VECTOR FORM EM AND SUBOPTIMAL JOINT STATE AND PARAMETER ESTIMATION

Weichang Li

Massachusetts Institute of Technology/
Woods Hole Oceanographic Institution
lwc@mit.edu

James C. Preisig

Dept. of Applied Ocean Physics and Engineering
Woods Hole Oceanographic Institution
jpreisig@whoi.edu

ABSTRACT

The expectation maximization (EM) algorithm combined with the Kalman filter (KF) can be applied iteratively to yield state estimates and ML estimates of the parameters of a linear dynamical system. In this paper new recursive forms of the accumulated second-order state moments which constitute the core of the joint state and parameter estimator are derived. The new recursions are in vector form and follow directly from the state smoothing formula as well as the properties of the Kronecker product. By loosening the optimal constraints, the new recursion forms lead to efficient suboptimal algorithms that are time-recursive. Convergence control of these recursive algorithms by exponential weighting is also considered.

1. INTRODUCTION

State estimation in linear dynamical systems with unknown parameters is a problem of practical interest in numerous applications. In many cases the parameters of the dynamic model for a real system are not known exactly and need to be estimated. In [1], an EM algorithm combined with the Kalman state smoother was proposed to compute the ML estimates of the system parameters while also providing the state estimates. Later in [2] the idea was applied to the problem of speech enhancement where the authors also suggested a suboptimal sequential algorithm in which the state smoother is replaced by a filter. In all these cases the parameter estimator is based on sums of the second-order state moments which must be computed individually (and in a forward-backward scheme when a state smoother is used) and stored. Recently in [3], a class of finite-dimensional filters were proposed to directly estimate the sums of those second-order state moments, instead of estimating the moments first and then accumulating. This greatly reduces the storage requirement and is computationally more efficient. The derivation was based on the notion of measure change. In [4], the same filters were employed in a suboptimal recursive algorithm whose convergence was proved under certain conditions. However, the recursion forms proposed in [3] have to be implemented element-wise for the accumulated moments, and the suboptimal algorithm of [4] can have a very slow convergence rate under certain conditions. A new vector form recursion scheme is presented in this paper. The derivation is based on the state smoothing formula and the properties of the Kronecker product, and is mathematically simpler than that in [3]. The new recursion forms also yield new insight into the stability of the recursions and motivates suboptimal recursive algorithms.

The paper is organized as follows: section 2 formulates the problem and reviews the EM algorithm; in section 3 the new recursion forms of the accumulated moments are derived. Section 4 presents two suboptimal algorithms. Numerical results are presented in section 5 and finally a conclusion follows in section 6.

Throughout the paper, the following notation is used: superscripts *, ^{T} and ^{h} stand for complex conjugate, transpose and Hermitian, respectively. $\mathbf{I}$ and $\mathbf{0}$ denote identity matrices and zero matrices (or vectors) of appropriate size. $Vec(\mathbf{A})$ denotes the vector obtained by stacking all columns of the matrix $\mathbf{A}$, and $Mat(\cdot)$ the inverse operator of Vec. $\otimes$ denotes the Kronecker product.

2. PROBLEM FORMULATION

Consider the following state-space model:

$$\mathbf{x}_{k+1} = \mathbf{A}(\theta)\mathbf{x}_k + \mathbf{w}_k \tag{1a}$$

$$y_k = \mathbf{c}_k\mathbf{x}_k + v_k \tag{1b}$$

where $\mathbf{x}_k \in \mathbb{C}^M$ denotes the state process. The state transition matrix $\mathbf{A}(\theta) \in \mathbb{C}^{M\times M}$ is parameterized by the unknown set θ. y_k is the scalar observation. Both $\mathbf{w}_k$ and v_k are zero-mean Gaussian white processes, with covariances $\mathbf{Q}_w$ and σ_v^2, respectively. They are mutually and serially independent, and independent from $\mathbf{x}_0$.

The goal is to estimate the state process $\mathbf{x}_n$ based on the observation sequence $\mathcal{Y}_n = \{y_1, \cdots, y_n\}$ assuming $\mathbf{c}_k$, $k = 1, \cdots, n$ are known. The parameter set θ also needs to be estimated. Therefore it may be viewed as a problem of joint state and parameter estimation.

According to [1] (and [5] for details on EM), given $\mathcal{Y}_n$, at the lth iteration the EM parameter estimation yields

$$\widehat{\theta}_n^{(l)} = arg\ \min_{\theta}\ J(\theta) \tag{2a}$$

$$J(\theta) \triangleq E\left\{\sum_{i=1}^{n} \|\mathbf{x}_i - \mathbf{A}(\theta)\mathbf{x}_{i-1}\|^2_{Q_w^{-1}} \middle| \mathcal{Y}_n; \widehat{\theta}_n^{(l-1)}\right\} \tag{2b}$$

As a generalization that will become useful later, let

$$J_\lambda(\theta) \triangleq E\left\{\sum_{i=1}^{n} \lambda^{n-i} \|\mathbf{x}_i - \mathbf{A}(\theta)\mathbf{x}_{i-1}\|^2_{Q_w^{-1}} \middle| \mathcal{Y}_n; \widehat{\theta}_n^{(l-1)}\right\} \tag{3}$$

where $0 < \lambda \leq 1$. Then $J_\lambda(\theta) = J(\theta)$ when $\lambda = 1$.

Consider the case $\theta = Vec(\mathbf{A})$. Solving $\partial J_\lambda(\theta)/\partial\theta = \mathbf{0}$ yields

$$\widehat{\mathbf{A}}_n^{(l)} = \widehat{\mathbf{H}}_1^{(l-1)}[n]\big[\widehat{\mathbf{H}}_0^{(l-1)}[n]\big]^{-1} \tag{4}$$

where it can be shown that

$$\begin{aligned}\widehat{\mathbf{H}}_0^{(l-1)}[n] &\triangleq E\left\{\sum_{i=1}^{n} \lambda^{n-i}\mathbf{x}_{i-1}\mathbf{x}_{i-1}^h \middle| \mathcal{Y}_n; \widehat{\mathbf{A}}_n^{(l-1)}\right\} \\ &= \sum_{i=1}^{n} \lambda^{n-i}\big[\widehat{\mathbf{x}}_{i-1|n}^{(l-1)}\big(\widehat{\mathbf{x}}_{i-1|n}^{(l-1)}\big)^h + \mathbf{P}_{i-1|n}^{(l-1)}\big]\end{aligned} \tag{5a}$$

$$\begin{aligned}\widehat{\mathbf{H}}_1^{(l-1)}[n] &\triangleq E\left\{\sum_{i=1}^{n} \lambda^{n-i}\mathbf{x}_{i}\mathbf{x}_{i-1}^h \middle| \mathcal{Y}_n; \widehat{\mathbf{A}}_n^{(l-1)}\right\} \\ &= \sum_{i=1}^{n} \lambda^{n-i}\big[\widehat{\mathbf{x}}_{i|n}^{(l-1)}\big(\widehat{\mathbf{x}}_{i-1|n}^{(l-1)}\big)^h + \mathbf{P}_{i|n}^{(l-1)}\big(\mathbf{J}_{s,i}^{(l-1)}\big)^h\big]\end{aligned} \tag{5b}$$

(5a) -(5b) will be called the *filtered estimate of the accumulated state moments* with lag $j = 0$ and 1, respectively. Here $\widehat{\mathbf{x}}_{i|n}^{(l-1)}$ and $P_{i|n}^{(l-1)}$ are the smoothed state estimate and the error covariance, respectively. In addition,

$$\begin{aligned}\mathbf{J}_{s,i}^{(l-1)} &\triangleq \mathbf{P}_{i-1|i-1}^{(l-1)}\big(\widehat{\mathbf{A}}_n^{(l-1)}\big)^H\big(\mathbf{P}_{i|i-1}^{(l-1)}\big)^{-1} &(6a)\\ &= \big(\widehat{\mathbf{A}}_n^{(l-1)}\big)^{-1}\big[\mathbf{I} - \mathbf{Q}_w\big(\mathbf{P}_{i|i-1}^{(l-1)}\big)^{-1}\big] &(6b)\end{aligned}$$

with $\mathbf{P}_{i-1|i-1}^{(l-1)}$ and $\mathbf{P}_{i|i-1}^{(l-1)}$ being the error covariance matrices of the filtered and predicted state estimates, respectively. $\mathbf{J}_{s,i}^{(l-1)}$ is the closed-loop state matrix of the smoothed estimator [6]. Here the n dependencies in $\mathbf{J}_{s,i}^{(l-1)}$, $\mathbf{P}_{i-1|i-1}^{(l-1)}$ and $\mathbf{P}_{i|i-1}^{(l-1)}$ have all been dropped for notational simplicity. Intuitively, (4)-(5b) may be viewed as estimating $\mathbf{A}$ from *observations* $\widehat{\mathbf{x}}_{i|n}$ of which the uncertainty is accounted for by the second terms in (5a) and (5b).

The algorithm proposed in [1] alternates between (4)-(6) and the Kalman state smoother, and iteratively it yields parameter and state estimates. The problem is that it requires increasingly large storage space. In the next section a new recursive algorithm is derived that updates $\widehat{\mathbf{H}}_i^{(l-1)}[n]$ directly in vector form.

3. RECURSION OF THE ACCUMULATED STATE MOMENTS

The basic idea of the new recursion scheme is to compute $\widehat{\mathbf{H}}_j^{(l-1)}[n]$ directly, which is similar to that of [3]. However, unlike in [3] where the algorithm is derived via measure changes and updates $\widehat{\mathbf{H}}_j^{(l-1)}[n]$ elementwise, the new recursion here is in vector form and follows directly from the state smoothing formula and the properties of the Kronecker product. For notational simplicity, in this section the iteration index (l) is dropped.

Note the following fixed-point smoothing formulas ([6]):

$$\begin{cases} \widehat{\mathbf{x}}_{i|n} = \widehat{\mathbf{x}}_{i|n-1} + \mathbf{P}_{i,n}\mathbf{c}_n^h R_{e_n}^{-1} e_n \\ \mathbf{P}_{i|n} = \mathbf{P}_{i|n-1} - \mathbf{P}_{i,n}\mathbf{c}_n^h R_{e_n}^{-1}\mathbf{c}_n\mathbf{P}_{i,n}^h \end{cases} \tag{7}$$

where $n \geq i$; e_n, R_{e_n} are the innovation and its variance; $\mathbf{P}_{i,n} = E\big[\tilde{\mathbf{x}}_{i|i-1}\tilde{\mathbf{x}}_{n|n-1}^h\big]$ is the cross error covariance, and

$$\mathbf{P}_{i,n} = \mathbf{P}_{i|i-1}\big[\mathbf{F}_{p,n-1}\mathbf{F}_{p,n-2}\cdots\mathbf{F}_{p,i}\big]^h \tag{8a}$$

$$\mathbf{F}_{p,i} = \widehat{\mathbf{A}}_n\big[\mathbf{I} - \mathbf{P}_{i|i-1}\mathbf{c}_i^h R_{e_i}^{-1}\mathbf{c}_i\big] \tag{8b}$$

for $n > i$. $\widehat{\mathbf{A}}_n$ is the latest estimate of $\mathbf{A}$; $\mathbf{F}_{p,i}$ is the transition matrix of the one-step state prediction. From (6) it can be shown

$$\mathbf{J}_{s,i+1} = \mathbf{P}_{i|i-1}\mathbf{F}_{p,i}^h\mathbf{P}_{i+1|i}^{-1} \tag{9}$$

Combining (9) with (8a), for $i < n$ it follows

$$\mathbf{P}_{i,n} = \big[\Pi_{j=i}^{n-1}\mathbf{J}_{s,j+1}\big]\mathbf{P}_{n|n-1} \triangleq \mathbf{T}_{i,n}\mathbf{P}_{n|n-1} \tag{10}$$

where $\mathbf{T}_{i,n}$ is clear from the second equality.

Taking Vec on both (5a) and (5b), using (7) as well as the following identities [7]:

$$Vec\big(\mathbf{ABC}\big) = \big(\mathbf{C}^T \otimes \mathbf{A}\big)Vec\big(\mathbf{B}\big) \tag{11}$$

$$\big(\mathbf{AB}\big) \otimes \big(\mathbf{CD}\big) = \big(\mathbf{A} \otimes \mathbf{C}\big)\big(\mathbf{B} \otimes \mathbf{D}\big) \tag{12}$$

it yields the following recursions

$$\begin{aligned}Vec\big(\widehat{\mathbf{H}}_1[n]\big) = {}& \lambda\big[Vec\big(\widehat{\mathbf{H}}_1[n-1]\big) + \mathbf{\Omega}_{1,n}Vec\big(\mathbf{M}_n\big) \\ &+ \mathbf{\Upsilon}_{1,n}\mathbf{N}_n^* + \mathbf{\Gamma}_{1,n}\mathbf{N}_n\big] \\ &+ Vec\big[\widehat{\mathbf{x}}_{n|n}\widehat{\mathbf{x}}_{n-1|n}^h + \mathbf{P}_{n|n}\mathbf{J}_{s,n}^h\big]\end{aligned} \tag{13a}$$

$$\begin{aligned}Vec\big(\widehat{\mathbf{H}}_0[n]\big) = {}& \lambda\big[Vec\big(\widehat{\mathbf{H}}_0[n-1]\big) + \mathbf{\Omega}_{0,n}Vec\big(\mathbf{M}_n\big) \\ &+ Vec\big[Mat\big(\mathbf{\Gamma}_{0,n}\mathbf{N}_n\big)\big]^h + \mathbf{\Gamma}_{0,n}\mathbf{N}_n\big] \\ &+ Vec\big[\widehat{\mathbf{x}}_{n-1|n}\widehat{\mathbf{x}}_{n-1|n}^h + \mathbf{P}_{n-1|n}\big]\end{aligned} \tag{13b}$$

where $\widehat{\mathbf{x}}_{n-1|n}$ and $\mathbf{P}_{n-1|n}$ are computed from $\widehat{\mathbf{x}}_{n-1|n-1}$ and $\mathbf{P}_{n-1|n-1}$ according to (7). In addition, for $j = 0, 1$.

$$\mathbf{M}_n \triangleq \mathbf{P}_{n|n-1}\mathbf{c}_n^h\big[R_{e_n}^{-1}e_n e_n^h R_{e_n}^{-1} - R_{e_n}^{-1}\big]\mathbf{c}_n\mathbf{P}_{n|n-1} \tag{14a}$$

$$\mathbf{N}_n \triangleq \mathbf{P}_{n|n-1}\mathbf{c}_n^h R_{e_n}^{-1} e_n \tag{14b}$$

$$\mathbf{\Omega}_{j,n} \triangleq \Sigma_{i=1}^{n-1}\lambda^{n-1-i}\mathbf{T}_{i-1,n}^* \otimes \mathbf{T}_{i-1+j,n} \tag{14c}$$

$$\mathbf{\Gamma}_{j,n} \triangleq \Sigma_{i=1}^{n-1}\lambda^{n-1-i}\widehat{\mathbf{x}}_{i-1|n-1}^* \otimes \mathbf{T}_{i-1+j,n} \tag{14d}$$

$$\mathbf{\Upsilon}_{j,n} \triangleq \Sigma_{i=1}^{n-1}\lambda^{n-1-i}\mathbf{T}_{i-1,n}^* \otimes \widehat{\mathbf{x}}_{i-1+j|n-1} \tag{14e}$$

with (14c)-(14e) recursively computed as follows($j = 0, 1$):

$$\begin{bmatrix} \mathbf{\Omega}_{j,n+1}^T \\ \mathbf{\Gamma}_{j,n+1}^T \\ \mathbf{\Upsilon}_{j,n+1}^T \end{bmatrix} = \lambda \begin{bmatrix} \mathbf{G}_{n+1}^T & \mathbf{0} & \mathbf{0} \\ \mathbf{NJ}_{n+1}^T & \mathbf{J}_{s,n+1}^T & \mathbf{0} \\ \mathbf{JN}_{n+1}^T & \mathbf{0} & \mathbf{J}_{s,n+1}^h \end{bmatrix} \begin{bmatrix} \mathbf{\Omega}_{j,n}^T \\ \mathbf{\Gamma}_{j,n}^T \\ \mathbf{\Upsilon}_{j,n}^T \end{bmatrix}$$
$$+ \begin{bmatrix} \mathbf{G}_{n+1}^T & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{J}_{s,n+1}^T & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{J}_{s,n+1}^h \end{bmatrix} \begin{bmatrix} \mathbf{J}_{s,n}^h \otimes \mathbf{L}_{j,n}^T \\ \widehat{\mathbf{x}}_{n-1|n}^h \otimes \mathbf{L}_{j,n}^T \\ \mathbf{J}_{s,n}^h \otimes \widehat{\mathbf{x}}_{n-1+j|n}^T \end{bmatrix} \quad (15)$$

where $\mathbf{L}_{0,n} = \mathbf{J}_{s,n}$, $\mathbf{L}_{1,n} = \mathbf{I}$. And $\mathbf{G}_{n+1} = \mathbf{J}_{s,n+1}^* \otimes \mathbf{J}_{s,n+1}$, $\mathbf{NJ}_{n+1} = \mathbf{N}_n^* \otimes \mathbf{J}_{s,n+1}$, $\mathbf{JN}_{n=1} = \mathbf{J}_{s,n+1}^* \otimes \mathbf{N}_n$. The proof is quite straightforward by repeatedly applying (12) hence is not included here. Note that (13) and (15) can be arranged such that the update at time n only needs e_n, $\mathbf{R}_{e_n}$, $\widehat{\mathbf{x}}_{n|n-1}$,$\widehat{\mathbf{x}}_{n-1|n-1}$, $\mathbf{P}_{n|n-1}$, $\mathbf{P}_{n|n}$, all obtainable from the one-step KF update.

Equation (15) has a block lower tridiagonal transition matrix hence it is stable if and only if all the eigenvalues of $\lambda \mathbf{J}_{s,n+1}$ are inside the unit circle. For $\lambda = 1$ this is equivalent to saying that the KF assuming $\mathbf{A} = \widehat{\mathbf{A}}_n^{(l-1)}$ is stable (equivalently $\big(\widehat{\mathbf{A}}_n^{(l-1)}, \mathbf{Q}_w^{1/2}\big)$ and $\big(\widehat{\mathbf{A}}_n^{(l-1)}, \mathbf{c}[n]\big)$ are completely stabilisable and completely detectable, respectively). This follows from the fact that the KF state estimates admit the following recursion

$$\big(\mathbf{P}_{n|n}^{-1}\widehat{\mathbf{x}}_{n|n}\big) = \mathbf{J}_{s,n}^h \big(\mathbf{P}_{n-1|n-1}^{-1}\widehat{\mathbf{x}}_{n-1|n-1}\big) + \mathbf{c}_{n-1}^h \mathbf{Q}_v^{-1} y_n \quad (16)$$

which is stable iff all the eigenvalues of $\mathbf{J}_{s,n}$ are inside the unit circle. To summarize, we have the following theorem:

Theorem 1 *Consider the linear dynamical system* (1)*, suppose Kalman Filter assuming* $\mathbf{A} = \widehat{\mathbf{A}}_n^{(l-1)}$ *is applied to the system and at time* n *generates* $\widehat{\mathbf{x}}_{n|n-1}$, $\widehat{\mathbf{x}}_{n|n}$, $\mathbf{P}_{n|n-1}$, $\mathbf{P}_{n|n}$, e_n, R_{e_n}*, then the accumulated second-order state moments,* $\widehat{\mathbf{H}}_0^{(l-1)}[n]$ *and* $\widehat{\mathbf{H}}_1^{(l-1)}[n]$*, both defined in* (5)*, can be updated recursively according to* (13) *and* (15)*. Furthermore, that* $\big(\widehat{\mathbf{A}}_n^{(l-1)}, \mathbf{Q}_w^{1/2}\big)$ *is completely stabilisable and* $\big(\widehat{\mathbf{A}}_n^{(l-1)}, \mathbf{c}_n\big)$ *is completely detectable is the sufficient (also necessary if* $\lambda = 1$*) condition for* (15) *to be exponentially stable.* The discussion prior to the theorem has essentially established the proof.

The overall EM algorithm is outlined as follows: Consider at time n when the data $\mathcal{Y}_n$ are available,

1. At the lth iteration, let $\mathbf{A} = \widehat{\mathbf{A}}_n^{(l-1)}$.
2. for $t = 1, \cdots, n$, applying Kalman Filter to get $\widehat{\mathbf{x}}_{t|t-1}$, $\widehat{\mathbf{x}}_{t|t}$, $\mathbf{P}_{t|t-1}$, $\mathbf{P}_{t|t}$, e_t, R_{e_t}, and computing $\widehat{\mathbf{H}}_1^{(l-1)}[t]$ and $\widehat{\mathbf{H}}_0^{(l-1)}[t]$ using (13) and (15);
3. Let $\widehat{\mathbf{A}}_n^{(l)} = \widehat{\mathbf{H}}_1^{(l-1)}[n]\big[\widehat{\mathbf{H}}_0^{(l-1)}[n]\big]^{-1}$;
4. Let $l \leftarrow l + 1$, repeat steps 1-3 until $\widehat{\mathbf{A}}_n^{(l)}$ converges.

When the next observation becomes available, let $\widehat{\mathbf{A}}_{n+1}^{(1)} = \widehat{\mathbf{A}}_n^{(l)}$, and repeat the iteration above.

The main advantage of the recursion scheme (13) and (15) is that it can lead to time-recursive suboptimal algorithms as discussed in the next section.

4. SUBOPTIMAL ALGORITHMS

The EM algorithm involves multiple iterations and in each iteration $\widehat{\mathbf{H}}_j^{(l-1)}[n]$, for $j = 0, 1$, must be recursively computed all the way from $\widehat{\mathbf{H}}_j^{(l-1)}[1]$ using the latest parameter estimate. For most online applications this is not feasible. In this section two suboptimal recursive algorithms are proposed based on the following modifications of (3) and (5):
A. Algorithm *subem I*: (for $j = 0, 1$)

$$J_\lambda(\theta) \triangleq \Sigma_{i=1}^n \lambda^{n-i} E\big\{\|\mathbf{x}_i - \mathbf{A}(\theta)\mathbf{x}_{i-1}\|_{Q_w^{-1}}^2 \big| \mathcal{Y}_n; \widehat{\theta}_{i-1}\big\} \quad (17)$$

$$\widehat{\mathbf{H}}_j[n] \triangleq \Sigma_{i=1}^n \lambda^{n-i} E\big\{\mathbf{x}_{i-1+j}\mathbf{x}_{i-1}^h \big| \mathcal{Y}_n; \widehat{\mathbf{A}}_{i-1}\big\} \quad (18)$$

B. Algorithm *subem II*: (17)-(18) with $\mathcal{Y}_n$ replaced by $\mathcal{Y}_i$.

Both algorithms are non-iterative and involve only one-step update of $\widehat{\mathbf{H}}_j[n]$. *subem I* still uses the recursions (13) and (15), but only runs one step from $n-1$ to n. Algorithm *subem II* admits even simpler recursions (for $j = 0, 1$):

$$\widehat{\mathbf{H}}_j[n] = \lambda \widehat{\mathbf{H}}_j[n-1] + E\big\{\mathbf{x}_{n-1+j}\mathbf{x}_{n-1}^h \big| \mathcal{Y}_n; \widehat{\mathbf{A}}_{n-1}\big\} \quad (19)$$

where $\widehat{\mathbf{H}}_j[n-1]$ is not updated by the newest observation and the latest parameter estimate. In both cases, the following recursion holds for $\widehat{\mathbf{A}}_n$:

$$\widehat{\mathbf{A}}_n = \widehat{\mathbf{A}}_{n-1} + \big(\mathbf{L}_1[n] - \widehat{\mathbf{A}}_{n-1}\mathbf{L}_0[n]\big)\widehat{\mathbf{H}}_0^{-1}[n] \quad (20)$$

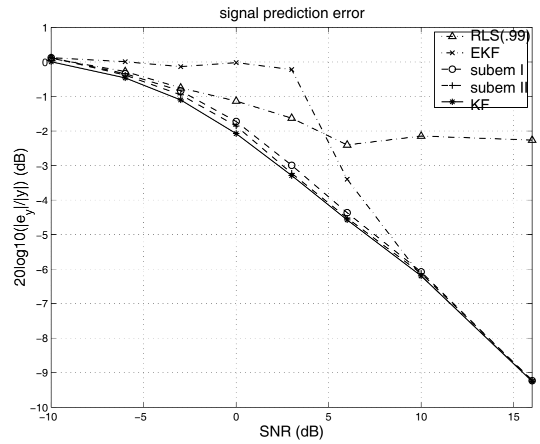
where $\mathbf{L}_i[n] \triangleq \widehat{\mathbf{H}}_i[n] - \lambda \widehat{\mathbf{H}}_i[n-1]$, $i = 0, 1$.

The algorithm of [4] is a special case of *subem I* with $\lambda = 1$, aided with a stable projection of $\widehat{\mathbf{A}}_n$. In [2], exponential weighting was introduced in a suboptimal algorithm to deal with parameter variation. In fact, the role of λ is two-fold. In addition to tracking variable parameters, it can also accelerate the convergence of both algorithms when the parameter is constant, at the price of a higher steady state estimation error. This is demonstrated in the next section.

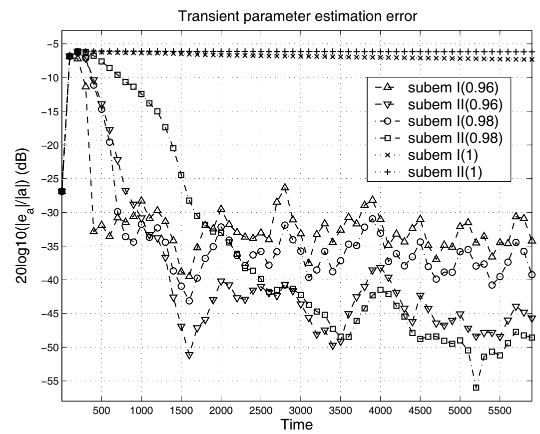
5. NUMERICAL RESULTS

This section presents a numerical example of estimation of state-space model with unknown parameters. Algorithms *subem I*, *II* are compared with the exponentially weighted recursive least squares (EWRLS) algorithm, the extended Kalman filter (EKF) and the Kalman filter. The data was generated according to model (1) such that $\mathbf{A}$ is a 6×6 diagonal matrix with complex diagonal elements all close to the unit circle.The process noise had a unit variance. $\mathbf{c}_k$ were derived from a Gaussian pseudo-random sequence $\mathbf{g}_k$ such that $\mathbf{c}_k = 1$ if $\mathbf{g}_k \geq 0$ and -1 otherwise. The observation noise variance was then determined by the SNR values. The EWRLS had a forgetting factor .99. The EKF jointly estimated the states and the parameters. The Kalman filter, as a benchmark, knew the true value of $\mathbf{A}$. *subem I* and *II* both took variable values of λ at different SNRs in order to keep the convergence time nearly the same. Figure 1(a) plots the normalized steady-state mean squared prediction error vs. SNR. It shows that both *subem I*, *II* out-perform the EKF in the lower SNR region and out-perform the EWRLS in the high SNR region . This is because the state trajectory is nearly circular at the chosen values of $\mathbf{A}$ hence the EKF has a very slow convergence rate at the low SNR region. For the EWRLS, at the high SNR region its error is dominated by the tracking error caused by the mismatch between the assumed and the true state dynamics which cannot be improved substantially by increasing SNR.

The convergence of parameter estimation using *subem I*, *II* is also studied. The data was generated as above with $SNR = 3dB$. Figure 1(b) plots, for $\lambda = 1, .98, .96$ respectively, the transient parameter MSE for both algorithms, averaged over 60 points. At $\lambda = 1$ neither algorithm converged within the observation duration (the two flat lines in Figure 1(b)). At $\lambda = .98$ and .96 both algorithms converged, with faster rate but higher steady-state MSE at $\lambda = .96$. With a same λ, *subem I* converges faster than *subem II*, but has a higher steady-state parameter MSE. Numerical analysis suggests that the minimum eigenvalue of $\widehat{\mathbf{H}}_0$ is smaller in *subem I* than in *subem II* which in turn leads to larger adapting gain in (20) for *subem I*. Intuitively, the larger gain results in a shorter averaging window which yields faster convergence but a higher steady-state error.



(a) Comparing *subem I, II, KF, EKF* and *EWRLS*



(b) Convergence of parameter estimation in *subem I, II*

Fig. 1. Joint state and parameter estimation.

6. CONCLUSION

New recursive forms of updating the accumulated state moments are derived which can be used in the EM algorithm for joint state and parameter estimation.The new scheme also motivates two suboptimal algorithms which, as numerical results have shown, out-perform the EWRLS and the EKF algorithms. The impact of the exponential weighting on the convergence of these algorithms are also demonstrated numerically.

7. REFERENCES

[1] R. Shumway and D. Stoffer, "An Approach to Time Series Smoothing and Forecasting Using the EM Algorithm ," *J. Time Series Anal.*, vol. 3, no. 4, pp. 253–264, 1982.

[2] E. Weinstein, A. Oppenheim, M. Feder, and J. Buck, "Iterative and Sequential Algorithms for Multisensor Signal Enhancement," *IEEE Trans. Signal Proces*, vol. 42, no. 4, pp. 846–859, April 1994.

[3] R. Elliott and V. Krishnamurthy, "New Finite-Dimensional Filters for Parameter Estimation of Discrete-Time Linear Gaussian Models," *IEEE Trans. Automat Contr*, vol. 44, no. 5, pp. 938–951, May 1999.

[4] R. Elliott, J. Ford, and J. Moore, "On-line almost-sure parameter estimation for partially observed discrete-time linear systems with known noise characteristics," *Int. J. of Adapt Contr and Signal Proces*, vol. 16, pp. 435–453, 2002.

[5] A. Dempster, N. Laird, and D. Rubin, "Maximum Likelihood from Imcomplete Data via the EM algorithm," *J. R. Statistical Society, B*, vol. 39, no. 1, pp. 1–38, 1977.

[6] T. Kailath, A. Sayed, and B. Hassibi, *Linear Estimation*, Prentice Hall, 2000.

[7] A Graham, *Kronecker Products and Matrix Calculus: with Applications*, Ellis Horwood, 1981.