Adaptive Estimation Techniques for Hidden Markov Models

Vikram Krishnamurthy

September 1991

A thesis submitted for the degree of Doctor of Philosophy of the Australian National University

Department of Systems Engineering
Research School of Physical Sciences
and
Engineering
The Australian National University

Chapter 5

Signal Processing of Semi-Markov Models with Decaying States

5.1 Introduction

Hidden Markov models with discrete states have been widely used to model noisy physical systems in communication systems, speech and sonar processing and more recently in biological signal processing of ion channel currents in cell membranes. In these models, the transition probabilities and symbol probabilities are assumed to be independent of the time. Thus the time the homogeneous Markov chain spends in a state is statistically determined by the geometric probability mass distribution [7]. In certain physical systems, however, the transition and/or symbol probabilities are a function of the time to the last transition. Such processes are called semi-Markov processes [8].

One example of a semi-Markov process, are certain biological fractal signals from ionic channel current measurements, as has been studied in an earlier paper [56]. Here the signal behaviour when appropriately time scaled is independent of the bandwidth of the aliasing filter and sampling rate. Its 'ruggedness' is characterized by its fractal dimension. In such a case, fractal models [56], [9], [10], with their time-varying transition probabilities dependent on the time to the last transition become more appropriate than scalar Markov models that are homogeneous, i.e., with time-invariant transition probabilities. Another example is the discrete-semi Markov chain with Poisson-distributed transition times. Such models are widely used in the modelling of seismic signals and images [67], [58].

More sophisticated hidden discrete semi-Markov stochastic processes encountered in some physical systems are such that subsequent to a step transition governed by transition probabilities dependent on the time to the last such step transition, there is exponential (or other) decay until the next step transition. That is, there is memory in the signal model which can be reasonably modelled by some dynamical filter, possibly nonlinear. Signal processing for hidden semi-Markov processes with memory have not been well developed or studied.

The initial motivation for our work has been the need to study certain cell membrane currents which exhibit obvious exponential decay behaviour between step transitions. These are similar to piecewise constant signals passed through a dynamic channel as studied in telecommunication literature [59], save that here the residual memory prior to the next transition appears negligible, or equivalently there is negligible intersymbol interference. In the telecommunication channel situation, the effective number of discrete states and processing effort grows exponentially with the memory length, whereas with our channels since the memory of a past step transition is taken to be negligibly small compared to a new step transition, we show that the channel memory does not affect the computational effort. The schemes we propose seek to estimate the decay rates between step transitions, the time varying transition probabilities of the imbedded stochastic process, and the noise statistics if unknown. This work is preliminary to current investigations where less apparent, more complex resonant dynamics are conjectured.

This work also has been in part motivated by our curiosity to see if HMM signal processing techniques naturally generalize to semi-Markov models and if so, whether there is any incentive for working with such generalized techniques instead of standard HMM techniques as currently used in many areas of signal processing. Our previous work has been with first and higher order HMM and Hidden Fractal Model representations of ion channels in cell membranes [56], [5] and forms the basis of this work.

Another motivation for dealing with such Hidden semi-Markov processes is to see how well the associated HMM processing techniques perform functional learning, a topic of earlier work in the general field of nonlinear dynamic system identification [60] and control [61]. Specifically, here we deal with learning the time functions that govern the transition probabilities and also with learning the time response of an unknown non-linear dynamical filter from which the observations are obtained.

In this chapter, we first recall from [56] reformulations of the scalar time-varying HMM as a 2-vector first order homogeneous HMM. The reformulation is carried out by augmenting the underlying scalar stochastic process with the time spent to the last transition. However unlike [56], due to the presence of the channel memory until the next transition, the symbol probabilities are also a function of the time to the last transition. This allows us to cope with memory in the case when the residual memory (memory effects at a new transition) are negligible. This is indeed the case in many biological applications [11], [12], [13] where the decay time constants are much smaller than the average duration time in a state. Our main task then is to generalize the techniques in [56] to cope with symbol probabilities which are a function of the time to the last transition. In the most general formulation, we allow for different memory characteristics for different states of the semi-Markov chain. It is theoretically possible in some time-varying models that the number of possible discrete-states for the time to the last transition is the number of observations in the data set. However, it turns out that in many models with practical significance (e.g. fractal models) the number quantization levels for this time variable can be significantly reduced via aggregation.

Once the hidden semi-Markov model has been formulated as an augmented homogeneous HMM, known HMM techniques such as the vector versions of the forward-backward algorithm along with the Baum-Welch [17] [18], [19] re-estimation formulae are applied to achieve improved estimates of both the signals and the signal model parameters, including signal levels, transition probabilities and noise statistics. Repeated application of the forward-backward algorithm and the re-estimation formulae achieves maximum *a posteriori* probability (MAP) or conditional mean (CM) signal estimates [5], [52], [21]. To "learn" the signal model parameters with any degree of certainty, there must be sufficient "excitation" of all the possible transitions between the various states. Once the augmented HMM is estimated, it is a straightforward procedure to compute the signal channel memory characteristics in terms of the filter used to model the memory, and the parameters of the time-varying functions that govern the transition probabilities.

When the noise is known to be Gaussian, it is advantageous to re-estimate the means and variances of the symbol probabilities rather than evaluate these probabilities at the various symbol values. This can be done by using the Expectation Maximization (EM) algorithm. In this chapter we propose modifications of this EM formulation which lead to faster convergence and improved state estimates. We also deal with the special case when the dynamical filters are identical for the various states of the semi-Markov process and the states are equally spaced. In such a case it is possible to further reduce the number of estimated parameters. The resulting EM formulation leads to improved signal estimates.

The chapter is organized as follows: In Section 5.2 we formulate the Hidden semi-Markov problem as a 2-vector homogeneous HMM problem and review learning and estimation objectives of HMM schemes. In Section 5.3, estimation and re-estimation schemes are presented for vector HMMs. In Section 5.4, we present simulation studies. Finally, some conclusions are presented in Section 5.5.

5.2 Problem Formulation

In this section first the signal model is described. Next we characterize noisy channels with memory for the scalar semi-Markov process and reformulate the scalar semi-Markov Hidden Markov models as augmented 2-vector Hidden Markov Models. Finally we recall Hidden Markov Model learning and estimation objectives and describe processing schemes for our signal models.

5.2.1 Signal Model

Here scalar finite-state semi-Markov models are described, with an illustration of the fractal model. Then the noisy channel is formulated as a dynamical, possibly nonlinear, filter.

Multi-state semi-Markov model:

Consider a discrete-time, finite-state stochastic process s_k, $k \geq 0$, where for each k, s_k is a random variable taking on a finite number N_s of possible states $q_1, \ldots, q_{N_s}$. Assume that the transition probabilities $a_{ij}(k)$ at time k defined as $a_{ij}(k) \triangleq P(s_{k+1} = q_j | s_k = q_i)$ being functions of the time to the last transition at time k, denoted t_k, are functions of the form:

$$a_{ij}(t_k) : t_k \mapsto [0,1] \quad \text{for } i,j \in [1, N_s] \tag{5.1}$$

since a_{ij} are probabilities. For convenience we work with the number of discrete-time samples after the last transition rather than the actual time. Thus for a sampling time interval of T_s, t_k is quantized to a finite number N_t of possible integer values $\tau_1, \ldots, \tau_{N_t}$, such as $\tau_i = i$ or $\tau_i = 2i$. Without any computational effort and memory constraints, it would be reasonable to take τ_i as the integer i, for $i = 1, \ldots, N_t$. An upper bound on N_t is the total number of observations in the data set. However, as described later, state aggregation allows N_t to be bounded by more realistic values. Notice that when $a_{ij}(t_k)$ is independent of t_k, the transition probabilities are independent of time, and s_k reduces to a homogeneous first order Markov process.

Fractal model example: Fractal stochastic processes [56], [9], [10] are used to model the opening and closing of channel gates in cell membranes. For a continuous-time fractal stochastic process, the rate of transition from one state to another is a function of the time scale in which it is observed. The transition rate matrix of a two state fractal process is of the form

$$\mathbf{H}(t) = \begin{bmatrix} -k_1\, t^{1-D_1} & k_1\, t^{1-D_1} \\ k_2\, t^{1-D_2} & -k_2\, t^{1-D_2} \end{bmatrix} \tag{5.2}$$

where t is the time to the last transition. D_1, D_2 are termed the fractal dimensions and k_1, k_2 are the initial setpoints. For a physically and mathematically plausible density function, $1 \leq D_i < 2$ in (5.2). The corresponding transition probabilities are the solutions of the matrix differential equation

$$\dot{\mathbf{A}}(t) = \mathbf{H}(t)\, \mathbf{A}(t), \;\; \mathbf{A}(0) = \mathbf{I}$$

where $\mathbf{A}(t) = (a_{ij}(t),\; a_{ij}(t) = P(s_t = q_j | s_0 = q_i))$. In general it is not possible to obtain the transition probabilities in a closed form. However when $D_1 = D_2$, then $\mathbf{A}(t) = \exp(\int_0^t \mathbf{H}(\sigma)\, d\sigma)$ and the transition probabilites can be obtained in a closed form.

The discrete-time version of the continuous-time fractal model is obtained by appropriate lowpass (anti-aliasing) filtering and sampling at greater than or equal to the Nyquist sampling rate. The filter bandwidth then determines the temporal resolution. Let $f_s = 1/T_s$ be the sampling frequency. The discrete transition probabilities of the sampled fractal process cannot be obtained in a closed form in general. However, it is easily shown that,

$$a_{11}(t_k) - a_{21}(t_k) = a_{22}(t_k) - a_{12}(t_k) = \exp(k_1 \frac{t_k^{2-D_1} - t_{k-1}^{2-D_1}}{2 - D_1} + k_2 \frac{t_k^{2-D_2} - t_{k-1}^{2-D_2}}{2 - D_2}). \tag{5.3}$$

This illustrates the behaviour of the transition probabilities: with increasing t_k, $a_{11}(t_k)$ and

$a_{22}(t_k)$ increase and asymptotically approach 1; $a_{12}(t_k)$, $a_{21}(t_k)$ asymptotically approach 0. We shall also use (5.3) subsequently when dealing with state aggregation.

Notice that when $D_i = 1$, the transition probabilities are independent of time and s_k reduces to a homogeneous first order Markov process.

Noisy measurement process with truncated memory:

Consider the case where the semi- Markov process s_k is filtered yielding r_k and is imbedded in noise, that is, indirectly observed by measurements y_k. Denote the sequence $y_1, y_2, \ldots, y_k$ by $\mathbf{Y}_k$.

Recall that s_k is a semi-Markov process with states q_i, $i = 1, \ldots, N_s$. The filtered process r_k is dependent on the current state value of s_k, denoted q_i, and the time to the last transition of s_k denoted t_k. We denote this dependence as

$$r_k = f(q_i, t_k) \tag{5.4}$$

Here $f(.)$ is constrained to any smooth function with the following requirements:

(i) Negligible residual memory: if $t_k = 1$ then the value of r_k is independent of the value of r_{k-1}.

(ii) Identifiability: for some t_k and $i \neq j$, $f(q_i, t_k) \neq f(q_j, t_k)$.

A special case of interest is when the measurements y_k consist of r_k corrupted by zero mean, normally distributed white noise as follows:

$$y_k = r_k + w_k, \quad w_k \sim N[0, \sigma_w^2]. \tag{5.5}$$

Exponentially decaying states example [11], [13]: Consider the case when the channel is modelled as a bank of IIR filters $C_i(q^{-1})$ depending on the state q_i of the semi-Markov process. Then r_k is of the form

$$r_k = \begin{cases} c_0(i)\, q_i \text{ for } t_k = 1 \\ C_i(q^{-1})\, r_{k-1} = c_1(i)\, r_{k-1} + c_2(i)\, r_{k-2} + \ldots + c_{R(i)}(i)\, r_{k-R(i)} \text{ otherwise} \end{cases} \tag{5.6}$$

More simply r_k is an exponentially decaying state if $R(i) = 1$ and $0 < c_1(i) \leq 1$. Exponentially decaying states are used to model certain cell membrane currents which exhibit exponential decay between step transitions. Here q_1 is termed the ground state and does not decay exponentially. For $t_k = 1$, r_k is usually considered as q_i since the rise time from the ground state to the state level q_i is much smaller compared to the decay time. Also typically the decay time in a state is much shorter than the duration time in a state so that the residual memory just prior to a transition is negligible.

5.2.2 Reformulation as Hidden Markov Model

First the semi-Markov process is reformulated as a vector homogeneous Markov process. Then we deal with the observations of this imbedded Markov process obtained through the noisy channel with memory (5.4) as a Hidden Markov process.

Formulaton of semi-Markov process as vector homogeneous Markov process

The class of scalar semi-Markov models in (5.1) can be modelled as a homogeneous first order 2-vector Markov process as follows: Define the 2-vector process $\mathbf{S}_k$ as $\mathbf{S}_k = (t_k, s_k)$ for each $k \geq 0$. Clearly $\mathbf{S}_k$ is a finite-state process with $N = N_s N_t$ states. Here N_t is taken as the maximum duration time in any state considering the observation sequence of length T.

It is easily shown that the 2-vector stochastic process $\mathbf{S}_k$, as defined above with $\tau_i = i$ for $i = 1, \dots, N_t$ is a homogeneous, first order Markov process (see [56]).

Notice that

$$t_{k+1} = \begin{cases} t_k + 1 & \text{if } s_{k+1} = s_k \text{ and } t_k < N_t \\ 1 & \text{otherwise} \end{cases} \tag{5.7}$$

So t_{k+1} depends only on t_k, s_k and s_{k+1}. Also from (5.7) the transition probabilities for the homogeneous vector process $\mathbf{S}_k$ are

$$P(\mathbf{S}_{k+1}|\mathbf{S}_k) = \begin{cases} a_{ii}(\tau_h) & \text{if } \mathbf{S}_k = (\tau_h, q_i) \text{ and } \mathbf{S}_{k+1} = (\tau_h + 1, q_i);\ 1 \leq \tau_h < N_t \\ a_{ij}(\tau_h) & \text{if } \mathbf{S}_k = (\tau_h, q_i) \text{ and } \mathbf{S}_{k+1} = (1, q_j),\ i \neq j,\ 1 \leq \tau_h \leq N_t \\ 0 & \text{otherwise} \end{cases} \tag{5.8}$$

which are independent of k.

Remark: If for some integer $N_t' < N_t$, τ_i is defined more generally as $\tau_i = i$, $i = 1, \dots, N_t' - 1$ and $\tau_{N_t'} = i : N_t' \leq i \leq N_t$, the first order Markov property still holds. However, it is readily shown that $\mathbf{S}_k$ is then not necessarily homogeneous. For certain functions $a_{ij}(t_k)$ that "saturate" beyond some t_k, $\mathbf{S}_k$ is homogeneous and aggregation of states in the saturation region is possible as discussed below. ∘

Notation: Denote the set of $N = N_s N_t$ states $\{(\tau_1, q_1), \ldots, (\tau_{N_t}, q_{N_s})\}$ as $\{\mathbf{Q}_1, \mathbf{Q}_2, \ldots, \mathbf{Q}_N\}$, although, not necessarily in the same order. We will denote elements of this set by integer subscripts, usually m or n. Also for $Q_m = (\tau_h, q_i)$ and $Q_n = (\tau_l, q_j)$ where $m, n \in [1, N]$, $\tau_h, \tau_l \in [1, N_t]$, $q_i, q_j \in [1, N_s]$, denote the transition probabilities of the homogeneous $\mathbf{S}_k$ process by

$$\mathbf{A} = (a_{mn}); \quad a_{mn} = a_{(h,i),(l,j)} \triangleq P(\mathbf{S}_{k+1} = \mathbf{Q}_n | \mathbf{S}_k = \mathbf{Q}_m). \tag{5.9}$$

Hidden semi-Markov Models with truncated channel memory

Consider a semi-Markov processes observed through the noisy channel with memory defined by (5.4). Assume that the semi-Markov process s_k and hence the associated vector Markov process $\mathbf{S}_k = (t_k, s_k)$ are hidden, that is indirectly observed by measurements y_k. The vector of probability functions $\mathbf{b}(.) = (b_m(.)) = P(y_k | \mathbf{S}_k = \mathbf{Q}_m)$ where $\mathbf{Q}_m = (\tau_h, q_i)$ is a function of the time to the last transition τ_h. So

$$b_m(y_k) = b_{(h,i)}(y_k) = P(y_k | t_k = \tau_h, s_k = q_i). \tag{5.10}$$

Also assume the independence property

$$P(y_k | \mathbf{S}_k = \mathbf{Q}_m, \mathbf{S}_{k-1} = Q_n, \mathbf{Y}_{k-1}) = P(y_k | \mathbf{S}_k = \mathbf{Q}_m). \tag{5.11}$$

Further, assume that the initial state probability vector $\underline{\pi} = (\pi_m)$ is defined from

$$\pi_m = P(\mathbf{S}_1 = \mathbf{Q}_m). \tag{5.12}$$

The transition probabilities $\mathbf{A}$ are defined as in (5.9). The vector HMM for the $\mathbf{S}_k$ process is denoted $\lambda = (\mathbf{A}, \mathbf{b}(.), \underline{\pi})$. Of course λ also denotes the Hidden semi-Markov model for the $\mathbf{S}_k$ process.

In the special case when the measurements y_k consist of r_k corrupted by zero mean, normally distributed white noise, (see (5.5))

$$b_m(y_k) = (\sqrt{2\,\pi}\,\sigma_w)^{-1} \exp(-(y_k - r_k)^2 / (2\,\sigma_w^2))$$

Notice that the independence assumption on the model reflects itself here as an independence 'whiteness' assumption on w_k. The theory developed in Section 5.3.3 and simulations presented subsequently deal with this special case. The theory and algorithms in the rest of this chapter, however, do not rely on this additive normally distributed noise assumption.

5.2.3 Aggregation

In proving that $\mathbf{S}_k$ is a homogeneous Markov process above, it is assumed that $\tau_i = i$ and N_t is the maximum duration time in any state. Since we may not know this value we can set $N_t = T$, the length of the entire observation sequence, but then the number of states $N = N_t N_s$ will be excessive for computational purposes. Is it possible to quantize the time to the last transition to N_t' states, where $N_t' << T$ with negligible error? We propose to do so by "aggregating" [54] the states $(\tau_{N_t'}, q_i), (\tau_{N_t'+1}, q_i), \ldots, (\tau_{N_t}, q_i)$ into a aggregated state

$$\overline{\mathbf{Q}}_{N_t',i} = \{(\tau_{N_t'}, q_i), (\tau_{N_t'+1}, q_i), \ldots, (\tau_{N_t}, q_i)\}. \tag{5.13}$$

Aggregation Property: We have proved in [56] that the above aggregation leads to negligibly small errors for a certain class of functions $a_{ij}(t_k)$ defined in (5.1) satisfying for arbitrary small $\epsilon > 0$

$$|a_{ij}(t_k) - a_{ij}(t_{k-1})| < \epsilon \quad \text{for } i, j \in [1, N_s] \text{ and } N_t' < t_k < N_t = T \tag{5.14}$$

More specifically, the fractal model with transition probabilities that exponentially converge, see 5.3, and the discrete semi-Markov process with Poisson distributed transition times satisfy (5.14) and so can be aggregated. Simulations show that in most cases for such processes, choosing N_t so that $\epsilon < 0.05$ is adequate. For the rest of this chapter we set $N_t = N_t' << T$ where N_t' is suitably large (usually less than 50) to result in negligible error. Of course, now (5.7) is modified as

$$t_{k+1} = \begin{cases} t_k + 1 & \text{if } s_{k+1} = s_k \text{ and } t_k < N_t \\ t_k & \text{if } s_{k+1} = s_k \text{ and } t_k = N_t \\ 1 & \text{otherwise} \end{cases} \tag{5.15}$$

Transition probability matrix sparseness: Notice that with $\tau_h = h$ for $h < N_t = N_t'$, the relationship between (5.1) and (5.9) is

$$a_{ii}(\tau_h) = a_{(h,i),(h+1,i)}, \text{ and } a_{ij}(\tau_h) = a_{(h,i),(1,j)}, \quad i \neq j, \ 1 \leq \tau_h < N_t. \tag{5.16}$$

Clearly $\mathbf{A}$ has $(N_s N_t)^2 = N^2$ elements. However, since t_{k+1} is restricted as in (5.15) to only three possible values, simple calculations show that $(N^2 - N_s^2 N_t)$ elements of $\mathbf{A}$ are zero. For $i, j \in [1, \ldots N_s]$, only the following elements of $\mathbf{A}$ are not necessarily zero:

$$a_{(h,i),(h+1,i)}, \quad \tau_h \neq N_t; \quad a_{(h,i),(1,j)}, \ i \neq j; \quad \text{and} \quad a_{(N_t,i),(N_t,i)}. \tag{5.17}$$

Consequently, in any scheme to estimate $\mathbf{A}$, only the $N_s^2 N_t$ elements of A in (5.17) need be estimated.

5.2.4 Learning and Estimation Objectives

Given the semi-Markov signal model reformulated as a homogeneous Markov model as described above, and given observations $y_1, y_2, \ldots, y_k$ denoted $\mathbf{Y}_k$, there are three inter-related HMM problems which can be solved [52], [21] and interpreted to yield results for the underlying semi-Markov process.

1. Evaluate the likelihood of a given model λ_i generating the given sequence of data, denoted $P(\mathbf{Y}_k|\lambda_i)$. This allows comparison of a set of models $\{\lambda_i\}$ to select the most likely, given the observations.

2. Estimate signal statistics such as *a posteriori* probabilities for an assumed model λ and data sequence. For a fixed interval T, estimate

$$\underline{\gamma}_{k|T} = (\gamma_{k|T}(m)); \quad \gamma_{k|T}(m) \triangleq P(\mathbf{S}_k = \mathbf{Q}_m|\mathbf{Y}_T, \lambda). \tag{5.18}$$

From $\gamma_{k|T}(m)$, state sequence estimates conditioned on λ, denoted $\{\hat{\mathbf{S}}_k|\lambda\}$, such as maximum a posteriori (MAP) can be generated as

$$\hat{\mathbf{S}}_k^{\mathrm{MAP}} = \mathbf{Q}_n \ \text{ where } n = \arg \max_{1 \leq m \leq N} \gamma_{k|T}(m) \tag{5.19}$$

Of course if $\mathbf{Q}_n = (\tau_h, q_j)$ then $\hat{s}_k^{\mathrm{MAP}} = q_j$ in (5.19).

Also in the special case when the noise is Gaussian, estimate its mean and variance: For the state $\mathbf{Q}_m = (\tau_h, q_i)$, the estimated mean $\overline{\mu}_m$ is obtained as (see [18])

$$\overline{\mu}_m = \frac{\sum_{k=1}^{T} \gamma_k(m)\, y_k}{\sum_{k=1}^{T} \gamma_k(m)} \tag{5.20}$$

and the estimated variance is calculated as

$$\overline{\sigma}_m^2 = \frac{\sum_{k=1}^{T} \gamma_k(m)(y_k^2 - \mu_m^2)}{\sum_{k=1}^{T} \gamma_k(m)} \tag{5.21}$$

3. Processing of the observations based on a model assumption λ, and adjustment (re-estimation) of the model parameters (functions) $(\mathbf{A}, \mathbf{b}(.), \underline{\pi})$, such that the likelihood of the updated model $\overline{\lambda} = (\overline{\mathbf{A}}, \overline{\mathbf{b}}(.), \overline{\underline{\pi}})$ given the observation sequence, $P(Y_T|\overline{\lambda}) > P(Y_T|\lambda)$, and repeating until convergence. The objective is to achieve the most likely model λ^{ML} amongst the set $\lambda = (\mathbf{A}, \mathbf{b}(.), \underline{\pi})$, given the observations.

5.3 Estimation and Re-estimation

We now proceed to solve Problems 1 to 3 above for the hidden semi-Markov model reformulated as a HMM. The formulae of this section turn out to be identical to more familiar ones for scalar HMMs, as in [5], but with scalars s_k replaced by 2-vectors $\mathbf{S}_k$ and the states q_i, q_j replaced by $\mathbf{Q}_m, \mathbf{Q}_n$ which can be associated with the pair of scalar states (τ_h, q_i) and (τ_l, q_j).

5.3.1 Forward-backward Procedure

Consider an observation sequence $\mathbf{Y}_T$ of length T and assumed signal generating model λ.

Forward and backward vector variables are defined as probabilities conditioned on the model λ, for $m = 1, 2, \ldots, N$ as follows:

$$\begin{aligned} \underline{\alpha}_k &= (\alpha_k(m)); \quad \alpha_k(m) \triangleq P(\mathbf{Y}_k, \mathbf{S}_k = \mathbf{Q}_m|\lambda) \\ \underline{\beta}_k &= (\beta_k(m)); \quad \beta_k(m) \triangleq P(\overline{\mathbf{Y}}_k|\mathbf{S}_k = \mathbf{Q}_m, \lambda) \end{aligned} \tag{5.22}$$

where $\overline{\mathbf{Y}}_k$ denotes the future sequence $y_{k+1}, y_{k+2}, \ldots, y_T$.

Recursive formulae for $\underline{\alpha}_k$ and $\underline{\beta}_k$ are readily calculated [56], [5], [21] as

$$\alpha_k(n) = \left(\sum_{m=1}^{N} \alpha_{k-1}(m)\, a_{mn}\right) b_n(y_k), \quad \alpha_1(n) = \pi_n\, b_n(y_1)$$

$$\beta_k(m) = \sum_{n=1}^{N} a_{mn}\, b_n(y_{k+1})\, \beta_{k+1}(n), \quad \beta_T(m) = 1 \tag{5.23}$$

where $\mathbf{Q}_n = (\tau_h, q_j)$ for any $1 \leq \tau_h < N_t$. Updating $\underline{\alpha}_k$ and $\underline{\beta}_k$ requires the order of $N^2\, M$ multiplications and additions at each time instant k because only $N^2\, M$ components of $\mathbf{A}$ are not necessarily zero, see (5.17).

The optimal *a posteriori* probabilities associated with Problem 2 are given from [5], [21]

$$\underline{\gamma}_k = (\gamma_k(m)); \;\; \gamma_k(m) = \frac{\alpha_k(m)\, \beta_k(m)}{\sum_{m=1}^{N} \alpha_k(m), \beta_k(m)}, \quad m = 1, 2, \ldots, N. \tag{5.24}$$

Then using (5.19) we obtain MAP fixed-interval smoothed estimates. The likelihood function, which is the end result of the Problem 1, is calculated as

$$L_T \triangleq P(\mathbf{Y}_T|\lambda) = \sum_{m=1}^{N} \alpha_T(m). \tag{5.25}$$

5.3.2 Baum-Welch Re-estimation Formulae

Define the joint conditional probability

$$\zeta_k(m,n) \triangleq P(\mathbf{S}_k = \mathbf{Q}_m, \mathbf{S}_{k+1} = \mathbf{Q}_n, \mathbf{Y}_T|\lambda) = \alpha_k(m)\, a_{mn}\, b_n(y_{k+1})\, \beta_{k+1}(n) \tag{5.26}$$

Re-estimation of $\mathbf{A}$, $\underline{\pi}$ are given from

$$\overline{a}_{mn} = \frac{\sum_{k=1}^{T-1} \zeta_k(m,n)}{\sum_{k=1}^{T-1} \sum_{n=1}^{N} \zeta_k(m,n)}; \;\; \overline{\pi}_m = \frac{\sum_{n=1}^{N} \zeta_1(m,n)}{\sum_{m=1}^{N} \sum_{n=1}^{N} \zeta_1(m,n)} \tag{5.27}$$

In updating the vector of probability functions $\mathbf{b}(.)$, it is reasonable to do this at a finite number of points $v_1, v_2, \ldots, v_M$ in the range of the signals y_k. Quantizing y_k to these levels, gives a quantized signal $\overline{y}_k$ and allows re-estimation of $b_m(v_j)$ as

$$\overline{b}_m(v_j) = \frac{\sum_{k=1}^{T} \gamma_k(m)\, \delta(v_j - \overline{y}_k)}{\sum_{k=1}^{T} \gamma_k(m)} \text{ where } \mathbf{Q}_m = (\tau_h, q_i) \text{ for any } 1 \leq \tau_h < N_t \tag{5.28}$$

Here $\delta(t)$ is unity when $t = 0$ and zero otherwise.

It is proved in [52] that with the re-estimation formulae (5.27), (5.28), the model $\overline{\lambda} = (\overline{\mathbf{A}}, \overline{\mathbf{b}(.)}, \overline{\pi})$ is more likely than λ, given the observations, in that $P(\mathbf{Y}_T|\overline{\lambda}) \geq P(\mathbf{Y}_T|\lambda)$.

Remark: The Baum-Welch algorithm belongs to a class of numerical algorithms called Expectation Maximization (EM) algorithms [20], [15]. In Section 5.3.3, we shall consider other more suitable EM algorithms in the presence of Gaussian noise. ∘

Determining transition probability parameters and filter coefficients:

Here we describe how the parameters that govern the time-varying transition probabilities and the filter coefficients are obtained from the Baum-Welch estimates.

Transition Probability parameters: Note that from the Baum-Welsh estimate $\overline{\mathbf{A}}$ of $\mathbf{A}$, the parameters of the function a_{ij} (see (5.1)) are obtained by fitting functions of the form a_{ij} to $\overline{a}_{(h,i),(h+1,j)}$, $1 \leq \tau_h < N_t$ (see (5.16)) in a least squares sense. However, because of the finite length of the observation sequence, for large values of τ_h it is often the case that there are not sufficient such transitions to get a reliable estimate of $a_{(h,i),((.),j)}$. Also, we ignore $a_{(1,i),((.),j)}$ since it includes all events that happen faster than our time scale of interest, this time scale being determined by the bandwidth of the lowpass filter used. Thus typically for a few thousand observations, it makes sense to use $a_{(h,i),(h+1,j)}$ only for $2 \leq \tau_h \leq 5$ (say) in the least squares fit.

Filter coefficients: If the functional form of the dynamical filter is known then the actual parameters can easily be obtained by least squares fitting. For example if the channel is modelled by IIR filters (see (5.6), then the coefficients of the IIR filters $C_i(q^{-1})$ are calculated from the estimates of $b(.)$ as follows: Let $b^*_{h,i} = \max_j b_{h,i}(v_j)$. Also define the vectors $\mathbf{b}^*_i = [b^*_{R(i)+2,i}, \ldots, b^*_{N_t,i}]$ and the vector of filter coefficients $\mathbf{c}(i) = [c_0(i), \ldots, c_{R(i)}(i)]^T$. Then estimates of the filter coefficients are obtained by solving for $\mathbf{c}^{(i)}$ from

$$\mathbf{b}^*_i = \mathbf{B}\,\mathbf{c}^{(i)}$$

in a least squares sense where

$$\mathbf{B} = \begin{bmatrix} b^*_{R(i)+1,i} & b^*_{R(i),i} & \cdots & b^*_{1,i} \\ b^*_{R(i)+2,i} & b^*_{R(i)+1,i} & \cdots & b^*_{2,i} \\ \vdots & \vdots & \vdots & \vdots \\ b^*_{N_t-1,i} & b^*_{N_t-2,i} & \cdots & b^*_{N_t-R(i)-1,i} \end{bmatrix} \tag{5.29}$$

5.3.3 Fast converging Re-estimation Algorithm for Gaussian Noise

In this subsection we present efficient methods of re-estimating $b_m(.)$. When these schemes are used in conjunction with the Baum-Welch equations for re-estimating the transition probabilities, convergence rates are significantly improved.

Efficient re-estimation in Gaussian noise (Scheme 1): When the Markov process is imbedded in Gaussian noise, then the EM algorithm can be used to directly re-estimate the mean $\overline{\mu}_m$ and variance $\overline{\sigma}_m^2$ instead of re-estimating the M parameters $b_m(v_j), j = 1, \ldots, M$ for each m. Once $\overline{\mu}_m$ and $\overline{\sigma}_m^2$ have been calculated using (5.20) and (5.21), the Gaussian density $b_m(.)$ can be updated as

$$b_m(v_j) = (\sqrt{2\pi}\,\overline{\sigma}_m)^{-1} exp(-(v_j - \overline{\mu}_m)^2/(2\overline{\sigma}_m^2))$$

where $\mathbf{Q}_m = (\tau_h, q_i)$.

Fast converging algorithm (Scheme 2): We now present a modification of the above EM algorithm for even faster convergence.

It is proved in [23] that if the observations were obtained by adding white Gaussian noise to an imbedded independent process (rather than a markov process) then the convergence rates of the resulting EM re-estimation equations for the mean and variance can be improved as follows:

Define the reestimated mean and variance obtained from the standard EM re-estimation formulae after the pth pass as $\psi^{(p)}$ where $\psi^{(p)} = (\overline{\mu}_m, \overline{\sigma}_m^2)$. Then the standard EM re-estimation formulae can be written as

$$\psi^{(p+1)} = G(\psi^{(p)}).$$

where G represents the EM re-estimation (see [15], pp 85-88). Let $\overline{\psi}^{(p)}$ be the re-estimated model obtained after the pth pass of the fast converging EM algorithm. Then the estimates from the $p+1$ pass of the fast converging EM algorithm are

$$\overline{\psi}^{(p+1)} = \overline{\psi}^{(p)} + \eta(G(\overline{\psi}^{(p)}) - \overline{\psi}^{(p)})$$

with $\eta = 1$ corresponding to the standard EM scheme. It is shown in [15] that strongly consistent maximum likelihood estimates $\overline{\psi}$ exist, and that provided that the initial estimates are chosen close enough to $\overline{\psi}$, local convergence occurs for $0 \leq \eta < 2$. Asymptotical optimal rates of convergence occur for some η^* with $1 < \eta^* < 2$. Typically, if the component densities are well separated, η^* is near 1 and convergence is fast, otherwise η^* is 2 and convergence is slow.

We conjecture that these results generalize to the case where the imbedded process is a Markov processes as follows: Let $\overline{\psi}^p$ be the re-estimated mean and variance after the pth pass. Then

$$\overline{\psi}^{(p+1)} = \overline{\psi}^{(p)} + \eta(\psi^{(p+1)}) - \overline{\psi}^{(p)}). \tag{5.30}$$

Here $\psi^{(p+1)} = (\overline{\mu}_m, \overline{\sigma}_m^2)$ is calculated from (5.20) and (5.21) and $0 < \eta < 2$. Computer simulations show the that the rates of convergence are indeed increased when using the above modified algorithm with $1 < \eta < 2$. Also if the convergence is fast for the standard case, choosing η close to 2 does not significantly slow convergence. However, most often the initial model and hence the initial mean and variance estimates are chosen arbitrarily and may be quite different from the most likely values. In such cases choosing η close to 2 significantly improves convergence rates. So we recommend in general that η be chosen close to 2.

Exponentially Decaying States: When it is known that the Markov states decay exponentially as in (5.6) then the EM algorithm can be used to directly obtain the exponential decay constants $c_i(i)$, $i = 1, \ldots N_s$.

E step: This involves computing $Q(\lambda, \overline{\lambda})$ which is the expectation of the log of the likelihood function of the fully categorized data (see [20], [15] for details). Here

$$Q(\lambda, \overline{\lambda}) = \sum_{m=1}^{N} \sum_{n=1}^{N} \sum_{k=1}^{T-1} \zeta_k(m,n) \log \overline{a}_{mn} + \sum_{n=1}^{N} \gamma_1(n) \log \overline{\pi}_n + Q_2 \tag{5.31}$$

where

$$Q_2 = \sum_{h=1}^{N_t} \sum_{i=1}^{N_s} \sum_{k=1}^{T} \gamma_k(h,i) \log\left(\frac{1}{\sqrt{2\pi}\sigma_w} \exp \frac{-(y_k - \overline{\mu}_{h,i})^2}{2\sigma_w{}^2}\right). \tag{5.32}$$

Because we apriori assume that the states decay exponentially according to (5.6), it follows that $\overline{\mu}_{h,i} = (\overline{c_1}(i))^{h-1} \overline{\mu}_{1,i}$, $h = 1, \ldots, N_t$, $i = 1, \ldots, N_s$. So Q_2 can be rewritten as

$$Q_2 = \sum_{h=1}^{N_t} \sum_{i=1}^{N_s} \sum_{k=1}^{T} \gamma_k(h.i) \log\left(\frac{1}{\sqrt{2\pi}\sigma_w} \exp \frac{-(y_k - (\overline{c_1}(i))^{h-1} \overline{\mu}_{1,i})^2}{2\sigma_w{}^2}\right). \tag{5.33}$$

M step: This involves finding $\overline{\lambda}$ to maximize $\mathcal{Q}(\lambda, \overline{\lambda})$. Maximizing the first two terms on the right hand side of (5.31) yields the standard Baum-Welch equations for re-estimating $\overline{a}_{mn}$ and $\overline{\pi}_m$. Now consider maximizing the third term in (5.31) namely $\mathcal{Q}_2$ which is expressed as (5.33). Solving for $\partial \mathcal{Q}_2 / \partial \overline{\mu}_{1,i} = 0$ yields

$$\overline{\mu}_{1,i} = \frac{\sum_{k=1}^{T} \sum_{h=1}^{N_t} \gamma_k(h,i)\,(\overline{c_1}(i))^{h-1}\, y_k}{\sum_{k=1}^{T} \sum_{h=1}^{N_t} \gamma_k(h,i)\,(\overline{c_1}(i))^{2h-2}} \tag{5.34}$$

Also solving for $\partial \mathcal{Q}_2 / \partial \overline{c_1}(i) = 0$ yields

$$\sum_{k=1}^{N} \sum_{h=1}^{N_t} \gamma_k(h,i)\,(h-1)\,(\overline{c_1}(i))^{h-2}\, y_k = \sum_{k=1}^{N} \sum_{h=1}^{N_t} \gamma_k(h,i)\,(h-1)\,(\overline{c_1}(i))^{2h-3}\, \overline{\mu}(1,i) \tag{5.35}$$

Based on (5.34) and (5.35), we now propose three methods for re-estimating $\overline{c_1}(i)$ and $\overline{\mu}(1,i)$. These methods are also used in Chapter 6.

1. *Standard EM Re-estimation*: To solve for $\overline{c_1}(i)$ and $\overline{\mu}(1,i)$, we need to solve (5.34) and (5.35) simultaneously. This can be done for example by substituting (5.34) for $\overline{\mu}(1,i)$ in (5.35). Then the resulting equation can be solved numerically to yield estimates for $\overline{c_1}(i)$ and then $\overline{\mu}(1,i)$ can be obtained from (5.34). The disadvantage of this method is that we need to numerically solve a $3\,N_t - 4$ order polynomial equation for $c_i(1)$.

2. *Block Component Re-estimation* [67]: Like the standard EM algorithm, this scheme ensures that the likelihood function improves after each pass. Here $\overline{c_1}(i)$ and $\overline{\mu}(1,i)$ are updated in alternate passes. So we keep $\overline{\mu}(1,i)$ constant in one pass and update $\overline{c_1}(i)$ in that pass. Then we keep $\overline{c_1}(i)$ constant in the next pass and update $\overline{\mu}(1,i)$ and so on. Note that the transition probabilities are updated every pass since their re-estimation equations are independent of maximizing $\mathcal{Q}_2$.

3. *Quasi-EM Re-estimation*: This scheme is similar to that proposed in [65] and [66]. In re-estimating $\overline{c_1}(i)$ and $\overline{\mu}(1,i)$, we use (5.34) and (5.35) with $\overline{c_1}(i)$ and $\overline{\mu}(1,i)$ in the right hand sides of the equations replaced by $c_1(i)$ and $\mu(1,i)$. Simulations show that this quasi-EM algorithm yields acceptable results. Of course, this algorithm is also much simpler to implement than the standard EM algorithm since (5.34) and (5.35) do not need to be solved simultaneously. So only a $2\,N_t - 3$ order polynomial equation needs to be solved numerically.

A very important issue for the above three methods is the existence of a unique solution

$\overline{c_1}(i)$ satisfying $0 < \overline{c_1}(i) \leq 1$. While we have not been able to prove uniqueness, extensive simulations seem to confirm that there is a unique $\overline{c_1}(i)$ in the range $0 < \overline{c_1}(i) \leq 1$. In any case, if more than one solution does exist, then in the M step we choose that solution which maximizes Q_2.

Equally spaced means: In some biological examples [13] it is apriori known that the means $\mu_{h,i}$, $i = 1 \ldots, N_s$ are equally spaced. An example is when the Markov states are equally spaced and the filters are identical. We can exploit the knowledge of the means being equally spaced to speed up convergence and further decrease the number of parameters estimated as follows: Let $\Delta \triangleq q_{i+1} - q_i$, $i = 2, \ldots, N_s - 1$ be the state separation. Since the filters are such that the means are equally spaced, let $\mu_{h,i+1} - \mu_{h,i} = \Delta_h$, $i = 1, \ldots, N_s - 1$. So we only need estimate Δ_h and $\mu_{h,1}$ for each h, $h = 1, \ldots, N_t$.

It is proved in the appendix that the maximization (M) step of the EM algorithm yields estimates of these as

$$\overline{\mu}_{h,1} = \frac{\sum_{k=1}^{T} \gamma_k(h,1)\, y_k}{\sum_{k=1}^{T} \gamma_k(h,1)}, \quad h = 1, \ldots, N_t \tag{5.36}$$

$$\overline{\Delta}_h = \frac{\sum_{k=1}^{T} \sum_{i=2}^{N_s} (i-1)\, \gamma_k(h,i)(y_k - \overline{\mu}_{h,1})}{\sum_{k=1}^{T} \sum_{i=2}^{N_s} (i-1)^2 \gamma_k(h,i)} \tag{5.37}$$

Hence by re-estimating $\overline{\mu}_{h,1}$ and $\overline{\Delta}_h$ at each pass we can compute

$$\overline{\mu}_{h,i} = \overline{\mu}_{h,1} + (i-1)\,\overline{\Delta}_h, h = 1, \ldots, N_t, \; i = 2, \ldots, N_s$$

Usually in biological examples the 'baseline' state does not decay, i.e., $\mu_{h,1} = 0, h = 1, \ldots, N_t$. In such cases use (5.36) to re-estimate $\overline{\mu}_{h,2}$. Then compute $\overline{\Delta}_h$ from (5.37) with $\overline{\mu}_{h,2}$ replacing $\overline{\mu}_{h,1}$.

More generally, if the function $g(\Delta_h, i)$ defined by

$$\mu_{h,i} = \mu_{h,1} + g(\Delta_h, i), \; g(\Delta_h, 1) = 0 \tag{5.38}$$

is known, then it is shown similarly to the proof in the appendix that $\overline{\mu}_{h,1}$ is obtained from (5.37) and that Δ_h is computed by solving

$$\sum_{k=1}^{T} \sum_{i=2}^{N_s} \gamma_k(h,i)(y_k - \overline{\mu}_{h,1} - g(\overline{\Delta}_h, i)) \frac{\partial g(\overline{\Delta}_h, i)}{\partial \overline{\Delta}_h} = 0 \tag{5.39}$$

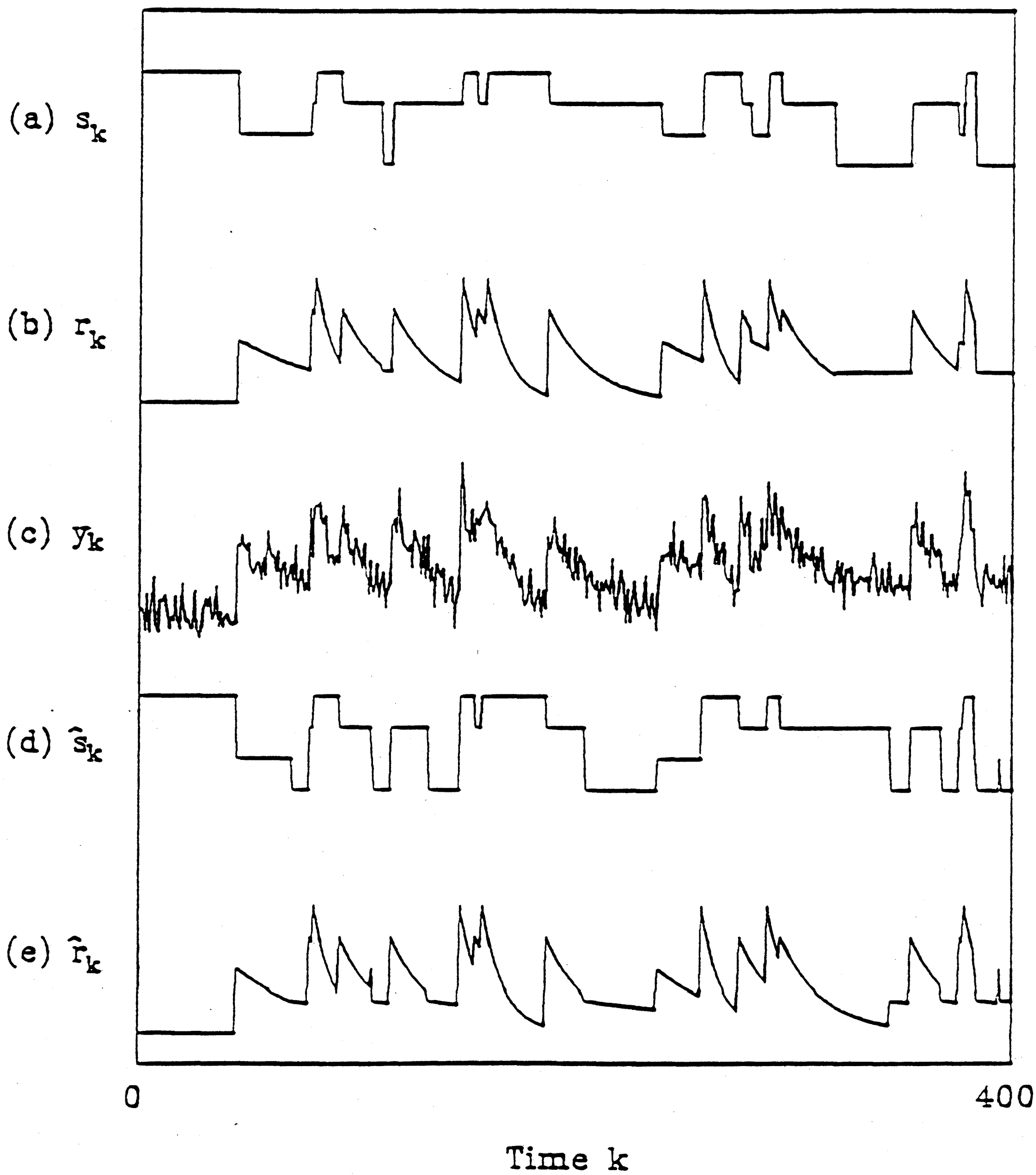


Fig. 5.1 : Performance of Scheme 2 on noisy exponentially decaying Markov signal

5.4 Computer Studies

First we illustrate the effectiveness of the algorithms presented in this chapter on computer generated data. In particular we compare likelihood functions and signal estimates. Then we present results of the above algorithms on a segment of data obtained from cell membrane currents. See [56] for a brief description of associated implementation aspects including Adaptive scaling.

Remark: When modelling finite segments of data certain transitions may only occur rarely or perhaps not at all. The transition probabilities for these "rare events" cannot be reliably re-estimated using any statistical approach due to the lack of sufficient "excitation". Thus in interpreting model estimation results via the re-estimation equations on finite data, not too much credence should be given to the estimation of low probability events. $\circ$

Hidden Fractal Model data: See Chapter 4 for results on the simulation performance of the algorithms on fractal data.

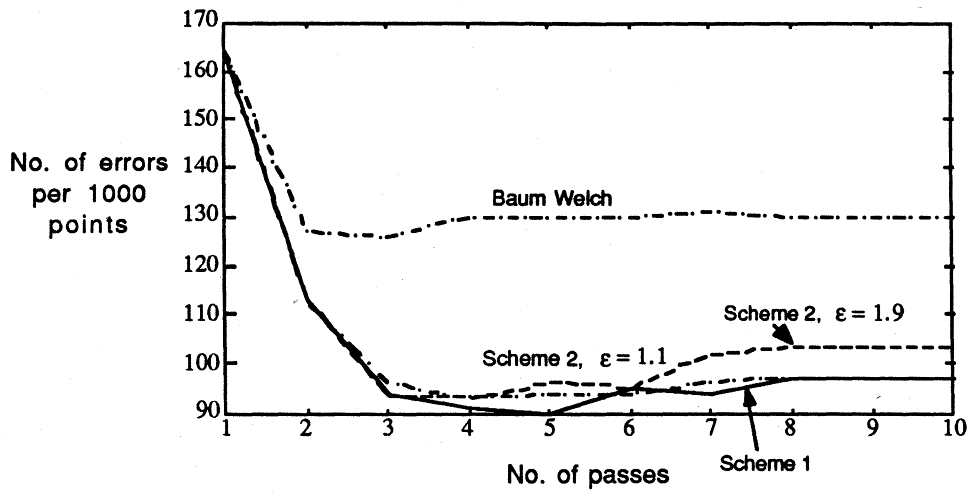
Hidden Markov model with Decaying states:

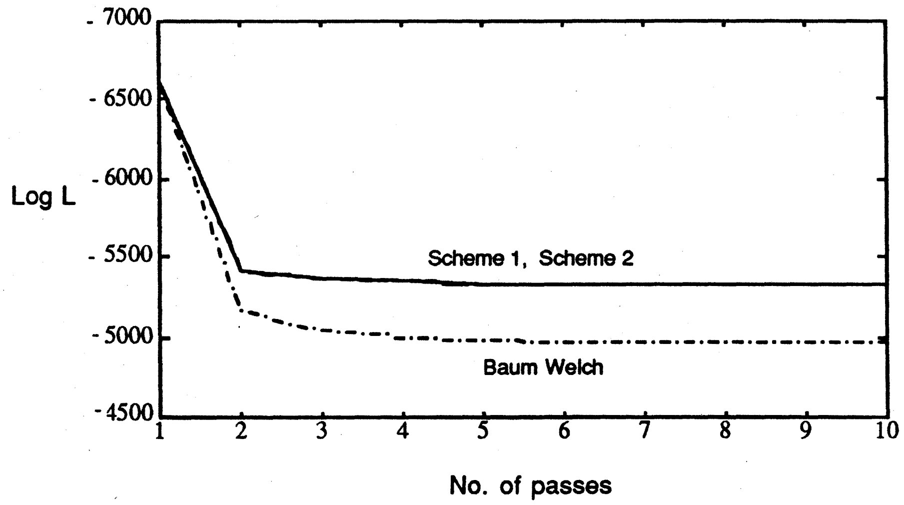
Example 1: In this example we illustrate the performance of the proposed re-estimation schemes.

A 10000 point four-state first order Markov chain was generated with transition probabilities $a_{ii} = 0.97$, with the states at 0, 1, 2 and 3. The resulting Markov chain was exponentially filtered with decay constants $[c_1(1), c_1(2), c_1(3), c_1(4)] = [1.0\ 0.98\ 0.95\ 0.90]$ where $c_1(i)$ are defined in (5.6). Also $c_0(i) = 1$ in (5.6). Zero mean white Gaussian noise of standard deviation $\sigma_w = 0.5$ was added to the exponentially filtered signal.

Fig.5.1 a,b and c show 400 points of the Markov chain s_k, the filtered chain r_k, and the observations y_k respectively. After applying Scheme 2 of Section 5.3.3 for 40 passes, the resulting MAP state estimates $\hat{s}_k$ for the corresponding 400 points is plotted in Fig.5.1d.

A least squares fit on the estimated state levels versus time to the last transition for $1 \leq t_k \leq 10$ yields estimates of the decay constants as [0.991 0.973 0.948 0.901] which are close to the true decay constants. By filtering the estimated Markov chain $\hat{s}_k$ with these estimated decay constants, the estimated filtered chain $\hat{r}_k$ is obtained. In Fig.5.1e is plotted 400 points of $\hat{r}_k$.





initial decay estimates = [1.0, 0.98, 0.95, 0.90] = true decay values

Fig. 5.2: Comparison of re-estimation schemes assuming true decay constants

Example 2: We compare the performance of the Baum-Welch scheme to that of Schemes 1 and 2 presented in Section 5.3.3.

To the exponentially filtered signal of Example 1 was added zero mean white Gaussian noise of standard deviation $\sigma_w = 0.25$.

To establish a benchmark of the best possible performance we started with initial filter (decay) estimates equal to the true values. Fig.5.2 compares the number of state estimate errors per 1000 points and the log likelihood functions $\log L$ (L is defined in (5.25)) with successive passes of the Baum-Welch scheme, and Scheme 1 and Scheme 2 described in Section 5.3.3. Notice that although $\log L$ for the Baum-Welch re-estimation is greater than that for Scheme 1 and 2, the state estimates using Scheme 1 and 2 are superior. This is so because unlike the Baum-Welch re-estimation, in Schemes 1 and 2 the probability mass function is constrained to be Gaussian. Any further increase in the likelihood function compared to that of Scheme 1 and 2 occurs when the re-estimated probability mass function is no longer Gaussian. Thus although the Baum-Welch equations give a better likelihood function, the state estimates are worse compared to Schemes 1, 2.

We then started with initial decay estimates of [0.80.80.80.85]. Fig. 5.3 shows the number of state errors with successive passes. Also the log likelihood is plotted versus successive passes. Observe that Scheme 1 and 2 perform much better in state estimation than the standard Baum-Welch scheme. Also notice that the likelihood function with successive passes in Scheme 2 increases faster than that of Scheme 1 and also the number of state estimation errors are fewer. This illustrates the advantage of Scheme 2 over Scheme 1.

Exponentially Decaying States:

Here we assume that it is apriori known that the Markov states decay exponentially with time according to (5.6). We now illustrate the Quasi-EM algorithm which uses (5.34) and (5.35) with $\overline{c_1}(i)$ and $\overline{\mu}(1,i)$ in the right hand sides of the equations replaced by $c_1(i)$ and $\mu(1,i)$.

A 10000 point two-state first order Markov chain was generated with transition probabilities $a_{ii} = 0.95$, with the states q_i at 1 and 2. The resulting Markov chain was exponentially filtered with decay constants $[c_1(1),\ c_1(2)] = [0.9,\ 0.9]$ where $c_1(i)$ are defined in (5.6). Also $c_0(i) = 1$ in (5.6). Zero mean white Gaussian noise of standard deviation $\sigma_w = 0.5$ was added to the exponentially filtered signal.

Fig.5.4(a), (b) and (c) show 200 points of the Markov chain s_k, the filtered chain r_k, and the observations y_k respectively. The initial state level estimates were arbitrarily chosen as $\overline{q}_1 = 0.4$ and $\overline{q}_2 = 1.4$. Also the initial decay constants were chosen as $[\overline{c}_1(1),\ \overline{c}_1(2)] = [0.8,\ 0.8]$.

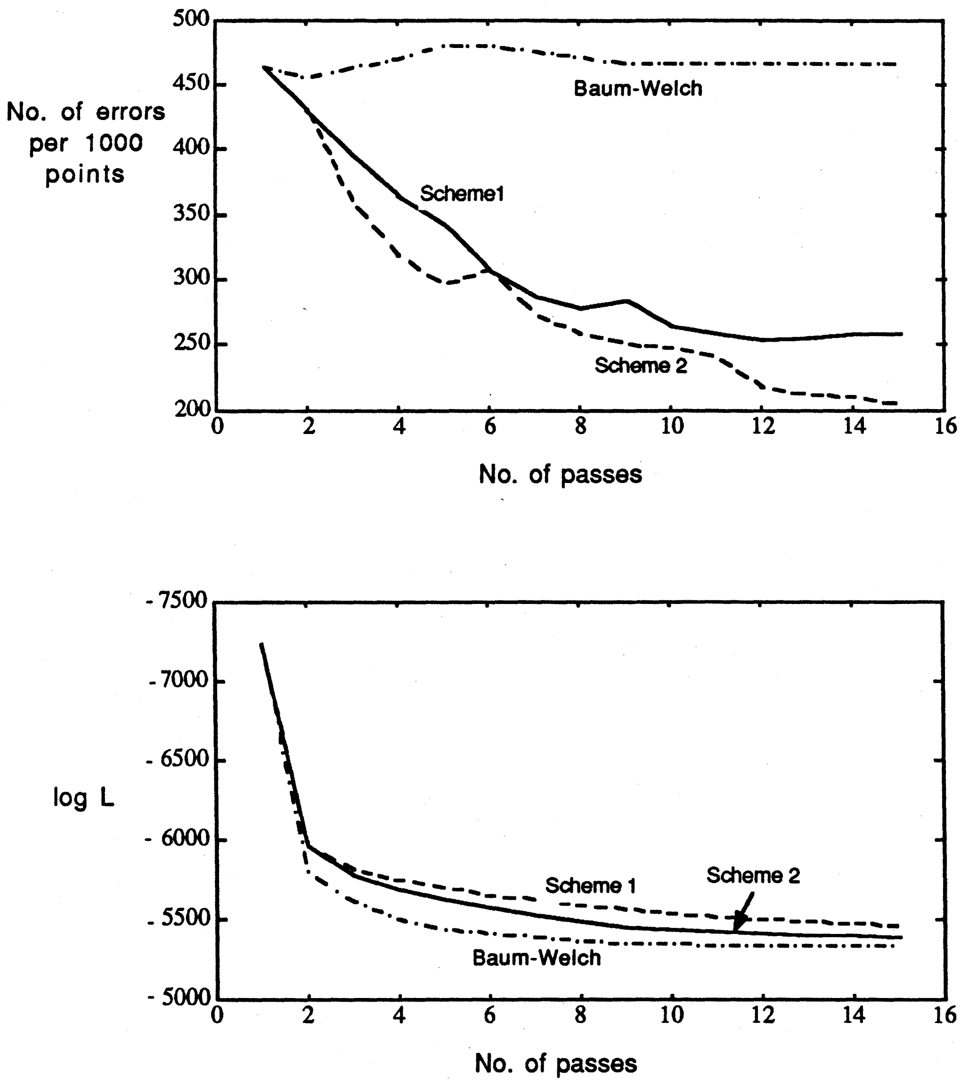
The quasi-EM algorithm was run for 30 passes on the data y_k. Fig. 5.4(d) shows the estimates of the decay constant with successive passes. Fig.5.4(e) shows the estimates of the state levels. Notice that these estimates converge to the true values. Also the transition probability estimates converge to the true values.

This and other simulations not reported here confirm that if it is apriori known that the states decay exponentially, then the EM algorithm can be used to obtain estimates of the decay constant along with the transition probabilities and state levels.

Cell Membrane data

Here we describe an example of a process with exponentially decaying states. The data to model channels in cell membranes was obtained as follows: Spontaneous synaptic currents were recorded in voltage-clamped CA1 cells in rat hippocampal slices at room temperature. The pyramidal neuron was impaled with a microelectrode, which was filled with 3 M KCl, and then its intracellular voltage was clamped at -60 mV with respect to the extracellular medium, using a voltage clamp amplifier. This instrument permits voltage clamping with one electrode which is switched at high frequency between current passing and voltage recording modes. The recorded currents, which show a rapid rise and decays back exponentially, are spontaneous inhibitory post-synaptic currents generated by the opening of chloride-selective channels activated by γ-aminobutyric acid.

The state levels were apriori known to be 0, 50, 84, 184 and the noise variance was $\sigma_w^2 = 196$. Also the 'ground' state 0 was assumed not to decay. Fig.5.5a shows a 400 point segment of the data y_k. Fig.5.5b shows the signal estimate $\hat{v}_k$ for the same segment after 20 passes of Scheme 2 with $\eta = 1.9$ and $N_t = 50$. Fig 5.5c shows the estimated imbedded Markov chain $\hat{s}_k$ for the segment. Using a least squares fit (5.29), the exponential decay constants for the three states (excluding the ground states) were calculated to be 0.91, 0.97, 0.95.



initial decay estimates = [0.8, 0.8, 0.8, 0.85]
true decay values = [1, 0.98, 0.95, 0.95]

Fig. 5.3: Comparison of re-estimation schemes

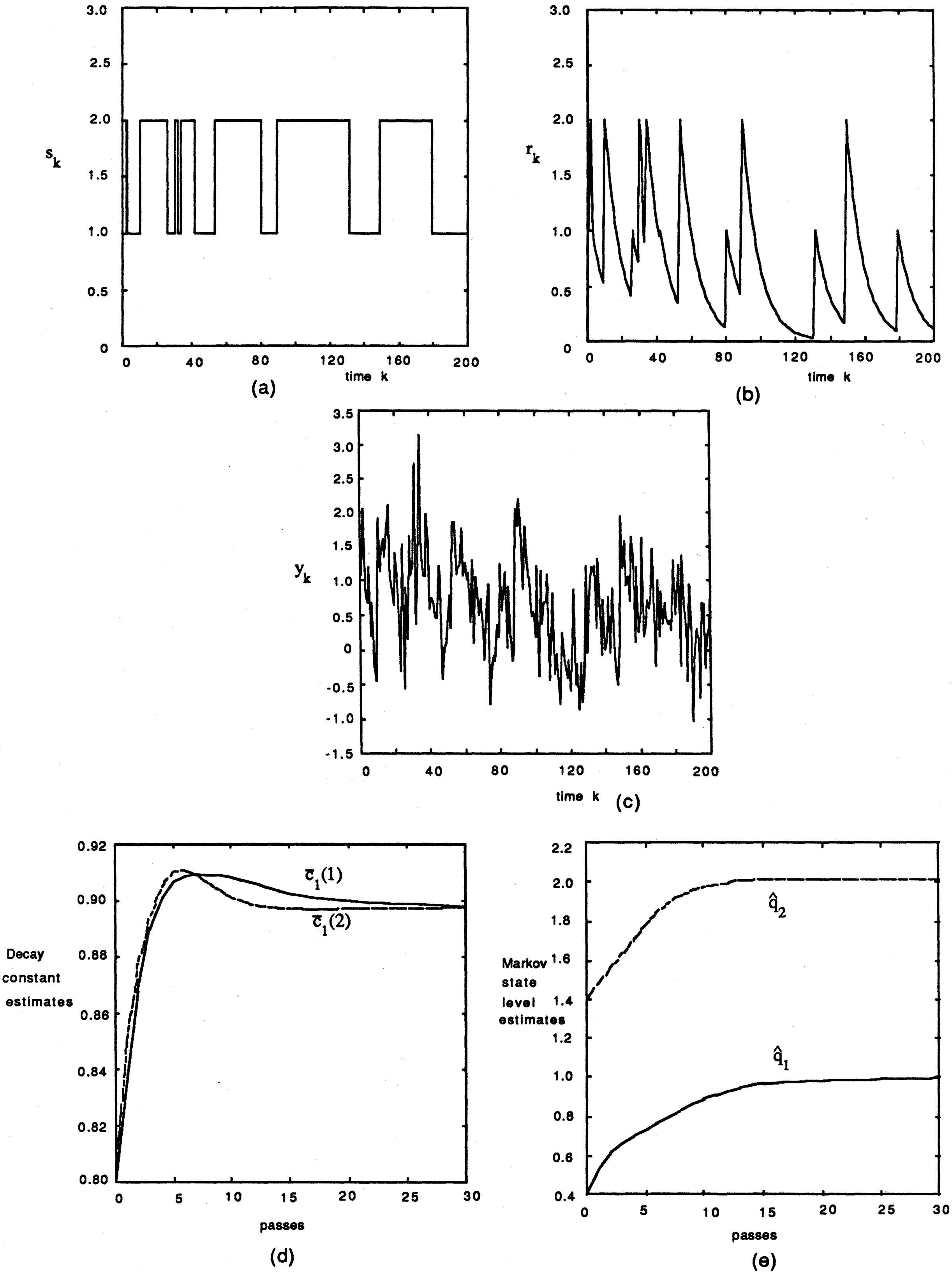


Fig. 5.4 : EM estimation of Markov chain with exponentially decaying states

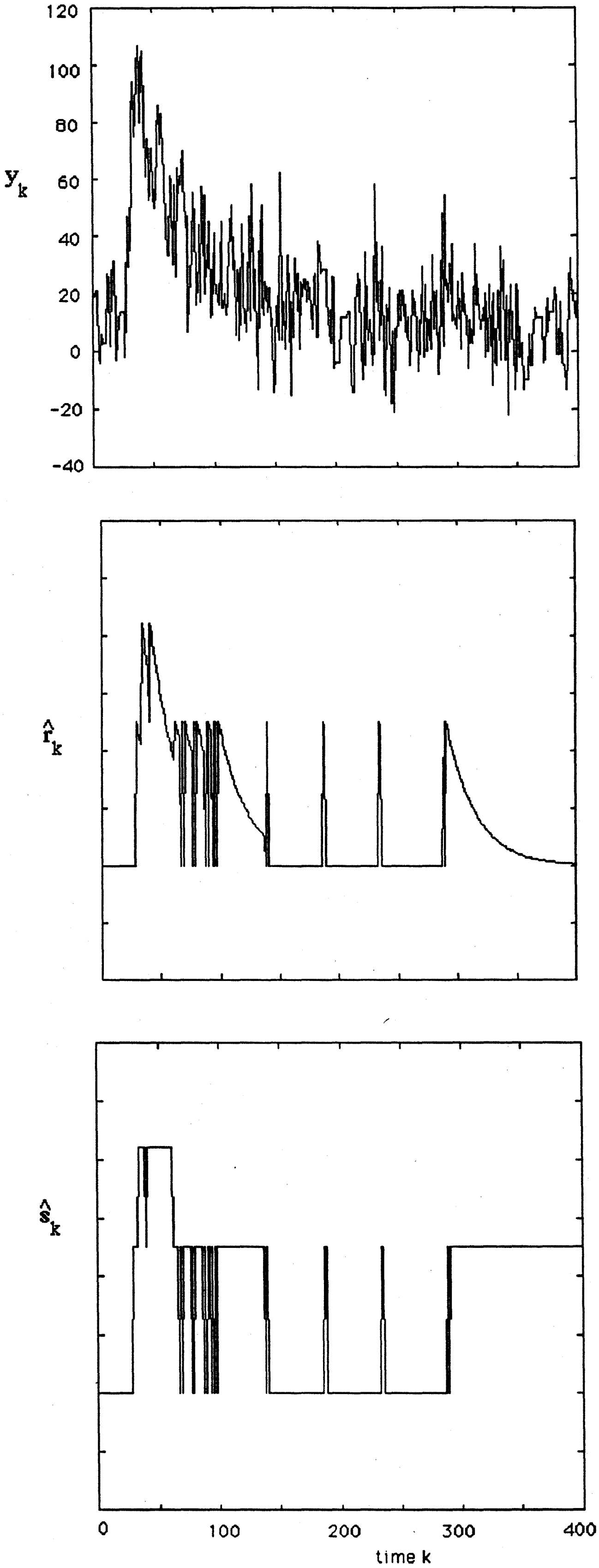


Fig.5.5: Sample segments of noisy observations, estimated signal and underlying Markov process of synaptic neuronal currents

Once the decay constants were estimated, we computed the least squares estimate of $\overline{\mu}_{1,i}$, $i = 2,3,4$. Because the rise time to a state is significantly faster than the decay, it is reasonable to expect that $\overline{\mu}_{1,i}$ are satisfactory estimates of the state levels of the imbedded Markov chain. The means $\overline{\mu}_{1,i}$ were calculated to be 49.8, 90.7, 201.4 which indeed are close to the actual states levels.

5.5 Conclusions

In this chapter schemes for estimating semi-Markov processes imbedded in white (or near white) noise with decaying states have been proposed. We have reformulated the Hidden semi-Markov Model problem in the scalar case as first order 2-vector, homogeneous Hidden Markov Model (HMM) problem in which the state consists of the signal augmented by the time to the last transition. With this reformulation, we have applied HMM signal processing techniques based on the forward-backward algorithm and the Baum-Welch re-estimation formulae. We also have proposed variations of the Expectation Maximization (EM) scheme for faster convergence in yielding optimal estimates of the signals and signal model parameters, including level transition probabilities and noise statistics. Finally, an example from biological signal processing of cell membrane channel currents is studied.

Appendix

E step: This involves computing $\mathcal{Q}(\lambda, \overline{\lambda})$ which is the expectation of the log of the likelihood function of the fully categorized data (see [20], [15] for details). Here

$$\mathcal{Q}(\lambda, \overline{\lambda}) = \sum_{m=1}^{N} \sum_{n=1}^{N} \sum_{k=1}^{T-1} \zeta_k(m,n) \log \overline{a}_{mn} + \sum_{n=1}^{N} \gamma_1(n) \log \overline{\pi}_n + \mathcal{Q}_2 \qquad (A1)$$

where

$$\mathcal{Q}_2 = \sum_{h=1}^{N_t} \sum_{i=1}^{N_s} \sum_{k=1}^{T} \gamma_k(h,i) \log\left(\frac{1}{\sqrt{2\pi}\sigma_w} \exp \frac{-(y_k - \overline{\mu}_{h,1} - (i-1)\overline{\Delta}_h)^2}{2\sigma_w{}^2}\right) \qquad (A2)$$

M step: This involves finding $\overline{\lambda}$ to maximize $\mathcal{Q}(\lambda, \overline{\lambda})$. Maximizing the first two terms on the right hand side of (A1) yields the standard Baum-Welch equations for re-estimating

$\overline{a}_{mn}$ and $\overline{\pi}_m$. Now consider maximizing the third term in (A1) namely Q_2: Solving for

$$\frac{\partial Q_2}{\partial \overline{\mu}_{h,1}} = 0, \; \frac{\partial Q_2}{\partial \overline{\Delta}_h} = 0$$

immediately yield (5.36) and (5.37).

Chapter 6

Hidden Markov Model Signal Processing in presence of unknown deterministic interferences

6.1 Introduction

Hidden Markov Model (HMM) processing techniques are used to extract discrete-time finite-state Markov signals imbedded in Gaussian white noise [21], [52]. Here, motivated by examples in neuro-physical signal processing of ionic currents in cell membrane channels explored in more detail in a companion paper [6], we consider the case where in addition to Gaussian white noise, the Markov process is also corrupted by a deterministic signal of known form but unknown parameters. We consider two such examples of deterministic disturbances:

1. Periodic or almost periodic disturbances, with time variable k, of the form $\sum_{i=1}^{p} a_i \sin(w_i k + \phi_i)$ where the frequency components w_i, amplitudes a_i and phases ϕ_i are unknown. An example of such periodic disturbance with known frequency components (the fundamental and harmonics) is the periodic 'hum' from the electricity mains which may be too costly to eliminate from some experimental environments.

2. Drift in the set of state levels of the Markov process in the form of the polynomial

$\sum_{i=1}^{p} a_i k^i$ with the coefficients a_i unknown. For example, such drift can occur in cell membrane channel current measurements due to the slow development of liquid junction potentials arising from two different solutions of ionic compositions [14]. This drift can be adequately modelled by a polynomial function of time.

It might be argued that the classical problem of periodic disturbance suppression with known frequency components can be effectively solved by notch filtering or averaging methods. However, traditional schemes such as notch filtering have a considerable transient response and so distort the imbedded Markov chain. Also, simulations (not presented here) show that even in low noise, heuristic methods such as averaging observations separated by multiples of the period yield unsatisfactory estimates. Variations of this method involving more complex moving average (or median) filtering to average out the noise do not appear to yield significant improvement.

More recently, in [64], a generalization of the notch filtering approach is presented for eliminating exponential interferences including sinusoids from finite-length discrete-time signals such as step responses. This scheme however, does not exploit the Markovian nature of the imbedded signal. Moreover, such schemes do not learn anything about the periodic disturbance. Thus there is incentive to seek optimal suppression or estimation of the deterministic disturbance. In deriving optimal schemes, all the apriori information available must be used: the nature of the deterministic disturbances (eg. periodic or polynomial), the Markovian characteristics of the imbedded process and the presence of white Gaussian noise.

Methods for estimating a periodic signal in noise are usually highly sensitive to frequency uncertainty. In our experience, estimates of the frequency components of the periodic disturbance obtained by a Fast Fourier Transform (FFT) on the data are not sufficiently accurate, especially in high noise. Hence the need for deriving schemes that achieve optimal frequency estimation of the periodic signal in a mixture of white noise and a finite-state Markov process.

In this chapter, we derive maximum likelihood estimates for the parameters of the deterministic signals, the noise statistics and the transition probabilities, state levels and signal estimates of the imbedded Markov chain. First the problem is formulated as a Hidden Markov Model (HMM) problem and then the Expectation Maximization (EM) [20] algorithm is applied to obtain the maximum likelihood estimates. In the periodic signal case, maximum likelihood estimates of the frequency components, phases and amplitudes are obtained. The algorithms we present here are attractive because of their sound theoretical basis and do not require undue complexity in their implementation. Further, simulation studies confirm their optimal (maximum likelihood) performance.

The chapter is organized as follows: In Section 6.2 the problem is formulated as a HMM problem. In Section 6.3, our techniques for obtaining the maximum likelihood estimates are described. Finally simulation examples are presented in Section 6.4 and conclusions in Section 6.5.

6.2 Problem Formulation

In this section, first the signal model is described and then estimation objectives are presented.

Signal Model: Consider a discrete-time finite-state homogeneous Markov process s_k. At each time k, s_k takes on one of a finite number N of states $q_1, q_2, \dots, q_N$, organized as a vector $\mathbf{q}$. Denote the state transition probabilities as $a_{mn} = P(s_{k+1} = q_n | s_k = q_m)$ and the corresponding transition probability matrix by $\mathbf{A} = (a_{mn})$.

Let $p_k(\Theta)$, parametrized by $\Theta \in \mathbb{R}^R$ denote a deterministic disturbance. We assume that the functional form of p_k is known but the parameter vector $\Theta = (\theta_1, \dots, \theta_R)$ is unknown. eg. in the periodic or almost periodic case, $\Theta = (a_1, \dots, a_p, \phi_1, \dots, \phi_p)$ and $p_k(\Theta) = \sum_{i=1}^{p} a_i \sin(w_i\, k + \phi_i)$. In the polynomial drift case, $\Theta = (a_1, \dots, a_p)$ and $p_k(\Theta) = \sum_{i=1}^{p} a_i\, k^i$.

To make a connection with the theory of HMMs, we assume that the Markov process s_k is hidden, that is indirectly observed by measurements y_k, where

$$y_k = s_k + p_k + w_k. \tag{6.1}$$

Here w_k is zero mean normally distributed Gaussian white noise of variance σ_w^2. We denote the sequence $y_1, y_2, \dots, y_k$ by Y_k. Define the vector of probability functions $\mathbf{b}(\Theta, \sigma_w^2, \mathbf{q}, y_k) = (b_i(\Theta, \sigma_w^2, \mathbf{q}, y_k))$ where

$$b_i(\Theta, \sigma_w^2, \mathbf{q}, y_k) \triangleq P(y_k - p_k(\Theta) | s_k = q_i) = \frac{1}{\sqrt{2\pi}\,\sigma_w} \exp\left(-\frac{(y_k - q_i - p_k(\theta))^2}{2\,\sigma_w^2}\right). \tag{6.2}$$

For notational convenience, in the rest of the chapter we shall denote $b_i(\Theta, \sigma_w^2, \mathbf{q}, y_k)$ by $b_i(y_k)$.

Because w_k is white, the independence property $P(y_k - p_k | s_k = q_i, s_{k-1} = q_j, Y_{k-1}) = P(y_k - p_k | s_k = q_i)$ which is essential for formulating the problem as a HMM holds. Also we assume that the initial state probability vector $\underline{\pi} = (\pi_i)$ is defined from $\pi_i = P(s_1 = q_i)$. The HMM is denoted $\lambda = (\mathbf{A}, \Theta, \sigma_w^2, \mathbf{q}, \underline{\pi})$.

Estimation Objectives: Given the above signal model, there are three inter-related HMM problems which can be solved [21], [52] and interpreted to yield results for the underlying Markov process.

1. Evaluate the likelihood of a given model λ_i generating the given sequence of data, denoted $P(\mathbf{Y}_k | \lambda_i)$. This allows comparison of a set of models $\{\lambda_i\}$ to select the most likely, given the observations.

2. Estimate statistics of the signal s_k such as *a posteriori* probabilities for an assumed model λ and data sequence. For a fixed interval T, estimate

$$\underline{\gamma}_k = (\gamma_k(m)); \quad \gamma_k(m) \triangleq P(s_k = q_m | \mathbf{Y}_T, \lambda). \tag{6.3}$$

From $\gamma_k(m)$, state sequence estimates conditioned on λ, such as maximum a posteriori (MAP) can be generated as

$$\hat{s}_k^{\mathrm{MAP}} = q_m \quad \text{where } m = \arg \max_{1 \le i \le N} \gamma_k(i) \tag{6.4}$$

3. Process the observations based on a model assumption λ, and re-estimate the model parameters (functions) $(\mathbf{A}, \Theta, \sigma_w^2, \mathbf{q}, \underline{\pi})$, such that the likelihood of the observation sequence given the updated model $\overline{\lambda} = (\overline{\mathbf{A}}, \overline{\Theta}, \overline{\sigma}_w^2, \overline{\mathbf{q}}, \overline{\underline{\pi}})$ namely $P(\mathbf{Y}_t | \overline{\lambda})$ is improved, i.e., $P(Y_T | \overline{\lambda}) > P(Y_T | \lambda)$, and repeating until convergence. The objective is to achieve the most likely model λ^{ML} amongst the set $\lambda = (\mathbf{A}, \Theta, \sigma_w^2, \mathbf{q}, \underline{\pi})$, given the observations.

6.3 Estimation and Re-estimation

Here we present solutions to the three problems formulated above.

6.3.1 Forward-Backward Procedure

The solution to Problems 1 and 2 involve the Forward-Backward procedure which we briefly describe now:

Consider an observation sequence $\mathbf{Y}_T$ and assumed signal generating model λ. Forward and backward vector variables are defined as probabilities conditioned on the model λ, for $m = 1, 2, \ldots, N$ as follows:

$$\begin{aligned} \underline{\alpha}_k &= (\alpha_k(m)); \quad \alpha_k(m) \triangleq P(\mathbf{Y}_k, s_k = q_m|\lambda) \\ \underline{\beta}_k &= (\beta_k(m)); \quad \beta_k(m) \triangleq P(\overline{\mathbf{Y}}_k|s_k = q_m, \lambda) \end{aligned} \tag{6.5}$$

where $\overline{\mathbf{Y}}_k$ denotes the future sequence $y_{k+1}, y_{k+2}, \ldots, y_T$.

Recursive formulae for $\underline{\alpha}_k$ and $\underline{\beta}_k$ are readily calculated [21] as

$$\begin{aligned} \alpha_k(n) &= \left(\sum_{m=1}^{N} \alpha_{k-1}(m)\, a_{mn} \right) b_n(y_k), \quad \alpha_1(n) = \pi_n\, b_n(y_1) \\ \beta_k(m) &= \sum_{n=1}^{N} a_{mn}\, b_n(y_{k+1})\, \beta_{k+1}(n), \quad \beta_T(m) = 1 \end{aligned} \tag{6.6}$$

The optimal *a posteriori* probabilities associated with Problem 2 are given from [21]

$$\underline{\gamma}_k = (\gamma_k(m)); \quad \gamma_k(m) = \frac{\alpha_k(m)\, \beta_k(m)}{\sum_{m=1}^{N} \alpha_k(m)\beta_k(m)}, \quad m = 1, 2, \ldots, N. \tag{6.7}$$

Then using (6.4) we obtain MAP fixed-interval smoothed estimates. The likelihood function, which is the end result of the Problem 1, is calculated as

$$L_T \triangleq P(\mathbf{Y}_T|\lambda) = \sum_{m=1}^{N} \alpha_T(m). \tag{6.8}$$

6.3.2 EM algorithm

The solution of Problem 3 involves the EM algorithm. It is an iterative algorithm; each iteration consists of the Expectation step (E step) and the Maximization step (M step).

E step: This involves computing $\mathcal{Q}(\lambda, \overline{\lambda})$ which is the expectation of the log of the likelihood function of the fully categorized data (see [20], [15] for details). Here

$$\mathcal{Q}(\lambda, \overline{\lambda}) = \sum_{m=1}^{N} \sum_{n=1}^{N} \sum_{k=1}^{T-1} \zeta_k(m,n) \log \overline{a}_{mn} + \sum_{n=1}^{N} \gamma_1(n) \log \overline{\pi}_n + \mathcal{Q}_2 \tag{6.9}$$

where

$$\mathcal{Q}_2 = \sum_{k=1}^{T-1} \sum_{n=1}^{N} \gamma_k(n) \log \frac{1}{\sqrt{2\pi}\overline{\sigma}_w} \exp\left(-\frac{(y_k - (\overline{q}_n + p_k(\overline{\Theta})))^2}{2\overline{\sigma}_w^2}\right) \tag{6.10}$$

$$\zeta_k(m,n) \triangleq P(s_k = q_m, s_{k+1} = q_n, \mathbf{Y}_T | \lambda) = \alpha_k(m)\, a_{mn}\, b_n(y_{k+1})\, \beta_{k+1}(n) \tag{6.11}$$

and $\overline{q}_n$ is the estimate of the Markov state level q_n.

M step: This involves finding $\overline{\lambda}$ to maximize $\mathcal{Q}(\lambda, \overline{\lambda})$. The optimal HMM λ is loosely denoted $\overline{\lambda}$. Maximizing the first two terms on the right hand side of (6.9) yields the standard Baum-Welch equations [17], [18], [19] for re-estimating $\overline{a}_{mn}$ and $\overline{\pi}_m$, namely,

$$\overline{a}_{mn} = \frac{\sum_{k=1}^{T-1} \zeta_k(m,n)}{\sum_{k=1}^{T-1} \sum_{n=1}^{N} \zeta_k(m,n)}; \quad \overline{\pi}_m = \gamma_1(n) \tag{6.12}$$

Also solving for $\frac{\partial \mathcal{Q}_2}{\partial \overline{q}_n} = 0$ yields the new estimate $\overline{q}_n$ as

$$\overline{q}_n = \frac{\sum_{k=1}^{T-1} \gamma_k(n)(y_k - p_k(\overline{\Theta}))}{\sum_{k=1}^{T-1} \gamma_k(n)} \tag{6.13}$$

Solving for $\frac{\partial \mathcal{Q}_2}{\partial \overline{\sigma}_w^2} = 0$ gives

$$\overline{\sigma}_w^2 = \frac{1}{T} \sum_{k=1}^{T-1} \sum_{n=1}^{N} \gamma_k(n)(y_k - \overline{q}_n - p_k(\overline{\Theta}))^2 \tag{6.14}$$

We also require to solve for $\overline{\theta}_i$ in $\frac{\partial \mathcal{Q}_2}{\partial \overline{\theta}_i} = 0$ to maximize $\mathcal{Q}_2$. Details are developed for periodic and polynomial drift disturbances in the next two subsections.

6.3.3 Periodic Disturbances

To fix ideas, we first consider the case where the frequency components of the periodic (or almost periodic) disturbance are known. Later in this subsection, the more general case with unknown frequency components will be considered.

Here $\Theta = (a_1, \ldots, a_p, \phi_1, \ldots, \phi_p)$ and $p_k(\Theta) = \sum_{i=1}^{p} a_i \sin(w_i\, k + \phi_i)$ where $0 < \phi_i \leq \pi$,

$a_i \in \mathbf{R}$. Also $w_i = 2\pi/T_i$ where T_i is the period of the i th sinusoidal component.

Solving for $\frac{\partial Q_2}{\partial \overline{a}_i} = 0$ yields the new estimate

$$\overline{a}_i = \frac{\sum_k \sum_n \gamma_k(n)(y_k - \overline{q}_n - \sum_{j \neq i} \overline{a}_j \sin(w_j k + \overline{\phi}_j)) \sin(w_i k + \overline{\phi}_i)}{\sum_k \sin^2(w_i k + \overline{\phi}_i)} \tag{6.15}$$

Also, solving $\frac{\partial Q_2}{\partial \phi_i} = 0$ yields

$$c_1 \cos \overline{\phi}_i + c_2 \sin \overline{\phi}_i = d_1 \cos 2\overline{\phi}_i + d_2 \sin 2\overline{\phi}_i \tag{6.16}$$

where

$$c_1 = \sum_k \sum_n \gamma_k(n)(y_k - \overline{q}_n - \sum_{j \neq i} \overline{a}_j \sin(w_j k + \overline{\phi}_j)) \cos(w_i k)$$

$$c_2 = -\sum_k \sum_n \gamma_k(n)(y_k - \overline{q}_n - \sum_{j \neq i} \overline{a}_j \sin(w_j k + \overline{\phi}_j)) \sin(w_i k)$$

$$d_1 = \frac{\overline{a}_i}{2} \sum_k \sin 2\, w_i k; \quad d_2 = \frac{\overline{a}_i}{2} \sum_k \cos 2\, w_i k$$

Based on (6.13), (6.14), (6.15) and (6.16) we now propose three methods for re-estimating $\overline{q}_n$, $\overline{\sigma}^2_w$, $\overline{a}_i$ and $\overline{\phi}_i$.

1.**Standard EM Re-estimation**: To solve for $\overline{q}_n$, $\overline{\sigma}^2_w$, $\overline{a}_i$ and $\overline{\phi}_i$, we need to solve (6.13), (6.14), (6.15) and (6.16) simultaneously. This is easily done (although the algebraic manipulation is tedious) as follows: From (6.15) and (6.13), $\overline{a}_i$ and $\overline{q}_n$ can be expressed as explicit functions of $\overline{\phi}$. Then substituting for these in (6.16), we solve for (6.16) numerically for $\overline{\phi}$ and so calculate $\overline{q}_n$, $\overline{\sigma}^2_w$ and $\overline{a}_i$.

2.**Block Component Re-estimation** [67]: Here we iteratively maximize Q_2 with respect to one variable at a time keeping the others constant. Thus we keep a_i, ϕ_i and σ_w constant and update q_n. Then we use the updated q_n, keep ϕ_i and σ_w constant and re-estimate a_i and so on. This scheme ensures that the likelihood function improves after each pass. Thus in each pass one of the equations (6.13), (6.14), (6.15), (6.16) is used with $\overline{q}_n$, $\overline{\sigma}^2_w$, $\overline{a}_i$ and $\overline{\phi}_i$ in the right hand sides of the equations replaced by q_n, σ^2_w, a_i and ϕ_i. Note that the transition probabilities are updated every pass since their re-estimation equations are independent of maximizing Q_2.

3.**Quasi-EM Re-estimation**: We now propose an alternative scheme which is not strictly EM, but the resulting equations are much simpler. The scheme we propose is similar to that in [65], [66]. In re-estimating $\overline{q}_n$, $\overline{\sigma}_w^2$, $\overline{a}_i$ and $\overline{\phi}_i$, we use (6.13), (6.14), (6.15) and (6.16) with $\overline{q}_n$, $\overline{\sigma}_w^2$, $\overline{a}_i$ and $\overline{\phi}_i$ in the right hand sides of the equations replaced by q_n, σ_w^2, a_i and ϕ_i. Then because c_1, c_2, d_1 and d_2 are independent of $\overline{\phi}_i$, (6.16) is easily solved for $\overline{\phi}_i$ numerically. The simulations we present subsequently use this quasi-EM algorithm and show that the algorithm yields acceptable results.

Frequency Re-estimation:

So far we have assumed that the frequency of the components of the periodic signal are known precisely. Here we deal with the case when the frequency components are unknown.

It might seem feasible to estimate the frequency components by applying a Fast Fourier Transform (FFT) on the data. However FFTs are not very accurate especially in high noise. We show subsequently in simulation studies that the performance of the above proposed schemes rapidly degrade even with small errors in the frequency. So the FFT is not suitable for obtaining estimates of the frequency components.

We now propose a scheme for re-estimating the frequencies which is an extension of the EM scheme proposed above for estimating the amplitudes and phases.

Here because the frequency components are also unknown, the parameter vector to be estimated is $\Theta = (a_1, \ldots, a_p, \phi_1, \ldots, \phi_p, w_1, \ldots, w_p)$. Again we maximize Q_2 defined in (6.10) with w_i in Q_2 now replaced by $\overline{w}_i$. Setting the derivatives of Q_2 with respect to $\overline{q}_n$, $\overline{\sigma}_w^2$, $\overline{a}_i$ and $\overline{\phi}_i$ to zero yield (6.13), (6.14), (6.15) and (6.16) respectively. Setting $\partial Q_2 / \partial \overline{w}_i = 0$ yields

$$f_i(\overline{\mathbf{w}}) \triangleq \sum_{k=1}^{T} h_k \cos(\overline{w}_i\, k+\overline{\phi}_i) - \overline{a}_i\, k \sin(2(\overline{w}_i\, k+\overline{\phi}_i)) - k \sum_{j\neq i} \overline{a}_j\, \sin(\overline{w}_j\, k+\overline{\phi}_j) \cos(\overline{w}_i\, k+\overline{\phi}_i) = 0. \tag{6.17}$$

where $h_k = k(y_k - \sum_{n=1}^{N} \gamma_k(n)\, \overline{q}_n)$ and $\overline{\mathbf{w}} = (\overline{w}_1, \ldots, \overline{w}_p)^T$.

Equation (6.17) has numerous solutions. The solution which we choose to yield a maximum likelihood estimate is that which maximizes Q_2. Of course if intervals in which the various frequency components occur are apriori known (these intervals can be found from a FFT), then only the solutions to (6.17) belonging to these intervals need to be considered.

We propose solving (6.17) numerically using the Newton-Raphson method. Define

$\mathbf{F}(\overline{\mathbf{w}}) = (f_1(\overline{\mathbf{w}}), \ldots, f_p(\overline{\mathbf{w}})$. Then the Newton Raphson update equation is

$$\overline{\mathbf{w}}^{(n+1)} = \overline{\mathbf{w}}^{(n)} - \mathbf{G}^{-1}\,\mathbf{F}(\overline{\mathbf{w}}^{(n)}) \tag{6.18}$$

where $\mathbf{G} = (g_{ij})$, $g_{ij} = \partial f_i / \partial \overline{w}_j$. So

$$\begin{aligned} g_{ii} &= -\sum_k k\, h_k \sin(\overline{w}_i\, k + \overline{\phi}_i) - \overline{a}_i k^2 \cos(2(\overline{w}_i\, k + \overline{\phi}_i)) + k^2 \sum_{j \neq i} \overline{a}_j \sin(\overline{w}_j\, k + \overline{\phi}_j) \sin(\overline{w}_i\, k + \overline{\phi}_i) \\ g_{ij} &= -\overline{a}_j \sum_k k^2 \cos(\overline{w}_i\, k + \overline{\phi}_i) \cos(\overline{w}_j\, k + \overline{\phi}_j) \end{aligned} \tag{6.19}$$

Remark 1: In the implementation of the above scheme, it is necessary to store the T scalars, h_k, $k = 1, \ldots, T$. So the memory requirements here, $(N+1)T$, are only slighly more than that for standard HMM processing (NT).

Remark 2: Any one of the EM, quasi-EM or Block Component Re-estimation can be used.

Remark 3: Equation (6.17) is rather illconditioned. However, as demonstrated in simulations subsequently, double precision arithmetic appears adequate for its numerical solution.

○

6.3.4 Drift

Here $\Theta = (a_1, \ldots, a_p)$ and $p_k(\Theta) = \sum_{i=1}^p a_i\, k^i$. Then $\frac{\partial Q_2}{\partial a_i} = 0$ yields

$$\sum_k \sum_n \gamma_k(n)(y_k - \overline{q}_n) k^i = \sum_k p_k(\overline{\Theta})\, k^i,\ i = 1, \ldots, p \tag{6.20}$$

Solving these p equations which are linear in $\overline{a}_i$ yield $\overline{a}_i$. Again any one of the EM, quasi-EM or Block Component Re-estimation schemes can be used.

6.4 Simulation Studies

We illustrate the performance of the proposed algorithms for periodic disturbances and polynomial drift.

6.4.1 Periodic Signals

We present six simulation examples to illustrate the performance of the proposed re-estimation schemes.

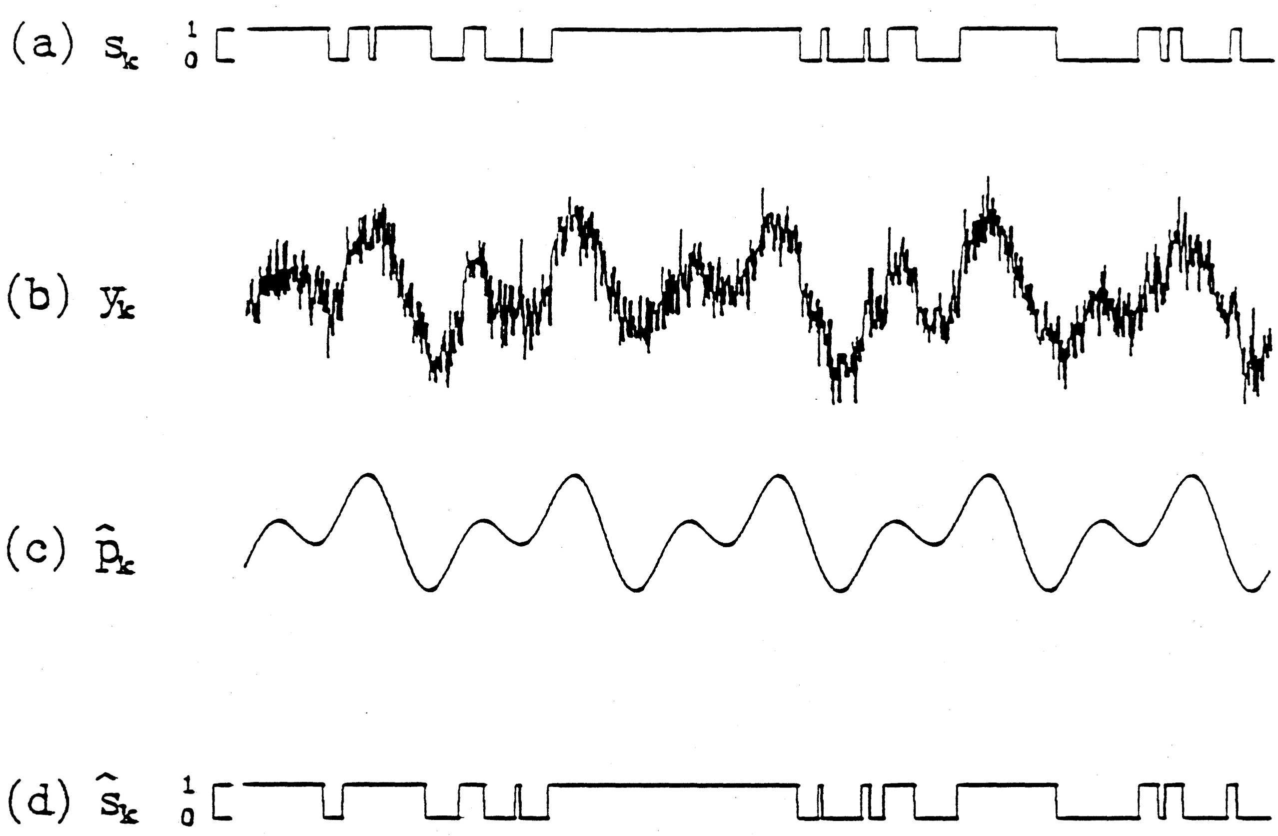


Fig. 6.1 : Performance of quasi-EM algorithm, $\sigma_w = 0{\cdot}5$

Example 1 (Medium noise): The purpose of this simulation example is to show that the proposed algorithms satisfactorily learn the Markov state levels, state estimates and transition probabilities, and also amplitude and phase of the components of the periodic signal (with known frequency components) from the sum of the Markov signal, periodic signal and Gaussian white noise with known variance.

A 20,000-point 2 state discrete-time Markov chain was generated with transition probabilities $a_{ii} = 0.97$, $a_{ij} = 0.03$. The state levels of the Markov process were $q_1 = 0$ and $q_2 = 1$. To this chain was added zero-mean white Gaussian noise of known standard deviation $\sigma_w = 0.5$. Also a periodic signal $p_k = a_1 \sin(2\pi k/200 + \phi_1) + a_2 \sin(2\pi k/100 + \phi_2)$ was added with $\Theta = (a_1, a_2, \phi_i, \phi_2) = (0.8, 0.8, 0, 0)$. The initial Markov state level estimates were taken as $\overline{q}_1 = 1.5$, $\overline{q}_2 = 2.5$. Also the initial estimate $\overline{\Theta} = (0.75, 0.75, 0.50.5)$ and $\overline{a}_{ii} = 0.95$. Fig.6.1a and 6.1b show a 1000-point sample of the Markov chain and observations (obtained by adding the Markov chain, periodic signal and white noise).

After 500 passes, $\overline{q}_1 = -0.011$ and $\overline{q}_2 = 0.997$ and $\overline{a}_{11} = \overline{a}_{22} = 0.969$. Also after 500 passes $\overline{\Theta} = (1.002, 0.994, 3.95 \times 10^{-3}, 2.22 \times 10^{-6})$. Using these values of $\overline{\Theta}$ we plotted 1000 points of the estimated periodic signal $\hat{p}_k$ in Fig.6.1c. In the scale of Fig.6.1, the plots of $\hat{p}_k$ and p_k are identical. Finally, Fig.6.1d shows the corresponding 1000 points of the estimated Markov signal $\hat{s}_k$. Here MAP estimates (see (6.4)) were used.

The plots in Fig.6.1 show that the proposed scheme perform adequately in a 'medium' noise environment.

Example 2 (High noise): This simulation example shows that the proposed algorithms perform satisfactorily even when the Gaussian noise has a large known variance.

To the Markov chain in Example 1 was added zero-mean white Gaussian noise of known standard deviation $\sigma_w = 4$. Also a periodic signal $p_k = a_1 \sin(2\pi k/200 + \phi_1) + a_2 \sin(2\pi k/100 + \phi_2)$ was added where $\Theta = (a_1, a_2, \phi_i, \phi_2) = (0.8, 0.8, \pi/4, 0)$. Fig.6.2a shows a 1000-point sample of the Markov chain, Fig.6.2b shows the periodic signal and Fig.6.2c shows the observations obtained by adding the Markov chain, periodic signal and white Gaussian noise.

Arbitrary initial state level estimates of -0.4 and 0.3 and transition probabilities of 0.95 were assumed. Also the initial estimate of $\overline{\Theta} = (0.4, 0.4, 0.5, 0.5)$. Starting with these initial estimates, the quasi-EM algorithm was run for 500 passes. Fig.6.2d shows one period of the initial estimate of the periodic signal, the estimate after 1 and 2 passes.

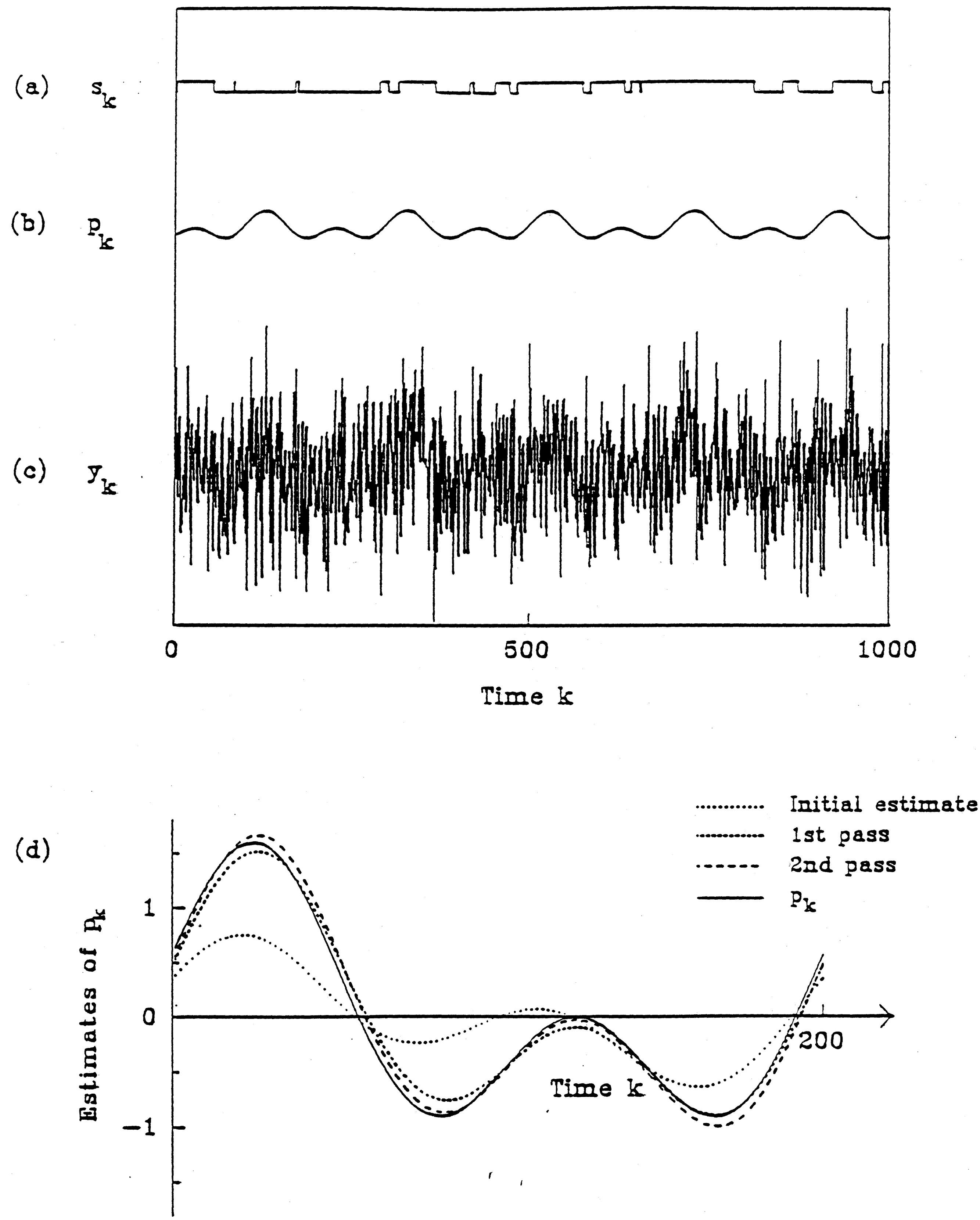


Fig. 6.2: Performance of Quasi-EM algorithm. $\sigma_w = 4$

Also one period of true periodic signal is shown for comparison. Fig.6.2d shows that even after 2 passes the estimated signal is close to the true periodic signal. After 500 passes, $\overline{\Theta} = (0.851, 0.820, 0.711, 2.19 \times 10^{-3})$. With the scale of Fig.6.2d, the estimated periodic signal after 500 passes is indistinguishable from that of the true periodic signal. Also after 500 passes $a_{11} = 0.9654$, $a_{22} = 0.9661$ and the state level estimates are -4.21×10^{-2} and 0.956. Obviously as expected, due to the large noise variance, the state estimates of the Markov chain are not as good as that in the low noise case. However, simulations not presented here show that the number of errors in the MAP state estimates of the Markov chain are very close to the case when there is no periodic disturbance present.

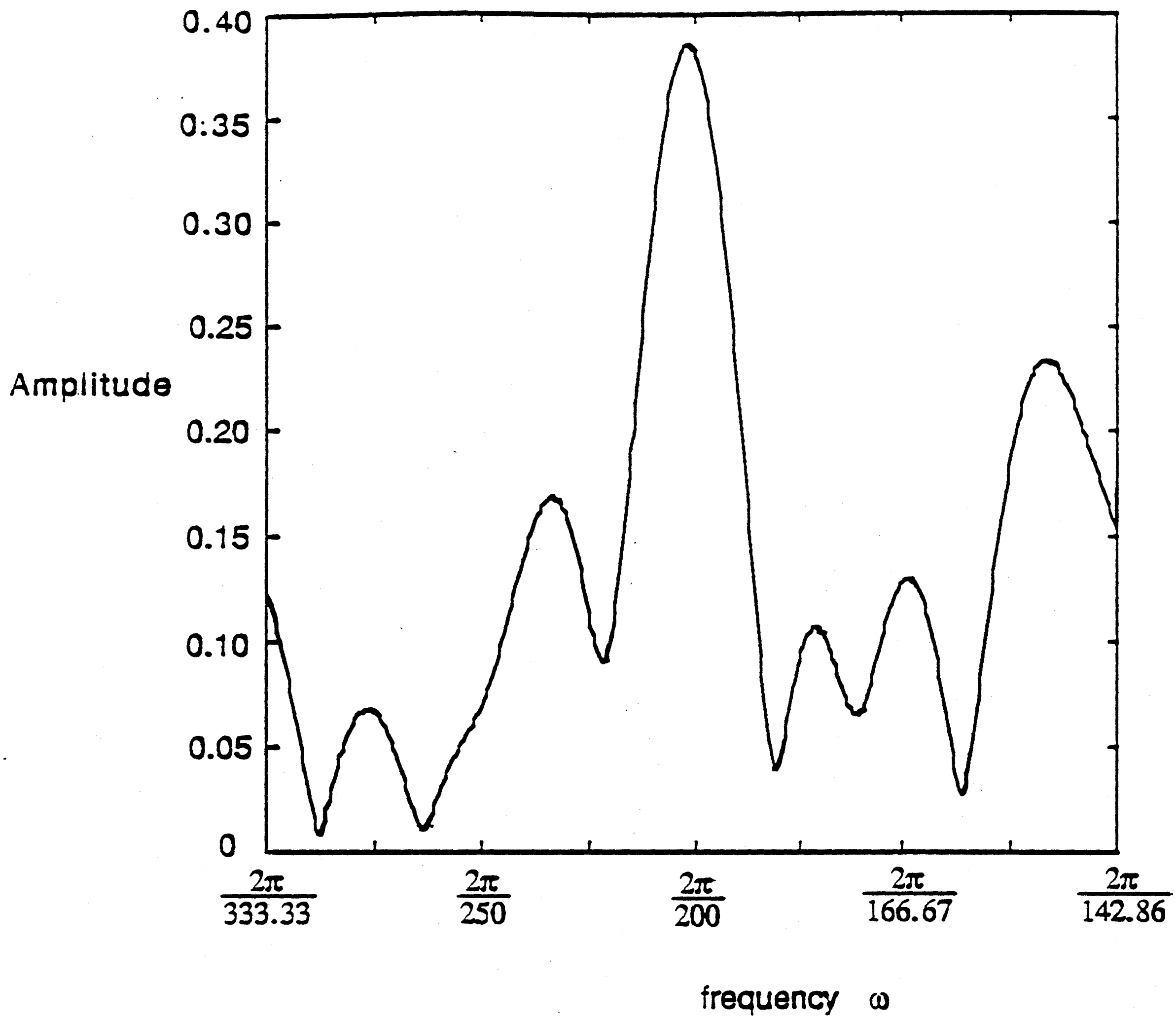
Example 3 (Frequency Sensitivity): Here it is shown that using an FFT yields relatively inaccurate frequency estimates. Also using these inaccurate estimates with our proposed scheme (without frequency re-estimation) yields unsatisfactory results. In Example 4 we demonstrate that our proposed frequency re-estimation scheme yields satisfactory estimates.

To the 20,000-point Markov chain of Example 1, was added zero mean white Gaussian noise with $\sigma_w = 4$ and a periodic signal $p_k = a \sin(2\pi i/200 + \phi)$ where $a = 0.8$, $\phi = \pi/4$. The same initial estimates of Example 1 were used for the Markov process. For the sinusoid, initial estimates were 0.5, 0.5 for the phase and amplitude respectively. A zoom FFT with a frequency resolution of $2\pi/10^5$ was taken (see Fig.6.3) and the period estimate obtained was 201.21. So the period estimate for re-estimation was taken to be 201.21.

We ran our re-estimation scheme (without frequency re-estimation) with these initial parameters. The state level estimates obtained after 500 passes are -0.27 and 1.10, the estimated amplitude and phase of the sinusoid are -0.84 and -0.56. Clearly these are not satisfactory.

Example 4 (Frequency Re-estimation): The proposed frequency re-estimation scheme for the data of Example 3 is now illustrated. We use the block component re-estimation scheme with frequency updates every odd pass and updates of amplitude, phase and Markov levels every even pass. Also double precision arithmetic is used in the numerical solution of (6.17). We start with an initial period estimate of 100.0.

For convenience, define a 'normalized' version of Q_2 as $J \triangleq -\sum_{k=1}^{T} a^2 \sin^2(w\,k + \phi) + 2a(\sum_{n=1}^{N} \gamma_k(n)\, q_n - y_k) \sin(w\,k + \phi)$. It is easily shown that for this example, maximizing J with respect to w is equivalent to maximizing Q_2 with respect to w.



FFT frequency resolution = $\frac{2\pi}{10^5}$

Fig. 6.3: Zoom FFT amplitude spectrum to estimate sinusoidal frequency

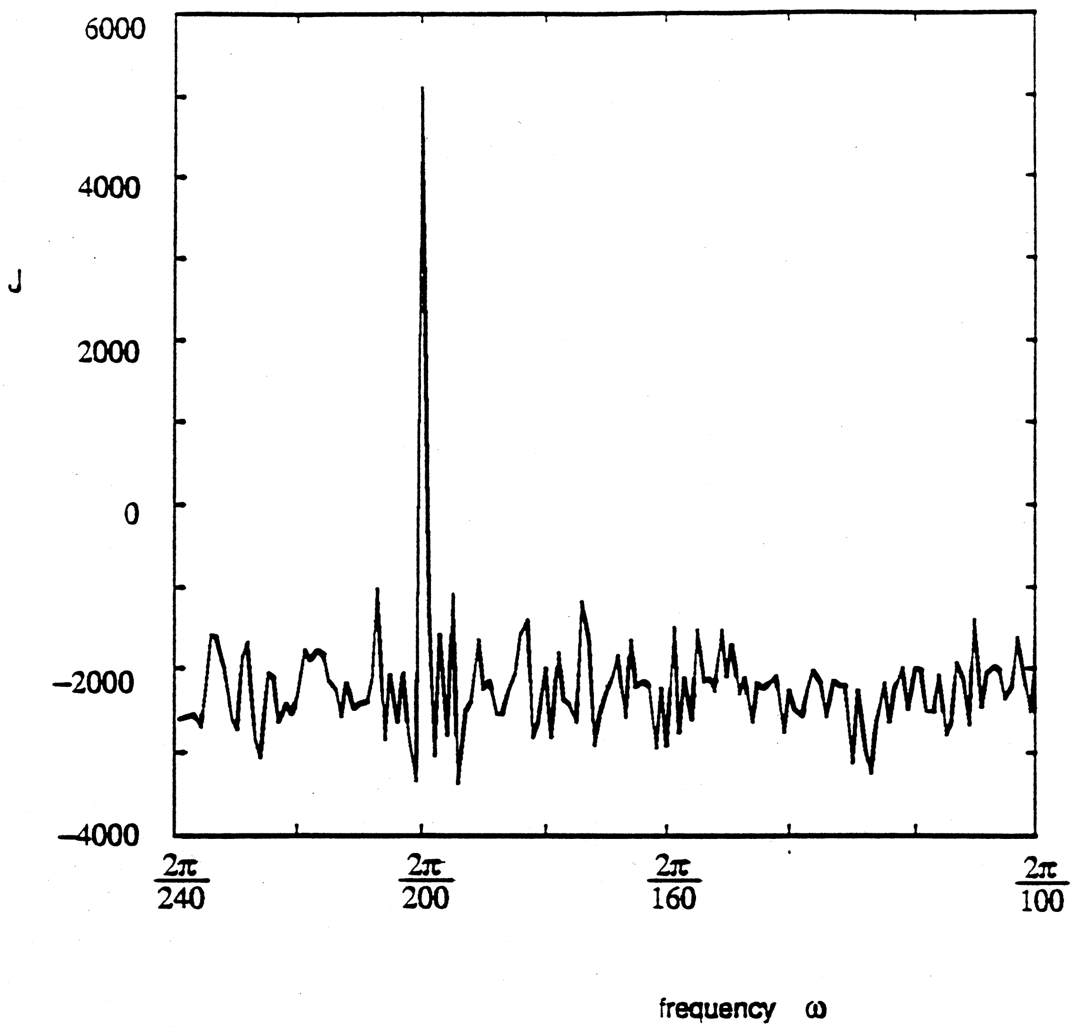


Fig. 6.4: Frequency Re-estimation performance in first pass

Fig.6.4 shows a plot of J versus frequency in the first pass. It illustrates that J and hence Q_2 have a large number of local maxima. Thus (6.17) has a large number of solutions. The maximum likelihood frequency estimate is obtained by finding the global maximum from this set of local maxima. Notice the prominent global maximum of J or Q_2 occurs close to to $w = 2\pi/200$, infact the maximum is at $w = 2\pi/199.78$. The fact that the global maxima is prominent and very close to the true frequency even in the first pass shows the attractiveness of the re-estimation scheme.

With subsequent passes the estimates get even closer to the true values. After 500 passes the period estimate is 199.943. Also the estimated amplitude and phase are 0.8515 and 0.62 respectively. The Markov state level estimates are -2.15×10^{-2}, 1.02 and the transition probability estimates are $\overline{a}_{11} = 0.9678$, $\overline{a}_{22} = 0.9685$. Thus the frequency estimate approaches the true value and also the estimates of the amplitude, phase, levels and transition probabilities are satisfactory.

Example 5 (Detection problem): Given noisy observations, an important detection problem is to detect the presence or absence of a Markov signal. In this example, we illustrate the performance of our proposed schemes in the absence of a Markov signal, i.e, the observations are obtained by adding white Gaussian noise to a periodic signal.

To $p_k = a_1 \sin(2\pi k/200 + \phi_1) + a_2 \sin(2\pi k/100 + \phi_2)$, where $\Theta = (a_1, a_2, \phi_1, \phi_2) = (1.0, 1.0, 0, 0)$ was added Gaussian zero mean white noise with $\sigma_w = 4$, yielding observations y_k. Fig.6.5a and b show 1000 points of p_k and y_k respectively.

As might be expected, using a standard HMM processing yields unsatisfactory estimates of the levels of the Markov process (which are actually zero). Fig. 6.5c shows the Markov state level estimates using standard HMM processing.

We then ran 500 passes of our re-estimation scheme on the data. Fig.6.5d shows the estimates of the Markov state levels which approach zero. Also the estimate $\overline{\Theta} = (1.066, 1.025, 0.033, -0.037)$ which is very close to Θ.

Thus the proposed schemes can be satisfactorily used to detect the absence as well as presence of a Markov signal.

Example 6 (Biological Example): We illustrate the use of the re-estimation algorithms for the analysis of transmembrane ionic currents. A patch of membrane containing a single ion channel was excised from a neuron and a transmembrane potential of -60 mV was applied. Upon the application of 10μM γ-aminobutyric acid, the channel opens and

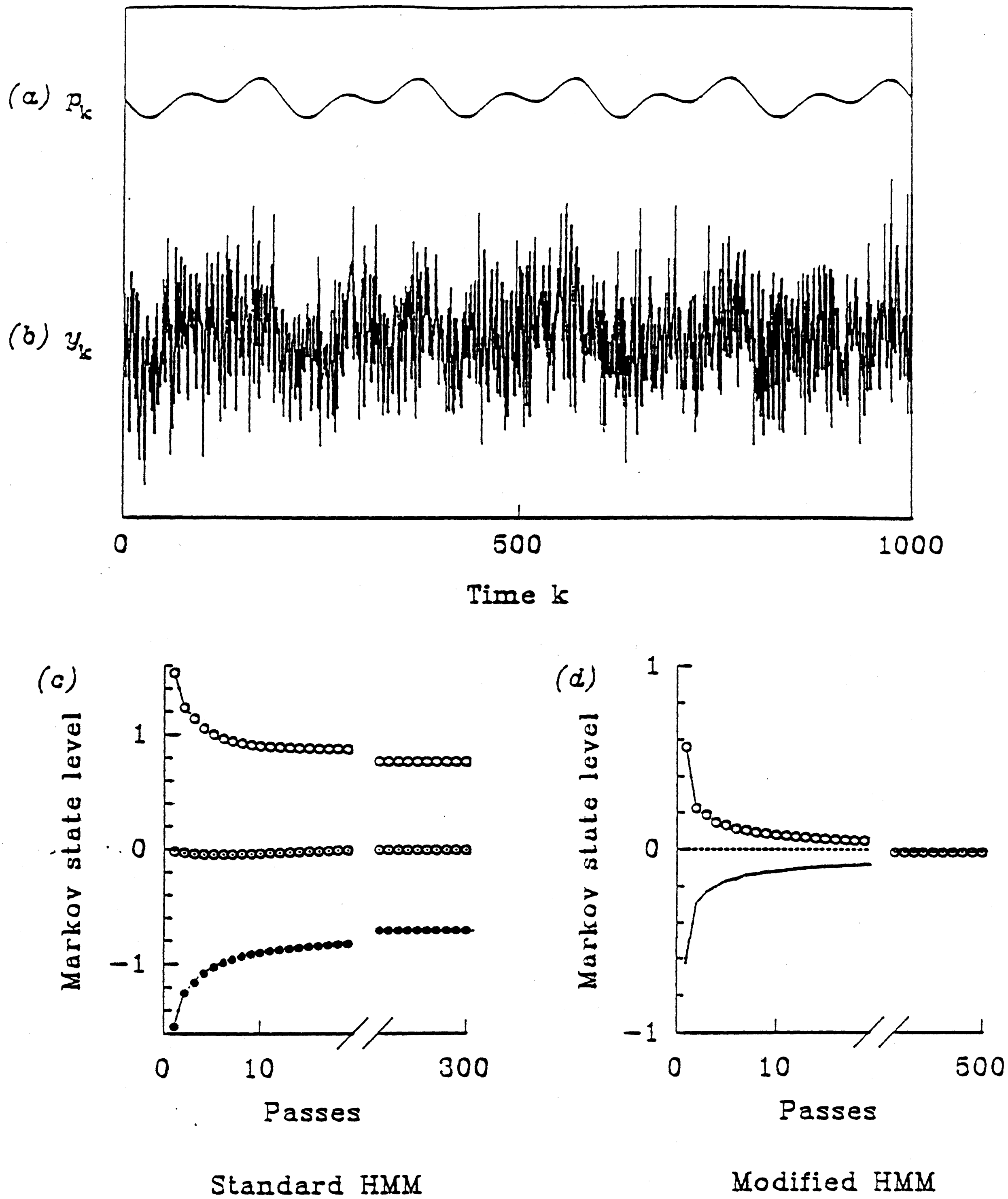


Fig. 6.5 : Detection of absence of Markov signal

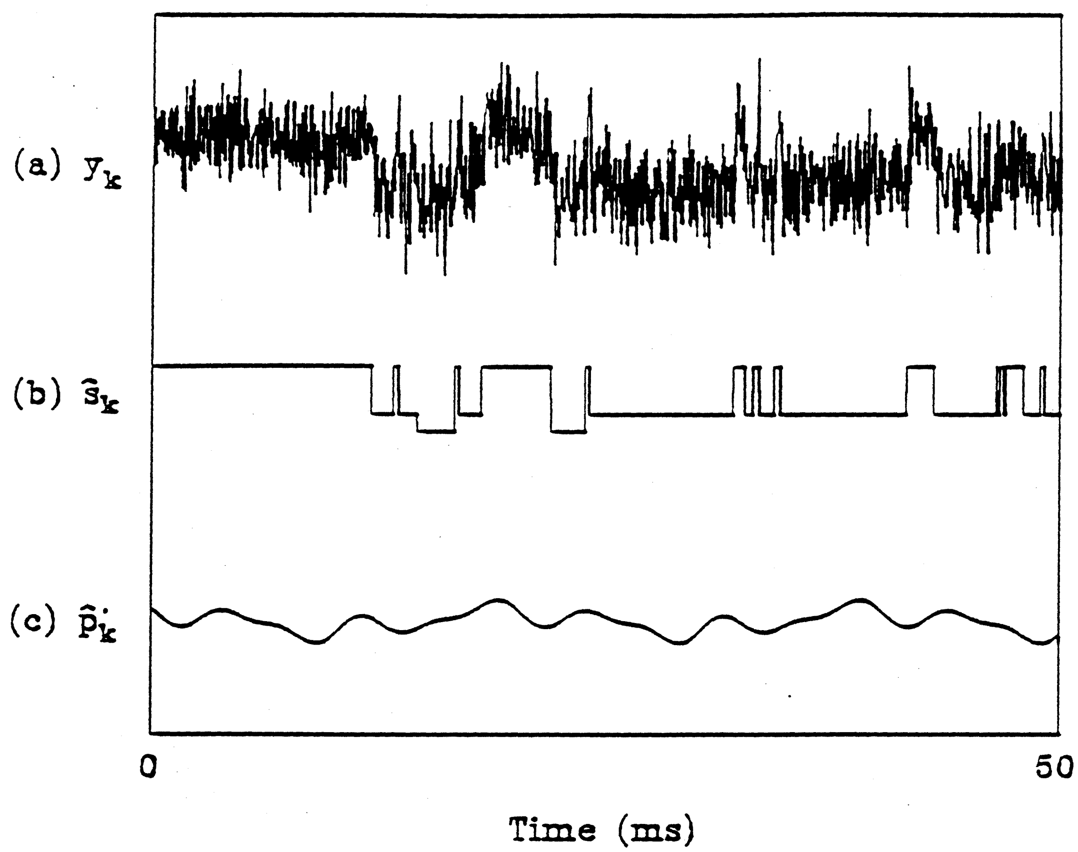


Fig. 6.6 : Hum elimination in biological data

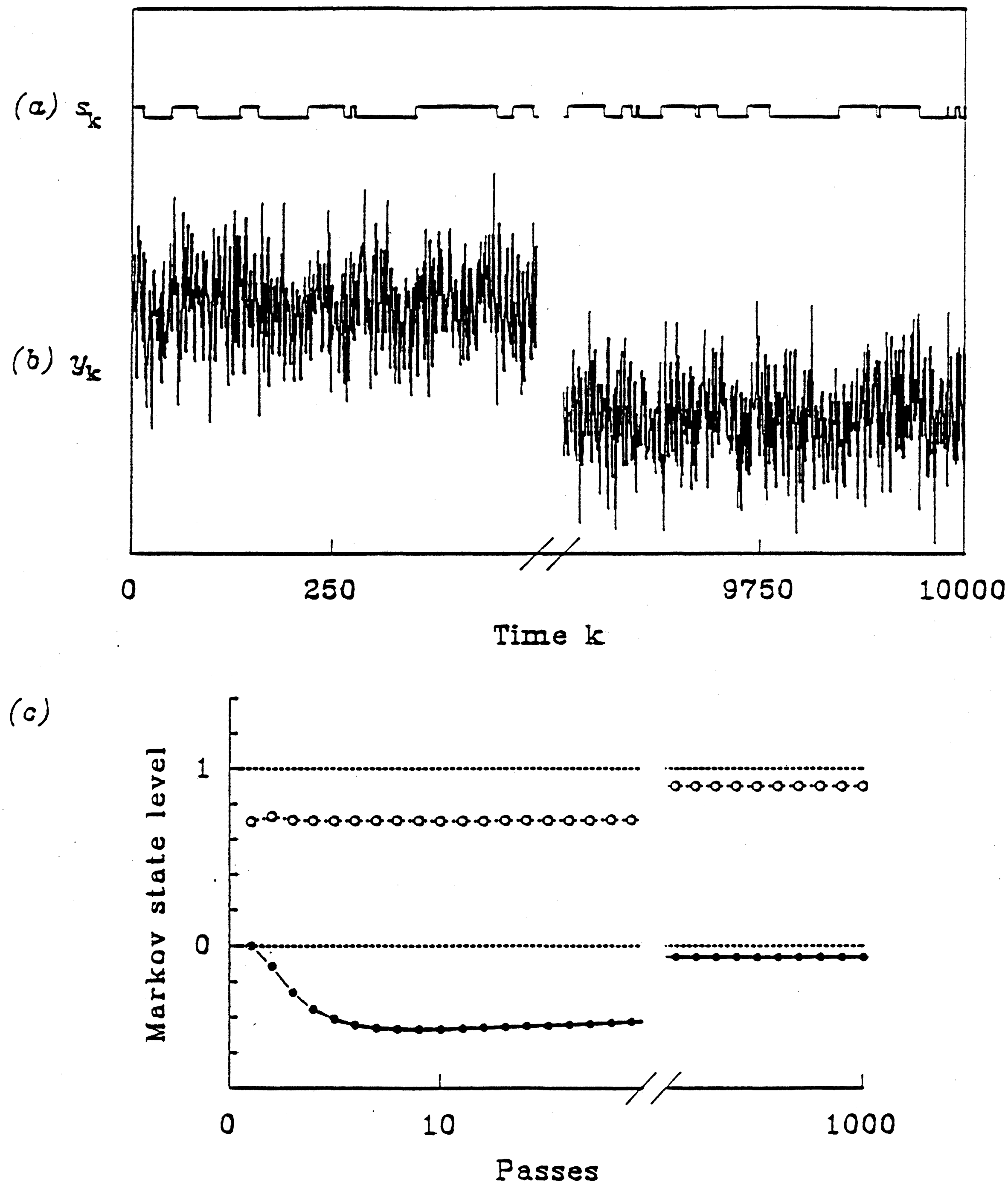


Fig. 6.7 : Re-estimation in the presence of linear drift

permits the influx of chloride ions from the outside of the membrane toward the inside, giving the downward deflection of the current record (Fig. 6.6a). The record showing channel currents is contaminated by white Gaussian noise with $\sigma_w = 0.4$ pA, as well as AC 'hum'. Biologists usually would discard such data, however expensive the experiment, due to the significant hum.

We assume the 'hum' is periodic with unknown amplitude and phase. An FFT reveals frequency components of approximately 50 Hz, 150 Hz and 250 Hz.

We ran our re-estimation scheme on this data. The MAP estimate of the Markov signal after 30 passes is shown in Fig.6.6b. Also the estimated periodic disturbance $\hat{p}_k$ is in Fig.6.6c. For clarity, the amplitude of $\hat{p}_k$ is magnified by twofold.

Although the observed data is severely contaminated by the periodic interference, the estimate of signal sequence using our re-estimation algorithm clearly reveals the two Markov states (conductance levels) of the ion channel at 26 pS (pico siemens) and 35 pS.

6.4.2 Polynomial drift

We consider the same Markov chain used in the examples above. To this chain was added zero mean Gaussian white noise with $\sigma_w = 4$. Also polynomial drift of the form $a\,k$ was added with $a = 1 \times 10^{-3}$.

Assuming known σ_w, we used the Block Component Re-estimation updating $\overline{a}$ and $\overline{q}_n$ in alternate passes. The same initial values were used for the Markov chain as in Example 1. Also $\overline{a}$ was initialized to 0.5×10^{-3}.

Fig.6.7a, b show 1000-point segments of the Markov chain s_k and the observations y_k. Fig.6.7c shows the evolution of the Markov state level estimates with successive passes. After 500 passes the Markov state level estimates are -6.51×10^{-2} and 0.90. Also $\overline{a}_{11} = \overline{a}_{22} = 0.967$. Finally the estimate of the drift coefficient is $\overline{a} = 1.006 \times 10^{-3}$, which shows that the proposed scheme yields satisfactory results.

6.5 Conclusions

In this chapter we have considered the problem of Hidden Markov Model signal processing when the Markov signal is buried in a mixture of white Gaussian noise and a deterministic signal. Two examples of deterministic signals were considered: periodic signals with unknown frequency components, amplitudes and phases, and polynomial drift in the states. We have proposed Maximum likelihood techniques based on EM methods for estimating the transition probabilities, state levels and state estimates of the Markov process and also for estimating the parameters of the deterministic disturbance. Following on from this research, in the next chapter we derive on-line identification schemes to cope with slowly time-varying amplitudes and phases in sinusoidal disturbances and also time-varying drift.

Chapter 7

On-line Hidden Markov Model Signal Processing and Elimination of Deterministic Interferences

7.1 Introduction

An important class of nonlinear signal models is that in which homogeneous finite-state Markov signals are imbedded in noise. The signal model is characterized in terms of state signal levels, state transition probabilities and noise characteristics. When the signal model is known, maximum likelihood signal estimates can be obtained using dynamic programming (the Viterbi algorithm [21], [62], [63]) which involves causal processing with delay, i.e., fixed-lag smoothing. However, to achieve certain signal statistics necessary for signal model identification, the standard approach is noncausal, involving a forward and backward pass. This constitutes 'fixed-interval smoothing' since the estimates are based on the entire data batch of observations. Repeated applications of this so called forward-backward algorithm can be used, along with repeated applications of the Baum-Welch [17], [18], [19] re-estimation formulae to achieve maximum likelihood estimates of the signal model parameters, including signal levels, transition probabilities and noise statistics [21]. This procedure constitutes the Expectation Maximization (EM) [20] algorithm.

A limitation of the above off-line EM methods is the 'curse of dimensionality' which arises because the required computational effort, speed and memory requirements are in proportion to the square of the number of states of the finite Markov model. Memory requirements are also proportional to the length of data being processed. There is thus an incentive to explore online (sequential) algorithms to seek improvements in terms of memory and computational speed.

Sequential algorithms have been proposed in [15] for the case where the imbedded signal is independent and identically distributed (iid). Specifically , in [15] a 'sequential' EM algorithm is presented which asymptotically achieves maximum likelihood estimates. In this chapter we develop on-line schemes for the case when the imbedded signal is Markov. In such a case, it is necessary to replace the fixed-interval forward-backward algorithm by fixed-lag schemes to obtain an effective re-estimation scheme.

The concept of using fixed-lag smoothed estimates in on-line signal processing is well understood for linear stochastic signal models [68]. Also, its application to finite-state Markov models has been studied in [69]. The results in [69] evolved from discretizing Wonham's stochastic differential equation [70] yielding fixed-point smoothing equations leading to fixed-lag smoothing equations. The algorithms proposed in [69], [70] even for the filtering (causal) case, appear subject to numerical instability problems. Here we develop smoothing schemes based on Hidden Markov Models that are numerically robust. Also the algorithms in [69] assume complete knowledge of the state levels, transition probabilities, and noise statistics. We focus attention on adaptive identification of the signal model, including signal levels, while achieving adaptive signal estimates and statistics.

In this chapter we first develop fixed- and 'sawtooth'-lag alternatives to the fixed-interval forward-backward scheme. The 'sawtooth' lag schemes are the most attractive form from a computational point of view. It has memory requirements proportional to $\Delta_{\min}$, and computational effort is within a constant of the forward-backwards algorithm, being invariant of $\Delta_{\min}$ and $\Delta_{\max}$. The lag Δ varies in a sawtooth fashion between $\Delta_{\min}$ and $\Delta_{\max}$. For completeness, we also derive *fixed-lag smoothing* versions of the forward-backward scheme (i.e., causal processing schemes with delay Δ being a constant). The proposed fixed-lag schemes turn out to be computationally less attractive than the sawtooth-lag smoothing schemes, requiring computational effort proportional to Δ. Typically with a delay Δ of 20 or so, our sawtooth-lag and fixed-lag smoothed estimates approach the optimal fixed interval smoothed estimate of the forward-backward algorithm. Simulations show that larger values of Δ do not yield significantly improved smoothed estimates.

Whereas Baum-Welch re-estimation is done on a block of data (typically a few thousand observations) and uses fixed-interval estimates, our re-estimation scheme is designed to be implemented recursively, thereby achieving on-line learning. Thus in our *on-line recursive estimation formulae* the signal model estimates and signal statistics are updated at each time instant, with appropriate forgetting of past estimates. This bootstrapping leads to improved convergence. This results in truly on-line reestimation in that often only one pass is required through the data to learn the signal model.

As an application of the proposed on-line techniques, we consider the case where in addition to white Gaussian noise (WGN), the Markov process is also corrupted by a deterministic signal of known functional form with unknown parameters. This application is motivated by examples in neuro-physical signal processing of ionic currents in cell membrane channels [6]. We consider two such examples of deterministic disturbances:

1. Periodic or almost periodic disturbances with unknown phases and amplitudes of the components of the periodic signal. An example of such periodic disturbances with approximately known frequency components (fundamental and harmonics) is the periodic 'hum' from the electricity mains which can be too costly to eliminate from some experimental environments.

2. Polynomial drift in the states of the Markov process. Such polynomial drift often occurs in cell-membrane channel measurements due to slow development of liquid junction potential arising from two different solutions of ionic conpositions [14].

We have developed off-line maximum likelihood schemes based on the EM algorithm that effectively extract the Markov signals and the deterministic inteferences [71]. However, these off-line schemes assume constant parameters which can be unrealistic. eg. the amplitude, phase of the ac mains and so the ac hum can change slowly with time. The on-line schemes we propose here deal with slowly varying parameters or parameters which undergo infrequent jump changes.

The chapter is organized as follows: In Section 7.2 we formulate our signal model as a Hidden Markov Model, define learning and estimation objectives, and review standard HMM techniques. In Section 7.3, fixed and sawtooth-lag on-line estimation schemes are presented. We propose and justify our on-line re-estimation scheme in Section 7.4. In Section 7.5, we consider the case when deterministic interferences are also present. Simulation studies are presented in Section 7.6 and some conclusions are drawn in Section 7.7.

7.2 Hidden Markov Model Signal Processing

In this section we describe the signal model, define learning and estimation objectives and review the standard off-line forward-backwards and Baum-Welch algorithms.

7.2.1 Signal Model

The underlying dynamics are assumed to be governed by a finite-state, discrete-time homogeneous first order Markov process. Consequently, the state s_k at time k is one of a finite number N of states $\mathbf{q} = q_1, q_2, \ldots, q_N$. The transition probability matrix is $\mathbf{A} = (a_{ij})$ where $a_{ij} = P(s_{k+1} = q_j | s_k = q_i)$. Of course $a_{ij} \geq 0$, $\sum_{j=1}^{N} a_{ij} = 1$, for each i.

We now assume that the Markov process s_k is hidden, that is indirectly observed by measurements y_k where

$$y_k = s_k + w_k, \quad w_k \sim N[0, \sigma_w^2].$$

Here w_k is zero mean mean white Gaussian noise (WGN). We denote the sequence $y_1, y_2, \ldots, y_k$ by Y_k. Define the symbol probabilities $\mathbf{b}(.) = (b_i(.))$ where

$$b_i(y_k) = P(y_k | s_k = q_i) = (\sqrt{2\pi}\,\sigma_w)^{-1} \exp(-(y_k - q_i)^2/(2\,\sigma_w^2))$$

are assumed invariant of k, with an independence property $P(y_k | s_k = q_i, s_{k-1} = q_j, Y_{k-1}) = P(y_k | s_k = q_i)$. Also we assume that the initial state probability vector $\underline{\pi} = (\pi_i)$ is defined from $\pi_i = P(s_1 = q_i)$. The HMM is denoted $\lambda = (\mathbf{A}, \mathbf{q}, \underline{\pi})$.

Our Markov model is assumed to be first order for simplicity. Extending the theory to m-th order Markov processes and semi-Markov processes is straightforward [56], although the associated computations are more formidable. Aggregation (reduced state) estimation techniques can be used as in [74], [75] to simplify the computations. The algorithms proposed in this chapter assume stationarity of the underlying transition probabilities. However, the proposed algorithms are also suitable for slowly varying transition probabilities and transition probabilities with infrequent 'jump' changes.

7.2.2 Learning and Estimation Objectives

Given the signal model as described above and given observations $y_1, y_2, \ldots, y_k$ denoted $\mathbf{Y}_k$, there are three inter-related problems which can be solved [52], [21]. In this chapter we provide suboptimal on-line solutions to these problems.

1. Evaluate the likelihood of a given model λ_i generating the given sequence of data, denoted $P(\mathbf{Y}_k|\lambda_i)$. This allows comparison of a set of models $\{\lambda_i\}$ to select the most likely, given the observations.

2. Estimate signal statistics such as a posteriori probabilities for an assumed λ and data sequence. For *off-line* processing for a fixed interval T, estimate

$$\underline{\gamma}_{k|T} = (\gamma_{k|T}(i)); \quad \gamma_{k|T}(i) \triangleq P(s_k = q_i|\mathbf{Y}_T, \lambda). \tag{7.1}$$

and calculate fixed-interval state probabilities (histograms)

$$\mathbf{h}_{T|T} = (h_{T|T}(i)); \quad h_{T|T}(i) \triangleq P(q_i|\mathbf{Y}_T, \lambda) = \frac{1}{T}\sum_{k=1}^{T} \gamma_{k|T}(i) \tag{7.2}$$

For *on-line* processing with a fixed or sawtooth-lag Δ, estimate signal probabilities

$$\underline{\gamma}_{k|k+\Delta} = (\gamma_{k|k+\Delta}(i)); \quad \gamma_{k|k+\Delta}(i) \triangleq P(s_k = q_i|\mathbf{Y}_{k+\Delta}, \lambda) \tag{7.3}$$

and calculate state probabilities $h_{\Delta|T}(i) = \frac{1}{T-\Delta}\sum_{k=1}^{T-\Delta} \gamma_{k|k+\Delta}(i)$.

Let us use $\underline{\gamma}_{k|(.)}$ to denote either $\underline{\gamma}_{k|T}$ or $\underline{\gamma}_{k|k+\Delta}$ depending on whether the processing is off-line or on-line. From $\gamma_{k|(.)}(i)$, maximum a posteriori (MAP) state estimates can be generated as

$$\hat{s}_k^{\mathrm{MAP}} = q_j \text{ where } j = \arg\max_{1\leq i\leq N} \gamma_{k|(.)}(i) \tag{7.4}$$

3. Processing of the observations based on a model assumption λ and re-estimation of the model parameters (functions) $(\mathbf{A}, \mathbf{q}, \underline{\pi})$ so as to obtain maximum likelihood estimates. In the off-line case the Baum-Welch re-estimation formulae yield estimates $\overline{\lambda}$ after each pass, such that the updated model $\overline{\lambda} = (\overline{\mathbf{A}}, \overline{\mathbf{q}}, \overline{\underline{\pi}})$ is more likely, i.e., $P(Y_T|\overline{\lambda}) \geq P(Y_T|\lambda)$.

In the online case the updated model λ_k at time k approaches asymptotically the maximum likelihood model for sufficently large k. Optimal schemes for solving the three problems using off-line processing are presented in [21] and reviewed below. In Sections

7.3 and 7.4, we present suboptimal on-line solutions.

Review of Standard Forward-backward Procedure Forward and backward vector variables are defined as probabilities conditioned on the model λ, for $i = 1, 2, \ldots, N$ as follows:

$$\underline{\alpha}_k = (\alpha_k(i)); \quad \alpha_k(i) \triangleq P(\mathbf{Y}_k, s_k = q_i|\lambda); \quad \underline{\beta}_k = (\beta_k(i)); \quad \beta_k(i) \triangleq P(\overline{\mathbf{Y}}_k|s_k = q_i, \lambda) \tag{7.5}$$

where $\overline{\mathbf{Y}}_k = y_{k+1}, y_{k+2}, \ldots, y_T$. Recursive formulae for $\underline{\alpha}_k$ and $\underline{\beta}_k$ are readily calculated [21] as

$$\alpha_k(j) = \left(\sum_{i=1}^{N} \alpha_{k-1}(i)\, a_{ij}\right) b_j(y_k), \quad \alpha_1(j) = \pi_j\, b_j(y_1)$$
$$\beta_k(i) = \sum_{j=1}^{N} a_{ij}\, b_j(y_{k+1})\, \beta_{k+1}(j), \quad \beta_T(i) = 1. \tag{7.6}$$

Updating $\underline{\alpha}_k$ and $\underline{\beta}_k$ requires the order of N^2 multiplications and additions at each time instant k.

Also $\gamma_k(i) = \alpha_k(i)\, \beta_k(i) / \sum_{i=1}^{N} \alpha_k(i)\, \beta_k(i)$ Notice that calculating $\underline{\gamma}_k$ involves first computing and storing the T vectors $\underline{\beta}_1$ to $\underline{\beta}_T$. Then as $\underline{\alpha}_k$, $1 \leq k \leq T$ is calculated, $\underline{\gamma}_k$ can also be computed.

The likelihood function, which is the end result of the Problem 1, is calculated as

$$L \triangleq P(\mathbf{Y}_T|\lambda) = \sum_{i=1}^{N} \alpha_T(i). \tag{7.7}$$

Review of EM algorithm and Baum-Welch Formulae The solution of Problem 3 in the off-line case involves the EM algorithm. It is an iterative algorithm; each iteration consists of an Expectation step (E step) and a Maximization step (M step).

E step: This involves computing $Q(\lambda, \overline{\lambda})$ which is the expectation of the log of the likelihood function of the fully categorized data (see [15]) for details). Here

$$Q(\lambda, \overline{\lambda}) = \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{k=1}^{T-1} \zeta_k(i,j) \log \overline{a}_{ij} + \sum_{i=1}^{N} \gamma_1(i) \log \overline{\pi}_i + Q_2 \tag{7.8}$$

where

$$Q_2 = \sum_{k=1}^{T-1}\sum_{i=1}^{N} \gamma_k(i) \log \frac{1}{\sqrt{2\pi}\sigma_w} \exp\left(-\frac{(y_k - \overline{q}_i)^2}{2\sigma_w^2}\right) \tag{7.9}$$

$$\zeta_k(i,j) \triangleq P(s_k = q_i, s_{k+1} = q_j, \mathbf{Y}_T|\lambda) = \alpha_k(i)\, a_{ij}\, b_j(y_{k+1})\, \beta_{k+1}(j) \tag{7.10}$$

and $\overline{q}_i$ is the estimate of the Markov state level q_i.

M step: This involves finding $\overline{\lambda}$ to maximize $Q(\lambda, \overline{\lambda})$. The optimal HMM λ is loosely denoted $\overline{\lambda}$. Maximizing each terms in Q yields the standard Baum-Welch equations

$$\overline{a}_{ij} = \frac{\sum_{k=1}^{T-1} \zeta_k(i,j)}{\sum_{k=1}^{T-1}\sum_{j=1}^{N} \zeta_k(i,j)}; \quad \overline{\pi}_i = \gamma_1(i); \quad \overline{q}_i = \frac{\sum_{k=1}^{T-1} \gamma_k(i)\, y_k}{\sum_{k=1}^{T-1} \gamma_k(i)} \tag{7.11}$$

Since the noise is assumed to be Gaussian, the symbol probabilities can be re-estimated as

$$\overline{b}_i(y_k) = (\sqrt{2\pi}\,\sigma_w)^{-1} \exp(-(y_k - \overline{q}_i)^2/(2\,\sigma_w^2))$$

The appealing property of EM is that $P(\mathbf{Y}_T|\overline{\lambda}) \geq P(\mathbf{Y}_T|\lambda)$

7.3 On-line Estimation

We present fixed-lag and sawtooth-lag smoothing schemes based on the standard backward recursions reviewed in the previous section. We seek to significantly decrease the memory requirement of storing T vectors to compute $\underline{\gamma}$ in the standard scheme. It turns out that the fixed-lag recursions are less attractive to implement than the sawtooth-lag ones. However, the former are included for completeness and to lead into the sawtooth-lag schemes.

7.3.1 Fixed-lag Estimation

For a fixed-lag of Δ, let us define the backward vector variable for $i = 1, \ldots, N$ as

$$\underline{\beta}_{k|k+\Delta} = (\beta_{k|k+\Delta}(i)); \; \beta_{k|k+\Delta}(i) = P(\overline{\mathbf{Y}}_{k,k+\Delta}|s_k = q_i, \lambda); \; \beta_{k+\Delta|k+\Delta}(i) = 1 \tag{7.12}$$

where $\overline{\mathbf{Y}}_{k,k+\Delta}$ denotes the future Δ observations $y_{k+1}, y_{k+2}, \ldots, y_{k+\Delta}$ and $2 \leq \Delta$ and $k + \Delta \leq T$. Thus $\mathbf{Y}_k$ and $\overline{\mathbf{Y}}_{k,k+\Delta}$ together form the $k+\Delta$ length sequence $\mathbf{Y}_{k+\Delta}$. Of course $\underline{\beta}_{k|T} = \underline{\beta}_k$ and $\overline{\mathbf{Y}}_{k,T} = \overline{\mathbf{Y}}_k$.

It turns out that to achieve a recursive form for $\underline{\beta}_{k|k+\Delta}$ analogous to (7.6) requires a sawtooth lag as described later. So the most attractive method for evaluating $\underline{\beta}_{k|k+\Delta}$ for the fixed-lag case is to compute it directly at each instant k. This requires of the order of $\Delta\, N^2$ multiplications and additions at each k and is as follows.

Lemma 7.1: Let $\mathbf{B}(k+1)$ be the diagonal matrix $diag(b_1(y_{k+1}),\ldots,b_N(y_{k+1}))$. Then

$$\underline{\beta}_{k|k+\Delta} = rowsum\{\mathbf{A}\,\mathbf{B}(k+1)\cdots\mathbf{A}\,\mathbf{B}(k+\Delta)\} \tag{7.13}$$

where $\mathbf{A}$ is the transition probability matrix.

Proof: From (7.12) and (7.6) it follows that

$$\begin{aligned}\underline{\beta}_{k|k+\Delta} &= rowsum\{\mathbf{A}\,\mathbf{B}(k+1)\,\underline{\beta}_{k+1|k+\Delta}\}\\ &= rowsum\{\mathbf{A}\,\mathbf{B}(k+1)\,\mathbf{A}\,\mathbf{B}(k+2)\,\underline{\beta}_{k+2|k+\Delta}\}\\ &\;\;\vdots\\ &= rowsum\{\mathbf{A}\,\mathbf{B}(k+1)\cdots\mathbf{A}\,\mathbf{B}(k+\Delta)\,\underline{\beta}_{k+\Delta|k+\Delta}\}\end{aligned}$$

But $\beta_{k+\Delta|k+\Delta}(i) = 1,\ i = 1,\ldots,N$ by definition and so (7.13) follows. •

The suboptimal *a posteriori* probabilities associated with Problem 2 can be computed similarly to the off-line case as

$$\underline{\gamma}_{k|k+\Delta} = (\gamma_{k|k+\Delta}(i));\ \ \gamma_{k|k+\Delta}(i) = \frac{\alpha_k(i)\,\beta_{k|k+\Delta}(i)}{\sum_{i=1}^{N}\alpha_k(i)\,\beta_{k|k+\Delta}(i)},\ \ i = 1,2,\ldots,N. \tag{7.14}$$

Then using (7.4) the fixed-lag MAP and CM state sequence estimates are obtained.

Lemma 7.1 implies that for the fixed-lag case $\underline{\beta}_{k|k+\Delta}$ can be computed forwards. This has two advantages: First, only one vector $\underline{\beta}_{k|k+\Delta}$ needs to be stored to compute $\underline{\gamma}_{k|k+\Delta}$. Second, if we can update estimates of $\mathbf{A}$ at each iteration (we give algorithms for this in Section 7.4) then the updated $\mathbf{A}$ at each time instant k can be used to calculate $\underline{\alpha}_k$ and $\underline{\beta}_{k|k+\Delta}$ leading to faster convergence in the solutions of Problems 2 and 3.

7.3.2 Sawtooth-lag Estimation

In the sawtooth-lag scheme we work with a lag varying from a fixed lag Δ_{min} to another of $\Delta_{\text{max}} \geq 2\,\Delta_{\text{min}}$ maintaining an average lag of $(\Delta_{\text{min}} + \Delta_{\text{max}})/2$. Let us consider a subinterval $[k_1, k_1 + \Delta_{\min}]$ with $k_1 + \Delta_{\max} \leq T$ and define a sawtooth lag Δ at time k as

$$\Delta \triangleq k_1 + \Delta_{\max} - k, \quad k_1 \leq k \leq k_1 + \Delta_{\min}. \tag{7.15}$$

Also define the backward vector variable from (7.12) with Δ, now k dependent, as in (7.15)

$$\underline{\beta}_{k|k+\Delta} = (\beta_{k|k+\Delta}(i)); \; \beta_{k|k+\Delta}(i) = P(\overline{\mathbf{Y}}_{k,k+\Delta}|s_k = q_i, \lambda), \; i = 1, \ldots, N. \tag{7.16}$$

Of course $\overline{\mathbf{Y}}_{k,k+\Delta}$ denotes the observations $y_{k+1}, y_{k+2}, \ldots, y_{k+\Delta}$ where $k + \Delta = k_1 + \Delta_{max}$ is fixed for each interval $[k_1, k_1 + \Delta_{\max}]$. Then using the same derivation approach as for (7.6), we have

$$\beta_{k|k+\Delta}(i) = \sum_{j=1}^{N} a_{ij}\, b_j(y_{k+1})\, \beta_{k+1|k+\Delta}(j), \quad \beta_{k+\Delta|k+\Delta}(i) = 1 \tag{7.17}$$

To calculate $\underline{\gamma}_{k|k+\Delta}$ we first compute $\underline{\beta}_{k|k+\Delta}$ for $k_1 \leq k \leq k_1 + \Delta$ using (7.17) and store for $k_1 \leq k \leq k_1 + \Delta_{\min}$. Then for the range $[k_1, k_1 + \Delta_{\min}]$ of k, we compute $\underline{\alpha}_k$ using (7.6) and update $\underline{\gamma}_{k|k+\Delta}$ using (7.14).

Remark 1: The sawtooth-lag scheme is computationally attractive because at each time instant k, the computations involved in calculating $\underline{\beta}_{k|\Delta}$ are of order $O(N^2)$. In contrast, the fixed-lag scheme requires $O(\Delta\, N^2)$ computations.

Remark 2: There is the possibility of trading off computation time with memory. For example, by choosing $\Delta_{\max} = 2\,\Delta_{\min}$, the memory requirements are minimal and the computation effort required in calculating $\underline{\beta}_{k|\Delta}$ is twice that for the forward-backward procedure. However, by choosing $\Delta_{\max} >> \Delta_{\min}$, the computation effort required approaches the forward-backward procedure but the memory requirements are increased in proportion to $\Delta_{\max}$. $\circ$

7.4 Parameter Re-estimation

In this section we propose on-line re-estimation formulae, discuss implementation aspects and give a theoretical justification.

On-line Re-estimation: Let $\lambda_k = (\mathbf{A}(k),\ \mathbf{q}(k))$ be the model estimate at time k. Then we propose that λ_{k+1} is calculated as:

$$
\begin{aligned}
a_{ij}(k+1) &= a_{ij}(k) + \frac{a_{il_i}(k)\,\zeta_k(i,j) - a_{ij}(k)\,\zeta_k(i,l_i)}{\sum_{t=1}^{k} a_{il_i}(k)\,\zeta_t(i,j)/a_{ij}(k) + a_{ij}(k)\,\zeta_t(i,l_i)/a_{il_i}(k)}, \quad j \neq l_i \\
a_{il_i}(k+1) &= 1 - \sum_{j=1}^{N} a_{ij}(k), \quad j \neq l_i \qquad (7.18) \\
q_i(k+1) &= q_i(k) + \frac{\gamma_k(i)(y_k - q_i(k))}{\sum_{t=1}^{k} \gamma_t(i)} \qquad (7.19)
\end{aligned}
$$

where l_i is any integer in $[1, N]$.

Remark : Notice that the above re-estimation formulae (7.18) ensure that $\sum_j a_{ij} = 1$.

○

Of course because the noise is Gaussian, once the state levels $q_i^{(k)}$ are known then the symbol probability estimates can be calculated as

$$
b_i(k, y_k) = \frac{1}{\sqrt{2\,\pi}\sigma_w} \exp\left(-\frac{y_k - q_i(k)}{2\sigma_w^2}\right) \qquad (7.20)
$$

Implementation aspects: We discuss two implementation aspects which enhance performance on finite length data.

1. Sawtooth-lag implementation: In the actual implementation, it would be extremely inefficient if $\underline{\zeta}_k$ and $\underline{\gamma}_k$ were calculated at each time instant for the updated model λ_k. Instead we use the sawtooth-lag variables $\underline{\zeta}_{k|k+\Delta}$ and $\underline{\gamma}_{k|k+\Delta}$. Also π_i is estimated as $\gamma_{1|1+\Delta}(i)$.

2. Exponential forgetting: Simulations have shown that to learn the model parameters quickly, it is advantageous to do exponential forgetting. For the case of periodic disturbance, we propose resetting the denominators of (7.18) and (7.19) to a fraction of their value when the terms exceed a particular threshold. We propose a similar scheme for the polynomial drift case. Other forgetting schemes may also be considered, see [76].

3. Sufficient excitation: In principle l_i can be chosen as any integer in $[1, N]$ in (7.18). However for finite lengths of data appropriate selection of l_i yields more accurate results. When $\mathbf{A}$ is diagonally dominant, then best results are obtained by choosing $l_i = i$. This is because estimates of a_{ii} will be more accurate than a_{ij}, $j \neq i$. So using $l_i = i$ in (7.18) involves using a_{ii} which can be estimated accurately. (The estimates of a_{ii} are more ac-

curate than a_{ij} when $\mathbf{A}$ is diagonally dominant since the number of transitions in a finite length of data from the state q_i to another state q_j are relatively few. So a_{ij} cannot be reliably estimated).

4. Specifying exact number of states: Like the Baum-Welch scheme, the on-line scheme does not require knowing the exact number of Markov states present as long as some realistic upper bound on the number is specified. After running the on-line scheme with this specified upper bound, we neglect the states for which the state probabilities $h_{\Delta|T}$ (see Section 7.2.2) are close to zero. Simulations not presented here confirm that $h_{\Delta|T}$ yields satisfactory estimates of the state probabilities.

Justification of the on-line re-estimation equations: We now justify the on-line equations (7.18) and (7.19).

Let $I_c(\lambda_k)$ be the Fisher information matrix of the complete (fully categorized) data, i.e., $I_c(\lambda_k) = \partial^2 \mathcal{Q}(\lambda, \lambda_k)/\partial \lambda_k^2$ where

$$\begin{aligned}\mathcal{Q}(\lambda, \lambda_k) \;=\; & \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{t=1}^{k} \zeta_t(i,j) \log \overline{a}_{ij}(k) + \sum_{t=1}^{k}\sum_{i=1}^{N} \gamma_t(i) \log \frac{1}{\sqrt{2\pi}\sigma_w} \exp\left(-\frac{(y_t - \overline{q}_i(k))^2}{2\sigma_w^2}\right) \\ & + \sum_{i=1}^{N} \gamma_1(i) \log \overline{\pi}_i \end{aligned} \tag{7.21}$$

Notice that $\mathcal{Q}(\lambda, \lambda_k)$ is the expectation of the log likelihood of the fully categorized data until time k. For the off-line case $\mathcal{Q}$ defined in (7.8) is simply $\mathcal{Q}(\lambda, \lambda_T)$.

Let $I(\lambda_k)$ be the Fisher Information matrix $\partial^2 \log L/\partial \lambda_k^2$, where L is defined in (7.7).

Also let $S(\lambda_k, y_k)$ be the score vector, see [15] for details. It is proved in [76], [24] that the score vector can be computed by taking the conditional expectation of the complete data score. The data score is $\partial \mathcal{Q}_k/\partial \lambda_k$ where

$$\mathcal{Q}_k(\lambda, \lambda_k) = \sum_{i=1}^{N}\sum_{j=1}^{N} \zeta_k(i,j) \log \overline{a}_{ij}(k) + \sum_{i=1}^{N} \gamma_k(i) \log \frac{1}{\sqrt{2\pi}\sigma_w} \exp\left(-\frac{(y_k - \overline{q}_i(k))^2}{2\sigma_w^2}\right). \tag{7.22}$$

Now consider the following theorem from [15].

Theorem 7.1: Given appropriate regularity conditions [15], [72] provided $2I(\lambda)\, I_c(\lambda)^{-1} > 1$, the EM algorithm can be written as the following recursion:

$$\lambda_{k+1} = \lambda_k + (I_c(\lambda_k))^{-1}\, S(\lambda_k, y_k). \tag{7.23}$$

Moreover

$$\sqrt{k}(\lambda_k - \lambda) \to N\{0, I_c(\lambda)^{-2}\, I(\lambda)/(2I(\lambda)\, I_c(\lambda)^{-1} - 1)\} \tag{7.24}$$

in distribution as $k \to \infty$. Here λ is the model corresponding to a local maximum in the likelihood function L. •

Remark 1: Notice that (7.23) is a stochastic approximation procedure [25]. The regularity conditions for convergence of (7.23) to the maximum likelihood model are presented in [15]. Verification of these conditions is not straightforward, see [15], [73] for details.

Remark 2: If the noise is not from the exponential family of distributions then the above recursion is an approximation to the EM algorithm. Since in our case w_k is Gaussian (which is in the exponential family of distributions) the recursion is exact [72].

Remark 3 Effect of initial estimates: An important condition for (7.23) to converge to the true model when the system is in the model set is that the initial estimate be chosen sufficiently close to the true value. A theoretical analysis of the effect of initial estimates is beyond the scope of this chapter. However, in the case where additive white Gaussian noise with known variance is present, there could well be a global convergence result; perhaps there is one in the literature which we have been unable to find. Certainly, we have carried out extensive simulations upto $\sigma_w = 3.0$ and $N = 4$ with initial estimates of q_i as far away as $3\,\sigma_w$ away from the true values and have not found initial conditions for which convergence to the true model does not occur. We discuss the effect of initial estimates in Section 7.6.1. ∘

We now obtain expressions for $I_c(\lambda_k)$ and $S(\lambda_k, y_k)$ in (7.23).

Lemma 7.2: $I_c(\lambda_k) = diag(I_c^A(\lambda_k), I_c^Q(\lambda_k))$ where $I_c^A(\lambda_k)$ is defined as the diagonal matrix with elements $\partial^2 I_c/\partial a_{ij}^2$, $j \neq l_i$, $I_c^Q(\lambda_k)$ is a diagonal matrix with elements $\partial^2 I_c/\partial q_i^2$. These are evaluated as

$$I_c^A(\lambda_k)\text{blockdiag}(P_1, \ldots, P_N),\ P_i = \text{diag}(p_{i1}, \ldots, p_{i,l_i-1}, p_{i,l_i+1}, \ldots, p_{iN})$$

$$p_{ij} = \frac{\sum_{t=1}^{k} a_{il_i}^2(k)\,\zeta_t(i,j) + a_{ij}^2(k)\,\zeta_t(i,l_i)}{a_{ij}^2(k)\,a_{il_i}^2(k)}),\ j \neq l_i \tag{7.25}$$

and

$$I_c^Q(\lambda_k) = diag(\frac{\sum_{t=1}^{k}\gamma_t(1)}{\sigma_w^2}, \ldots, \frac{\sum_{t=1}^{k}\gamma_t(N)}{\sigma_w^2}) \tag{7.26}$$

Proof: The first term in (7.21) can be rewritten as

$$\sum_{t}^{k}\sum_{i=1}^{N}\sum_{j=1}^{N}\zeta_t(i,j)\log\overline{a}_{ij}(k)+\sum_{t=1}^{k}\sum_{i=1}^{N}\zeta_t(i,l_i)\log(1-\sum_{j=1}^{N}\overline{a}_{ij}(k)),\ j\neq l_i \tag{7.27}$$

Then evaluating $\partial^2 Q/\partial\lambda_k^2$ directly yields (7.25) and (7.26). •

Lemma 7.3: The score vector $S(\lambda_k, y_k)$ can be computed as $S(\lambda_k, y_k) = [S_A(\lambda_k, y_k), S_Q(\lambda_k, y_k)]'$ where $S_A(.)$ is a vector with elements $\partial Q_k/\partial a_{ij}$, $j \neq l_i$, $S_Q(.)$ is a vector with elements $\partial Q_k/\partial q_i$

$$S_A(\lambda_k, y_k) = (S_A(1), S_A(2), \dots, S_A(N))^T$$

$$S_A(i) = (s_A(i,1), \dots, s_A(1, l_i - 1), s_A(i, l_i + 1), \dots, s_A(i, N)) \tag{7.28}$$

where

$$s_A(i,j) = \left(\frac{a_{il_i}(k)\,\zeta_k(i,j) - a_{ij}(k)\,\zeta_k(i,l_i)}{a_{ij}(k)\,a_{il_i}(k)}\right),\ j \neq l_i$$

$$S_Q(\lambda_k, y_k) = (\gamma_1(k)(y_k - q_1(k))^2/\sigma_w^2, \dots, \gamma_N(k)(y_k - q_N(k))^2/\sigma_w^2) \tag{7.29}$$

Proof: Evaluating $\partial Q_k/\partial\lambda_k$ yields the above equations. •

Now substituting for $I_c(.)$ and $S(.)$ in (7.23) yields the on-line re-estimation formulae.

7.5 Elimination of Deterministic Interferences

We now illustrate an application of the proposed on-line techniques. We consider the case where in addition to white Gaussian noise (WGN), the Markov process s_k is also corrupted by a deterministic signal of known form with unknown parameters.

Let $p_k(\Theta)$, parametrized by $\Theta \in \mathbb{R}^R$ denote the deterministic disturbance. We assume that the functional form of p_k is known but the parameter vector $\Theta = (\theta_1, \dots, \theta_R)$ is unknown. In addition, it is assumed that Θ is slowly time-varying or undergoes infrequent jump changes. We consider two such examples of deterministic disturbances:

1. Periodic or almost periodic disturbances, $p_k(\Theta) = \sum_{n=1}^{p} a_n \sin(w_n k + \phi_n)$ and $\Theta = (a_1, \dots, a_p, \phi_1, \dots, \phi_p)$.
2. Drift in the states of the Markov process, $p_k(\Theta) = \sum_{n=1}^{p} a_n k^n$ and $\Theta = (a_1, \dots, a_p)$.

Signal Model: We assume that s_k is hidden, that is indirectly observed by measurements y_k, where

$$y_k = s_k + p_k + w_k. \tag{7.30}$$

Define the vector of probability functions $\mathbf{b}(\Theta, \sigma_w^2, \mathbf{q}, y_k) = (b_i(\Theta, \sigma_w^2, \mathbf{q}, y_k))$ where

$$b_i(\Theta, \sigma_w^2, \mathbf{q}, y_k) \triangleq P(y_k - p_k(\Theta)|s_k = q_i) = \frac{1}{\sqrt{2\pi}\,\sigma_w} \exp(-\frac{(y_k - q_i - p_k(\theta))^2}{2\,\sigma_w^2}). \tag{7.31}$$

For notational convenience, in the rest of this section we denote $b_i(\Theta, \sigma_w^2, \mathbf{q}, y_k)$ by $b_i(y_k)$. The resulting HMM is denoted $\lambda = (\mathbf{A}, \Theta, \mathbf{q}, \underline{\pi})$.

On-line Re-estimation: Again given the above signal model, three interelated problems can be posed and solved. The first two problems are identical to that in Section 7.2.2. So their solutions are identical to that in Section 7.3, with $b_i(y_k)$ defined in (7.31). Problem 3 now involves on-line re-estimation to asymptotically obtain the maximum likelihood model $\lambda = (\mathbf{A}, \Theta, \mathbf{q}, \underline{\pi})$. We now propose and justify on-line re-estimation formulae.

Let $\lambda_k = (\mathbf{A}_k, \mathbf{q}(k), \Theta(k))$ be the model estimate at time k. For notational convenience we shall drop the time dependency of λ_k on the right hand side of the re-estimation equations, i.e., we shall write $\lambda_k = (\mathbf{A}, \mathbf{q}, \Theta)$. Then $\lambda_{k+1} = (\mathbf{A}_{k+1}, \mathbf{q}(k+1), \Theta(k+1))$ is calculated as:

$\mathbf{A}_{k+1}$ is calculated from (7.18).

$$q_i(k+1) = q_i + \frac{\gamma_k(i)(y_k - q_i - p_k(\Theta))}{\sum_{t=1}^{k} \gamma_t(i)} \tag{7.32}$$

For periodic disturbances, Θ_{k+1} is re-estimated as

$$a_n(k+1) = a_n + \frac{(y_k - \sum_{n=1}^{p} a_n \sin(w_n k + \phi_n) - g_k)\sin(w_n k + \phi_n)}{\sum_{t=1}^{k} \sin^2(w_n t + \phi_n)} \tag{7.33}$$

$$\phi_n(k+1) = \phi_n + \frac{(y_k - \sum_{n=1}^{p} a_n \sin(w_n k + \phi_n) - g_k) a_n \cos(w_n k + \phi_n)}{\sum_{t=1}^{k} a_n^2 \cos^2(w_n t + \phi_n) + (y_t - \sum_{n=1}^{p} a_n \sin(w_n t + \phi_n) - g_t) a_n \sin(w_n t + \phi_n)} \tag{7.34}$$

For polynomial drift, Θ_{k+1} is re-estimated as

$$a_n(k+1) = a_n + \frac{(y_k - \sum_{n=1}^{p} a_n k^n - g_k)k^n}{\sum_{t=1}^{k} a_n t^{2n}} \tag{7.35}$$

In the above equations, $g_k = \sum_{i=1}^{N} \gamma_k(i)\, q_i$.

(Recall that in the above equations, the right hand side parameters are estimates at time k; for notational convenience we have dropped this time dependency).

Remark 1 Frequency Re-estimation: On-line re-estimation equations for updating the frequency components w_n can also be derived similarly. The equations are:

$$w_n(k+1) = w_n + \frac{k(y_k - \sum_{n=1}^{p} a_n \sin(w_n k + \phi_n) - g_k)\cos(w_n k + \phi_n)}{\sum_{t=1}^{k} t^2(a_n \cos(w_n t + \phi_n) + (y_t - \sum_{n=1}^{p} a_n \sin(w_n t + \phi_n) - g_t)\sin(w_n t + \phi_n))} \tag{7.36}$$

However, there a large number of local maximia in $\mathcal{Q}(\lambda, \lambda_k)$ [71], So there is no guarantee that (7.36) will converge to a global maxima.

We have developed off-line frequency re-estimation techniques in [71]. We believe that it may be possible to combine the off-line scheme with (7.36) to track time varying frequencies. Alternatively it may be possible to run a bank of on-line processers and select at each time k the estimate which maximizes $\mathcal{Q}(\lambda, \lambda_k)$. Further discussion of frequency re-estimation is beyond the scope of this chapter.

Remark 2 Independent and Identically distributed (iid) chain: We briefly consider the case when the imbedded chain is iid instead of Markov. Then the observations are characterized by the model $\lambda = (\underline{\pi}, \mathbf{q}, \Theta)$ where $\pi_i = P(s_k = q_i)$. The model estimate at time k is $\lambda_k = (\underline{\pi}(k), \mathbf{q}(k), \Theta(k))$ where $\pi_i(k)$ is the estimate of π_i at time k. The online re-estimation formulae are:

$$\pi_i(k+1) = \pi_i(k) + k^{-1}(\gamma_k(i) - \pi_i(k)) \tag{7.37}$$

$$q_i(k+1) = q_i(k) + \gamma_k(i)(y_k - q_i(k) - p_k(\Theta(k))/(k\,\pi_i(k)) \tag{7.38}$$

Here $\underline{\gamma}_k$ is easily computed as

$$\gamma_k(i) \triangleq P(s_k = q_i | y_k) = \pi_i(k) b_i(k, y_k) / \sum_{j=1}^{N} \pi_i(k) b_i(k, y_k) \tag{7.39}$$

The above equations are derived in [15], [72] using a similar approach to Section 7.4. The amplitude and phase re-estimation equations are given by (7.33) and (7.34) with $\underline{\gamma}_k$ defined in (7.39). These can be derived using a similar proof to that outlined below for the Markov case. $\circ$

Implementation details:

1. Again in the actual implementation, $\underline{\gamma}_k$ and $\underline{\zeta}_k$ are replaced by the **sawtooth-lag** variables.

2. Applying forgetting leads to quicker convergence. Apart from the forgetting scheme outlined in Section 7.4. for the Markov chain, we propose setting the denominators of (7.33) and (7.34) to a fraction of their value when these terms exceed a particular threshold. We illustrate the advantages of applying forgetting in the simulations.

Justification: The expectation of the log of the likelihood of the fully categorized data until time k is (see [71]) is

$$\begin{aligned} \mathcal{Q}(\lambda, \lambda_k) &= \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{t=1}^{k} \zeta_t(i,j)\log \overline{a}_{ij}(k) + \sum_{t=1}^{k}\sum_{i=1}^{N} \gamma_t(i)\log \frac{1}{\sqrt{2\pi}\sigma_w}\exp\left(-\frac{(y_t - \overline{q}_i(k) - p_k(\Theta_k))^2}{2\sigma_w^2}\right) \\ &+ \sum_{i=1}^{N} \gamma_1(i)\log \overline{\pi}_i \end{aligned} \quad (7.40)$$

We now obtain expressions for $I_c(\lambda_k)$, $S(\lambda_k, y_k)$ for the periodic disturbance case and show that the re-estimation equations follow from (7.23) of Theorem 7.1. The re-estimation equations in the case of polynomial drift can be derived similarly; for brevity we omit the proofs. Again for notational convenience we shall drop the time dependency of λ_k on the right hand side of the re-estimation equations.

Lemma 7.4: $I_c(\lambda_k) = diag(I_c^A(\lambda_k), I_c^Q(\lambda_k), I_c^\Theta(\lambda_k))$ where $I_c^A(\lambda_k)$ is computed in (7.25), $I_c^Q(\lambda_k)$ is computed in (7.26) and $I_c^\Theta(\lambda_k) = diag(I_c^a, I_c^\phi)$,

$$I_c^a = diag(-(\sum_{t=1}^{k}\sin^2(w_1 k + \phi_1)/\sigma_w^2), \ldots, -(\sum_{t=1}^{k}\sin^2(w_p k + \phi_p)/\sigma_w^2)) \quad (7.41)$$

$I_c^\phi = diag(I_c^\phi(1), \ldots, I_c^\phi(p))$ where for $m = 1, \ldots, p$

$$I_c^\phi(m) = -(a_m^2 \sum_{t=1}^{k} cos^2(w_m k + \phi_m) + (y_k - g_k - \sum_{n=1}^{p} a_n \sin(w_n k + \phi_n))a_m \sin(w_m k + \phi_m)/\sigma_w^2), \quad (7.42)$$

Proof: With $\mathcal{Q}(\lambda, \lambda_k)$ defined in (7.40), evaluating $\partial^2\mathcal{Q}/\partial\lambda_k^2$ yields the above equations. •

Lemma 7.5: $S(\lambda_k, y_k) = (S_A(\lambda_k, y_k), S_Q(\lambda_k, y_k), S_\Theta(\lambda_k, y_k))$

where $S_A(.)$ is computed in (7.28),

$$S_Q(\lambda_k, y_k) = (\gamma_1(k)(y_k - q_1(k) - p_k(\Theta_k))^2/\sigma_w^2, \ldots, \gamma_N(k)(y_k - q_N(k) - p_k(\Theta_k))^2/\sigma_w^2) \quad (7.43)$$

$S_\Theta(\lambda_k, y_k) = (S_a, S_\phi)$ where $S_a(\lambda_k, y_k) = (S_a(1), \ldots, S_a(p))$,

$$S_a(m) = (y_k - g_k - \sum_{n=1}^{p} a_n \sin(w_n k + \phi_n)) \sin(w_m k + \phi_m)/\sigma_w^2, \quad m = 1, \ldots, p \quad (7.44)$$

$S_\phi(\lambda_k, y_k) = (S_\phi(1), \ldots, S_\phi(p))$ where for $m = 1, \ldots, p$

$$S_\phi(m) = (y_k - g_k - \sum_{n=1}^{p} a_n \sin(w_n k + \phi_n)) a_m \cos(w_m k + \phi_m)/\sigma_w^2 \quad (7.45)$$

Proof: The data score is simply $\partial Q_k / \partial \lambda_k$ where

$$Q_k(\lambda, \lambda_k) \triangleq \sum_{i=1}^{N} \sum_{j=1}^{N} \zeta_k(i,j) \log \overline{a}_{ij}(k) + \sum_{i=1}^{N} \gamma_k(i) \log \frac{1}{\sqrt{2\pi}\sigma_w} \exp\left(-\frac{(y_k - \overline{q}_i(k) - p_k(\Theta_k))^2}{2\sigma_w^2}\right) \quad (7.46)$$

•

7.6 Simulation Studies

We illustrate the proposed schemes for the case where only white Gaussian nose is present and the case when also a periodic interference is present.

7.6.1 Markov Signals imbedded in White Gaussian Noise

Sawtooth-lag signal Extraction: Here we examine the effect of sawtooth-lag on the signal estimates. We assume that the transition probabilities and Markov levels are known.

A 200000 point two state Markov chain with $a_{11} = a_{22} = 0.97$ and $q_1 = 0$, $q_2 = 1$ was generated. To this was added zero mean white Gaussian noise (WGN) sequences with standard deviations $\sigma_w = 0.5, 1.0, 1.5, 2.0$ respectively to yield 4 sets of 200000 point observations.

With the initial transition probability estimates set at 0.97 and initial state level estimates set at 0 and 1, our online scheme was run on the 4 sets of observations. MAP signal estimates were obtained as in (7.4). Fig.7.1 shows the number of errors in the MAP signal estimates per 10000 points plotted versus sawtooth-lag Δ_{min}. Δ_{max} was taken as $2\,\Delta_{\text{min}}$.

Notice that for small noise variance σ_w, choosing $\Delta_{\text{min}} = 4$ yields satisfactory results. For larger σ_w, it is necessary to increase the lag to obtain acceptable signal estimates. In all cases increasing σ_w increases the number of errors.

Effect of sawtooth-lag on convergence: Using the same 4 observation sequences as above we want to illustrate the effect of lag on convergence of the on-line scheme.

Initial transition probability estimates $a_{ii}(1) = 0.8$ were chosen. In Fig. 7.2 we plot $a_{11}(k)$ versus k at 20000 point intervals for sawtooth lags with $\Delta_{\text{min}} = 2$ and 20 ($\Delta_{\text{max}} = 2\Delta_{\text{min}}$).

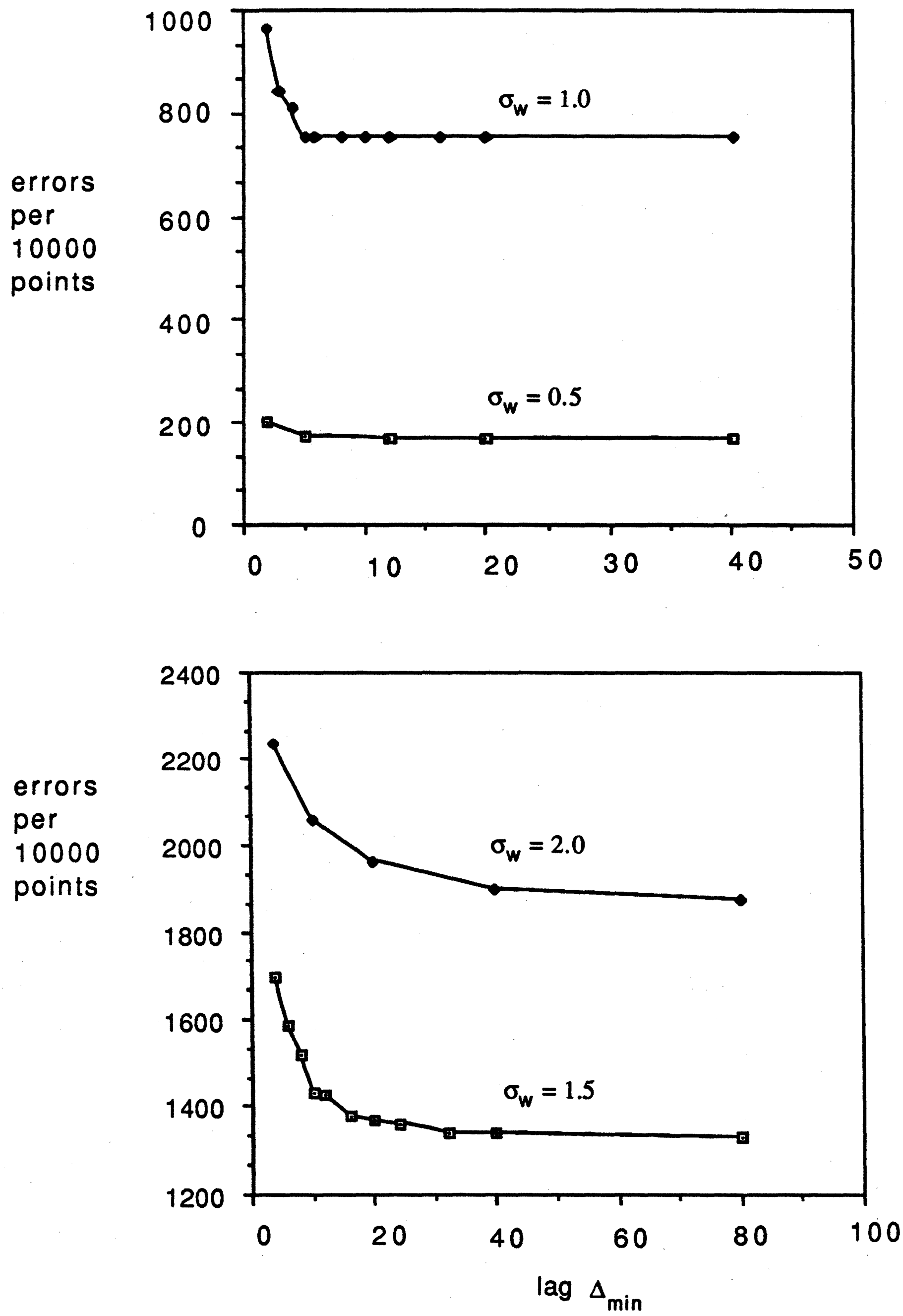
Notice that for the low noise case ($\sigma_w = 0.5$) the convergence is independent of lag. With increasing noise larger lags result in faster convergence. As in the state estimation case above, simulations show that there is no appreciable improvement in convergence beyond a certain lag. eg. for $\sigma_w = 1$, choosing $\Delta_{\text{min}} > 20$ does not result in significant improvements. Also convergence slows with increasing σ_w.

Comparison with Baum-Welch Re-estimation: We compare the performance of the proposed on-line scheme with the Baum-Welch (off-line) scheme.

A 100000 point two state Markov chain with $a_{11} = a_{22} = 0.97$ and $q_1 = 0$, $q_2 = 1$ was generated. To this was added zero mean WGN of standard deviation $\sigma_w = 2.0$ yielding the observations.

Initial state level estimates were taken as $q_1(1) = 0.1$ and $q_2(1) = 0.6$. The initial transition probability estimates were $a_{ii}(1) = 0.9$. Also the sawtooth-lag was chosen as $\Delta_{\text{min}} = 20$, $\Delta_{\text{max}} = 40$. Fig.7.3 shows the performance of the on-line algorithm on the observations. Notice that the level estimates $q_i(k)$ are close to the true values q_i from k=30000 onwards. The transition probability estimates $a_{11}(k)$, $a_{22}(k)$ also go towards the true values.

Fig.7.4 shows the performance of the Baum-Welch (off-line) re-estimation scheme on the same observation sequence. The same initial level and transition probability estimates were used as in the on-line case. Notice that the on-line estimates $q_i(k)$ for large k are close to the off-line estimates $\overline{q}_i$ in the 10th pass. Also the on-line estimates $a_{ii}(k)$ are close to that of the off-line estimates after 7 passes. These illustrate the significant improvement in convergence of the on-line scheme due to the bootstrapping.



Note $\Delta_{max} = 2\,\Delta_{min}$

Fig. 7.1 : Sawtooth-lag smoother performance

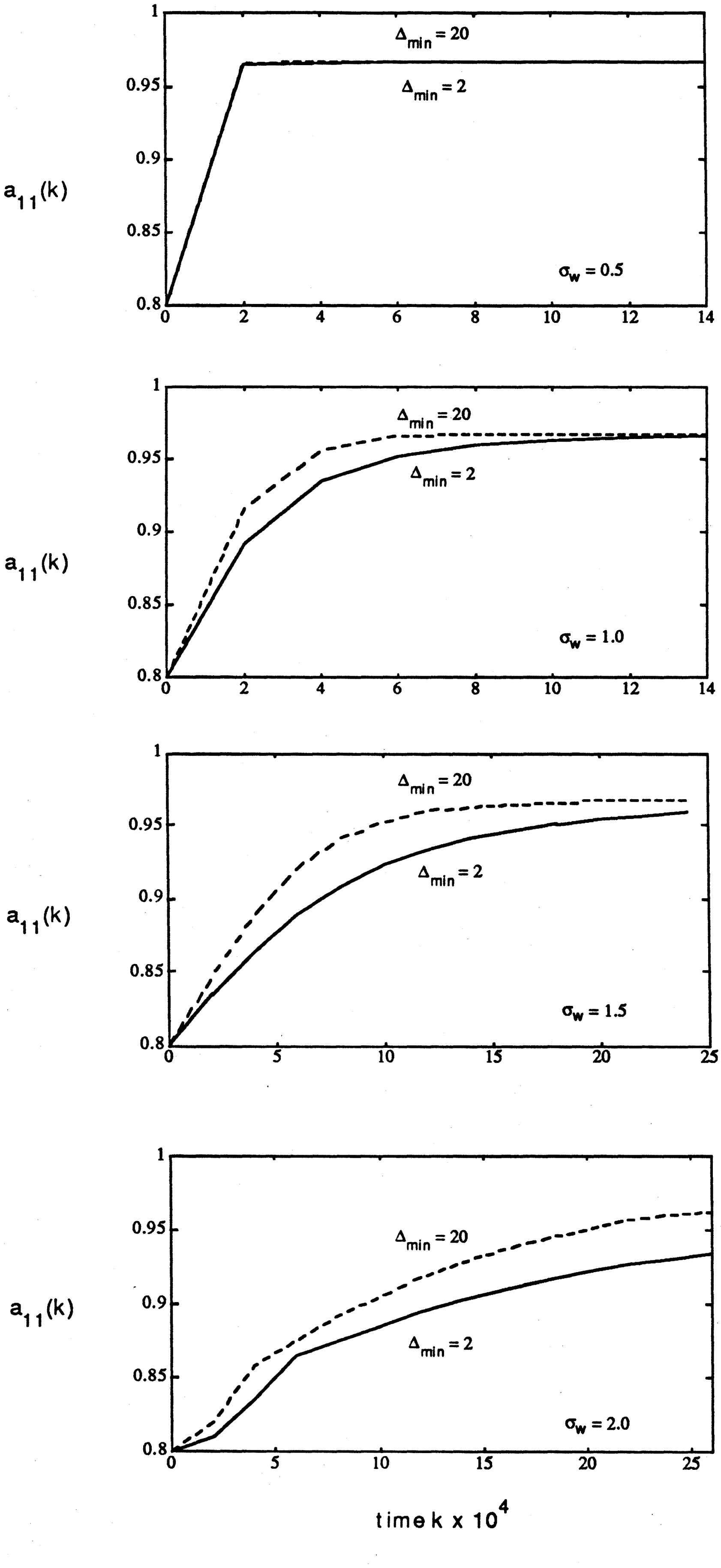


Fig. 7.2 : Effect of sawtooth-lag and noise variance on convergence

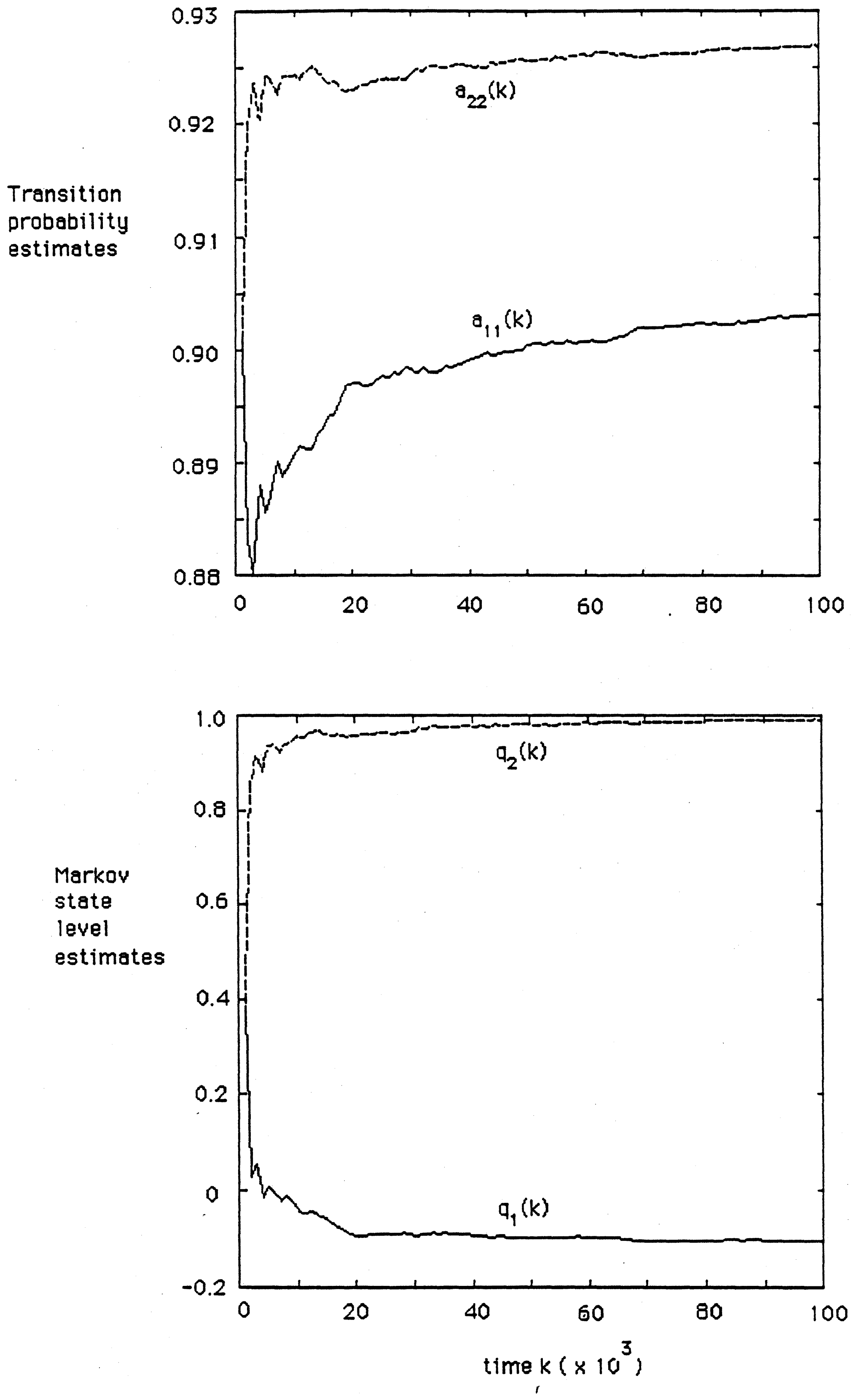
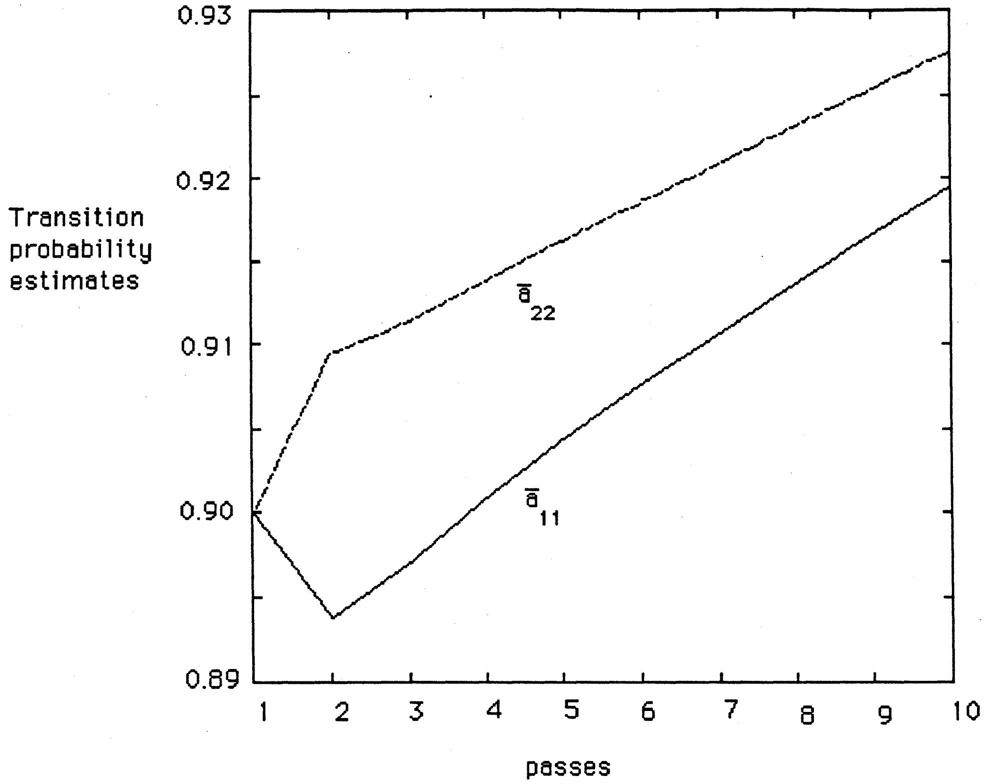


Fig. 7.3: On-line re-estimation scheme performance



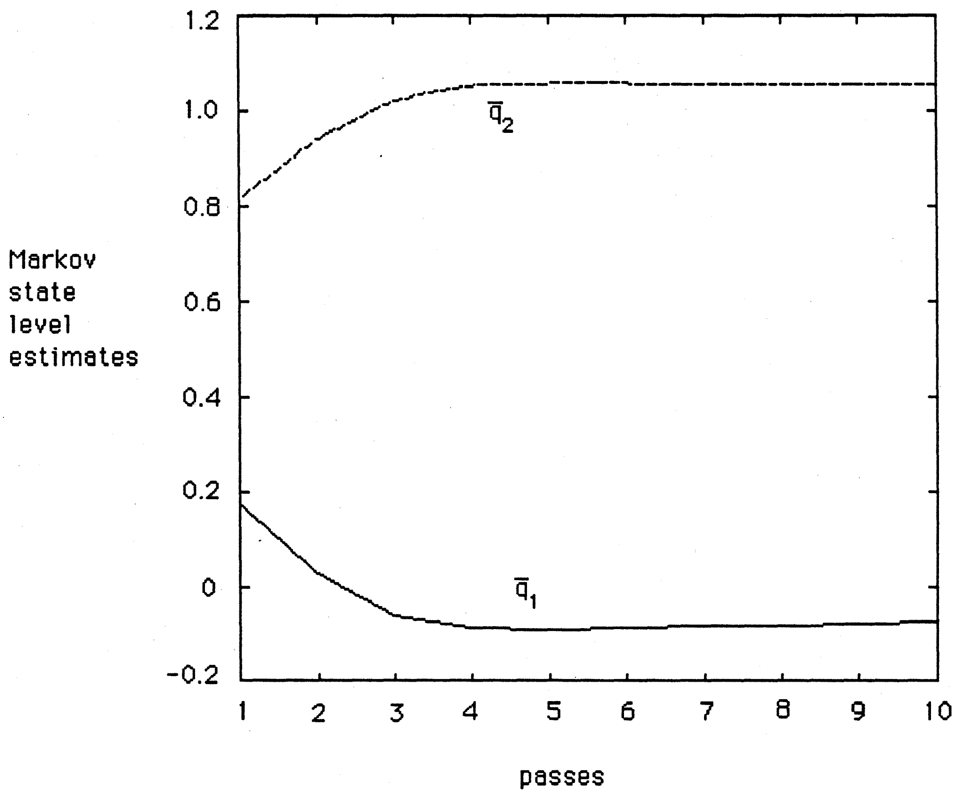


Fig. 7.4: Performance of Baum-Welch re-estimation

Illustration of forgetting: We illustrate the use of forgetting. For variety we work with a 4 state Markov chain.

A four state Markov chain with $a_{ii} = 0.85$ and state levels at 1, 2, 3 and 4 was generated. To this was added WGN with $\sigma_w = 1$ resulting in the observations. Initial estimates were $a_{ii}(1) = 0.9$, $q_i(1) = 1,\ 1.5,\ 2,\ 2.5$. Fig.7.5 shows the estimates of the levels versus time with and without forgetting. The estimates were plotted at 50000 point intervals.

In the scheme with forgetting, the denominators of (7.18) and (7.19) were divided by 500 after every 50000 points. Notice that the estimates with forgetting are very close to the true values after 50000 points. The estimates without forgetting after 750,000 points are still not close to the true values. (Actually they approach the true value after 3 million points). This shows that forgetting can significantly speed up convergence.

Effect of initial estimates: Extensive simulations have been carried out to ascertain the effect of initial estimates on the performance of the proposed on-line scheme. We have tested the on-line scheme over the following range of models, assuming $q_i = i$:

$N = 2, 3, 4$, $a_{ii} = 0.5$ to 0.99, σ_w upto 2

and over the following range of initial conditions:

$a_{ii}(1) = 0.1$ to 0.99

$q_i(1)$ upto $3\,\sigma_w$ away from true values q_i.

In all cases over this range of models and initial conditions, the on-line scheme yielded estimates that converged to the true model. Of course, when σ_w is large convergence is slow and so longer data sequences are required to check for convergence. The initial conditions (particularly level estimates) do however appear to have some effect on convergence rates as described below.

Fig.7.6 shows the effect of 5 different initial estimates on the on-line scheme for the same data as in the previous simulation example. To speed up convergence, the same forgetting scheme as in the previous simulation example was used. The initial estimates are:

Fig.7.6a: $a_{ii}(1) = 0.9$, $\mathbf{q}(1) = (0.5, 1, 1.5, 2)$

Fig.7.6b: $a_{ii}(1) = 0.9$, $\mathbf{q}(1) = (0, 0.5, 1, 1.5)$

Fig.7.6c: $a_{ii}(1) = 0.9$, $\mathbf{q}(1) = (-2, -1.5, -1, -0.5)$

Fig.7.6d: $a_{ii}(1) = 0.9$, $\mathbf{q}(1) = (0, 0.05, 0.1, 0.15)$

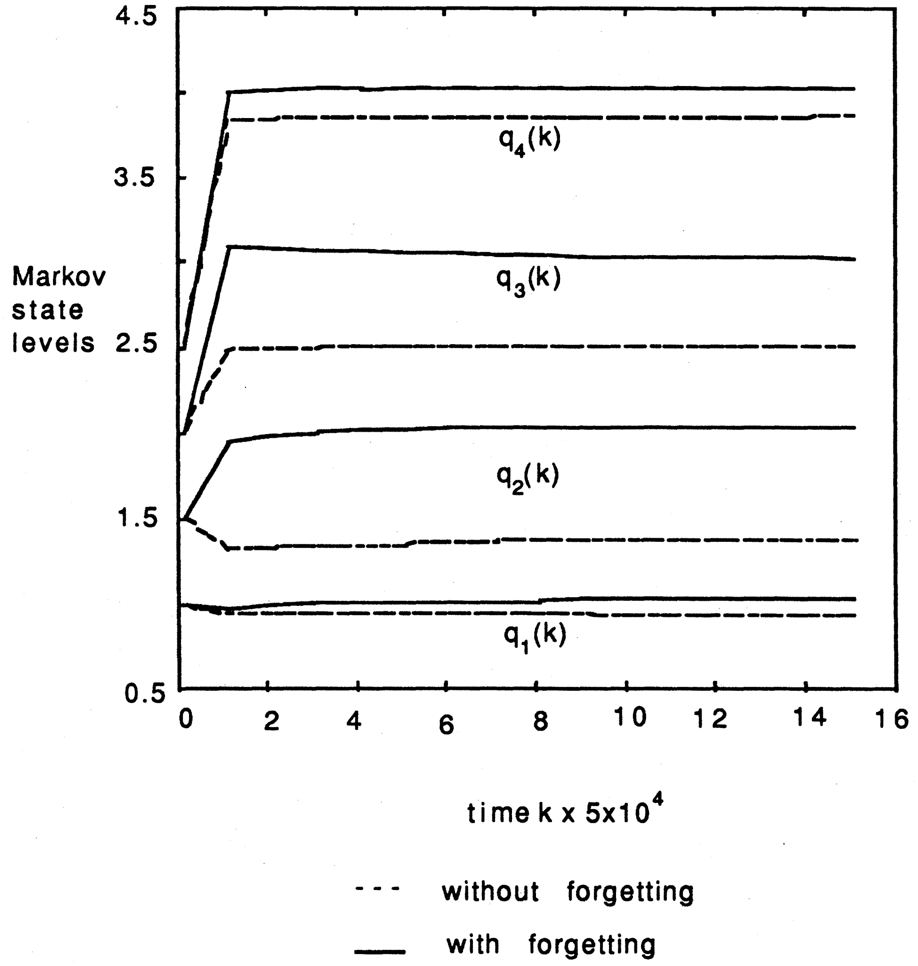


Fig. 7.5 : Level learning with and without forgetting

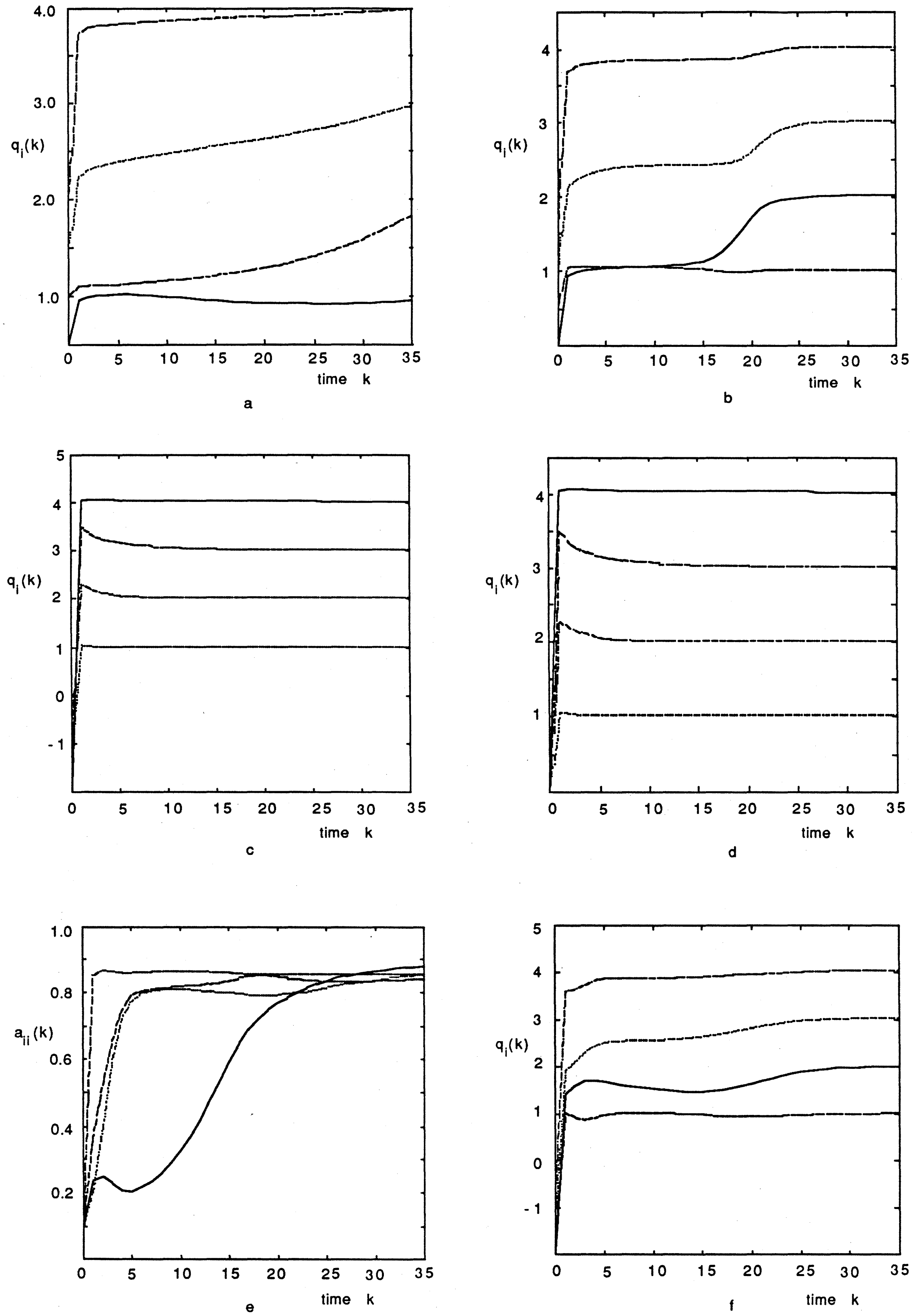


Fig. 7.6 : Effect of initial estimates

Fig.7.6e and Fig.7.6f: $a_{ii}(1) = 0.1$, $\mathbf{q}(1) = (-2, -1.5, -1, -0.5)$.

Fig.7.6a to Fig.7.6d show the state level estimates $\mathbf{q}(k)$ for the above initial conditions. The estimates are plotted every 5×10^4 points and the time scale is in units of 5×10^4. Notice that when the initial estimates are chosen such that more than one initial value is close to a particular state then the convergence can slow down since both the estimates may track the same state level for some time (see Fig.7.6a and especially Fig.7.6b). Nevertheless, the level estimates converge to the true values. Similarly, the transition probability estimates converge to the true values (not shown in the Figs.)

Fig.7.6e shows the transition probability estimates when the initial estimates are chosen at some distance from the true value $a_{ii} = 0.1$. Again the probabilities converge to the true values. Also the level estimates converge (Fig.7.6f).

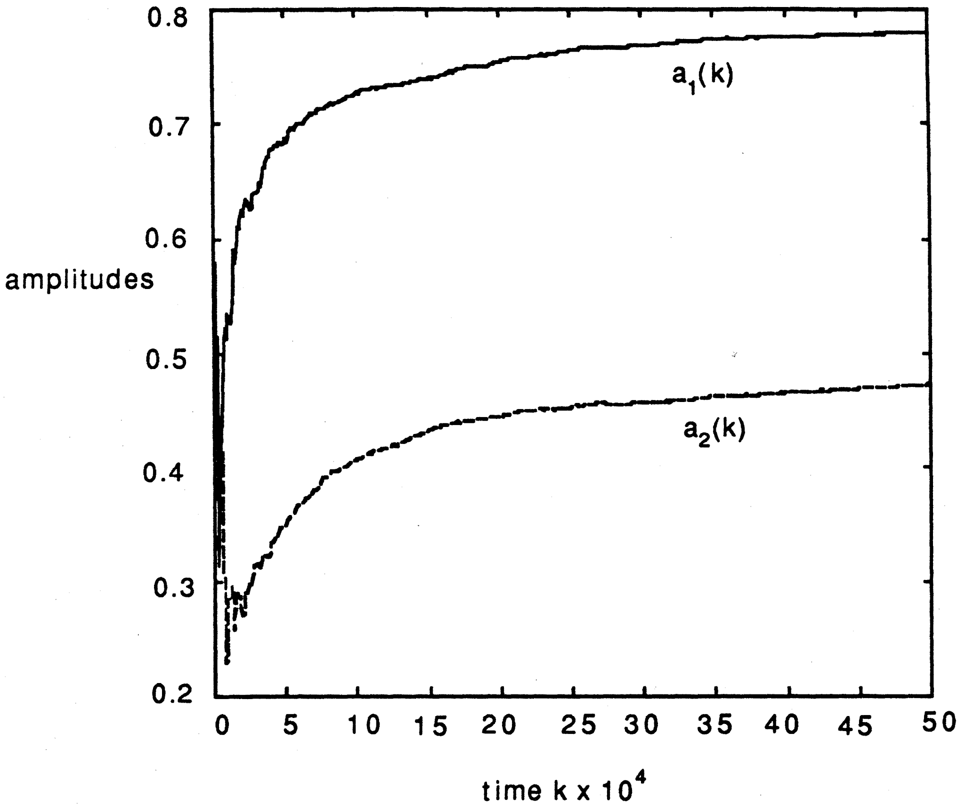
7.6.2 Markov Signal imbedded in White noise and Periodic interference

Here we illustrate the performance of the proposed schemes when the Markov chain is imbedded in periodic interference in addition to WGN.

Illustration of proposed scheme: The purpose of this example is to show that the proposed algorithm satisfactorily learns the Markov state levels, transition probabilities, and also amplitude and phase components of the periodic signal (with known frequency components) from the sum of the Markov signal, periodic signal and WGN of known variance.

A 40000-point 2 state Markov chain was generated with $a_{ii} = 0.97$ and $q_1 = 0$, $q_2 = 1$. To this chain was added zero mean WGN with $\sigma_w = 0.5$. Also a periodic interference $a_1 \sin(w_1 k + \phi_1) + a_2 \sin(w_2 k + \phi_2)$ was added where $\Theta = (a_1, a_2, \phi_1, \phi_2) = (0.8, \pi/3, 0.5, 0)$. Initial Markov estimates were taken as $q_1(1) = -0.1$, $q_2(1) = 0.6$ and $a_{ii}(1) = 0.90$. Also the initial estimate $\Theta_1 = (0.5, 0.5, 0.3, 0.3)$.

Fig.7.7 shows the amplitude and phase estimates plotted versus time. Notice that after 50,000 points the amplitudes and phases are very close to the true values. At $k = 50000$, the state level estimates were $q_1(k) = 0.021, q_2(k) = 1.003$ and the transition probability estimates were $a_{11}(k) = 0.963, a_{22}(k) = 0.967$ which are also close to the true values. Thus the proposed schemes satisfactorily cope with the presence of periodic interferences.



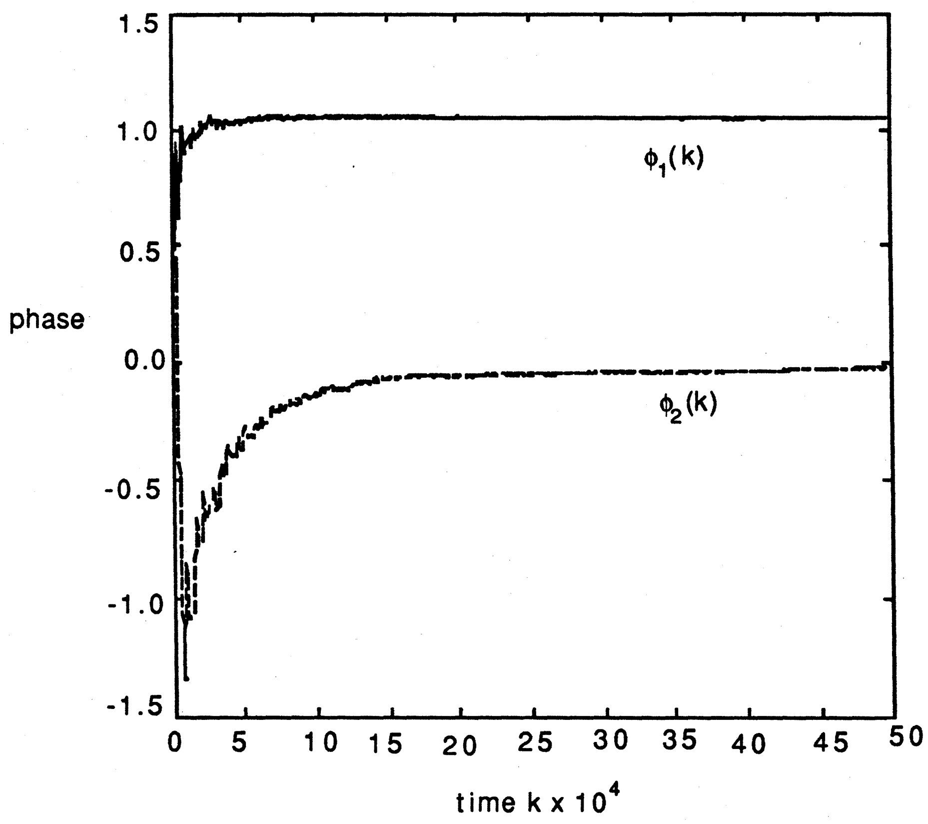


Fig. 7.7 : On-line estimation of phases and amplitudes of periodic interference

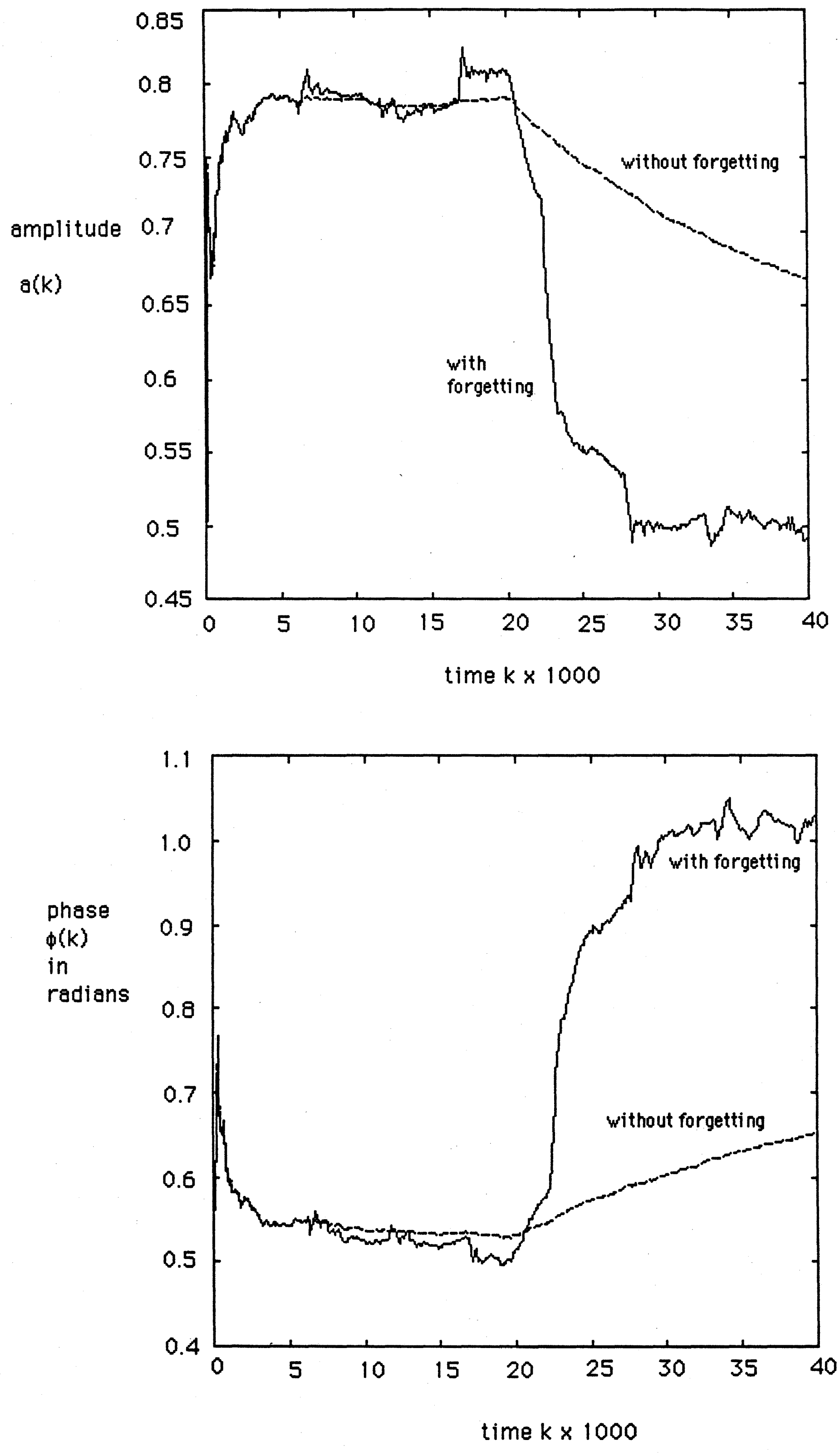


Fig. 7.8 : On-line estimation of periodic interference with jump changes in phase and amplitude

Tracking sinusoid with jump change in phase and amplitude: Here we consider a Markov chain imbedded in sinusoidal interference with jump changes in amplitude and phase in addition to white noise. We illustrate the performance of the proposed on-line schemes in tracking the amplitudes and phases.

To the Markov chain of the previous example was added sinusoidal interference $a(k)\sin(wk+\phi(k))$ where
$a(k) = 0.8$ and $\phi(k) = \pi/6$ for $1 \leq k \leq 20000$
$a(k) = 0.5$ and $\phi(k) = \pi/3$ for $20001 \leq k \leq 40000$.
Then WGN with $\sigma_w = 0.5$ was added to yield the observations.

We ran our standard on-line scheme (without forgetting) and the on-line scheme with forgetting on the data. In the scheme with forgetting, when the denominator of (7.33) exceeded 3000, the denominators of (7.33) and (7.34) were reset to a tenth of their value. Initial estimates were taken as $a_{ii}(1) = 0.8$, $q_1(1) = 0$, $q_2(1) = 0.6$. The initial amplitude and phase were chosen as 0.5 and 0.5 radians respectively.

Fig. 7.8 shows how the amplitude and phase are tracked versus time. Notice that our standard scheme learns the amplitude and phase for $k < 20000$. Then it slowly starts moving towards the new amplitude and phase. In contrast with forgetting the convergence is fast to the new amplitude and phase. Of course with forgetting the estimates drift a little due to the local effects. For the standard on-line scheme, the the state level estimates at $k = 50000$ were $q_1(k) = -0.005$, $q_2(k) = 1.004$ and the transition probabilities were $a_{11}(k) = 0.961$, $a_{22}(k) = 0.964$. For the on-line scheme with forgetting at $k = 50000$, $q_1(k) = -0.007$, $q_2(k) = 1.007$, $a_{11}(k) = 0.965$, $a_{22}(k) = 0.968$.

7.7 Conclusions

In this chapter we have proposed robust, memory efficient, fixed-lag and sawtooth-lag smoothing schemes. Moreover, we have proposed and validated on-line variations of the Baum-Welch re-estimation formulae based on fixed-lag and sawtooth-lag smoothed estimates. Simulations confirm that significant improvements in convergence when these on-line schemes are used.

As an application of the proposed schemes we have obtained on-line estimates of the Markov signal model and periodic signal from a mixture of the Markov process, periodic signal and additive white Gaussian noise. With the on-line re-estimation it is possible to deal with sinusoidal disturbances with slowly varying amplitudes, and phases.

In future work we hope to develop on-line versions of the modified EM algorithm suggested in [77] which results in significantly improved rates of convergence for the off-line case.

Chapter 8

Conclusions, Research in progress and Scope for further Research

This thesis studied estimation techniques for systems characterized by continuous and discrete-state Hidden Markov Models (HMMs).

Under continuous-state models, new schemes were proposed for estimating ARMAX systems with jump changes in parameters and order using Kalman filtering type schemes. Also techniques for estimating ARMAX systems using modified versions of Extended Least Squares (ELS) schemes were proposed.

Under discrete-state models, maximum likelihood (ML) estimates of the HMMs were obtained using the Expectation-Maximization (EM) algorithm. The HMMs we considered included Hidden semi-Markov Models e.g., Hidden Fractal Models, Hidden semi-Markov Models with exponentially decaying states and HMMs with unknown deterministic disturbances in addition to white noise. Also adaptive on-line estimation schemes were proposed for HMMs.

In this chapter we summarize some of the important contributions of this thesis. Also preliminary results from research topics motivated by the previous chapters are discussed. Then directions for further research are highlighted.

8.1 Details of Contributions

We detail the contributions of this thesis under theoretical studies, new algorithms and practical applications.

8.1.1 Theoretical Studies

The following are the theoretical contributions of this thesis:

- Chapter 2 considered a Kalman filter based scheme called the Perturbed Kalman Filter (PKF) for estimating ARMAX systems with jump parameter and order changes. Theorem 2.1 proved that the trace of the covariance matrix P_k of the proposed perturbed Kalman filter varied approximately linearly with time k and that the change in order of the system could be computed from the slope of the trace of P_k versus k. These results were crucial for the successful operation of the PKF.

- We summarize the contributions of Chapter 3 for relaxing the Strictly Positive Real (SPR) condition in Extended Least Squares (ELS) Schemes.

  1. Continuous-time relaxed Strictly Positive Real (SPR) ELS schemes using prefiltering and overparametrization were proposed. Apart from being of theoretical interest, a strong motivation for developing the continuous-time results is their relevance for discrete-time schemes derived from fast sampled continuous-time schemes. The continuous-time results are important in establishing that no insurmountable problems arise should the sampling rate increase.

  2. Discrete-time results were also derived. In the usual ARMAX notation where the noise model is characterized in terms of the polynomial $C(q)$, it was shown that instead of expanding C in the delay operator which is the existing approach, using more suitable operators (such as Laguerre expansions) yields significantly lower order regression vectors with fewer parameters to be estimated.

  3. We specified the overparametrization necessary to satisfy the SPR condition. It was proved that by restricting the zeros of the C polynomial to lie inside a particular compact set, it was possible to specify the degree of overparametrization required and the prefilter so that the SPR condition was satisfied for all polynomial C whose zeros lie inside the set. The compact set we specified was a circle in the s and z planes.

  4. Necessary and sufficient conditions for the persistence of excitation (PE) of the regression vectors using output controllability characterizations for continuous-time and output reachability characterizations for discrete-time were estab-

lished.

5. Convergence rates were established for the proposed ELS schemes and also for a Least Squares scheme that operated in parallel with the ELS scheme to obtain estimates of the ARMAX parameters.

- Also in Chapter 3, necessary and sufficient conditions for the input of a multi-input system to be PE for continuous-time systems were provided. Besides being an important result in its own right, this was used to establish the PE of the regression vector in the proposed continuous-time ELS scheme.

- ML estimation schemes for discrete-state HMMs were derived using the EM algorithm.

1. In Chapter 4, ML estimation schemes for Hidden semi-Markov processes, in particular, fractal processes were derived. Then in Chapter 5, ML estimation of Hidden semi-Markov processes with decaying or filtered states was considered. The basic results were derived by maximizing the conditional expectation of the likelihood of the complete data, $\mathcal{Q}$. Constraints were introduced in the optimization problem when for example it was apriori known that the filters (decay) constants for the various states were identical.

2. ML estimation schemes for Hidden Markov processes imbedded in white noise and deterministic disturbances of known form with unknown parameters were presented in Chapter 6. Basically the known functional form $p_k(\Theta)$ of the deterministic interference was incorporated in the function $\mathcal{Q}$. Then optimizing $\mathcal{Q}$ with respect to Θ yielded the model estimates after each pass. In particular, when the deterministic disturbance was periodic, estimates for the frequencies, phases and amplitudes were obtained apart from estimates of the Markov model. The techniques presented in Chapter 6 can also be used to address the problem of estimating sinusoids in correlated noise, the Markov chain being viewed as correlated noise.

3. Finally, in Chapter 7 on-line estimation schemes were derived which asymptotically yield ML estimates. The basic result used was that increasing $\mathcal{Q}_k$ at each time instant k, also increases the likelihood of λ_k which is the estimated

model at time k. In deriving the schemes, the Fisher information matrix of the complete data was also derived.

8.1.2 Algorithms

The following algorithms were developed in this thesis:

- **Perturbed Kalman Filter (PKF)**: This Kalman filter (KF) based algorithm, developed in Chapter 2, was used for identifying ARMAX systems with jump changes in parameters that include order changes. The order changes were viewed as the introduction of overparametrization. By detecting the degree of overparametrization or equivalently, the change in order of the system, noise was injected into the standard KF algorithm resulting in the PKF. This injection of noise ensured that pole-zero cancellations occurred at the origin and consequently ill-conditioning did not arise. (Of course the PKF specializes to a KF in the absence of overparametrization). Heuristic schemes were proposed for avoiding false alarms. Simulation studies confirmed that the PKF performed as if the jump parameter changes did not include order changes. In comparison, the standard KF did not perform satisfactorily in the presence of order changes.

- **Relaxed SPR ELS algorithm**: This discrete-time algorithm proposed in Chapter 3 is particularly applicable to fast-sampled systems where the zeros of the C polynomial are close to the unit circle. In comparison, other relaxed SPR schemes require an extremely large degree of overparametrization to achieve the SPR condition. A comprehensive design rule for selecting the degree of overparametrization and the prefilter was proposed. Also techniques were proposed for dealing with the case when the zeros of C are spread. A Least Squares scheme was proposed which when run in parallel with the ELS scheme yields the actual ARMAX parameter estimates. ELS algorithms for output-error systems were also proposed. A simulation example was presented for a second order ARMAX system that failed to satisfy the SPR condition. Our relaxed SPR algorithm required an overparametrization of 2 to achieve satisfactory identification whereas an existing algorithm required overparametrization by 6 or larger to achieve satisfactory results.

- **Hidden Fractal Processing algorithm:** This algorithm proposed in Chapter 4, yields ML estimates for fractal stochastic processes imbedded in white noise. Because the transition probabilities of the fractal process cannot be obtained in closed form from the transition rate matrix, it was necessary to do a least squares fit or plot histograms to determine the fractal dimensions. Simulation studies showed that higher order Markov processes could also be used to approximate fractal processes. Aggregation techniques were proposed for decreasing computational complexity and memory requirements. Also, due to the structure of the transition probability matrix **A**, out of the $(N_s\, N_t)^2$ elements only $N_s^2\, N_t$ elements were required to be estimated. So computational and memory requirements were further reduced. Finally maximum aposteriori (MAP) state estimates were also obtained.

- **Semi-Markov Processing with exponential (or other) decay in states:** The ML algorithm proposed in Chapter 5 estimates filtered Markov chains imbedded in white noise. Again state aggregation and the structure of the **A** matrix were exploited to reduce computational and memory requirements. Modifications to the EM algorithm were proposed to obtain faster convergence. When it was apriori known that the decay constants (filters) for the various states were identical, then it became possible to further reduce the number of estimated parameters, resulting in improved model estimates. Simulations presented show that the convergence rates are actually improved when the modified EM scheme was used.

- **Algorithms for eliminating periodic disturbances and polynomial drift:** These ML algorithms were proposed in Chapter 6. In addition to obtaining estimates of the Markov model, they obtain maximum likelihood estimates of the frequencies, phases and amplitudes of the components of the periodic signal. Three possible implementation techniques were suggested: EM, pseudo-EM and Block Component algorithms. Of these the pseudo-EM and the Block Component algorithms were the most convenient to implement. Simulations showed that for phases and amplitudes, only one (global) maxima exists in the conditional expectation of the complete data likelihood function $\mathcal{Q}$. However, to estimate frequencies there were a multitude of local maxima. So for frequency re-estimation, it was necessary to search for the global maxima of $\mathcal{Q}$. Also algorithms for ML estimation of the coefficients of polynomial drift in the Markov states were proposed. Simulation studies were presented which showed that accurate estimates were obtained even when the noise variance was large.

- **On-line algorithms**: On-line HMM estimation algorithms were proposed in Chapter 7. The on-line estimates asymptotically approach the ML estimates. The main advantages of the on-line schemes are: the ability to cope with time varying models, reduced memory requirements from $O(N^2 T)$ to $O(N^2 \Delta)$ in the sawtooth-lag case where $\Delta << T$. Algorithms were also proposed to eliminate periodic disturbances and polynomial drift with time-varying characteristics. Simulations show that they obtain satisfactory estimates even when the noise has a large variance. Also, schemes which incorporated forgetting factors were proposed to speed up convergence.

8.1.3 Applications

- **Neuro-biological applications**: The algorithms proposed for discrete-state HMMs have been applied to extract the statistics of channel currents in cell membranes. Also they have helped provide insight into the modelling of the channel currents and confirm hypothesis about the nature of the currents, e.g., whether they are fractal or homogeneous Markov.

  In Chapter 4, an example was presented for estimating the fractal dimensions of the open and closed states of channel currents from experimental data. The fractal dimension was found to be similar to experimental values obtained by other researchers for channel currents. The techniques proposed in Chapter 4 have also been used to model other time variations in the transition probabilities such as damped sinusoidal variations in the transition probabilities.

  The schemes in Chapter 5 have also been motivated by neuro-biological applications. In particular, spontaneous inhibitory post-synaptic currents generated by the opening of chloride-selective channels show a rapid rise and then decay back exponentially to the ground state. We presented an example in Chapter 5 of extracting such currents from noisy observations. The schemes proposed have also been used to model more complicated resonant dynamics than exponential decays.

- **Removal of deterministic interferences from experimental data:** Segments of data recorded experimentally are frequently corrupted by 'ac hum' from the experimental mains. It is often very expensive to eliminate from the experimental environment. The algorithms proposed in Chapter 6 have been used to remove ac hum from actual experimental data as detailed in simulation example 6.

  Polynomial drift is another commonly found deterministic interference in neurobiological signal processing. A cause for such drift is the gradual change in the liquid junction potential between the recording electrode and the external ionic solution. (Whenever two solutions of different ionic compositions come into contact, a non-zero potential difference is generated). Simulations (not presented in this thesis) have been carried out to successfully eliminate drift in experimental data.

8.2 Research in Progress

We have recently initiated studies in the following two topics which form an interesting sequel to the problems studied in this thesis.

- **Implementing the EM algorithm as a Finite dimensional Filter:** It has been shown in [78] that the EM algorithm for extracting continuous-time Markov chains from noisy observations can be implemented as a finite dimensional filter. These results are extended in [79] and finite dimensional filters and smoothers are obtained which are related to the Wonham filter of a noisily observed Markov chain. The filters and smoothers in [79] are expressed in the Zakai form with the resulting advantage that the equations are linear in the conditional expectations and are driven by the observation process, rather than the innovations.

  The advantages of these schemes are that the forward-backward algorithm is not required. As described in Chapter 7, the forward-algorithm is inherently a fixed-interval smoothing scheme. This means that the algorithm has large memory requirements and cannot be implemented on-line. In contrast, the filtering schemes proposed in [79] can be implemented on-line.

  With the collaboration of Prof. Robert Elliott, the author has conducted preliminary studies into discretizing the Zakai equations of the continuous-time filters and smoothers derived in [79]. The resulting discrete-time filters and smoothers were then used to extract discrete-time Markov chains from white noise. Of course due to the presence of Ito terms in the filtering and smoothing equations, naive discretization of the continuous-time equations can result in large errors in the corresponding discretized equations. So we used the Milshtein approximation [80]. This discretization yields approximations with errors of $O(h)$ where h is the sampling interval.

Simulation results so far have shown that the discretized Zakai equations for estimating the transition probabilities of discrete-time chains yield satisfactory results. In addition the Milshtein approximation removes most of the numerical ill-conditioning problems reported in [69] where a naive discretization of the Wonham filters and smoothers were used.

However, simulations show that the discretized Zakai equations for the state levels (also called the drift coefficients in [79]) of the Markov chain do not yield satisfactory estimates. We have tried using higher-order approximations instead of the Milshtein approximation, e.g., $O(h^{3/2})$ approximations proposed in [81]. The results were still unsatisfactory.

It is interesting though that when the original expression for the level estimates (for which Zakai filters and smoothers are derived in [79]) is discretized, simulation results show that the level estimates are quite satisfactory.

- **ML Estimation of coupled Markov chains in white Gaussian noise:** In collaboration with Dr. R. Kennedy, the author has recently proposed techniques which use the EM algorithm to extract 'coupled', discrete-time Markov chains imbedded in white Gaussian noise. We now briefly describe this problem and the proposed estimation technique.

  By a coupled Markov chain, we mean that the chain is obtained from (eg. by summing) two or more homogeneous Markov chains that are not necessarily independent. Also, we assume that the constituent Markov chains have identical transition probability matrices and state levels.

  In the special case when the constituent chains are independent, the transition probability matrix of the resulting chain, denoted $\mathbf{A}_I$, is the tensor (Kronecker) product of the transition probability matrices (which are assumed identical) of the constituent chains. Also in the special case when the constituent chains are fully coupled, transitions from one state to another occur simultaneously in all the chains. Let us denote the transition probability matrix of the fully coupled chain as $\mathbf{A}_C$.

In general, when the constituent chains are neither independent nor fully coupled, the resulting transition probability matrix can be expressed as

$$\mathbf{A} = (1-\kappa)\mathbf{A}_I + \kappa\,\mathbf{A}_C, \quad 1 \leq \kappa \leq 1. \tag{8.1}$$

Here κ is the coupling factor and is a measure of the degree of independence of the constituent chains. That is, when $\kappa = 0$, the chains are independent and when $\kappa = 1$ the chains are fully coupled.

In principle, the standard Baum-Welch re-estimation equations can be used to obtain maximum likelihood estimates of all the elements of the matrix $\mathbf{A}$ when the chain in imbedded in WGN. However, in practice this is computationally complex and has very large memory requirements since $\mathbf{A}$ in general will be a very large matrix. Notice that because $\mathbf{A}_I$ is obtained as the tensor products of identical matrices, it can be parametrized by a small number of parameters, far fewer than the the actual number of elements of $\mathbf{A}_I$. Similarly, $\mathbf{A}_C$ can also be parametrized by a small number of parameters. This motivates the necessity of developing maximum likelihood techniques which exploit the structure of the $\mathbf{A}$ matrix and estimates its parameters (rather than all its elements) in a efficient manner.

Our proposed technique proceeds as follows: We express $\mathcal{Q}$ (the conditional expectation of the likelihood function of the complete data) in terms of the parametrization of $\mathbf{A}$ instead of in terms of the elements of $\mathbf{A}$. The EM algorithm is then used to estimate the parameters.

Because the state space of the constituent Markov processes are assumed identical, many of the symbol probabilities are equal. Using this fact together with the parametrization of $\mathbf{A}$, it is possible to significantly reduce the computational complexity and memory requirements of the forward-backward algorithm in the E step. The M step involves deriving re-estimation formulae for the parameters of $\mathbf{A}$ and the distinct symbol probabilities.

Preliminary simulations show that these techniques yield satisfactory estimates of the imbedded coupled chain.

Coupled Markov chains are of significant neuro-biological interest. The opening and closing (state) of a single cell channel in a cell membrane is affected by the states of its neighbours. Preliminary qualitative comparisons of the noisy observations of such processes with neuro-biological data from single cell channels show that the two are very similar.

In future work we hope to relate this probabilistic approach to modelling channels in cell membranes to the results of modelling the dynamics of the channels using Lagrangian dynamics.

8.3 Scope for further Research

We now suggest additional problems for further research as a sequel to this thesis.

- **Maximum likelihood ARMAX estimation:** In Chapter 3, we proposed a relaxed SPR ELS scheme for estimating ARMAX systems. It is of considerable interest to compare this with ML techniques for estimating ARMAX systems. Also the EM algorithm could be used in the ML estimation.

  The results of Chapter 6 should provide a starting point for developing such techniques. Recall that in Chapter 3 we mentioned that the SPR condition was not satisfied when deterministic disturbances such as sinusoids are present in addition to white noise. In Chapter 6, we presented a maximum likelihood technique for estimating HMMs in the presence of deterministic interferences and white noise.

- **Filtered Noisy Markov Chains:** Estimation of Hidden semi-Markov processes with exponentially decaying states in Chapter 5 assumed that there was negligible residual memory between transitions. In communication systems, it is often the case that intersymbol-interference is present and so the residual memory is not negligible. It is of interest to develop maximum likelihood schemes which deal with intersymbol interference. In such a case the E step of the EM algorithm is particularly hard to implement. The Mean Field theory from statistical mechanics is one way of obtaining a good approximation to the E step. It is of considerable interest to see if the mean field theory can be used to deal with the case when intersymbol interference is present. Also on-line versions could be developed.

- **On-line Techniques:** In the on-line schemes we have used a stochastic approximation scheme based on the EM algorithm. It is of considerable interest to implement on-line schemes with Newton algorithms to yield significantly faster convergence.

  Recall that the ML estimation of HMMs is a constrained optimization problem. The aim is to maximize the likelihood function with the constraint (in the simplest case) being $\sum_j a_{ij} = 1$. Gradient flow techniques [82] can be used to perform constrained optimizations. It is useful to try and use these gradient flow techniques in ML estimation.

- **Multi-harmonic frequency estimation:** In the extraction in the presence of deterministic interferences we have assumed that the periodic interference can have unrelated frequency components. In many applications e.g., multiharmonic frequency estimation it is apriori known that the frequencies are harmonically related. In such a case this apriori knowledge can easily be incorporated into the optimization problem of Chapter 6 as an additional constraint. It is of interest to compare this technique with other standard techniques for multi-harmonic frequency estimation, e.g., Extended Kalman filtering.

- **Continuous state EM algorithm:** It is worthwhile to examine whether the results in discrete-state HMM processing such as the Baum-Welch scheme carry over to continuous-state HMMs. It would be interesting to compare the results with an Extended Kalman Filter.

- **Discrete-time filters for implementing the EM algorithm:** In Section 8.2 we briefly detailed techniques for implementing the continuous-time EM algorithm as a finite dimensional filter. For dealing with discrete-time chains, the continuous-time equations were discretized. It would be of significant importance to derive discrete-time filters rather than obtain them as a result of approximating continuous-time filters.

Bibliography

[1] B.D.O. Anderson and J.B. Moore, *Optimal Filtering*, Prentice Hall, N.J., 1979.

[2] V. Panuska, *A stochastic approximation method for identification of linear systems using adaptive filtering*, Joint Auto. Control. Conf., Ann Arbor, 1968.

[3] P.C. Young, *The use of linear regression and related procedures for the identification of dynamic processes*, Proc. 7th IEEE Symposium of Adaptive Processes, UCLA, 1968.

[4] L. Ljung and T. Soderstrom, *Thoery and Practice of Recursive Identification*, MIT Press, Cambridge, Mass., 1983.

[5] S.H. Chung, J.B. Moore, L. Xia, L.S. Premkumar and P.W. Gage, *Hidden Markov Model Techniques for extracting small Ionic currents from noise*, Phil. Trans. of Royal Society, Lond. B, 329, pp 265-285, 1990.

[6] S.H. Chung, V. Krishnamurthy, J.B. Moore, *Adaptive Processing techniques based on Hidden Markov Models for characterizing very small channel currents buried in noise and deterministic interferences*, Phil. Trans. of Royal Society, (to appear).

[7] R.A. Howard, *Dynamic Probabilistic Systems*, Vol.1, New York, Wiley, 1971.

[8] R.A. Howard, *Dynamic Probabilistic Systems*, Vol.2, New York, Wiley, 1971.

[9] L.S. Liebovitch, et al, *Fractal model of ion-channel kinetics*, Biochimica et Biophysica Acta, Elsevier, 896, pp 173-180, 1987.

[10] L.S. Liebovitch and J.M. Sullivan, *Fractal Analysis of a voltage-depedent potassium channel from cultured mouse hippocampal neurons*, Biophysical Journal, Biophysical Society, Vol. 52, pp 979-988, Dec. 1987.

[11] J.M. Bekkers, C.F. Stevens, *Presynaptic mechanism for long-term protection in the hippocampus*, Nature 346, pp 724-729, 1990.

[12] G.L. Collingridge, P.W. Gage, B. Robertson *Inhibitory post-synaptic currents in rat hippocampal CA1 neurons*, J. Physiology, Vol. 356, pp 551-564, 1984.

[13] R.J. Sayer, M.J. Friedlander and S.J. Redman, *The time course and amplitude of the EPSPs evoked at synapses between pairs of CA3/CA1 neurons in the hippocampal slice*, Jou. of Neurosci., 10, pp 826-836, 1990.

[14] P.H. Barry and J.M. Diamond, *Junction potentials, electrode standard potentials and other problems in interpreting electrical properties in membranes*, Journal of Membrane Biology, Vol.3, pp 93-122, 1970.

[15] D.M. Titterington, A.F.M. Smith, U.E. Makov, *Statistical Analysis of Finite Mixture Distributions*, Wiley Series in Prob. and Math. Stat., John Wiley and Sons Ltd., 1985.

[16] B.S. Everitt, D.J. Hand, *Finite Mixture Distributions*, Chapman and Hall, London, 1981.

[17] L.E. Baum and T. Petrie, *Statistical inference for probabilistic functions of finite state Markov chains*, Ann.Math.Stat., Vol.37, pp 1554-1563, 1966.

[18] L.E. Baum, T. Petrie, G. Soules and N. Weiss, *A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains*, Ann.Math.Stat., Vol.41, No.1, pp 164-171, 1970.

[19] L.E. Baum, *An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes*, Inequalities, Vol.3, pp 1-8, 1972.

[20] A.P. Dempster, N.M. Laird and D.B. Rubin, *Maximum likelihood estimation from incomplete data via the EM algorithm*, J.R. Statist. Soc. B, Vol. 39, pp 1-38, 1977.

[21] L.R. Rabiner, *A tutorial on Hidden Markov Models and selected applications in speech recognition*, Proc. IEEE, Vol.77, No.2, pp 257-285, 1989.

[22] N.P. Dick and D.C. Bowden, *Maximum likelihood estimation for mixtures of two normal distributions*, Biometriks, Vol. 29, pp 781-790, 1973.

[23] B.C. Peters and H.F. Walker, *An iterative procedure for obtaining maximum likelihood estimates of the parameters for a mixture of normal distributions*, SIAM J. Applied Math., Vol. 35, pp 362-378, 1978.

[24] T.A. Louis, *Finding the observed information matrix when using the EM algorithm*, J. Royal Stat. Soc., vol.44(B), pp 226-233, 1982.

[25] P. Hall, C.C. Heyde, *Martingale limit theory and its application*, Academic Press, N.Y., 1980.

[26] L. Xia, J.B. Moore and M. Gevers, *On adaptive estimation and pole assignment of overparametrized systems*, Int. J. Adaptive Control and Signal Processing, Vol. 1, pp 143-160, 1987.

[27] L. Xia and J.B. Moore, *Recursive identification of overparametrized systems*, IEEE Trans Automatic Control, Vol 34, pp 327-331, 1989.

[28] B.D.O. Anderson, *Adaptive systems, lack of persistency of excitation and bursting phenomena*, Automatica, Vol 21, pp 247-258, 1985.

[29] A.H. Becker Jr., P.R. Kumar and C.Z. Wei, *Adaptive control with the stochastic approximation algorithm: geometry and convergence*, IEEE Trans. Automatic Control, AC-30, pp 330-338, 1985.

[30] H.F. Chen and L. Guo, *Adaptive control with recursive identification for stochastic linear systems*, Controls and Dynamics-Advances in Theory and Applications, Vol 26(2), Academic Press, NY, pp 277-327, 1987.

[31] M. Green and J.B. Moore, *Persistence of excitation in linear systems*, Proc. ACC, Boston, MA, pp. 412-417, 1985.

[32] L. Guo, L.Xia and J.B. Moore, *Tracking randomly varying parameters*, Math. Control, Signals Syst., 1989.

[33] D.Q. Mayne, *Optimal non-stationary estimation of the parameters of a linear system with Gaussian inputs*, J. Electron. Control, Vol 14. pp 101, 1963.

[34] J.B. Moore, *A universality advantage of stochastic excitation signals for adaptive control*, Syst. Control letters, Vol 9, pp 55-58, 1987.

[35] G.C. Goodwin and K.S. Sin, *Adaptive Filtering, Prediction and Control*, Prentice Hall, 1984.

[36] J.B. Moore, *Convergence of Continuous-time Stochastic ELS Parameter Estimation*, Stochastic Processes and their Applications, Vol.27, pp 195-215, 1988.

[37] M. Gevers, V. Wertz, G.C. Goodwin, *A Parameter Estimation Algorithm for Continuous time Stochastic Adaptive Control*, Proc. 27th IEEE Conf. on Decision and Control, pp 1922-1923, Dec. 1988.

[38] J.B. Moore, M. Niedzwiecki, L. Xia, *Identification/Prediction Algorithms for ARMAX models with Relaxed Positive Real Conditions*, Int. J. of Adaptive Control and Signal Processing, Vol.4, No.1, pp 49-68, Feb 1990.

[39] B. Wahlberg, *System Identification using high-order models, revisited*, Technical Report EE/CICS8903, Univ. of Newcastle, 1989.

[40] M. Green, J.B. Moore, *Persistence of Excitation in Linear Systems*, Systems and Control Letters, Vol.7, pp 351-360, 1986.

[41] H. Chen and L. Guo, *Convergence rate of Least Squares Identification and Adaptive control for Stochastic Systems*, Int. J. Contr., Vol.44, pp 1459-1476, 1986.

[42] H. Chen, J.B. Moore, *Convergence rates for Continuous-time Stochastic ELS Parameter Estimation*, IEEE Trans. Auto. Contr., Vol.AC-32, No.3, March 1987.

[43] N. Shimkin, A Feuer, *Persistency of Excitation in Continuous time Systems*, System and Control Letters, Vol.9, No.3, Sept. 1987.

[44] C.T. Chen, *Introduction to Linear Systems Theory*, Holt Rinehart Winston, 1970.

[45] H. Chen, L. Guo, *A Robust Stochastic Adaptive Controller*, IEEE Trans. Auto. Contr., Vol.33, No.11, pp 1035-1043, Nov. 1988.

[46] B. Wahlberg, *Limit results for sampled systems*, Int. J. Control, Vol.48, No.3, pp 1267-83, 1988.

[47] S. Shah, G.F. Franklin, *On satisfying strict positive real condition for convergence by overparametrisation*, IEEE Trans. Auto. Contr., Vol.AC-27, pp 715-16, 1982.

[48] B. Sakmann and E. Naher ed., *Single Channel Recording*, Plenum Press, New York, 1983.

[49] S.J. Korn and R. Horn, *Statistical Discrimination of Fractal and Markov Models of Single-channel Gating*, Biophysical Journal, Biophysical Society, vol. 54, pp 871-877, Nov. 1988.

[50] L.S. Liebovitch, et al, *Ion channel kinetics: a model based on fractal scaling rather than multi-state Markov processes*, Math. Biosci., Vol. 84, pp 37-68, 1987.

[51] B.B. Mandelbrot, *The Fractal Geometry of Nature*, W.H. Freeman and Co. Publishers, San Francisco, 1983.

[52] S.E. Levinson, L.R. Rabiner and M.M. Sondhi, *An introduction to the application of the theory of probabilistic functions of a Markov process to automatic speech recognition*, The Bell System Tech. J., No.6, pp 1035-1074, 1983.

[53] D.R. Cox, *Renewal Theory*, Science Paperbacks, London, 1962.

[54] J.G. Kemeny and J.L. Snell, *Finite Markov Chains*, Van Nostrand, N.Y., 1960.

[55] O.B. McManus, et al., *Fractal models are inadequate for the kinetics of four different channels*, Biophysical Journal, Biophysical Society, Vol. 54, pp 859-870, Nov. 1988.

[56] V. Krishnamurthy, J.B. Moore, S.H. Chung, *On Hidden Fractal Model Signal Processing*, to appear in Signal Processing, 1991.

[57] J. Goutsias, J.M. Mendel, *Optimal Simultaneous Detection and Estimation of filtered Discrete Semi-Markov Chains*, IEEE Trans. Info. Theory, Vol.34, No.3, pp 551-568, May 1988.

[58] R. Godfrey, F. Muir, F. Rocca, *Modelling seismic impedance with Markov chains*, Geophys., Vol.45, pp 1351-1372, 1980.

[59] S. Benetto, E. Biglieri, V. Castellani, *Digital Transmission Theory*, Prentice Hall, N.J, 1987.

[60] J.E. Perkins, I.M.Y. Mareels, J.B. Moore, *Functional Learning in Signal Processing via Least Squares*, ISSPA, Gold Coast, pp 263-266, Aug. 1990.

[61] J. Imae, L. Irlicht, G. Obinata, J.B. Moore, *Enhancing Optimal Controllers via techniques from Robust Adaptive Control*, Dec. 1990.

[62] G.D. Forney, Jr., *The Viterbi algorithm*, Proc. IEEE, Vol.61, No.3, pp 268-278, 1973.

[63] J.K. Omura, *On the Viterbi decoding algorithm*, IEEE Trans. Info. Theory, Vol.IT-15, pp 177-179.

[64] O. Paiss, *Elimination of Exponential interference from Finite-Length Discrete Signals*, IEEE Trans. ACSSP, Vol.30, No.12, pp 2189-2191 Dec.1990.

[65] D.W. Hosmer, *A comparison of iterative maximum likelihood estimates of the parameters of a mixture of two normal distributions under three different types of sample*, Biometrics, Vol.29, pp 761-770, 1973.

[66] K.M. Leytham, *Maximum likelihood estimates for the parameters of mixture distributuins*, Water Resour. Res., Vol.20, pp 896-902.

[67] J. Goutsias, J.M. Mendel, *Optimal Simultaneous Detection and Estimation of Filtered Discrete Semi-Markov Chains*, IEEE Trans. Info. Theory, Vol.34, No.3, pp 551-568, May 1988.

[68] J.B. Moore, *Discrete-time fixed-lag smoothing algorithms*, Automatica, vol.9, no.2, pp 163-174, 1973.

[69] D. Clements and B.D.O. Anderson, *A nonlinear Fixed-Lag smoother for Finite-state Markov Processes*, IEEE Trans. Info. Theory, Vol.IT-21, No.4, pp 446-452, July 1975.

[70] W.M. Wonham, *Some applications of stochastic differential equations to optimal nonlinear filtering*, SIAM J. Contr., Vol.2, pp 347-369, 1965.

[71] V. Krishnamurthy, J.B. Moore, S.H. Chung, *Hidden Markov Model Signal Processing for eliminating unknown deterministic interferences*, submitted to IEEE Trans. Auto. Control, 1991.

[72] D.M. Titterington, *Recursive parameter estimation using Incomplete data*, J.R. Stat. Soc. B, Vol.46, No.2, pp 257-267, 1984.

[73] J. Behboodian, *Information matrix for a mixture of two normal distributions*, J. Stat. Comp. Simul., 1, pp 295-314, 1972.

[74] M.V. Eyuboglu and S.V.H. Qureshi, *Reduced state sequence estimations with set partitioning and decision feedback*, IEEE Trans. Comm., No.36, pp 12-20, 1988.

[75] A. Duel-Hallen and C. Heegard, *Delayed decision-feedback sequence estimation*, IEEE Trans. Comm., No.37, pp.428-436, 1989.

[76] E. Wienstein, M.Feder, A.V. Oppenheim, *Sequential Algorithms for parameter estimation based on the Kullback-Leibler Information measure*, IEEE Trans. ASSP, vol.38, No.9, Sept. 1990.

[77] I. Meilijson, *A fast improvement of the EM algorithm on its own terms*, J. Royal. Stat. Soc., Vol.51, No. 1, Ser. B, 1989.

[78] O. Zeitouni and A. Dembo, *Exact filters for the estimation of the number of transitions of finite-state continuous-time Markov processes*, IEEE Trans. Information Theory, Vol. IT-34, pp 890-893, 1988.

[79] R.J. Elliott, *New Finite Dimensional Filters and Smoothers Related to the Wonham Filter*, submitted to IEEE Trans. Information Theory, 1991.

[80] G.N. Milshtein, *Approximate integration of stochastic differential equations*, Theory Prob. Appl., Vol. 19, pp 557-562, 1974.

[81] N.J. Newton, *An Asymptotically Efficient Difference Formula for Solving Stochastic Differential Equations*, Stochastics, Vol. 19, pp 175-206, 1986.

[82] R.W. Brockett, *Dynamical Systems that sort lists, diagonalize matrices and solve linear programming problems*, Proc. of 27th CDC, Austin, Texas, pp 799-803, Dec. 1988.