Reduced-complexity Smoothing for Hidden Markov Models

Louis Shue[1], Subhrakanti Dey[2]

Abstract

In this paper, we investigate approximate smoothing schemes for a class of hidden Markov models (HMM), namely, HMMs with underlying Markov chains that are nearly completely decomposable. The objective is to obtain substantial computational savings. Our algorithm can not only be used to obtain aggregate smoothed estimates, but can be used to obtain systematically approximate full-order smoothed estimates with computational savings, unlike many of the aggregation methods proposed earlier.

Key Words: State aggregation, nearly completely decomposable, hidden Markov model, slow-fast decomposition, reduced-complexity

1 Introduction

Nearly completely decomposable Markov chains (NCDMC) are Markov chains where one can readily identify a hierarchical structure of two or more levels. Typically consisting of a large number of states, the states of NCDMCs can be easily grouped together in what we shall term 'super-states' [5], with strong interactions (i.e. high probability of transition) between states in groups, while interactions between such groups are weak (i.e. small probability of transition). In this paper we propose an approximate smoothing algorithm for partially observed NCDMCs – i.e. a class of hidden Markov models (HMMs) – which utilises this hierarchical structure and provides an estimate to the conditional state probabilities, but with reduced order of computations compared to if exact smoothing was carried out.

NCDMCs have been extensively studied in [3], which considered NCDMCs with transition probability matrices of the form $P = I_n + A + \epsilon B$, n denoting the total number of states. Since n is often large, this led to the development of aggregation methods to reduce computations. It is well known that such Markov chains arise in studies of queueing systems and computer systems. Previous research in this direction has concentrated on approximations in obtaining the steady-state distributions. Various approaches have been put forth to derive from P an aggregate version of smaller dimension, resulting in schemes of $O(\epsilon^2)$ [3] approximation and an iterative scheme with (potentially) arbitrary accuracy [1]. There have been several other studies that contributed to the development of decomposition-aggregation methods for obtaining reduced-order approximations for uncontrolled [6], as well as controlled Markov chains [4], [7].

The state estimation of HMMs where the underlying Markov chain is an NCDMC was first investigated in [5], where apart from the structure inherent in NCDMCs, an additional assumption that the observation probabilities reflect the block structure of the Markov states was made. That is, for a given observation symbol, the state-to-observation mapping is constant for all states within the same super-state. The applicability of such block-structured observation probability matrices not only lies in modelling of management systems (where top levels of management are only interested in macro-behaviour rather than micro-behaviour) 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 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). It was demonstrated in [5] that substantial computational savings can be obtained in calculating an $O(\epsilon^2)$ approximation to the aggregate filtered estimates via a decoupling scheme for this class of hidden Markov models. It was also seen that some aggregation methods (including Courtois' method) may be adapted to obtain comparable results as far as aggregate filtered estimates are concerned. However, the algorithm proposed in [5] can be used to obtain $O(\epsilon^2)$ approximation to the full-order filtered state estimates whereas none of the aggregation methods can be adapted to achieve that. The computational savings in calculating approximate full-order estimates were not as large as in the case of the aggregate estimates, but nevertheless, the algorithm of [5] is the first to obtain systematic approximations to aggregate and full-order filtered estimates with computational savings.

The focus of this paper is to present an approxim-

[1]Centre for Signal Processing, Nanyang Technological University, Singapore 639798.

[2]Department of Electrical & Electronic Engineering, University of Melbourne, Parkville, Victoria 3052, Australia.

ate scheme for *smoothing* with performance at least comparable with smoothing obtained by using aggregate transition probability matrices via e.g. [1, 3], but with significantly reduced number of computations. Secondly, our method provides a means of computing the full-order smoothed estimates, which is not possible with previous aggregation schemes. By following closely the techniques introduced in [5], it is then seen that our smoothing algorithm is an $O(\epsilon^2)$ in approximation of the exact smoothed probabilities. Further, we can also derive approximations of arbitrary accuracy, but at the cost of increasing computational complexity. This has not been pursued in this paper.

2 Signal Model

Analogous to [1], the underlying NCDMC of n states is characterised by the row-stochastic transition probability matrix $P \in \mathbb{R}^{n\times n}$:

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

where I_n is the $(n \times n)$ identity matrix and

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

where $A_{ii} \in \mathbb{R}^{n_i\times n_i}\forall i$, $\sum_i n_i = n$, and 0 denotes zero matrices of appropriate dimensions. In (1), $\epsilon > 0$ is a small perturbation parameter. Note that $I_n + A$ is also row-stochastic, and rows of A and B sum to 0. We also make the following assumption:

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

For small ϵ, the states can be clustered into N groups such that there is strong interaction between the states in a given group, but weak interaction between the groups. Following [5], we will term the N blocks the 'superstates'. We denote the state of the full Markov chain as $X_k \in \{1, 2, \ldots, n\}$, and the l-th superstate is denoted by S_l, $l \in \{1, 2, \ldots, N\}$. Without loss of generality, let $S_1 = \{1, 2, \ldots, n_1\}$, and $S_2 = \{n_1 + 1, n_1 + 2, \ldots, n_1 + n_2\}$, etc. Consequently, in this notation

$$\Pr(X_k \in S_l) = \sum_{i=1+n_1+\ldots+n_{l-1}}^{n_1+n_2+\ldots+n_l} \Pr(X_k = i).$$

The state-to-measurement mapping is given by the measurement matrix C, where $c_{ij} = \Pr(Y_k = i|X_k = j)$, $i \in \{1, 2, \ldots, M\}$ and $j \in \{1, 2, \ldots, n\}$, and Y_k denotes the discrete observation at time k. In the subsequent discussions, we will use the shorthand $\mathcal{Y}_k$ to denote $\{Y_0, Y_1, \ldots, Y_k\}$. Since our main interest is to obtain smoothed estimates for aggregate states, we will assume (as in [5]) that C has a particular block structure, such that $c_{ij} = \bar{c}_{il}, \forall j \in S_l, \forall i$, i.e. the measurements reflect the superstates only. Finally we will make the following standard assumption since otherwise the measurements contain very little information of the Markov states.

Assumption 2.2 $min_{ij} c_{ij} \geq \underline{c} > 0$.

3 Aggregate Smoothed Estimates

3.1 Exact Smoothing Equations

To construct an HMM fixed-lag smoother we proceed similarly to [2] (which extends Kalman filtering results to smoothing results), by constructing an augmented signal model, consisting of the original Markov chain and a state X_j of which the smoothed estimate is sought after. A filtered estimate of the augmented model at time $k(> j)$ will then contain within it a filtered estimate of the state X_k, and also something equivalent to a smoothed estimate of the original model at time j.

Definition 3.1 *For each $k \geq j$, let $\mathcal{Z}_k = Z_{j,k} = \begin{bmatrix} X_j & X_k \end{bmatrix}'$ be an augmented state vector consisting of the states of the original Markov chain, at a fixed time j, and a variable time k.*

From Definition 3.1, it can be seen that $\mathcal{Z}_k$ can only assume the values $(1, 1)$, $(1, 2)$, $\ldots$, $(1, n)$, $(2, 1)$, $\ldots$, (n, n). It follows (see [8] for details) then that the transition probability matrix for such a Markov chain is $\mathcal{P} = I_n \otimes P$, where $\otimes$ denotes Kronecker product. We shall now argue that the output process Y_k of the original HMM can also be regarded as the output process of an HMM with state $\mathcal{Z}_k$, for $k \geq j$. That is, suppose that the output process associated with $\mathcal{Z}_k$ is the same as before, and consequentlyPr$(Y_k = \ell|\mathcal{Z}_k) =$ Pr$(Y_k = \ell|X_k)$, this means that the corresponding observation matrix $\mathcal{C}$ for the augmented Markov chain is $\mathcal{C} = \begin{bmatrix} C & C & \ldots & C \end{bmatrix} = 1'_n \otimes C$.

Denote the $(1 \times n^2)$ filtered probability vector for $\mathcal{Z}_k$ as $\Pi_{j,k|k}$, with each component being

$$\Pi_{j,k|k}(I) = \Pr(\mathcal{Z}_k = I|\mathcal{Y}_k)$$

where $I \in \{(1, 1), (1, 2), \ldots, (n, n)\}$. The recursion for this probability vector is (see also [8]):

$$\begin{aligned} \Pi_{j,k+1|k+1} &= \frac{1}{Z_{k+1}} \Pi_{j,k|k} \mathcal{P} \mathcal{C}_{Y_{k+1}} \\ &= \frac{1}{Z_{k+1}} \Pi_{j,k|k} \left(I_n \otimes P\right) \left(I_n \otimes C_{Y_{k+1}}\right) \end{aligned} \tag{2}$$

where $C_{Y_{k+1}} = \text{diag}(c_{mi})$ when $Y_{k+1} = m$, $i = 1, 2, \ldots, n$, and $Z_{k+1} = \Pi_{j,k|k} \mathcal{P} \mathcal{C}_{Y_{k+1}} 1_{n^2}$ is a scalar normalising constant which ensures $\sum_{i=1}^{n^2} \Pi_{j,k+1|k+1}(i) = 1$. 1_p denotes a column vector of length p with all entries equal to 1.

By definition, the i-th entry of the smoothed probability vector $\Pi_{j|j+\Delta}$ at time j with lag Δ for the unaugmented HMM is just

$$\begin{aligned}\Pi_{j|j+\Delta}(i) &= \Pr(X_j = i|\mathcal{Y}_{j+\Delta}) \\ &= \sum_{\ell=1}^{n} \Pr(X_j = i, X_{j+\Delta} = \ell|\mathcal{Y}_{j+\Delta}).\end{aligned}$$

Hence the smoothed probability vector $\Pi_{j|j+\Delta}$ for the unaugmented HMM can be evaluated by summing appropriate terms in the filtered probability vector for the augmented model

$$\begin{aligned}\Pi_{j|j+\Delta} &= \Pi_{j,j+\Delta|j+\Delta} \begin{bmatrix} 1_n & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & 1_n \end{bmatrix} \\ &= \Pi_{j,j+\Delta|j+\Delta}(I_n \otimes 1_n) \\ &= \frac{1}{Z_{j,j+\Delta}} \Pi_{j,j|j} \left(I_n \otimes P\right) \left(I_n \otimes C_{Y_{j+1}}\right) \dots \\ &\qquad \left(I_n \otimes P\right) \left(I_n \otimes C_{Y_{j+\Delta}}\right) \left(I_n \otimes 1_n\right) \\ &= \frac{1}{Z_{j,j+\Delta}} \Pi_{j,j|j} \left(I_n \otimes U_{j+1,j+\Delta} 1_n\right) \end{aligned} \tag{3}$$

where we have used the property $(A \otimes B)(C \otimes D) = AC \otimes BD$, and $Z_{j,j+\Delta}$ is a normalising constant while

$$U_{j+1,j+\Delta} = PC_{Y_{j+1}} PC_{Y_{j+2}} \dots PC_{Y_{j+\Delta}}.$$

Remark 3.1 *We can in fact rewrite* (3) *in a more compact form involving products of* $(n \times n)$ *matrices. That is, in terms of the corresponding filtered estimate*

$$\Pi_{j|j+\Delta} = \frac{1}{Z_{j,j+\Delta}} \operatorname{diag}(\Pi_{j|j}) U_{j+1,j+\Delta} 1_n.$$

3.2 Aggregate HMM Smoother

Let us consider as in [1] the nonsingular transformation $\Gamma = \begin{bmatrix} W_1 & W_2 \end{bmatrix}$ such that $I_n = \begin{bmatrix} W_1 & W_2 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \end{bmatrix}$, where $W_1 \in \mathbb{R}^{n \times N}$ and the i-th diagonal blocks in $W_2 \in \mathbb{R}^{n \times (n-N)}$, $V_1 \in \mathbb{R}^{N \times n}$ and $V_2 \in \mathbb{R}^{(n-N) \times n}$ are given as follows:

$$W_1 = \begin{bmatrix} 1_{n_1} & \cdots & 0 \\ 0 & \ddots & 0 \\ 0 & \cdots & 1_{n_N} \end{bmatrix} \tag{4a}$$

$$W_2^{(i)} = \begin{bmatrix} 0 \dots 0 \\ I_{n_i - 1} \end{bmatrix} \tag{4b}$$

$$V_1^{(i)} = \begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix} \tag{4c}$$

$$V_2^{(i)} = \begin{bmatrix} -1_{n_i - 1} & I_{n_i - 1} \end{bmatrix} \tag{4d}$$

We note that the aggregate fixed-lag smoothed state estimates can be represented as

$$\zeta_{j|j+\Delta} = \Pi_{j|j+\Delta} W_1 \tag{5}$$

where $\zeta_{j|j+\Delta} \in \mathbb{R}^{1 \times N}$ and $\zeta_{j|j+\Delta}(i) = \Pr(X_j \in S_i|\mathcal{Y}_{j+\Delta}) = \sum_{\ell \in S_i} \Pr(X_j = \ell|\mathcal{Y}_{j+\Delta})$.

We will now indicate the steps to obtain the aggregate smoothed estimates using (4a)-(4d).

Step 1. Denote the product of $\Pi_{j,k|k}$ with the i-th diagonal block of $I_n \otimes \begin{bmatrix} W_1 & W_2 \end{bmatrix}$ as $\begin{bmatrix} \zeta_{j,k}^{(i)} & \eta_{j,k}^{(i)} \end{bmatrix}$ with $\zeta_{j,k}^{(i)} \in \mathbb{R}^{1 \times N}$ and $\eta_{j,k}^{(i)} \in \mathbb{R}^{1 \times (n-N)}$, where each $\zeta_{j,k}^{(i)}$ is

$$\begin{aligned}\zeta_{j,k}^{(i)} = [&\Pr(X_j = i, X_k \in S_1|\mathcal{Y}_k) \quad \Pr(X_j = i, X_k \in S_2|\mathcal{Y}_k) \\ &\dots \quad \Pr(X_j = i, X_k \in S_N|\mathcal{Y}_k)].\end{aligned}$$

Rewriting (2) in terms of $\zeta_{j,k}^{(i)}$ and $\eta_{j,k}^{(i)}$, $i = 1, 2, \dots, n$,

$$\begin{aligned}&\Pi_{j,k+1|k+1} \left(I_n \otimes \begin{bmatrix} W_1 & W_2 \end{bmatrix}\right) \\ &\quad = \frac{1}{Z_{k+1}} \Pi_{j,k|k} \left(I_n \otimes \begin{bmatrix} W_1 & W_2 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \end{bmatrix}\right) \times \\ &\qquad \left(I_n \otimes P\right) \left(I_n \otimes C_{Y_{k+1}}\right) \left(I_n \otimes \begin{bmatrix} W_1 & W_2 \end{bmatrix}\right)\end{aligned}$$

or

$$\begin{aligned}&\begin{bmatrix} \zeta_{j,k+1}^{(1)} & \eta_{j,k+1}^{(1)} & \cdots & \zeta_{j,k+1}^{(n)} & \eta_{j,k+1}^{(n)} \end{bmatrix} \\ &\quad = \frac{1}{Z_{k+1}} \begin{bmatrix} \zeta_{j,k}^{(1)} & \eta_{j,k}^{(1)} & \cdots & \zeta_{j,k}^{(n)} & \eta_{j,k}^{(n)} \end{bmatrix} \times \\ &\qquad \left(I_n \otimes \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} PC_{Y_{k+1}} \begin{bmatrix} W_1 & W_2 \end{bmatrix}\right)\end{aligned} \tag{6}$$

Note that this recursion can also be written as:

I: Unnormalised Update

$$\begin{aligned}&\begin{bmatrix} \zeta_{j,k+1}^{(1)u} & \eta_{j,k+1}^{(1)u} & \cdots & \zeta_{j,k+1}^{(n)u} & \eta_{j,k+1}^{(n)u} \end{bmatrix} \\ &\quad = \begin{bmatrix} \zeta_{j,k}^{(1)} & \eta_{j,k}^{(1)} & \cdots & \zeta_{j,k}^{(n)} & \eta_{j,k}^{(n)} \end{bmatrix} \times \\ &\qquad \left(I_n \otimes \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} PC_{Y_{k+1}} \begin{bmatrix} W_1 & W_2 \end{bmatrix}\right)\end{aligned} \tag{7}$$

where $\zeta_{j,k+1}^{(i)u}, \eta_{j,k+1}^{(i)u}$ denotes the unnormalised variables for each i, j.

II: Normalization

$$\begin{aligned}&\begin{bmatrix} \zeta_{j,k+1}^{(1)} & \eta_{j,k+1}^{(1)} & \cdots & \zeta_{j,k+1}^{(n)} & \eta_{j,k+1}^{(n)} \end{bmatrix} \\ &= \frac{1}{Z_{k+1}} \begin{bmatrix} \zeta_{j,k+1}^{(1)u} & \eta_{j,k+1}^{(1)u} & \cdots & \zeta_{j,k+1}^{(n)u} & \eta_{j,k+1}^{(n)u} \end{bmatrix}\end{aligned} \tag{8}$$

where it is easy to show that $Z_{k+1} = \sum_{i=1}^{n} \zeta_{j,k+1}^{(i)u} 1_N$.

Step 2. At each $k = j + \Delta$, where Δ denotes the smoothing lag, the full-order smoothed probability vector is computed by summing the contribution from each X_j,

$$\begin{aligned}\Pi_{j|j+\Delta} = &\begin{bmatrix} \zeta_{j,j+\Delta}^{(1)} & \eta_{j,j+\Delta}^{(1)} & \cdots & \zeta_{j,j+\Delta}^{(n)} & \eta_{j,j+\Delta}^{(n)} \end{bmatrix} \times \\ &(I_n \otimes 1_{sn})\end{aligned} \tag{9}$$

where $\Pi_{j|j+\Delta}(i) = \Pr(X_j = i|\mathcal{Y}_{j+\Delta})$ and $1_{sn} = [1'_N \quad 0 \quad \cdots \quad 0]' \in \mathbb{R}^{n\times 1}$.

Step 3. The aggregate smoothed probability vector is then computed using (5). We then have

$$\begin{bmatrix}\zeta_{j|j+\Delta} & \eta_{j|j+\Delta}\end{bmatrix} = \Pi_{j|j+\Delta}\begin{bmatrix}W_1 & W_2\end{bmatrix}. \tag{10}$$

Remark 3.2 *Similar to Remark 3.1,* (7), (8) *can also be written as*

$$\begin{bmatrix}\zeta^{(1)u}_{j,k+1} & \eta^{(1)u}_{j,k+1}\\ \zeta^{(2)u}_{j,k+1} & \eta^{(2)u}_{j,k+1}\\ \vdots & \\ \zeta^{(n)u}_{j,k+1} & \eta^{(n)u}_{j,k+1}\end{bmatrix} = \begin{bmatrix}\zeta^{(1)}_{j,k} & \eta^{(1)}_{j,k}\\ \zeta^{(2)}_{j,k} & \eta^{(2)}_{j,k}\\ \vdots & \\ \zeta^{(n)}_{j,k} & \eta^{(n)}_{j,k}\end{bmatrix}\begin{bmatrix}V_1\\ V_2\end{bmatrix}\times PC_{Y_{k+1}}\begin{bmatrix}W_1 & W_2\end{bmatrix} \tag{11}$$

where $\begin{bmatrix}\zeta^{(i)}_{j,k+1} & \eta^{(i)}_{j,k+1}\end{bmatrix} = \frac{1}{Z_{k+1}}\begin{bmatrix}\zeta^{(i)u}_{j,k+1} & \eta^{(i)u}_{j,k+1}\end{bmatrix}$, $\forall i = 1,2,\ldots,n$.

In [5] it has been shown that such calculations (for a fixed i), can be performed approximately by decoupling $\zeta^{(i)}_{j,k}$ from $\eta^{(i)}_{j,k}$. The subsequent calculations (whether for aggregate or full-order smoothed estimates) require reduced number of computations (as they only require recursive computations of $\zeta^{(i)}_{j,k+1}$, $i = 1,2,\ldots,n$) and the estimates are of $O(\epsilon^2)$ in approximation. We will show a similar development in the next section.

4 Approximate $O(\epsilon^2)$ Aggregate Smoother

In this section, we will adopt the decoupling transformation technique as used in [5] to obtain approximate aggregate smoothed state estimates.

As indicated in (11), for fixed j and at each k the computation of $\begin{bmatrix}\zeta^{(i)}_{j,k} & \eta^{(i)}_{j,k}\end{bmatrix}$ involves only products of $(n\times n)$ matrices. We will now recall some results from [5] which are directly applicable in the present situation. Denote the transformed variables as $\begin{bmatrix}\bar{\zeta}^{(i)}_{j,k} & \bar{\eta}^{(i)}_{j,k}\end{bmatrix}$ given by

$$\begin{bmatrix}\bar{\zeta}^{(i)}_{j,k} & \bar{\eta}^{(i)}_{j,k}\end{bmatrix} = \begin{bmatrix}\zeta^{(i)}_{j,k} & \eta^{(i)}_{j,k}\end{bmatrix}\begin{bmatrix}I_N & L_k\\ 0 & I_{n-N}\end{bmatrix}. \tag{12}$$

where, it is trivial to demonstrate that

$$\begin{bmatrix}I_N & L_k\\ 0 & I_{n-N}\end{bmatrix}^{-1} = \begin{bmatrix}I_N & -L_k\\ 0 & I_{n-N}\end{bmatrix}. \tag{13}$$

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. In [5], it was shown that under certain assumptions holding involving the system parameters and ϵ, this uniform boundedness assumption is valid. Examples can certainly be found where such assumptions can be verified to hold. We will not include these conditions here, for details see [5].

To simplify the notation, let us introduce the shorthand

$$\begin{bmatrix}\tilde{A}^k_{11} & \tilde{A}^k_{12}\\ \tilde{A}^k_{21} & \tilde{A}^k_{22}\end{bmatrix} = \begin{bmatrix}V_1\\ V_2\end{bmatrix} PC_{y_{k+1}}\begin{bmatrix}W_1 & W_2\end{bmatrix} \tag{14}$$

where $\tilde{A}^k_{11} = \tilde{A}^k_1 + \epsilon\tilde{B}^k_1$, $\tilde{A}^k_{12} = \tilde{A}^k_2 + \epsilon\tilde{B}^k_2$, $\tilde{A}^k_{21} = \tilde{C}^k_1 + \epsilon\tilde{D}^k_1$ and $\tilde{A}^k_{22} = \tilde{C}^k_2 + \epsilon\tilde{D}^k_2$, and the individual terms are

$$\tilde{A}^k_1 = V_1(I_n + A)C_{y_{k+1}}W_1,\ \tilde{B}^k_1 = V_1BC_{y_{k+1}}W_1 \tag{15a}$$
$$\tilde{A}^k_2 = V_1(I_n + A)C_{y_{k+1}}W_2,\ \tilde{B}^k_2 = V_1BC_{y_{k+1}}W_2 \tag{15b}$$
$$\tilde{C}^k_1 = V_2(I_n + A)C_{y_{k+1}}W_1,\ \tilde{D}^k_1 = V_2BC_{y_{k+1}}W_1 \tag{15c}$$
$$\tilde{C}^k_2 = V_2(I_n + A)C_{y_{k+1}}W_2,\ \tilde{D}^k_2 = V_2BC_{y_{k+1}}W_2 \tag{15d}$$

Using (12) and (14) (and writing only the i-row of (11) for simplicity), we have the following recursion

$$\begin{aligned}&\begin{bmatrix}\bar{\zeta}^{(i)}_{j,k+1} & \bar{\eta}^{(i)}_{j,k+1}\end{bmatrix}\\ &= \frac{1}{Z_{k+1}}\begin{bmatrix}\bar{\zeta}^{(i)}_{j,k} & \bar{\eta}^{(i)}_{j,k}\end{bmatrix}\begin{bmatrix}I_N & -L_k\\ 0 & I_{n-N}\end{bmatrix}\begin{bmatrix}\tilde{A}^k_{11} & \tilde{A}^k_{12}\\ \tilde{A}^k_{21} & \tilde{A}^k_{22}\end{bmatrix}\begin{bmatrix}I_N & L_k\\ 0 & I_{n-N}\end{bmatrix}\\ &= \frac{1}{Z_{k+1}}\begin{bmatrix}\bar{\zeta}^{(i)}_{j,k} & \bar{\eta}^{(i)}_{j,k}\end{bmatrix}\begin{bmatrix}\tilde{A}^k_{11} - L_k\tilde{A}^k_{21} & 0\\ \tilde{A}^k_{21} & \tilde{A}^k_{21}L_{k+1}\tilde{A}^k_{22}\end{bmatrix}\end{aligned} \tag{16}$$

where L_k satisfies

$$(\tilde{A}^k_{11} - L_k\tilde{A}^k_{21})L_{k+1} = L_k\tilde{A}^k_{22} - \tilde{A}^k_{12}, \quad L_0 = 0. \tag{17}$$

It was shown in [5] that by taking advantage of fact that $\epsilon > 0$ is small, L_k can be solved iteratively by expressing L_k as power series in ϵ and truncate at some finite powers of ϵ. This approximation works only under the assumption that $L_k, \forall k$ is uniformly bounded. Here, we may add ([5] for details) that the conditions under which the uniform boundedness holds indicate that for a given set of system parameters, ϵ has to be sufficiently small, which is not surprising. We stress again that the principal reason for introducing the decoupling transformation is to reduce the order of computations, at the expense of having to approximate the true smoothed estimates. Further, from our simulation studies no appreciable decrease in performance has been observed when the approximate technique for computing (16) was used. Finally we note that the aggregate smoothed estimates can also be obtained by using Courtois' $N\times N$ aggregate matrix and the aggregate observation probability matrix of size $M\times N$. The subsequent smoothed estimate can be obtained by substituting

$$P^{\text{ag}} = I_N + \epsilon\left[V_1 - V_1AW_2(V_2AW_2)^{-1}V_2\right]BW_1 \tag{18a}$$
$$C^{\text{ag}} = (C^{\text{ag}}_{mn}) = (\Pr(Y_k = m|X_k \in S_n)) \tag{18b}$$

for P and C respectively into (11). However, we emphasise that we cannot obtain the full-state smoothed

estimates by Courtois' approach, and also we cannot easily extend these aggregation ideas to the case $C = C + \epsilon C_f$.

Before recalling the main result stated in [5], we make the following additional assumptions, the motivation and sufficient conditions for which can be found in [5].

Assumption 4.1 *ϵ is small enough that $1/Z_k = 1/\tilde{Z}_k + O(\epsilon^2)$ for $k \geq k_0$.*

Assumption 4.2 *The evolution $z_{k+1} = z_k(\tilde{A}^k_{21} L_{k+1} + \tilde{A}^k_{22})$ (namely the recursion for $\bar{\eta}^{(i)}_{j,k}$ in (16)) where $z_k \in \mathbb{R}^{1\times(n-N)}$ is exponentially stable,*

Theorem 4.1 *Consider a hidden Markov model with the system matrices P, C as given in Section 2. Assume Assumptions 2.1, 2.2, 4.1 and 4.2 hold. Also assume that $\{L_k\}$ is a sequence of uniformly bounded matrices evolving in a compact set. Denote the filtered probability vector as $\Pi_{k|k} \in \mathbb{R}^{1\times n}$, where $\Pi_{k|k}(i) = \Pr(X_k = i|\mathcal{Y}_k)$, and the aggregated formulation of $\Pi_{k|k}$ as*

$$\begin{aligned}\begin{bmatrix}\bar{\zeta}_k & \bar{\eta}_k\end{bmatrix} &= \Pi_{k|k}\begin{bmatrix}W_1 & W_2\end{bmatrix}\begin{bmatrix}I_N & L_k \\ 0 & L_{n-N}\end{bmatrix} \\ &= \begin{bmatrix}\zeta_k & \eta_k\end{bmatrix}\begin{bmatrix}I_N & L_k \\ 0 & L_{n-N}\end{bmatrix} \qquad (19)\end{aligned}$$

which can be recursively computed as

$$\begin{aligned}\begin{bmatrix}\bar{\zeta}_{k+1} & \bar{\eta}_{k+1}\end{bmatrix} &= \frac{1}{Z_{k+1}}\times \\ &\begin{bmatrix}\bar{\zeta}_k & \bar{\eta}_k\end{bmatrix}\begin{bmatrix}\tilde{A}^k_{11} - L_k\tilde{A}^k_{21} & 0 \\ \tilde{A}^k_{21} & \tilde{A}^k_{21}L_{k+1} + \tilde{A}^k_{22}\end{bmatrix} \qquad (20)\end{aligned}$$

where the meaning of each of the terms are given in the discussion for (14). Then there exists a large enough but finite k_0 such that for $k \geq k_0$ an $O(\epsilon^2)$ approximations for ζ_k (denoted as $\tilde{\zeta}_k$) and η_k (denoted as $\tilde{\eta}_k$) can be computed recursively as follows:

$$\tilde{\zeta}_{k+1} = \frac{1}{\bar{Z}_{k+1}}\tilde{\zeta}_k[\tilde{A}^k_1 + \epsilon(\tilde{B}^k_1 - L(0)\tilde{D}^k_1)] \qquad (21)$$

$$\tilde{\eta}_{k+1} = -\tilde{\zeta}_{k+1}[L(0) + \epsilon L_k(1)] \qquad (22)$$

$$L(0) = V_1 A W_2 (V_2 A W_2)^{-1} \qquad (23)$$

$$L_{k+1}(1) = (\tilde{A}^k_1)^{-1} L_k(1)\tilde{C}^k_2 + Q_k \qquad (24)$$

$$Q_k = (\tilde{A}^k_1)^{-1}\left(L(0)\tilde{D}^k_2 + L(0)\tilde{D}^k_1 L(0) - \tilde{B}^k_1 L(0) - \tilde{B}^k_2\right). \qquad (25)$$

where $\bar{Z}_{k+1} = \tilde{\zeta}_k[\tilde{A}^k_1 + \epsilon(\tilde{B}^k_1 - L(0)\tilde{D}^k_1)]1_N$, $\tilde{\zeta}_{k_0-1} = \zeta_{k_0-1}$ and $\tilde{\eta}_{k_0-1} = \eta_{k_0-1}$.

Remark 4.1 *It is clear by direct substitution of $\bar{\zeta}^{(i)}_{j,k}$ for $\bar{\zeta}_k$ and similarly replacing $\bar{\eta}_k$ by $\bar{\eta}^{(i)}_{j,k}$, for all $i = 1, 2, \ldots, n$, that Theorem 4.1 is applicable also in the calculation of (16).*

	Additions	Multiplications + Divisions[1]
FS: Exact	$\Delta n^2(n-1) + (n^2-1)$ $+n(n-1)$	$\Delta n^3 + n$
Decoupled	$\Delta nN(N-1) + (nN-1)$ $+n(N-1)$	$\Delta nN^2 + n$
AG: Exact	$\Delta n^2(n-1) + 2n^2 - N - 1$	$\Delta n^3 + N$
Decoupled	$\Delta nN(N-1) + 2nN - N - 1$	$\Delta nN^2 + N$
Courtois	$\Delta N^2(N-1) + (N^2-1)$ $+N(N-1)$	$\Delta N^3 + N$

Table 1: Comparison of number of computations for each smoothing scheme. FS: full-state; A: aggregate.

We will now indicate the savings in computations when the decoupling transformation is used. Note that the transformation by multiplying by $\begin{bmatrix}W_1 & W_2\end{bmatrix}$ is not included since this only affects the initial condition at each time j. In the following discussion, we also assume the matrices in (14) can be precomputed and held in memory. In fact, as we shall see in the sequel, only $2M$ matrices of size $N \times N$ and $(n-N)\times(n-N)$ are required since $\bar{\eta}^{(i)}_{j,k} \to 0, \forall i, j$, for k sufficiently large (for details see [5]). The comparison of the savings in computations at each j for a given smoothing lag Δ are given in Table 1.

5 Simulations

In this section, we will compare the exact full-order and aggregate smoothed estimates with those obtained using our decoupling scheme, as well as commenting on the results obtainable from the aggregation schemes of [1] and [3]. The results were obtained using 20000 data points, averaged over 20 sets, with a fixed smoothing lag of 50. This smoothing lag was chosen by noting that there was no significant improvement in smoothing performance when the lag exceeded this value. The aim is to illustrate the claim of $O(\epsilon^2)$ approximation that was made in Section 4. The error criteria used is $\mathbb{E}|\Pi_{k|k+\Delta} - \Pi_k|$, where Π_k denotes the state vector at time k; for the comparison of the various approximate schemes Π_k is replaced by the approximate smoothed probability vectors.

We will use the same example as in [5]:

$$A = \begin{bmatrix}-0.35 & 0.25 & 0.1 & 0 & 0 & 0 & 0 & 0 \\ 0.15 & -0.65 & 0.5 & 0 & 0 & 0 & 0 & 0 \\ 0.55 & 0.15 & -0.7 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -0.3 & 0.3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.3 & -0.3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.4 & 0.25 & 0.15 \\ 0 & 0 & 0 & 0 & 0 & 0.3 & -0.42 & 0.12 \\ 0 & 0 & 0 & 0 & 0 & 0.15 & 0.35 & -0.5 \end{bmatrix}$$

[1] We have only indicated the number of divisions required in principle for normalisation; in practice, more divisions are necessary to prevent numerical underflows in the calculations.

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

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

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

	Average approximation error	
ϵ	Exact Smoothing	Decoupling Scheme
0.001	1.22759636430736	1.22759762653319
0.005	1.22291595821324	1.22293122070098
0.01	1.23981392134711	1.23981692582461
0.02	1.26983658972958	1.26983627705158
0.05	1.33095599692731	1.33094864102834
0.1	1.39437014954947	1.39440441531637

Table 2: Comparison of exact (full-order) smoothing with decoupling scheme.

	Average approximation error		
ϵ	Decoupling Scheme	Adapted Courtois	Adapted Aldhaheri/Khalil
0.001	0.0000009482637	0.0000009482637	0.0000007076929
0.005	0.0000240804968	0.0000240804968	0.0000200219345
0.01	0.0000965298827	0.0000965298827	0.0000745328654
0.02	0.0003493682743	0.0003493682743	0.0002741025425
0.05	0.0016591628470	0.0016591628470	0.0012952105279
0.1	0.0046733190777	0.0046733190777	0.0035186308956

Table 3: Average approximation error of various aggregate smoothing schemes relative to exact aggregate smoothing.

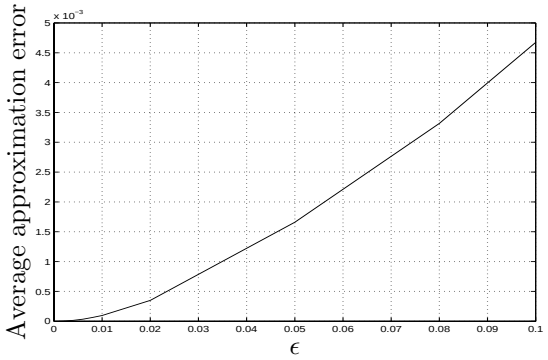

Figure 1: Aggregate smoothing via decoupling relative to the exact aggregate smoothing.

The difference between exact and approximate smoothing using our decoupling scheme is shown in Table 2. It can be seen the performance of the approximate method is comparable to the exact smoother under the noise conditions of these simulations.

A comparison of the approximate methods is tabulated in Table 3, where it can be seen that our decoupling method results in the same performance as Courtois' aggregation procedure. The $O(\epsilon^2)$ approximation can be clearly seen in Fig. 1. However, the aggregation matrix of Aldhaheri/Khalil is seen to consistently outperform our method by some small amounts. Nevertheless, we emphasise that the objective here is to compute the aggregate conditional densities with reduced computations in a systematic fashion. Our algorithm certainly offers an algorithm for doing that. Furthermore, our method offers the possibility of computing the full-order estimates while the other two aggregation methods cannot be extended to include such cases. It was also shown in [5] that for a special class of HMMs where the underlying NCDMC is *i.i.d.*, then one can obtain $O(\epsilon^3)$ to the aggregate filtered estimates with large computational savings whereas none of the aggregation methods discussed here can be adapted to achieve such savings. Such results also hold for smoothing.

6 Conclusions

In this paper, we propose an algorithm for obtaining approximate smoothed state estimates for a class of hidden Markov models with (possibly large-scale) underlying NCDMCs. These approximations are of order $O(\epsilon^2)$ and they result in substantial computational savings.

References

[1] R. W. Aldhaheri and H. K. Khalil. Aggregation of the policy iteration method for nearly completely decomposable Markov chains. *IEEE Transactions on Automatic Control*, 36(2):178–187, Feb. 1991.

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

[3] P. J. Courtois. *Decomposability, Queueing and Computer Systems Applications*. Academic Press, 1977.

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

[5] S. Dey. Reduced-complexity filtering for partially observed nearly completely decomposable Markov chains. *IEEE Transactions on Signal Processing*, to appear.

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

[7] R. G. Phillips and P. V. Kokotovic. A singular perturbation approach to modelling and control of Markov chains. *IEEE Transactions on Automatic Control*, 26:1087–1094, 1981.

[8] L. Shue, B. D. O. Anderson, and S. Dey. Exponential stability of filters and smoothers for hidden Markov models. *IEEE Transactions on Signal Processing*, 46(8):2180–2194, 1998.