x(t-1) + \gamma(t)Q(t;\ x(t-1),\ \psi(t)) \\ \psi(t) = A(x(t-1))\psi(t-1) + B(x(t-1))e(t) \end{cases} \tag{1.1}$$

where $\{x(t)\}$ and $\{\psi(t)\}$ are sequences of estimates and observations respectively. $A(\cdot)$ and $B(\cdot)$ are two matrix functions. $Q(\cdot;\ \cdot,\ \cdot)$ is a deterministic function which together with the gain sequence $\{\gamma(t)\}$ determine the algorithm.

Introduce the set D_s as

$$D_s = \{x | A(x) \text{ is stable}\}. \tag{1.2}$$

Now, take a fixed $x \in D_s$ and define the random process $\psi(t, x)$

$$\begin{cases} \psi(t, x) = A(x)\psi(t, x) + B(x)e(t) \\ \psi(0, x) = 0 \end{cases} \tag{1.3}$$

$\psi(t, x)$ is welldefined and since $x \in D_s$ it approaches stationarity exponentially for stationary sequences $\{e(t)\}$.

Define the function

$$f(x) = \lim_{t\to\infty} E\ Q(t;\ x,\ \psi(t, x)) \tag{1.4}$$

where the expectation is over the distribution of $\{e(t)\}$. Introduce the differential equation

$$\frac{d}{d\tau} x_D(\tau) = f(x_D(\tau)) \tag{1.5}$$

where the subscript D is used to distinguish between the solution to (1.5) and the estimates from (1.1).

The basis for the present analysis is the theorems 4 and 5 in Ljung (1975). There is stated that only stable stationary points to the differential equation (1.5) are possible convergence points for the algorithm (1.1). Thus if x^* is a convergence point then

$$f(x^*) = 0 \tag{1.6}$$

and

$$F(x^*) = \frac{d}{dx} f(x)\Big|_{x=x^*} \tag{1.7}$$

has all eigenvalues in the left half plane $\{\text{Re } x \leq 0\}$. Thus the eigenvalues of $F(x^*)$ are closely related to the convergence properties of the algorithm (1.1).

In the recursive parameter estimation algorithms studied here the vector of estimates can be partitioned as

$$x = (\theta^T, r^T)^T \tag{1.8}$$

where θ contains the unknown parameters. The estimation algorithm can be written as

$$\begin{cases} \theta(t) = \theta(t-1) + \gamma(t) \cdot R^{-1}(t) \cdot Q_1(\theta(t-1), \varphi(t)) \\ R(t) = R(t-1) + \gamma(t)[Q_2(\theta(t-1), \varphi(t)) - R(t-1)] \\ \psi(t) = \begin{bmatrix} \varphi(t) \\ \rho(t) \end{bmatrix} = A(\theta(t-1))\psi(t-1) + B(\theta(t-1))e(t) \end{cases} \tag{1.9}$$

i.e. the vector r in (1.8) is composed of elements from the R matrix. The differential equation corresponding to (1.9) is

$$\begin{cases} \frac{d}{d\tau} \theta_D(\tau) = R_D^{-1}(\tau) \cdot f_1(\theta_D(\tau)) \\ \frac{d}{d\tau} R_D(\tau) = f_2(\theta_D(\tau)) - R_D(\tau) \end{cases} \tag{1.10}$$

where

$$\begin{cases} f_1(\theta_D(\tau)) = \lim\limits_{t\to\infty} E\, Q_1(\theta_D(\tau), \varphi(t, \theta_D(\tau))) \\ f_2(\theta_D(\tau)) = \lim\limits_{t\to\infty} E\, Q_2(\theta_D(\tau), \varphi(t, \theta_D(\tau))) \end{cases} \tag{1.11}$$

The linearization of (1.10) around $\theta_D = \theta^*$ and $R_D = f_2(\theta^*)$ is straightforward

$$\begin{cases} \frac{d}{d\tau}(\theta_D(\tau)-\theta^*) = K(\theta^*)(\theta_D(\tau)-\theta^*) \\ \frac{d}{d\tau}(R_D(\tau)-f_2(\theta^*)) = Z(\theta^*)(\theta_D(\tau)-\theta^*) - (R_D(\tau)-f_2(\theta^*)) \\ K(\theta^*) = f_2^{-1}(\theta^*)\, \frac{d}{d\theta} f_1(\theta)\big|_{\theta=\theta^*} \\ Z(\theta^*) = \frac{d}{d\theta} f_2(\theta)\big|_{\theta=\theta^*} \end{cases} \tag{1.12}$$

The $F(x^*)$ matrix is thus blocktriangular

$$F(x^*) = \begin{pmatrix} K(\theta^*) & 0 \\ Z(\theta^*) & -I \end{pmatrix}$$

and the stability properties of the linearization are determined by $K(\theta^*)$.

The eigenvalues of two different matrices K will be calculated. The results of these calculations can be applied to many algorithms. Here the Extended Least Squares (ELS) method, the algorithm by Landau mentioned above and the self-tuning regulator by Åström and Wittenmark are treated. The eigenvalues for a modified version of the ELS method will also be calculated. The application of the theory to these algorithms is discussed in Ljung (1975, 1976a, b, c). The self-tuning regulator case is also treated in Ljung and Wittenmark (1974).

It is assumed throughout the discussion that the orders of all the involved polynomials are known and that the correct number of parameters are estimated.

The rest of this part of the report is organized as follows. In Chapter 2 the algorithms mentioned above are treated and the $K(\theta^*)$ matrix is determined. It is shown that for all of the algorithms except the modified ELS method it can be written as

$$K(\theta^*) = -[E\ \varphi(t, \theta^*)\varphi^T(t, \theta^*)]^{-1} E\varphi(t, \theta^*)\tilde{\varphi}^T(t, \theta^*)$$

where $\varphi(t, \theta^*)$, cf. (1.9), is composed of data and noise elements and $\tilde{\varphi}(t, \theta^*)$ is a filtered version of $\varphi(t, \theta^*)$. In the modified ELS method $K(\theta^*)$ is given by

$$K(\theta^*) = -[E\ \tilde{\varphi}(t, \theta^*)\varphi^T(t, \theta^*)]^{-1} E\tilde{\varphi}(t, \theta^*)\tilde{\varphi}^T(t, \theta^*)$$

The eigenvalues to these matrices are calculated in Chapter 3 and the result is used for a treatment of the local convergence properties of the algorithms. In Chapter 4 the eigenvalue result is used as a tool for evaluation of algorithms. A new method for ARMA process identification is also given. Some examples in Chapter 5 and a summary in Chapter 6 concludes this part of the report.

2. Some Specific Algorithms

The algorithms mentioned above are presented in this chapter. The corresponding differential equations are given together with the linearizations. The algorithms are devoted one section each, 2.1 to ELS and a modification thereof, 2.2 to Landau's algorithm and 2.3 to the self-tuning regulator. In Section 2.4 the similarities between the algorithms are discussed.

2.1. THE BASIC AND A MODIFIED ELS ALGORITHM

When parameters in an autoregressive model for a timeseries with correlated noise are to be estimated, a straightforward application of the Least Squares method results in nonconsistent parameter estimates. If the noise is modeled as a moving average process, the estimation problem can be approximately solved on a least squares basis. The innovations are then estimated as a sequence of residuals. This approach, here called ELS-Extended Least Squares was originally suggested by Panuska (1968, 1969) and Young (1968). Different versions of the algorithm are discussed in e.g. Young (1974), Talmon and van den Boom (1973) and Kashyap (1974).

The convergence properties of ELS are treated in Ljung, Söderström and Gustavsson (1975). It is shown that the algorithm does not converge for all systems. This is also discussed in Goedheer (1976). Sufficient conditions for global convergence are given in Ljung (1976b).

The method will now be briefly described. Refer to for example Young (1974) or Ljung, Söderström and Gustavsson (1975) for details.

An ARMA process is described by the equation

$$A_0(q^{-1})y(t) = C_0(q^{-1})e(t) \tag{2.1}$$

where $\{y(t)\}$ is the output process and $\{e(t)\}$ a stationary sequence of independent random variables such that all moments exist. The mean value of $\{e(t)\}$ is supposed to be zero and the variance σ^2. $A_0(q^{-1})$ and $C_0(q^{-1})$ are two relatively

prime polynomials in the backward shift operator q^{-1} of orders n_a and n_c

$$A_0(q^{-1}) = 1 + a_1^0 q^{-1} + \ldots + a_{n_a}^0 q^{-n_a} \tag{2.2a}$$

$$C_0(q^{-1}) = 1 + c_1^0 q^{-1} + \ldots + c_{n_c}^0 q^{-n_c} \tag{2.2b}$$

It is assumed that the polynomials are asymptotically stable. Introduce the reciprocal polynomials $A_0^f(q)$ and $C_0^f(q)$ defined by

$$A_0^f(q) = q^{n_a} A_0(q^{-1}) = q^{n_a} + \ldots + a_{n_a}^0 \tag{2.3}$$

and similarly for $C_0^f(q)$.

Using the vectors

$$\varphi_0(t) = [-y(t-1), \ldots, -y(t-n_a), e(t-1), \ldots, e(t-n_c)]^T \tag{2.4a}$$

$$\theta_0 = [a_1^0, \ldots, a_{n_a}^0, c_1^0, \ldots, c_{n_c}^0]^T \tag{2.4b}$$

equation (2.1) may be written

$$y(t) = \varphi_0^T(t)\ \theta_0 + e(t) \tag{2.5}$$

If the parameter vector θ_0 is unknown the unmeasurable noise sequence is estimated with a sequence of residuals which is treated as an input signal in an ordinary least squares approach. Hence, the data and parameter vectors are

$$\varphi(t) = [-y(t-1), \ldots, -y(t-n_a), \varepsilon(t-1), \ldots, \varepsilon(t-n_c)]^T \tag{2.6a}$$

$$\theta(t) = [a_1(t), \ldots, a_{n_a}(t), c_1(t), \ldots, c_{n_c}(t)]^T \tag{2.6b}$$

where $a_j(t)$ is the estimate of a_j^0 obtained at time t. $\{\varepsilon(t)\}$ is the residual sequence

$$\varepsilon(t) = y(t) - \varphi^T(t)\theta(t-1) \tag{2.7}$$

From here on, the basic and the modified versions of ELS differ. Therefore, each of them is given a subsection of the chapter.

2.1.1. The Basic ELS Algorithm

In the basic ELS algorithm the parameter estimates are recursively updated according to

$$\begin{cases} \theta(t) = \theta(t-1) + \frac{1}{t} \cdot \frac{R^{-1}(t-1)\varphi(t)\varepsilon(t)}{1+\frac{1}{t}[\varphi^T(t)R^{-1}(t-1)\varphi(t)-1]} \\ \varepsilon(t) = y(t) - \varphi^T(t)\theta(t-1) \\ R(t) = R(t-1) + \frac{1}{t}[\varphi(t)\varphi^T(t) - R(t-1)] \end{cases} \tag{2.8}$$

The differential equation describing the asymptotic behaviour of this algorithm is

$$\frac{d}{d\tau}\theta_D(\tau) = R_D^{-1}(\tau) f_1(\theta_D(\tau)) \tag{2.9a}$$

$$\frac{d}{d\tau}R_D(\tau) = f_2(\theta_D(\tau)) - R_D(\tau) \tag{2.9b}$$

where

$$f_1(\theta) = E\ \varphi(t,\theta)\varepsilon(t,\theta) \tag{2.10a}$$

$$f_2(\theta) = E\ \varphi(t,\theta)\varphi^T(t,\theta) \tag{2.10b}$$

The expectation is taken with respect to the distribution of $\{e(t)\}$ when $\{y(t)\}$ and $\{\varepsilon(t,\theta)\}$ are stationary processes obtained from (2.1) and (2.7) for fixed θ. The stability region is

$$D_s = \{\theta \mid \theta = (a_1, \ldots, a_{n_a}, c_1, \ldots, c_{n_c})^T;\ C^f(z) \text{ stable}\}$$

It is easily seen that $f_1(\theta)$ in (2.10a) can be rewritten as

$$f_1(\theta) = -\tilde{f}_1(\theta)(\theta-\theta_0) \tag{2.11}$$

where

$$\tilde{f}_1(\theta) = E\ \varphi(t,\theta)\tilde{\varphi}(t,\theta)^T \tag{2.12}$$

$$\tilde{\varphi}(t,\theta) = \frac{1}{C_0(q^{-1})}\varphi(t,\theta) \tag{2.13}$$

cf. Ljung (1976b) where these expressions are calculated. Clearly $\theta = \theta_0$ is a stationary point. It is shown in Ljung, Söderström and Gustavsson (1975) that in this case, when the orders of the estimated polynomials are equal to the true orders, θ_0 is the only stationary point. Thus the K matrix is, cf. Chapter 1,

$$K(\theta_0) = -f_2^{-1}(\theta_0)\tilde{f}_1(\theta_0) = \\ = -[E\varphi(t,\theta_0)\varphi^T(t,\theta_0)]^{-1} E\varphi(t,\theta_0)\tilde{\varphi}^T(t,\theta_0) \tag{2.14}$$

2.1.2. Young's Modification of the ELS Algorithm

In Young (1976) the basic ELS algorithm is modified to include also filtered values of φ.

Introduce $\eta(t)$, a filtered version of the data vector $\varphi(t)$

$$\eta(t) = \frac{1}{C(q^{-1},t)}\varphi(t) \tag{2.15}$$

where $C(q^{-1}, t)$ is the estimate at time t of the C_0-polynomial.

The algorithm is

$$\begin{cases} \theta(t) = \theta(t-1) + \frac{1}{t} \cdot \frac{1}{1+\frac{1}{t}(\varphi^T(t)R^{-1}(t-1)\eta(t)-1)} \cdot \\ \qquad \cdot R^{-1}(t-1)\eta(t)(y(t) - \varphi^T(t)\theta(t-1)) \\ R(t) = R(t-1) + \frac{1}{t}(\eta(t)\varphi(t)^T - R(t-1)) \end{cases} \tag{2.16}$$

The functions $f_1(\theta)$ and $f_2(\theta)$ entering in the differential equations are

$$f_1(\theta) = E\ \eta(t, \theta)\varepsilon(t, \theta) \tag{2.17a}$$

$$f_2(\theta) = E\ \eta(t, \theta)\varphi^T(t, \theta) \tag{2.17b}$$

The expectation is taken with respect to $\{e(t)\}$ where $\{y(t)\}$ and $\{\varepsilon(t, \theta)\}$ are stationary processes obtained from (2.1) and (2.7). $\eta(t, \theta)$ is the filtered version of $\varphi(t, \theta)$ obtained from (2.15). The stability region coincides with the

stability region for the basic ELS algorithm.

Rewrite $f_1(\theta)$ from (2.17a) in the same manner as in Section 2.1.1 to yield

$$f_1(\theta) = -\tilde{f}_1(\theta)(\theta-\theta_0) = -E\eta(t, \theta)\tilde{\varphi}(t, \theta)(\theta-\theta_0)$$

where $\tilde{\varphi}(t, \theta)$ is given by (2.13). The true value θ_0 of the unknown parameter vector thus indeed is a stationary point and linearization around θ_0 gives

$$K(\theta_0) = -f_2^{-1}(\theta_0)\tilde{f}_1(\theta_0) =$$

$$= -[E\tilde{\varphi}(t, \theta_0)\varphi^T(t, \theta_0)]^{-1}E\tilde{\varphi}(t, \theta_0)\tilde{\varphi}^T(t, \theta_0) \qquad (2.18)$$

since

$$\eta(t, \theta_0) = \tilde{\varphi}(t, \theta_0) = \frac{1}{C_0(q^{-1})}\varphi(t, \theta_0)$$

2.2. THE ALGORITHM BY LANDAU

This algorithm is presented in Landau (1976). There it is shown that the estimates are asymptotically unbiased if a certain transfer function is positive real, cf. Chapter 4. The algorithm is also discussed in Ljung (1976b). In this paper global convergence w.p.1 to the true values of the parameters is shown under the same conditions. The following is a brief review of the algorithm. Refer to Landau's article for details.

The process is assumed to be given by the equation

$$A_0(q^{-1})y(t) = B_0(q^{-1})u(t) + w(t) \qquad (2.19)$$

where $\{w(t)\}$ is an unmeasurable disturbance, $\{u(t)\}$ and $\{y(t)\}$ the input and output sequences respectively. $\{w(t)\}$ is supposed to be a stationary stochastic process with rational spectral density such that all moments exist and $\{u(t)\}$ a stationary sequence of independent random variables such that all moments exist. The mean value of $\{u(t)\}$ is supposed to

be zero and the variance σ^2. The $\{u(t)\}$ and $\{w(t)\}$ sequences are supposed to be independent. The polynomials $A_0(q^{-1})$ and $B_0(q^{-1})$ are of order n_a and n_b respectively, with A_0 as in (2.2a) and

$$B_0(q^{-1}) = b_0^0 + b_1^0 q^{-1} + \ldots + b_{n_b}^0 q^{-n_b}$$

With

$$\theta_0 = [a_1^0, \ldots, a_{n_a}^0, b_0^0, \ldots, b_{n_b}^0]^T$$

$$\varphi_0(t) = [-y(t-1), \ldots, -y(t-n_a), u(t), \ldots, u(t-n_b)]^T$$

the process equation (2.19) can be written

$$y(t) = \varphi_0^T(t)\theta_0 + w(t) \tag{2.20}$$

The unknown parameters in the A_0 and B_0 polynomials are estimated from a model

$$A(q^{-1})y_M(t) = B(q^{-1})u(t)$$

or if the vectors

$$\theta(t) = (a_1(t), \ldots, a_{n_a}(t), b_0(t), \ldots, b_{n_b}(t))^T$$

$$\varphi(t) = (-y_M(t-1), \ldots, -y_M(t-n_a), u(t), \ldots, u(t-n_b))^T$$

are used

$$y_M(t) = \varphi^T(t)\theta(t)$$

Introduce the sequences

$$\varepsilon_0(t) = y(t) - y_M(t) \tag{2.21}$$

$$\varepsilon(t) = y(t) - \varphi^T(t)\theta(t-1) + \sum_{i=1}^{n_a} c_i^0 \varepsilon_0(t-i) \tag{2.22}$$

where c_i^0, $i = 1, \ldots, n_a$ are some suitably chosen numbers. The parameter estimates $\theta(t)$ are recursively calculated according to (2.8) with $\varepsilon(t)$ given by (2.22). This gives the differential equation (2.9) with

$$f_1(\theta) = -E\varphi(t, \theta)\tilde{\varphi}^T(t, \theta)\ (\theta-\theta_0) \tag{2.23a}$$

$$f_2(\theta) = E\varphi(t, \theta)\varphi^T(t, \theta) \tag{2.23b}$$

The expectations are taken over the distributions of $\{u(t)\}$ and $\{w(t)\}$. The vector $\tilde{\varphi}(t,\theta)$ is defined as

$$\tilde{\varphi}(t, \theta) = \frac{C_0(q^{-1})}{A_0(q^{-1})}\ \varphi(t, \theta)$$

$$C_0(q^{-1}) = 1 + c_1^0 q^{-1} + \ldots + c_{n_a}^0 q^{-n_a}$$

The stability region is given by

$$D_s = \{\theta \mid \theta = (a_1, \ldots, a_{n_a}, b_0, \ldots, b_{n_b})^T;\ A^f(z) \text{ stable}\}$$

Since the orders of the $A(q^{-1})$ and $B(q^{-1})$ polynomials are the same as the orders of the true polynomials $A_0(q^{-1})$ and $B_0(q^{-1})$ the obvious stationary point $\theta = \theta_0$ is unique (Ljung 1976b). Thus the K-matrix is

$$K(\theta_0) = -f_2^{-1}(\theta_0)f_1(\theta_0) =$$

$$= -[E\varphi(t, \theta_0)\varphi^T(t, \theta_0)]^{-1}E\varphi(t, \theta_0)\tilde{\varphi}^T(t, \theta_0) \tag{2.24}$$

2.3. THE SELF-TUNING REGULATOR

The self-tuning regulator is presented in Åström and Wittenmark (1973) and Wittenmark (1973). The properties of the algorithm are also discussed in Ljung and Wittenmark (1974), Åström, Borisson, Ljung and Wittenmark (1977) and Ljung and Wittenmark (1976). The convergence question is treated in Ljung (1976b), where sufficient conditions for global convergence are given.

Assume that the system is described by the difference equation

$$y(t+1) + a_1^0 y(t) + \ldots + a_{n_a}^0 y(t-n_a+1) =$$

$$= b_0^0 u(t-k) + \ldots + b_{n_b}^0 u(t-k-n_b) + v(t+1) \tag{2.25a}$$

or if polynomial notation is used

$$A_0(q^{-1})y(t+1) = b_0^0 B_0(q^{-1})u(t-k) + v(t+1) \tag{2.25b}$$

$\{u(t)\}$ and $\{y(t)\}$ are the input and output sequences of the system. The time delay $k \geq 0$. $\{v(t)\}$ is a sequence of random variables. The system is supposed to be minimum phase. Rewrite (2.25) as

$$\begin{aligned} &y(t+k+1) + \alpha_0^0 y(t) + \ldots + \alpha_{n_a-1}^0 y(t-n_a+1) = \\ &\qquad = \beta_0^0[u(t) + \beta_1^0 u(t-1) + \ldots + \beta_{n_b+k}^0 u(t-n_b-k)] + w(t+k+1) \\ &y(t+k+1) + \alpha(q^{-1})y(t) = \beta_0^0 \beta(q^{-1})u(t) + w(t+k+1) \end{aligned} \tag{2.26}$$

where $\{w(t)\}$ is a moving average of order k of the $\{v(t)\}$ process.

After introduction of θ_0 and $\varphi_0(t)$ from

$$\theta_0 = (\alpha_0^0, \ldots, \alpha_{n_a-1}^0, 0, \ldots, 0, \beta_1^0, \ldots, \beta_{n_b+k}^0)^T$$

$$\varphi_0(t) = (-y(t), \ldots, -y(t-n_a+1), -y(t-n_a), \ldots, -y(t-s),$$
$$, \beta_0^0 u(t-1), \ldots, \beta_0^0 u(t-n_b-k))^T$$

where there are $s+1-n_a$ zeroes in θ_0, $s \geq n_a-1$, (2.26) can be written as

$$y(t+k+1) - \beta_0^0 u(t) = \varphi_0^T(t)\theta_0 + w(t+k+1) \tag{2.27}$$

The choice of the number s is further discussed below.

The equation (2.27) is the starting point for the estimation. Assume that $\beta_0^0 = b_0^0$ and k are known. The model is then

$$y(t+k+1) - \beta_0^0 u(t) = \varphi^T(t)\theta + w_1(t+k+1) \tag{2.28}$$

where the vectors $\varphi(t)$ and θ are

$$\theta = [\alpha_0, \ldots, \alpha_s, \beta_1, \ldots, \beta_{n_b+k}]^T$$

$$\varphi(t) = [-y(t), \ldots, -y(t-s), \beta_0^0 u(t-1), \ldots, \beta_0^0 u(t-n_b-k)]^T$$

The actual estimate of θ_0, $\theta(t)$, is given by a least squares estimate from (2.8) in the same manner as in the basic ELS and in the Landau algorithms. The residual sequence $\{\varepsilon(t)\}$ entering in the algorithm is calculated as

$$\varepsilon(t) = y(t) - \beta_0^0 u(t-k-1) - \varphi^T(t-k-1)\theta(t-1) \tag{2.29}$$

The input $\{u(t)\}$ is computed as timevarying linear feedback from the output

$$u(t) = -\frac{1}{\beta_0^0}\varphi(t)^T\theta(t) \tag{2.30}$$

This choice of input is discussed and motivated in Åström and Wittenmark (1973). The main reason for it is that if $\{v(t)\}$ is a white noise sequence then this regulator with the correct number of parameters in the estimation converges to

$$u(t) = -\frac{1}{\beta_0^0}\varphi_0^T(t)\theta_0$$

which is a minimum variance regulator for the process (2.25) in this case. However, one of the main properties of the self-tuning regulator is that minimum variance control asymptotically is achieved with the regulator (2.30) also when the noise $\{v(t)\}$ is a moving average of a white noise sequence. The convergence point for the parameter estimation is in this case,

$$\theta_{MV} = (-g_0,\ldots,-g_s, h_1, \ldots, h_{n_b+k}) \tag{2.31}$$

if the parameter estimates converge and if there are enough parameters in the estimation. The parameters g_i and h_j are calculated from the polynomial equalities, cf. Åström (1970)

$$\begin{cases} C_0(q^{-1}) = A_0(q^{-1})F(q^{-1}) + q^{-k-1}G(q^{-1}) \\ H(q^{-1}) = B_0(q^{-1})F(q^{-1}) \end{cases} \tag{2.32}$$

where $A_0(q^{-1})$ and $B_0(q^{-1})$ are defined in (2.25) above and $C_0(q^{-1})$ is the moving average description of $\{v(t)\}$, i.e.

$$v(t) = e(t) + c_1^0 e(t-1) + \ldots + c_{n_c}^0 e(t-n_c) = C_0(q^{-1})e(t) \tag{2.33}$$

$\{e(t)\}$ is a stationary sequence of independent random variables with zero mean and all moments existing. The F and G polynomials are defined by

$$F(q^{-1}) = 1 + f_1 q^{-1} + \ldots + f_k q^{-k} \tag{2.34a}$$

$$G(q^{-1}) = g_0 + g_1 q^{-1} + \ldots + g_s q^{-s}; s = \max(n_a-1,\ n_c-k-1,\ 0) \tag{2.34b}$$

Clearly, if the number of α-parameters is smaller than $s+1$, θ_{MV} can never be a convergence point for the estimation algorithm. Note that when $C_0 \equiv 1$, i.e. when $\{v(t)\}$ is white noise, $\theta_0 = \theta_{MV}$. Also note that when a minimum variance regulator is used the output is a moving average of order k over the sequence $\{e(t)\}$, i.e.

$$y(t) = e(t) + \ldots + f_k e(t-k) = F(q^{-1})e(t) \tag{2.35}$$

where the $F(q^{-1})$ polynomial is the same as in equation (2.32).

Contrary to the former algorithms the parameter estimates in this algorithm also influences the value of the process output since the input $\{u(t)\}$ is calculated via feedback (2.30). The differential equations will however be the same as for ELS, i.e. (2.9) with

$$f_1(\theta) = E\varphi(t-k,\ \theta)\varepsilon(t+1,\ \theta) \tag{2.36a}$$

$$f_2(\theta) = E\varphi(t-k,\ \theta)\varphi^T(t-k,\ \theta) \tag{2.36b}$$

The expectation is to be taken over the distribution of $\{e(t)\}$ assuming stationarity of the involved processes (2.25), (2.30) and (2.29).

The stability region is

$$D_s = \{\theta \mid \theta = \{\alpha_0, \ldots, \alpha_s, \beta_1, \ldots, \beta_{n_b+k}\};$$

$$z^{s+1-n_a} A_0^f(z)\beta^f(z) - B_0^f(z)\alpha^f(z) = 0$$

$$\Rightarrow |z| < 1\}$$

The linearization of the differential equation starts with a rewriting of $\varepsilon(t+1, \theta)$ in (2.36a).

$$\varepsilon(t+1, \theta) = y(t+1, \theta) = \varphi(t-k, \theta)^T\theta_0 + \beta_0^0 u(t-k) + C_0\bar{F}e(t+1)$$

where (2.29), (2.33) and (2.26) have been used. The $\bar{F}$ polynomial is of order k and emanates from the rewriting of (2.25) to (2.26) with the identity

$$1 = A_0\bar{F} + q^{-k-1}\bar{G} \tag{2.37}$$

Using (2.30) for $u(t-k)$ and introducing θ_{MV} leads to

$$\varepsilon(t+1, \theta) = \varphi(t-k, \theta)^T(\theta_0-\theta_{MV}) + \varphi(t-k, \theta)^T(\theta_{MV}-\theta) + C_0\bar{F}e(t+1)$$

Now, (2.37), (2.32) and (2.25) gives

$$\varphi(t-k, \theta)^T(\theta_0-\theta_{MV}) = (1-C_0)y(t+1,\theta)-(\bar{F}-F)C_0e(t+1)$$

which leads to

$$\varepsilon(t+1, \theta) = (1-C_0)\varepsilon(t+1, \theta) + \varphi(t-k, \theta)^T(\theta_{MV}-\theta) + C_0Fe(t+1)$$

$$\varepsilon(t+1, \theta) = \frac{1}{C_0(q^{-1})}\varphi(t-k, \theta)^T(\theta_{MV}-\theta) + Fe(t+1)$$

Denote the filtered value of $\varphi(t-k, \theta)$ by $\tilde{\varphi}(t-k, \theta)$. The equation (2.36a) is then

$$f_1(\theta) = -E\varphi(t-k, \theta)\tilde{\varphi}(t-k, \theta)^T(\theta-\theta_{MV}) =$$

$$= -\tilde{f}_1(\theta)(\theta-\theta_{MV})$$

It is shown in Åström and Wittenmark (1973) that θ_{MV} in fact is the only stationary point when the correct number of parameters are estimated. The matrix K thus is

$$K(\theta_{MV}) = -f_2^{-1}(\theta_{MV})\tilde{f}_1(\theta_{MV}) \tag{2.38}$$

2.4. SUMMARY OF THE ALGORITHMS

The linearized equations (1.12) show that the matrix K is crucial for the stability of the differential equation and hence for the convergence of the algorithm. In the discussed four algorithms this matrix has been written as

$$K(\theta) = -f_2^{-1}(\theta)\tilde{f}_1(\theta)$$

where the functions $\tilde{f}_1(\theta)$ and $f_2(\theta)$ are calculated above. These functions are principally equal for the ELS, the Landau algorithm and the self-tuning regulator. The corresponding K matrices can all be written

$$K(\theta) = -[E\varphi(t, \theta)\varphi^T(t, \theta)]^{-1}E\varphi(t, \theta)\tilde{\varphi}^T(t, \theta) \tag{2.39}$$

where $\varphi(t, \theta)$ is composed of old data and noise elements and $\tilde{\varphi}(t, \theta)$ is a filtered version of $\varphi(t, \theta)$.

These three algorithms can be regarded as special cases of an algorithm where the K matrix has the form (2.39). The filtered vector $\tilde{\varphi}(t, \theta)$ is calculated as

$$\tilde{\varphi}(t, \theta) = H(q^{-1})\varphi(t, \theta) \tag{2.40}$$

where $H(q^{-1})$ is causal, rational and asymptotically stable filter. The eigenvalues for this more general type of algorithm are calculated in the next chapter. The algorithm is said to be <u>symmetric</u> since $f_2(\theta)$ is a symmetric matrix.

The modified ELS algorithm do not fit into this form and has to be treated separately.

3. Eigenvalue Calculation and Local Convergence Results

In this chapter the eigenvalues of the matrices K, defined in (1.12), will be calculated for the recursive schemes presented in the previous chapter.

3.1. THE GENERALIZED SYMMETRIC LEAST SQUARES METHOD

The eigenvalues of the matrix

$$K(\theta) = -[E\varphi(t, \theta)\varphi^T(t, \theta)]^{-1}E\varphi(t, \theta)\tilde{\varphi}^T(t, \theta) \tag{3.1}$$

with

$$\tilde{\varphi}(t, \theta) = H(q^{-1})\varphi(t, \theta)$$

and $H(q^{-1})$ a causal, rational and asymptotically stable filter will be calculated. The result is interpreted as local convergence conditions for the three algorithms from Chapter 2.

3.1.1. Preliminaries

This section contains two lemmas needed for the eigenvalue calculations. The first lemma concerns the influence of a white noise component in the $\varphi(t, \theta)$ vector.

Lemma 1. Given the signal $S(s)$ with n elements which are outcomes from random processes. Given also the signal $W(s) = (w(s-1), \ldots, w(s-m))^T$ where the m elements are consecutive outcomes from a sequence of uncorrelated random variables $\{w(s)\}$ with zero mean value. Form the signals $\tilde{S}(s)$ and $\tilde{W}(s)$ from

$$\tilde{S}(s) = H(q^{-1})S(s); \quad \tilde{W}(s) = H(q^{-1})W(s)$$

where $H(q^{-1})$ is a causal, rational and asymptotically stable filter with $H(0) = 1$. Suppose that $E\, WS^T = 0$ and $E\, W\tilde{S}^T = 0$. Then the set of eigenvalues to the matrix

$$-\left[E\begin{bmatrix}S\\W\end{bmatrix}\begin{bmatrix}S\\W\end{bmatrix}^T\right]^{-1} E\begin{bmatrix}S\\W\end{bmatrix}\begin{bmatrix}\tilde{S}\\\tilde{W}\end{bmatrix}^T = -\left[E\ MM^T\right]^{-1} E M\tilde{M}^T$$

are given by the eigenvalues of

$$-\left[E\ SS^T\right]^{-1}\ E\ S\tilde{S}^T$$

and

$$-\left[E\ WW^T\right]^{-1}\ E\ W\tilde{W}^T$$

Moreover, the matrix has at least m eigenvalues in -1.

Proof. The lemma is proved in Appendix A. □

The m eigenvalues in -1 in the lemma corresponds to white noise components in the data vector. In the following lemma the rest of the eigenvalues are determined for a special form of the vector signal $S(t)$.

Lemma 2. Given a vector signal

$$S(t) = [s_1, \ldots, s_p, s_{p+1}, \ldots, s_n]^T$$

where the first p elements are stationary stochastic ARMA processes generated by

$$s_k(t) = \frac{B_k(q^{-1})}{A(q^{-1})}\, w(t) = \frac{b_0^k+\ldots+b_{n-1}^k q^{-n+1}}{1+a_1q^{-1}+\ldots+a_nq^{-n}}\, w(t);\ k = 1, \ldots, p$$

$\{w(t)\}$ is a sequence of uncorrelated random variables with mean value zero and variance σ^2. The polynomial A is supposed to be asymptotically stable.

The remaining $n-p$ elements in S are lagged values of a stationary ARMA process

$$s_{p+k}(t) = z(t-k+1) = \frac{C(q^{-1})}{A(q^{-1})}\, w(t-k+1) =$$

$$= \frac{c_0+c_1q^{-1}+\ldots+c_pq^{-p}}{1+a_1q^{-1}+\ldots+a_nq^{-n}}\, w(t-k+1);\ k = 1, \ldots, n-p$$

Given also the filtered signals

$$\tilde{S}(t) = H(q^{-1})S(t) = (\tilde{s}_1(t), \ldots, \tilde{s}_n(t))^T$$

where $H(q^{-1})$ is asymptotically stable

$$H(q^{-1}) = \frac{R(q^{-1})}{P(q^{-1})} = \frac{1+r_1q^{-1}+\ldots+r_\ell q^{-\ell}}{1+p_1q^{-1}+\ldots+p_m q^{-m}}$$

Then the eigenvalues of the matrix

$$-(E\ SS^T)^{-1}E\ S\tilde{S}^T$$

are

$$-H(\alpha_k) \quad \text{where} \quad A^f(\alpha_k) = 0,\ k = 1, \ldots, n$$

Proof. The proof is found in Appendix A. □

Corollary. If $S = (s_1, \ldots, s_n)^T$ with

$$s_k(t) = \frac{B_k(q^{-1})}{A(q^{-1})}\, w(t),\ k = 1, \ldots, n$$

and the same assumptions on the involved entities as above in Lemma 2 are supposed to hold, then the eigenvalues of

$$-\left[E\ SS^T\right]^{-1}E\ S\tilde{S}^T$$

are

$$-H(\alpha_k) \quad \text{where} \quad A^f(\alpha_k) = 0;\ k = 1, \ldots, n$$

Proof. The result follows immediately from the first part of the preceeding lemma, see Appendix A. □

These two lemmas and the subsequent corollary will be combined into a theorem concerning the eigenvalues to the matrix K in (3.1). The eigenvalue result is then applied on the three special algorithms.

3.1.2. Local convergence

Theorem 1. Consider a stationary random process

$$z(t) = \frac{D(q^{-1})}{F(q^{-1})G(q^{-1})}\, v(t) \tag{3.2}$$

where

$$D(q^{-1}) = d_0 + d_1 q^{-1} + \ldots + d_{n_d} q^{-n_d}$$

$$F(q^{-1}) = 1 + f_1 q^{-1} + \ldots + f_{n_f} q^{-n_f}$$

$$G(q^{-1}) = 1 + g_1 q^{-1} + \ldots + g_{n_g} q^{-n_g}$$

G is supposed to be asymptotically stable. $\{v(t)\}$ is a moving average

$$v(t) = F(q^{-1})e(t)$$

where $\{e(t)\}$ is a stationary sequence of independent random variables with mean value zero and variance σ^2.

Introduce the vectors $\varphi(t)$ and $\tilde{\varphi}(t)$ as

$$\varphi(t) = [-z(t-1), \ldots, z(t-n_f-n_g), v(t), \ldots, v(t-n_d)]^T$$

$$\tilde{\varphi}(t) = H(q^{-1})\varphi(t)$$

$$H(q^{-1}) = \frac{R(q^{-1})}{P(q^{-1})} = \frac{1+r_1 q^{-1}+\ldots+r_{n_r} q^{-n_r}}{1+p_1 q^{-1}+\ldots+p_{n_p} q^{-n_p}}$$

where $H(q^{-1})$ is asymptotically stable.

Then the eigenvalues of

$$-[E\varphi(t)\varphi^T(t)]^{-1} E\varphi(t)\tilde{\varphi}^T(t) \tag{3.3}$$

are

$$\begin{cases} -1 \text{ of multiplicity } n_f + n_d + 1 \\ -H(\alpha_k) \text{ where } G^f(\alpha_k) = 0;\ k = 1, \ldots, n_g \end{cases}$$

Proof. The proof is given in Appendix B. □

Corollary. Under the assumptions of the theorem, with the exceptions that $D(q^{-1})$ in (3.2) is supposed to have the constant term equal to 1, and that $\varphi(t)$ is supposed not to contain the element $v(t)$, the eigenvalues to

$$-(E\ \varphi\varphi^T)^{-1}\ E\varphi\tilde{\varphi}^T$$

are

$$\begin{cases} -1 \text{ with multiplicity } n_f + n_d \\ -H(\alpha_k) \text{ where } G^f(\alpha_k) = 0;\ k = 1, \ldots, n_g \end{cases}$$

Proof. The result follows immediately from the proof of Theorem 1, cf. Appendix B. □

This theorem and its corollary will now be applied on the ELS method, Landau's algorithm and the self-tuning regulator.

Theorem 2. If the parameter estimates from the ELS algorithm, described above in Section 2.1.1, converge to $\theta = \theta_0$ then the eigenvalues of $K(\theta_0)$

$$-\frac{1}{C_0(\alpha_k)} \text{ where } A_0^f(\alpha_k) = 0,\ k = 1, \ldots, n_a$$

have negative real parts.

Proof. The proof is given in Appendix C. □

Theorem 3. If the parameter estimates from the algorithm by Landau as described above in Section 2.2 converge to $\theta = \theta_0$ then the eigenvalues of $K(\theta_0)$

$$-\frac{C_0(\alpha_k)}{A_0(\alpha_k)} \text{ where } A_0^f(\alpha_k) = 0,\ k = 1, \ldots, n_a$$

have negative real parts.

Proof. The proof is given in Appendix C. □

Finally, the self-tuning regulator is treated.

Theorem 4. If the parameter estimation in the self-tuning regulator as described above in Section 2.3 converges to $\theta = \theta_{MV}$, the eigenvalues of $K(\theta_{MV})$

$$-\frac{1}{C_0(\beta_k)} \quad \text{where} \quad B_0^f(\beta_k) = 0, \; k = 1, \ldots, n_b$$

have negative real parts.

Proof. Also this proof is given in Appendix C. □

3.2. LOCAL CONVERGENCE RESULTS FOR THE MODIFIED ELS ALGORITHM

In this algorithm the matrix K is given by (cf. (2.18))

$$-f_2^{-1}(\theta_0)\tilde{f}_1(\theta_0) = -[E\,\tilde{\varphi}(t, \theta_0)\varphi(t, \theta_0)^T]^{-1} \cdot E\,\tilde{\varphi}(t, \theta_0)\tilde{\varphi}(t, \theta_0)^T \tag{3.4}$$

where θ_0 and $\varphi(t, \theta_0)$ are defined in Section 2.1. The filter involved in the calculation of $\tilde{\varphi}$ is $1/C_0(q^{-1})$, as in the basic ELS scheme.

A necessary condition for local convergence of this algorithm is then given by the following theorem.

Theorem 5. Consider the modified ELS method as described in Section 2.1. If the parameter estimates from this algorithm converge to $\theta = \theta_0$ then the eigenvalues of $K(\theta_0)$

$$-\frac{1}{C_0(\alpha_k)} \quad \text{where} \quad A_0^f(\alpha_k) = 0, \quad k = 1, \ldots, n_a$$

and

$$-\frac{1}{C_0(\gamma_k)} \quad \text{where} \quad C_0^f(\gamma_k) = 0, \quad k = 1, \ldots, n_c$$

have negative real parts.

Proof. The proof is given in Appendix D. □

3.3. DISCUSSION

Consider a parameter estimation algorithm which is connected to an ordinary differential equation

$$\frac{d}{d\tau}\,\theta_D(\tau) = h(\theta_D(\tau)) \tag{3.5}$$

with the linearization around $\theta_D(\tau) = \theta^*$

$$\frac{d}{d\tau}\,(\theta_D(\tau) - \theta^*) = K(\theta^*)(\theta_D(\tau) - \theta^*)$$

cf. Chapter 1. If this matrix $K(\theta^*)$ can be represented as (3.3), its eigenvalues given by Theorem 1 give a necessary and sufficient condition for local stability of the differential equation. In turn this gives a necessary condition for the algorithm to converge to θ^*. Thus the eigenvalue result makes it possible to characterize the systems for which θ^* is not a possible convergence point, i.e. to prove divergence.

The eigenvalue condition is not sufficient for convergence of the estimates. If that is to be shown, the domain of attraction for the stationary point must be contained in a compact subset D_1 of an open connected subset of the stability area D_s. The parameter estimates $\theta(t)$ must infinitely often with probability one belong to D_1. Furthermore the data vector must be bounded by a finite valued random variable C, i.e. $|\varphi(t)| < C$ infinitely often w.p.1 (Ljung 1975). These conditions on $\varphi(t)$ and $\theta(t)$ are referred to as boundedness conditions.

It is possible to make the boundedness condition fulfilled without knowing the true values of the parameters, cf. Ljung (1976c). Thus in order to show asymptotic consistency of the parameter estimation the major step is to construct a Lyapunov function for (3.5) to handle the domain of attraction of the stationary point.

4. Some Aspects on the Construction of Algorithms

In the first section of this chapter estimation of parameters in an ARMA process is treated. A new algorithm with nice local convergence properties is proposed. A more general discussion on the filter $H(q^{-1})$ in symmetric algorithms is given in Section 4.2. Some conditions on the filter, which imply that the eigenvalues of the matrix K in the algorithm have negative real parts are established.

4.1. ESTIMATION OF PARAMETERS IN AN ARMA PROCESS

Consider estimation of the parameters in the time series model

$$A_0(q^{-1})y(t) = C_0(q^{-1})e(t)$$

In the algorithms of ELS type the estimates are calculated by the equations

$$\begin{cases} \theta(t) = \theta(t-1) + \frac{1}{t} \cdot \frac{1}{1+\frac{1}{t}[x_1^T(t)R^{-1}(t-1)x_2(t)-1]} \cdot \\ \qquad\qquad \cdot R^{-1}(t-1)x_2(t)[y(t) - \varphi^T(t)\theta(t-1)] \\ R(t) = R(t-1) + \frac{1}{t}[x_2(t)x_1^T(t) - R(t-1)] \end{cases} \tag{4.1}$$

cf. (2.8) for the basic ELS algorithm and (2.16) for the modified version.

The matrix K, defined by (1.12), is for the algorithm (4.1) given by

$$K(\theta_0) = -[E\, x_2(t,\theta_0)x_1^T(t,\theta_0)]^{-1} E\, x_2(t,\theta_0)\tilde{\varphi}^T(t,\theta_0) \tag{4.2}$$

cf. Section 2.1. The vector $\tilde{\varphi}(t,\theta_0)$ is a filtered data vector

$$\tilde{\varphi}(t,\theta_0) = \frac{1}{C_0(q^{-1})}\varphi(t,\theta_0) \tag{4.3}$$

where $\varphi(t,\theta_0)$ is given in (2.4a).

In the basic ELS algorithm

$$x_1(t) = x_2(t) = \varphi(t)$$

$$x_1(t,\theta_0) = x_2(t,\theta_0) = \varphi(t,\theta_0)$$

where $\varphi(t)$ is given in (2.6a) and (2.7).

The equation (4.3) indicates however that it might be worthwhile to filter data. Since the polynomial $C_0(q^{-1})$ is not known the estimated polynomial $C(q^{-1},t)$ can be used. In Young's modification of the algorithm $x_2(t)$ is filtered by $1/C(q^{-1},t)$, i.e.

$$x_1(t) = \varphi(t)$$

$$x_2(t) = \frac{1}{C(q^{-1},t)}\varphi(t)$$

The expression of the matrix K (4.2) suggests two other algorithms containing filtered data. If both $x_1(t)$ and $x_2(t)$ are filtered, i.e.

$$x_1(t) = x_2(t) = \frac{1}{C(q^{-1},t)}\varphi(t)$$

then

$$x_1(t,\theta_0) = x_2(t,\theta_0) = \frac{1}{C_0(q^{-1})}\varphi(t,\theta_0) = \tilde{\varphi}(t,\theta_0)$$

This gives

$$K(\theta_0) = -[E\,\tilde{\varphi}(t,\theta_0)\tilde{\varphi}^T(t,\theta_0)]^{-1}E\,\tilde{\varphi}(t,\theta_0)\tilde{\varphi}^T(t,\theta_0) = -I$$

i.e. the true value of the parameter vector, θ_0, is always a possible convergence point. This is the Recursive Maximum Likelihood method proposed by Åström. It is analysed in Söderström (1973). It is treated also in Ljung, Söderström and Gustavsson (1975) and in Ljung (1976c). A similar method

is derived by Fucht and Čarapic (1976). When applied to an ARMA process this method converges w.p.1 to the true values of the parameters.

The non-symmetric method, corresponding to

$$x_1(t) = \frac{1}{C(q^{-1},t)}\,\varphi(t)$$

$$x_2(t) = \varphi(t)$$

appears new. This algorithm seems attractive since the local stability of the corresponding differential equation, linearized around $\theta = \theta_0$ is determined by

$$K(\theta_0) = -\left[E\,\varphi(t,\theta_0)\tilde{\varphi}^T(t,\theta_0)\right]^{-1} E\,\varphi(t,\theta_0)\tilde{\varphi}^T(t,\theta) = -I$$

i.e. the true value θ_0 of the parameter vector is always a possible convergence point. Thus this method has desirable local convergence properties. The global convergence properties have however not been studied.

4.2. INFLUENCE OF THE FILTER TRANSFER FUNCTION

Consider a symmetric algorithm with,

$$K(\theta) = -\left[E\,\varphi(t,\theta)\varphi^T(t,\theta)\right]^{-1} E\,\varphi(t,\theta)\tilde{\varphi}^T(t,\theta)$$

cf. (2.39)

$$\varphi(t,\theta) = \left[-z(t-1),\ldots,-z(t-n_f-n_g),\, v(t),\ldots,v(t-n_d)\right]^T$$

$$\tilde{\varphi}(t,\theta) = H(q^{-1})\varphi(t,\theta)$$

Data are assumed to be generated by

$$z(t) = \frac{D(q^{-1})}{F(q^{-1})\cdot G(q^{-1})}\,v(t)$$

$$v(t) = F(q^{-1})\,e(t)$$

where $\{e(t)\}$ is white noise, cf. Theorem 1.

The eigenvalues to $K(\theta)$, that are influenced by the filter are

$$-H(\alpha_i) \qquad \text{where} \quad G^f(\alpha_i) = 0 \quad i = 1,\ldots,n_g$$

cf. Theorem 1. The problem is to explore under what conditions these eigenvalues have negative real parts.

Firstly, suppose that the filter $H(z)$ has all poles and zeroes outside the unit disc and that the poles of the polynomial $G^f(z)$ are real. Then

$$H(x) > 0; \quad x \text{ real}, \ |x| < 1$$

since $H(0) = 1 > 0$. Consequently the eigenvalues of $K(\theta)$ are in the open left half plane and the stationary point is a possible convergence point.

Secondly, if the conditions on $H(z)$ are strengthened by demanding that $H(z)$ is positive real, i.e.

a) $H(x)$ is real for real x
b) $H(z)$ have all poles outside the unit circle
c) $\text{Re } H(e^{i\omega}) > 0 \quad -\pi < \omega \leq \pi$

the eigenvalues of $K(\theta)$ have strictly negative real parts as long as the polynomial $G^f(z)$ is asymptotically stable. Hence, the conditions on the polynomial $G^f(z)$ are relieved and irrespective of the generating system, the corresponding stationary point is a possible convergence point to the algorithm.

Finally, for all other nonpositive real, stable, and minimum phase filters $H(z)$ there is a subset of the unit circle such that if any of the zeroes to the $G^f(z)$ polynomial belongs to it the parameter estimation diverges. It was demonstrated above that the real axis is not contained in this subset.

5. EXAMPLES

The algorithms introduced in Chapter 2 will be further examined. The basic ELS method and the self-tuning regulator are treated simultaneously as well as the modified ELS algorithm and the algorithm by Landau. The corresponding eigenvalue results were given in Theorems 2, 3, 4 and 5.

5.1. THE BASIC ELS METHOD AND THE SELF-TUNING REGULATOR

It is wellknown that the ELS method and the self-tuning regulator might be divergent. Two specific examples of this will be considered.

Example 1. In Ljung, Söderström and Gustavsson (1975) the system

$$y(t) + 0.9y(t-1) + 0.95y(t-2) = e(t) + 1.5e(t-1) + 0.75e(t-2)$$

is shown to give a nonconverging parameter estimation. The filter

$$H(z) = \frac{1}{C(z)} = \frac{1}{1+1.5z+0.75z^2}$$

is nonpositive real. The zeroes of the $A^f(z)$ polynomial are $-0.450 \pm i\ 0.865$ and the eigenvalues $(0.162 \pm i\ 1.383, -1, -1)$. Theorem 2 then implies that the parameter vector with the true values of the parameters is not a possible convergence point. □

Example 2. A self-tuning regulator is applied to the system

$$y(t+1) - 1.6y(t) + 0.75y(t-1) = u(t) + u(t-1) + 0.9u(t-2) + e(t+1) + 1.5e(t) + 0.75e(t-1)$$

in Ljung and Wittenmark (1974). It is demonstrated that the parameter estimation is divergent. The filter $H(z)$ is not positive real. The zeroes of the $B^f(z)$ polynomial are $-0.5 \pm i\ 0.806$ and the eigenvalues $(0.136 \pm 1.643, -1, -1)$.

Thus, referring to Theorem 4, the behaviour of the algorithm is possible to explain by the fact that the minimum variance controller parameters do not constitute a possible convergence point. □

In both of these algorithms $H(z) = 1/C(z)$ where C is asymptotically stable. If it is of first order, i.e. $C(z) = 1 + c_1 z$, $\text{Re } C(z) > 0$ as long as $|z| < 1$. The eigenvalue condition for convergence to the desired convergence point is thus fulfilled. For a second order C-polynomial, i.e. $C(z) = 1 + c_1 z + c_2 z^2$, the stability region in the (c_1, c_2) plane is shown in Figure 1. If (c_1, c_2) belongs to the striped part of this figure $H(z) = 1/C(z)$ is positive real.

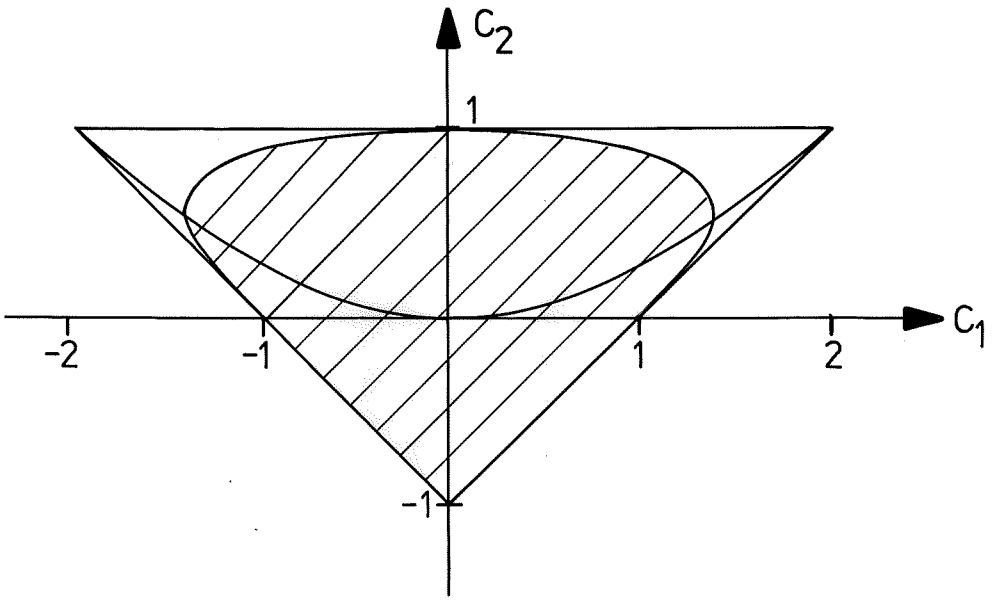


Figure 1. Stability region in the (c_1, c_2) plane for a second order polynomial $C(z) = 1 + c_1 z + c_2 z^2$. For (c_1, c_2) in the striped part the filter $1/C(z)$ is positive real.

Consider the ELS method. It was shown in Chapter 4 that if $H(z)$ is not positive real, there is a subset of the unit disc such that if any zero to $A^f(z)$ belongs to this subset,

the estimation diverges. Take a second order polynomial C and suppose that $A^f(a+ib) = 0$. Then

$$\text{Re } H(a+ib) > 0 \quad \Leftrightarrow \quad \text{Re } C(a+ib) > 0$$

$$[a + c_1/2c_2]^2 - b^2 > -1/c_2 + [c_1/2c_2]^2$$

This is a hyperbola in (a, b) for fixed (c_1, c_2). If $c_1^2 = 4c_2$, i.e. on the parabola in Figure 1, this equation describes two straight lines which for $c_2 < 0.5$ do not intersect the unit circle. For $(c_1, c_2) = (2, 1)$ or $(-2, 1)$ the maximum subset of the unit disc giving divergent parameter estimation is achieved.

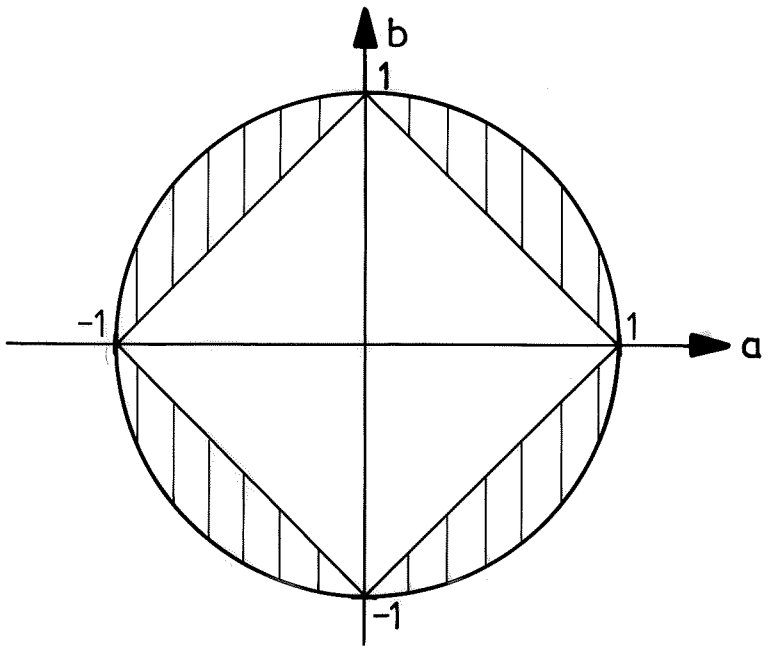


Figure 2. The ELS method applied on a timeseries with $(c_1, c_2) = (2, 1)$ or $(-2, 1)$ diverges if any zero of $A^f(z)$ belongs to the shaded area.

5.2. THE MODIFIED ELS METHOD AND LANDAU's ALGORITHM

In both the modified ELS algorithm and in the algorithm by Landau it is essential to determine $C(\gamma_j)$ where $C^f(\gamma_j) = 0$ $j = 1, \ldots, n_c$. The $C(z)$ polynomial may be factorized as

$$C(z) = \prod_{i=1}^{n_c} (1-z\gamma_i)$$

where γ_i may be complex. To determine the sign of $\text{Re } C(\gamma_j)$ the argument of $C(\gamma_j)$ will be studied.

$C(\gamma_j)$ is a real, positive number for real γ_j. If, however, γ_j is complex then in general

$$\arg C(\gamma_j) = \sum_{i=1}^{n_c} \arg(1-\gamma_j\gamma_i)$$

will not be equal to zero. Hence $\text{Re } C(\gamma_j)$ can be negative.

Consider a third order polynomial

$$C^f(z) = (z-c)(z-(a+ib))(z-(a-ib))$$

The argument of $C(a+ib)$ is

$$\arg C(a+ib) = \arg(1-c(a+ib)) + \arg(1-(a+ib)^2)$$

These two angles are illustrated in Figure 3. Straightforward calculations of the argument gives

$$\tan(\arg C(a+ib)) = \frac{\frac{-cb}{1-ac} + \frac{-2ab}{1-(a^2-b^2)}}{1-(\frac{-cb}{1-ac})(\frac{-2ab}{1-(a^2-b^2)})}$$

which is infinite for

$$c = \frac{1-a^2+b^2}{a(3b^2+1-a^2)} \qquad (5.1)$$

or

$$b^2 = \frac{(1-a^2)(1-ac)}{3ac-1}$$

This defines for constant c a subset A_s of the unit disc in the (a, b)-plane such that for (a, b) in A_s Re C(a+ib) < 0, i.e. the corresponding parameter estimation diverges. A_s is symmetric with respect to both the a- and the b-axes. The part of A_s that is contained in the first quadrant is shown in Figures 4 A, B and C for different values of c. Note that for $c \leq 0.5$ A_s vanishes.

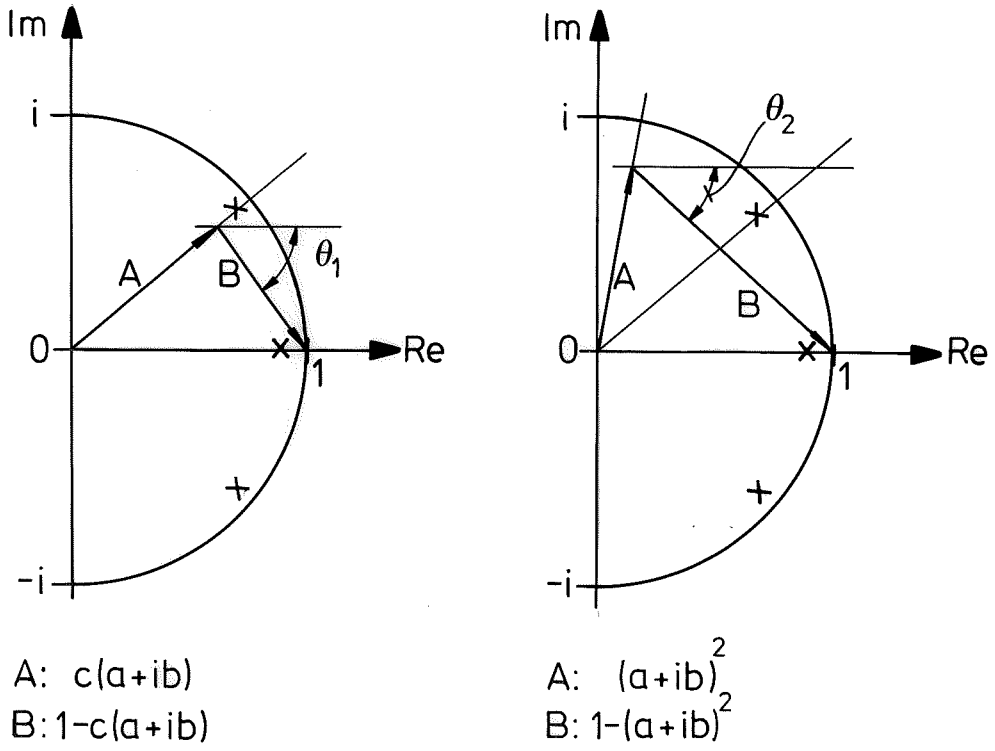


Figure 3. The argument of C(a+ib) is the sum of the two angles θ_1 and θ_2
$\theta_1 = \arg(1-c(a+ib))$ and
$\theta_2 = \arg(1-(a+ib)^2)$.
Clearly $\theta_1 + \theta_2$ can be made greater than $\pi/2$.
The crosses show the zeroes of the polynomial
$C^f(z) = (z-c)(z-(a+ib))(z-(a-ib))$
where $c = |a+ib| = 0.9$ and
$\arg(a+ib) = 40^o$.

Choose $a^2 = 1 - x$ and $b^2 = xy$. The critical value of c (from (5.1)) is then

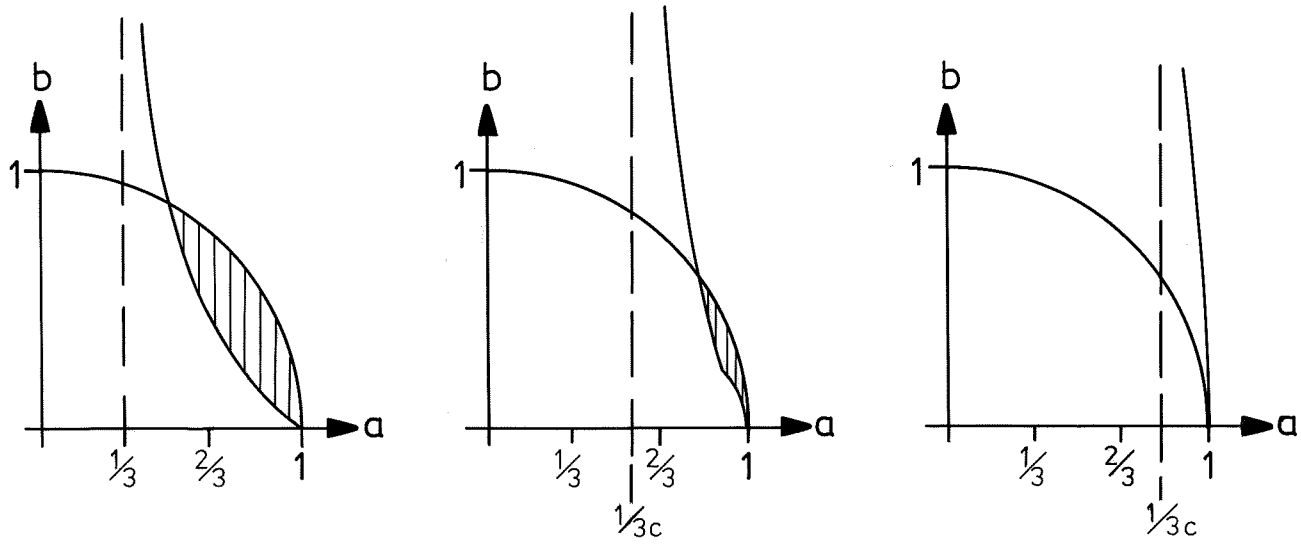
$$c = \frac{1+y}{\sqrt{1-x}(1+3y)}$$

If especially $x = 0.36$ and $y = 0.16/0.36$ the complex poles will be $0.8 \pm i\ 0.4$, with distance from the origin 0.8. The critical value of c is ~ 0.77. Then for the C-polynomial

$$C(z) = (1-0.8z)(1-(0.8+i0.4)z)(1-(0.8-i0.4)z)$$

the eigenvalues $-1/C(\gamma_i)$, $i = 1, 2, 3$ are

$$(-11.97,\ 0.558 \pm i\ 12.58)$$



A) $c = 1$ B) $1/2 < c < 1$ C) $1/3 < c \leq 1/2$

Figure 4. Let $A_s = \{(a, b) \mid a^2 + b^2 < 1,\ c > 0$
$\mathrm{Re}(1-c(a+ib))(1-(a+ib)^2)(1-(a^2+b^2)) < 0\}$
The shaded region is the part of A_s that is in the first quadrant of the unit disc in the (a, b)-plane. A_s is symmetric with respect to both the a- and the b-axes. It vanishes for $c \leq 0.5$.

and with the C-polynomial

$$C(z) = (1-0.9z)(1-(0.8+i0.4)z)(1-(0.8-i0.4)z)$$

the corresponding eigenvalues are

$$(-25.30,\ 2.998 \pm i12.95)$$

This example is applicable to both of the considered algorithms. In the Landau algorithm the nominator polynomial is put equal to 1 (cf. Chapter 2.2).

6. Summary

In this part local convergence of some recursive parameter estimation algorithms has been treated. The results have been applied on some specific algorithms namely the Extended Least Squares algorithm and a modification theoreof, an algorithm proposed by Landau and the self-tuning regulator by Åström and Wittenmark.

The key result is the calculation of the eigenvalues to a matrix occuring in a linearized nonlinear differential equation. This equation describes the asymptotic behaviour of the algorithm and only stable stationary points are possible convergence points to the algorithm. If the matrix is

$$-[E\ \varphi(t)\varphi(t)^T]^{-1}\ E\ \varphi(t)\tilde{\varphi}(t)^T$$

where $\varphi(t)$ is a data vector composed of old output and noise components from the process description

$$y(t) = \frac{D(q^{-1})}{F(q^{-1})G(q^{-1})}\ v(t) = \frac{D(q^{-1})}{G(q^{-1})}\ e(t)$$

and

$$\tilde{\varphi}(t) = H(q^{-1})\varphi(t)$$

with $H(q^{-1})$ a causal, rational and asymptotically stable filter then the eigenvalues are -1 of multiplicity $n_d + n_f(+1)$ and $-H(\gamma_k)$ $k = 1, \ldots, n_g$ where n_d, n_f and n_g are the orders of the D, F and G polynomials and $G^f(\gamma_i) = 0$. Thus, if the filter $H(z)$ is positive real, all the eigenvalues have negative real part since $G^f(z)$ is supposed to be stable.

The eigenvalue result gives useful insight into the properties of the algorithm. It makes it possible to characterize the system for which a stationary point to the algorithm is not a possible convergence point.

This result is then applied on the special algorithms mentioned. For the ELS algorithm and the self-tuning regulator, the filter is $H(q^{-1}) = 1/C_0(q^{-1})$. The true value of the parameters is thus not a possible convergence point for the ELS algorithm if the numbers $-1/C_0(\alpha_i)$ have positive real part. α_i is a zero to the A_0^f-polynomial in the true time series description. When the self-tuning regulator is considered the critical eigenvalues are $-1/C_0(\beta_i)$ where β_i is a zero to the $B_0^f(z)$ polynomial in the true system description.

In the algorithm by Landau, the filter is $H(q^{-1}) = C_0(q^{-1})/A_0(q^{-1})$ where the polynomial $C_0(q^{-1})$ is at user's disposal. The true value of the parameter vector is thus a possible convergence point if $-C_0(\alpha_k)/A_0(\alpha_k)$ has negative real part. α_k is a zero to $A_0^f(z)$.

In the algorithm by Young, finally, the eigenvalues are $-1/C_0(\alpha_k)$ as in the basic ELS but also $-1/C_0(\gamma_i)$ where γ_i is a zero to the $C_0^f(z)$ polynomial.

These necessary conditions for the basic ELS algorithm, the method by Landau and the self-tuning regulator have a counterpart in a result in Ljung (1976 b, c) where he shows that it is sufficient for global convergence w.p. 1 to the true value of the parameter vector that the filter $H(z) - 1/2$ is positive real. In Landau (1976) the same condition (cf. Ljung 1976 c) is used to show that the parameter estimates are asymptotically unbiased. The properties of the filter $H(q^{-1})$ are thus intimately tied to the convergence of the recursive algorithms.

Two of these algorithms, i.e. the basic and the modified ELS method, are apt for estimation of parameters in a time series. Through simple modifications in these algorithms and the corresponding differential equations both the Recursive Maximum Likelihood method and a new algorithm where the true value of the parameter vector always is a possible convergence point for the parameter estimation have been constructed.

7. References

Åström, K.J. (1968): Lectures on the Identification Problem – The Least Squares Method. TFRT-3004, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden

Åström, K.J. (1970): Introduction to Stochastic Control Theory. Academic Press, New York

Åström, K.J. (1974): A Self-Tuning Parameter Estimator. TFRT-3114, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden

Åström, K.J., Borisson, U., Ljung, L. and Wittenmark, B. (1977): Theory and Application of Self-Tuning Regulators. To be published in Automatica 13. This is an expanded version of a paper given at the 6th IFAC World Congress 1975 in Boston, Mass.

Åström, K.J. and Eykhoff, P. (1971): System Identification – A Survey. Automatica 7, 123-162

Åström, K.J. and Wittenmark, B. (1973): On Self Tuning Regulators. Automatica 9, 185-199

Box, G.E.P. and Jenkins, G.M. (1970): Time Series Analysis: Forecasting and Control. Holden-Day, San Francisco

Eykhoff, P. (1974): System Identification. Parameter and State Estimation. John Wiley and Sons, London

Fucht, B.P. and Čarapic, M. (1976): On-Line Maximum Likelihood Algorithm for the Identification of Dynamic Systems. Preprints 4th IFAC Symposium on Identification and System Parameter Estimation, Tbilisi, USSR

Goedheer, L.D. (1976): Comparison of Several Identification Methods and their Mutual Relations. Department of Electrical Engineering, University of Technology, Eindhoven, the Netherlands

Isermann, R., Baur, U., Bamberger, W., Kneppo, P. and Siebert, H. (1974): Comparison of Six On-Line Identification and Parameter Estimation Methods. Automatica 10, 81-103

Kashyap, R.L. (1974): Estimation of Parameters in a Partially Whitened Representation of a Stochastic Process. IEEE Transactions of Automatic Control AC-19, 13-21

Kashyap, R.L. and Rao, A.R. (1976): Dynamic Stochastic Models from Empirical Data. Academic Press, New York

Landau, I.D. (1974): A Survey of Model Reference Adaptive Techniques – Theory and Applications. Automatica 10, 353-379

Landau, I.D. (1976): Unbiased Recursive Identification Using Model References Adaptive Techniques. IEEE Transactions on Automatic Control AC-21, 194-202

Ledwich, G. and Moore, J.B. (1976): Multivarible Self-Tuning Filters. Report, Department of Electrical Engineering, University of Newcastle, Australia

Ljung, L. (1975): Theorems for the Asymptotic Analysis of Recursive Stochastic Algorithms. TFRT-3096, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden

Ljung, L. (1976 a): Analysis of Recursive Stochastic Algorithms. TFRT-7097, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden. To be published in IEEE Tr-AC 22, Aug 1977.

Ljung, L. (1976 b): On Positive Real Transfer Functions and the Convergence of Some Recursive Schemes. TFRT-3138, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden. To be published in IEEE Tr-AC 22, Aug 1977.

Ljung, L. (1976 c): Convergence of an Adaptive Filter Algorithm. Report LiTH-ISY-I-0120, Department of Electrical Engineering, Linköping University, Linköping, Sweden. To appear in Int J Control.

Ljung, L., Söderström, T. and Gustavsson, I. (1975): Counter-examples to General Convergence of a Commonly Used Recursive Identification Method. IEEE Transactions on Automatic Control AC-20, 643-652

Ljung, L. and Wittenmark, B. (1974): Asymptotic Properties of Self-Tuning Regulators. TFRT-3071, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden

Ljung, L. and Wittenmark, B. (1976): On a Stabilizing Property of Adaptive Regulators. Preprints 4th IFAC Symposium on Identification and System Parameter Estimation, Tbilisi, USSR

Panuska, V. (1968): A Stochastic Approximation Method for Identification of Linear Systems Using Adaptive Filtering. Proceedings JACC

Panuska, V. (1969): An Adaptive Recursive Least Squares Identification Algorithm. Proceedings 8th IEEE Symposium on Adaptive Processes

Saridis, G. (1974): Comparison of Six On-Line Identification Algorithms. Automatica 10, 69-79

Söderström, T. (1973): An On-Line Algorithm for Approximate Maximum Likelihood Identification of Linear Dynamic Systems. TFRT-3052, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden

Söderström, T., Ljung, L. and Gustavsson, I. (1974): A Comparative Study of Recursive Identification Methods. TFRT-3085, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden

Talmon, J.L., van den Boom, A.J.W. (1973): On the Estimation of Transfer Function Parameters of Process and Noise Dynamics Using a Single-Stage Estimator. 3rd IFAC Symposium on Identification and System Parameter Estimation, the Hague/Delft

Wittenmark, B. (1973): A Self-Tuning Regulator. TFRT-3054, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden

Young, P.C. (1968): The Use of Linear Regression and Related Procedures for the Identification of Dynamic Processes. Proceedings 7th IEEE Symposium on Adaptive Processes

Young, P.C. (1974): Recursive Approaches to Time Series Analysis. Bulletin of the Institute of Mathematics and its Applications, 10, 209-224

Young, P.C. (1976): Some Observations on Instrumental Variable Methods of Time-Series Analysis. Int. J. Control 23, 593-612

APPENDICES

Notations

All integrals are evaluated along the positively oriented unit circle.

$E(\cdot)$ denotes mathematical expectation.

The subscript 0 on the polynomials in Theorems 2-5 is omitted.

APPENDIX A

Proofs of Lemmas 1 and 2

Proof of Lemma 1 (p.169)

The characteristic equation for the matrix

$$-\left[E\begin{bmatrix}S\\W\end{bmatrix}\begin{bmatrix}S\\W\end{bmatrix}^T\right]^{-1} E\begin{bmatrix}S\\W\end{bmatrix}\begin{bmatrix}\tilde{S}\\\tilde{W}\end{bmatrix}^T = -[E\ MM^T]^{-1}\ E\ M\tilde{M}^T \qquad (A.1)$$

is

$$0 = \det(\lambda I+[E\ MM^T]^{-1}[E\ M\tilde{M}^T]) =$$

$$= \det[E\ MM^T]^{-1}\ \det[\lambda\ E\ MM^T+E\ M\tilde{M}^T]$$

But the covariance matrices EMM^T and $EM\tilde{M}^T$ are blocktriangular since $EWS^T = EW\tilde{S}^T = 0$. Thus the equation to solve is

$$0 = \det[\lambda\ E\ SS^T+E\ S\tilde{S}^T]\cdot\det[\lambda EWW^T + EW\tilde{W}^T]$$

Hence, the first part of the lemma is proven.

The matrix $E\ WW^T$ is diagonal since the elements of W are mutually uncorrelated. The (i, j):th element in $EW\tilde{W}^T$ is

$$E\ w(s-i)\tilde{w}(s-j) = \begin{cases} 0 & i < j \\ E\ w^2(s-i) & i = j \\ x_{ij} & i > j \end{cases}$$

where x_{ij} is uninteresting in this context. Thus

$$0 = \det[\lambda\ E\ WW^T+E\ W\tilde{W}^T] =$$

$$= \det\left[\begin{bmatrix}\lambda\ E\ w^2(s-1) & & 0\\ & \ddots & \\ 0 & & \lambda\ E\ w^2(s-m)\end{bmatrix}+\begin{bmatrix}E\ w^2(s-1) & & 0\\ & \ddots & \\ x_{ij} & & E\ w^2(s-m)\end{bmatrix}\right]$$

$$= (\lambda+1)^m\cdot E\ w^2(s-1)\cdot,,,\cdot E\ w^2(s-m)$$

The lemma is thus proven.

□

Proof of Lemma 2 (p. 170)

First suppose that the zeroes of the A^f polynomial are discrete.

The characteristic equation for the matrix

$$-(ESS^T)^{-1} ES\tilde{S}^T \tag{A.2}$$

$$S = [s_1, \ldots, s_p, s_{p+1}, \ldots, s_n]^T = [S_1^T \; S_2^T]^T$$

is

$$0 = \det(\lambda ESS^T + ES\tilde{S}^T) \tag{A.3}$$

The partitioning of S induces a partitioning of the matrices in (A.3). Each of the resulting four parts is studied separately.

A typical element in $ES_1S_1^T$ is (Åström (1970))

$$E\, s_k(t)s_j(t) = \frac{\sigma^2}{2\pi i} \int \frac{B_k^f(z)}{A^f(z)} \cdot \frac{B_j(z)}{A(z)} \frac{dz}{z} \; ; \; \begin{matrix} k = 1, \ldots, p \\ j = 1, \ldots, p \end{matrix}$$

Since the $A^f(z)$ polynomial is supposed to have all zeroes inside the unit disc and the $B_k^f(z)$ polynomial has at least one zero in the origin, only the residues in the zeroes of the $A^f(z)$ polynomial affect the value of the integral. Thus

$$E\, s_k(t)s_j(t) = \sigma^2 \sum_{r=1}^{n} \mathrm{Res}_{\alpha_r} \left\{ \frac{B_k^f(z)}{A^f(z)} \cdot \frac{B_j(z)}{A(z)} \cdot \frac{1}{z} \right\}; \; \begin{matrix} k = 1, \ldots, p \\ j = 1, \ldots, p \end{matrix}$$

where $\mathrm{Res}_{z_0}\{D(z)\}$ is the residue of $D(z)$ in $z = z_0$. $\{\alpha_r\}$, $r = 1, \ldots, n$ are the zeroes of the $A^f(z)$ polynomial.

Since the zeroes of the $P^f(z)$ polynomial are inside the unit disc the same arguments applied to an element of the matrix $E\, S_1\tilde{S}_1^T$ give

$$E\, s_k(t)\tilde{s}_j(t) = \sigma^2 \sum_{r=1}^{n} \mathrm{Res}_{\alpha_r} \left\{ \frac{B_k^f(z)}{A^f(z)} \cdot \frac{B_j(z)}{A(z)\cdot z} \cdot H(z) \right\}; \; \begin{matrix} k = 1, \ldots, p \\ j = 1, \ldots, p \end{matrix}$$

Next, consider the elements in $E\, S_1S_2^T$ and $E\, S_1\tilde{S}_2^T$. The (k, j):th element in the first of these is

$$E\, s_k(t) z(t-j+1) = \frac{\sigma^2}{2\pi i} \int \frac{B_k^f(z)}{A^f(z)} \cdot \frac{C(z)}{A(z)} \frac{z^{j-1} dz}{z} =$$

$$= \sigma^2 \sum_{r=1}^{n} \mathrm{Res}_{\alpha_r} \left\{ \frac{B_k^f(z)}{A^f(z)} \cdot \frac{C(z)}{A(z)} \cdot z^{j-2} \right\}; \quad \begin{matrix} k = 1, \ldots, p \\ j = 1, \ldots, n-p \end{matrix}$$

and in the second

$$E\, s_k(t) \tilde{z}(t-j+1) = \frac{\sigma^2}{2\pi i} \int \frac{B_k^f(z)}{A^f(z)} \cdot \frac{C(z)}{A(z)} \cdot H(z) \cdot \frac{z^{j-1} dz}{z} =$$

$$= \sigma^2 \sum_{r=1}^{n} \mathrm{Res}_{\alpha_r} \left\{ \frac{B_k^f(z)}{A^f(z)} \cdot \frac{C(z)}{A(z)} H(z) z^{j-2} \right\};$$

$$k = 1, \ldots, p; \; j = 1, \ldots, n-p$$

which follows immediately in the same manner as above.

Next, the (k, j):th element in the matrices $E\, S_2 S_1^T$ and $E\, S_2 \tilde{S}_1^T$ are determined.

$$E\, z(t-k+1) s_j(t) = \frac{\sigma^2}{2\pi i} \int z^{1-k} \cdot \frac{C^f(z)}{A^f(z)} \cdot \frac{B_j(z)}{A(z)} \frac{dz}{z} =$$

$$= \frac{\sigma^2}{2\pi i} \int z^{1-k} \frac{z^{n-p}(c_0 z^p + c_1 z^{p-1} + \ldots + c_p)}{A^f(z)} \cdot \frac{B_j(z)}{A(z)} \frac{dz}{z} =$$

$$= \sigma^2 \sum_{r=1}^{n} \mathrm{Res}_{\alpha_r} \left\{ \frac{(c_0 z^p + \ldots + c_p)}{A^f(z)} \cdot \frac{B_j(z)}{A(z)} \cdot z^{n-p-k} \right\}; \quad \begin{matrix} k = 1, \ldots, n-p \\ j = 1, \ldots, p \end{matrix}$$

and

$$E\, z(t-k+1) \tilde{s}_j(t) = \sigma^2 \sum_{r=1}^{n} \mathrm{Res}_{\alpha_r} \left\{ \frac{(c_0 z^p + \ldots + c_p)}{A^f(z)} \frac{B_j(z)}{A(z)} H(z) z^{n-p-k} \right\}$$

$$k = 1, \ldots, n-p; \; j = 1, \ldots, p$$

which follows from the $A^f(z)$ and $P^f(z)$ polynomials being stable.

Finally, the matrices $E\, S_2 S_2^T$ and $E\, S_2 \tilde{S}_2^T$ with the following (k, j):th elements are studied.

$$E\, z(t-k+1)z(t-j+1) = \frac{\sigma^2}{2\pi i}\int z^{j-k}\,\frac{z^{n-p}(c_0z^p+\ldots+c_p)}{A^f(z)}\cdot\frac{C(z)}{A(z)}\,\frac{dz}{z} =$$

$$= \sigma^2 \sum_{r=1}^{n} \mathrm{Res}_{\alpha_r}\left\{\frac{(c_0z^p+\ldots+c_p)}{A^f(z)}\,\frac{C(z)}{A(z)}\,z^{n-p+j-k-1}\right\}$$

$$k = 1, \ldots, n-p;\ j = 1, \ldots, n-p$$

$$E\, z(t-k+1)\tilde{z}(t-j+1) = \sigma^2 \sum_{r=1}^{n} \mathrm{Res}_{\alpha_r}\left\{\frac{(c_0z^p+\ldots+c_p)}{A^f(z)}\cdot\frac{C(z)}{A(z)}\cdot\right.$$

$$\left.\cdot H(z)\cdot z^{n-p+j-k-1}\right\};\ \begin{matrix} k = 1, \ldots, n-p \\ j = 1, \ldots, n-p \end{matrix}$$

since there are no poles in the origin for any combination of (k, j).

Consequently

$$0 = \det(\lambda\, E\, SS^T + E\, S\tilde{S}^T) = \sigma^{2n}\det\Big[\sum_{r=1}^{n}(\lambda+H(\alpha_r))\pi_r\Big] \qquad (A.4)$$

where π_r is the matrix of residues in α_r. It is of rank one and can be written

$$\pi_r = \begin{pmatrix} \dfrac{B_1^f(\alpha_r)}{\bar{A}^f(\alpha_r)\alpha_r} \\ \vdots \\ \dfrac{B_p^f(\alpha_r)}{\bar{A}^f(\alpha_r)\alpha_r} \\ \dfrac{(c_0\alpha_r^p+\ldots+c_p)\alpha_r^{n-p-1}}{\bar{A}^f(\alpha_r)} \\ \vdots \\ \dfrac{(c_0\alpha_r^p+\ldots+c_p)}{\bar{A}^f(\alpha_r)} \end{pmatrix} \begin{pmatrix} \dfrac{B_1(\alpha_r)}{A(\alpha_r)} \\ \vdots \\ \dfrac{B_p(\alpha_r)}{A(\alpha_r)} \\ \dfrac{C(\alpha_r)}{A(\alpha_r)} \\ \vdots \\ \dfrac{C(\alpha_r)\alpha_r^{n-p-1}}{A(\alpha_r)} \end{pmatrix}$$

where

$$\bar{A}^f(\alpha_r) = (\alpha_r-\alpha_1)\ldots(\alpha_r-\alpha_{r-1})(\alpha_r-\alpha_{r+1})\ldots(\alpha_r-\alpha_n)$$

Hence, the solutions to (A.4) are

$$\lambda = -H(\alpha_r), \quad r = 1, \ldots, n$$

Now, suppose that the A^f polynomial has multiple zeroes. It is then always possible to do a small perturbation in the coefficients with a polynomial $\varepsilon D^f(z)$ where ε is a small number such that the zeroes of the resulting polynomial $A^f(z) + \varepsilon D^f(z)$ are distinct and strictly inside the unit disc. The elements in the matrix $\lambda\ E\ SS^T + E\ S\tilde{S}^T$ then look like

$$\int \frac{M(z)}{[A^f(z)+\varepsilon D^f(z)][A(z)+\varepsilon D(z)]}\, dz =$$

$$= \int \frac{M(z)}{A^f(z)\ A(z)}\, dz - \varepsilon \int \frac{[A^f(z)D(z)+D^f(z)A(z)+\varepsilon D^f(z)D(z)]M(z)\,dz}{A^f(z)A(z)(A^f(z)+\varepsilon D^f(z))(A(z)+\varepsilon D(z))}$$

Therefore, as the eigenvalues of $(E\ SS^T)^{-1}E\ S\tilde{S}^T$ depend continuously on the elements in the matrix, there is a continuous function relating the eigenvalues to the zeroes of the A^f polynomial. Thus if $\varepsilon \to 0$, i.e. the A^f polynomial gets multiple zeroes, a multiple eigenvalue of the same multiplicity as the corresponding zero to the A^f polynomial is formed. Thereby the proof of the lemma is complete. □

APPENDIX B

Proof of Theorem 1 (p.172)

The aim is to construct a new basis in the space spanned by the elements in the vector φ. This new basis will be such that it can be decoupled into two parts which are uncorrelated. It is achieved by partitioning the data into the innovations which are needed to describe the v-elements in φ, i.e. $e(t), \ldots, e(t-n)$, say and, essentially, the predictions of certain z-elements based on information up to and including $n-1$.

Firstly

$$v(s) = F(q^{-1})e(s) = e(s) + f_1 e(s-1) + \ldots + f_{n_f} e(s-n_f); \quad s = t, \ldots, t-n_d \tag{B.1}$$

i.e. the $n_f + n_d + 1$ elements $e(t-i)$, $i = 0, \ldots, n_f + n_d$ of the $\{e(t)\}$ sequence are needed to describe the v-elements in φ.

Secondly, the elements of $\{z(t)\}$ which are included in φ must be described. Separate the cases $n_g < n_d + 1$ and $n_g \geq n_d + 1$.

Suppose that $n_g < n_d + 1$. It will be shown that the innovations above and the minimum mean square error predictions $\hat{z}(t-n_f-i \mid t-n_f-n_d-1)$ of $z(t-n_f-i)$, $i = 1, \ldots, n_g$ can be used to describe all the z-elements. Clearly

$$z(t-i) = \hat{z}(t-i \mid t-n_f-n_d-1) + \varepsilon(t-i) \tag{B.2}$$

where $\varepsilon(t-i)$ is a linear sum of the innovations $e(t-i), \ldots, e(t-n_f-n_d)$ for $i = 1, \ldots, n_f+n_g$ (cf. Box and Jenkins (1970), Åström (1970)). $z(t)$ is given by

$$z(t) = \frac{D(q^{-1})}{F(q^{-1})G(q^{-1})} \quad v(t)$$

or, with (B.1)

$$z(t-i) = -g_1 z(t-i-1) - \ldots - g_{n_g} z(t-i-n_g) + d_0 e(t-i) + \ldots + d_{n_d} e(t-i-n_d) \tag{B.3}$$

The minimum mean square error prediction is given by the conditional mean. Therefore, calculate the conditional mean of (B.3) given information up to and including $t - n_f - n_d - 1$.

$$\hat{z}(t-i|t-n_f-n_d-1) = -g_1 \hat{z}(t-i-1|t-n_f-n_d-1) - \ldots - g_{n_g} \hat{z}(t-i-n_g|t-n_f-n_d-1) + d_0 \hat{e}(t-i|t-n_f-n_d-1) + \ldots + d_{n_d} \hat{e}(t-i-n_d|t-n_f-n_d-1)$$

Since the innovations are independent

$$\hat{e}(t-j|t-n_f-n_d-1) = 0 \qquad j = 1, \ldots, n_f + n_d$$

which means that $\hat{z}(t-i|t-n_f-n_d-1)$ is a linear combination of $\hat{z}(t-n_f-1|t-n_f-n_d-1), \ldots, \hat{z}(t-n_f-n_g|t-n_f-n_d-1)$ for $i = 1,\ldots,n_f$.

Hence, for $n_g < n_d + 1$

$$\varphi(t) = \Omega\eta(t)$$

where Ω is a constant square matrix and

$$\eta(t) = [\hat{z}(t-n_f-1|t-n_f-n_d-1), \ldots, \hat{z}(t-n_f-n_g|t-n_f-n_d-1), , e(t), \ldots, e(t-n_f-n_d)]^T \tag{B.4}$$

Now suppose that $n_g \geq n_d + 1$. In this case the new basis vector is

$$\eta(t) = [\hat{z}(t-n_f-1|t-n_f-n_d-1), \ldots, \hat{z}(t-n_f-n_d|t-n_f-n_d-1), , z(t-n_f-n_d-1), \ldots, z(t-n_f-n_g), e(t), \ldots, e(t-n_f-n_d)]^T \tag{B.5}$$

i.e. there exists a constant square matrix Ω such that

$$\varphi(t) = \Omega\eta(t)$$

This is shown in the same manner as above and this part of the proof is therefore omitted.

Clearly also the filtered versions of the φ and η vectors are related with the same Ω matrices.

The equation for the eigenvalues is then

$$0 = \det[\lambda I + (E\ \varphi\varphi^T)^{-1}\ E\ \varphi\tilde{\varphi}^T] =$$

$$= \det[\lambda I + (E\ \eta\eta^T)^{-1} E\ \eta\tilde{\eta}^T]$$

Both of the $\eta(t)$-vectors can be partitioned as $\eta(t) = (s(t), w(t))^T$ where $w(t) = (e(t), \ldots, e(t-n_f-n_d))^T$. Thus from the construction of $\eta(t)$ it follows that $E\ w(t)s(t)^T = 0$ and since $H(q^{-1})$ is causal also $E\ w(t)\tilde{s}(t)^T = 0$. Lemma 1 then implies that there are $n_f + n_d + 1$ eigenvalues in -1 corresponding to the innovations part in $\eta(t)$.

The remaining eigenvalues are to be determined using Lemma 2 and its corollary. Therefore, consider the calculation of the predictions. They are computed as

$$\hat{z}(t-n_f-i \mid t-n_f-n_d-1) = \frac{K_{n_d+1-i}(q^{-1})}{G(q^{-1})}\, e(t-n_f-n_d-1) \qquad \text{(B.6)}$$

where the polynomial $K_{n_d+1-i}(q^{-1})$ is determined by the identity (Åström 1970)

$$D(q^{-1}) = G(q^{-1})L(q^{-1}) + q^{-n_d-1+i} K_{n_d+1-i}(q^{-1}) \qquad \text{(B.7)}$$

The polynomial $L(q^{-1})$ is of degree n_d-i.

Consider first the case $n_g < n_d + 1$ (B.4). In this case $i = 1, \ldots, n_g$. The degree of the $K_{n_d+1-i}(q^{-1})$ polynomial is n_g-1 independent of i. This is seen if the highest degree terms in the identity (B.7) are considered. Therefore the conditions for the corollary to Lemma 2 are fulfilled. The remaining eigenvalues are thus in this case $-H(\alpha_k)$ where $G^f(\alpha_k) = 0$, $k = 1, \ldots, n_g$.

Finally, study the eigenvalues to the s part of the η vector

when $n_g \geq n_d + 1$. In this case the index i in (B.6) ranges from 1 to n_d. The degree of the $K_{n_d+1-i}(q^{-1})$ polynomial is $n_g - 1$ (from (B.7)). Moreover, since there are n_d predictions in (B.5) all of the conditions for Lemma 2 are fulfilled. Hence, the remaining eigenvalues are also in this case in $-H(\alpha_k)$ where $G^f(\alpha_k) = 0$, $k = 1, \ldots, n_g$. The proof of the theorem is thereby complete. □

Proof of Corollary (p.173)

The corollary follows immediately since the innovations term $e(t)$ in $y(t)$ in the proof of Theorem 1 is needed only to handle $v(t)$ in the theorem. □

APPENDIX C

Proofs of Theorems 2, 3 and 4

Proof of Theorem 2 (p.173)

According to the theorem by Ljung referenced in Chapter 1 the eigenvalues to the matrix

$$-[E\ \varphi(t,\ \theta_0)\varphi^T(t,\ \theta_0)]^{-1}\ E\ \varphi(t,\ \theta_0)\tilde{\varphi}^T(t,\ \theta_0) \qquad (C.1)$$

with

$$\tilde{\varphi}(t,\ \theta_0) = \frac{1}{C_0(q^{-1})}\ \varphi(t,\ \theta_0)$$

must have negative real part when the algorithm is locally convergent to θ_0. Since in this case

$$\varphi(t,\ \theta_0) = [-y(t-1),\ \ldots,-y(t-n_a),\ e(t-1),\ \ldots,\ e(t-n_c)]^T$$

the corollary to Theorem 1 is applicable. It states that there are n_c eigenvalues to (C.1) = (2.14) in -1, and that the remaining are in $-1/C_0(\alpha_k)$ where $A_0^f(\alpha_k) = 0$; $k = 1,\ \ldots,\ n_a$.

It is thus clear that if the real part of any of these numbers is strictly positive, the linearization is unstable which means that the algorithm cannot converge to θ_0. □

Proof of Theorem 3 (p.173)

Since the vector φ is

$$\varphi(t,\ \theta_0) = [-y_M(t-1),\ \ldots,\ -y_M(t-n_a),\ u(t),\ \ldots,\ u(t-n_b)]^T$$

the eigenvalues to the matrix (2.24) are -1 with multiplicity $n_b + 1$ and $-C_0(\alpha_k)/A_0(\alpha_k)$ where $A_0^f(\alpha_k) = 0$, $k = 1,\ \ldots,\ n_a$ according to Theorem 1.

Thus, if any of these numbers have strictly positive real part, convergence to the stationary point θ_0 cannot be achieved. □

Proof of Theorem 4 (p.174)

The φ vector in the self-tuning regulator is

$$\varphi(t) = [-y(t), \ldots, -y(t-s), \beta_0^0 u(t-1), \ldots, \beta_0^0 u(t-n_b-k)]$$

where $\{y(t)\}$ is a moving average of order k

$$y(t) = F(q^{-1})e(t)$$

The input $u(t)$ is calculated from

$$u(t) = - \frac{G(q^{-1})}{b_0^0 B_0(q^{-1})F(q^{-1})} y(t)$$

Thus the eigenvalues of the matrix (2.38) are -1 with multiplicity $s+k+1$ and $-1/C_0(\beta_k)$ where $B_0^f(\beta_k) = 0$; $k = 1, \ldots, n_b$ according to Theorem 1.

The desired result thus follows from the theorem by Ljung referenced in Chapter 1. □

APPENDIX D

Proof of Theorem 5 (p.174)

The proof aims at verifying that the eigenvalues of the matrix (3.4) = (2.18) are given by the numbers mentioned in the formulation of the theorem. The result then follows as above in Theorems 2, 3 and 4. The subscript 0 is omitted.

First suppose that the n_a zeroes of the $A^f(z)$ polynomial and the n_c zeroes of the $C^f(z)$ polynomial are discrete.

The equation to be solved is

$$0 = \det(\lambda + f_2(\theta_0)^{-1}\tilde{f}_1(\theta_0)) =$$

$$= \lambda^{n_a+n_c} \cdot \det f_2(\theta_0)^{-1} \cdot \det(\tfrac{1}{\lambda}\tilde{f}_1(\theta_0) + f_2(\theta_0)) \qquad (D.1)$$

Clearly, $\lambda \neq 0$ since the model is supposed not to be over-parametrized. Denote $1/\lambda$ by μ.

The partitioning of $\varphi(t)$ into

$$\varphi(t) = [-y(t-1), \ldots, -y(t-n_a), e(t-1), \ldots, e(t-n_c)]^T =$$

$$= [\varphi_1, \varphi_2]^T$$

induces a partitioning of $\tilde{\varphi}$ as well as of $\tilde{f}_1(\theta_0)$ and $f_2(\theta_0)$. The elements in these matrices will now be studied.

Consider first the (k, j):th element in $E\,\tilde{\varphi}_1\tilde{\varphi}_1^T$. It is

$$E \frac{1}{A(q^{-1})} e(t-k) \cdot \frac{1}{A(q^{-1})} e(t-j) =$$

$$= \frac{\sigma^2}{2\pi i} \int z^{j-k} \frac{z^{n_a}}{A^f(z)} \frac{1}{A(z)} \cdot \frac{dz}{z}; \quad \begin{matrix} k = 1, \ldots, n_a \\ j = 1, \ldots, n_a \end{matrix}$$

Since the $A^f(z)$ polynomial is supposed to be asymptotically stable and there are no poles in the origin, the value of the covariance is

$$E\ \frac{1}{A(q^{-1})}\ e(t-k)\ \frac{1}{A(q^{-1})}\ e(t-j) =$$

$$= \sigma^2 \sum_{r=1}^{n_a} \mathrm{Res}_{\alpha_r}\left\{\frac{z^{n_a+j-k-1}}{A^f(z)}\cdot\frac{1}{A(z)}\right\};\quad k = 1, \ldots, n_a,\ j = 1, \ldots, n_a$$

where $A^f(\alpha_r) = 0,\ r = 1, \ldots, n_a$.

The same argumentation can now be applied to give the elements in the matrix $E\ \tilde{\varphi}_1\varphi_1^T$ i.e.

$$E\ \frac{1}{A(q^{-1})}\ e(t-k)\cdot\frac{C(q^{-1})}{A(q^{-1})}\ e(t-j) =$$

$$= \sigma^2 \sum_{r=1}^{n_a} \mathrm{Res}_{\alpha_r}\left\{\frac{z^{n_a+j-k-1}}{A^f(z)}\cdot\frac{C(z)}{A(z)}\right\};\quad k = 1, \ldots, n_a,\ j = 1, \ldots, n_a$$

$$A^f(\alpha_r) = 0;\ r = 1, \ldots, n_a$$

An examination of the elements in the matrices $E\ \tilde{\varphi}_1\tilde{\varphi}_2^T$ and $E\ \tilde{\varphi}_1\varphi_2^T$ shows that there are no poles to the integrands except in the zeroes of the $A^f(z)$ polynomial.

In the same manner it is shown that the elements in the matrices $E\ \tilde{\varphi}_2\tilde{\varphi}_1^T$, $E\ \tilde{\varphi}_2\varphi_1^T$, $E\ \tilde{\varphi}_2\tilde{\varphi}_2^T$ and $E\ \tilde{\varphi}_2\varphi_2^T$ are determined by the residues in the zeroes to the $C^f(z)$ polynomial since all the integrands lack poles in the origin.

Consequently the equation (D.1) is

$$0 = \det(\mu\tilde{f}_1(\theta_0)+f_2(\theta_0)) =$$

$$= \det\left[\sum_{k=1}^{n_a}(\mu+C(\alpha_k))\ P_k + \sum_{\ell=1}^{n_c}(\mu+C(\gamma_\ell))Q_\ell\right] \qquad (D.2)$$

where $C^f(\gamma_\ell) = 0,\ \ell = 1, \ldots, n_c$. Here P_k is the matrix of residues in α_k and Q_ℓ the matrix of residues in γ_ℓ. All of these matrices are of rank 1, cf. the proof of Lemma 2. Hence the solutions to (D.2) are $-C(\alpha_k)$, $k = 1, \ldots, n_a$ and $-C(\gamma_\ell)$, $\ell = 1, \ldots, n_c$.

The solutions to (D.1), i.e. the eigenvalues to (2.18), are then $-1/C(\alpha_k)$, $k = 1, \ldots, n_a$ where $A^f(\alpha_k) = 0$ and

$-1/C(\gamma_k)$, $k = 1, \ldots, n_c$ where $C^f(\gamma_k) = 0$.

The same kind of perturbation calculation that was performed in Lemma 2 for multiple zeroes to the $A^f(z)$ polynomial can be performed also here. The result is that the calculated eigenvalues are valid also when either or both of the $A^f(z)$ and $C^f(z)$ polynomials have multiple zeroes.

Invoking the theorem referenced in Chapter 1 the conclusion concerning local convergence follows since if any of the eigenvalues have strictly positive real part, the linearized differential equation is unstable. □