Recursive Least Squares Identification with Forgetting of Old Data

Hägglund, Tore

1983

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

Link to publication

Citation for published version (APA):
Hägglund, T. (1983). *Recursive Least Squares Identification with Forgetting of Old Data*. (Technical Reports TFRT-7254). Department of Automatic Control, Lund Institute of Technology (LTH).

Total number of authors:
1

LUND UNIVERSITY

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

CODEN:LUTFD2/(TFRT-7254)/1-038/(1983)

RECURSIVE LEAST SQUARES IDENTIFICATION WITH FORGETTING OF OLD DATA

TORE HÄGGLUND

DEPARTMENT OF AUTOMATIC CONTROL
LUND INSTITUTE OF TECHNOLOGY

MARCH 1983

LUND INSTITUTE OF TECHNOLOGY
DEPARTMENT OF AUTOMATIC CONTROL
Box 725
S 220 07 Lund 7 Sweden

Document name
Report

Date of issue
March 1983

Document number
CODEN:LUTFD2/(TFRT-7254)/1-038/(1983)

Author(s)
Tore Hägglund

Supervisor
Karl Johan Åström, Per Hagander

Sponsoring organization
Swedish Board of Technical Development
Contract 82-3430

Title and subtitle

Recursive least squares identification with forgetting of old data

Abstract

The identification problem of time-varying systems is treated. Earlier methods to forget old data are reviewed and interpreted in terms of information handling.

A new method is suggested, where a constant amount of information is retained in the estimator. The goal is thus to keep the variance of the estimates at a constant level, even when the excitaion changes. The "P-matrix" is shown to converge to a constant diagonal matrix. The new method solves the problems caused by nonuniform excitation (both in time and in space), e.g. wind-up in the estimator.

Key words

Classification system and/or index terms (if any)

Supplementary bibliographical information

ISSN and key title

ISBN

Language
English

Number of pages
38

Security classification

Recipient's notes

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

RECURSIVE LEAST SQUARES IDENTIFICATION

WITH FORGETTING OF OLD DATA

Tore Hägglund

Department of Automatic Control
Lund Institute of Technology
March 1983

Contents

1. INTRODUCTION

Since the beginning of the seventies, adaptive control has been an area of increasing research interest. Several theoretical results concerning stability and convergence have been reported in the literature as well as successful implementations. A résumé is given in Åström (1981).

Most of the theoretical work has been devoted to asymptotic properties such as convergence and stability, while much less attention has been paid to the transient behaviours. The analysis is mostly carried out under the assumption of constant system parameters, while the main purpose of the adaptive controller is the ability to adapt to time-varying systems. The reason is, that the general adaptive control problem, being both nonlinear and stochastic, is so difficult. It has not yet been possible to derive theoretical results apart from in rather restricted cases. Since the ability of handling time-varying parameters nevertheless is a key problem, it has been faced in the applications. It is not surprising, that most progress in solving the problem is to find in the application literature.

A self-tuning regulator can be thought of as composed of some distinct parts, where the estimator is an essential one. It is in the parameter estimator problem arise when the parameters are time-varying. There are many different methods which could be used for parameter estimation, e.g. recursive least squares, stochastic approximation, instrumental variable and maximum likelihood methods. A review of recursive estimation methods is given in Söderström et al (1974).

In the estimator, a model of the plant is to be derived from inputs and measured output data. If the plant to be identified is time-varying, old input and output pairs may be bad descriptions of the actual model, and should therefore be discounted. The most common way of discounting old data is to use a forgetting factor (λ), see Åström and Wittenmark (1973). This will cause an exponential weighting of the data, since a measurement that was received n samples ago is weighted proportional to

$$\lambda^n = e^{n\cdot\ln(\lambda)} \qquad 0 < \lambda \le 1$$

The choice of λ is a trade-off between fast adaptation and long term quality of the estimates. This trade-off can sometimes be unsatisfactory. It is desirable to discount quickly when the model is changing or just has changed and to make a slower discounting when the parameters are constant or the excitation is bad.

Another problem that may occur when a constant forgetting factor less than one is used is the estimation wind-up. The method of exponential weighting of the incoming data works well if the incoming information is uniformly distributed. Especially in the servo problem, when the major excitations come from the variations of the command signal, this is not the case. (This may be one reason why most applications are devoted to the regulator problem.) Discounting during time intervals of bad excitation may then be to large, leading to uncertain estimates and numerical problems.

In order to avoid these problems, attempts to use time-variable forgetting factors have been made. E.g. Fortesque et al (1981) and Wellstead and Sanoff (1981) use a forgetting factor which is dependent on the magnitude of the residuals, i.e. the output prediction error. Irving (1979) proposed a forgetting factor that keeps the trace of the P matrix constant. The P matrix, which is a gain factor in the estimator, is defined in the next chapter.

The estimation can also be restarted repeatedly, instead of using a forgeting factor. This is successfully practiced by Evans and Betz (1982).

In all the proposals above, little is assumed about the nature of the parameter variations. When more a priori information is present, more sophisticated solutions are possible. If the parameters e.g. can be modeled by stochastic difference equations

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

where A and the statistics of $v(t)$ are known, the Extended Kalman filter is suitable. In Åström (1980), the problem of estimating parameters which are a sum of an ARMA signal and a piece-wise deterministic signal is considered. Several papers have also been devoted to the problem when the parameters switch between a limited number of sets, see e.g. Wittenmark (1979) and Millnert (1982).

In this report, the problem of estimating parameters in time-varying systems will be discussed from an information handling point of view. Old proposals are first reviewed in these terms. The consideration leads to a new suggested way of facing the problem. A recursive identification procedure is derived where the discounting is dependent on the information available and the amount of incoming information. The estimator discounts past data in such a way that a specified accuracy of the parameter estimates is obtained.

2. IDENTIFICATION OF SYSTEMS WITH TIME-VARYING PARAMETERS

In this chapter, identification of time-varying systems is discussed. The recursive least squares method is first examined and former proposals of modifications to treat time-varying parameters are explained in terms of information handling. The discussion leads to a suggested new approach, where a constant amount of information about the parameters is retained.

2.1 The recursive least squares algorithm

Throughout this report, discrete time single-input single-output systems described by the equation

$$y(t) = \theta(t-1)^T \varphi(t) + e_n(t) \qquad (2.1)$$

where $\{y(t)\}$ is the measured output sequence, $\{\theta(t)\}$ are the parameters to be estimated, $\{\varphi(t)\}$ vectors containing past inputs and outputs and finally $\{e_n(t)\}$ a noise sequence of independent random variables, will be considered. Among all possible identification methods, the recursive least squares (LS) estimator has become the most common. For a thorough description of the LS method, see Kendall and Stuart (1961). In the (weighted) LS estimation procedure, the vector $\hat{\theta}(t)$ which minimizes the loss function

$$L(\theta) = \sum_{i=1}^{t} \frac{1}{\omega(t,i)} [y(i) - \theta^T \varphi(i)]^2 \qquad (2.2)$$

with respect to θ is chosen. A desirable choice of the weights $\omega(t,i)$ would be the variances of the corresponding measurements. Compare with the minimum variance estimator in case of known regression vectors $\{\varphi(t)\}$. As will be shown below, the key problem in identification of time-varying systems is the lack of knowledge about these variances.

At every time instant t, an estimation of the parameters $\theta(t)$ based on measurements in the period $[0,t]$ is to be done. From Equation (2.1), the relation between the measurements and the parameters $\theta(t)$ can be derived as

$$y(i) = \theta(i-1)^T \varphi(i) + e_n(i) =$$

$$= \theta(t-1)^T \varphi(i) + (\theta(i-1) - \theta(t-1))^T \varphi(i) + e_n(i) \stackrel{\Delta}{=}$$

$$\stackrel{\Delta}{=} \theta(t-1)^T \varphi(i) + e_m(t,i) + e_n(i) \qquad (2.3)$$

Comparing Equation (2.3) with (2.1), it is seen that the old measurements are corrupted by not only noise disturbances, but also an error originating from changes of the model. This model error may be caused by e.g. a change of working point in a nonlinear system, changes in the process depending on temperature variations, wear or aging, failing sensors or actuators.

In the LS method, each measurement is weighted depending on its uncertainty, see Equation (2.2). As seen above, this uncertainty has two components, namely the measurement noise e_n and the model error e_m. In the sequel, it will be more convenient to work with the corresponding variances instead of the errors. Let $\sigma(t,i)^2$ denote the variance of the measurement y(i) when applied to the estimation at time t. It can then be separated into two components

$$\sigma(t,i)^2 = \sigma_m(t,i)^2 + \sigma_n(i)^2 \qquad (2.4)$$

where $\sigma_m(t,i)^2$ is the component caused by the model error and $\sigma_n(i)^2$ is the noise variance at time i. As said above, it would be desirable to choose the weights $\omega(t,i)$ in the loss function $L(\theta)$ equal to $\sigma(t,i)^2$.

<u>Examples</u>

1. $\sigma_m(t,i)^2 = 0$, $\sigma_n(i)^2 = \sigma^2$.
Both the parameters and the noise level are constant. All the measurements then have the same uncertainty, i.e. $\sigma(t,i)^2 = \sigma^2$.

2. $\sigma_m(t,i)^2 = ((1/\lambda)^{t-i}-1)\sigma^2$, $\sigma_n(i)^2 = \sigma^2$.
The noise level is constant and the parameters of the model are slowly time-varying. The uncertainty of the measurements is increasing exponentially with time, i.e. $\sigma(t,i)^2 = (1/\lambda)^{t-i}\sigma^2$. This case corresponds to the well-known

discounting with a constant forgetting factor. □

The LS estimate of $\theta(t)$ is given by

$$\hat{\theta}(t) = (\Phi(t)^T V(t)^{-1} \Phi(t))^{-1} \Phi(t)^T V(t)^{-1} Y(t) \tag{2.5}$$

where

$$\Phi(t) = \begin{bmatrix} \varphi(1)^T \\ \varphi(2)^T \\ \vdots \\ \varphi(t)^T \end{bmatrix}, \quad Y(t) = \begin{bmatrix} y(1) \\ y(2) \\ \vdots \\ y(t) \end{bmatrix}, \quad V(t) = \mathrm{diag}(\omega(t,i))$$

See Kendall and Stuart (1961). To simplify the notations in the sequel, the notation

$$v(t) = \omega(t,t) = \hat{\sigma}_n(t)^2$$

will be used. In case of constant parameters, i.e. $\sigma_m(t,i) = 0$, and known noise variances, $\omega(t,i) = v(i)$ and the recursive version of the LS algorithm becomes

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \frac{1}{v(t)} P(t)\varphi(t)\varepsilon(t)$$

$$\varepsilon(t) = y(t) - \varphi(t)^T \hat{\theta}(t-1) \tag{2.6}$$

$$P(t) = P(t-1) - \frac{P(t-1)\varphi(t)\varphi(t)^T P(t-1)}{v(t) + \varphi(t)^T P(t-1)\varphi(t)}$$

where

$$P(t) = (\Phi(t)^T V(t)^{-1} \Phi(t))^{-1} \tag{2.7}$$

In case of normal distribution of the data, $P(t)^{-1}$ is an estimate of Fisher's information matrix. $P(t)^{-1}$ will be used as a measure of the information available at time t throughout this report.

The recursive version of the LS algorithm in case of time-varying parameters is mostly not as simple as Equations (2.6). In some restrictive cases, it is however possible to get fairly compact expressions. A familiar example is given below.

Example

If $\sigma_m(t,i)^2 = ((1/\lambda)^{t-i}-1)\sigma^2$ and $\sigma_n(i)^2 = \sigma^2$, the equations become

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \frac{1}{\sigma^2} P(t)\varphi(t)\varepsilon(t)$$

$$\varepsilon(t) = y(t) - \varphi(t)^T\hat{\theta}(t-1) \tag{2.8}$$

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

These equations of LS estimation with exponential discounting of old data are often used in adaptive control. Normally, the P(t)-matrices are scaled with σ^2. It is then not necessary to know the value of σ^2.

□

2.2 The weighting problem

The purpose of LS estimation is to find the parameters $\theta(t)$ which minimize the loss function (2.2), where the weighting coefficients are equal to the corresponding variances. In case of known variances $\sigma(t,i)^2$, the solution is simply given by Equation (2.5). The problem is however, that $\sigma(t,i)$ is normally not known, neither is $\sigma_m(t,i)$ nor $\sigma_n(i)$. Therefore, $\sigma(t,i)$ must be estimated or hypothesized in some way.

Before the new proposed method of choosing $\sigma(t,i)$ is presented, some former choices will first be reviewed. Although people have not always thought in terms of variances of the measurements, old proposals for handling time-varying parameters in LS-estimation can be explained in these terms.

First of all, some assumptions about $\sigma_m(t,i)$ and $\sigma_n(i)$ must be stated. The problem is meaningless if $\sigma_m(t,i)$ is varying as much as the states $\varphi(t)$. The notation of variance also becomes meaningless if the variance is varying as much as the stochastic variable itself. The following assumption is therefore made.

Assumption 1: If the estimated parameters or the noise level vary, they vary slowly and/or seldom compared with the time constants of the system.

□

The assumption means e.g. that large step changes may not occur frequently. The assumption should not be any limitation, but rather a check that the problem is well formulated.

The proposed methods to choose $\sigma(t,i)$ can be grouped according to further assumptions that are put upon $\sigma_m(t,i)$ and $\sigma_n(i)$. Each method belongs to one of the following four cases.

	Constant parameters $\sigma_m(t,i) = 0$	Time-varying parameters
Constant noise level $\sigma_n(i) = \sigma$	1	2
Time-varying noise level	3	4

Case 1. This is a very common assumption, especially in the analysis. The problem, which is more a tuning problem than a pure adaptive control problem, is rather easy to solve. The LS algorithm becomes independent of the noise level σ. A forgetting factor equal to one, or e.g. $\lambda = 1 - \exp(-t/T)$ to eliminate erroneous initial values, can be used.

Case 2. This is also a common assumption. It is often also assumed that the parameters change slowly, and a forgetting factor less than one is used, i.e. $\sigma_m(t,i)^2 = ((1/\lambda)^{t-i}-1)\sigma^2$. If λ is constant, it is at least implicitly also assumed that the parameters are changing all the time at a regular rate.

Experiments with different forgetting factors in different elements of the P matrix have been made. The reason is then a priori knowledge about differences concerning rates of changes or amount of incoming information of the different parameters

Sometimes, a time-varying forgetting factor is used, combined with some technique to estimate variations in $\sigma_m(t,i)$, see e.g. Kershenbaum et al (1981). It is then no longer assumed that the parameters are changing slowly or at a constant rate.

Case 3. This problem is very seldom treated, if treated at all.

Case 4. This problem is also seldom treated, especially in the analysis. This report is devoted to the problem 4.

With the additional assumptions described above, the LS algorithm mostly becomes quite simple, compare e.g. Equations (2.8). The corresponding adaptive controllers usually work well, if the assumptions are fulfilled. It is however also well-known that the algorithms can behave badly if the assumptions are violated. Some examples are given below.

Case 1. If the parameters would vary, a very slow adaptation will result. The controller will behave almost like a constant regulator.

Case 2. If the parameters are varying at a slower rate than assumed, the uncertainty of the estimates will get unnecessarily large. A remarkable situation is the so called "estimation wind-up" when the P matrix "explodes", though the parameters are constant. If on the other hand the parameters are varying faster than assumed, the convergence will be slow.

The value of the forgetting factor is chosen as a compromise between fast adaptation and high stationary accuracy of the estimates. An increasing noise level will cause an equivalent increase of the uncertainty of the parameter estimates. The old value of the forgetting factor is then often a bad choice.

The use of variable forgetting factors is often even more dependent on the assumptions. An increase of the noise level $\sigma_n(i)$ will in most algorithms be interpreted as a variation of the parameters, i.e. an increase of $\sigma_m(t,i)$. This is a serious mistake. It means that the algorithm believes that old measurements are more uncertain than the new ones, while the situation is the opposite. The result is, that old information is thrown away, when the algorithm instead should take extra care of those measurements, bearing in mind the poor information that will come in the future. The situation is exemplified in Chapter 4.

2.3 How should the weighting problem be solved?

In the previous section, it was shown that a reasonable weighting of the measurements could be made if the system fulfilled certain further assumptions. It was however also shown that severe problems could occur if these assumptions were violated.

In this section, the more general problem will be treated, i.e. no further assumptions than Assumption 1 will be put upon the system. It corresponds to case 4 in the previous section.

There are two different types of problems concerning the information handling in the LS estimator. The first one is that the incoming information may be poor, because of bad excitation or large variations of the parameters. The problem can be solved by dual control. It means that, when the incoming information is too poor, information is actively forced into the estimator by adding extra signals to the input.

The other type of problem is when the incoming information is sufficient, but it is handled incorrectly as described in the previous section. In other words, the problem is caused by bad correspondence between real and estimated values of $\sigma_m(t,i)$ and $\sigma_n(i)$. The situation can be improved by better estimates of these variances. Remember that it is very important to distinguish between the two variances, since they imply opposite actions.

The accuracy obtainable in estimating the variances is unfortunately rather poor. By fault detection, it is possible to detect large increases of $\sigma_m(t,i)$. There will however always be some time delay between the parameter change and the detection, and the estimate will only be of the quantitative form "it has increased". It is possible to get a fairly accurate estimate of $\sigma_n(i)$ under stationary conditions. It will however take some time before fast changes of the noise level are distinguished from parameter changes. The a priori knowledge about $\sigma_m(t,i)$ and $\sigma_n(i)$ is mostly poor.

The purpose of the LS estimator is to provide estimates of the parameter vector $\theta(t)$ with a reasonable accuracy. Depending on the lack of knowledge about the uncertainties of the different measurements, the weighting problem becomes troublesome. Many of the problems are caused by the way this weighting is done. Instead of using assumptions about how parameters and noise level may vary, a different approach will be suggested, where the overall problem, namely the accuracy of the estimates $\hat{\theta}(t)$ is considered. As will be shown, several problems can be avoided by relating the weighting directly to the accuracy, i.e. the amount of information available, and to the incoming information. The following method is therefore proposed.

> Discount past data in such a way that, if the parameters were constant, a constant desired amount of information is retained.

The information amount is defined in Equation (2.7) by the inverse P matrix. More precisely, the algorithm that will be presented in the next chapter transforms the P matrix to a diagonal matrix with equal diagonal elements. The diagonal elements may be interpreted as approximations of the variances of the corresponding parameters. The weights $\omega(t,i)$ are therefore to be chosen such that this property is reached. The time horizon will consequently vary, depending on the incoming information. If the signals are noisy with a small information content, the time horizon will be long. If no information is coming in at all, nothing will be thrown away. If on the other hand the incoming information content is large, old measurement can be discounted quickly and a fast adaptation to new parameter values is possible. The method requires an estimate of the noise level.

The inverse P matrix is a good measure of the information content only in case of constant parameters. In case of parameter changes, the inverse P matrix will falsely indicate good estimation accuracy, leading to unnecessarily slow parameter adaptation. A fault detection procedure will therefore be incorporated to speed up the adaptation in case of parameter changes by increasing the P matrix.

3. A CONSTANT INFORMATION ESTIMATOR

A recursive parameter estimator derived according to the ideas presented in Chapter 2 will now be presented. The structure for the updating of the P matrix is first determined. Then the equations describing the updating of the parameter estimates $\hat{\theta}(t)$ are derived. Finally, it is shown that the algorithm will provide an estimator that stationary retains a constant desired amount of information. For sake of simplicity, it will be assumed that the parameters are constant throughout the derivation of the updating formulae. The fault detection procedure will therefore not be introduced until the end of the chapter, when the time-varying case is considered.

3.1 Updating of the P matrix

As said before, the information content of the estimator is defined as the inverse P matrix. The goal for the estimator is to weight the incoming data in such a way that the P matrix is transformed into a diagonal matrix with equal diagonal elements, say $a \cdot I$. The value of a is an approximation of the variances of the parameter estimates.

According to Equations (2.5) and (2.7), the LS estimator is given by

$$\hat{\theta}(t) = (\Phi(t)^T V(t)^{-1} \Phi(t))^{-1} \Phi(t)^T V(t)^{-1} Y(t) =$$

$$= P(t)\Phi(t)^T V(t)^{-1} Y(t) \qquad (3.1)$$

Let $V(t;t+k)$ denote the upper left quadratic t-dimensional submatrix of $V(t+k)$. The updating of the information matrix is then given by

$$P(t)^{-1} = \Phi(t)^T V(t)^{-1} \Phi(t) =$$

$$= (\Phi(t-1)^T \ \varphi(t)) \begin{bmatrix} V(t-1;t)^{-1} & 0 \\ 0 & v(t)^{-1} \end{bmatrix} \begin{bmatrix} \Phi(t-1) \\ \varphi(t)^T \end{bmatrix} =$$

$$= \Phi(t-1)^T V(t-1;t)^{-1} \Phi(t-1) + \varphi(t) v(t)^{-1} \varphi(t)^T \qquad (3.2)$$

It is seen from Equation (3.2) that the new information only affects the inverse P matrix in the $\varphi(t)$ direction. If the discounting is made properly, it is only necessary to discount data in the direction where the new information is entering. Therefore choose $V(t-1;t)$ such that

$$\Phi(t-1)^T V(t-1;t)^{-1}\Phi(t-1) = \Phi(t-1)^T V(t-1)^{-1}\Phi(t-1) - \alpha(t)\varphi(t)\varphi(t)^T \qquad (3.3)$$

where $\alpha(t)$ is a scalar. The $P(t)$ matrix is then updated as

$$P(t) = [\Phi(t-1)^T V(t-1)^{-1}\Phi(t-1) + (v(t)^{-1}-\alpha(t))\varphi(t)\varphi(t)^T]^{-1} =$$

$$= P(t-1) - \frac{P(t-1)\varphi(t)\varphi(t)^T P(t-1)}{(v(t)^{-1}-\alpha(t))^{-1} + \varphi(t)^T P(t-1)\varphi(t)} \qquad (3.4)$$

Obviously, $\alpha(t)$ must be nonnegative since a negative $\alpha(t)$ would mean an addition of information instead of a removal. Comparing Equations (3.2) and (3.3) is seen that $\alpha(t)$ should be equal to $v(t)^{-1}$ in stationarity, i.e. when the P matrix has reached its desired value. From the equations above it is also concluded that the P matrix may get nonpositive if too large values of $\alpha(t)$ are chosen. The following theorem gives a bound on $\alpha(t)$ such that the P matrix is positive definite if $\alpha(t)$ is chosen within those bounds.

Theorem 3.1: Given a sequence of matrices $\{P(t)\}$ which are satisfying Equation (3.4). If the initial matrix $P(0)$ is positive definite, then $P(t)$ will be positive definite for all t if $\alpha(t)$ lies within the bounds

$$0 \le \alpha(t) < v(t)^{-1} + \frac{1}{\varphi(t)^T P(t-1)\varphi(t)}$$

Proof: The proof is divided into two parts. First study the interval

1. $0 \le \alpha(t) \le v(t)^{-1}$

The inverse P matrix is updated as

$$P(t)^{-1} = P(t-1)^{-1} + (v(t)^{-1}-\alpha(t))\varphi(t)\varphi(t)^T$$

Since the second term is nonnegative, it is clear from recursion that the inverse P matrices and consequently the P matrices are positive definite.

2. $v(t)^{-1} < \alpha(t) < v(t)^{-1} + \dfrac{1}{\varphi(t)^T P(t-1)\varphi(t)}$

If $\alpha(t)$ is chosen in this interval, the second term in Equation (3.4) always becomes positive. By recursion it is therefore concluded that the P matrices stay positive definite even in this interval.

□

Summing up, the updating of the P matrix is derived above. It is given by Equation (3.4) with the bounds on $\alpha(t)$ given in Theorem (3.1). Later on, $\alpha(t)$ will be defined such that the desired properties of the estimator are obtained.

3.2 Updating of the estimates $\hat{\theta}(t)$

The new way of updating the P matrix described above will of course influence the updating of the estimates. The new equations will be derived from Equation (3.1), by using the following lemma.

Lemma 3.1: If $\Phi(t-1)$ has full rank (which is assumed throughout the report), the following equation holds.

$$\Phi(t-1)^T V(t-1;t)^{-1} Y(t-1) = \Phi(t-1)^T V(t-1)^{-1} Y(t-1) - \mu(t)\varphi(t) \tag{3.5}$$

where $\mu(t)$ is a scalar.

Proof: $(V(t-1)^{-1} - V(t-1;t)^{-1})$ is symmetric, and can therefore be diagonalized by an orthogonal matrix $Q(t)$, i.e.

$$(V(t-1)^{-1} - V(t-1;t)^{-1}) = Q(t)^T \Lambda(t) Q(t)$$

where $\Lambda(t)$ is diagonal. Equation (3.3) can now be written as

$$\Phi(t-1)^T Q(t)^T \Lambda(t) Q(t) \Phi(t-1) = \alpha(t)\varphi(t)\varphi(t)^T$$

The right hand side is a rank one matrix. All matrices to the left except $\Lambda(t)$ are of full rank. $\Lambda(t)$ must consequently be of rank one, and since it is diagonal, all elements except one must be zero. Hence

$$\Phi(t-1)^T Q(t)^T \Lambda(t) = \varphi(t) z(t)^T$$

where $z(t)$ is a vector. Finally

$$\Phi(t-1)^T(V(t-1)^{-1}-V(t-1;t)^{-1})Y(t-1) =$$

$$= \Phi(t-1)^T Q(t)^T \Lambda(t) Q(t) Y(t-1) = \varphi(t) z(t)^T Q(t) Y(t-1) =$$

$$= \mu(t)\varphi(t) \qquad \square$$

The updating formula for the parameter estimates can now be derived from Equation (3.1).

$$\hat{\theta}(t) = P(t)\Phi(t)^T V(t)^{-1} Y(t) =$$

$$= P(t)(\Phi(t-1)^T \ \varphi(t)) \begin{bmatrix} V(t-1;t)^{-1} & 0 \\ 0 & v(t)^{-1} \end{bmatrix} \begin{bmatrix} Y(t-1) \\ y(t) \end{bmatrix} =$$

$$= P(t)(\Phi(t-1)^T V(t-1;t)^{-1} Y(t-1) + \varphi(t) v(t)^{-1} y(t)) \qquad (3.6)$$

Using Lemma 3.1 in Equation (3.6) gives

$$\hat{\theta}(t) = P(t)(\Phi(t-1)^T V(t-1)^{-1} Y(t-1) + (v(t)^{-1} y(t) - \mu(t))\varphi(t)) =$$

$$= \left[I - \frac{P(t-1)\varphi(t)\varphi(t)^T}{(v(t)^{-1}-\alpha(t))^{-1} + \varphi(t)^T P(t-1)\varphi(t)} \right] P(t-1) \cdot$$

$$\cdot (\Phi(t-1)^T V(t-1)^{-1} Y(t-1) + (v(t)^{-1} y(t) - \mu(t))\varphi(t)) =$$

$$= \hat{\theta}(t-1) + \frac{P(t-1)\varphi(t)}{(v(t)^{-1}-\alpha(t))^{-1} + \varphi(t)^T P(t-1)\varphi(t)} \cdot$$

$$\cdot \left[\frac{1}{1-\alpha(t)v(t)} y(t) - \frac{v(t)}{1-\alpha(t)v(t)} \mu(t) - \varphi(t)^T \hat{\theta}(t-1) \right]$$

$$(3.7)$$

where Equation (3.4) is used in the second equality.

It is now time to determine $\mu(t)$. The value of $\mu(t)$ is defined by Equations (3.3) and (3.5). It is dependent of $\Phi(t-1)$, $V(t-1)$, $Y(t-1)$ and $\varphi(t)$. Since the full $\Phi(t-1)$, $V(t-1)$ and $Y(t-1)$ matrices are not stored, a modified $\mu(t)$ has to be used. If $y(t)$ is equal to the predicted output, i.e. $y(t) = \varphi(t)^T \hat{\theta}(t-1)$, no change of the estimated