Reduced-Complexity Filtering for Partially Observed Nearly Completely Decomposable Markov Chains

Subhrakanti Dey, *Member, IEEE*

***Abstract*—This paper provides a systematic method of obtaining reduced-complexity approximations to aggregate filters for a class of partially observed nearly completely decomposable Markov chains. It is also shown why an aggregate filter adapted from Courtois' aggregation scheme has the same order of approximation as achieved by the algorithm proposed in this paper. This algorithm can also be used systematically to obtain reduced-complexity approximations to the full-order filter as opposed to algorithms adapted from other aggregation schemes. However, the computational savings in computing the full-order filters are substantial only when the large scale Markov chain has a large number of weakly interacting blocks or "superstates" with small individual dimensions. Some simulations are carried out to compare the performance of our algorithm with algorithms adapted from various other aggregation schemes on the basis of an average approximation error criterion in aggregate (slow) filtering. These studies indicate that the algorithms adapted from other aggregation schemes may become *ad hoc* under certain circumstances. The algorithm proposed in this paper however, always yields reduced-complexity filters with a guaranteed order of approximation by appropriately exploiting the special structures of the system matrices.**

***Index Terms*—Hidden Markov models, queuing analysis, reduced-order systems, singularly perturbed systems, state estimation.**

I. INTRODUCTION

NEARLY completely decomposable Markov chains were first studied by Simon and Ando [1]. These Markov chains are usually large scale, and show strong interactions within groups and weak interactions between the groups. They are known to have a transition probability matrix of the structure $P = I_n + A + \epsilon B$. Applications of such Markov chains to queueing networks and computer systems have been reported in [2] by Courtois who extensively studied these Markov chains. There have been several other studies that contributed to the development of decomposition-aggregation methods for obtaining reduced-order approximations for uncontrolled [3], as well as controlled Markov chains [4], [5]. Essentially, these works were concerned with obtaining approximations for the stationary distribution of the Markov chain. While in [2], an aggregation method is developed that results in an $O(\epsilon)$ approximation of the exact stationary distribution, [6] gives a singular perturbation interpretation to Courtois' aggregation. In [7], stochastic complementation was used to develop an aggregation procedure to obtain the exact stationary distribution. The singular perturbation approach to study aggregation of finite-state Markov chains has been also studied in [8]–[10]. A more recent work is [11] where a clever transformation technique is used to develop an aggregation method that can be used to obtain any arbitrary order of approximation to or even the exact stationary distribution for the purpose of obtaining an aggregation of the policy iteration method in infinite-horizon optimal control of such Markov chains. The problem of the infinite horizon average cost control problem for such Markov chains was also addressed in [12], [13]. It was shown that the optimal solution can be approximated by an optimal solution to the so called *limit Markov control problem* for a sufficiently small ϵ. Algorithms were also provided for achieving these control strategies.

In this paper, it is our objective to study a class of partially observed nearly completely decomposable Markov chains where the observations belong to a discrete set of finite cardinality. Note that partially observed Markov chains are also known as hidden Markov models which have applications in speech recognition [14], adaptive equalization of communication channels [15], biological signal processing, etc. To the best of our knowledge, there has been no systematic way of obtaining reduced-order approximations to conditional probability filters for hidden Markov models in the case when the underlying Markov chain is nearly completely decomposable. This approximation problem with the objective of reducing the number of computations becomes particularly simple when the state to output (observation) transition probability matrix has a block structure in the sense that the state to output transition probabilities are constant over all the states belonging to the same "group" of the Markov chain. It is also easy to extend the analysis to the case where this transition probability matrix is a small perturbation of the block structure. The applicability of such block-structured observation probability matrices not only lies in modeling of management systems (where top levels of management are only interested in macro-behavior rather than micro-behavior) but also in real engineering applications like distributed control environments particularly with communication constraints. For example, in an environment where multiple sensors are sending information, it might not be possible to send fine information due to bit-rate constraints, and hence it might just be practical to send coarser information (e.g., information about the macro-states). This may be of use in hierarchical control systems also, where a controller at one of the top levels of the hierarchy may not want fine information since it may only want to control transitions from

Manuscript received September 10, 1999; revised August 16, 2000. The associate editor coordinating the review of this paper and approving it for publication was Prof. Hideaki Sakai.

The author is with the Department of Electrical and Electronic Engineering, University of Melbourne, Parkville, Victoria 3052, Australia.

Publisher Item Identifier S 1053-587X(00)10158-8.

1053–587X/00$10.00 © 2000 IEEE

one macro-state to another (e.g., the controller may want to know that a failure has occurred and not what particular kind of failure it is). In the rest of the paper, we will only be interested in these classes of partially observed Markov chains.

In this paper, we provide a systematic way of obtaining an $O(\epsilon^2)$ approximation to the aggregate (slow) filter and show that one can adapt Courtois' aggregation method to obtain the same algorithm (which has not been reported in literature either). However, our algorithm also provides a systematic way to obtain an $O(\epsilon^2)$ approximation for the full-order filter with reduced number of computations when there are a large number of blocks in the underlying Markov chain with the individual block sizes being small. It is not clear as to how to adapt Courtois' aggregation method (or for that matter any aggregation method, e.g., the one in [11]) to obtain an $O(\epsilon^2)$ approximation to the full-order filter with reduced number of computations. In a special case where the unperturbed Markov chain (that is the completely decomposable part of the Markov chain with a transition probability matrix I_n+A) has got individually independent and identically distributed subchains, we show that one can obtain (using our algorithm) $O(\epsilon^3)$ approximation to the aggregate filter with reduced number of computations, whereas the other aggregation methods cannot be adapted to achieve that. Our method essentially consists of the amalgamation of two known techniques. First we decompose the full-order conditional probability filter into the slow (aggregate) and a mixed (mixture of fast and slow) mode using the same transformation technique as used in [11]. Then we decouple these two modes using a decoupling transformation as found in [6], [16] or for analyzing stability of adaptive systems in [17]. It is shown that under certain assumptions (which are reasonable for a small enough ϵ) holding our approximation schemes yield reduced order computations. We compare the performances of our algorithm, and algorithms adapted from Courtois' aggregation method and the aggregation method provided in [11] in all these different cases on the basis of an average approximate error criterion for the aggregate filter computed through simulation studies. Under certain circumstances, the *ad hoc* nature of algorithms adapted from these other aggregation methods becomes exposed. Also, we re-iterate that under all these circumstances our algorithm continues to have the edge over algorithms adapted from other aggregation methods in that our algorithm provides a systematic way of obtaining $O(\epsilon^2)$ approximations to the full-order filter with reduced number of computations.

In Section II, we describe the class of hidden nearly completely decomposable Markov chains we are interested in. In Section III, we decompose the conditional probability filter into the aggregate and a mixed mode. In Section IV, we use the decoupling technique and show under what circumstances one can obtain approximate calculations with reduced order computations for the aggregate as well as the full-order filter. Section V shows some simulation results. Finally we present some concluding remarks in Section VI.

II. Signal Model

In this section, we describe the particular classes of partially observed nearly completely decomposable Markov chains that we are interested in. A nearly completely decomposable Markov chain in a complete probability space $(\Omega, \mathcal{F}, \mathcal{P})$ comprising of n states is characterized by a transition probability matrix $P \in \mathbb{R}^{n\times n}$ which has the following structure:

$$P = I_n + A + \epsilon B \tag{1}$$

where I_n is the identity matrix of order $n \times n$

$$A = \begin{bmatrix} A_{11} & 0 & \cdot & \cdot & 0 \\ 0 & A_{22} & 0 & \cdot & \cdot \\ 0 & 0\cdot & \cdot\cdot & \cdot & \cdot \\ \cdot\cdot & \cdot\cdot & \cdot & \cdot & \cdot \\ 0 & \cdot & 0 & \cdot & A_{NN} \end{bmatrix}$$

where $A_{ii} \in \mathbb{R}^{n_i \times n_i}$, $\forall\, i$, $\sum_i n_i = n$, $\epsilon > 0$ is a small perturbation parameter, and $B \in \mathbb{R}^{n\times n}$. It is obvious that there are N blocks in the Markov chain within each of which the dynamics is fast and every so often, the chain leaves one block to visit another. Since ϵ is small, the rate at which these inter-block transitions occur is slow. For all i, $I_{n_i} + A_{ii}$ is row-stochastic, and so is P. Obviously, the row-sums of A and B are zero. We make the following key assumption.

Assumption 2.1: P and $I_{n_i} + A_{ii}$, $\forall\, i$ are irreducible.

Traditionally, one is essentially interested in cases where $n \gg N$. If one is only interested in obtaining reduced complexity approximate computations for the aggregate filter, having $n \gg N$ is sufficient. However, as we will see, the cases where one can save in $O(\epsilon^2)$ approximate computation of the full-order filter involve smaller block sizes with a reasonable number of blocks (where N is still a lot less than n, roughly speaking). We also term these N blocks as "superstates." Note that the probability (or conditional probability) of the Markov chain belonging to a particular superstate is the sum of probabilities (or conditional probabilities) of the chain belonging to its constituent states. We denote the state of the n-state Markov chain at time k as $X_k \in \{1, 2, \cdots, n\}$, and the lth superstate is denoted by S_l, $l = 1, 2, \cdots, N$. Without loss of generality, $S_1 = \{1, 2, \cdots, n_1\}$, $S_2 = \{n_1 + 1, n_1 + 2, \cdots, n_1 + n_2\}$, etc.

The states of the Markov chain are not directly observed, rather observed in imperfect observations. In other words, associated with the state X_k, there is a nondeterministic observation (or measurement) Y_k. For the purpose of this paper, we assume that Y_k belongs to a discrete set of finite cardinality. More specifically, $Y_k \in \{1, 2, \cdots, M\}$ and $\mathcal{P}(Y_k = i | X_k = j) = c_{ij}$, $i = 1, 2, \cdots, M$, $j = 1, 2, \cdots, n$. Such a signal model (irrespective of whether the underlying Markov chain is nearly completely decomposable or not) is also known as a hidden Markov model (HMM). Note also that $\sum_i c_{ij} = 1, \ \forall\, j$, that is, the observation probability matrix $C = (c_{ij})$ is column-stochastic. In what follows, we will be interested in a special structure of C, where the observations reflect the superstates only, i.e., $c_{ij} = \bar{c}_{il}, \ \forall\, j \in S_l, \ \forall\, i$. We will show that this special structure can let one obtain substantial savings in computations if one is interested in obtaining $O(\epsilon^2)$ approximations to the conditional probability estimates for the HMM. For the time being, we make another key but standard assumption.

Assumption 2.2: $\min_{i,j} c_{ij} \geq \tilde{c} > 0$.

Remark 1: The above assumption obviously implies that there exists a $\bar{c}$ such that $\max_{i,j} c_{ij} \leq \bar{c} < 1$.

We will also analyze a case (of course, a very restrictive one) where the Markov chains represented by the block-stochastic matrices $I_{n_i} + A_{ii}$ are independently and identically distributed (*i.i.d*) (note that in this case, C does not need to be "block" structured). This leads to an $O(\epsilon^3)$ approximation to the aggregate filter with reduced number of computations and further reduction in computations to $O(\epsilon^2)$ approximation to the full-order filter. However, these specialties need not be taken into consideration for the analysis to follow in the next section which holds for any P of the structure given by (1) and any column-stochastic C that satisfies Assumption 2.2. We will remind the reader of these special structures when we analyze approximate reduced order computations.

III. Aggregate Filtering of Hidden Markov Models

It is well known that the conditional filtered state estimate for a hidden Markov model is defined in the following way:

$$\alpha_k(i) = \mathcal{P}(X_k = i | \mathcal{Y}_k) \tag{2}$$

where $\mathcal{Y}_k$ is the complete filtration generated by the σ algebra $\sigma(Y_0, Y_1, \cdots, Y_k)$. Defining the row vector $\alpha_k \triangleq (\alpha_k(1)\ \alpha_k(2) \cdots \alpha_k(n))$, one can obtain the following recursion [14]

$$\begin{aligned} \alpha_{k+1} &= \frac{1}{Z_{k+1}} \alpha_k P C(Y_{k+1}) \\ \alpha_0 &= \pi_0 C(Y_0) \end{aligned} \tag{3}$$

where $C(Y_{k+1}) = \mathrm{diag}\{c_{i1}\ \ c_{i2} \cdots c_{\mathrm{in}}\}$ if $Y_{k+1} = i$ and $Z_{k+1} = \alpha_k P C(Y_{k+1}) 1_n$ is the normalization factor (with 1_n being the n-length column vector of all 1-s) and π_0 is the row vector representing the initial distribution of X_0.

As we are interested in the aggregate filtering, we note that

$$\zeta_k(j) = \mathcal{P}(X_k \in S_j | Y_k) = \sum_{l \in S_j} \mathcal{P}(X_k = l | Y_k). \tag{4}$$

Obviously the row vector $\zeta_k \triangleq (\zeta_k(1)\, \zeta_k(2) \cdots \zeta_k(N))$ denotes the aggregate filtered state estimates and can be represented by

$$\zeta_k = \alpha_k W_1 \tag{5}$$

where $W_1 \in \mathbb{R}^{n \times N}$ is given by

$$W_1 = \begin{bmatrix} 1_{n_1} & 0 & 0 & 0 \\ 0 & 1_{n_2} & \cdot & \cdot \\ \cdot & 0 & \cdot & 0 \\ 0 & \cdot & \cdot & 1_{n_N} \end{bmatrix}.$$

Now, following the same techniques as in [11], we choose another matrix $W_2 \in \mathbb{R}^{n \times (n-N)}$ such that the transformation $\Gamma = [W_1\ W_2]$ is nonsingular. Let $\eta_k = \alpha_k W_2$. Let also $\Gamma^{-1} = \left[\begin{smallmatrix} V_1 \\ V_2 \end{smallmatrix}\right]$ where obviously $V_1 \in \mathbb{R}^{N \times n}$ and $V_2 \in \mathbb{R}^{(n-N) \times n}$. The particular choice of W_2 will be discussed later. One can now rewrite (3) as

$$\begin{aligned} [\zeta_{k+1}\ \eta_{k+1}] &= \frac{1}{Z_{k+1}} [\zeta_k\ \eta_k] \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} P C(Y_{k+1}) [W_1\ W_2] \\ &= \frac{1}{Z_{k+1}} [\zeta_k\ \eta_k] \begin{bmatrix} \tilde{A}_{11}^k & \tilde{A}_{12}^k \\ \tilde{A}_{21}^k & \tilde{A}_{22}^k \end{bmatrix} \end{aligned} \tag{6}$$

where $\tilde{A}_{11}^k = \tilde{A}_1^k + \epsilon \tilde{B}_1^k$, $\tilde{A}_{12}^k = \tilde{A}_2^k + \epsilon \tilde{B}_2^k$, $\tilde{A}_{21}^k = \tilde{C}_1^k + \epsilon \tilde{D}_1^k$, $\tilde{A}_{22}^k = \tilde{C}_2^k + \epsilon \tilde{D}_2^k$ and they are given by the following equations:

$$\begin{aligned} \tilde{A}_1^k &= V_1 (I_n + A) C(Y_{k+1}) W_1, & \tilde{B}_1^k &= V_1 B C(Y_{k+1}) W_1 \\ \tilde{A}_2^k &= V_1 (I_n + A) C(Y_{k+1}) W_2, & \tilde{B}_2^k &= V_1 B C(Y_{k+1}) W_2 \\ \tilde{C}_1^k &= V_2 (I_n + A) C(Y_{k+1}) W_1, & \tilde{D}_1^k &= V_2 B C(Y_{k+1}) W_1 \\ \tilde{C}_2^k &= V_2 (I_n + A) C(Y_{k+1}) W_2, & \tilde{D}_2^k &= V_2 B C(Y_{k+1}) W_2. \end{aligned} \tag{7}$$

It is not hard to see that (6) can be carried out in two steps ($\forall\, k$).

Step 1) Calculate the unnormalized quantities ζ_{k+1}^u, η_{k+1}^u according to the following recursion:

$$[\zeta_{k+1}^u\ \eta_{k+1}^u] = [\zeta_k\ \eta_k] \begin{bmatrix} \tilde{A}_{11}^k & \tilde{A}_{12}^k \\ \tilde{A}_{21}^k & \tilde{A}_{22}^k \end{bmatrix} \tag{8}$$

where $\zeta_0 = \alpha_0 W_1$, $\eta_0 = \alpha_0 W_2$.

Step 2) Normalize ζ_{k+1}^u, η_{k+1}^u by the normalization factor $Z_{k+1} = \sum_{j=1}^{N} \zeta_{k+1}^u(j)$.

Note also that

$$\alpha_k = \zeta_k V_1 + \eta_k V_2. \tag{9}$$

Now, if we were only interested in exact aggregate filtering, (i.e., only in computations of ζ_k) we still have to calculate ζ_k, η_k, $\forall\, k$ and the amount of computations involved in this can be evaluated to be the following [assuming the matrices in (7) have been pre-computed]:

- number of multiplications for each k
$$N^2 + 2N(n-N) + (n-N)^2 + n = n^2 + n;$$
- number of additions for each k
$$\begin{aligned} &(n-N-1)N + (N-1)N + (N-1)(n-N) \\ &\quad + (n-N-1)(n-N) + n + N - 1 \\ &= n^2 - n + N - 1. \end{aligned}$$

However, we have not yet exploited the fact that ϵ is a small positive number and presumably, we should be able to utilize that fact in obtaining approximate aggregate filtered estimates with reduced order computations. In the next section, we discuss how we can do that.

Before proceeding on to the next section, let us choose W_2 in such a way that the computations of the matrices in (7) become very simple. Again, we adopt the ideas in [11] and the ith diagonal block in W_2, namely $W_2^{(i)}$ ($\in \mathbb{R}^{n_i \times n_i - 1}$), is chosen to be

$$W_2^{(i)} = \begin{bmatrix} 0 \cdots 0 \\ I_{n_i - 1} \end{bmatrix}. \tag{10}$$

For this choice, the ith diagonal blocks of V_1, V_2 turn out to be

$$V_1^{(i)} = [1\ \ 0\ \ 0 \cdots 0]$$

$$V_2^{(i)} = \begin{bmatrix} -1 & \cdot & \\ -1 & \cdot & \\ \cdot & \cdot & I_{n_i - 1} \\ \cdot & \cdot & \\ -1 & \cdot & \end{bmatrix}. \tag{11}$$

Here, $V_1^{(i)} \in \mathbb{R}^{1 \times n_i}$, $V_2^{(i)} \in \mathbb{R}^{(n_i - 1) \times n_i}$.

For the above choices of V_1, V_2, W_1, W_2, one can easily show that the matrices $\tilde{A}_1^k$, $\tilde{A}_2^k$, $\tilde{C}_1^k$ and $\tilde{C}_2^k$ are block diagonal matrices for all k, more specifically, $\tilde{A}_1^k$ is diagonal, $\tilde{A}_2^k$ is block diagonal with the ith block being a row vector of size $n_i - 1$, $\tilde{C}_1^k$ is block diagonal with the ith block being a column vector of size $n_i - 1$ and $\tilde{C}_2^k$ is block diagonal with the ith block being a square matrix of size $(n_i - 1) \times (n_i - 1)$. We will later see that for the special choice of a "block-structured" observation probability matrix C (as discussed in Section II), one can obtain $\tilde{C}_1^k = 0$ which will be useful for approximate calculations with reduced number of computations.

Note also that, since the matrices $\tilde{A}_1^k$, $\tilde{A}_2^k$, $\tilde{C}_1^k$ and $\tilde{C}_2^k$ depend only on Y_{k+1} which is finitely-valued, one can essentially pre-compute the matrices $\tilde{A}_1^k$, $\tilde{A}_2^k$, $\tilde{C}_1^k$ and $\tilde{C}_2^k$ for each possible value of Y_{k+1} and store them in a lookup table. During the filtering operations, as and when we get a specific observation, we can obtain the corresponding matrices by just looking up the table.

IV. Approximate ($O(\epsilon^2)$) Reduced Order Computations for the Aggregate Filter

In this section, we will be primarily concerned about obtaining approximations to the aggregate filtered estimate ζ_k with reduced order computations. To do this, we need to introduce a decoupling transformation following ideas in [6], [16] such that the transformed variables $[\overline{\zeta}_k \ \overline{\eta}_k]$ are given by

$$[\overline{\zeta}_k \ \overline{\eta}_k] = [\zeta_k \ \eta_k] \begin{bmatrix} I_N & L_k \\ 0 & I_{n-N} \end{bmatrix}. \tag{12}$$

This also implies that

$$[\zeta_k \ \eta_k] = [\overline{\zeta}_k \ \overline{\eta}_k] \begin{bmatrix} I_N & -L_k \\ 0 & I_{n-N} \end{bmatrix}. \tag{13}$$

Note that one can relate the unnormalized versions of $\overline{\zeta}_k$, $\overline{\eta}_k$, denoted by $\overline{\zeta}_k^u$, $\overline{\eta}_k^u$, respectively, by the same decoupling transformation, provided the normalization factor is the same

$$[\zeta_k^u \ \eta_k^u] = [\overline{\zeta}_k^u \ \overline{\eta}_k^u] \begin{bmatrix} I_N & -L_k \\ 0 & I_{n-N} \end{bmatrix}. \tag{14}$$

Here $\{L_k \in \mathbb{R}^{N \times (n-N)}\}$ is assumed to be a sequence of uniformly bounded time-varying matrices to be solved for. Using this together with (6), one can obtain the following recursion in the transformed variables

$$\begin{aligned} [\overline{\zeta}_{k+1} \ \overline{\eta}_{k+1}] = \frac{1}{Z_{k+1}} & [\overline{\zeta}_k \ \overline{\eta}_k] \\ & \cdot \begin{bmatrix} \tilde{A}_{11}^k - L_k \tilde{A}_{21}^k & 0 \\ \tilde{A}_{21}^k & \tilde{A}_{21}^k L_{k+1} + \tilde{A}_{22}^k \end{bmatrix} \end{aligned} \tag{15}$$

where L_k satisfies the following recursive equation:

$$\left(\tilde{A}_{11}^k - L_k \tilde{A}_{21}^k\right) L_{k+1} = L_k \tilde{A}_{22}^k - \tilde{A}_{12}^k, \quad L_0 = 0. \tag{16}$$

Note that one can recursively solve for L_k from the above equation provided $\tilde{A}_{11}^k - L_k \tilde{A}_{21}^k$ is invertible for every k. This involves multiplications of matrices which are not necessarily sparse at each k and requires a large number of computations. A well known technique [6] is to exploit the fact that $\epsilon > 0$ is a small positive number and truncate a power series expansion of L_k in ϵ at some finite power. One then solves iteratively for the coefficients of the truncated power series. In order to reduce computational complexity, one needs to truncate the series in such a way that the coefficients to be solved for can be computed recursively with reduced number of computations. This of course will restrict the order of approximation. Note however that this approximation is only valid if L_k is uniformly bounded. Notice also from [6], [16] that this method will work efficiently if $\tilde{A}_{21}^k$, $\tilde{A}_{22}^k$ are such that $\tilde{C}_1^k = 0$, $\tilde{C}_2^k = 0$ or in other words, $\tilde{A}_{21}^k = \epsilon \tilde{D}_1^k$, $\tilde{A}_{22}^k = \epsilon \tilde{D}_2^k$. Note that in general, one cannot find cases where such conditions will hold. However, with some special choices of the observation probability matrix C or the transition probability matrices $I_{n_i} + A_{ii}$, one can obtain some simple situations. In the following, we discuss such cases.

A. Observations Only Reflecting the Aggregate System

In this subsection, we consider a C that has the special structure as discussed briefly in Section II such that $c_{ij} = \overline{c}_{il}$, $\forall\, j \in S_l$, $\forall\, i$. In this case, one can easily work out that $\tilde{C}_1^k = 0$ for the particular choices of the matrices V_2, W_1 as discussed before. Note also that for this special structure of C, the ith diagonal element in $\tilde{A}_1^k$ is $\overline{c}_{ji}$ if $Y_{k+1} = j$. Similarly all the elements of the ith diagonal block in $\tilde{A}_2^k$, $\tilde{C}_2^k$ are scaled by $\overline{c}_{ji}$ if $Y_{k+1} = j$. This is going to be vital in analyzing the stability of (16) as outlined below. For example, for this simple scaling property, $(\tilde{A}_1^k)^{-1} \tilde{A}_2^k$ can be written as a time-invariant matrix $G_0 \in \mathbb{R}^{N \times (n-N)}$.

One can now rewrite (16) as follows:

$$L_{k+1} = \left(\tilde{A}_1^k\right)^{-1} L_k \tilde{C}_2^k - G_0 + \epsilon G_{k+1}, \quad L_0 = 0 \tag{17}$$

where

$$G_{k+1} = \left(\tilde{A}_1^k\right)^{-1} \left(L_k \tilde{D}_2^k - \tilde{B}_1^k L_{k+1} + L_k \tilde{D}_1^k L_{k+1} - B_2^k\right).$$

Let us now introduce the following notations. We denote by $|\,.\,|$ the Euclidean vector norm, by $||\,.\,||_2$ the Frobenius norm for a matrix (note that this is a matrix norm for a square matrix) and by $||\,.\,||_\infty$ the maximum absolute row sum matrix norm. Note also that $|v| = ||M||_2$ when $v' = \mathrm{vec}(M')$.

The solution to (17) can be written as $L_k = L_k(0) + \epsilon \tilde{L}_k(\epsilon)$ where $L_k(0)$ is the solution to (17) when $\epsilon = 0$. In other words, one can solve for $L_k(0)$ from the following equation:

$$L_{k+1}(0) = \left(\tilde{A}_1^k\right)^{-1} L_k(0) \tilde{C}_2^k - G_0, \quad L_0(0) = 0. \tag{18}$$

Due to the scaling property (mentioned in the beginning of this section) of the individual blocks in the block diagonal matrices $\tilde{A}_1^k, \tilde{C}_2^k, \tilde{A}_2^k$ (remembering that $\tilde{A}_1^k$ is actually diagonal), it is easy to show that (18) has a steady-state solution $L(0)$ which only depends on the matrix A. In fact, $L(0) = V_1 A W_2 (V_2 A W_2)^{-1}$ (noting that $V_2 A W_2$ is invertible [11]). Since we are mainly interested in the solution to (17) as $k \to \infty$, we can write $L_k = L(0) + \epsilon \tilde{L}_k(\epsilon)$. It is usual to expand $\tilde{L}_k(\epsilon)$ as a power series in ϵ and rewrite $L_k = L(0) + \epsilon L_k(1) + \epsilon^2 L_k(2) + \cdots$ and obtain approximation to L_k by truncating this infinite series at some finite power of ϵ. However, one needs to establish a uniform bound on L_k before one can obtain a valid approximation. In what follows, we will be looking for sufficient conditions such that L_k belongs to a compact set $\mathcal{D} \triangleq \{L\colon ||L||_2 < (1 + \epsilon \overline{L}) ||L_0||_2\}$ that is

$$||\tilde{L}_k(\epsilon)||_2 < \epsilon \overline{L} ||L_0||_2$$

and

$$\|L_k\|_2 < \left(1+\epsilon\overline{L}\right)\|L_0\|_2, \quad \forall\, k.$$

Since $Y_{k+1}, \quad \forall\, k$ can only take finitely many values, note that one can easily obtain the following over bounds $\overline{a}_1, \overline{b}_1, \overline{b}_2, \overline{d}_1, \overline{d}_2$ such that $\|(\tilde{A}_1^k)^{-1}\|_2 \le \overline{a}_1$, $\|\tilde{B}_1^k\|_2 \le \overline{b}_1$, $\|\tilde{B}_2^k\|_2 \le \overline{b}_2$, $\|\tilde{D}_1^k\|_2 \le \overline{d}_1$, $\|\tilde{D}_2^k\|_2 \le \overline{d}_2$, $\forall\, k$.

Writing $l'_k = \mathrm{vec}(L'_k)$, $g'_0 = \mathrm{vec}(G'_0)$, and $g'_k(L_{k-1}, L_k) = \mathrm{vec}(G'_k)$, one can rewrite (17) as

$$\begin{aligned} l_{k+1} =& l_k\left(\left(\tilde{A}_1^k\right)^{-1} \otimes \tilde{C}_2^k\right) - g_0 \\ &+ \epsilon g_{k+1}(L_k, L_{k+1}), \quad l_0 = 0. \end{aligned} \tag{19}$$

Obviously, repeating this operation iteratively, one can express l_{k+1} as

$$\begin{aligned} l_{k+1} =& l(0) + \epsilon\left[\sum_{p=0}^{\infty} g_{k+1-p}(L_{k-p}, L_{k+1-p})\right. \\ &\left.\cdot \prod_{q=p-1}^{0} M_{k+1-q}\right], \qquad l_0 = 0 \end{aligned} \tag{20}$$

where $l(0) = \mathrm{vec}(L(0)')'$, $M_{k+1} = (\tilde{A}_1^k)^{-1} \otimes \tilde{C}_2^k$ and $\prod_{q=-1}^{0} M_{k+1-q} = I_{N\times(n-N)}$. It is easy to see that if $L_k \in \mathcal{D}$, then for every k, $|g_{k+1}(L_k, L_{k+1})| < \rho(\epsilon, \overline{L})$ where

$$\begin{aligned} \rho\left(\epsilon, \overline{L}\right) =& \overline{a}_1\left[\left(1+\epsilon\overline{L}\right)\|L(0)\|_2\right. \\ &\left.\cdot\left(\overline{d}_2 + \overline{b}_1 + \left(1+\epsilon\overline{L}\right)\|L(0)\|_2\overline{d}_1\right) + \overline{b}_2\right]. \end{aligned}$$

Define the following matrices as follows.

Definition 4.1: For $i = 1, 2, \cdots, N$

$$\tilde{M}_i = V_2^{(i)}(I_{n_i} + A_{ii})W_2^{(i)}. \tag{21}$$

With this definition, it is easy to show that if $Y_{k+1} = j$, then

$$\tilde{C}_2^k = \begin{bmatrix} \overline{c}_{j1}\tilde{M}_1 & 0 & \cdot & \cdot & 0 \\ 0 & \overline{c}_{j2}\tilde{M}_2 & 0 & \cdot & \cdot \\ 0 & 0\,\cdot & \cdot\,\cdot & \cdot & \cdot \\ \cdot\,\cdot & \cdot\,\cdot & \cdot & \cdot & \cdot \\ 0 & \cdot & 0 & \cdot & \overline{c}_{jN}\tilde{M}_N \end{bmatrix} \tag{22}$$

and $M_{k+1} \in \mathbb{R}^{N(n-N)\times N(n-N)}$ is also a block-diagonal matrix consisting of N subblock-diagonal matrices and the lth diagonal block of the ith subblock-diagonal matrix [of order $(n_l - 1) \times (n_l - 1)$] is given by

$$M_{k+1}^{(i,\,l)} = \frac{\overline{c}_{jl}}{\overline{c}_{ji}}\tilde{M}_l, \qquad i = 1, 2, \cdots, N, \quad l = 1, 2, \cdots, N. \tag{23}$$

We now make the following assumptions in order to find sufficient conditions for obtaining a bounded solution to (19).

Assumption 4.1:

$$\left\|\tilde{M}_l\right\|_\infty < 1, \qquad l = 1, 2, \cdots, N. \tag{24}$$

Assumption 4.2: If $E[\ln(\overline{c}_{Y_k l}/\overline{c}_{Y_k i})]$ exists and is evaluated to be γ_{li}, then

$$\gamma_{li} + \ln\left\|\tilde{M}_l\right\|_\infty < -\lambda_{li} < 0, \qquad i \ne l. \tag{25}$$

Remark 2: Suppose Assumptions 2.1 and 2.2 hold. In this case

$$E\left[\ln\left(\frac{\overline{c}_{Y_k l}}{\overline{c}_{Y_k i}}\right)\right] = \sum_j\sum_m \overline{c}_{jm}\ln\left(\frac{\overline{c}_{jl}}{\overline{c}_{ji}}\right)\left(\sum_{r\in S_m}\pi(r)\right)$$

where $\pi \in \mathbb{R}^{1\times n}$ is the solution $\pi = \pi P$, $\pi 1_n = 1$ such that π is the steady state distribution of the Markov chain. $\pi(r)$ is the rth element of the vector. For how to obtain approximations or even the exact value of π without numerical ill-conditioning for nearly completely decomposable Markov chains, see [11].

Note that Assumption 4.2 implies that for every l, i, there exists a $t_{li} > 0$ such that

$$\left\|\prod_{q=p-1}^{0} M_{k+1-q}^{(i,\,l)}\right\|_\infty < \exp(-p\lambda_{li}), \quad \text{for some } p > t_{li}. \tag{26}$$

A proof of this statement is given in the Appendix as a part of the proof of the following Lemma.

Lemma 4.1: Suppose Assumptions 2.1, 2.2, 4.1, 4.2 hold. Then there exists a bounded solution to (19) such that $L_k \in D$ for every k if the following inequalities are satisfied:

$$\rho(\epsilon, \overline{L})\left[\overline{S} + \sqrt{N(n-N)}\,\frac{\exp(-(t^*+1)\delta^*)}{1-\exp(-\delta^*)}\right] < \|L(0)\|_2\overline{L} \tag{27}$$

$$\epsilon\overline{a}_1\left[\overline{d}_2 + \overline{b}_1 + 2\left(1+\epsilon\overline{L}\right)\overline{d}_1\|L(0)\|_2\right] < 1 \tag{28}$$

where in (27), $t^* = \max_{i,l,i\ne l} t_{li}$, $\delta^* = \min(\beta^*, \lambda^*)$, $0 < \beta^* < \min_l \ln(1/\|\tilde{M}_l\|_\infty)$, $\lambda^* = \min_{i,\,l,\,i\ne l}\lambda_{li}$ and $\overline{S}$ is a finite number depending only on the system parameters A, C.

Remark 3: Notice that Assumptions 2.1, 2.2, 4.1, and 4.2 can be readily verified for a given HMM. Finding ϵ, $\overline{L}$ that satisfy (28) is straightforward as well. Verifying that these values of ϵ, $\overline{L}$ satisfy (27) however, is not straightforward, since one needs to evaluate t^*, δ^* and $\overline{S}$. We maintain though that this can be done in principle. In the section on simulation studies, we illustrate with an example (instead of finding the range of ϵ, $\overline{L}$) that there are choices of ϵ, $\overline{L}$ such that there is a bounded solution to L_k that does not tend to explode in finite time, thus guaranteeing that our chosen set of assumptions does not lead to a vacuous problem. Also, one can possibly obtain a set of less restrictive inequalities (particularly for large values of n) involving ϵ, $\overline{L}$ if one treats each row of L_k separately instead of treating the entire matrix using the vec operation. However, we refrain from such exercise in the interest of not introducing unnecessary complications.

The boundedness of L_k allows us to write L_k as a power series in ϵ as discussed before

$$L_k = L(0) + \epsilon L_k(1) + \cdots. \tag{29}$$

Substituting this in (16) and equating the coefficients of ϵ, one can obtain the recursion for $L_k(1)$ as follows:

$$L_{k+1}(1) = \left(\tilde{A}_1^k\right)^{-1} L_k(1)\tilde{C}_2^k + Q_k \tag{30}$$

where

$$Q_k = \left(\tilde{A}_1^k\right)^{-1}\left(L(0)\tilde{D}_2^k + L(0)\tilde{D}_1^k L(0) - \tilde{B}_1^k L(0) - \tilde{B}_2^k\right).$$

Remark 4: Suppose Assumptions 2.1, 2.2, 4.1, and 4.2 hold. Then the boundedness of $L_k(1)$ as obtained from (30) follows from the fact that the evolution $l_{k+1} = l_k M_{k+1}$ is exponentially stable (as discussed in the Appendix for the proof of Lemma 4.1) and Q_k is bounded.

It will be clear presently that in order to save in number of computations for obtaining $O(\epsilon^2)$ approximation, we only need to consider solving for $L_k(1)$. Higher order approximations to L_k do not result in computational reductions. Below, we discuss how $L(0)$ and $L_k(1)$ let us obtain reduction in computation.

But first, notice that using (14) and (31), the decoupled fast mode can now be recursively expressed as

$$\overline{\eta}_{k+1} = \frac{1}{Z_{k+1}} \overline{\eta}_k \left(\tilde{A}_{21}^k L_{k+1} + \tilde{A}_{22}^k\right) \tag{31}$$

where $Z_{k+1} = \sum \zeta_{k+1}^u(i)$. We make the following additional assumption.

Assumption 4.3: The evolution $z_{k+1} = z_k(\tilde{A}_{21}^k L_{k+1} + \tilde{A}_{22}^k)$ where $z_k \in \mathbb{R}^{n-N}$ is exponentially stable.

Remark 5: Note that $\tilde{A}_{21}^k L_{k+1} + \tilde{A}_{22}^k = \tilde{C}_2^k + \epsilon(\tilde{D}_2^k + \tilde{D}_1^k L_{k+1})$. It is clear from (22), Assumption 2.2, and Assumption 4.1 that the evolution $z_{k+1} = z_k \tilde{C}_2^k$ is exponentially stable. If $L_k \in \mathcal{D}$ and ϵ is small enough, then this property of exponential stability is likely to be preserved for the evolution $z_{k+1} = z_k(\tilde{A}_{21}^k L_{k+1} + \tilde{A}_{22}^k)$. In other words, one can obtain a sufficient condition further restricting the range of ϵ [in addition to (27), (28)] with the following inequality:

$$\begin{aligned} &\overline{c}\left(\max_i \sqrt{n_i - 1} \left\|\tilde{M}_i\right\|_\infty\right) \\ &\quad + \epsilon\left[\overline{d}_2 + \left(1 + \epsilon\overline{L}\right) \|L(0)\|_2 \overline{d}_1\right] < 1. \end{aligned} \tag{32}$$

This inequality in addition to (27) and (28) may appear to be restrictive but we reiterate that they only provide a set of sufficient (and not necessary) conditions. In the section on simulation studies, we illustrate with an example how the above assumption is satisfied for the given choice of hidden Markov model.

Note that Assumption 4.3 guarantees that $\overline{\eta}_k \to 0$ asymptotically. The rate of this decay is determined by the fast eigenvalues of $\tilde{A}_{21}^k L_{k+1} + \tilde{A}_{22}^k$ and how close they are to the origin. Hence there exists a large enough but finite k_0 such that for $k \geq k_0$, $|\overline{\eta}_k|$ is of $O(\epsilon^2)$. Set $\tilde{\zeta}_k = \zeta_k$, $\tilde{\eta}_k = \eta_k$ for $k < k_0$. For $k \geq k_0$, one can use (15), (13) to obtain the following reduced-complexity $O(\epsilon^2)$ approximations to ζ_k, η_k

$$\begin{aligned} \tilde{\zeta}_{k+1} &= \frac{1}{\tilde{Z}_{k+1}} \tilde{\zeta}_k \left(\tilde{A}_{11}^k - L(0)\tilde{A}_{21}^k\right) \\ \tilde{\eta}_{k+1} &= -\tilde{\zeta}_{k+1}(L(0) + \epsilon L_{k+1}(1)) \end{aligned} \tag{33}$$

where $\tilde{Z}_{k+1} = \tilde{\zeta}_{k+1} 1_N$. Given that $\tilde{A}_{21}^k = \epsilon\tilde{D}_1^k$, notice that L_k has been replaced by $L(0)$ in the recursion for ζ_k, and L_{k+1} has been replaced by $L(0) + \epsilon L_{k+1}(1)$ in the recursion for η_{k+1} in (33). Implicitly, we have assumed that ϵ is small enough such that the normalization procedure [division by $\tilde{Z}_{k+1}$ in (33)] does not affect the order of approximation of the unnormalized quantity. It should be clear from above that $Z_{k_0} = \tilde{Z}_{k_0} + O(\epsilon^2)$. This automatically implies that $|\zeta_{k_0} - \tilde{\zeta}_{k_0}| = O(\epsilon^2)$ if $(1/Z_{k_0}) = (1/\tilde{Z}_{k_0}) + O(\epsilon^2)$. By induction, one can then show that $|\zeta_k - \tilde{\zeta}_k| = O(\epsilon^2)$ provided $(1/Z_k) = (1/\tilde{Z}_k) + O(\epsilon^2)$ for $k > k_0$. More formally, we make the following additional assumption.

Assumption 4.4: ϵ is small enough such that $(1/Z_k) = (1/\tilde{Z}_k) + O(\epsilon^2)$ for $k \geq k_0$.

Remark 6: It is worth pointing out that the above assumption can be made more formal and rigorous by finding inequalities involving ϵ, $\overline{L}$ and $\overline{J} > 0$ such that $|\zeta_k - \tilde{\zeta}_k| < \epsilon^2 \overline{J}$ for $k \geq k_0$. However, we do not pursue this matter any further for the sake of simplicity.

We now summarize the main results of this section in the following theorem. The proof follows as a direct consequence of the previous discussions and is omitted.

Theorem 1: Suppose Assumptions 2.1, 2.2, 4.1, 4.2, 4.3, and 4.4 hold. Also, suppose the inequalities (27), (28) hold. Then there exists a large enough but finite k_0 such that for $k \geq k_0$, an $O(\epsilon^2)$ approximation for ζ_{k+1} denoted by $\tilde{\zeta}_{k+1}$ (and the unnormalized version by $\tilde{\zeta}_{k+1}^u$) can be obtained recursively by the following two steps:

$$\begin{aligned} \tilde{\zeta}_{k+1}^u &= \tilde{\zeta}_k \left[\tilde{A}_1^k + \epsilon\left(\tilde{B}_1^k - L(0)\tilde{D}_1^k\right)\right], \quad \tilde{\zeta}_{k_0-1} = \zeta_{k_0-1} \\ \tilde{\zeta}_{k+1} &= \frac{1}{\left[\tilde{\zeta}_{k+1}^u 1_N\right]} \tilde{\zeta}_{k+1}^u. \end{aligned} \tag{34}$$

Similarly, an $O(\epsilon^2)$ approximation for η_k (for $k \geq k_0$) is given by

$$\tilde{\eta}_k = -\tilde{\zeta}_k \left(L(0) + \epsilon L_k(1)\right), \quad \tilde{\eta}_{k_0-1} = \eta_{k_0-1}. \tag{35}$$

Since $[\tilde{A}_1^k + \epsilon(\tilde{B}_1^k - L_0\tilde{D}_1^k)]$ can be pre-computed and stored for all possible values of the observation Y_k, the number of multiplications involved to carry out (34) (for each k) are N^2 (for the vector–matrix multiplication) and N for the normalization whereas the number of additions are $N^2 - 1$. This is considerably less than the number of multiplications and additions to carry out the exact computation ($n^2 + n$ and $n^2 - n + N - 1$, respectively).

Note also that (34) is also the same recursion as one can obtain by using Courtois' aggregate matrix and the aggregate observation probability matrix of size $N \times N$ which is given by

$$\begin{aligned} \tilde{\zeta}_{k+1}^u &= \tilde{\zeta}_k P_{court} C_{ag}(Y_{k+1}) \\ \tilde{\zeta}_{k+1} &= \frac{1}{\left[\tilde{\zeta}_{k+1}^u 1_N\right]} \tilde{\zeta}_{k+1}^u \end{aligned} \tag{36}$$

where

$$P_{court} = I_N + \epsilon[V_1 - V_1 A W_2 (V_2 A W_2)^{-1} V_2] B W_1$$

and

$$C_{ag}(Y_{k+1}) = \text{diag}\left\{\overline{c}_{i1}\ \overline{c}_{i2} \cdots \overline{c}_{iN}\right\} \qquad \text{if } Y_{k+1} = i.$$

In other words, one can use Courtois' aggregation method to form an aggregate matrix to obtain an $O(\epsilon^2)$ approximation to the slow (aggregate) filter. However, if one is interested in obtaining an approximation to the full order filter, it is not clear as to how one can adapt Courtois' or any other aggregation method (including Khalil's method). This is where our algorithm has the advantage, in the sense that it gives a systematic way of obtaining an approximate full order filter α_k, with less number of computations [for $O(\epsilon^2)$ approximation]. This will be clear as we proceed.

Now, going back to (30), notice that Q_k can be pre-computed for all possible values of the observation Y_{k+1} and stored. It follows then that the number of multiplications and additions needed to solve for $L_k(1)$ for each k are given by $N\sum_{l=1}^{N}(n_l-1)^2+N(n-N)$ and $N\sum_{l=1}^{N}(n_l-2)(n_l-1)+N(n-N)$, respectively. Including the number of multiplications and additions to carry out this operation for each k and the recursion given by (35), the total number of multiplications [to obtain an $O(\epsilon^2)$ approximation for the full-order filter α_k] stand at $N\sum_{l=1}^{N}(n_l-1)^2+(n-N)(2N+1)$ and the total number of additions stand at $N\sum_{l=1}^{N}(n_l-2)(n_l-1)+(n-N)2N$, respectively. Note that to save in computation in order to obtain $O(\epsilon^2)$ approximation for η_k and hence the full order filter α_k, the individual block sizes need to be small with a reasonable number of blocks. We illustrate this with the following example of a Markov chain with $n=200,\ N=40$ such that $n_l=5,\ \forall\, l$. The total number of multiplications and additions for the exact computations (for each k) are 40 200 and 40 039. The number of multiplications only to compute the approximate aggregate filter is 1640, the number of additions being 1599. To compute the approximate full order filter, the total number of multiplications is 40 200 and the total number of additions is 33 799. So, the number of additions is less by 15.58%.

B. Independent and Identically Distributed Markov Subchains

If the individual Markov chains denoted by $I_{n_i}+A_{ii}$ are *i.i.d.*, then it can be easily shown that $\tilde{C}_1^k=0,\ \tilde{C}_2^k=0,\ \forall\, k$. In this case, the computation of $L_k(1),\ \forall\, k$ is simplified even further since it is easy to see from (30) that $L_k(1)=Q_{k-1}$ can now be pre-computed for all possible values of the observation Y_k and stored. This leads to further savings in computations for obtaining approximations to the full-order filter. Note also that in this case $L(0)$ is given by $-V_1AW_2$, and one can obtain an $O(\epsilon^3)$ approximation to the slow (aggregate) filter (for some $k>0$ large enough) computed by the following recursion:

$$\begin{aligned}\tilde{\zeta}_{k+1}^u &=\tilde{\zeta}_k\left[\tilde{A}_1^k+\epsilon\left(\tilde{B}_1^k-(L(0)+\epsilon L_k(1))\,\tilde{D}_1^k\right)\right]\\ \tilde{\zeta}_{k+1} &=\frac{1}{\left[\tilde{\zeta}_{k+1}^u 1_N\right]}\,\tilde{\zeta}_{k+1}^u.\end{aligned} \tag{37}$$

The number of computations needed to carry out this recursion remains the same as that of computing (34) since $\tilde{A}_1^k+\epsilon(\tilde{B}_1^k-(L(0)+\epsilon L_k(1))\tilde{D}_1^k)$ can be pre-computed and stored for all possible values of the pair $(Y_k,\,Y_{k+1})$.

Remark 7: Analyzing the stability of (16) is understandably trivial in this case and in any case follows as a special case of the previous subsection.

It is clear now that we can obtain an $O(\epsilon^3)$ approximation to the slow (aggregate) filter in the case when the individual matrices $I_{n_i}+A_{ii}$ represent *i.i.d* Markov chains. The algorithm adapted from Courtois' aggregation method [as given by (36)] cannot achieve that. In fact, we compare our algorithm with methods adapted from Courtois' and Khalil's aggregation algorithms in the next section on the basis of an average approximation error for the slow (aggregate) filter and illustrate when our algorithm can achieve better results.

Remark 8: It is easy to extend these results when the state to output transition probability matrix C is given by $C=C_s+\gamma C_f$ where $\gamma>0$ is of $O(\epsilon)$, C_s represents the superstates only (C in the previous section), and C_f has zero column sums. If one is interested in comparing our algorithm with algorithms adapted from other aggregation methods, it is not clear how one can obtain an aggregate C matrix to carry out the aggregate filtering recursions in this case. One way to do this is to obtain

$$\mathcal{P}(Y_k=m|X_k\in S_l)=\frac{1}{\pi_l^{ag}}\sum_{i\in S_l}\mathcal{P}(Y_k=m|X_k=i)\pi_i$$

where $\pi_l^{ag}=\sum_{i\in S_l}\pi_i,\ l=1,2,\cdots,N$ and π_i is the stationary probability $\lim_{k\to\infty}\mathcal{P}(X_k=i)$. We do not elaborate on these observation probability matrices any more to avoid repetition. We provide some brief comments based on simulation studies in the next section.

V. Examples and Simulation Studies

In this section, we look at two examples. The first is an 8-state nearly completely decomposable Markov chain with the following specifications as shown in (38) at the bottom of the next page. Obviously, for this example, $N=3,\ n=8$. The observation probability matrix is chosen to be the following:

$$C=\begin{bmatrix}0.25&0.25&0.25&0.4&0.4&0.57&0.57&0.57\\0.32&0.32&0.32&0.5&0.5&0.16&0.16&0.16\\0.43&0.43&0.43&0.1&0.1&0.27&0.27&0.27\end{bmatrix}.$$

For this example, the matrices $W_1,\,W_2,\,V_1,\,V_2$ are the following:

$$W_1=\begin{bmatrix}1&0&0\\1&0&0\\1&0&0\\0&1&0\\0&1&0\\0&0&1\\0&0&1\\0&0&1\end{bmatrix},\quad W_2=\begin{bmatrix}0&0&0&0&0\\1&0&0&0&0\\0&1&0&0&0\\0&0&0&0&0\\0&0&1&0&0\\0&0&0&0&0\\0&0&0&1&0\\0&0&0&0&1\end{bmatrix}$$

$$V_1=\begin{bmatrix}1&0&0&0&0&0&0&0\\0&0&0&1&0&0&0&0\\0&0&0&0&0&1&0&0\end{bmatrix}$$

$$V_2=\begin{bmatrix}-1&1&0&0&0&0&0&0\\-1&0&1&0&0&0&0&0\\0&0&0&-1&1&0&0&0\\0&0&0&0&0&-1&1&0\\0&0&0&0&0&-1&0&1\end{bmatrix}.$$

$L(0)$ for this example is given by the following matrix:

$$L(0)=\begin{bmatrix}-0.25&-0.25&0&0&0\\0&0&-0.5&0&0\\0&0&0&-0.4048&-0.2121\end{bmatrix}.$$

It is easy to verify that Assumptions 2.1, 2.2, 4.1, and 4.2 are satisfied in this case. Instead of finding choices for ϵ, $\overline{L}$ that satisfy (27), (28), and (32), we illustrate with the following figures that L_k is bounded for $\epsilon = 0.05$ over a simulation performed over a sequence of 25 000 points (Fig. 1) while Fig. 2 illustrates the exponential stability of the evolution $\overline{\eta}_{k+1} = \overline{\eta}_k(\tilde{A}_{21}^k L_{k+1} + A_{22}^k)$ is exponentially stable for a large enough but finite k [the plots are shown for $\overline{\eta}_k(1)$, $\overline{\eta}_k(2)$ and $\overline{\eta}_k(3)$ only to avoid congestion]. Multiple trials result in similar observations. Also, we verify Assumption 4.4 in our algorithm before carrying out the normalization procedure. The following simulation results illustrate that there are choices of ϵ for which the Assumption 4.4 is indeed satisfied. We present the following results obtained with simulations over a sequence of 25 000 points of a computer generated Markov chain and a corresponding sequence of observations. All results are averaged over 20 simulations. We compute the exact slow (aggregate) filter ζ_k, and the approximate [order of approximation being $O(\epsilon^2)$] slow filter $\tilde{\zeta}_k$ [given by (34)]. Our performance measure is $E[\|\zeta_k - \tilde{\zeta}_k\|^2]$ which (from ergodicity assumptions) is estimated by the average approximation error $\lim_{T\to\infty}(1/T)\sum_{k=1}^{T}\|\zeta_k - \tilde{\zeta}_k\|^2$. We compare the performance of our algorithm with Courtois' aggregation method and Khalil's aggregation method as found in [11]. We have already noted that our algorithm can be shown to have the same performance as the algorithm adapted from Courtois' aggregation procedure in this case. As for Khalil's aggregation method, we use the aggregate matrix proposed for the exact computation of the stationary distribution [11, eqs. (4.4) and (4.5)]. It is seen from Table I that our algorithm (or the algorithm using Courtois' aggregate matrix) is outperformed by the algorithm adapted from Khalil's aggregation method as far as the average approximation error of the aggregate filter is concerned. However, we reiterate that there is no systematic way to adapt any of these aggregation methods to obtain an $O(\epsilon^2)$ approximation to the full order filter α_k whereas our proposed algorithm provides a procedure for doing that with less number of computations specially when there are many weakly interacting superstates with small individual dimensions.

We now take another example of an 8-state Markov chain with the following A (all other specifications remain the same) where the individual $I_{n_i} + A_{ii}$ represent *i.i.d.* Markov chains as

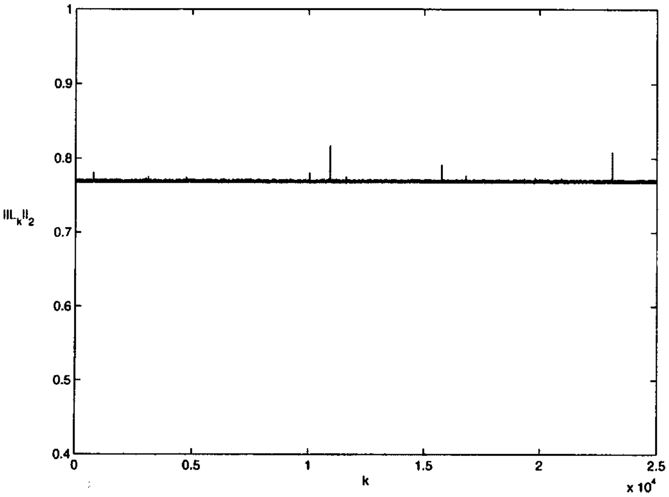

Fig. 1. Plot of $\|L_k\|_2$ versus k.

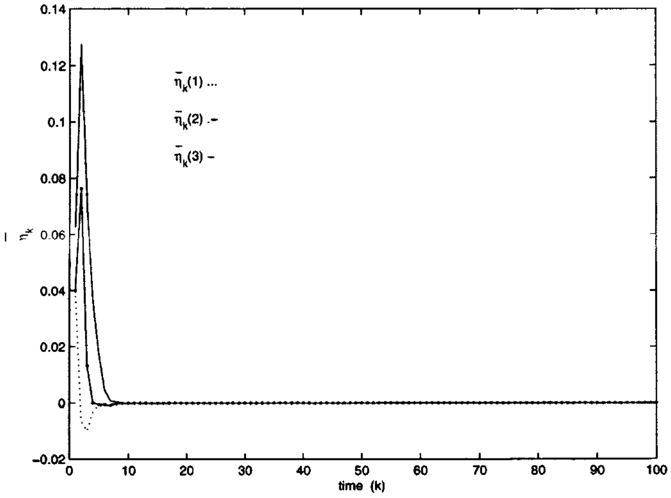

Fig. 2. Exponential stability of $\overline{\eta}_k$.

$$A = \begin{bmatrix} -0.35 & 0.25 & 0.10 & 0 & 0 & 0 & 0 & 0 \\ 0.15 & -0.65 & 0.50 & 0 & 0 & 0 & 0 & 0 \\ 0.55 & 0.15 & -0.70 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -0.3 & 0.3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.3 & -0.3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.40 & 0.25 & 0.15 \\ 0 & 0 & 0 & 0 & 0 & 0.30 & -0.42 & 0.12 \\ 0 & 0 & 0 & 0 & 0 & 0.15 & 0.35 & -0.50 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.1 & 0.15 & -1.0 & 0.6 & 0.05 & 0.0 & 0.05 & 0.05 \\ 0 & 0.1 & -0.9 & 0.5 & 0.05 & 0.05 & 0.1 & 0.1 \\ 0.01 & 0.01 & -0.4 & 0.2 & 0.05 & 0.05 & 0.04 & 0.04 \\ 0.02 & 0.42 & 0.01 & 0.01 & -0.61 & 0.025 & 0.1 & 0.025 \\ 0.45 & 0.01 & 0.4 & -1.0 & 0.01 & 0.1 & 0.01 & 0.02 \\ 0.01 & 0.05 & 0.01 & 0.01 & 0.05 & 0.01 & -0.15 & 0.01 \\ 0.03 & 0.01 & 0.03 & 0.04 & 0.01 & 0.01 & 0.01 & -0.14 \\ 0.01 & 0.05 & 0.01 & 0.01 & 0.05 & -0.16 & 0.01 & 0.02 \end{bmatrix} \qquad (38)$$

TABLE I
COMPARISON OF AVERAGE APPROXIMATION ERROR IN AGGREGATE FILTERING

ϵ	Average approximation error	
	Our algorithm (Adapted Courtois' method)	Adapted Khalil's method
0.005	3.1854×10^{-8}	1.0489×10^{-8}
0.01	1.1678×10^{-7}	5.8685×10^{-8}
0.05	1.199×10^{-5}	4.1207×10^{-6}
0.08	4.2599×10^{-5}	1.0989×10^{-5}
0.1	9.2947×10^{-5}	1.8427×10^{-5}

TABLE II
COMPARISON OF AVERAGE APPROXIMATION ERROR IN AGGREGATE FILTERING

ϵ	Average approximation error		
	Adapted Courtois' method	Adapted Khalil's method	Our algorithm
0.005	1.2623×10^{-8}	1.7858×10^{-9}	7.9708×10^{-13}
0.01	1.912×10^{-7}	1.779×10^{-8}	3.4247×10^{-11}
0.05	1.4682×10^{-5}	9.1341×10^{-7}	4.5697×10^{-8}
0.08	5.1109×10^{-5}	2.7550×10^{-6}	3.4967×10^{-7}
0.1	9.6890×10^{-5}	4.1106×10^{-6}	9.6539×10^{-7}

shown in (39) at the bottom of the page. Note that $L(0)$ in this case is given by the following matrix:

$$L(0) = \begin{bmatrix} -0.25 & -0.10 & 0 & 0 & 0 \\ 0 & 0 & -0.3 & 0 & 0 \\ 0 & 0 & 0 & -0.25 & -0.15 \end{bmatrix}.$$

In this case, we compare algorithms using Courtois' and Khalil's aggregation methods with our algorithm which computes the slow filter that is an $O(\epsilon^3)$ approximation to the exact slow filter. Note that this approximate slow filter is given by (37). Table II shows that our algorithm outperforms the algorithms based on Courtois' and Khalil's aggregation methods. We also compared our algorithm with adapted versions of Courtois' and Khalil's aggregation methods in the case when $C = C_s + \epsilon C_f$. While our algorithm and the adapted version of Courtois' aggregation method yield comparable results, the adapted version of Khalil's aggregation method results in very high values of average approximation errors, thus exposing the *ad hoc* nature of the algorithms obtained by adapting such aggregation methods. In conclusion, it is fair to say that our algorithm provides a systematic way of obtaining guaranteed approximations to the aggregate or even the full order filters by appropriately exploiting the structure of the system to reduce computations, while adapting various other aggregation methods may not always result in a better performance.

A. Numerical Issues

There are few key numerical issues that are worth pointing out.

- *Forgetting of initial conditions*:

  It is well known that the conditional probability filters for the classes of HMM's being discussed in this paper forget initial conditions exponentially fast [18]. This of course implies that if $\tilde{\zeta}_k, \tilde{\eta}_k$ are "good" approximations of ζ_k, η_k, they should also forget initial conditions exponentially fast. It is not difficult to show from (34) that $\tilde{\zeta}_k$ forgets initial conditions exponentially fast for a small enough ϵ. The matrix $\tilde{A}_1^k + \epsilon(\tilde{B}_1^k - L(0)\tilde{D}_1^k)$ in (34) can be rewritten as

  $$V_1 \left[P - \epsilon AW_2(V_2AW_2)^{-1}V_2B\right] C(Y_{k+1})W_1.$$

  It is easily seen that for a small enough ϵ, $P - \epsilon AW_2(V_2AW_2)^{-1}V_2B$ is a stochastic matrix, since B has zero row-sums. Observing that V_1, W_1 are row-allowable and column-allowable, respectively, with the nonzero entries being 1, and all the entries in the diagonal matrix $C(Y_{k+1})$ are positive, the property of forgetting of initial conditions follows immediately (by appealing to the results established in [18]) for $\tilde{\zeta}_k$ for a small enough ϵ.
- *Non-negativity of $\tilde{\zeta}_k, \tilde{\eta}_k$*:

  The exact quantities ζ_k, η_k are probability elements and hence they are nonnegative. For small values of ϵ, $\tilde{\zeta}_k$, $\tilde{\eta}_k$ therefore are likely to be nonnegative. Note that the nonnegativity of $\tilde{A}_1^k + \epsilon(\tilde{B}_1^k - L(0)\tilde{D}_1^k)$ (as discussed in the previous dot point) guarantees the nonnegativity of $\tilde{\zeta}_k$. The nonnegativity of $\tilde{\eta}_k$ does not follow easily from (35). However, one can certainly obtain sufficient conditions involving inequalities in $\epsilon, \overline{L}$ such that $\tilde{\zeta}_k, \tilde{\eta}_k$ are nonnegative $\forall k$, in addition to (27), (28), and (32), thus unnecessarily restricting the range of $\epsilon, \overline{L}$ even further. We refrain from such an exercise to reduce unnecessary complications but just note that for a small enough ϵ such nonnegativity constraints are likely to be satisfied. However, it may occasionally happen for an unfortunately large choice of ϵ for a given HMM that some elements of $\tilde{\eta}_k$ may assume small negative values. In such cases, a practical solution is to reset these quantities to zero and re-normalize the approximation to α_k (given by $\tilde{\zeta}_k V_1 + \tilde{\eta}_k V_2$). One has to rely on the property of forgetting of initial conditions to assume that the errors introduced due to these artificial resetting of values of certain quantities are not going to affect the long-term performance.

$$A = \begin{bmatrix} -0.35 & 0.25 & 0.10 & 0 & 0 & 0 & 0 & 0 \\ 0.65 & -0.75 & 0.10 & 0 & 0 & 0 & 0 & 0 \\ 0.65 & 0.25 & -0.90 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -0.30 & 0.30 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.70 & -0.70 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.40 & 0.25 & 0.15 \\ 0 & 0 & 0 & 0 & 0 & 0.60 & -0.75 & 0.15 \\ 0 & 0 & 0 & 0 & 0 & 0.60 & 0.25 & -0.85 \end{bmatrix}. \qquad (39)$$

- *How large an ϵ can one handle?*:

  Note that (27), (28), and (32) form a set of sufficient (and not necessary) conditions on ϵ, $\overline{L}$. One may be able to apply this algorithm even when the value of ϵ is larger than permitted by these inequalities although this is not known *a priori*. In such a case, a practical algorithm is one which forces $L_k(1)$ to evolve within a certain prescribed compact region such that every time it goes outside this region, it is forced back inside the region by resetting $L_k(1)$ to zero. Obviously this resetting destroys the $O(\epsilon^2)$ approximation property of $\tilde{\eta}_k$. In this case, one can at best hope for $O(\epsilon)$ approximation for η_k and hence α_k in this case. Simulation studies with such an algorithm were seen to preserve the much needed stability in cases where the values of ϵ were large enough to cause large variations in the values of $L_k(1)$. However, this was achieved at the expense of a higher approximation error.

VI. Conclusions

We address the problem of reduced-order filtering for a class of partially observed nearly completely decomposable Markov chains in this paper. We provide a systematic way to obtain approximate [with order of approximation being $O(\epsilon^2)$] aggregate filtering with $N^2 + N$ number of multiplications where the order of the Markov chain $n \gg N$. Surprisingly, this method turns out to be identical with a method obtained by adapting Courtois' aggregation method, thus proving that one can adapt Courtois' algorithm to obtain a method for obtaining an $O(\epsilon^2)$ approximation to the aggregate filter computations with reduced complexity for these classes of Markov chains. However, our algorithm also provides a systematic way to obtain $O(\epsilon^2)$ approximation to the full-order filter with reduced number of computations when there are large number of super-states in the Markov chain with small individual dimensions. In the special case where $I_{n_i} + A_{ii},\ \forall i$ represents *i.i.d* Markov chains, we can obtain $O(\epsilon^3)$ approximation to the aggregate filter and further savings in computations to the full-order filter using our algorithm whereas algorithms adapted from some other aggregation methods are incapable of achieving that. The *ad hoc* nature of algorithms adapted from other aggregation methods becomes clear through various simulation examples when the objective is to obtain reduce-order approximate filters for such classes of hidden Markov models. Extensions of these results to obtain reduced-order approximation to risk-sensitive filters and controllers for hidden Markov models [also known as partially observed Markov decision processes (POMDP)] are under investigation.

Appendix
Proof of Lemma 4.1

First, note that if $\lim_{T\to\infty}(\sum_{k=1}^{T} s_k/T) = b < a$, it is easy to show that for some large enough but finite T, $\sum_{k=1}^{T} s_k < Ta$. From Assumption 4.1, it is immediate that $\|\prod_{k=0}^{p-1}\tilde{M}_l\|_\infty \le \exp(-\beta^* p)$ for any $0 < \beta^* < \min_l \ln(1/\|\tilde{M}_l\|_\infty)$, for all $l = 1, 2, \cdots, N$ and for any $p \ge 1$. From limit theorems of sums of chain dependent random processes [19], it is known that the strong law of large numbers holds for $\ln(\overline{c}_{Y_k l}/\overline{c}_{Y_k i})$, that is,

$$\lim_{t\to\infty}\frac{1}{t}\sum_{k=1}^{t}\ln\left(\frac{\overline{c}_{Y_k l}}{\overline{c}_{Y_k i}}\right) = \gamma_{li}\ \text{w.p.1} \tag{40}$$

where γ_{li} is defined in Assumption 4.2. It follows from this and Assumption 4.2 that there exists a large enough but finite t_{li} such that for $t > t_{li}$

$$\exp\left(\sum_{k=1}^{t}\ln\left[\frac{\overline{c}_{Y_k l}}{\overline{c}_{Y_k i}}\left\|\tilde{M}_l\right\|_\infty\right]\right) < \exp(-t\lambda_{li}). \tag{41}$$

It is immediate from the above result that

$$\left\|\prod_{q=p-1}^{0} M_{k+1-q}^{(i,l)}\right\|_\infty < \exp(-p\lambda_{li}) \quad \text{for some } p > t_{li}.$$

Choosing $t^* = \max_{i,l,i\neq l} t_{li}$ and $\lambda^* = \min_{i,l,i\neq l}\lambda_{li}$, one can write for any i, $l = 1, 2, \cdots, N$, and $p > t^*$

$$\left\|\prod_{q=p-1}^{0} M_{k+1-q}^{(i,l)}\right\|_\infty < \exp(-p\lambda^*). \tag{42}$$

Since a product of block-diagonal matrices is also block-diagonal with the ith individual block in the product matrix being the product of the ith blocks in the individual matrices in the product, one can write

$$\begin{aligned}
&\left\|\prod_{q=p-1}^{0} M_{k+1-q}\right\|_\infty\\
&= \max_{i,l,i\neq l}\left(\left\|\prod_{q=p-1}^{0}\tilde{M}_l\right\|_\infty, \left\|\prod_{q=p-1}^{0} M_{k+1-q}^{(i,l)}\right\|_\infty\right)\\
&= \max\left(\max_l\left\|\prod_{q=p-1}^{0}\tilde{M}_l\right\|_\infty, \max_{i,l,i\neq l}\left\|\prod_{q=p-1}^{0} M_{k+1-q}^{(i,l)}\right\|_\infty\right)\\
&< \max\left(\exp(-p\beta^*), \exp(-p\lambda^*)\right)\\
&< \exp(-p\delta^*), \qquad \text{for } p > t^*
\end{aligned} \tag{43}$$

where $\delta^* = \min(\beta^*, \lambda^*)$.

It follows from above that $\|\prod_{q=p-1}^{0} M_{k+1-q}\|_2 < \sqrt{N(n-N)}\exp(-p\delta^*)$ for $p > t^*$. Now consider (20). From here onwards, the analysis is identical to a time-scale decomposition and averaging analysis of an adaptive control problem that can be found in [17, pp. 105–107].

Define on $\mathcal{D}$ the operator

$$\begin{aligned}
T_k(L) = l(0) + \epsilon &\left[\sum_{p=0}^{\infty} g_{k+1-p}(L_{k-p}, L_{k+1-p})\right.\\
&\left.\cdot\prod_{q=p-1}^{0} M_{k+1-q}\right].
\end{aligned} \tag{44}$$

Noting that $|g_{k+1}(L_k, L_{k+1})| < \rho(\epsilon, \overline{L})$ and letting $\overline{S} = \sum_{p=0}^{t^*}\prod_{q=p-1}^{0}\|M_{k+1-q}\|_2$ (which is bounded since t^* is finite and it is obvious that $\overline{S}$ can only grow at most exponentially at the worst case), one can show that $T_k(L)$ is a well defined operator if ϵ, $\overline{L}$ satisfy (27). (28) guarantees that it is a contraction operator and with the unique fixed point l_{k+1} that satisfies (20) by construction.

Acknowledgment

The author is grateful to Prof. J. B. Moore, Prof. B. D. O. Anderson, Prof. R. R. Bitmead, and Prof. R. Boel for helpful insights and suggestions.

References

[1] H. A. Simon and A. Ando, "Aggregation of variables in dynamic systems," *Econometrica*, vol. 29, pp. 111–138, 1961.

[2] P. J. Courtois, *Decomposability Queueing and Computer System Applications*. New York: Academic, 1977.

[3] V. G. Gatisgory and A. A. Prevozvanskii, "Aggregation of states in a Markov chain with weak interactions," *Kybernetica*, pp. 91–98, May–June 1975.

[4] F. Delebecque and J. P. Quadrat, "Optimal control of Markov chains admitting strong and weak interactions," *Automatica*, vol. 17, pp. 281–296, 1981.

[5] R. G. Phillips and P. V. Kokotovic, "A singular perturbation approach to modeling and control of Markov chains," *IEEE Trans. Automat. Contr.*, vol. 26, pp. 1087–1094, 1981.

[6] H. K. Khalil and J. O'Reilly, *Singular Perturbation Methods in Control: Analysis and Design*. New York: Academic, 1986.

[7] C. D. Meyer, "Stochastic complementation, uncoupled Markov chains, and the theory of nearly reducible systems," *SIAM Rev.*, vol. 31, pp. 240–272, June 1989.

[8] M. Coderch, A. S. Willsky, S. S. Sastry, and D. A. Casatanon, "Hierarchical aggregation of singularly perturbed finite state Markov processes," *Stochastics*, vol. 8, pp. 259–289, 1983.

[9] J. R. Rohlicek and A. S. Willsky, "The reduction of perturbed Markov generators: An algorithm exposing the role of transient states," *J. Assoc. Comput. Mach.*, vol. 35, pp. 675–696, 1988.

[10] F. Delebecque, J. P. Quadrat, and P. V. Kokotovic, "A unified view of aggregation and coherency in networks and Markov chains," *Int. J. Contr.*, vol. 40, pp. 939–952, 1984.

[11] R. W. Aldhaheri and H. K. Khalil, "Aggregation of the policy iteration method for nearly completely decomposable Markov chains," *IEEE Trans. Automat. Contr.*, vol. 36, pp. 178–187, Feb. 1991.

[12] M. Abbad, J. Filar, and T. R. Bielecki, "Algorithms for singularly perturbed limiting average markov control problems," *IEEE Trans. Automat. Contr.*, vol. 37, pp. 1421–1425, Sept. 1992.

[13] M. Abbad and J. Filar, "Perturbation and stability theory for markov control problems," *IEEE Trans. Automat. Contr.*, vol. 37, pp. 1415–1420, Sept. 1992.

[14] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, pp. 257–285, 1989.

[15] V. Krishnamurthy, S. Dey, and J. P. LeBlanc, "Blind equalization of IIR channels using hidden Markov models and extended least squares," *IEEE Trans. Signal Processing*, vol. 43, pp. 2994–3006, Dec. 1995.

[16] M. R. Azimi-Sadjadi and K. Khorasani, "Reduced order strip Kalman filtering using singular perturbation method," *IEEE Trans. Circuits Syst.*, vol. 37, pp. 284–290, Feb. 1990.

[17] B. D. O. Anderson and R. R. Bitmead, *Stability of Adaptive Systems: Passivity and Averaging Analysis*. Cambridge, MA: MIT Press, 1986.

[18] L. Shue, B. D. O. Anderson, and S. Dey, "Exponential stability of filters and smoothers for hidden Markov models," *IEEE Trans. Signal Processing*, vol. 46, pp. 2180–2194, Aug. 1998.

[19] G. L. O'Brien, "Limit theorems for sums of chain-dependent processes," *J. Appl. Prob.*, vol. 11, pp. 582–587, 1974.

Subhrakanti Dey (S'94–M'96) was born in Calcutta, India, in 1968. He received the B.T. and M.T. degrees from the Department of Electronics and Electrical Communication Engineering, Indian Institute of Technology, Kharagpur, in 1991 and 1993, respectively, and the Dr.Phil. degree from the Department of Systems Engineering, Research School of Information Sciences and Engineering, Australian National University, Canberra, in 1996.

Since February 2000, he has been a Lecturer with the Department of Electrical and Electronic Engineering, University of Melbourne. From September 1995 to September 1997 and September 1998 to February 2000, he was a Postdoctoral Research Fellow with the Department of Systems Engineering, Australian Naitonal University. From September 1997 to September 1998, he was a Postdoctoral Research Associate with the Institute for Systems Research, University of Maryland, College Park. His current research interests include signal processing for telecommunications, wireless communications and networks, performance analysis of communication networks, stochastic and adaptive estimation, and control and statistical and adaptive signal processing.