V. Petridis and Ath. Kehagias.
"A Multi-Model Algorithm for Parameter Estimation of Time Varying Nonlinear Systems".

This paper has appeared in the journal:
Automatica, Vol.34, pp.469-475, 1998.

A Multi-model Algorithm for Parameter Estimation of Time-varying Nonlinear Systems

V. Petridis [a] and Ath. Kehagias [b]

[a] *Department of Electrical and Computer Engineering*
Aristotle University of Thessaloniki
Thessaloniki, GR 54006
Greece
e-mail: petridis@vergina.ee.auth.gr
fax: +3031-996331, tel: +3031-996331

[b] *Department of Mathematics and Computer Science*
American College of Thessaloniki
Thessaloniki, GR 55510
Greece
and
Department of Electrical and Computer Engineering
Aristotle University of Thessaloniki
Thessaloniki, GR 54006
Greece
e-mail: kehagias@egnatia.ee.auth.gr

Abstract

We present a new on-line multi-model algorithm for parameter estimation of time-varying nonlinear systems. The time-variation is captured by assuming that the system parameters change according to a Markovian mechanism. The algorithm postulates a finite number of possible values of the system parameter and computes recursively the credit function of each parameter value, according to its predictive accuracy. A convergence analysis of the algorithm is presented which indicates that the algorithm estimates correctly the parameter value, in the time intervals between source switchings. This conclusion is corroborated by numerical experiments.

Key words: Parameter Estimation; System Identification; Nonlinear Systems; Multiple Models.

1 Introduction

Many methods have been developed to solve the problem of parameter estimation for dynamical systems (Ljung, 1987). Of particular interest is the case of on-line algorithms which are used to estimate time-varying parameters. Here we present such an algorithm which assumes a *nonlinear* dynamical system. The system is time-varying: its parameter changes values according to a Markovian model switching mechanism. The algorithm starts with a finite number of models, each corresponding to one of the parameter values, and selects the "phenomenologically best" parameter value; namely the one which produces the best fit to the observed behavior of the system. Our algorithm is related to the *Partition Algorithm* (PA) presented in (Hilborn & Lainiotis, 1969; Lainiotis, 1971; Lainiotis & Plataniotis, 1994; Sims, Lainiotis & Magill, 1969). PA is suitable for the parameter estimation of a linear dynamical system with Gaussian noise in the input and output; no provision is made for model switching. Under these assumptions, an algorithm is developed for exact computation of the models' posterior probabilities; these are used for Maximum a Posteriori (MAP) estimation of the unknown parameter. This method has been used extensively in a number of applications, including parameter estimation and system identification (Kehagias, 1991; Lainiotis & Plataniotis, 1994; Petridis, 1981).

Our algorithm is more general than the PA: it applies to nonlinear systems and requires no probabilistic assumptions regarding the noise. Furthermore, while there are several convergence studies of the PA *without a switching mechanism* (Anderson & Moore, 1979; Kehagias, 1991; Tugnait, 1980), as far as we know, the analysis presented here is the first one that handles the Markovian switching assumption.

It should be mentioned that similar multi-model algorithms, incorporating dynamical systems and Markovian model switching have been used for *state estimation* (Caputi, 1995; Dufour & Betrtrand, 1994; Magill, 1965) and *control* (Athans et al., 1977; Dufour & Bertrand, 1993; Narendra, 1994, 1995). This approach is also related to work on time series classification, presented in (Kehagias & Petridis; Petridis & Kehagias, 1996a, 1996b). In addition, there is a connection to predictive hidden Markov models (Kenny, Lennig & Mermelstein, 1990). These approaches are all related to *black-box* models of the dynamical system; on the other hand, the approach presented here makes use of *structured* models.

2 The Parameter Estimation Algorithm

Consider a dynamical system described by the following equations

$$x_s = f(x_{s-1}, u_s; z_s) \qquad y_s = g(x_{s-1}, u_s; z_s) + w_s \tag{1}$$

where s is the time index, x_s is state vector ($x_s \in \Re^N$), y_s is the output ($y_s \in \Re^M$), u_s is the input ($u_s \in \Re^J$), and w_s is a white noise vector ($w_s \in \Re^M$). z_s is a time varying parameter taking values in the set $\Theta = \{\theta_1, \theta_2, ..., \theta_K\}$. z_s varies in a Markovian way, described by the transition probability matrix P: $P_{mk} \doteq Prob(z_s = \theta_k | z_{s-1} = \theta_m)$.

At every time step we want to identify the member of Θ which *best* matches the observed sequence y_1, y_2, ..., y_t, ... An algorithm is now presented to solve this problem. Consider a set of *models*, which evolve according to the following equations ($k = 1, 2, ..., K$):

$$x_s^k = f(x_{s-1}^k, u_s; \theta_k), \qquad y_s^k = g(x_{s-1}^k, u_s; \theta_k). \tag{2}$$

Equation (2), describes the evolution of the system (1), with the parameter fixed at θ_k and without noise input. The estimation algorithm is based on the following idea: if $z_s = \theta_k$ for some time interval, then y_s^k must be close to y_s over the same interval. This is formalized by introducing a (recursively updated) credit function for each value of k. We will work with quantities which evolve at time-steps 1, $L+1$, $2L+1$ ... . Hence we use a new time variable t, where $t = \frac{s-1}{L} + 1$. We adopt the convention of denoting the new quantities by capital letters (while original quantities are displayed by small letters):

$$Z_1 \doteq z_1, \qquad Y_1 \doteq [y_1, y_2, ..., y_L], \qquad Y_1^k \doteq [y_1^k, y_2^k, ..., y_L^k], \qquad k = 1, ..., K \tag{3}$$

$$Z_2 \doteq z_{L+1}, \qquad Y_2 \doteq [y_{L+1}, y_{L+2}, ..., y_{2L}], \qquad Y_2^k \doteq [y_{L+1}^k, y_{L+2}^k, ..., y_{2L}^k], \qquad k = 1, ..., K$$

... .

Of course by taking L=1 we revert to the original variables. Now consider the following quantities π_t^k, $k = 1, 2, ..., K$, $t = 0, 1, ...$, which are updated recursively according to the following formula.

$$\pi_t^k = \frac{\left(\sum_{m=1}^K \pi_{t-1}^m \cdot R_{mk}\right) \cdot e^{-\frac{|Y_t - Y_t^k|^2}{L\sigma^2}}}{\sum_{l=1}^K \left(\sum_{m=1}^K \pi_{t-1}^m \cdot R_{ml}\right) \cdot e^{-\frac{|Y_t - Y_t^l|^2}{L\sigma^2}}}. \tag{4}$$

Here $|\cdot|$ denotes Euclidean norm and matrix R is the L-th power of the matrix P: $R \doteq P^L$; σ is an *error scale parameter*; a high value of σ mollifies the effect of large error $Y_t - Y_t^k$, at the expense of slower updating of π_t^k.

It follows from (4) that $0 \leq \pi_t^k \leq 1$. We propose the following interpretation: when π_t^k is large, parameter $Z_t = \theta_k$ matches well the observed data block Y_t; similarly, when π_t^k is small, $Z_t = \theta_k$ does not match well the data block Y_t. Hence we call π_t^k the *credit functions*. This interpretation is justified by considering the terms appearing in (4). A large π_{t-1}^m value implies that θ_m matched well the previous output data block of the system. A large R_{mk} value indicates that the θ_m to θ_k parameter switch is likely. A large $e^{-\frac{|Y_t - Y_t^k|^2}{L\sigma^2}}$ value (i.e. a small $\frac{|Y_t - Y_t^k|^2}{L\sigma^2}$ value) indicates that parameter θ_k matches well Y_t, the data block observed at time t. These factors are combined through equation(4) to evaluate the credit of parameter value θ_k at the current time step t. Finally, the denominator ensures that the π_t^k credits are scaled so as to lie in the [0,1] interval.

The computation takes account of the entire output history: $e^{-\frac{|Y_t - Y_t^k|^2}{L\sigma^2}}$ in equation(4) introduces an explicit dependence on the last L outputs; π_{t-1}^m, $m = 1, 2, ..., K$, introduces an explicit dependence on the previous L outputs *and* an implicit dependence (through π_{t-2}^m) on the previous L outputs and so on. Regarding π_0^k, the credit assigned to θ_k at time $t = 0$ (before any output has been observed), we can just assume all models to be equally credible: $\pi_0^k = \frac{1}{K}$, $k = 1, 2, ..., K$. Finally, we choose the estimate $\hat{Z}_t$ as follows

$$\hat{Z}_t \doteq arg \max_{\theta_k \in \Theta} \pi_t^k. \tag{5}$$

In other words, at time t, having observed $Y_1, ..., Y_t = y_1, y_2, ... y_{tL}$, we claim that the current value of the parameter is $\hat{Z}_t$, where $\hat{Z}_t$ maximizes π_t^k.

Combining (2), (3), (4) and (5) the proposed estimation algorithm is obtained. This algorithm works well in practice, as has been established by numerical experimentation. In addition to the experimental justification, the algorithm can be justified mathematically: in Section 3 we show that if the parameter switches sufficiently slowly, the length of the data block is sufficiently large and one model provides better output predictions than the remaining ones, then, in intervals between switchings, the algorithm selects the corresponding parameter value.

In the design of this algorithm, we were motivated by the Partition Algorithm (Hilborn & Lainiotis, 1969; Lainiotis, 1971; Sims, Lainioits & Magill, 1969). This algorithm applies to linear systems and is placed in a probabilistic framework. In particular the π_t^k quantities can be interpreted as posterior probabilities: $\pi_t^k = \Pr(Z_t = \theta_k | Y_1, ..., Y_t)$ and eq.(4) as Bayes' rule. However, as

the reader has noticed, our algorithm requires no probabilistic interpretation; its justification is purely *phenomenological*, in that the parameter value best fitting the output is selected. The removal of the probabilistic interpretation enlarges the scope of application: the PA is based on the assumption that posterior probabilities (corresponding to our π_t^k credit functions) are computed exactly. However, this computation can only be carried out under the assumption of linear systems and Gaussian noise. Our algorithm applies to nonlinear systems as well; also knowledge of the statistical characteristics of noise is not necessary. In addition, a number of modifications of eq.(4) are possible, which retain the basic idea of recursively updating credit according to predictive accuracy, but have no formal similarity to Bayes' rule. In short, we believe that the probabilistic interpretation may be illuminating but also limits the versatility of our algorithm and related schemes.

3 Convergence

It is desirable to provide some theoretical justification of our algorithm. The desired result is this: while the parameter value remains fixed (say at θ_n) the credit functions tend to values such that $\pi_t^n > \pi_t^k$, for $n \neq k$. This ensures correct estimation of the parameter.

We prove a somewhat weaker result, which is based on the study of deterministic quantities p_t^k; these are selected in such a way that they approximate the random quantities π_t^k. We then prove that p_t^k have the desired behavior; to the extent that the π_t^k are approximated closely, these also will have the desired behavior that ensures correct estimation.

Suppose, for the time being, that the parameter value is fixed at $Z_t = \theta_n$. Define

$$A_{kn} \doteq \lim_{L\to\infty} \frac{|Y_t - Y_t^k|^2}{L \cdot \sigma^2} = \lim_{L\to\infty} \frac{\sum_{s=(t-1)L+1}^{tL} |y_s - y_s^k|^2}{L \cdot \sigma^2}. \tag{6}$$

Here σ is the error scale parameter defined in the previous section. Also define $\alpha_{kn} = e^{-A_{kn}}$ (note that $0 < \alpha_{kn} < 1$ for all k, n). Consider now the following quantities ($t = 1, 2, ...$ and $k = 1, 2, ..., K$):

$$p_t^k = \frac{\left(\sum_{m=1}^K p_{t-1}^m \cdot R_{mk}\right) \cdot \alpha_{kn}}{\sum_{l=1}^K \left(\sum_{m=1}^K p_{t-1}^m \cdot R_{ml}\right) \cdot \alpha_{ln}}. \tag{7}$$

We will prove that the p_t^k's as given by (7) are convergent; this is the conclusion of the following theorem.

Theorem 1 *Consider the system defined by eq.(7), with α_{kn} defined by eq.(6) for k, n= 1, 2, ..., K. Suppose that for a fixed n ($1 \leq n \leq K$) the following conditions hold*

(A1) *$R > 0$ (i.e. $R_{kl} > 0$ for k, l= 1, 2, ..., K),*
(A2) *there is some $\epsilon > 0$ such that for all k we have $R_{kk} > 1 - \epsilon$ and for $l \neq k$ $R_{kl} < \epsilon$,*
(A3) *for all $k \neq n$ we have $\frac{1+K\epsilon}{1-\epsilon} < \frac{\alpha_{nn}}{\alpha_{kn}}$*

then $\lim_{t\to\infty} p_t^k$ exists for $k = 1, 2, ..., K$ and $\lim_{t\to\infty} p_t^n > \lim_{t\to\infty} p_t^k$ for all $k \neq n$.

Proof : The proof is presented in the Appendix.

Remark 1: The theorem states that, as long as the parameter is fixed to the value θ_n, each p_t^k converges to some limiting value; also, the largest limiting value corresponds to the true parameter value θ_n. Hence, if θ_n stays fixed for a long enough time, then eventually p_t^n will become larger than all p_t^l, for $l \neq n$, which will ensure correct parameter estimation. If **(A3)** holds for every n, and the intervals between parameter switchings are long enough, then convergence holds for every time interval and every parameter value. The theorem pertains to p_t^k, as computed by equation(7).

Remark 2: We also require (condition **(A1)**) that R is positive, i.e. $R > 0$. This means that no parameter switch is completely unlikely. In fact it would suffice to assume that R is primitive, i.e. there is some d such that $R^d > 0$; we take $d = 1$ to simplify the analysis, but we would reach the same conclusions for any $d \geq 1$. In addition we require (condition **(A2)**) that there is some $\epsilon > 0$ such that for all l we have $R_{ll} > 1 - \epsilon$ and $R_{lk} < \epsilon$ for $k \neq l$. This is a condition for "slow switching". If parameter switching took place at a fast rate, there would not be enough time for the p_t^k to converge to some limit between model switchings. Slow switching is guaranteed if R_{ll} is significantly larger than R_{lk}.

Remark 3: Finally, it is required that for a fixed n and all $l \neq n$ the inequality $\frac{1+K\epsilon}{1-\epsilon} < \frac{\alpha_{nn}}{\alpha_{ln}}$ holds. This is a "goodness-of-model" condition. Recall that α_{ln} expresses the exponentiated error of parameter θ_l, when parameter θ_n is active. Convergence to the true parameter θ_n requires at least that θ_n generates the smallest average error, i.e. $1 < \frac{\alpha_{nn}}{\alpha_{ln}}$, for all $l \neq n$. The condition we impose here is somewhat stronger, since $1 < \frac{1+K\epsilon}{1-\epsilon}$.

Remark 4: Suppose now that for some time interval the active parameter value is θ_0 (outside of $\Theta = \{\theta_1, \theta_2, ..., \theta_K\}$). However, there is some n such that for all $l \neq n$ we have $\frac{1+K\epsilon}{1-\epsilon} < \frac{\alpha_{n0}}{\alpha_{l0}}$. Then, as will be explained in the Appendix, the result of the theorem still holds true. In other words, among all parameter

values in the search set, the algorithm picks the one that best fits the output.

Remark 5: In place of the exponential function of the error $e^{-\frac{|Y_t - Y_t^k|^2}{L\sigma^2}}$, a more general function $f(|Y_t - Y_t^k|)$ can be used; all that is required is that $f(\cdot)$ is a decreasing function.

Remark 6: Now, let us consider the relationship between p_t^k and π_t^k. First, consider n and L fixed and introduce the following assumption

(B) For $k = 1, 2, ..., K$ we have $A_{kn} \simeq \frac{|Y_t - Y_t^k|^2}{L\sigma^2}$.

This assumption will be true if the original system has ergodic behavior and L is large. Second, suppose that the convergence of p_t^k (which is guaranteed by the theorem) takes place (up to desirable accuracy) within some time, say T_c. Finally, suppose that the system operates with fixed parameter θ_n, for some time, say T_s. If $L << T_c << T_s$, then it is reasonable that π_t^k, as given by (4), is approximated by p_t^k, as given by (7). An additional discussion of this issue is provided in the conclusions section.

4 Experiments

We test our algorithm on the task of rotor resistance estimation of an AC induction motor. The AC induction motor is described (in discrete time) by the following nonlinear equations.

$$X(s) = \tau \cdot A^{-1} \cdot \{-B \cdot X(s-1) + U(s)\} \tag{8}$$

$$w(s) = w(s-1) + \tau \cdot \left\{ \frac{3\rho^2}{8J} L_0 \left(i_{qs}(s-1) i_{dr}(s-1) - i_{ds}(s-1) i_{qr}(s-1) \right) - \frac{\rho}{2J} T_L(s) \right\}; \tag{9}$$

here $X(s) = [i_{qs}(s), i_{ds}(s), i_{qr}(s), i_{dr}(s)]$, $U(s) = [V_{qs}(s), V_{ds}(s), 0, 0]$ and

$$A = \begin{bmatrix} L_s & 0 & L_0 & 0 \\ 0 & L_s & 0 & L_0 \\ L_0 & 0 & -L_r & 0 \\ 0 & L_0 & 0 & L_r \end{bmatrix} \qquad B = \begin{bmatrix} r_s & 0 & 0 & 0 \\ 0 & r_s & 0 & 0 \\ 0 & -w(s-1)L_0 & r_r & -w(s-1)L_r \\ w(s-1)L_0 & 0 & w(s-1)L_r & r_r \end{bmatrix}. \tag{10}$$

The stator currents are $i_{qs}(s)$, $i_{ds}(s)$, the rotor currents are $i_{qr}(s)$, $i_{dr}(s)$, the angular velocity is $w(s)$, the stator voltages are $V_{ds}(t)$, $V_{qs}(s)$ and the load

torque is $T_L(s)$. τ is the integration step; r_s, r_r are the stator and rotor resistances, L_s, L_r, L_0 are the stator, rotor and mutual inductances respectively; J is the moment of inertia and ρ is the number of pole pairs. The state vector is $[X(s), w(s)]$ and the input vector is $[U(s), T_L(s)]$. All the parameters can be measured, except for r_r, which depends on operating conditions. But r_r is necessary for the efficient and economic control of angular velocity. This is a standard on-line parameter estimation problem. We measure $i_{qs}(s)$, $i_{ds}(s)$; the vector $[i_{qs}(s), i_{ds}(s)]$ (s=1, 2, ...) play the role of the time series y_s (s=1, 2, ...). This may have been produced by any model corresponding to a specific value r_r; that is, r_r plays the role of the model parameter Z_t.

The AC induction motor is simulated, mixing the stator current output with additive noise at various levels, described by the Signal - to - Noise Ratio (SNR). Each simulation is run for 10000 time steps (τ = 0.5 ms). A three phase AC voltage of 220 V RMS value and a torque T_L=1.5 N·m are used as input. The real motor parameters are: r_s=11.58 Ω, L_s =0.071 H, L_r=0.072 H, L_0=0.069 H, J=0.089 kg·m^2 , B=0 Nt·sec/m, ρ=2. The effect of r_r variation is simulated by using ten r_r values: from time t=0.0 to 0.5 s the real rotor resistance r_r= 4.9 Ω, from 0.5 to 1.0 s r_r=5.9 Ω and so on until the value 13.9 Ω. Ten models are used (K=10), with r_r values of 5, 6,..., 14 Ω. For real r_r value 4.9 Ω, the best estimate is 5 Ω; similarly for r_r=5.9, 6.9, ... The time step used in equations(8), (9) is τ=0.5 ms. Several sampling times are used for the measurement of the stator current, namely $\bar{\tau}$= 0.5, 1, 2, 3 ms. Equation (4) is used for the computation of π_t^k. The value L=10 is used; σ is computed experimentally from the prediction errors computed by (2). Regarding the switch matrix R, it is taken equal to P^L, where P is a band matrix, with all diagonal elements equal (P_{11}= P_{22}= ... = P_{KK}) and close to one; also $P_{k,k-1}$= $P_{k,k+1}$= 1−P_{kk}/2. All other elements were set equal to zero. This represents an assumption that, as r_r varies, it can only move to neighboring values. Several P_{kk} values were used.

The results for various combinations of noise level, $\bar{\tau}$, and R_{kk} are presented in Figures 1 to 5. The results are expressed by an estimation figure of merit $c = N_1/N_0$. Here N_0 is the total number of time steps and N_1 the number of time steps where the best parameter value was picked. In each figure we list c for $\bar{\tau}$=0.5, 1, 2, 3 ms and for various noise levels (starting with the noise free case and advancing through SNR=20.00, 10.00, 6.66, 5.00, 4.00, 3.33, 2.50). What differs in each figure is the values of σ and R_{kk}.

5 Conclusions

We have presented an on-line, multi-model algorithm for the estimation of the switching parameter of a dynamical system. The algorithm is based on

the comparison of output behaviors of the true system and those of a set of models, each tuned to a particular value of the parameter. Our algorithm is a generalized version of the Partition Algorithm; in particular, minimal probabilistic assumptions are required, the new algorithm applies to nonlinear systems as well, and the case of a switching parameter is handled by the introduction of a switching mechanism. While a probabilistic interpretation of our results is possible, it is not necessary and in fact it may be limiting possible extensions of the applicability of the algorithm and variations of the form of update eq.(4). We provide a convergence analysis of the algorithm, which is based on the approximation of the probabilistic credit functions π_t^k by the deterministic quantities p_t^k. The latter are proved to converge (between parameter switchings) to values which ensure selection of the parameter value that best approximates the observed system behavior. Numerical experiments show that the algorithm estimates the parameter value with very good accuracy.

In the future, we want to provide a more detailed analysis of the connection between π_t^k and p_t^k, with the final goal of establishing convergence of π_t^k to one for the best model. We are currently examining this problem; let us present some introductory remarks. A method is required to establish convergence either of π_t^k (in some stochastic sense) or of the expected value $E(\pi_t^k)$. Taking expectations in both sides of equation (4) is a possible first move in writing an equation that expresses expected π_t^k in terms of expected π_{t-1}^k and expected $e^{-\frac{Y_t - Y_t^k}{L\sigma^2}}$ (i.e. similar to eq.(7)). However, a way must be found to replace the expectation of a product with a product of expectation. Another possibility is to use Ljung's and Kushner's methods and establish convergence of the stochastic difference equation (4) by studying the properties of a deterministic differential equation. However, this approach requires rewriting eq.(4) in the form: $\pi_t^k = \pi_{t-1}^k + \epsilon \cdot f(\pi_{t-1}^1, Y_t - Y_t^1$,..., $\pi_{t-1}^K, Y_t - Y_t^K)$, or finding an equation of such form that approximates (4).

In addition, we want to extend our algorithm so that it is applicable to problems with high dimensional parameter space. In this case, the quantization approach used in Section 4 is not practical, due to the large number of models that must be employed. What is required is to replace quatization by a more sophisticated method of searching the parameter space, by a "divide-and-conquer" approach, e.g. using a genetic algorithm (which employs the credit functions π_t^k to create successive generations of models), a simulated annealing or a Monte Carlo Markov Chain scheme, such as described in (Gilks et al, 1996). However, such methods are computationally intensive and their on-line application may present difficulties.

A Appendix: Proof of the Convergence Theorem

Here we present the proof of the convergence theorem. To prove Theorem 1, we will work with auxiliary quantities q_t^k, defined as follows, for $k = 1, 2, ..., K$.

$$q_t^k = \left(\sum_{m=1}^{K} q_{t-1}^m \cdot R_{mk} \right) \cdot \alpha_{kn} \tag{A.1}$$

Comparing (7) and (A.1), se see that the q_t^k's are simply the unscaled versions of the p_t^k's. This is actually proved in the following Lemma.

Lemma 2 *For p_t^k as given by (7) and q_t^k as given by (A.1), define $A_t = \sum_{m=1}^{K} q_t^m$. Suppose that for $k = 1, ..., K$ p_1^k and q_1^k are chosen so that $\frac{q_1^1}{p_1^1} = \frac{q_1^2}{p_1^2} = ... = \frac{q_1^K}{p_1^K}$. Then, for $m = 1, 2, ..., K$ and for $t = 1, 2, ...$ we have $q_t^m = A_t \cdot p_t^m$.*

Proof By induction. For $t = 1$

$$\frac{q_1^1}{p_1^1} = \frac{q_1^2}{p_1^2} = ... = \frac{q_1^K}{p_1^K} = \frac{q_1^1 + ... + q_1^K}{p_1^1 + ... + p_1^K} = \frac{q_1^1 + ... + q_1^K}{1} = A_1. \tag{A.2}$$

Now suppose that the proposition holds for $t = r$. Then $p_r^m = \frac{1}{A_r} q_r^m$ for $m = 1, 2, ..., K$ and

$$p_{r+1}^k = \frac{\frac{1}{A_r} \left(\sum_{m=1}^{K} q_r^m \cdot R_{mk} \right) \cdot \alpha_{kn}}{\frac{1}{A_r} \sum_{l=1}^{K} \left(\sum_{m=1}^{K} q_r^m \cdot R_{ml} \right) \cdot \alpha_{ln}} = \frac{q_{r+1}^m}{\sum_{l=1}^{K} q_{r+1}^l} = \frac{q_{r+1}^m}{A_{r+1}}. \tag{A.3}$$

and the proof is complete. ∎

Now, to prove convergence we work for a while with the auxiliary quantities q_t^k rather than the p_t^k. Define $q_t = [q_t^1, q_t^2, ..., q_t^K]$ and

$$A = \begin{bmatrix} \alpha_{1n} & 0 & ... & 0 \\ 0 & \alpha_{2n} & ... & ... \\ ... & ... & ... & ... \\ 0 & ... & ... & \alpha_{Kn} \end{bmatrix} \qquad R = \begin{bmatrix} R_{11} & R_{12} & ... & R_{1K} \\ R_{21} & R_{22} & ... & ... \\ ... & ... & ... & ... \\ R_{K1} & ... & ... & R_{KK} \end{bmatrix} \qquad Q = RA; \tag{A.4}$$

then (A.1) can be rewritten as

$$q_t = q_{t-1} RA = q_{t-1} Q \tag{A.5}$$

Q is a positive matrix. Since $\alpha_{kn} < 1$ for $k = 1, ..., K$ and $\sum_{l=1}^{K} R_{kl} = 1$, we have $\sum_{l=1}^{K} Q_{kl} < 1$. The following theorem holds for the powers of Q.

Theorem 3 $Q^t = \lambda^t \cdot w' \cdot v' + O(t^M |\mu|^t)$.

Proof (Seneta, 1987), p.7.

Here λ is the (positive) maximum modulus eigenvalue of Q (guaranteed to exist by the Perron-Frobenius theorem, (Seneta, 1987, p.1); w, v are the associated (strictly positive) right and left eigenvectors, i.e. $wQ = \lambda w$, $Qv = \lambda v$ and $wv = 1$; μ is the second maximum modulus eigenvalue, with multiplicity M. We have the following Lemma.

Lemma 4 $|\lambda| = \lambda < 1$.

Proof That $|\lambda| = \lambda$ follows from the Perron-Frobenius theorem. Now define $c_m \doteq \sum_{n=1}^{K} Q_{mn} < 1$. Suppose $\lambda \geq 1$. We have

$$wQ = \lambda w \Rightarrow \sum_m w_m Q_{mn} = \lambda w_n \Rightarrow \sum_n \sum_m w_m Q_{mn} = \lambda \sum_n w_n \Rightarrow$$

$$\sum_m \sum_n w_m Q_{mn} = \lambda \sum_n w_n \Rightarrow \sum_m w_m \sum_n Q_{mn} = \lambda \sum_n w_n \Rightarrow \sum_m w_m c_m = \lambda \sum_m w_m \Rightarrow$$

$$\sum_m w_m (c_m - \lambda) = 0. \tag{A.6}$$

This leads to a contradiction, because we know that $\lambda \geq 1$ and, for all m, $w_m > 0$ and $c_m < 1$; so we have a sum of strictly negative numbers that equals less than zero. The proof is complete. ∎

Now we will prove the following Lemma.

Lemma 5 *Define* $\gamma_{ml} = \frac{v_m}{v_l}$. *For* $t = 1, 2, ...$ *and* $m, l = 1, 2, ..., K$ *we have*

$$\frac{q_t^m}{q_t^l} = \frac{[Q^t]_{nm}}{[Q^t]_{nl}} \to \gamma_{ml} \quad as \quad t \to \infty. \tag{A.7}$$

Proof From Theorem 3 and Lemma 4 it follows that as $t \to \infty$

$$\frac{1}{\lambda^t} Q^t \to \begin{bmatrix} w_1 \\ ... \\ w_K \end{bmatrix} \cdot \begin{bmatrix} v_1 \ ... \ v_K \end{bmatrix} = \begin{bmatrix} w_1 v_1 & w_1 v_2 & ... & w_1 v_K \\ w_2 v_1 & w_2 v_2 & ... & ... \\ ... & ... & ... & ... \\ w_K v_1 & ... & ... & w_K v_K \end{bmatrix}. \tag{A.8}$$

Then, for $i, j, m = 1, 2, ..., K$

$$\left.\begin{array}{l} \frac{[Q^t]_{im}}{\lambda^t} \to w_i v_m \\ \frac{\lambda^t}{[Q^t]_{jm}} \to \frac{1}{w_j v_m} \end{array}\right\} \Rightarrow \frac{[Q_{im}]^t}{[Q^t]_{jm}} \to \frac{w_i}{w_j} = \beta_{ij} \tag{A.9}$$

Similarly, for $i, j = 1, 2, ..., K$ (and fixed n)

$$\left.\begin{array}{l} \frac{[Q^t]_{ni}}{\lambda^t} \to w_n v_i \\ \frac{\lambda^t}{[Q^t]_{nj}} \to \frac{1}{w_n v_j} \end{array}\right\} \Rightarrow \frac{[Q^t]_{ni}}{[Q^t]_{nj}} \to \frac{v_i}{v_j} = \gamma_{ij} \tag{A.10}$$

Since $q_t = q_{t-1}Q$, $q_{t-1} = q_{t-2}Q$ etc., finally $q_t = q_0 Q^t$. Then we have for $m, l = 1, 2, ..., K$

$$q_t^m = q_0^1 [Q^t]_{1m} + q_0^2 [Q^t]_{2m} + ... + q_0^K [Q^t]_{Km}. \tag{A.11}$$

$$\frac{1}{q_t^l} = \frac{1}{q_0^1 [Q^t]_{1l} + q_0^2 [Q^t]_{2l} + ... + q_0^K [Q^t]_{Kl}}. \tag{A.12}$$

Taking $i = 1, 2, ..., K$ and $j = n$ in (A.9) one obtains

$$\frac{[Q^t]_{1m}}{[Q^t]_{nm}} \to \beta_{1n}, \qquad \frac{[Q^t]_{2m}}{[Q^t]_{nm}} \to \beta_{2n}, \qquad ..., \qquad \frac{[Q^t]_{Km}}{[Q^t]_{nm}} \to \beta_{Kn}; \tag{A.13}$$

then applying to (A.11) one obtains

$$\frac{q_t^m}{[Q^t]_{nm}} \to \left(q_0^1 \beta_{1n} + q_0^2 \beta_{2n} + ... + q_0^K \beta_{Kn} \right). \tag{A.14}$$

Similarly, in (A.9) take $m = l$, $i = 1, 2, ..., K$ and $j = n$ to obtain

$$\frac{[Q^t]_{1l}}{[Q^t]_{nl}} \to \beta_{1n}, \qquad \frac{[Q^t]_{2l}}{[Q^t]_{nl}} \to \beta_{2n}, \qquad ..., \qquad \frac{[Q^t]_{Kl}}{[Q^t]_{nl}} \to \beta_{Kn}; \tag{A.15}$$

combining (A.15) with (A.12) one obtains

$$\frac{[Q^t]_{nl}}{q_t^l} \to \frac{1}{(q_0^1 \beta_{1n} + q_0^2 \beta_{2n} + ... + q_0^K \beta_{Kn})}. \tag{A.16}$$

Multiply (A.14) and (A.16) to obtain

$$\frac{q_t^m}{[Q^t]_{nm}} \cdot \frac{[Q^t]_{nl}}{q_t^l} \to 1. \tag{A.17}$$

Also, by (A.10) with $i = l$ and $j = m$, one obtains

$$\frac{[Q^t]_{nl}}{[Q^t]_{nm}} \to \gamma_{lm} \tag{A.18}$$

Combining (A.17) and (A.18) one finally obtains

$$\frac{q_t^m}{q_t^l} \cdot \gamma_{lm} \to 1 \Rightarrow \frac{q_t^m}{q_t^l} \to \frac{1}{\gamma_{lm}} = \frac{1}{\frac{v_l}{v_m}} = \frac{v_m}{v_l} = \gamma_{ml} \tag{A.19}$$

and the proof is complete. ∎

Now we prove the convergence theorem.

Proof of Theorem 1 From Lemma 5 we know that for $m, l = 1, 2, ..., K$ we have $\frac{q_t^m}{q_t^l} \to \gamma_{ml}$. From Lemma 1 we know that for $m, l = 1, 2, ..., K$ and $t = 1, 2, ...$ we have $q_t^m = A_t \cdot p_t^m$ and $q_t^l = A_t \cdot p_t^l$. Hence

$$\lim_{t\to\infty} \frac{q_t^m}{q_t^l} = \lim_{t\to\infty} \frac{A_t \cdot p_t^m}{A_t \cdot p_t^l} = \lim_{t\to\infty} \frac{p_t^m}{p_t^l} = \gamma_{ml} \tag{A.20}$$

In other words

$$\left.\begin{array}{l} \frac{p_t^1}{p_t^l} \to \gamma_{1l} \\ \frac{p_t^2}{p_t^l} \to \gamma_{2l} \\ \quad ... \\ \frac{p_t^K}{p_t^l} \to \gamma_{Kl} \end{array}\right\} \Rightarrow \frac{\sum_m p_t^m}{p_t^l} \to \sum_m \gamma_{ml} \Rightarrow \frac{1}{p_t^l} \to \sum_m \gamma_{ml} \Rightarrow p_t^l \to \frac{1}{\sum_m \gamma_{ml}} = p^l. \tag{A.21}$$

It is easy to show that

$$\frac{p_t^l}{p_t^n} = \frac{\sum_m p_{t-1}^m \cdot R_{ml}}{\sum_m p_{t-1}^m \cdot R_{mn}} \cdot \frac{\alpha_{ln}}{\alpha_{nn}}. \tag{A.22}$$

Then, taking the limit as $t \to \infty$ one obtains

$$\frac{p^l}{p^n} = \frac{\sum_m p^m \cdot R_{ml}}{\sum_m p^m \cdot R_{mn}} \cdot \frac{\alpha_{ln}}{\alpha_{nn}}. \tag{A.23}$$

Now suppose that the conclusion of the theorem is false. In that case the maximum of p^k occurs for some $m \neq n$. In other words for some $m \neq n$ and

for all $k \neq m$, we have $p^m \geq p^k$. In particular

$$\frac{p^m}{p^n} = \frac{\sum_k p^k \cdot R_{km}}{\sum_k p^k \cdot R_{kn}} \cdot \frac{\alpha_{mn}}{\alpha_{nn}} \leq \frac{p^m \cdot \sum_k R_{km}}{p^n \cdot R_{nn}} \cdot \frac{\alpha_{mn}}{\alpha_{nn}} \leq \frac{p^m}{p^n} \cdot \frac{1+K\epsilon}{1-\epsilon} \cdot \frac{\alpha_{mn}}{\alpha_{nn}} \Rightarrow$$

$$1 \leq \frac{1+K\epsilon}{1-\epsilon} \cdot \frac{\alpha_{mn}}{\alpha_{nn}} \Rightarrow \frac{\alpha_{nn}}{\alpha_{mn}} \leq \frac{1+K\epsilon}{1-\epsilon} \tag{A.24}$$

which contradicts **(A3)**. Hence the proof is complete. ∎

Remark It has been assumed so far that θ_n is a member of Θ, the search set. However, suppose now that the actual parameter value is θ_0, *outside* Θ. Further suppose that the conditions **(A1), (A2), (A3)** still hold. In particular, **(A3)** now becomes: " for all $l \neq n$ we have $1 < \frac{1+K\epsilon}{1-\epsilon} < \frac{\alpha_{n0}}{\alpha_{l0}}$." This means that the n-th model (in the set Θ) matches the output behavior of the true system better than any other model in Θ. It will be observed that the proof still goes through without any modifications and the conclusion of the theorem still holds: $\lim_{t\to\infty} p_t^m$ exists for $m = 1, 2, ..., K$ and $\lim_{t\to\infty} p_t^n > \lim_{t\to\infty} p_t^l$ for all $l \neq n$. Hence the parameter is estimated correctly, in the sense that the value selected approximates the output behavior of the true system better than any other parameter value in the search set Θ.

References

[1] Anderson, B.D.O. and Moore, J.B. (1979). *Optimal Filtering.* Prentice-Hall, Englewood Cliffs, 1979.

[2] Athans, M. et al. (1977). "The stochastic control of the F-8C aircraft using a multiple model adaptive control method - part I: equilibrium flight", *IEEE Trans on Automatic Control*, vol.22, pp.768-780.

[3] Caputi, M.J. (1995). "A necessary condition for effective performance of the multiple model adaptive estimator", *IEEE Trans.on Aerospace and Electronic Systems*, Vol.31, pp.1132-1138.

[4] Dufour, F. and Bertrand, P. (1994). "The filtering problem for continuous-time linear systems with Markovian switching coefficients", *System and Control Letters*, vol.23, pp.453-461.

[5] Dufour, F. and Bertrand, P. (1993). "Stabilizing control for hybrid models", *IEEE Trans. on Automatic Control*, vol.39, pp.2354-2357.

[6] Gilks, W.R. , Richardson, S. and Spiegelhalter, D.J. (1996). *Markov Chain Monte Carlo in Practice*, Chapman and Hall, London.

[7] Hilborn, C.G. and Lainiotis, D.G. (1969). "Unupervised learning minimum risk pattern classification fro dependent hypotheses and dependent measurements", *IEEE Trans. on Systems Sci. and Cybernetcis*, vol.5, pp.109-115.

[8] Kehagias, Ath. (1991). "Convergence properties of the Lainiotis partition algorithm", *Control and Computers*, vol.19, pp.1-6.

[9] Kehagias, Ath. and Petridis, Vas. "Predictive Modular Neural Networks for Time Series Classification", *Neural Networks*, vol.10, pp.31-49.

[10] Kenny, P., Lennig, M. and Mermelstein, P. (1990). "A linear predictive HMM for vector-valued observations with applications to speech recognition", *IEEE Trans. on Acoustics, Speech and Signal Proc.*, vol.38, pp.220-225.

[11] Lainiotis, D.G. (1971). "Optimal adaptive estimation: structure and parameter adaptation", *IEEE Trans. on Automatic Control*, vol.16, pp. 160-170.

[12] Lainiotis, D.G. and Plataniotis, K.N. (1994). "Adaptive Dynamic Neural Network Estimation", in *Proceedings of Int. Joint Conference on Neural Networks 1994*, vol.6, pp. 4736-4745.

[13] Ljung, L. (1987). *System Identification: Theory for the User*, Prentice Hall.

[14] Magill, D.T. (1965). "Optimal adaptive estimation of sampled stochastic processes", *IEEE Trans. on Automatic Control*, Vol.10, pp.434-439.

[15] Narendra, K. (1995). "Adaptation and learning using multiple models, switching and tuning", *IEEE Control Systems*, pp.37-51.

[16] Narendra, K. and Balakhrishnan, J. (1994). "Improving transient response of adaptive control systems using multiple models and switching", *IEEE Trans. on Automatic Control*, Vol.39, pp.1861-1866.

[17] Petridis, V. (1981). "A method for bearings-only verlocity and position estimation", *IEEE Trans. on Automatic Control*, vol.26, pp.488-493.

[18] Petridis, V. and Kehagias, Ath. (1996a). "A Recurrent Network Implementation of Time Series Classification", *Neural Computation*, vol.8, pp.357-372.

[19] Petridis, V. and Kehagias, Ath. (1996b). "Modular Neural Networks for MAP Classification of Time Series and the Partition Algorithm", *IEEE Trans. on Neural Networks*, vol.7, pp.73-86.

[20] E. Seneta (1987). *Non-Negative Matrices*, 1987, Wiley, New York.

[21] Sims, F.L., Lainiotis, D.G. and Magill, D.T. (1969). "Recursive algorithm for the calculation of the adaptive Kalman filter coefficients", *IEEE Trans. on Automatic Control*, vol.14, pp.215-218.

[22] Tugnait, J.K. (1980). "Convergence analysis of partitioned adaptive estimators under continuous parameter uncertainty", *IEEE Trans. on Automatic Control*, vol.25, pp.569-573.

FIGURE CAPTIONS

FIGURE 1 Classification figure of merit c for σ =0.1, P_{kk} =0.99, L=10.
FIGURE 2 Classification figure of merit c for σ =0.005, P_{kk} =0.99, L=10.
FIGURE 3 Classification figure of merit c for σ =0.01, P_{kk} =0.99, L=10.
FIGURE 4 Classification figure of merit c for σ =0.01, P_{kk} =0.995, L=10.
FIGURE 5 Classification figure of merit c for σ =0.01, P_{kk} =0.98, L=10.

Figure 1

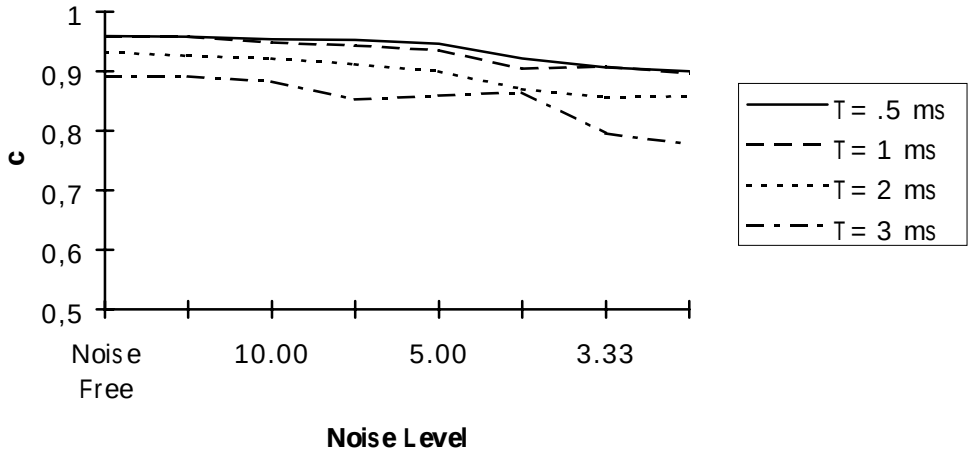

Figure 1: Classification figure of merit c for sigma = 0.1, P_{kk} = 0.99, L = 10.

Figure 2

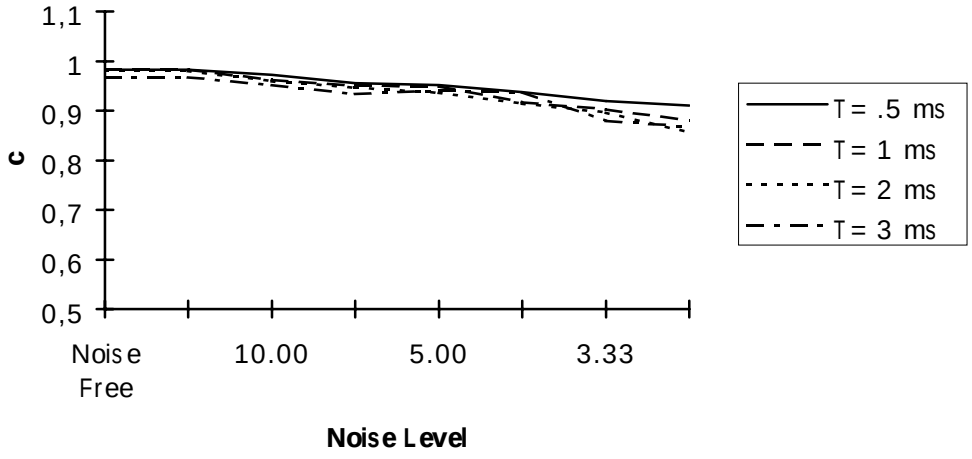

Figure 2: Classification figure of merit c for sigma = 0.005, P_{kk} = 0.99, L = 10.

Figure 3

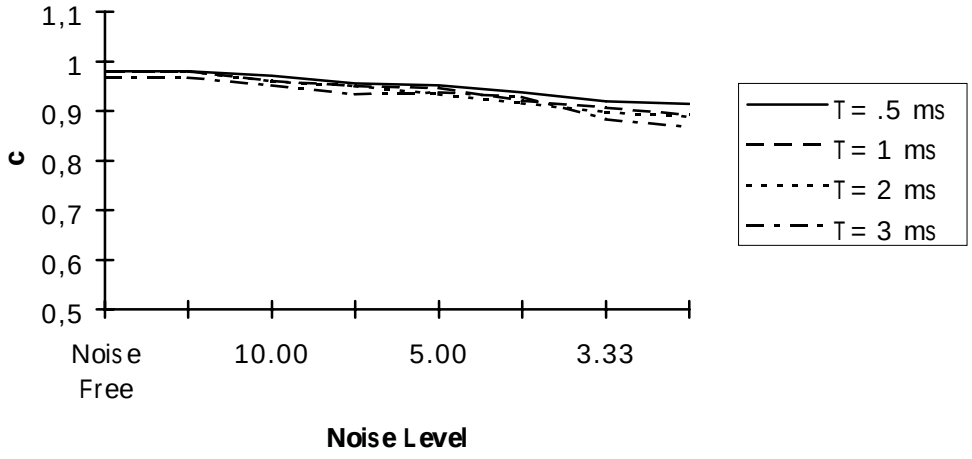

Figure 3: Classification figure of merit c for sigma = 0.01, P_{kk} = 0.99, L = 10.

Figure 4

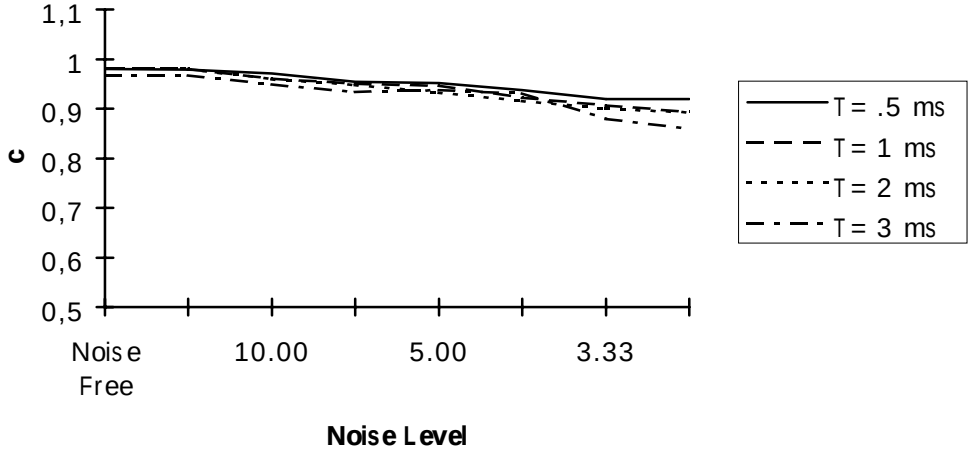

Figure 4: Classification figure of merit c for sigma = 0.01, P_{kk} = 0.995, L = 10.

Figure 5

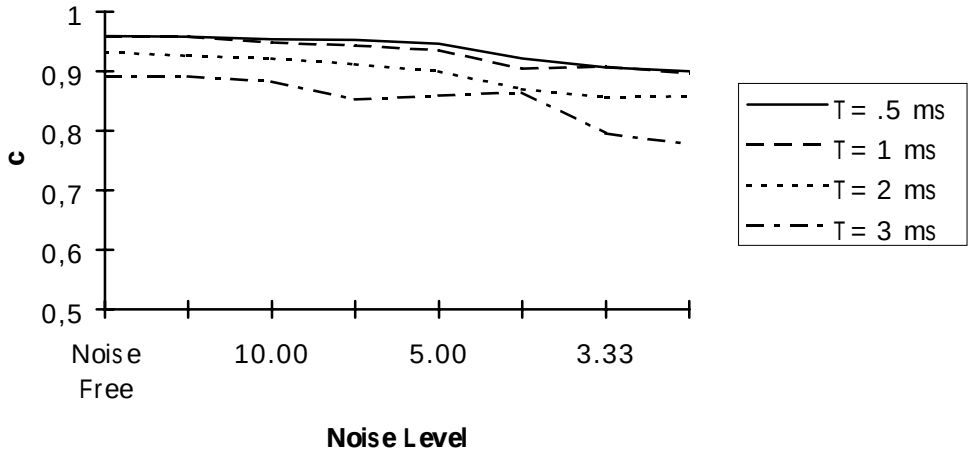

Figure 5: Classification figure of merit c for sigma = 0.01, P_{kk} = 0.98, L = 10.