Ath. Kehagias.
"Convergence Properties of the Lainiotis Partition Algorithm".

This paper has appeared in the journal:
Control and Computers, Vol. 19, pp. 1-6, 1991.

CONVERGENCE PROPERTIES OF THE LAINIOTIS PARTITION ALGORITHM

Athanasios Kehagias
Division of Applied Mathematics
Brown University
Providence, RI 02912, USA

May 26 1990

This paper appeared in *Control and Computers*, Vol. 19, No. 1, pp. 1-6, 1991.

Abstract

We consider the convergence of the Lainiotis Partition Algorithm for system identification. This algorithm computes a MAP model estimate of the system. It is proven that under certain weak conditions the algorithm converges to the "truest" (in a mean square error sense) model. In the course of the discussion, we consider certain factors that determine the convergence behavior of the algorithm. To illustrate the convergence concepts, we apply the algorithm to the identification of the parameters of a d.c. motor.

Keywords: Identification, Parameter Estimation, Bayes' Methods, d.c. motor.

1 Introduction

A popular system identification algorithm is Lainiotis' partition algorithm [1-4], which is characterized by good computational properties and tolerance to imperfect knowledge of the system's noise statistics. The algorithm has been used for system identification of aircraft, d.c. motors [5], seismic phenomena [6] and so forth.

The algorithm starts with a sequence of observations of a physical system and a finite set of possible linear models to explain these observations. The selection of the truest, i.e., the most likely, model is performed by computing the probability of every model in the set, conditional on the observations of the system. The model with highest probability is selected as the truest model.

This procedure is called an identification *epoch*. It should be noted that this is a Bayesian Maximum A Posteriori probability (MAP) approach. A geometric interpretation of the identification procedure is the following: each of the candidate models is considered as a point in an abstract space (*parameter space*). One identification epoch corresponds to finding the most likely point in a region of the parameter space. We can run the algorithm for several identification epochs, successively refining the set of possible models and obtaining truer models in later epochs. Geometrically, we are picking the most likely point from a sequence of progressively smaller regions in the parameter space. Hopefully, eventually the region shrinks to a point that is globally most likely in space.

An interesting question is: When does the algorithm converge? Given a finite set of models and an unlimited amount of observations, will one of the models have consistently highest conditional probability? Furthermore, will we keep getting a consistently most likely model in successive identification epochs, when the model set gets more and more refined?

Lainiotis and Sengbush [7] were the first to observe via simulation the fact that the algorithm does indeed converge to the model closest to the true system. Similar *empirical* analyses can be found in Magill [8], Hilborn and Lainiotis [9,10], and Saridis [11]. However, little theoretical work has been done to date on the convergence question.

The basic object of interest in the convergence analysis is the ratio of conditional probabilities of two models in the model set (e.g., equation (3)). For example, Aoki in [12] points out that such ratios form a martingale. However, he sets forth a general analysis of Bayes procedures, without specific convergence conditions for the partition algorithm (in fact, his book predates the algorithm).

Moore et al. derive convergence conditions for the partition algorithm in [13-15]. Their analysis is based on a strong assumption: that the observations come from an ergodic physical system. Stochastic dynamical systems that exhibit oscillatory or transient behavior (for example, the d.c. motor of Section 4) are not ergodic. Also, the type of convergence (i.p, a.s.?) is not clarified.

Tugnait also considers the convergence of the partition algorithm in [16]. He proves powerful results without assuming ergodicity, but he assumes another condition that is unlikely to be satisfied in practice, namely, that the error covariance matrix is uniformly continuous as a function of the model parameters. In fact, to guarantee this condition, Tugnait assumes, in essence, that the full state vector is observed. This is a very restrictive condition. The d.c. motor of Section 4 does not satisfy it; neither do many other systems of practical interest.

Here we prove two convergence theorems, using a slightly different point of view. The first theorem guarantees convergence in the mean and the second convergence with probability 1. Both theorems hold under very weak assumptions,that are likely to be satisfied for every system of practical interest. Like other authors in the past, we base our analysis on an examination of the ratios of conditional probabilites. However, we do *not* assume the actual system to be in the model set, nor that the system can be necessarily described by any linear model. Instead we take a purely phenomenological approach and consider the truest model to be the one that best fits the observations. What is proven is, essentially, that if a model consistently fits the observations with least mean square error, it will be selected as the best model by the partition algorithm. Depending on the conditions the model satisfies, the convergence will be in the mean, or with probability 1 (w.p. 1). The conditions required for either theorem are very weak and likely to be satisfied by any system of practical interest.

The convergence analysis also helps us in understanding what happens when the algorithm fails to discriminate between models. In essence, when two models perform equally well in fitting the observations, the algorithm will fail to select one over the other. This *commonsense* statement is made more precise in the course of the proof of the convergence theorems.

The paper is organized as follows: in Section 2 we present an outline of the algorithm and discuss its operation; we also identify certain quantities that are crucial to convergence. In Section 3 the convergence theorems are proven and the significance of the convergence conditions is discussed. In Section 4 the algorithm is used to identify the parameters of a *real physical system* (a d.c. motor). Through successive applications of the partition algorithm, we obtain progressively more refined estimates of the motor parameters. Eventually the algorithm reaches its discrimination limit and further iterations yield no improvement of the estimates. This agrees well with the theoretical analysis of the previous section: we have reached a point where the conditions of convergence are no longer fulfilled. We conclude in section 5 with a summary of our theoretical results and a discussion of their relevance to the practical problem of motor identification.

2 The Identification Problem: Some Preliminaries

The Identification Problem is defined as follows. Given the linear, discrete time, stochastic system

$$x(n) = F(n)x(n-1) + G(n)u(n) \tag{1}$$

$$y(n) = H(n)x(n) + v(n) \tag{2}$$

with $u(n), v(n)$ Gaussian, zero-mean, white noise processes over a probability space $(\Omega, \mathcal{F}, P)$, with statistics:

$$E(u(n)) = 0, \quad E(v(n)) = 0$$

$$E(u(n)u^T(m)) = Q(n)\delta(n-m), \quad E(v(n)v^T(m)) = R(n)\delta(n-m)$$

$$E(u(n)v^T(m)) = 0, \quad m, n = 1, 2, ...$$

(all the processes in (1) and (2) are vectors; we take $y(n)$ to be $M-$dimensional, and the other processes to have appropriate dimensions) we want to *identify* the system. That is, we want to determine the values of certain unknown elements of F, G, H, Q, R, on the basis of observations

$$Y(n) = [y(1), y(2), ..., y(n)], \quad n = 1, 2, ...$$

Remark 1. Another (perhaps more realistic) description of the problem is: Given a sequence of observations $y(1), y(2), ..., y(n)$ (which may come from a linear or

nonlinear system), find a system of the form (1),(2) that reproduces $y(1), ..., y(n)$ as closely as possible.

The Lainiotis partition algorithm [17] solves the identification problem in the following way: We postulate K possible models $\mathcal{M}_i = (F_i, G_i, H_i, Q_i, R_i)$ $i = 1, 2, ..., K$. Call the set $\{\mathcal{M}_i, \quad i = 1, 2, ..., K\}$ the *model set*. Collect all the N unknown parameters in the vector $\mathbf{a}$ that can take any of the values $\mathbf{a}_i = [a_{i1}, a_{i2}, .., a_{iN}]$, $i = 1, 2, ..., K$. Call this vector the *parameter vector*. It is an element of the N-dimensional Euclidean *parameter space*. In this sense, the model set is a set of points in the parameter space and our task is to find the truest point in this set, or, in other words, to search through the parameter space for the most likely point. We assume that $\mathcal{M}_i$, $i = 1, 2, ..., K$ is uniformly completely controllable (**UCC**) and uniformly completely observable (**UCO**) (as defined by Jazwinski [12]).

The algorithm assigns conditional probabilities on each of the K vectors $\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_K$. We write, for short, $p_{i,n} = Prob(\mathbf{a} = \mathbf{a}_i \mid Y(n))$ $(i = 1, 2, ..., K,$ $n = 1, 2, ...)$. The details of the computation can be found in [10]. We will use here the following update equation:

$$p_{i,n} = \frac{det(P(n \mid \mathbf{a}_i))^{1/2} \cdot exp(-\frac{1}{2} \cdot V(n \mid \mathbf{a}_i)) \cdot p_{i,n-1}}{\sum_{j=1}^{K} det(P(n \mid \mathbf{a}_j))^{1/2} \cdot exp(-\frac{1}{2} \cdot V(n \mid \mathbf{a}_j)) \cdot p_{j,n-1}} \tag{3}$$

where

$$V(n \mid \mathbf{a}_j) = \tilde{y}_j^T(n \mid \mathbf{a}_j) P^{-1}(n \mid \mathbf{a}_j) \tilde{y}_j(n \mid \mathbf{a}_j) \tag{4}$$

and $\tilde{y}(n \mid \mathbf{a}_j)$ is the error term given by

$$\tilde{y}(n \mid \mathbf{a}_j) = y(n) - \hat{y}(n \mid \mathbf{a}_j) \tag{5}$$

Here $\hat{y}(n \mid \mathbf{a}_j)$, is the optimal linear estimate of $y(n)$, computed by a Kalman filter **matched** to the model $\mathcal{M}_j$. Similarly $P(n \mid \mathbf{a}_j)$ is the covariance matrix of $\tilde{y}_j(n \mid \mathbf{a}_j)$:

$$P(n \mid \mathbf{a}_j) = E(\tilde{y}^T(n \mid \mathbf{a}_j) \tilde{y}(n \mid \mathbf{a}_j)) \tag{6}$$

computed by the Kalman filter matched to the model $\mathcal{M}_j$. As more observations $y(n)$, $n = 1, 2, ...$ come in, we update the conditional probabilities (3). At time n, we assume the true model of the system to be the model $\mathcal{M}_i$ such that $p_{in} > p_{jn}$ $\forall j \neq i$. If this i remains the same for all n with substantially higher p_{in} than any p_{jn}, then we have good reason to believe that $\mathcal{M}_i$ is indeed the truest model.

We proceeed to establish some preliminaries that will be useful in the convergence analysis of the next section. From now on, for brevity, we will write $\tilde{y}_j(n)$ instead of $\tilde{y}(n \mid \mathbf{a}_j)$ and $P_j(n)$ instead of $P(n \mid \mathbf{a}_j)$. Also assume $p_{i,0} = \frac{1}{K}$ $\forall i$. Incidentally, note that the probabilities p_{in} are themselves random quantities, dependent on the observations: $y(1), ..., y(n)$. This implies that when we talk about their convergence properties we must specify the type of convergence. In

what follows we will be concerned with *convergence in the mean* (Theorem 1) or convergence w.p. 1 (Theorem 2).

The following lemma is a consequence of the UCC-UCO property and will be used in the proof of Theorems 1 and 2:

LEMMA: Assume $\forall i$ $\mathcal{M}_i$ is UCC-UCO (as defined in [17]). Then $\forall i \in \{1,2,...,K\}$ $\exists \gamma_i$, δ_i s.t.

$$0 < \gamma_i \cdot I \leq P_i \leq \delta_i \cdot I. \tag{7}$$

PROOF: Jazwinski [12]. ■

From (7) it follows that $\forall q \neq 0, \forall i \in \{1,2,...,K\}, \forall n$ integer

$$0 < \gamma_i \|q\|^2 \leq q^T P_i(n) q \leq \delta_i \|q\|^2 \tag{8}$$

$$0 < \delta_i^{-1} \|q\|^2 \leq q^T P_i^{-1}(n) q \leq \gamma_i^{-1} \|q\|^2 \tag{9}$$

We will now define some quantities that bear on the performance of the algorithm and will be used in the proof of the convergence theorems. The model selection will be done on the basis of the observed error $\tilde{y}_j(n)$, $n = 1,2,...$. It is natural to consider as best the model that accumulates less error on the average (we try to minimize $\sum_{m=1}^{\infty} E(\|\tilde{y}_j(m)\|^2)$.

First define

$$\alpha_{ij} = \limsup_{n \to \infty} \frac{\sum_{m=1}^{n} E(\|\tilde{y}_i(m)\|^2)}{\sum_{m=1}^{n} E(\|\tilde{y}_j(m)\|^2)} \tag{10}$$

The α_{ij}'s, exist always and they are a measure of the goodness of a model relative to another. Namely, the smaller α_{ij}, the better is $\mathcal{M}_i$ relative to $\mathcal{M}_j$.

Also define

$$\mathcal{A}_{ij}(\theta, m) = \{\alpha : E\left(exp\left[-\sum_{k=1}^{n} \left(\|\tilde{y}_i(k)\|^2 - \theta \cdot \|\tilde{y}_j(k)\|^2 \right) \right] \right) \leq \alpha^n \quad \forall n \geq m\},$$

$$\hat{\alpha}_{ij}(\theta, m) = \inf \mathcal{A}_{ij}(\theta, m) \tag{11}$$

In a trivial sense, $\hat{\alpha}_{ij}(\theta, m)$ exists always. Even when $\mathcal{A}_{ij}(\theta, m)$ is empty, we have $\hat{\alpha}_{ij}(\theta, m) = \infty$. The more interesting question is: when is $\hat{\alpha}_{ij}(\theta, m) < 1$? It is always true for $\theta = 1$. If it is true $\forall m \geq m_0$ and for large values of θ, then $\mathcal{M}_i$ is a much better model than $\mathcal{M}_j$, in some sense. This is so because, if $\sum_k \|\tilde{y}_i(k)\|^2$ is much greater, on the average, than $\sum_k \|\tilde{y}_j(k)\|^2$, then we also expect $exp\left(-\sum_{k=1}^{n} \left(\|\tilde{y}_i(k)\|^2\right)\right)$ to be much smaller than $exp\left(-\sum_{k=1}^{n} \left(\|\tilde{y}_j(k)\|^2\right)\right)$ in some average sense.

3 Convergence of the Partition Algorithm

Now we will prove two theorems on convergence of the Partition Algorithm.

THEOREM 1: Given a process $y(n)$, $n = 1, 2, ...$ (possibly generated by the system (1),(2)) and K models, $\mathcal{M}_i$, $i = 1, 2, ..., K$, all of them UCC-UCO. Then $\forall i, j$ such that the following conditions hold:
Condition 1(a): $\forall j \in \{1, 2, ..., K\}$, $\exists \beta_j > 0, n_i$ s.t. $\forall n \geq n_i$

$$E(\sum_{m=1}^{\infty} \|\tilde{y}_j(m)\|^2) > \beta_j \cdot n \tag{12}$$

Condition 1(b): $\exists \epsilon_{ij} > 0$ s.t.

$$M \cdot \log(\frac{\gamma_j}{\delta_i}) + \beta_j \cdot (\frac{1}{\delta_j} - \frac{\alpha_{ij} + \epsilon_{ij}}{\gamma_i}) > 0 \tag{13}$$

(where α_{ij} is defined in (10)), we have $\forall Q$ $\exists n_0$ s.t. $\forall n \geq n_0$

$$E\left(\frac{Prob(\mathbf{a} = \mathbf{a}_i \mid Y(n))}{Prob(\mathbf{a} = \mathbf{a}_j \mid Y(n))}\right) > e^Q$$

PROOF: Define

$$\Lambda_{ij,n} = \log \frac{p_{i,n}}{p_{j,n}}$$

Then, applying (3) recursively, we get

$$\Lambda_{ij,n} = \sum_{m=1}^{n} \log det(P_j(m))^{1/2} + \frac{1}{2} \sum_{m=1}^{n} \tilde{y}_j^T(m) P_j(m)^{-1} \tilde{y}_j(m)$$

$$- \sum_{m=1}^{n} \log det(P_i(m))^{1/2} - \frac{1}{2} \sum_{m=1}^{n} \tilde{y}_i^T(m) P_i^{-1} \tilde{y}_i(m) \tag{14}$$

Now, from (14),

$$E(\Lambda_{ij,n}) = \sum_{m=1}^{n} \log det(P_j(m))^{1/2} + \frac{1}{2} \sum_{m=1}^{n} E(\tilde{y}_j^T(m) P_j^{-1} \tilde{y}_j(m))$$

$$- \sum_{m=1}^{n} \log det(P_i(m))^{1/2} - \frac{1}{2} \sum_{m=1}^{n} E(\tilde{y}_i^T(m) P_i^{-1} \tilde{y}_i(m)) \geq$$

$$\sum_{m=1}^{n} \frac{M}{2} \log(\frac{\gamma_j}{\delta_i}) + \frac{1}{2} E\left(\sum_{m=1}^{n} \left(\frac{\|\tilde{y}_j(m)\|^2}{\delta_j} - \frac{\|\tilde{y}_i(m)\|^2}{\gamma_i}\right)\right) \geq$$

(Because of eqs (8), (9), and the limsup equation (10)) the inequality is true $\forall n \geq n_a$, some n_a,

$$\frac{1}{2} \cdot \left[nM \cdot \log(\frac{\gamma_j}{\delta_i}) + \sum_{m=1}^{n} E(\|\tilde{y}_j(m)\|^2) \cdot (\frac{1}{\delta_j} - \frac{\alpha_{ij} + \epsilon_{ij}}{\gamma_i}) \right] \geq$$

$$\frac{n}{2} \cdot \left[M \cdot \log(\frac{\gamma_j}{\delta_i}) + \beta_j \cdot (\frac{1}{\delta_j} - \frac{\alpha_{ij} + \epsilon_{ij}}{\gamma_i}) \right]$$

Since the term in the brackets is positive for some ϵ_{ij} (by (13)), we have that for any Q $\exists n_0$ s.t. $\forall n \geq n_0$

$$E(log\frac{p_{i,n}}{p_{j,n}}) > Q \tag{15}$$

Also by the fact that log is a concave function,

$$\log(E(x)) \geq E(\log(x)) \tag{16}$$

and now (15) and (16) imply that

$$\log(E(\frac{p_{i,n}}{p_{j,n}})) > Q \Rightarrow E(\frac{p_{i,n}}{p_{j,n}}) > e^Q$$

■

Remark 2. Condition 1(a) is a *mean value* statement. It says that the mean value of the cumulative prediction error for every model in the model set is increasing in time at a linear rate. This will certainly be true if there is a small additive white noise component, like $u(n)$, $v(n)$ in (1), (2).

Remark 3. Condition 1(b) has to do with the relative rate α_{ij} at which error accumulates for every model in the model set. It is required that for some model $\mathcal{M}_i$ this rate is very small. In fact, it has to be smaller than a function of the ratio of upper and lower error bounds (γ_i's, δ_i's, that we get from the postulated covariance matrices P_i) and the absolute rate β_i's of error accumulation.

Remark 4. If the above assumptions hold, the theorem guarantees that the mean value of $p_{i,n}/p_{j,n}$ increases without limit as n tends to infinity. This implies that, on the average, $p_{i,n}$ is much greater than $p_{j,n}$. This is a *mean value* statement; essentially it says that, on the average, one of the conditional probabilities is much higher than the others. We may reasonably expect that the actual value of this probability will be much larger than the values of the other probabilities, as well, but this need not be true all the time. It can be that for some i the conditional probability (3) is lower than the rest with high probability; but it is very high on a set of small probability, so it can still have the highest average between all the conditional probabilities.

Now we will strengthen our conditions, to obtain a result about the probability ratios themselves, rather than their mean values. This is done in the next

theorem, where convergence with probability 1 is proven.

THEOREM 2: Given a process $y(n)$, $n = 1, 2, ...$ (possibly generated by the system (1),(2)) and K models, $\mathcal{M}_i$, $i = 1, 2, ..., K$, all of them UCC-UCO. Also *Condition 2:* $\exists \zeta, n_0 \quad s.t.$

$$\hat{\alpha}_{ji}(\delta_j/\gamma_i, n) \cdot (\delta_i/\gamma_j)^{M\delta_j} < \zeta < 1, \qquad \forall n \geq n_0 \tag{17}$$

where $\hat{\alpha}_{ij}$ is defined in (11). Then, w.p. 1

$$\frac{Prob(\mathbf{a} = \mathbf{a}_{j,n} \mid Y(n))}{Prob(\mathbf{a} = \mathbf{a}_{i,n} \mid Y(n))} \to 0$$

PROOF: By the same method as in Theorem 1, we can get

$$\frac{p_{j,n}}{p_{i,n}} \leq (\frac{\delta_i}{\gamma_j})^{nM/2} \cdot exp\left[-\frac{1}{2\delta_j}\sum_{k=1}^{n}\left(\|\tilde{y}_j(k)\|^2 - \frac{\delta_j}{\gamma_i} \cdot \|\tilde{y}_i(k)\|^2\right)\right] \tag{18}$$

Now, by (18) we have that for any $\delta > 0$

$$Prob(\frac{p_{j,n}}{p_{i,n}} > \delta) \leq$$

$$Prob\left((\frac{\delta_i}{\gamma_j})^{nM/2} \cdot exp\left[-\frac{1}{2\delta_j}\sum_{k=1}^{n}\left(\|\tilde{y}_j(k)\|^2 - \frac{\delta_j}{\gamma_i} \cdot \|\tilde{y}_i(k)\|^2\right)\right] > \delta\right) =$$

$$Prob\left(exp\left[\sum_{k=1}^{n}\left(\|\tilde{y}_j(k)\|^2 - \frac{\delta_j}{\gamma_i}\|\tilde{y}_i(k)\|^2\right)\right] \geq \left(\frac{\delta}{(\frac{\delta_i}{\gamma_j})^{nM/2}}\right)^{2\delta_j}\right) \leq$$

(by the Markov inequality - see [8])

$$C(\delta) \cdot (\frac{\delta_i}{\gamma_j})^{M \cdot \delta_j \cdot n} \cdot E\left(exp\left[-\sum_{k=1}^{n}(\|\tilde{y}_j(k)\|^2 - \frac{\delta_j}{\gamma_i}\sum_{k=1}^{n}\|\tilde{y}_i(k)\|^2)\right]\right).$$

Here $C(\delta)$ is a constant that depends only on δ. Now, by the hypothesis of the theorem we see that

$$Prob(\frac{p_{j,n}}{p_{i,n}} > \delta) \leq C(\delta) \cdot \zeta^n \tag{19}$$

with $\zeta < 1$ (by (17)). But then $\sum_{n=1}^{\infty} Prob(\frac{p_{j,n}}{p_{i,n}} > \delta) < \infty$, for any δ, and so, by the Borel-Cantelli Lemma (see [13]) $\exists n_0$ such that $\forall n \geq n_0$

$$Prob(\frac{p_{j,n}}{p_{i,n}} > \delta) = 0$$

■

Remark 5. This theorem tells us that with probability 1, i.e., *almost certainly*, one of the probabilities will be arbitrarily higher than the rest. Hence the partition algorithm wil consistently select the corresponding model as being the truest one in the model set.
Remark 6. The condition that ensures this is that the truest model has less cumulative error (in the sense of mean exponent, as described by $\hat{a}_{ij}$) in (11), than any other model in the model set. The "amount" by which the cumulative error has to be least, depends on the error bounding constants γ_i's, δ_i's, which we obtain from the covariance matrices P_i.

4 An Example

The algorithm was tested on the following **real-world** problem of system identification: We want to identify the parameters of a d.c. motor operating in the linear region. We start with a sequence of observations from a real d.c. motor (**not** a computer simulation): $y(1)$, ... , $y(n)$. The observations will depend linearly on the state vector of the motor; there is a well-understood second order system of differential equations describing the evolution of the state vector for a d.c. motor operating in the linear region. We discretize time with a time step h to obtain the following difference state equations:

$$\begin{bmatrix} i(k+1) \\ w(k+1) \end{bmatrix} = \begin{bmatrix} 1-h\cdot R/L & -h\cdot F_E/L \\ h\cdot F_E/J & 1-h\cdot B/J \end{bmatrix} \begin{bmatrix} w(k) \\ i(k) \end{bmatrix} +$$

$$\begin{bmatrix} h/L & 0 \\ 0 & h/J \end{bmatrix} \begin{bmatrix} V(k) \\ T(k) \end{bmatrix} + u(k) \tag{20}$$

Here the state variables are $i(k)$ and $w(k)$: $i(k)$ is the rotor current, $w(k)$ the shaft angular velocity . $V(k)$, the input voltage and $T(k)$, the input torque, are the control variables. The time step (in seconds) is h. The noise $u(k) = [u_1(k)\ u_2(k)]^T$ is assumed zero mean, Gaussian, white. The parameters of the system are:

1. R is the resistance measured in Ohms
2. L is the inductance measured in Henrys
3. F_E is electromotive force coefficient measured in Volt-sec/rad
4. J is the moment of inertia measured in kg-m^2
5. B is the coefficient of friction measured in Nt-m-sec/rad

In the notation of the previous sections, the parameter vector is $\mathbf{a} = [R\ L\ K\ J\ B]$.

We turn the motor on by applying to it an input voltage of 50 Volts and zero torque. That is, $V(k)$=50 for $k = 1,2,...$ and $T(k)$= 0 for $k = 1,2,....$ We let the motor operate for a few seconds and observe its operation (as we mention below, we actually measure its current). Of course, the actual motor obeys (20) only approximately. We want to find a model of the form (20) that reproduces the observations as closely as possible. We will use the partition algorithm to find such a model.

We must choose what the observation $y(k)$ will be. We expect we will get better identification from observations of both $i(k)$ and $w(k)$. However, measuring accurately the angular velocity requires expensive instrumentation. On the other hand, it is easy to check that even when we are observing only the current $i(k)$, the system is uniformly completely observable, so the identification algorithm should work. We choose to observe only the current:

$$y(k) = i(k) + v(k) \tag{21}$$

We take 128 observations of the motor, one every 10 ms (h=.01 sec). That is, the observations span a time of 1.28 seconds. These observations are recorded digitally as 128 eight-bit numbers.

We proceed to compute the statistics of the observation error $v(k)$. The observations are digitally recorded, so there is quantization error; other than that, the observation is perfect. We assume the error to be Gaussian, white, and zero-mean. Now we will use information about the quantization method to compute the variance of the observation error, $R(k) = E(\tilde{y}(k)\tilde{y}^T(k))$. Set three standard deviations to be equal to one-half the resolution of the quantizer. We have an eight-bit quantizer to measure a maximum of 1 Amp current, and the resolution is approximately 4 mAmp's, so the standard deviation of $v(k)$ is approximately .7 mAmp.

We now compute, in a similar manner, the statistics of the state noise $u(k)$. We assume the system to be linear and so we expect, after a transient phase, a steady state operation. Indeed, examining the $i(k)$ observations, we notice an initial current peak, followed by a dip and then an approximately constant current region. However, in this last region we observe fluctuations in the data that exceed what can be explained as observation error. We attribute these fluctuations to state noise. Assuming the maximum fluctuation from the average value of the current to be three standard deviations, we compute the standard deviation of the current error ($u_1(k)$) to be 8 mAmp, i.e., 33% of the steady state value. We assume (arbitrarily) the angular velocity error to also have a standard deviation of 33% of the steady state to get a standard deviation of .01 rpm. Therefore, the diagonal elements of the covariance matrix $Q(k)$ are .008, .01. We assume the off-diagonal elements to be zero.

Admittedly, this is a rough estimate, but the partition algorithm is known to perform stably even when the noise statistics are not very accurately estimated.

Having estimated the noise statistics, we proceed to define the parameter vector. We have to make the following decisions: How many unknown parameters are there? (What is N?) How many models? (What is K ?) If the ith parameter can take K_i values, then $\mathbf{a}$ can take $K = K_1 \cdot K_2 \cdot ... \cdot K_N$ values. Then we would have to implement the algorithm with K different models. Obviously, for larger K the computational load gets bigger; so it is in our interest to keep both N and K_i, $i = 1,2,..,N$ small. We can achieve this by measuring some of the parameters directly, by standard lab techniques, rather than using the partition algorithm. We have fairly reliable ways to measure the resistance R and the constant F_E; we find them to be R=300 Ohm, F_E=1.2 Volt-sec/rad; this leaves L, J, B to be identified. It must be emphasized that the values of the L, J, B parameters are unknown to us; the only way we have to evaluate the goodness of the identified parameters is by comparing the performance of the true physical system with that of our computer model.

Given three unknown parameters L, J, B, the identification is essentially a search in the three-dimensional parameter space. We are trying to find a point (or a small neighborhood of points), that is, parameter values, such that the corresponding model will fit to our data. We achieve this in the following way: We choose some parameter vector in the parameter space and a big region around the vector; we span the region by K parameter vectors (i.e., models) and choose, with the partition algorithm, the most likely parameter vector. Now we choose a new, smaller region around the new vector and repeat the procedure. By successive iterations (*epochs* of the algorithm) we get progressively smaller regions in the parameter space, as long as the algorithm converges for each individual epoch. In the initial stages the models span a large region and so they are far apart; by the arguments of the previous section, convergence is guaranteed. As the regions get smaller, convergence is no longer guaranteed, but we have already zeroed in to a small set of possible models, all of them fitting the observations fairly well.

There are many ways to choose the region around the most likely parameter vector in each epoch; below we describe the two we used

1. Simultaneous Search: Choose an initial parameter vector. Call it $\mathbf{a}^1 = [L^1\ J^1\ B^1]$. Also choose parameter variations $\delta L, \delta J, \delta B$. Take the eight vectors $\mathbf{a}_j^1$, $j = 1,2,...,8$ defined by $[L^1 \pm \delta L, J^1 \pm \delta J, B^1 \pm \delta B]$. Apply the partition algorithm once; that is, compute all the p_{jn} $j = 1,2,...,8$, $n = 1,2,...,128$ and choose the "true" model to be that which has maximum $p_{j,128}$. Now take the "true" parameter vector $\hat{\mathbf{a}}^1 = [L^2\ J^2\ B^2] = \hat{\mathbf{a}}^1$, (i.e., the one that corresponds to the most likely model) and set $\mathbf{a}^2 = [L^2\ J^2\ B^2] = \hat{\mathbf{a}}^1$. Take the eight vectors $\mathbf{a}_j^2$, $j = 1,2,...,8$ defined by $[L^2 \pm \delta L, J^2 \pm \delta J, B^2 \pm \delta B]$. $i = 1,2,3$. Go through the next epoch of the algorithm. At the t-th epoch select the most probable vector $\hat{\mathbf{a}}^t$ in the parameter space and use this as $\boldsymbol{\alpha}^{t+1}$. This is a way to search through the three-dimensional parameter space in all three parameters simultaneously.

2. Sequential Search: Choose eight models: The ith model is $[L_1, J_1, B_3 +$

Table 1

Parameters	L	B	J
Epoch 1	1.5/.05	.0005	.00005
Epoch 2	1.5	.0011/.0001	.00005
Epoch 3	1.5	.0011	.00009/.00001

$i \cdot \delta B_3]$, $i = 1, ..., 8$. Select the most probable one. Vary the other parameters in the same way, one at a time. Repeat the process as many times as necessary. This is a search in the parameter space where one of the three parameters is searched for at a time. We cycle through the three parameters sequentially.

Initially, with both a sequential and a simultaneous search most models perform poorly and the algorithm has no trouble selecting the one that does much better than the rest. Eventually all models are concentrated in a small region of the parameter space and the algorithm cannot discriminate between them easily. There is no further convergence: the conditions of Theorems 1 and 2 are not satified.

In Figures 1-3, and 4-7 we can see the evolution of the identification algorithm, epoch by epoch. In particular, we see how the recursively computed probabilities p_{ij} evolve as more observations are used. We see that by the time step 128, one model has consistently higher probability than any other model; we also see, however, that in later epochs (e.g., epoch 4 of the simultaneous search) the second-best model has almost equal probability to that of the best model. This is the case when the algorithm reaches its limit of discrimination.

Each one of the figures shows how in one epoch, one of the models is selected as most likely. As we move to better approximations more than one model closely reproduces the observed data, and so the most probable model is almost as probable as the second runner. This corresponds to the situation where the cumulative squre errors of two models are very close; then the α quantities of Section 3 are not small enough to guarantee convergence. On the other hand, in the initial epochs, one model is clearly selected as best among the eight possible ones. Tables 1 and 2 show the models selected at different epochs for each type of search.

Table 1 outlines the history of sequential search. The first entry in every position of the table indicates the value for that parameter of the truest model selected at the end of the corresponding epoch. When there is a slash and a second entry, this indicates that this was the parameter varied at the particular epoch. For example, in epoch 1 we were varying L, by a step $\delta L = .05$, keeping B and J constant. The best model had parameter values $[L\ B\ J] = [1.5\ .0005\ .00005]$. The truest model for which we had convergence of the algorithm was found on epoch 3 and had $[L\ B\ J] = [1.5\ .0011\ .00009]$.

Figure 1: The probabilities p_{ij} as computed in epoch 1 of the sequential search. $i = 1, ..., 8$, $j = 1, ..., 128$. The probability of the truest model is shown by a solid line.

Figure 2: The probabilities p_{ij} as computed in epoch 2 of the sequential search. $i = 1, ..., 8$, $j = 1, ..., 128$. The probability of the truest model is shown by a solid line.

Figure 3: The probabilities p_{ij} as computed in epoch 3 of the sequential search. $i = 1, ..., 8$, $j = 1, ..., 128$. The probability of the truest model is shown by a solid line.

Figure 4: The probabilities p_{ij} as computed in epoch 1 of the simultaneous search. $i = 1, ..., 8$, $j = 1, ..., 128$. The probability of the truest model is shown by a solid line.

Figure 5: The probabilities p_{ij} as computed in epoch 2 of the simultaneous search. $i = 1, ..., 8$, $j = 1, ..., 128$. The probability of the truest model is shown by a solid line.

Figure 6: The probabilities p_{ij} as computed in epoch 3 of the simultaneous search. $i = 1, ..., 8$, $j = 1, ..., 128$. The probability of the truest model is shown by a solid line.

Figure 7: The probabilities p_{ij} as computed in epoch 4 of the simultaneous search. $i = 1, ..., 8$, $j = 1, ..., 128$. The probability of the truest model is shown by a solid line.

Table 2

Parameters	L	B	J
Epoch 1	1.5/.25	.0010/.0001	.0001/.00005
Epoch 2	1.55/.05	.00098/.00002	.00009/.00001
Epoch 3	1.54/.01	.00099/.00001	.000092/.000002
Epoch 4	1.535/.005	.00099/.00001	.000093/.000001

Table 2 gives the same kind of information for simultaneous search. Here every parameter is varied simultaneously, so we have in every position of the table two entries, separated by a slash. The first entry gives the actual truest value found in the corresponding epoch and the second one gives the step by which this parameter was varied. The final truest model, found after four epochs, had $[L\ B\ J] = [1.535\ .00099\ .000093]$, very close to that found by the sequential search.

In Figures 8 and 9 we see the actual output plotted against the optimal estimates computed according to the parameters of the "best" model chosen by simultaneous and sequential search.

5 Conclusions

We have proven that, given a collection of possible linear models to fit a series of observations, the partition algorithm will converge to one of the models if certain conditions are satisfied.

Consider first Theorem 1. Two conditions are necessary: Condition 1(a) requires that the cumulative expected error be growing linearly (even if slowly) for every model. Condition 1(b) says that there is some model $\mathcal{M}_i$ that has smaller cumulative expected error (see α_{ij}). Then, according to the theorem, the expected ratio of the probabilities p_{in}, p_{jn} goes to infinity for every j. The model with less error wins on the average.

For Theorem 2, we need a stronger condition: if the expected *exponentiated* difference of errors decreases exponentially, then we have convergence of the probability ratios to 0 with probability 1. That is, one model gets almost certainly probability 1 and all the other models get probability 0. So the preferred model, in other words the one that best fits the data, is almost certainly selected as the true one.

As already noted, the convergence conditions are expressed in terms of inequalities involving certain constants (13), (17). These constants belong to two

Figure 8: The current measurements for the actual motor (dotted line) and for the best model (solid line) selected by sequential search.

Figure 9: The current measurements for the actual motor (dotted line) and for the best model (solid line) selected by simultaneous search.

categories. On the one hand we have "error growth" constants, such as α_{ij}, $\hat{\alpha}_{ij}$, β_i. Assume that one of the models in the model set, say $\mathcal{M}_i$, is either the true model, i.e. the observations $y(1), y(2), ...$ are generated from it, or very close to the true model, and all the other models are very different. Then the error growth constants α_{ij}, $\hat{\alpha}_{ij}$ will have to be small for all j and, conversely, the constants α_{ji}, $\hat{\alpha}_{ji}$ will be large for all j. This is just a quantitative way to say that the error of model $\mathcal{M}_i$ has to be "small" compared to the error of other models. But what constitutes a small error? To determine this, we must compare the error growth constants with some error baseline. This baseline is provided by the γ_i and δ_i constants, which belong to the second category and provide bounds to the error of every model. In other words, the γ_i and δ_i constants provide some baseline against which we can measure the smallness of the error of the most succesful model.

From the above discussion, the following behavior can be expected from the partition algorithm. In the first few epochs of identification, we start with models that are widely different (they correspond to points in the parameter space that are far apart from each other). Of all these models, one (the truest) will be closest to the actual system, in the sense that it fits the observations relatively well; the rest of the models are so different from the truest model, that they will not fit the observations well. So the error growth rate constants (which are expected to be large for models that are distant from the true model) will be small for the truest model, and we will have quick convergence.

However, after several epochs we will end up with all models being in the same relatively small part of the parameter space and hence they will be almost equally good in fitting the observations. Then the error rate constants will all be close to 1 and the convergence will be slower. Eventually, all the models will have almost the same performance, and Conditions 1(b) or 2 will not be satisfied. Then convergence is not guaranteed, and the algorithm produces conditional probabilities that are all approximately equal; no one model has consistently higher probability.

We observe the behavior as expected when we run the algorithm on a difficult real system identification problem. As long as the algorithm chooses between models that are widely different, choice is easy and performance correct. When the choice is narrowed to a small region of the parameter space, the algorithm reaches its discrimination limit. However, good enough parameters have been identified at this point that a close fit to the observed data is possible. This is reflected in the fact that the best model selected reproduces very accurately the observed behavior of the physical d.c. motor. Note also the tolerance to the crude modeling of the noise statistics.

In conclusion, the analysis of the convergence of partition algorithm justifies in a precise manner the commonsense belief that "a model that fits well the observations is very likely to be the correct model". Also, the behavior of the algorithm at the limit of its resolution is explained. Finally, the application of the algorithm to a real-world problem proves its efficiency.

References

[1] D.G. Lainiotis, " Optimal Adaptive Estimation: Structure and Parameter Adaptation" *IEEE Trans. on Automatic Control, AC-106"*, April 1971, 160-170.

[2] D.G. Lainiotis, " Partitioning: A Unifying Framework for Adaptive Systems I: Estimation" *IEEE Proc. vol. 64*, August 1976, 1126-1143.

[3] D.G. Lainiotis, " Partitioning: A Unifying Framework for Adaptive Systems II: Control", *IEEE Proc., 64*, August 1976, 1182-1198.

[4] F.L. Sims, D.G. Lainiotis & D.T. Magill, " Recursive Algorithm for the Calculation of the Adaptive Kalman Filter Coefficients" , *IEEE Trans. on Automatic Control, AC-14*, April 1969, 215-218.

[5] A. Kehagias, *Partition Algorithm for System Identification*, Unpublished diploma thesis, Aristotelian University of Thessaloniki, Greece, 1984 (in Greek).

[6] D.G. Lainiotis, S.K Katsikas & S.D. Likothanasis, " Adaptive Deconvolution of Seismic Signals: Performance, Computational Analysis, Parallelism" , *IEEE Trans. Acoustics, Speech and Signal Processing, ASSP-36*, 1988, 1715-1734.

[7] RF.L. Sengbush & D.G. Lainiotis, " Simplified Parameter Quantization Procedure for Adaptive Estimation", *IEEE Trans. on Automatic Control, AC-14*, June 1969, 424-425.

[8] D.T. Magill, " Optimal Adaptive Estimation of Sampled Stochastic Processes", *IEEE Trans. on Automatic Control, AC-10*, October 1965, 434-439.

[9] C.G. Hilborn & D.G. Lainiotis, " Optimal Adaptive Filter Realizations for Stochastic Processes with an Unknown Parameter" , *IEEE Trans. on Automatic Control, AC-14*, December 1969, 767-770.

[10] C.G. Hilborn & D.G. Lainiotis " Optimal Estimation in the Presence of Unknown Parameters" , *IEEE Trans. on Systems Science and Cybernetics, SSC-5*, January 1969, 38-43.

[11] G. Saridis, *Self-organizing Control of Stochastic Systems*, New York, Dekker: 1977.

[12] M. Aoki, *Optimization of Stochastic Systems*, New York, Academic: 1967.

[13] B.D.O. Anderson, J.B. Moore & R.M. Hawkes, " Model Approximation Via Prediction Error Identification", *Automatica*, 14, 1978, 615-622.

[14] B.D.O. Anderson & J.B. Moore, *Optimal Filtering*, Englewood Cliffs, NJ, Prentice-Hall: 1979.

[15] R.M. Hawkes & J.B. Moore, " Performance Bounds for Adaptive Estimation", *Proc. IEEE*, 64, August 1976, 1143-1150.

[16] J.K. Tugnait, " Convergence Analysis of Partitioned Adaptive Estimators Under Continuous Parameter Uncertainty ", *IEEE Trans. on Automatic Control, AC-25*, June 1980, 569-573.

[17] A.H. Jazwinski, *Stochastic Processes and Nonlinear Filtering*, New York, Academic: 1972.

[18] M. Loeve, *Probability Theory*, New York, Van Nostrand: 1963.