Least Square Estimation of a Hidden Markov Chain Parameters

Junko Murakami and Thomas Taylor

***Abstract*— Among the algorithms used for the parameter estimate of discrete-time time-homogeneous Hidden Markov Models (HMM), Baum-Welch (B-W) algorithm, which is used to find the maximum likelihood estimate, is by far the most popular algorithm. However, one of its well-known shortcomings is an "overfitting" problem: it tends to fit the observation sequence well but not so much the HMM which generated the sequence, when the data size is small.**

Experiments show the least square error (LSE) estimator (or Bayes estimate) outperforms the B-W algorithm in such occasions. Although the computational complexity for the LSE estimator could be very high with a naive approach, we have shown that a polynomial complexity in the data size (with still exponential complexity in the state space size) can be achieved using an algorithm proposed in this paper, making the LSE estimation quite feasible in various applications.

***Keywords*— hidden Markov models, Baum-Welch algorithm, least square error estimate, Bayes estimate**

I. INTRODUCTION

HMMs are of considerable interest for science and for various applications. They consist of a Markov chain with "hidden" states and emissions that are statistically dependent on the states but can be observed. The model is parameterized by two conditional probability matrices, the transition and emission matrices. These models are widely applied in various fields such as speech recognition [9], [10], [11], [16], gene analysis [14], [15], [19], image and pattern recognition [5], [6], [13], magnetic recording channel analysis [1], precipitation models [20], heart rate variability [8], and so on.

The B-W algorithm [2], [3], one implementation of the Expectation Maximization (EM) algorithm [7], [17], is used in numerous HMM applications. However, among some of its well-known limitations [7], [12], it has an "overfitting" problem: it selects parameter values which have a high likelihood for the given observation sequence but which could have quite low likelihoods for other observation sequences generated by the same model, resulting in a lack of stability in its estimated parameter values. In particular, the parameter values that maximize the likelihood are often far away from the true parameters when the data set is small.

On the other hand, the LSE estimate, which is the exact expected value of the parameter set given an observation sequence, does not have such problem. Compared to what could typically be obtained using the B-W algorithm, the LSE estimates are remarkably closer to the actual parameters in such case.

Furthermore, while the B-W algorithm has a strong dependency on the initial parameters chosen, the LSE estimator is consistent, needs only one time computation, and can be obtained using a deterministic formula. However, if it is obtained in a naive way, the computational complexity increases exponentially with respect to the length of the observation sequence. The algorithm that we introduce here reduces this complexity to polynomial in the sequence length, while it is still exponential in the state space size.

J. Murakami is with School of Mathematics, Statistics and Computer Science, Victoria University of Wellington, PO Box 600, Wellington 6001, New Zealand junko.murakami@mcs.vuw.ac.nz

T. Taylor is with Department of Mathematics and Statistics, Arizona State University, PO Box 871804, Tempe, AZ 85287-1804, USA

II. HIDDEN MARKOV MODELS

In a discrete-space HMM, we have two sequences of states: a Markov chain state sequence, $S_1, S_2, S_3, \ldots$, and the observation sequence, $O_1, O_2, O_3, \ldots$, where the subscripts represent the corresponding discrete time t. The states of the Markov chain cannot be observed (are hidden), while the observed states depend on them.

To simplify the expressions, let $S^{i,j} = (S_i, S_{i+1}, \ldots, S_j)$ and $O^{i,j} = (O_i, O_{i+1}, \ldots, O_j)$. The model has the following properties:

$$P(S_{t+1}|O^{1,t}, S^{1,t}) = P(S_{t+1}|S_t) \qquad (1)$$

and

$$P(O_t|O^{1,t-1}, O^{t+1,n}, S^{1,n}) = P(O_t|S_t) \qquad (2)$$

for any $1 \leq t \leq n$. The first property (1) is a property of a Markov chain: the probability of transition from a state at time t, S_t, to the next state, S_{t+1}, depends only on the current state S_t and not on any prior states. The second property (2) shows that the probability of the observed state O_t at time t depends only on the current Markov chain state S_t, and not on any other states of the state sequence nor on any other observed states (see Fig. 1). In this paper we consider the state space $\xi \triangleq \{0, 1, \ldots, m-1\}$, for some non-negative integer m, and the emission state space $\{0, 1\}$.

Hence, to describe a HMM, we need two probability matrices, an $m \times m$ transition matrix $A = \{a_{ij}\}$, where $a_{ij} = P(S_{t+1} = j \mid S_t = i)$, and an $m \times 2$ emission matrix $B = \{b_{iu}\}$, where $b_{iu} = P(O_t = u \mid S_t = i)$; plus a vector for the initial state of the Markov chain $\pi = (\pi_0, \pi_1, \ldots, \pi_{m-1})$, $\pi_i = P(S_1 = i)$, where $i, j \in \xi$ and $u \in \{0, 1\}$ (see Fig. 2). We let the parameter set for HMM be $\theta = \{\pi, A, B\}$.

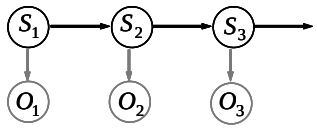

Fig. 1. A Markov chain state sequence $(S_1, S_2, S_3, \ldots)$ and the observation sequence, $(O_1, O_2, O_3, \ldots)$

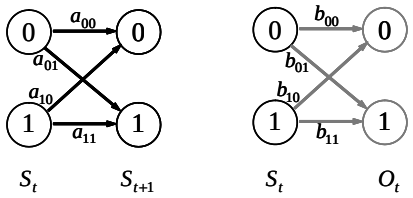

Fig. 2. Transition and observation probabilities when $m = 2$.

III. LEAST SQUARE ESTIMATION

What we compute here is the exact value of the LSE (Bayes) estimate $\hat{\theta} = E\left(\theta \mid O^{1,n}\right) = \int \theta\, P\left(\theta \mid O^{1,n}\right) d\theta$.

A. Formula for Expected Parameter Values

Using Bayes formula, assuming θ to be uniformly distributed (i.e., letting $P(\theta) = 1$), and taking the marginal distribution of $O^{1,n}$, we have

$$\hat{\theta} = \frac{1}{P\left(O^{1,n}\right)} \sum_{S^{1,n} \in \Omega_n} \int \theta\, P\left(S^{1,n}, O^{1,n} \mid \theta\right) d\theta \tag{3}$$

where Ω_n is the set of all possible values of $S^{1,n}$.

Now, by the properties of HMMs, the value of the integration above depends only on the following quantities:

$$k_{ij} = \{\text{number of times } S_t = i \text{ and } S_{t+1} = j\} \tag{4}$$

and

$$l_{iu} = \{\text{number of times } S_t = i \text{ and } O_t = u\}. \tag{5}$$

Let K be an $m \times m$ such that $K = \{k_{ij}\}$, and L be an $m \times 2$ matrix such that $L = \{l_{iu}\}$. Here the values of K and L depend on each value of $S^{1,n} \in \Omega_n$ and a particular $O^{1,n}$ observed. For notational convenience, we first define $d_{i\tilde{j}}$ such that $d_{i\tilde{j}} = k_{ij}$ if and only if $j \equiv i + \tilde{j}$ in modulo m, which makes d_{i0} the diagonal element in the ith row of the matrix K. For the sake of simpler expression, we let $\pi_i = \frac{1}{m}$, for all $i \in \xi$ (although it is easy to formulate the estimate for π). Furthermore, in order to avoid "averaging up" the probability distribution by symmetrically equivalent states, we evaluate the integrals under a condition $a_{ii} \geq a_{i+1,i+1}$ for all $i \in \{0, 1, \ldots, m-2\}$. Then, we have the following:

$$\begin{aligned} P\left(O^{1,n}\right) &= \sum_{S^{1,n} \in \Omega_n} \int P\left(S^{1,n}, O^{1,n} \mid \theta\right) d\theta \\ &= \frac{1}{m} \sum_{S^{1,n} \in \Omega_n} f \cdot G \sum_{i_{m-1}=0}^{\tilde{p}_{m-1}} \sum_{i_{m-2}=0}^{\tilde{p}_{m-2}} \\ &\qquad \cdots \sum_{i_1=0}^{\tilde{p}_1} g_{m-1,\,1}(I_{m-1})\; g_{m-2,\,1}(I_{m-2}) \\ &\qquad\qquad \cdots g_{11}(I_1) \cdot g_2(I_1), \end{aligned} \tag{6}$$

where

$$f = f(L) = \prod_{i=0}^{m-1} \frac{l_{i0}!\; l_{i1}!}{(l_{i0} + l_{i1} + 1)!}, \tag{7}$$

$$G = G(K) = \prod_{k=0}^{m-1} \left[\prod_{j=-1}^{m-4} \sum_{i=0}^{p_k(j)} g_{k0}(i,j) \right], \tag{8}$$

$$g_{k0}(i,j) = \frac{(-1)^i p_k(j)!}{i!(p_k(j) - i)!(d_{k,\,m-j-3} + i + 1)}, \tag{9}$$

$$g_{k1}(I_k) = \frac{(-1)^{i_k} \tilde{p}_k!}{i_k!(\tilde{p}_k - i_k)!\phi_k(I_k)}, \text{ and} \tag{10}$$

$$g_2(I_1) = \frac{(\phi_1(I_1) + d_{00})!\, \tilde{p}_0!}{(\phi_1(I_1) + d_{00} + \tilde{p}_0 + 1)!} \tag{11}$$

for I_k, $p_k(i)$, $\tilde{p}_k$, and $\phi_k(I_k)$ defined as

$$I_k = (i_k, i_{k+1}, \ldots, i_{m-2}, i_{m-1}), \tag{12}$$

$$p_k(i) = \sum_{j=m-i-2}^{m-1} d_{kj} + i + 1, \tag{13}$$

$$\tilde{p}_k = p_k(m-3) = \sum_{j=1}^{m-1} d_{kj} + m - 2, \text{ and} \tag{14}$$

$$\phi_k(I_k) = \sum_{j=k}^{m-1} (d_{j0} + i_j) + m - k. \tag{15}$$

In above, terms in the product or summation which upper index is less than the lower index should be set to 1.

Since $\int P\left(S^{1,n}, O^{1,n} \mid \theta\right) d\theta$ is actually a function of k_{ij} and l_{iu} once $O^{1,n}$ and $S^{1,n}$ are fixed, $i, j \in \xi$ and $u \in \{0, 1\}$, we let $\Phi(K, L) = \int P\left(S^{1,n}, O^{1,n} \mid \theta\right) d\theta$. (Recall $\{d_{i\tilde{j}}\}$ are just relabeled entries of K.) Furthermore, define K_{ij} (or L_{iu}) as the matrix that is obtained by first copying K (or L) then incrementing the entry on the ith row and jth column (or uth column) by one. Then the estimates are obtained as follows:

$$\hat{a}_{ij} = \frac{\sum_{S^{1,n} \in \Omega_n} \Phi(K_{ij}, L)}{\sum_{S^{1,n} \in \Omega_n} \Phi(K, L)} \tag{16}$$

and

$$\hat{b}_{iu} = \frac{\sum_{S^{1,n} \in \Omega_n} \Phi(K, L_{iu})}{\sum_{S^{1,n} \in \Omega_n} \Phi(K, L)} \tag{17}$$

Example with $m = 2$:

With the state space size $m = 2$, the above expressions become much simpler. Since the indices for the inside multiplication of G, from $j = -1$ to $m - 4$, is decreasing if $m = 2$, we have neither G nor g_{k0} in our final formula, which is

$$P\left(O^{1,n}\right) = \frac{1}{2} \sum_{s^{1,n} \in \Omega_n} f \sum_{i=0}^{k_{10}} g_{11}(i) g_2(i), \tag{18}$$

where

$$f = \prod_{i=0}^{1} \frac{l_{i0}!\, l_{i1}!}{(l_{i0} + l_{i1} + 1)!}, \tag{19}$$

$$g_{11}(i) = \frac{(-1)^i k_{10}!}{i!\,(k_{10} - i)!\,(k_{11} + i + 1)}, \text{ and} \tag{20}$$

$$g_2(i) = \frac{(k_{00} + k_{11} + i + 1)! k_{01}!}{(n - k_{10} + i + 1)!}. \tag{21}$$

The estimates can be obtained using (16) and (17); however, in order to avoid an exponential complexity with respect to the sequence length n, we use the method described below.

B. Algorithm for Polynomial Complexity

Although (6) involves a summation over the elements $S^{1,n}$ in Ω_n, which has m^n distinct elements, multiple number of elements in Ω_n correspond to the same value of (K, L) given a particular observation sequence $O^{1,n}$.

First, we see K and L can be expressed in a slightly different way. Let $V = \{v_{ij}\}$ be an $m \times m$ matrix such that $v_{ij} = (j-1)^{i-1}$ with 0^0 defined as 1, and let $R = \{r_{ij}\}$ be the inverse of V. Note V is a Vandermonde matrix and invertible for any positive integer m. Also, let k_i, $i = 1, 2, \ldots, m-1$, be the number of i's in $S^{1,n}$, and let l_1 be the number of 1's in $O^{1,n}$. Then, we observe the following equalities:

$$k_{00} = \sum_{i=1}^{m-1} \left[\sum_{j=1}^{m-1} k_{ij} - 2k_i + \left(\sum_{j=1}^{m-1} r_{ji} \right) (S_1^i + S_n^i) \right] + n - 1, \tag{22a}$$

$$k_{0j} = -\sum_{i=1}^{m-1} \left(k_{ij} + r_{ji} S_1^i \right) + k_j, \tag{22b}$$

$$k_{i0} = -\sum_{j=1}^{m-1} \left(k_{ij} + r_{ij} S_n^j \right) + k_i \tag{22c}$$

$$l_{00} = \sum_{i=1}^{m-1} (l_{i1} - k_i) - l_1 + n, \tag{22d}$$

$$l_{01} = -\sum_{i=1}^{m-1} l_{i1} + l_1, \quad \text{and} \tag{22e}$$

$$l_{i0} = -l_{i1} + k_i. \tag{22f}$$

The above implies that K and L can be determined by

$$\omega = \left\{ K_{m-1}, K^{m-1,\,m-1}, L_{m-1}, S_1, S_n \right\}, \tag{23}$$

where $K_{m-1} = (k_1, k_2, \ldots, k_{m-1})$, $K^{m-1,\,m-1}$ is K without the first row and the first column, $L_{m-1} = (l_{11}, l_{21}, \ldots, l_{m-1,\,1})$, and S_1 and S_n are the first and last states of $S^{1,n}$.

Now, let $h_n(\omega)$ be the number of sequences $S^{1,\,n} \in \Omega_n$ that generate the same value of ω given a particular $O^{1,\,n}$. The values of $h_n(\omega)$ can be obtained by the algorithm below. Given an observation sequence $O^{1,n}$ the algorithm sequentially finds the values of $h_t(\omega)$ for t starting from 1 up to n. Because of the symmetry in the distribution, note we only need to find $h_n(\omega)$ with S_1 fixed (to 0 in below).

Algorithm for finding h_n:

Let $h_1(\omega_0) = 1$, where ω_0 is an ω-value such that all the entries in the elements in ω is 0.

for t from 1 to $n-1$
  with all $\omega = (K_{m-1}, K^{m-1,\,m-1}, L_{m-1}, 0, S_t)$ such that $h_t(\omega) > 0$.
    (for the case $S_{t+1} = 0$)
    Increment $h_{t+1}(K_{m-1}, K^{m-1,\,m-1}, L_{m-1}, 0, 0)$ by the value $h_t(\omega)$

    (for the case $S_{t+1} = 1$)
    for S_{t+1} from 1 to $m-1$
      Obtain $\hat{\omega}$ from ω by incrementing
        (i) $k_{s_{t+1}}$ in K_{m-1} by one,
        (ii) $k_{s_t,\,s_{t+1}}$ in $K^{m-1\,m-1}$ by one, and
        (iii) $l_{s_{t+1},\,1}$ in L_{m-1} by O_{t+1},
      then by letting S_t take the value S_{t+1}.

      Increment $h_{t+1}(\hat{\omega})$ by the value $h_t(\omega)$.
    end for
  end with
end for

Let $\widetilde{\Omega}_n = \{\omega \mid h(\omega) > 0\}$. Since $0 \le k_{ij} \le n-1$ and $0 \le l_{iu} \le n$, while there are only limited values ω can logically take (e.g., k_{00}-value cannot exceed $k_0 - 1$, etc.), and since the algorithm goes through each $\omega \in \widetilde{\Omega}_n$ $n-1$ times, the computational complexity is less than cmn^{m^2} for some small constant c.

Example with $m = 2$:

With the state space size $m = 2$, the algorithm is as shown below. Again S_1 is fixed as 0.

Algorithm for finding h_n (for $m = 2$):

Let $h_1(0, 0, 0, 0, 0) = 1$.

for t from 1 to $n - 1$
  with all $\omega = (k_1, k_{11}, l_{11}, 0, S_t)$ such that $h_t(\omega) > 0$
    increment $h_{t+1}(\ k_1,\ k_{11},\ l_{11},\ 0,\ 0)$ and
    $h_{t+1}(k_1 + 1,\ k_{11} + S_t,\ l_{11} + O_{t+1},\ 0,\ 1)$
    by the value $h_t(\omega)$
end for

IV. RESULTS

The two estimators, the B-W and LSE, are compared in the following way:

A. Experimenting with Special-Case HMMs

First, we define $|\det(A)|$ as 'small' if $-0.1 \leq |\det(A)| \leq 0.1$ and 'large' if $0.9 \leq |\det(A)| \leq 1$. Similarly, $|b_{00} - b_{10}|$ as 'small' if $-0.1 \leq |b_{00} - b_{10}| \leq 0.1$ and 'large' if $0.9 \leq |b_{00} - b_{10}| \leq 1$. Then, 200 θ-values, with $m = 2$, are randomly picked for each of the possible combinations of 'small' and 'large' values; i.e., for

Case I: small $|\det(A)|$ and $|b_{00} - b_{10}|$ values,
Case II: small $|\det(A)|$ and large $|b_{00} - b_{10}|$ values,
Case III: large $|\det(A)|$ and small $|b_{00} - b_{10}|$ values, and
Case IV: large $|\det(A)|$ and $|b_{00} - b_{10}|$ values.

Although the differences between the cases are not too obvious if given just a small part of observation sequences, some of the observation sequences that appeared during the simulation are shown in Table I to get some idea how the sequences would look.

Then, using each θ-value picked, a Markov chain sequence, then an observation sequence of length n are generated and used to find the corresponding LSE and B-W estimates, where $n = 50$, 100, and 150. The B-W estimates are obtained by choosing the final estimate that corresponds to the largest number of initial estimates among 15 randomly picked initial estimates. (Empirically we see the one with the largest basin is most likely to give the maximum likelihood.) Finally, the root mean square errors (the Euclidean distance between the true parameter set and the estimate) with respect to the parameter set (a, b, x, y) are obtained; those for $n = 150$ are plotted in Fig. 3, while the mean and variance of the errors for all three n-values are shown in Table II. The plots for $n = 50$ and 100 are not shown here because they are very similar to the one for $n = 150$.

TABLE I
EXAMPLES OF THE OBSERVATION SEQUENCES FOR SPECIAL CASES

	observation sequences
Case I	(0 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0 0 0 1 1 0 0 0 0 . . .)
	(0 1 1 0 0 1 1 1 0 1 1 1 1 1 0 1 1 1 1 0 0 1 0 1 1 . . .)
	(1 0 0 1 1 0 1 1 1 1 1 1 0 1 1 0 1 1 1 1 0 0 1 1 0 1 . . .)
	(1 0 1 1 1 1 1 1 0 1 1 1 1 1 1 0 0 1 0 1 1 1 1 1 1 0 0 1 . . .)
Case II	(0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0 1 0 0 1 1 0 1 1 0 . . .)
	(0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 1 1 0 0 0 0 . . .)
	(1 0 1 0 1 0 0 0 0 0 0 0 0 0 1 1 1 0 1 1 0 0 1 1 1 0 . . .)
	(1 1 1 1 1 1 1 1 0 1 1 0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 . . .)
Case III	(0 0 0 0 1 0 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 . . .)
	(0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 . . .)
	(1 0 1 1 0 0 1 1 1 0 1 1 1 0 0 1 0 1 1 1 1 0 1 0 0 0 . . .)
	(1 1 0 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 1 . . .)
Case IV	(0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 0 . . .)
	(0 1 1 0 1 0 1 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 0 0 0 . . .)
	(1 1 1 0 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 . . .)
	(1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 . . .)

Regardless of the n-values, when $|b_{00} - b_{10}|$ is small (in Cases I and III, and significantly so in Case I), the means of the root mean squar error of the LSE estimates are less than those of the B-W estimates, while it is the opposite when $|b_{00} - b_{10}|$ is large (in Cases II and IV, and significantly so in Case IV). Meanwhile, we see the variance is always by far greater for the B-W estimates in all four cases, which indicates the overfitting problem of the B-W estimator.

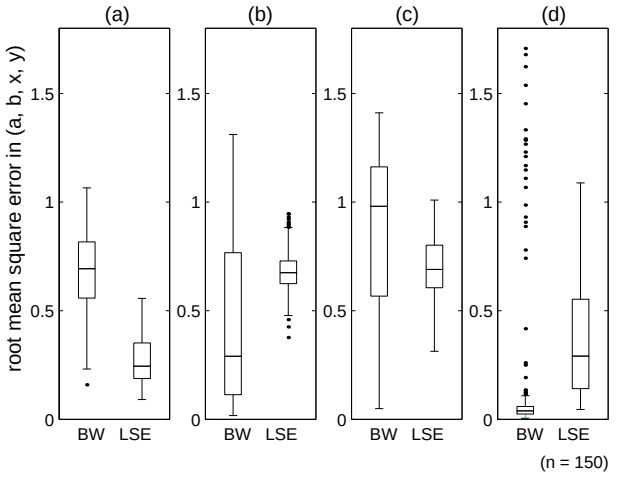

Fig. 3. The root mean square error for special cases: (a) small $|\det(A)|$ and $|b_{00} - b_{10}|$ values (Case I), (b) small $|\det(A)|$ and large $|b_{00} - b_{10}|$ values (Case II), (c) large $|\det(A)|$ and small $|b_{00} - b_{10}|$ values (Case III), and (d) large $|\det(A)|$ and $|b_{00} - b_{10}|$ values (Case IV).

B. Experimenting with wide range of HMMs

Again with the state space size $m = 2$ and for the length $n = 150$, 200 HMMs are picked in a same way described in the above, except that this time θ-values are

TABLE II

MEAN AND VARIANCE OF ROOT MEAN SQUARE ERRORS IN SPECIAL CASES

(a) $n = 50$

		mean		variance	
$\lvert\det(A)\rvert$	$\lvert b_{00} - b_{10}\rvert$	BW	LSE	BW	LSE
small	small	0.723	0.265	0.214	0.030
small	large	0.457	0.658	0.399	0.011
large	small	0.939	0.736	0.334	0.010
large	large	0.360	0.594	0.516	0.081

(b) $n = 100$

		mean		variance	
$\lvert\det(A)\rvert$	$\lvert b_{00} - b_{10}\rvert$	BW	LSE	BW	LSE
small	small	0.689	0.254	0.207	0.030
small	large	0.461	0.661	0.412	0.013
large	small	0.884	0.723	0.355	0.010
large	large	0.201	0.433	0.464	0.081

(c) $n = 150$

		mean		variance	
$\lvert\det(A)\rvert$	$\lvert b_{00} - b_{10}\rvert$	BW	LSE	BW	LSE
small	small	0.681	0.273	0.236	0.032
small	large	0.437	0.684	0.394	0.014
large	small	0.853	0.698	0.350	0.020
large	large	0.173	0.464	0.380	0.083

randomly picked with respect to the whole range of both the determinant $\det(A)$ and the difference $b_{00} - b_{10}$. Then for each θ, the LSE and B-W estimates are obtained in the same method described in the above. The first 100 of the resulting root mean square errors are plotted in Fig. 4(a) and the box plot that corresponds to those for the entire 200 HMMs is shown in Fig. 4(b).

We see that on average the LSE estimates are closer to the actual parameter set used (by $0.443 - 0.418 = 0.025$ in the mean) and also significantly more stable (by $0.351 - 0.032 = 0.318$ in the variance), compared to the B-W estimates.

In addition, under the same condition, the errors are found also for $n = 50$ and 100. See Table III for the results. Again, we see that the length of the observation sequence does not significantly effect the error for this range of n, and the LSE estimator outperforms the B-W estimator for all n with respect to the mean and significantly to the variance.

As for the complexity reduction, for example, when $n = 100$, we found $|\widehat{\Omega}_n| \approx 110,800 < d\,2^{100} = |\Omega_n|$, where the reduction rate $d \approx 8.74\,10^{-26}$, on average over 200 observation sequences generated from 200 HMMs.

V. CONCLUSIONS AND FUTURE WORKS

A. Conclusions

Empirically, the LSE estimates are much closer to the actual parameters used and also more stable, compared to the ones typically obtained by the E-M algorithm when

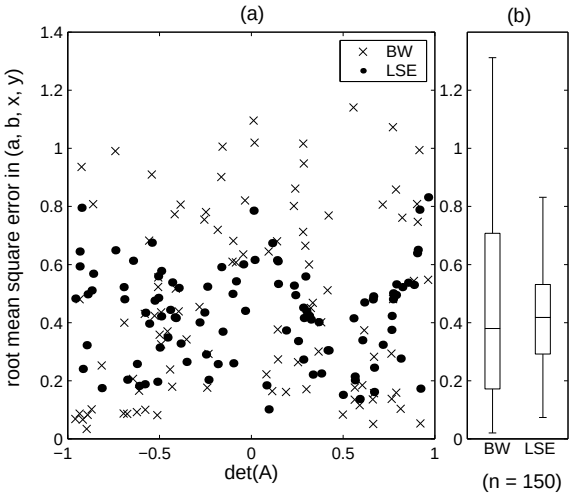

Fig. 4. (a) The root mean square error of the LSE (dot) and B-W (cross) estimates that correspond to the first 100 of the 200 HMMs generated are plotted against the determinant of the transition matrix A, and (b) the box plots of the errors that correspond to all 200 HMMs generated. ($m = 2$)

TABLE III

OVERALL MEAN, VARIANCE, AND MEDIAN OF ROOT MEAN SQUARE ERRORS

	mean		variance		median	
n	BW	LSE	BW	LSE	BW	LSE
50	0.537	0.467	0.339	0.030	0.485	0.456
100	0.464	0.429	0.347	0.035	0.402	0.430
150	0.443	0.418	0.351	0.032	0.380	0.418

the data size is small. Furthermore, with the algorithm we provided here, the computational complexity is polynomial in the length of the observation sequence, while it is still exponential with respect to the state space size.

Hence, we would recommend the LSE estimator in place of the B-W estimator for applications with limited data size, provided the state space size is small enough. Because of the advantages of the LSE estimator, we believe further study on this matter, mainly to further improve the computational complexity of the LSE, could be quite worthwhile in order to expand the range of suitable applications.

B. Future Works

While, in this paper, the observation state space size is restricted to 2, and the conditional probability for the observation sequence is kept constant; the current algorithm can be extended so that the observation state size is any integer in general, and various probability functions, such as a Poisson distribution function, can be used for the observation sequence in place of the observation matrix, as long as the observation state space is discrete.

As for the Markov chain state space size, the simulation in this paper is focused only on the size $m = 2$. While there are many applications that can utilize models with state space size 2 [4], [18], [21], more empirical study on the LSE of HMMs with larger state space size would be meaningful. However, at the moment, it is still relatively time consuming because of the higher computational complexity when $m > 2$.

There are two possible ways that we could consider in order to improve the current method of finding the LSE: one is to refine the way to implement the current algorithm by the choice of the programming scheme and/or of the type of a processor to implement it on; another is to use an algorithm other than the one introduced in this paper; e.g., particle filter methods.

VI. ACKNOWLEDGMENTS

We would like to thank my postdoctoral advisor, Dr. David Vere-Jones, Victoria University of Wellington, and NZIMA for making it possible to continue this research together with their valuable guidance.

REFERENCES

[1] D.M. Arnold, *Computing Information Rates of Finite-State Models with Application to Magnetic Recording*, Hartung Gorre, Wolfgang, 2003.

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

[3] L.E. Baum, T. Petrie, G. Soules, and N. Weiss, "A maximization technique occurring in the statistical analysis of probabilistic function of Markov chains," *Annuals of Mathematical Statistics*, vol. 41, pp. 164–171, Feb. 1970.

[4] I. Boesnach, J. Moldenhauer, C. Burgmer, T. Beth, V. Wank, and K. Bos, "Classification of phases in human motions by neural networks and hidden Markov models," *200f IEEE Conference on Cybernetics and Intelligent Systems*, vol. 2, pp. 976–981, Dec. 2004.

[5] I. Cohen, N. Sebe, A. Garg, L.S. Chen, and T.S. Huang, "Facial expression recognition from video sequences: temporal and static modeling," *Comput. Vision Image Understand.*, vol. 91, no. 1–2, pp. 160–187, Jul. 2003.

[6] R.I.A. Davis, B.C. Lovell, and T. Caelli, "Improved estimation of hidden Markov model parameters from multiple observation sequences," ed. R. Kasturi, D. Laurendeau, and C. Suen, *in Proc. 16th Int. Conf. Pattern Recognition*, vol. 2, Quebec, Canada, Aug. 2002, pp. 168–171.

[7] A.D. Dempster, N.M. Laird, and D.B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. R. Stat. Soc. Ser. B.*, vol. 39, no. 1, pp. 1–38, 1977.

[8] R. Fermandez and R. W. Picard, "Signal Processing for Recognition of Human Frustration," *in Proc. IEEE ICASSP 98*, Seattle, W.A. 1997.

[9] F. Jelinek, *Statistical Methods for Speech Recognition*, The MIT Press, Cambridge, MA, 1998.

[10] S.E. Levinson, L.R. Rabiner, and M.M. Sondhi, "An Introduction to the application of the theory of probabilistic functions of a Markov process to automatic speech recognition," *Bell System Tec. J.*, vol. 62, no. 4, pp.1035–1074, Apr. 1983.

[11] C.D. Manning and H. Schütze, *Foundations of Statistical Natural Language Processing*, MIT Press, 1999.

[12] T.K. Moon, "The expectation-maximization algorithm," *IEEE Signal Processing Magazine*, pp. 47–60, Nov. 1996.

[13] S. Müller, F. Wallhoff, F. Hülsken, and G. Rigoll, "Facial expression recognition using pseudo 3-D hidden Markov models," *in Proc. 16th Int. Conf. on Pattern Recognition*, ICPR 2002, Aug. 2002, pp. 11–15.

[14] V. Pavlovic, A. Garg, and S. Kasif, "A Bayesian framework for combining gene predictions," *Comput. Genomics*, Nov. 2000.

[15] J.S. Pedersen and J. Hein, "Gene finding with a hidden Markov model of genome structure and evolution," *Bioinformatics*, vol. 19, no. 2, pp. 219-227, 2003.

[16] L. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition*, Prentice Hall, Englewood Cliffs, NJ, 1993.

[17] R. Redner and H. Walker, "Mixture densities, maximum likelihood and the EM algorithm," *SIAM Review*, vol. 26, no. 2, pp. 195–239, Apr. 1984.

[18] B. Stenger, V. Ramesh, N. Paragios, F. Coetzee, and J.M. Buhmann, "Topology Free Hidden Markov Models: Application to Background Modeling," *in Proc. 8th IEEE Int. Conf. on Computer Vision*, vol. I, Vancouver, Canada, July 2001, pp. 294–301.

[19] G.E. Tusnády and I. Simon, "The HMMTOP transmembrane topology prediction server," *Bioinformatics*, vol. 17, no. 9, pp. 849-850, 2001.

[20] W. Zucchini and P. Guttorp, "A hidden Markov model for space-time precipitation," *Water Resources Research*, vol. 27, no. 8, pp. 1917-1923, 1991.

[21] Z. Xiong, R. Radhakrishnan, A. Divakaran, and T.S. Huang, "Audio-visual sports highlights extraction using coupled hidden Markov models," *Pattern Anal. Appl. J.*, vol. 8, no. 1–2, pp. 62–71, 2005.