Available at
www.ComputerScienceWeb.com
POWERED BY SCIENCE DIRECT

Signal Processing 83 (2003) 235–250

SIGNAL PROCESSING

www.elsevier.com/locate/sigpro

A hidden semi-Markov model with missing data and multiple observation sequences for mobility tracking

Shun-Zheng Yu, Hisashi Kobayashi*

Department of Electrical Engineering, Princeton University, NJ 08544, USA

Received 27 April 2001; received in revised form 7 December 2001

Abstract

A hidden Markov model (HMM) encompasses a large class of stochastic process models and has been successfully applied to a number of scientific and engineering problems, including speech and other pattern recognition problems, and DNA sequence comparison. A hidden semi-Markov model (HSMM) is an extension of HMM, designed to remove the constant or geometric distributions of the state durations assumed in HMM. A larger class of practical problems can be appropriately modeled in the setting of HSMM. A major restriction is found, however, in both conventional HMM and HSMM, i.e., it is generally assumed that there exists at least one observation associated with every state that the hidden Markov chain takes on. We will remove this assumption and consider the following situations: (i) observation data may be missing for some intervals; and (ii) there are multiple observation streams that are not necessarily synchronous to each other and may have different "emission distributions" for the same state. We propose a new and computationally efficient forward–backward algorithm for HSMM with missing observations and multiple observation sequences. The required computational amount for the forward and backward variables is reduced to O(D), where D is the maximum allowed duration in a state. Finally, we will apply the extended HSMM to estimate the mobility model parameters for the Internet service provisioning in wireless networks.

Keywords: Hidden Markov model (HMM); HMM with explicit duration; Hidden semi-Markov Model (HSMM); Missing data; Multiple observations; Ferguson algorithm; Forward–backward algorithm; Expectation–maximization (EM) algorithm; Maximum likelihood (ML) estimation; Maximum a posteriori probability (MAP) estimation; Mobility modeling; Wireless Internet service

1. Introduction

The hidden Markov model (HMM) technique has become one of the most successful techniques in the field of estimation and recognition (e.g., speech recognition [2,20,15], decoding in digital communications [1]). In the conventional HMM approach the state duration is either of a unit interval or implicitly assumed to be geometrically distributed to make the underlying process Markovian. A hidden semi-Markov model (HSMM) is an extension of HMM designed to allow general (i.e., non-geometric or non-exponential) distributions for the state durations [8]. Some authors [8,17,18] use terms such as "HMM with variable duration" and "HMM with explicit duration" to mean what we call in this paper an HSMM. To the best of our knowledge Ferguson [8] is the first that investigated the HSMM.

In the ordinary discrete-time HMM and HSMM, an observable output is "emitted" at every discrete time, even while the hidden Markov state remains unchanged. In some applications, however, observations may not necessarily be made frequently enough for

* Corresponding author.
E-mail address: shisashi@ee.princeton.edu (H. Kobayashi).

0165-1684/03/$ - see front matter
PII: S0165-1684(02)00378-X

one reason or another. In other words, clocks for observations may be coarser than the ones for the hidden Markov chain and its associated output emissions. In such cases, estimation of the state sequence and/or model parameters have to be made based on insufficient observations.

Another assumption commonly made in the conventional HMM and HSMM is that only one observable is associated with the hidden state. In some other applications, multiple observations may be available associated with the hidden state sequence. Furthermore, these multiple observation sequences may not be synchronous to each other. An example is found in our application discussed in Section 5, where geo-location measurement data and Web content request traffic are two observation sequences associated with a mobile wireless user, whose behavior is characterized as a hidden Markov chain. The two observation sequences are not synchronized in time. Therefore, multiple observations of such time cannot be represented as a single stream of vector-valued observations. Thus, we wish to extend the traditional "single observation sequence" model to a "multiple observation sequences" model, and develop the corresponding optimal estimation algorithm.

The HMM and HSMM have been well studied, but there are few papers in the literature that address estimation procedures for missing data. An exception is Bahl et al. [2], who considered an HMM in which observations associated with state transitions are possibly missing. They introduced a notion of "null observation", which is treated as a special output of a state transition, and the conventional algorithms for HMM are applied to a sequence that contains such null observations. Their approach, however, is not applicable to our HSMM with missing data because of its inherent Markovian assumption associated with the null observations. In the bioinformatics field, a generalization of HMM regarding two observation sequences, called "pair HMMs" or "profile HMM" [6] has been developed for modeling aligned pairs of DNA sequences, or sequence families based on multiple alignments. In this formulation, the state sequence is associated with the alignment of the gapped sequences, instead of the individual sequences. In contrast to this pair HMM, our multiple observation model deals with a common hidden Markov state sequence that is associated with each of the observation sequences that exist in parallel.

Among the conventional HMM approaches, the Viterbi algorithm [9] and the Baum–Welch algorithm [3] are perhaps the best known and most frequently used estimation or decoding algorithms. The BCJR algorithm devised by Bahl et al. [1] can be viewed as an extension of the Baum–Welch algorithm to deal with situations where the observable sequence is an output of a noisy channel whose input is the output produced by a state sequence. The conventional algorithm for HSMM proposed by Ferguson [8] may be computationally too complex to be of practical use in some applications, since it requires computation steps proportional to D^2, where D is the maximum allowable duration of any state. An alternative approach is to limit ourselves to "parametric HSMM" in which the duration distribution is characterized by parametric distributions such as Gaussian, Poisson or Gamma distributions. The duration distribution can also be combined with the state transition probabilities [22,24,5,21] and the observation probabilities [19].

The remainder of the present paper is organized as follows. Section 2 introduces our HSMM and six types of observation patterns. Section 3 develops new estimation algorithms regarding the missing observations. Section 4 describes our estimation algorithm for two observation sequences. We use the EM (estimation/maximization) algorithm to prove that our estimation algorithm with missing observations and two observation sequences is a maximum likelihood estimation solution. Section 5 applies the above results to mobility tracking in wireless Internet services provisioning. Section 6 presents some simulation results and Section 7 concludes the paper.

2. The models

Consider a Markov chain with M states that are labeled as $\{1, 2, \ldots, M\}$, in which the probability of transition from state m' to state m is denoted $a_{m'm}$, where $m, m' = 1, 2, \ldots, M$, and the initial state probability distribution is given by $\{\pi_m\}$. The Markov state is called a "hidden" state, when the state is not directly observable. If some output sequence that is probabilistically associated with the underlying hidden Markov chain

is observable, then this "doubly stochastic process" is referred to as a "hidden Markov model" or an HMM.

Let s_t denote the state that the system takes at time t, where $t = 1, 2, \ldots, T$. We denote the state sequence as $\{s_t\}$, but when we wish to be explicit about the interval, we adopt the notation s_a^b, meaning $\{s_t : a \leqslant t \leqslant b\}$. Similarly, let o_t denote the observable output at time t associated with state s_t, and let $b_m(o_t)$ be the probability of observing o_t, given $s_t = m$. We assume the "conditional independence" of outputs so that $b_m(o_a^b) = \prod_{t=a}^{b} b_m(o_t)$, where o_a^b represents the observation sequence from time a to time b. If, for instance, $\{o_t\}$ is a function of $\{s_t\}$ observed through a channel with additive white noise, the above simple product form holds.

In this paper, we assume the discrete-time model, unless stated otherwise. Thus, the duration of a given state m is a discrete random variable. In the conventional HMM we can treat only two cases regarding the state duration. It is either one time unit long or is geometrically distributed. In either case the state sequence $\{s_t\}$ becomes a Markov process, since the current state s_t depends on its past only through the most recent state s_{t-1}. We wish to allow a general distribution $p_m(d)$ for the state duration, for $d \leqslant D \leqslant \infty$, for all m. With this general distribution $p_m(d)$, the state sequence $\{s_t\}$ is no longer a Markov process, but is a semi-Markov process, hence the term HSMM.

Since we assume that the underlying semi-Markov process is not directly observable, the state sequence s_1^T and the model parameters such as $p_m(d)$ must be estimated from the observable output sequence $\{o_t\}$. We classify observation patterns into the following six types:

(a) Full observation—The outputs $\{o_t\}$ are fully observed with no missing observations. This corresponds to the conventional case that has been well studied.

(b) Deterministic observation—The outputs $\{o_t\}$ are observed only at predetermined epochs. Regular or periodic sampling is a typical example. The rest of $\{o_t\}$ will be missed, and such portion will be considerable, if the sampling is done infrequently. The number of hidden states that may be missed between any adjacent samples varies and is generally unknown.

(c) Random observation—The outputs $\{o_t\}$ are observed at randomly chosen instants. Such observation pattern may apply, when the measurement is costly or we are not interested in keeping track of state transitions so closely. In this case, some outputs may become "null" observations randomly.

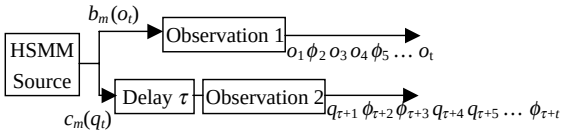

Fig. 1. An example of multiple observation sequences.

(d) State-dependent observation—Some of the outputs $\{o_t\}$ may become "null" observations, but the probability that a given o_t becomes such depends on the state s_t.

(e) Output-dependent observation—Some of the outputs $\{o_t\}$ may become "null" observations, but the probability that a given o_t becomes such depends on the output value o_t itself. For instance, when the output is too weak (in comparison with noise) at time t, such output may not be observed.

(f) Multiple observation sequences—Multiple observation sequences are associated with the hidden state sequence, and these observations may not be synchronized to each other.

In Fig. 1 we present a special case of observation type (f) defined above. We assume two sequences $\{o_t\}$ and $\{q_t\}$ are available as the outputs of an HSMM state sequence. The conditional probability that o_t appears when the state is at m is given by $b_m(o_t)$ and the corresponding conditional probability for the second output is given by $c_m(q_t)$. If we introduce some random delay τ between the two output sequences, these two sequences are no longer synchronized. The symbol ϕ_t represents the missed observation (i.e., null observation) of the output at time t.

Because the observation may not necessarily be made at every time interval, we denote the set of the observation time instants $G = \{t_1, t_2, t_3, \ldots, t_n\}$, where $1 \leqslant t_1, t_n \leqslant T$. Then we can denote the observation sequence

$$o_a^b = \{o_t : a \leqslant t \leqslant b, \text{ and } t \in G\}. \tag{1}$$

In Fig. 2(a) and (b), we give an illustrative case, where $M = 8$ and $T = 6$. State 2 lasts three time units,

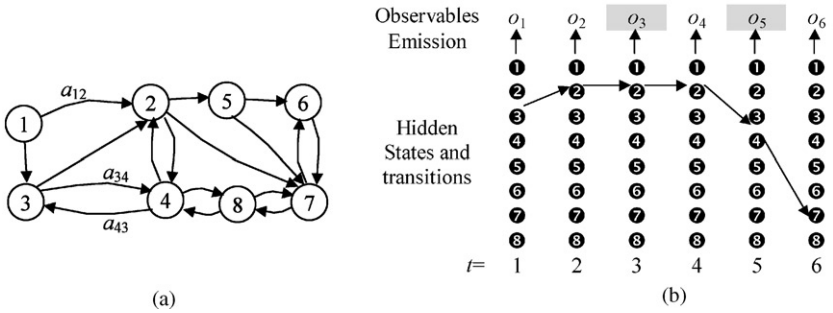

Fig. 2. (a) HSMM with $M = 8$ hidden states; (b) observable $\{o_t\}$, with $T = 6$.

i.e., $t = 2, 3, 4$. The observations o_3 and o_5 are missed, hence $G = \{t_1 = 1, t_2 = 2, t_3 = 4, t_4 = T = 6\}$.

Now we are in a position to discuss an estimation algorithm for HSMM with missed observations for each of the six types of observation patterns. We list in Table 1 important symbols used in this paper. Some of the symbols are already defined above, but the remainders will be introduced and explained in the subsequent sections.

3. Forward–backward algorithms for HSMM with missing observations

3.1. Full observation model and related work

The conventional case is that outputs emitted from the states of a hidden Markov chain are fully observed with no omission. Each state emits at least one observable output during its interval. This model has been well studied and such algorithms as the Viterbi algorithm [25,9], the Baum–Welch algorithm [3], the BCJR algorithm [1], and extensions of some of these algorithms [20] have been developed to estimate the state sequence and the model parameters. In these algorithms, the forward and backward variables are defined (although in the original Viterbi algorithm only the forward variable is defined). In the Ferguson algorithm [8], however, the following two forward variables are defined:

$$\alpha_t(m) = \Pr[o_1^t, \text{state } m \text{ ends at } t]$$
$$= \sum_{d=1}^{D} \alpha^*_{t-d}(m)\, p_m(d) \prod_{i=t-d+1}^{t} b_m(o_i) \qquad (2)$$

Table 1
Glossary of symbols

Term	Definition
M	Number of states in the HSMM.
D	The maximum duration of all states.
T	The total period of observations.
K	Number of distinct values that an observation o_t can take on.
L	Number of distinct values that an observation q_t can take on.
$a_{m,n}$	Transition probability from state m to state n.
π_m	Probability that the initial state is m.
$p_m(d)$	Probability that state m lasts d time units.
o_a^b	Observation sequence from time a to b.
$\boldsymbol{O}$	Entire observation sequence of $\{o_t\}$.
$b_m(k)$	Conditional probability that $o_t = k$ given $s_t = m$.
q_a^b	Second observation sequence from time a to b.
$\boldsymbol{Q}$	Entire observation sequence of $\{q_t\}$.
$c_m(l)$	Conditional probability that $q_t = l$ given $s_t = m$.
s_1^t	State sequence from time 1 to t.
λ	$\lambda = (\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{P}, \pi)$, the complete parameter set of HSMM.
$\alpha_t(m)$	$=\Pr[o_1^t$, state m ends at $t]$: Forward variable. See (2), (6).
$\alpha_t(m,\tau)$	Similar to $\alpha_t(m)$, defined for two observation sequences with delay τ. See (25).
$\rho_{t,d}(m)$	Forward variable. See (7), (10), (18), (22).
$\rho_{t,d}(m,\tau)$	Similar to $\rho_{t,d}(m)$, defined for two observation sequences with delay τ. See (24).
$\beta_t(m)$	$=\Pr[o_t^T \mid$ state m begins at $t]$: Backward variable. See (4), (15).
$\beta_t(m,\tau)$	Similar to $\beta_t(m)$, defined for two observation sequences with delay τ.
$\varphi_{t,d}(m)$	Backward variable. See (14), (20), (23).
$\varphi_{t,d}(m,\tau)$	Similar to $\varphi_{t,d}(m)$, defined for two observation sequences with delay τ.
$\gamma_t(m,\tau)$	$=\Pr[\boldsymbol{O}, \boldsymbol{Q}, s_t = m \mid \tau]$: Backward variable used to estimate s_t for given two observation sequences with delay τ. See (30).

and

$$\alpha_t^*(m) = \Pr\,[o_1^t, \text{state } m \text{ begins at } t+1]$$

$$= \sum_{m'=1}^{M} \alpha_t(m') a_{m'\,m} \tag{3}$$

for $t = 1, 2, \ldots, T$, with the initial and boundary conditions given by

$$\alpha_0^*(m) = \pi_m \quad \text{for } m = 1, 2, \ldots, M$$

and

$$\alpha_\tau^*(m) = 0 \quad \text{for all } \tau < 0 \text{ and } m = 1, 2, \ldots, M.$$

Similarly, the following two backward variables are defined:

$$\beta_t(m) = \Pr[o_t^T \mid \text{state } m \text{ begins at } t]$$

$$= \sum_{d=1}^{D} \beta_{t+d}^*(m) p_m(d) \prod_{i=t}^{t+d-1} b_m(o_i) \tag{4}$$

and

$$\beta_t^*(m) = \Pr\,[o_t^T \mid \text{state } m \text{ ends at } t-1]$$

$$= \sum_{m'=1}^{M} a_{mm'} \beta_t(m') \tag{5}$$

for $t = T, T-1, \ldots, 1$, with the initial and boundary conditions given by

$$\beta_{T+1}^*(m) = 1 \quad \text{for all } m = 1, 2, \ldots, M$$

and

$$\beta_\tau^*(m) = 0 \quad \text{for all } \tau > T+1 \text{ and } m = 1, 2, \ldots, M.$$

As is usually done in the HSMM formulation, we assume $a_{mm}=0$, for all $m=1, 2, \ldots, M$. This means that no transition back to the same state can occur. In the HMM, on the other hand, a_{mm} can be non-zero and the duration at state m is geometrically distributed, i.e.,

$$p_m(d) = a_{mm}^{d-1}(1 - a_{mm}), \quad 1 \leqslant d \leqslant \infty.$$

From (2) and (4) we can see that the Ferguson algorithm [8] requires a large number of computations to update the forward and the backward variables at every t. To be more specific, $D(D+1)/2$ multiplications are required just to compute the sum of product terms $\sum_{d=1}^{D} \prod_{i=t-d+1}^{t} b_m(o_i)$, and if we include other terms, it amounts to $[(D+5)D/2 + M - 1]M$ multiplications. Therefore, the Ferguson algorithm has computational complexity of $O(D^2)$ [20,21].

The product-and-sum term $\sum_{d=1}^{D} \prod_{i=t-d+1}^{t} b_m(o_i)$ required in computing (2) and (4) can be calculated more efficiently than in the original Ferguson's procedure by a recursion method, as suggested by Levinson [17] and further refined by Mitchell et al. [18]. In this method, the product-and-sum term can be computed with $(3D + M - 1)M$ multiplications, i.e., on the order of $O(D)$, but it requires retrieval of the saved probabilities, $b_m(o_t)$, obtained in the previous D observations $o_t, o_{t-1}, \ldots, o_{t-D+1}$, and D recursive steps to be performed at every t. Therefore, a total recursive steps required of the forward and backward algorithms increase by factor of D compared with the Ferguson algorithm.

From (2) and (3), we can see that by storing the products $a_{nm} p_m(d)$ for all n, m and d in advance, we can reduce computation by not performing the multiplication during the forward procedure. Similarly, storing $a_{mn} p_m(d)$ ahead of time would save computation in the backward procedure. Some authors [22,24,5,21] combine the duration distribution $p_m(d)$ and the state transition probability a_{mn} to form a new quantity $a_{mn}(d) = a_{mn} p_m(d)$, which represents the overall probability that the system stays in state m for d time units and then transits to state n. Alternatively, the duration distribution $p_m(d)$ can be combined with the observation probability $b_m(o_t)$ [19]. Obviously, with such combinations the number of parameters involved would increase significantly. For example, the number of the above defined $a_{mn}(d)$'s is M^2D, while the total number of a_{mn} and $p_m(d)$ is merely $M^2 + MD$. It means that these algorithms would require much more memory. Furthermore, computation to re-estimate these parameters would have to increase as well. The amount of computations required and the number of parameters to be estimated in several known HSMM algorithms [8,17,18,21,11] and our algorithm (to be discussed in the following sections) are compared and summarized in Table 2.

The HSMM reduces to an HMM if we set $p_m(d)$ to be a geometric distribution, as remarked earlier.

3.2. Regular and random observation models

Let us now consider cases where the observations are made independently of the hidden semi-Markov process. An example is regular or random sampling, as defined earlier. Similar to the full observation model,

Table 2
Comparison of the algorithms

	The forward computation required at each time t	The backward computation required at each time t	Computation required for re-estimation of parameters	Number of parameters to be estimated
Our algorithm	$(2D+M)M$	$(2D+M)M$	$(M^2+MD)T$	$M^2+MK+MD$
Ferguson algorithm [8]	$(0.5D^2+M)M$	$(0.5D^2+M)M$	$(M^2+MD)T+0.5D^2M$	$M^2+MK+MD$
Levinson [17] and Mitchell et al. [18]	$(3D+M)M$	$(3D+M)M$	$(M^2+2MD)T+M^2+2DM$	$M^2+MK+MD$
Krishnamurthy et al. [11] and Ramesh et al. [21]	$(MD+D)M$	$2M^2D$	$3M^2DT$	$M^2D+MK+M$

we use (2) to define the forward variable $\alpha_t(m)$ when the output o_t is observed, i.e., $t\in G$. If the observation at time t is missed (i.e., $t\notin G$), then we should re-define the forward variable $\alpha_t(m)$ as

$$\alpha_t(m)=\sum_k \Pr\,[o_1^{t-1}, o_t=k, \text{state } m \text{ ends at } t]$$

$$=\sum_{d=1}^{D}\alpha^*_{t-d}(m)\,p_m(d)\prod_{i=t-d+1}^{t-1}b_m(o_i),$$

$$t\notin G. \tag{6}$$

We introduce a new forward variable:

$$\rho_{t,d}(m)=\alpha^*_{t-d}(m)\cdot\Pr[o_{t-d+1}^{t}|s_{t-d+1}^{t}=m]$$

$$\text{for } d=1,2,\ldots,D, \tag{7}$$

and define

$$\rho_{t,0}(m)=\alpha^*_t(m), \tag{8}$$

where $\alpha^*_t(m)$ was defined earlier by (3). We obtain the following recursive formulae, for $t=1,2,\ldots,T$,

$$\rho_{0,0}(m)=\pi_m,\quad \rho_{0,d}(m)=0,$$

$$d=1,\ldots,D,\ \ 1\leqslant m\leqslant M, \tag{9}$$

$$\rho_{t,d}(m)=\begin{cases}\rho_{t-1,d-1}(m)b_m(o_t), & t\in G,\\ \rho_{t-1,d-1}(m), & t\notin G,\end{cases}$$

$$d=1,\ldots,D, \tag{10}$$

$$\alpha_t(m)=\sum_{d=1}^{D}\rho_{t,d}(m)\,p_m(d), \tag{11}$$

$$\rho_{t,0}(m)=\sum_{m'=1}^{M}\alpha_t(m')a_{m'm}. \tag{12}$$

Similarly, we define the backward variables and derive the backward recursive formulae. The backward recursions are defined as follows for $t=T,T-1,\ldots,1$:

$$\varphi_{T+1,0}(m)=1,\quad \varphi_{T+1,d}(m)=0,$$

$$d=1,\ldots,D,\quad 1\leqslant m\leqslant M, \tag{13}$$

$$\varphi_{t,d}(m)$$

$$\equiv\Pr[o_t^{t+d-1}|s_t^{t+d-1}=m]\cdot\beta^*_{t+d}(m)$$

$$=\begin{cases}\varphi_{t+1,d-1}(m)b_m(o_t), & t\in G,\\ \varphi_{t+1,d-1}(m), & t\notin G,\end{cases}\quad d\geqslant 1, \tag{14}$$

$$\beta_t(m)=\sum_{d=1}^{D}p_m(d)\varphi_{t,d}(m), \tag{15}$$

$$\varphi_{t,0}(m)=\beta^*_t(m)=\sum_{n=1}^{M}a_{mn}\beta_t(n). \tag{16}$$

Then, the state sequence and the parameters of this HSMM with regular or random sampling can be estimated using these forward and backward recursive formulae.

From (9) through (16), we can see that updating the forward variables (or the backward variables) requires $(2D+M-1)M$ multiplications at every t. The parameter $\rho_{t,d}(m)$ and $\varphi_{t,d}(m)$, for $d\geqslant 1$, can be treated as a temporary variable in the forward and backward

procedures because at every t the value $\rho_{t,d}(m)$ is determined by its value at the preceding time $t-1$, as suggested by (10). Similarly, $\varphi_{t,d}(m)$ depends on its value at $t+1$, as suggested by (14). The computational logic is easily realized by delay shift units (one for each observation interval delay).

A remark is in order regarding the computational procedures. Proper scaling is required in the recursion formulae to re-estimate the HSMM model parameters [20], because each term in the forward and backward variables is less than one and starts to decay exponentially towards zero as time index t grows (e.g., 10 or greater). The purpose of scaling is to avoid possible underflows or overflows in the computation. All we need to do is to replace the conditional probability distribution $b_m(o_t)$ used in the forward and backward formulae by a new quantity defined by

$$\dot{b}_m(o_t) = c_t b_m(o_t) \quad \text{for } m = 1,\dots,M \text{ and } t = 1,\dots,T, \tag{17}$$

where c_t is a suitably chosen scaling factor. The state estimation equation and the parameter re-estimation equations that we derive in the next sections will not be affected by these scaling factors, because both numerator and denominator of those equations (e.g., Eqs. (35) through (42)) will factor out the common term $\prod_{t=1}^{T} c_t$.

Appropriate values of the scaling factors c_t can be determined by requiring, for instance, the sum of the scaled α terms to be unity, i.e., $\sum_{m=1}^{M} \dot{\alpha}_t(m) = 1$ and $\sum_{m=1}^{M} \dot{\rho}_{t,0}(m) = 1$, at each t, for $1 \leqslant t \leqslant T$. Since the magnitudes of the scaled ρ, φ and β terms are comparable, the values of all the parameters remain within reasonable bounds, thus avoiding possible overflow or underflow problems.

3.3. State-dependent observation misses

As we defined earlier, the missing pattern of observations is called "state dependent", if some observations are missed because of a particular nature of the state the system happens to be in. Such "null" output, denoted ϕ, can be considered as an element in the output set associated with such state. We re-define the forward variable $\rho_{t,d}(m)$ of (10), for $d \geqslant 1$, as

$$\rho_{t,d}(m) = \begin{cases} \rho_{t-1,d-1}(m)\Pr[o_t | s_t = m], & t \in G, \\ \rho_{t-1,d-1}(m)\Pr[\phi_t | s_t = m], & t \notin G, \end{cases} \quad d \geqslant 1, \tag{18}$$

where

$$\Pr\{\phi_t \mid s_t = m\} + \sum_{k=1}^{K} \Pr\{o_t = k \mid s_t = m\} = 1. \tag{19}$$

Therefore, such case of state-dependent misses can be treated as a special situation of the full observation model discussed earlier. The forward recursive formulae (11) and (12) for the variables $\alpha_t(m)$ and $\rho_{t,0}(m)$ still hold in this case. Therefore, we can utilize the forward formulae (18), (11) and (12) to calculate the forward variables, where the initial values are given by (9).

Similarly we re-define the backward variable $\varphi_{t,d}(m)$ of (14), for $d \geqslant 1$, as

$$\varphi_{t,d}(m) = \begin{cases} \varphi_{t+1,d-1}(m)\Pr[o_t | s_t = m], & t \in G, \\ \varphi_{t+1,d-1}(m)\Pr[\phi_t | s_t = m], & t \notin G, \end{cases} \quad d \geqslant 1. \tag{20}$$

Then (20), (15) and (16) form the backward recursions with the initial values given by (13). We can use these forward and backward recursive formulae to estimate the parameters of the HSMM with state-dependent misses.

3.4. Output-dependent observation misses

When observation misses depend on the outputs, we define the "output-dependent miss probability" by

$$e(k) = \Pr[o_t \text{ is missed} \mid o_t = k]. \tag{21}$$

We define the forward variable $\alpha_t(m)$ by (2) or (6), depending on whether the output o_t is observed (i.e., $t \in G$) or missed (i.e., $t \notin G$). We re-define the forward variable $\rho_{t,d}(m)$ of (10) and (18), for $d \geqslant 1$, as

$$\rho_{t,d}(m) = \begin{cases} \rho_{t-1,d-1}(m) b_m(o_t), & t \in G, \\ \rho_{t-1,d-1}(m) \sum_k b_m(o_t = k)\, e(k), & t \notin G, \end{cases} \quad d \geqslant 1. \tag{22}$$

Then the forward recursive formulae (11) and (12) for the variables $\alpha_t(m)$ and $\rho_{t,0}(m)$ still hold. The forward formulae (22), (11) and (12) are used to calculate the forward variables, where the initial values are given by (9). Similarly we re-define the backward variable $\varphi_{t,d}(m)$ of (14) and (20), for $d \geqslant 1$, as

$$\varphi_{t,d}(m) = \begin{cases} \varphi_{t+1,d-1}(m) b_m(o_t), & t \in G, \\ \varphi_{t+1,d-1}(m) \sum_k b_m(k)\, e(k), & t \notin G, \end{cases}$$

$$d \geqslant 1. \tag{23}$$

Then (23), (15) and (16) form the backward recursions with the initial values given by (13). The hidden state sequence and model parameters of the HSMM with output-dependent misses can be estimated by using these forward and backward recursive formulae.

4. Estimation of HSMM with multiple observation sequences

We now discuss a case where multiple sequences of observations are available. These multiple observation sequences may have their observation intervals, starting points, sampling rates, etc. different from others. In Fig. 1 we show only two observation sequences $\{o_t\}$ and $\{q_t\}$. There is a delay τ between the two observation sequences, where τ takes on a value from $\{0, \pm 1, \pm 2, \ldots\}$. Either of the two streams can be any type of the observation and missing patterns discussed in Sections 3.1–3.4. As an example, we consider the case where $\{o_t\}$ is subject to output-dependent misses (as in Section 3.4) and $\{q_t\}$ is subject to regular or random sampling (as in Section 3.2). We denote the set of their observation time instants by G_o and G_q, respectively. Let $G_\tau = \{t : t \in G_o$ or $t + \tau \in G_q$, given $\tau\}$ with a minimum value t_τ and a maximum value T_τ. Denote by $b_m(o_t)$ and $c_m(q_{t+\tau})$ the conditional probabilities that the observations are o_t and $q_{t+\tau}$, respectively, when the system is in state m at time t. Similar to (10) and (22), we define the forward variable $\rho_{t,d}(m,\tau)$ for given delay τ by

$$\rho_{t,d}(m,\tau) \equiv \begin{cases} \rho_{t-1,d-1}(m,\tau) b_m(o_t) c_m(q_{t+\tau}), \\ \quad t \in G_o,\ t+\tau \in G_q, \\ \rho_{t-1,d-1}(m,\tau) b_m(o_t), \\ \quad t \in G_o,\ t+\tau \notin G_q, \\ \rho_{t-1,d-1}(m,\tau) c_m(q_{t+\tau}) \sum_k b_m(k)\, e(k), \\ \quad t \notin G_o,\ t+\tau \in G_q, \\ \rho_{t-1,d-1}(m,\tau) \sum_k b_m(k)\, e(k), \\ \quad t \notin G_o,\ t+\tau \notin G_q \end{cases} \tag{24}$$

for all $t_\tau \leqslant t \leqslant T_\tau$. The forward variable $\alpha_t(m,\tau)$ for two observation sequences with delay τ can be simply defined by

$$\alpha_t(m,\tau) \equiv \sum_{d=1}^{D} \rho_{t,d}(m,\tau)\, p_m(d), \quad t_\tau \leqslant t \leqslant T_\tau. \tag{25}$$

We also have

$$\rho_{t,0}(m,\tau) = \sum_{m'=1}^{M} \alpha_t(m',\tau) a_{m'm}, \quad t_\tau \leqslant t \leqslant T_\tau \tag{26}$$

with initial conditions similar to (9):

$$\rho_{t_\tau,0}(m,\tau) = \pi_m, \quad \rho_{t_\tau,d}(m,\tau) = 0, \quad d = 1, \ldots, D, \quad 1 \leqslant m \leqslant M, \tag{27}$$

where t_τ is the minimum t in G_τ.

The backward variable $\varphi_{t,d}(m,\tau)$ and $\beta_t(m,\tau)$ can be similarly defined for two observation sequences with delay τ. Denote the entire observation sequences by $\boldsymbol{O} = \{o_t : t \in G_o\}$ and $\boldsymbol{Q} = \{q_t : t \in G_q\}$. By applying the forward formulae of (24), (25) and (26), we can obtain the (joint) likelihood function of the two observation sequences $\boldsymbol{O}$ and $\boldsymbol{Q}$ for given delay τ:

$$\Pr[\boldsymbol{O}, \boldsymbol{Q} | \tau] = \sum_{m=1}^{M} \alpha_{T_\tau}(m,\tau), \tag{28}$$

where T_τ is the maximum t in G_τ.

The delay τ between the two observation sequences can be estimated from the observations, while all

the model parameters are assumed to be given and fixed:

$$\hat{\tau} = \arg\max_{\tau} \Pr[\boldsymbol{O}, \boldsymbol{Q}|\tau] = \arg\max_{\tau} \sum_{m=1}^{M} \alpha_{T_\tau}(m, \tau). \quad (29)$$

For simplicity of notation in the following sections, we use G to denote $G_{\hat{\tau}}$ for given $\hat{\tau}$ and assume $t_{\hat{\tau}} = 1$ and $T_{\hat{\tau}} = T$ without loss of generality. We denote the subsequences by $o_a^b = \{o_t : a \leqslant t \leqslant b,\ t \in G_o\}$ and $q_{a+\hat{\tau}}^{b+\hat{\tau}} = \{q_{t+\tau} : a \leqslant t \leqslant b,\ t + \hat{\tau} \in G_q\}$.

4.1. MAP estimate of states

After having obtained all the forward variables for all t and the estimate of the delay parameter $\hat{\tau}$, we can find the maximum a posteriori (MAP) estimate of state s_t, given the observations $\boldsymbol{O}, \boldsymbol{Q}$ and the model parameters, during the process of computing backward variables at every t. First, we define

$$\gamma_t(m, \hat{\tau}) = \Pr[\boldsymbol{O}, \boldsymbol{Q}, s_t = m|\hat{\tau}], \quad (30)$$

which is the joint conditional probability of the two sequences $\boldsymbol{O}$ and $\boldsymbol{Q}$ and state s_t, given the parameter of interest, $\hat{\tau}$, where $\hat{\tau}$ is determined by (29). Since we can write

$$\begin{aligned}\Pr[\boldsymbol{O}, \boldsymbol{Q}, s_{t+1} = m|\hat{\tau}] &= \Pr[\boldsymbol{O}, \boldsymbol{Q}, s_t = m \text{ and } s_{t+1} = m|\hat{\tau}] \\ &\quad + \Pr[\boldsymbol{O}, \boldsymbol{Q}, s_t \neq m, m \text{ begins at } t+1|\hat{\tau}] \quad (31)\end{aligned}$$

and

$$\begin{aligned}\Pr[\boldsymbol{O}, \boldsymbol{Q}, s_t = m|\hat{\tau}] &= \Pr[\boldsymbol{O}, \boldsymbol{Q}, s_t = m \text{ and } s_{t+1} = m|\hat{\tau}] \\ &\quad + \Pr[\boldsymbol{O}, \boldsymbol{Q}, m \text{ ends at } t,\ s_{t+1} \neq m|\hat{\tau}], \quad (32)\end{aligned}$$

we find the following *backward recursion* for $\gamma_t(m, \hat{\tau})$:

$$\gamma_t(m, \hat{\tau}) = \gamma_{t+1}(m, \hat{\tau}) + \alpha_t(m, \hat{\tau})\varphi_{t+1,0}(m, \hat{\tau}) - \rho_{t,0}(m, \hat{\tau})\beta_{t+1}(m, \hat{\tau}), \quad (33)$$

where we used the following relation based on Bayes' rule and the property of a first-order Markov chain that its current state depends on its past only through the most recent state, i.e.:

$$\begin{aligned}&\Pr[\boldsymbol{O}, \boldsymbol{Q}, m \text{ ends at } t, s_{t+1} \neq m|\hat{\tau}] \\ &\quad = \Pr[o_1^t, q_{1+\hat{\tau}}^{t+\hat{\tau}}, m \text{ ends at } t|\hat{\tau}] \\ &\qquad \times \Pr[o_{t+1}^T, q_{t+1+\hat{\tau}}^{T+\hat{\tau}}|o_1^t, q_{1+\hat{\tau}}^{t+\hat{\tau}}, m \text{ ends at } t, \hat{\tau}] \\ &\quad = \Pr[o_1^t, q_{1+\hat{\tau}}^{t+\hat{\tau}}, m \text{ ends at } t|\hat{\tau}] \\ &\qquad \times \Pr[o_{t+1}^T, q_{t+1+\hat{\tau}}^{T+\hat{\tau}}|m \text{ ends at } t, \hat{\tau}] \\ &\quad = \alpha_t(m, \hat{\tau})\varphi_{t+1,0}(m, \hat{\tau})\end{aligned}$$

and similarly

$$\begin{aligned}&\Pr[\boldsymbol{O}, \boldsymbol{Q}, s_t \neq m,\ m \text{ begins at } t+1|\hat{\tau}] \\ &\quad = \rho_{t,0}(m, \hat{\tau})\beta_{t+1}(m, \hat{\tau}).\end{aligned}$$

The initial condition for the backward variable $\gamma_t(m, \hat{\tau})$ is

$$\gamma_T(m, \hat{\tau}) = \Pr[\boldsymbol{O}, \boldsymbol{Q}, s_T = m|\hat{\tau}] = \alpha_T(m, \hat{\tau}), \quad (34)$$

which is obtained at the end of the forward algorithm. Obviously, we can calculate $\gamma_t(m, \hat{\tau}), t = T, T-1, \ldots, 1$, in conjunction with the backward algorithm, because the current value of $\gamma_t(m, \hat{\tau})$ is determined by the preceding values $\gamma_{t+1}(m, \hat{\tau})$, $\varphi_{t+1,0}(m, \hat{\tau})$ and $\beta_{t+1}(m, \hat{\tau})$ in the backward calculation, where $\alpha_t(m, \hat{\tau})$ and $\rho_{t,0}(m, \hat{\tau})$ are the stored values of the forward variables.

The MAP estimate of state s_t is defined as

$$\hat{s}_t = \arg\max_{1 \leqslant m \leqslant M} \Pr[s_t = m|\boldsymbol{O}, \boldsymbol{Q}, \hat{\tau}].$$

By Bayes' rule, we have

$$\begin{aligned}\hat{s}_t &= \arg\max_{1 \leqslant m \leqslant M} \gamma_t(m, \hat{\tau})/\Pr[\boldsymbol{O}, \boldsymbol{Q}|\hat{\tau}] \\ &= \arg\max_{1 \leqslant m \leqslant M} \gamma_t(m, \hat{\tau}), \quad t = T, T-1, \ldots, 1. \quad (35)\end{aligned}$$

From these equations, we can readily obtain the MAP estimate $\hat{s}_t$, given the observations $\boldsymbol{O}$, $\boldsymbol{Q}$, and the model parameters.

4.2. Re-estimation of the model parameters

The re-estimation algorithm is to update and improve estimates of the hidden state sequence and the model parameters, for given observation sequences $\boldsymbol{O}$ and $\boldsymbol{Q}$. First, we apply the forward–backward algorithm to obtain an estimate $\hat{\tau}$ of delay, and then a new estimate of the model parameters. Each time the model

parameters are updated, the estimation $\hat{\tau}$ of delay is re-estimated, and this interplay between the forward–backward algorithm and the ML estimation procedure is repeated until they converge to satisfactory solutions.

A posterior estimate of the transition probability $\hat{a}_{m\,n}$ is obtained as the expected number of transitions from states m to n divided by the expected total number of transitions out of state m [20,8]:

$$\hat{a}_{m\,n} = \frac{\sum_t \alpha_{t-1}(m,\hat{\tau})a_{m\,n}\beta_t(n,\hat{\tau})}{\sum_t \alpha_{t-1}(m,\hat{\tau})\varphi_{t,0}(m,\hat{\tau})}$$

$$\text{for } 1 \leqslant m \neq n \leqslant M \tag{36}$$

and

$$\hat{a}_{m\,m} = 0 \quad \text{for all } 1 \leqslant m \leqslant M. \tag{37}$$

Similarly, a posteriori estimate of the initial state probability should be given as the expected relative frequency with which the system is found in state m at $t = 1$:

$$\hat{\pi}_m = \frac{\pi_m \beta_1(m,\hat{\tau})}{\sum_m \pi_m \beta_1(m,\hat{\tau})}, \quad 1 \leqslant m \leqslant M, \tag{38}$$

where π_m is, as defined earlier, the initial distribution. We proceed in a similar manner, and revise the estimate of the state duration probability by the following expression:

$$\hat{p}_m(d) = \frac{\sum_t \rho_{t-1,0}(m,\hat{\tau})\, p_m(d)\varphi_{t,d}(m,\hat{\tau})}{\sum_t \rho_{t-1,0}(m,\hat{\tau})\beta_t(m,\hat{\tau})},$$

$$1 \leqslant m \leqslant M, \;\; d = 1,2,\ldots,D, \tag{39}$$

where the numerator represents the expected number of times that state m lasts exactly for d time units and the denominator of (39) is the expected total number of times that state m is visited regardless of any duration. Note that the denominator is equal to the sum of the numerators taken over d, for $d = 1,2,\ldots,D$.

The conditional probability of observing $o_t{=}k$ when the system is in state m is estimated by the following expression:

$$\hat{b}_m(k) = \frac{\sum_t \gamma_t(m,\hat{\tau})\delta(o_t - k)}{\sum_t \gamma_t(m,\hat{\tau})},$$

$$1 \leqslant m \leqslant M, \;\; 1 \leqslant k \leqslant K, \tag{40}$$

where $\gamma_t(m,\hat{\tau})$ is given by (33), and

$$\delta(o_t - k) = \begin{cases} 1, & o_t = k, \\ 0, & o_t \neq k. \end{cases} \tag{41}$$

Similarly,

$$\hat{c}_m(l) = \frac{\sum_t \gamma_t(m,\hat{\tau})\delta(q_{t+\hat{\tau}} - l)}{\sum_t \gamma_t(m,\hat{\tau})},$$

$$1 \leqslant m \leqslant M, \;\; 1 \leqslant l \leqslant L. \tag{42}$$

We note that observation $\{o_t\}$ and $\{q_t\}$ can be continuous-valued processes. Then the most general form of the probability density function (pdf) treated in the literature is not an arbitrary density function, but is a finite mixture of log-concave or elliptically symmetric functions (e.g., Gaussian) to insure that the parameters of the pdf can be re-estimated in a consistent way [20]. Therefore, in the continuous case, the re-estimation procedure of our HSMM can be similarly formulated as the continuous HMM [20].

From the re-estimation formulae (36) to (42), it can be seen that the re-estimation procedures can be combined with the backward algorithm, because they are the cumulative sums of the products of several variables for $t = T, T-1,\ldots,1$. During the backward recursion, we obtain the backward variables $\beta_t(m,\hat{\tau})$, $\varphi_{t,d}(m,\hat{\tau})$ and $\gamma_t(m,\hat{\tau})$, for $t = T, T-1,\ldots,1$. Multiplying them with the stored values of the forward variables $\rho_{t-1,0}(m,\hat{\tau})$ and $\alpha_{t-1}(m,\hat{\tau})$ at each t, and accumulating the products from $t = T$ through 1, we obtain the re-estimated parameters $\hat{a}_{m\,n}, \hat{b}_m(k), \hat{c}_m(l)$ and $\hat{p}_m(d)$, where $d = 1,\ldots,D$, $k = 1,\ldots,K$, $l = 1,\ldots,L$, and $n,m = 1,\ldots,M$. The backward variables $\beta_t(m,\hat{\tau})$, $\varphi_{t,d}(m,\hat{\tau})$ and $\gamma_t(m,\hat{\tau})$ can be viewed, therefore, as the "interim results" in this re-estimation procedure.

Using the theory associated with the well-known EM (expectation/maximization) algorithm [4], we can prove that the re-estimation procedure in our extended algorithm also increases the likelihood function of the model parameters. In other words, the re-estimation procedure leads to maximum likelihood estimates of these model parameters. Let λ represent the complete set of the model parameters to be estimated in the re-estimation procedure:

$$\lambda = (\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{P}, \pi), \tag{43}$$

where $\boldsymbol{A}{=}[a_{m'm}]_{M\times M}$ is the state transition probability matrix; $\boldsymbol{B} = [b_m(k)]_{M\times K}$ and $\boldsymbol{C} = [c_m(l)]_{M\times L}$, the observation probability matrices; $\boldsymbol{P} = [p_m(d)]_{M\times D}$, the state duration probability matrix; and $\boldsymbol{\pi} = [\pi_m]_{M\times 1}$, the initial state probability vector. The purpose is to

find maximum likelihood estimates of the model parameter set λ and the delay parameter τ, i.e., to find λ and τ such that the likelihood function $\Pr[\boldsymbol{O}, \boldsymbol{Q}|\lambda, \tau]$ is maximized for given $\boldsymbol{O}$ and $\boldsymbol{Q}$. Let

$$\lambda' = (\boldsymbol{A}', \boldsymbol{B}', \boldsymbol{C}', \boldsymbol{P}', \boldsymbol{\pi}') \tag{44}$$

be another possible set of the model parameters and τ' another delay parameter.

First we apply (29) to maximize the likelihood function $\Pr[\boldsymbol{O}, \boldsymbol{Q}|\lambda, \tau]$ for fixed λ. We have $\Pr[\boldsymbol{O}, \boldsymbol{Q}|\lambda, \hat{\tau}] \geqslant \Pr[\boldsymbol{O}, \boldsymbol{Q}|\lambda, \tau]$ for all τ. Hence, we let $\tau' = \hat{\tau}$. Then we use the forward–backward algorithm and the re-estimation formulae (36) to (42) for given λ and $\hat{\tau}$ to estimate parameters $\hat{a}_{m\,m'}$, $\hat{b}_m(k)$, $\hat{c}_m(l)$, $\hat{p}_m(d)$ and $\hat{\pi}_m$. We show that these re-estimated parameters can also increase the likelihood function.

Following the discussion given in [8], an auxiliary function is defined as

$$Q(\lambda, \lambda') = \sum_{s_1^T} \Pr[s_1^T, \boldsymbol{O}, \boldsymbol{Q}|\lambda, \hat{\tau}] \ln \Pr[s_1^T, \boldsymbol{O}, \boldsymbol{Q}|\lambda', \hat{\tau}]. \tag{45}$$

For fixed λ, if we can find any λ' such that $Q(\lambda, \lambda') > Q(\lambda, \lambda)$, then it can be shown that $\Pr[\boldsymbol{O}, \boldsymbol{Q}|\lambda', \hat{\tau}] > \Pr[\boldsymbol{O}, \boldsymbol{Q}|\lambda, \hat{\tau}]$, i.e., the likelihood increases. A key step in applying this result to our algorithm that involves two observation sequences with missing data is to obtain the following identity:

$$\begin{aligned}
&\frac{Q(\lambda, \lambda') - Q(\lambda, \lambda)}{P[\boldsymbol{O}, \boldsymbol{Q}|\lambda, \hat{\tau}]} \\
&= \sum_m \hat{\pi}_m \ln \frac{\pi'_m}{\pi_m} + \sum_{m',m} \hat{a}_{m'm} \ln \frac{a'_{m'm}}{a_{m'm}} \\
&\quad + \sum_{m,d} \hat{p}_m(d) \ln \frac{p'_m(d)}{p_m(d)} + \sum_{m,k} \hat{b}_m(k) \ln \frac{b'_m(k)}{b_m(k)} \\
&\quad + \sum_{m,l} \hat{c}_m(l) \ln \frac{c'_m(l)}{c_m(l)},
\end{aligned} \tag{46}$$

where $a'_{m\,m'}, b'_m(k), c'_m(l), p'_m(d)$ and π'_m are parameters to be found to maximize the auxiliary function of (46). Note that the set of steps (36) to (42) to determine $\hat{a}_{m\,m'}, \hat{b}_m(k), \hat{c}_m(l), \hat{p}_m(d)$ and $\hat{\pi}_m$ is equivalent to the E step in the EM algorithm.

The first three terms in the right hand side of (46) can be derived by following the discussion given in [8]. The last two terms are somewhat different, because of the two observation sequences with missing data. That is, we can show

$$\begin{aligned}
&\frac{1}{\Pr[\boldsymbol{O}, \boldsymbol{Q}|\lambda, \hat{\tau}]} \sum_{s_1^T} \Pr[s_1^T, \boldsymbol{O}, \boldsymbol{Q}|\lambda, \hat{\tau}] \\
&\quad \times \sum_{t \in G} \ln \frac{\Pr[o_t, q_{t+\hat{\tau}}|s_t, \lambda', \hat{\tau}]}{\Pr[o_t, q_{t+\hat{\tau}}|s_t, \lambda, \hat{\tau}]} \\
&= \sum_{s_1^T} \frac{\Pr[s_1^T, \boldsymbol{O}, \boldsymbol{Q}|\lambda, \hat{\tau}]}{\Pr[\boldsymbol{O}, \boldsymbol{Q}|\lambda, \hat{\tau}]} \sum_{t \in G} \sum_{m,k,l} \ln \frac{\Pr[k, l|m, \lambda', \hat{\tau}]}{\Pr[k, l|m, \lambda, \hat{\tau}]} \\
&\quad \times \delta(s_t - m)\delta(o_t - k)\delta(q_{t+\hat{\tau}} - l) \\
&= \sum_{m,k,l} \ln \frac{\Pr[k, l|m, \lambda', \hat{\tau}]}{\Pr[k, l|m, \lambda, \hat{\tau}]} \sum_{s_1^T} \sum_{t \in G} \frac{\Pr[s_1^T, \boldsymbol{O}, \boldsymbol{Q}|\lambda, \hat{\tau}]}{\Pr[\boldsymbol{O}, \boldsymbol{Q}|\lambda, \hat{\tau}]} \\
&\quad \times \delta(s_t - m)\delta(o_t - k)\delta(q_{t+\hat{\tau}} - l) \\
&= \sum_{m,k} \hat{b}_m(k) \ln \frac{b'_m(k)}{b_m(k)} + \sum_{m,l} \hat{c}_m(l) \ln \frac{c'_m(l)}{c_m(l)},
\end{aligned} \tag{47}$$

where G is the set that excludes the missed observation intervals, and the observations o_t and $q_{t+\hat{\tau}}$ for given state s_t and $\hat{\tau}$ are assumed independent, i.e., $\Pr[o_t, q_{t+\hat{\tau}}|s_t] = \Pr[o_t|s_t]\Pr[q_{t+\hat{\tau}}|s_t]$. By now, we can implement the M step of the EM algorithm, i.e., it can be proved that when we choose the re-estimated values of $a'_{m\,m'} = \hat{a}_{m\,m'}, b'_m(k) = \hat{b}_m(k), c'_m(l) = \hat{c}_m(l), p'_m(d) = \hat{p}_m(d)$ and $\pi'_m = \hat{\pi}_m$, then (46) is maximized.

5. Applications to mobility tracking in wireless networks

Location-based wireless services have become an active area of research in recent years [10]. These envisioned applications include navigation, emergency services, location specific advertising, location sensitive billing, local information, etc. Mobility of users presents significant technical challenges for us to provide efficient wireless access to the Internet. For a given individual mobile user, his/her location, velocity and direction will vary in time. It is therefore important to take into account dynamic mobile behavior in provisioning wireless Internet services.

We define the state of a mobile user in terms of a vector $(x_1, \ldots, x_n)$, where the ith component, x_i,

represents a value from a finite *attribute* space A_i. The attribute spaces represent properties of the mobile user such as its location, moving direction, speed, etc. The set of possible states for a mobile user is an n-dimensional vector space given by

$$S = A_1 \times \cdots \times A_n, \tag{48}$$

where $\times$ denotes the Cartesian product. The dynamic motion of a user, as defined by its time-varying attribute values, can then be described by its trajectory in this space.

We enumerate all possible states in S and label them as $1,\ldots,M$ such that the state space S can simply be represented as $S=\{1,\ldots,M\}$. The state transitions of a user are characterized by a Markov chain with transition probability matrix $\boldsymbol{A} = [a_{m'm}\colon\ m', m \in S]$. We note, however, that transitions among the states is limited due to constraints of street layout and we may assume that from a given state transitions can occur to on the order of ten neighboring states. Such considerations imply that the transition probability matrix will be highly sparse in practical applications.

We assume that the mobile user dwell time in a given state is a random variable taking values from the set $\{1,\ldots,D\}$, with a general probability distribution $p_m(d)$ and the corresponding matrix $\boldsymbol{P} = [p_m(d)\colon m \in S, d = 1,\ldots,D]$.

Our mobility model differs from previously proposed mobility models [23,16] in that it leads to a simple parametric representation of the mobile behavior that can be related to a general queuing network with multi-class users in which each service center consists of infinite server (IS) stations of multiple types. This representation allows us to capitalize on recent results in queuing and loss network theory [12]. This result in turn implies that to obtain the state distribution of mobile users, we need only two sets of parameters: the mean dwell time, d_m, in state m and the expected number of visits, e_m, which a user makes to state m in its "lifetime" (i.e., from the moment that it enters the system as an active user until it leaves the system by either moving out from the region or by turning its power off). Thus, only $2M$ pieces of numeric data per user class provide sufficient statistics of the user mobility, as far as the steady-state distribution and related performance measures are concerned.

In order to keep track of the user mobility, the semi-Markov model parameters must be estimated based on observations of the user states. This leads to an application of the type of HSMM discussed in the preceding sections. Let o_t denote the observed location of the user at time t. Note that the location of a given user is merely a portion (i.e., sub coordinates) of the state vector $(x_1,\ldots,x_n)$. The observation value o_t differs, however, from value that would correspond to the user's true location, due to geo-location error. We denote the observation probability matrix as $\boldsymbol{B} = [b_m(k)\colon\ m \in S, k = 1,\ldots,K]$, where $b_m(k)$ is the conditional probability that the geo-location value observed at time t is $o_t = k$, given that the user state is $s_t = m$. For simplicity, we assume that this probability distribution is time invariant.

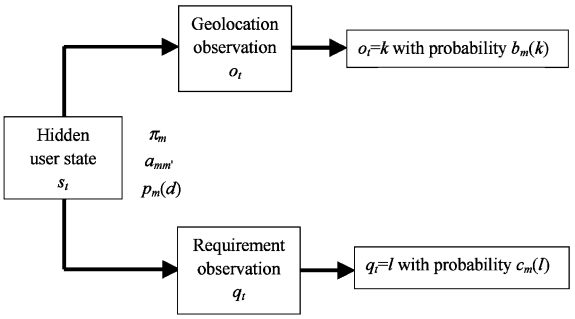

Fig. 3. The hidden semi-Markov process s_t and its geo-location and requirement observation processes (o_t, q_t).

In parallel with the (geo-location) observation process o_t defined above we now introduce the following "user requirement process" q_t, which takes on values $0,1,\ldots,L$, where $q_t = l$ means that the user requests object l (e.g. web content l). We assign $l = 0$ to a "null" object, i.e. the situation in which the user makes no request. The object l that the user requires generally depends on the user state m. Therefore, we define the user requirement probability matrix by $\boldsymbol{C} = [c_m(l)\colon m \in S, l = 0,1,\ldots,L]$, where $c_m(l)$ is the conditional probability that $q_t = l$ given $s_t = m$. We again, for simplicity, assume time invariance of this probability distribution. Fig. 3 summarizes our geo-location observation and user requirement processes (o_t, q_t), both of which are probabilistic functions of the underlying hidden semi-Markov process s_t.

The user state process s_t is characterized by $\boldsymbol{A}, \boldsymbol{P}$ and the initial state probability vector $\boldsymbol{\pi}$. Thus, the following five-tuple $\lambda = (\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{P}, \boldsymbol{\pi})$ specifies

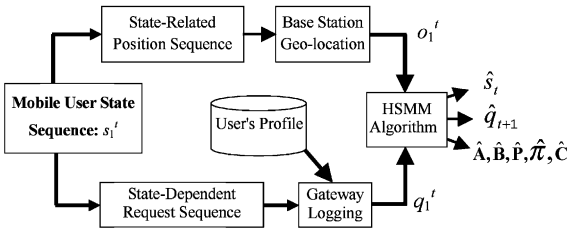

Fig. 4. Dynamic mobility tracking model.

our mobility and traffic model built on the discrete HSMM.

In this formulation, estimation of the state process s_t and re-estimation of the mobility model parameters $\lambda = (\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{P}, \boldsymbol{\pi})$ are made based on the pair processes (o_t, q_t). For various reasons, geo-location measurement and/or transmission of geo-location data may not take place frequently enough to allow precise tracking of the user's state all the time. The user may not necessarily send content requests when the user is in some state. Furthermore, the starting time of the geo-location observation sequence may not be the same as that of the requirement observation sequence and the two sequences may have different sampling instants. Therefore, the estimation and re-estimation algorithms discussed in the previous sections in handling missing observations and multiple observation sequences will be applicable to our model.

To keep track of the state of a mobile user, we apply our forward–backward and re-estimation algorithms for HSMM. The main steps in our tracking algorithm are summarized as follows:

1. Apply the *HSMM re-estimation algorithm* to obtain the initial estimates $\hat{\lambda} = (\hat{\boldsymbol{A}}, \hat{\boldsymbol{B}}, \hat{\boldsymbol{C}}, \hat{\boldsymbol{P}}, \hat{\boldsymbol{\pi}})$ of the mobility model parameters by using training data.
2. Apply the *HSMM forward–backward estimation algorithm* to estimate the current state $\hat{s}_t$ of the mobile user and to predict at time t the next requirement, $\hat{q}_{t+1}$, of the mobile user, based on both geo-location and requirement observation sequences o_1^t and q_1^t.
3. Refine the estimates $\hat{\lambda}=(\hat{\boldsymbol{A}}, \hat{\boldsymbol{B}}, \hat{\boldsymbol{C}}, \hat{\boldsymbol{P}}, \hat{\boldsymbol{\pi}})$ by applying the HSMM re-estimation algorithm to the given observation sequences.

Fig. 4 illustrates this dynamic mobility tracking model. The state sequence s_1^t associated with a mobile user is hidden in the sense that it is not directly observable. The observation sequence o_1^t is obtained from geo-location measurement. The request sequence q_1^t is obtained at a proxy server that is connected between the Internet and the wireless network. The two observation sequences o_1^t and q_1^t are the inputs to the HSMM algorithm for joint prediction and re-estimation. The HSMM parameter estimation algorithm produces an estimate, $\hat{s}_t$, of the current state of the user. In addition, a prediction, $\hat{q}_{t+1}$, of the next requirement of the user is produced as an output of the predicted value $\hat{s}_{t+1}$. That is,

$$\hat{q}_{t+1} = \arg\max_l \sum_m \left[\sum_{d=0}^{\min(t,D)} \rho_{t,d}(m) \times \left(1 - \sum_{d_1=1}^{d} p_m(d_1)\right) \right] c_m(l), \quad (49)$$

where the term in square brackets represents the conditional probability $\Pr[s_{t+1} = m, o_1^t, q_1^t | \lambda]$. Finally, it produces estimates, $\hat{\lambda} = (\hat{\boldsymbol{A}}, \hat{\boldsymbol{B}}, \hat{\boldsymbol{C}}, \hat{\boldsymbol{P}}, \hat{\boldsymbol{\pi}})$, of the model parameters. Based on the prediction and re-estimation, the system can, for instance, "prefetch" or "push" the most relevant contents to the mobile user. Thus, the mobility model can be used to enhance the performance of prefetch caching algorithms [14,13] and to characterize the wireless Internet access traffic. Such characterization can be utilized to optimize resource allocation and control policies. The reader may wonder whether one can develop dynamic mobility tracking using an "online" version of estimation algorithms for the HMM [7]. The main difficulty in such attempt would be that our model deals with HMM with explicit duration treatment, which leads to complex forward–backward and re-estimation formulae when the total length of the observation sequences is increased.

6. Simulation results

Our models and algorithms are proposed for potential applications to, for instance, "location-dependent services" that will be offered in next generation wireless Internet environments. Thus, at this point there are no really relevant empirical data that can be used to validate our models and algorithms. What we can provide in this limited circumstance is to illustrate usage of our mobility tracking model and related algorithms

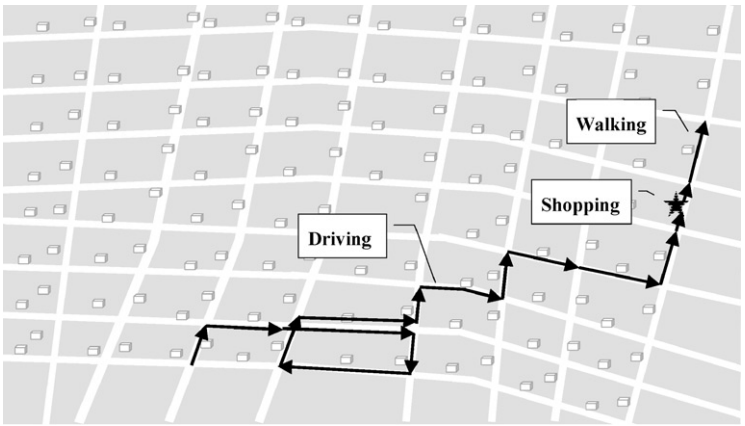

Fig. 5. An example of mobility.

by conducting Monte-Carlo simulation experiments. In such simulations, "observation" sequences in fact need to be artificially generated by assuming some underlying probabilistic models. Such questions as "what distributions should be assumed", and "whether or not some dependency or correlation should exist temporally and/or spatially" fall in the domain of what we often call "traffic or workload characterization", which by itself is a difficult yet important research problem. We hope that our work will motivate other researchers and practitioners engaged in this field to tackle such workload characterization issues.

We first specify appropriate mobile states by taking into account various constraints due to the street layout. We assume that in a serving area (of 1.6 by 1.6 km) there are 128 street segments in a mesh layout. Each street segment is about 200 m long. In referring to the state space defined by (48), we set here $n = 2$, i.e. $(x_1, x_2) \in S$, where the first attribute $x_1 \in A_1$ represents the street segment, hence $|A_1| = 256$. The second attribute $x_2 \in A_2$ takes one of the following five possible values, i.e., two possible directions of the user who may be walking along the given street segment; two possible directions if he/she is driving; and no significant motion. The last value of the attribute x_2 represents a situation where the mobile user is standing still or shopping in the area. Thus, in this simple state definition, the total number of states is $|S| = |A_1| \times |A_2| = 128 \times 5 = 640$. A user in a given state makes a transition to one of approximately ten other states associated with the neighboring street segments. In other words, the state transition probability matrix $\boldsymbol{A}$ is largely determined by the street layout, and is very sparse.

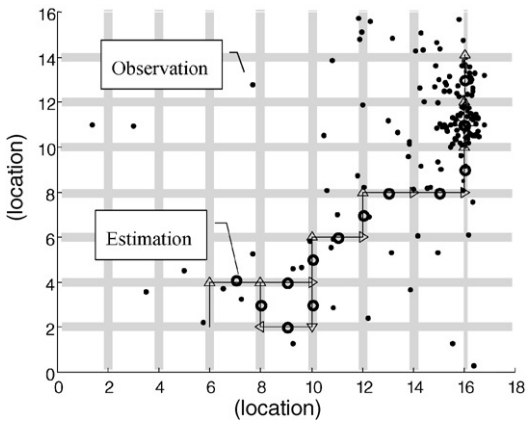

Fig. 6. Geo-location observation and state estimation.

To generate the observation sequences in the simulation, we first generate the mobile state sequence of a mobile user. An example of 1-trajectory of a mobile user is shown in Fig. 5.

In Fig. 6, we plot a set of geo-location observations: we made one observation every 20 s, and 180 measurements in an hour. Note that some of observed data are quite far from the user's true location, due to error or noise introduced in the geo-location process. We generate the observation sequence of requests made by the mobile user, by assuming some of the requests are location dependent (i.e., state dependent here).

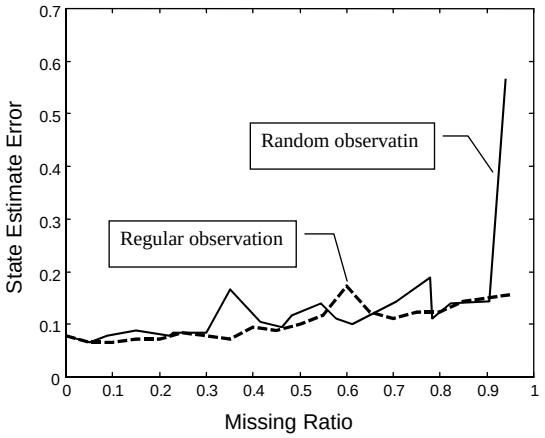

Fig. 7. State estimate errors with missing observations.

After having generated both the geo-location and requirement observation sequences, we can now start applying our forward–backward and re-estimation algorithms to estimate the model parameters and the mobile state sequence. For the two given observation sequences with the total number $T = 180$ and the total observation interval = 3600 s, the estimated state sequence $\hat{s}_1^T$ is shown in Fig. 6 in "circle" marks.

To evaluate the algorithm, we compare the estimated state sequence with the true state sequence. It shows that the state estimated from the geo-location sequence o_1^T alone is 92.22% accurate (i.e., 166 out of $T = 180$ samples). Out of the erroneously estimated states (7.78% or 14 out of T=180 samples), estimation of x_1 (i.e., user location) was still correct for 6.67% (i.e., 12 samples) and only estimation of x_2 (i.e., moving direction and velocity) was made incorrect. In the 1.11% (i.e., 2 out of 180 samples) both x_1 and x_2 were misestimated (occurred only in the beginning of the sequence). The state estimation based on the request sequence q_1^T alone is 96.11% accurate (i.e., 173 out of 180 samples). The state estimation using both the geo-location sequence o_1^T and the request sequence q_1^T achieved accuracy as high as 97.22% (i.e., 175 out of 180 samples).

State estimation errors obtained from geo-location data with regular and random observation are shown in Fig. 7 in a broken line and a solid line, respectively, as functions of percentage of missed observations, where the full sampling rate is $\frac{1}{20}$ s. From Fig. 7 we can see that the state estimation errors do not increase significantly with the increase in the missing ratio. This result indicates that base stations in a wireless network may not need to perform geo-location regularly or frequently in order to keep track of mobile users. This will possibly lead to an improvement in wireless network performance by reducing the power interference caused by geo-location implementation.

7. Conclusions

The underlying assumption in the existing HMM and HSMM models is that there is at least one observation produced per state visit and that observations are exactly the outputs (or "emissions") of states. In some applications, these assumptions are too restrictive. We extended the ordinary HMM and HSMM to the model with missing data and multiple observation sequences. We discussed the possible missing observation patterns and developed corresponding estimation algorithms. Our algorithm for the general HSMM achieves simplicity in computation and reduction in memory usage. The forward and backward algorithms involve the same computational structures and therefore can be implemented in identical hardware or programming module, in synchronism with the observed data stream, hence it can achieve high signal processing speed. The state estimation and parameter re-estimation algorithms are combined with the backward procedure, without the need for storing the backward variables, whereby reducing both computation time and storage space requirements.

We also proved that our estimation algorithm with missing observations leads to the maximum likelihood estimate. Finally, we discussed an application of HSMM with missing observations and two observation sequences to mobility tracking for providing wireless Internet services to mobile users, and have shown by simulation experiments that our algorithms indeed produce encouraging results.

Acknowledgements

This work has been supported, in part, from the grants from the New Jersey Center for Wireless Telecommunications, and NTT DoCoMo Inc.

We thank anonymous referees for their helpful comments and suggestions to improve the presentation of this work.

References

[1] L.R. Bahl, J. Cocke, F. Jelinek, J. Raviv, Optimal decoding of liner codes for minimizing symbol error rate, IEEE Trans. Inform. Theory (March 1974) 284–287.
[2] L.R. Bahl, F. Jelinek, R.L. Mercer, A maximum likelihood approach to continuous speech recognition, IEEE Trans. Pattern Anal. Mach. Intell. PAMI-5 (1983) 179–190.
[3] L.E. Baum, T. Petrie, G. Soules, N. Weiss, A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains, Ann. Math. Statist. 41 (1) (1970) 164–171.
[4] A.P. Dempster, N.M. Laird, D.B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, J. Roy. Statist. Soc. Ser. B (Methodological) 39 (1977) 1–38.
[5] P.M. Djuric, J.H. Chun, Estimation of nonstationary hidden Markov models by MCMC sampling, 1999 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP99), 1999, pp. 1737–1740.
[6] R. Durbin, S. Eddy, A. Krogh, G. Mitchison, Biological Sequence Analysis—Probabilistic Models of Proteins and Nucleic Acids, Cambridge University Press, Cambridge, MA, 1998.
[7] R.J. Elliott, L. Aggoun, J.B. Moore, Hidden Markov Models: Estimation and Control, Springer, New York, 1995.
[8] J.D. Ferguson, Variable duration models for speech, Symposium on the Application of Hidden Markov Models to Text and Speech, October 1980, 143–179.
[9] G.D. Forney, The Viterbi algorithm, Proc. IEEE 61 (3) (March 1973) 268–278.
[10] http://www-nrc.nokia.com/ietf-spatial/others.html.
[11] V. Krishnamurthy, J.B. Moore, S.H. Chung, Hidden fractal model signal processing, Signal Processing 24 (2) (August 1991) 177–192.
[12] H. Kobayashi, B.L. Mark, Product-form loss networks, in: J.H. Dshalalow (Ed.), Frontiers in Queueing: Models and Applications in Science and Engineering, CRC Press, Boca Raton, 1997, pp. 147–195.
[13] H. Kobayashi, S.-Z. Yu, Performance models of web caching and prefetching for wireless internet access, International Conference on Performance Evaluation: Theory, Techniques and Applications (PerETTA 2000), September 21–22, 2000, University of Aizu, Fukushima, Japan.
[14] H. Kobayashi, S.-Z. Yu, B. Mark, An integrated mobility and traffic model for resource allocation in wireless networks, The Third ACM International Workshop on Wireless Mobile Multimedia (WoWMoM-2000), August 11, 2000.
[15] K.Y. Lee, J. Lee, Recognition of noisy speech by a nonstationary AR HMM with gain adaptation under unknown noise, IEEE Trans. Speech Audio Process. 9 (7) (October 2001) 741–746.
[16] K.K. Leung, W.A. Massey, W. Whitt, Traffic models for wireless communication networks, IEEE J. Select. Areas Comm. 12 (8) (October 1994) 1353–1364.
[17] S.E. Levinson, Continuously variable duration hidden Markov models for automatic speech recognition, Comput. Speech Language 1 (1) (1986) 29–45.
[18] C. Mitchell, M. Harper, L. Jamieson, On the complexity of explicit duration HMMs, IEEE Trans. Speech Audio Process. 3 (2) (May 1995) 213–217.
[19] Y.K. Park, C.K. Un, O.W. Kwon, Modeling acoustic transitions in speech by modified hidden Markov models with state duration and state duration-dependent observation probabilities, IEEE Trans. Speech Audio Process. 4 (5) (September 1996) 389–392.
[20] L.R. Rabiner, A tutorial on hidden Markov models and selected application in speech recognition, Proc. IEEE 77 (2) (February 1989) 257–286.
[21] P. Ramesh, J.G. Wilpon, Modeling state durations in hidden Markov models for automatic speech recognition, Proceedings of ICASSP, 1992, pp. 381–384.
[22] B. Sin, J.H. Kim, Nonstationary hidden Markov model, Signal Processing 46 (1995) 31–46.
[23] S. Tekinay, Modeling and analysis of cellular networks with highly mobile heterogeneous sources, Ph.D. Dissertation, School of Information Technology and Engineering, George Mason University, 1994.
[24] S.V. Vaseghi, State duration modeling in hidden Markov models, Signal Processing 41 (1995) 31–41.
[25] A.J. Viterbi, Error bounds for convolutional codes and an asymptotically optimal decoding algorithm, IEEE Trans. Inform. Theory IT-13 (April 1967) 260–269.