Proceedings of the
46th IEEE Conference on Decision and Control
New Orleans, LA, USA, Dec. 12-14, 2007

A new approach for the identification of hidden Markov models

Bart Vanluyten, Jan C. Willems and Bart De Moor

Abstract— **In this paper, we consider the approximate identification problem for hidden Markov models, i.e. given a finite-valued output string generated by an unknown hidden Markov model, find an approximation of the underlying model. We propose a two-step procedure for the approximate identification problem. In the first step the underlying state sequence corresponding to the output sequence is estimated directly from the output data. In the second step the system matrices are calculated from the obtained state sequence and the given output sequence. In a simulation example the performance of our proposed method is compared with the performance of the classical Baum-Welch approach for identification of hidden Markov models.**

I. INTRODUCTION

Hidden Markov models (HMMs) were introduced in the literature in the late 1950s [1]. Twenty years later, HMMs started to be used in engineering applications, such as speech processing, image processing and bioinformatics. Despite the success in applications, many theoretical questions remain unanswered until now. For instance there does not exist a fundamental study of the exact identification problem, i.e. given an infinite output string generated by an unknown hidden Markov model of finite order, find the minimal underlying state dimension and calculate the exact system matrices of the underlying model. In this paper we consider the approximate realization problem for hidden Markov models, i.e. given a finite-valued output string of a hidden Markov model, find an approximation of the system matrices of the underlying model.

A popular approach for the identification of hidden Markov models is the Baum-Welch algorithm [4]. The Baum-Welch method iteratively increases the likelihood, where the likelihood is defined as the probability of the observed output string given the model. Despite the fact that Baum-Welch is widely used in practise, there are some drawbacks to the method. First of all, the solution is very sensitive to the initial choise of the system matrices. Moreover, the method garantees only that one will end up with a local maximum of the likelihood function, so we are not garanteed to find the global optimum. Finally, the Baum-Welch algorithm is very expensive from computational point of view.

In this paper we propose a technique that is inspired by the basic idea of subspace identification [5] for Gauss-Markov stochastic models. A typical subspace algorithm consists of two steps. In the first step the underlying state sequence is estimated directly from data. In the second step the system matrices are calculated from the state sequence and output sequence. Our proposed technique for identification of HMMs consists of the same two steps. In the first step the underlying state sequence is estimated directly from the output data and in the second step the system matrics are determined using this state sequence and the given output sequence. While subspace methods for Gauss-Markov systems need the singular value decomposition, our hidden Markov identification algorithm uses nonnegative matrix factorization techniques.

The paper is organized as follows. In section II, we introduce the notation for hidden Markov models. Section III shortly reviews the Baum-Welch approach for identification of HMMs. Section IV describes the nonnegative matrix factorization which is needed for our identification method. Section V contains the main results of this paper. It is first explained how the state sequence can be extracted from the given output sequence. Next we explain how the system matrices can be calculated from the estimated state sequence and the given output sequence. In Section VI, we apply our identification method on a simulation example and compare the results with the results of the Baum-Welch identification method.

The following notation is used. $\mathbb{R}_+$ is the set of nonnegative real numbers. If X is a matrix, then we mean with X_{ij} the i, j-th element of X, with $X_{i:}$, the i-th row of X and with $X_{:j}$, the j-th column of X. $X \geq 0$ denotes that the elements of X are nonnegative. With e we indicate a column vector with all elements equal to 1, i.e. $e := \begin{bmatrix} 1 & 1 & \ldots & 1 \end{bmatrix}^\top$.

II. HIDDEN MARKOV MODELS

Consider a stochastic process y defined on the time axis $\mathbb{N}$ taking values from a finite set $\mathbb{Y}$, called the output alphabet, with $|\mathbb{Y}|$ the cardinality of $\mathbb{Y}$. A *Mealy hidden Markov model* (HMM) of such stochastic process is defined as $(\mathbb{X}, \mathbb{Y}, \Pi, \pi(1))$, where

- $\mathbb{X}$ with $|\mathbb{X}| < \infty$ is the state alphabet, and $\mathbb{Y}$ is the output alphabet;
- $\pi(1)$ is a row vector in $\mathbb{R}_+^{|\mathbb{X}|}$ with $\pi(1)e = 1$;
- Π is a mapping from $\mathbb{Y}$ to $\mathbb{R}_+^{|\mathbb{X}|\times|\mathbb{X}|}$ with the matrix $\Pi_\mathbb{X} := \sum_{\mathrm{y}\in\mathbb{Y}} \Pi(\mathrm{y})$ such that $\Pi_\mathbb{X} e = e$.

One can think of an underlying state process x which generates the output process y. The process x takes values from the finite set $\mathbb{X}$ with cardinality $|\mathbb{X}|$. Without loss of generality, we take $\mathbb{X} = \{1, 2, \ldots, |\mathbb{X}|\}$. The element $\Pi_{ij}(\mathrm{y})$ is then equal to $P(x(t+1) = j, y(t) = \mathrm{y} | x(t) = i)$, the probability of going from state i to state j while producing

Bart Vanluyten, Jan C. Willems and Bart De Moor are with the Electrical Engineering Department, K.U.Leuven, Kasteelpark Arenberg 10, B-3001 Leuven, Belgium. Bart Vanluyten is a Research Assistant with the fund for Scientific Research-Flanders (FWO-Vlaanderen).

1-4244-1498-9/07/$25.00 ©2007 IEEE.

the output symbol y. The element $\pi_i(1)$ is equal to $P(x(1) = i)$, the initial distribution of the underlying state process.

Denote by $\mathbb{Y}^*$ the set of all finite strings with symbols from the set $\mathbb{Y}$ (including the empty string) and by $\mathbf{u} = u_1 u_2 \ldots u_{|\mathbf{u}|}$ a sequence from $\mathbb{Y}^*$, where $|\mathbf{u}|$ denotes the length of $\mathbf{u}$. Let $\mathcal{P} : \mathbb{Y}^* \mapsto [0, 1]$ be string probabilities, defined as $\mathcal{P}(\mathbf{u}) := P(y(1, 2, \ldots, |\mathbf{u}|) = u_1 u_2 \ldots u_{|\mathbf{u}|}) := P(y(1) = u_1, y(2) = u_2, \ldots, y(|\mathbf{u}|) = u_{|\mathbf{u}|})$. Of course, the string probabilities satisfy $\mathcal{P}(\phi) = 1$ and $\sum_{\mathrm{y}\in\mathbb{Y}} \mathcal{P}(\mathbf{u}\mathrm{y}) = \mathcal{P}(\mathbf{u})$[1]. Now it holds that

$$\mathcal{P}(\mathbf{u}) = \pi(1)\Pi(\mathbf{u})e,$$

where $\Pi(\mathbf{u}) := \Pi(u_1)\Pi(u_2) \ldots \Pi(u_{|\mathbf{u}|})$.

A *Moore hidden Markov model* is a more structured case of the Mealy hidden Markov model. In a Moore HMM, the generation of the next state and the generation of the output are independent.

If it holds for all $\mathbf{u} \in \mathbb{Y}^*$ that $\sum_{\mathrm{y}\in\mathbb{Y}} \mathcal{P}(\mathrm{y}\mathbf{u}) = \mathcal{P}(\mathbf{u})$ then the process is called *stationary*. Because of the fact that $\sum_{\mathrm{y}\in\mathbb{Y}} \mathcal{P}(\mathbf{u}\mathrm{y}) = \mathcal{P}(\mathbf{u})$ is due to consistency, we have for stationary processes that

$$\sum_{\mathrm{y}\in\mathbb{Y}} \mathcal{P}(\mathbf{u}\mathrm{y}) = \sum_{\mathrm{y}\in\mathbb{Y}} \mathcal{P}(\mathrm{y}\mathbf{u}).$$

A stationary hidden Markov model has the property that the state distribution is equal at every time instant $\pi(1) = \pi(2) = \ldots = \pi(t) = \pi$ where π equals the equilibrium state distribution, i.e.

$$\pi\Pi_{\mathbb{X}} = \pi.$$

In this paper we consider only stationary output strings and corresponding stationary models.

The (approximate) Mealy identification problem for hidden Markov models can be stated as

Given: an output string $u_1 u_2 \ldots u_T$ of length T of an unknown HMM with a finite number of states,

Find: an hidden Markov model $(\mathbb{X}, \mathbb{Y}, \Pi, \pi(1))$ of given order $|\mathbb{X}|$ such that the model is optimal (in a to be defined sense) with respect to the given output string.

III. BAUM-WELCH FOR IDENTIFICATION OF HMMS

In this section, we give the Baum-Welch algorithm for the Mealy identification problem. The Baum-Welch method is a maximum likelihood approach. This means that the system matrices $\Pi(\mathrm{y}), \mathrm{y} \in \mathbb{Y}$ and $\pi(1)$ are estimated such that the likelihood of the observed string is maximized, where the likelihood is defined as $P(y(1, 2, \ldots, T) = u_1 u_2 \ldots u_T \mid \lambda)$ where λ denotes the model. This maximum likelihood problem is solved using the expectation maximization approach. Expectation maximization is an iterative approach that starts with an initial guess for the model parameters λ and updates them iteratively such that the likelihood is nondecreasing in each step. It can be proven that the Baum-Welch update formulas end up in a local maximum (or a saddle point) of the likelihood surface.

Before being able to give the Baum-Welch update formulas, we need to define the forward variables $\alpha(t) \in \mathbb{R}^{1\times|\mathbb{X}|}$ and the backward variables $\beta(t) \in \mathbb{R}^{|\mathbb{X}|\times 1}$ as

$$\begin{aligned} \alpha_i(t) &:= P(x(t+1) = i, y(1, \ldots, t) = u_1 \ldots u_t) \\ \beta_i(t) &:= P(y(t, \ldots, T) = u_t \ldots u_T \mid x(t) = i). \end{aligned}$$

The forward variables can be calculated inductively as

$$\begin{aligned} \alpha(1) &= \pi\Pi(u_1), \\ \alpha(t+1) &= \alpha(t)\Pi(u_{t+1}), \end{aligned}$$

while the backward variables can be calculated as

$$\begin{aligned} \beta(T) &= \Pi(u_T)e, \\ \beta(t) &= \Pi(u_t)\beta(t+1). \end{aligned}$$

Next, we need the variables $\gamma_i(t)$ and $\xi_{ij}(t)$ defined as

$$\begin{aligned} \gamma_i(t) &:= P(x(t) = i | y(1, \ldots, T) = u_1 \ldots u_T), \\ \xi_{ij}(t) &:= P(x(t) = i, x(t+1) = j | y(1, \ldots, T) = u_1 \ldots u_T). \end{aligned}$$

These variables are related with the forward and backward variables through

$$\begin{aligned} \gamma_i(t) &= \frac{\alpha_i(t-1)\beta_i(t)}{\alpha(t-1)\beta(t)}, \\ \xi_{ij}(t) &= \frac{\gamma_i(t)\beta_j(t+1)\Pi(u_t)_{ij}}{\beta_i(t)}. \end{aligned}$$

Given an output sequence and an initial guess of the model, the Baum-Welch procedure calculates the variables $\gamma_i(t)$ and $\xi_{ij}(t)$ and then updates the model as

$$\begin{aligned} \pi_i(1) &= \gamma_i(1), \\ \Pi_{ij}(\mathrm{y}) &= \frac{\sum_{t=1}^{T-1} \delta_{u_t,\mathrm{y}}\xi_{ij}(t)}{\sum_{t=1}^{T-1} \gamma_i(t)}, \end{aligned}$$

where $\delta_{i,j}$ is the Kronecker delta, i.e. $\delta_{i,j} = 0$ if $i \neq j$ and $\delta_{i,j} = 1$ if $i = j$. Next, the variables $\gamma_i(t)$ and $\xi_{ij}(t)$ are recalculated and the model parameters are updated again. This procedure is proceeded until convergence.

IV. NONNEGATIVE MATRIX FACTORIZATION

In this section, we introduce the nonnegative matrix factorization problem as we will need this factorization in our identification approach. The nonnegative matrix factorization problem can be stated as follows: given a matrix $M \in \mathbb{R}_+^{m_1\times m_2}$, find a decomposition $M = VH$ with $V \in \mathbb{R}_+^{m_1\times a}$ and $H \in \mathbb{R}_+^{a\times m_2}$, and with a as small as possible. The minimal inner dimension a for which a decomposition exists is called the positive rank (p–rank) of M. It is clear that $0 \leq \mathrm{rank}(V) \leq \text{p–rank}(V) \leq \min\{m_1, m_2\}$. There does not exist a practical useful algorithm to find the positive rank of a general positive matrix. Furthermore, no algorithm is known to compute a nonnegative matrix factorization. Recently, the approximate nonnegative matrix factorization problem was introduced in [2]. The idea is that one choses the inner dimension a and looks for matrices V and H such that VH approximates P optimally in a certain distance measure.

[1] With $\mathbf{u}\mathrm{y}$, we mean the concatenation of the string $\mathbf{u}$ with the symbol y. Concatenation of two strings is defined analogously.

The Kullback-Leibler divergence is a popular such measure. The Kullback-Leibler divergence between two nonnegative matrices of the same size is defined as

$$D(A||B) = \sum_{ij}(A_{ij}\log\frac{A_{ij}}{B_{ij}} - A_{ij} + B_{ij}).$$

The approximate nonnegative matrix factorization problem can now be stated as

Problem 1: Given $M \in \mathbb{R}_+^{m_1\times m_2}$ and given a, minimize $D(M||VH)$ with respect to V (of size $m_1 \times a$) and H (of size $a \times m_2$), subject to the constraints $V, H \geq 0$.

Lee and Sueng propose iterative update formulas to solve problem 1. The convergence of these update formulas leads to the following theorem.

Theorem 1: [2], [3] The divergence $D(M||VH)$ is nonincreasing under the update rules

$$H_{i,l} \leftarrow H_{i,l}\frac{\sum_\mu V_{\mu i}\frac{M_{\mu l}}{(VH)_{\mu l}}}{\sum_\mu V_{\mu i}}, \quad V_{k,i} \leftarrow V_{k,i}\frac{\sum_\nu H_{i\nu}\frac{M_{k\nu}}{(VH)_{k\nu}}}{\sum_\nu H_{i\nu}}$$

The matrices V and H are invariant under these updates if and only if V and H are in a stationary point of the divergence $D(M||VH)$.

As the initial values for V and H have to be chosen nonnegative, the obtained matrices V and H are nonnegative.

V. SUBSPACE APROACH TO IDENTIFICATION OF HMMS

In this section we explain our approach to the identification problem of hidden Markov models. The approach consists of two steps. In the first step the underlying state process is estimated directly from the given output string. In the second step the system matrices are calculated from the obtained state sequence and the given output sequence.

A. Estimating the state sequence

In this section we explain how the state sequence can be extracted directly from output data. We first describe a method to find the underlying state sequence under the assumption that we are given a certain matrix containing string probabilities of the underlying HMM, and a nonnegative decomposition of this matrix. In a second step, we explain how this matrix with string probabilities can be estimated from data. Moreover, we show that the nonnegative decomposition of this matrix can be found using nonnegative matrix factorization techniques. By combining both steps we have a method to find the estimated state sequence directly from output data.

So suppose first that the matrix $M(i_1, i_2)$ of the underlying HMM $(\mathbb{X}, \mathbb{Y}, \Pi, \pi(1))$ is given, where

$$(M(i_1, i_2))_{kl} = \mathcal{P}(\mathbf{u}_k\mathbf{v}_l),$$

with $\mathbf{u}_k\mathbf{v}_l$ the concatenation of $\mathbf{u}_k$ and $\mathbf{v}_l$ and $\mathcal{U} := (\mathbf{u}_i, i = 1, 2, \ldots |\mathbb{Y}|^{i_1})$ and $\mathcal{V} := (\mathbf{v}_i, i = 1, 2, \ldots |\mathbb{Y}|^{i_2})$ are the lexicographical orderings of the string of length i_1 and i_2 respectively. Suppose also that we are given a decomposition of the matrix $M(i_1, i_2)$ in the form

$$M(i_1, i_2) = VH, \tag{1}$$

with

$$\begin{aligned} V &= \begin{bmatrix} \pi(1)\Pi(\mathbf{u}_1) \\ \pi(1)\Pi(\mathbf{u}_2) \\ \vdots \\ \pi(1)\Pi(\mathbf{u}_{|\mathbb{Y}|^{i_1}}) \end{bmatrix}, \\ H &= \begin{bmatrix} \Pi(\mathbf{v}_1)e & \Pi(\mathbf{v}_2)e & \ldots & \Pi(\mathbf{v}_{|\mathbb{Y}|^{i_2}})e \end{bmatrix}. \end{aligned}$$

The elements of V are then equal to

$$V_{k,i} = P(y(1, 2, \ldots, i_1) = \mathbf{u}_k, x(i_1 + 1) = i),$$

while the elements of H are equal to

$$H_{i,l} = P(y(i_1 + 1, i_1 + 2, \ldots, i_1 + i_2) = \mathbf{v}_l | x(i_1 + 1) = i).$$

Define $\tilde{V}$ as

$$\tilde{V} = (\text{diag}(Ve))^{-1}V,$$

such that

$$\tilde{V}_{k,i} = P(x(i_1 + 1) = i | y(1, 2, \ldots, i_1) = \mathbf{u}_k),$$

and $\hat{V}$ as

$$\hat{V}_{k,i} = \begin{cases} 1 & i = \text{argmax}_\lambda \tilde{V}_{k,\lambda}, \\ 0 & \text{else.} \end{cases}$$

The estimated state sequence matrix $\hat{X}_{i_1} \in \{0, 1\}^{(T-i_1)\times|\mathbb{X}|}$ is then defined as

$$\hat{X}_{i_1} = \begin{bmatrix} \hat{x}(i_1 + 1) \\ \hat{x}(i_1 + 2) \\ \vdots \\ \hat{x}(T) \end{bmatrix},$$

where

$$\hat{x}(t) = \hat{V}_{k,:},$$

with k the position of the string $u_{t-i_1} \ldots u_{t-1}$ in the lexicographical ordering of the strings of length i_1. Notice that the state estimate $\hat{x}(t)$ is a row vector of size $|\mathbb{X}|$ with all elements equal to zero except for the element at position i which is equal to 1, where i is the most likely state estimate for $x(t)$ given the past i_1 observations of $y(t - i_1) \ldots y(t - 1)$. It will become clear in the next section that we also need the most likely state estimate for $x(t + 1)$ based on the same observations. Denote by $\hat{x}^+(t + 1)$ the row vector of size $|\mathbb{X}|$ that contains only the element zero except for the element at position i which is equal to 1, where i is the most likely state estimate at time instant $t + 1$ given the observations of $y(t - i_1) \ldots y(t - 1)$. All these state estimates are stacked in the matrix $\hat{X}^+_{i_1+1}$ defined as

$$\hat{X}^+_{i_1+1} = \begin{bmatrix} \hat{x}^+(i_1 + 2) \\ \hat{x}^+(i_1 + 3) \\ \vdots \\ \hat{x}^+(T + 1) \end{bmatrix}.$$

To be able to calculate the vectors $\hat{x}^+(t + 1)$, the matrix $M(i_1 + 1, i_2)$ needs to be given as well as a decomposition of this matrix in the form

$$M(i_1 + 1, i_2) = WH, \tag{2}$$

where H is the same as before and

$$W = \begin{bmatrix} \pi(1)\Pi(\mathbf{u}_1\mathrm{y}_1) \\ \pi(1)\Pi(\mathbf{u}_1\mathrm{y}_2) \\ \vdots \\ \pi(1)\Pi(\mathbf{u}_{|\mathbb{Y}|^{i_1}}\mathrm{y}_{|\mathbb{Y}|}) \end{bmatrix}.$$

Indeed, $\hat{x}^+(t+1)$ can now be calculated as

$$\begin{aligned} \bar{W} &= \begin{bmatrix} e^\top & & & \\ & e^\top & & \\ & & \ddots & \\ & & & e^\top \end{bmatrix} W, \\ \tilde{W} &= (\mathrm{diag}(\bar{W}e))^{-1}\bar{W}, \\ \hat{W}_{k,i} &= \begin{cases} 1 & i = \mathrm{argmax}_\lambda \tilde{W}_{k,\lambda}, \\ 0 & \text{else}, \end{cases} \\ \hat{x}^+(t+1) &= \hat{W}_{k,:} \end{aligned}$$

where k is the position of the string $u_{t-i_1}\dots u_{t-1}$ in the lexicographical ordering of the strings of length i_1 and $e = \begin{bmatrix} 1 & 1 & \dots & 1 \end{bmatrix}^\top$ of size $|\mathbb{Y}|$.

So far we have described a method to find an estimated state sequence corresponding to the given output sequence. The method supposes that for certain choises of i_1 and i_2 the matrices $M(i_1, i_2)$ and $M(i_1+1, i_2)$ containing string probabilities are given. Moreover it is supposed that the nonnegative decompositions (1) and (2) of these matrices are given. We will now show that the matrices $M(i_1, i_2)$ and $M(i_1+1, i_2)$ can be estimated from data and that the nonnegative decomposition of these matrices can be approximated using the nonnegative matrix factorization technique. As a result the complete procedure to find the state sequence works directly from the given output data.

The matrices $M(i_1, i_2)$ and $M(i_1+1, i_2)$ contain string probabilities of strings of length i_1+i_2 and i_1+i_2+1. By assuming ergodicity, it is possible to estimate these string probabilities directly from the output sequence. The probability of a certain output string can be estimated as the number of times that the string occurs in the output sequence, divided by the maximum number of times that it could have occured. The estimated matrices are denoted by $M^{\mathrm{est}}(i_1, i_2)$ and $M^{\mathrm{est}}(i_1+1, i_2)$.

The nonnegative decomposition of the matrices $M^{\mathrm{est}}(i_1, i_2)$ and $M^{\mathrm{est}}(i_1+1, i_2)$ can be obtained by applying the nonnegative matrix factorization (Theorem 1) to find an approximate decomposition of the form

$$\begin{bmatrix} M^{\mathrm{est}}(i_1, i_2) \\ M^{\mathrm{est}}(i_1+1, i_2) \end{bmatrix} \simeq \begin{bmatrix} V^{\mathrm{est}} \\ W^{\mathrm{est}} \end{bmatrix} H^{\mathrm{est}}.$$

As there does not exist a practical useful procedure to determine the minimum inner dimension for which such a decomposition exists, we need to chose the inner dimension. Notice that the choise of the inner dimension is important as it will be the state dimension of the obtained model.

B. Calculating the system matrices

In this section we explain how the system matrices can be obtained from the estimated state sequences and the output sequence.

Theoretically it holds for $t = i_1+1, \dots, T$ and for $\mathrm{y} \in \mathbb{Y}$ that

$$\begin{aligned} &P(x(t+1), y(t) = \mathrm{y} \mid y(t-i_1, \dots, t-1) = u_{t-i_1}\dots u_{t-1}) = \\ &\qquad P\left(x(t) \mid y(t-i_1, \dots, t-1) = u_{t-i_1}\dots u_{t-1}\right)\Pi(\mathrm{y}), \end{aligned} \quad (3)$$

where

$$\begin{aligned} &P(x(t+1), y(t) = \mathrm{y} \mid y(t-i_1, \dots, t-1) = u_{t-i_1}\dots u_{t-1}), \\ &P\left(x(t) \mid y(t-i_1, \dots, t-1) = u_{t-i_1}\dots u_{t-1}\right) \end{aligned}$$

are row vectors with length equal to $|\mathbb{X}|$.

Now $P\left(x(t) \mid y(t-i_1, \dots, t-1) = u_{t-i_1}\dots u_{t-1}\right)$ is replaced by $\hat{x}(t)$. This means that the conditional distibution of $x(t)$ is replaced by a distribution where the most likely state estimate for $x(t)$ has a probability 1 and all other states have probability 0. On the other hand $P(x(t+1), y(t) = \mathrm{y} \mid y(t-i_1, \dots, t-1) = u_{t-i_1}\dots u_{t-1})$ is approximated by $\hat{x}^{\mathrm{y}}(t+1)$ defined as

$$\hat{x}^{\mathrm{y}}(t+1) = \begin{cases} \hat{x}^+(t+1) & u_t = \mathrm{y}, \\ \begin{bmatrix} 0 & 0 & \dots & 0 \end{bmatrix} & \text{else}. \end{cases}$$

This means that the conditional joint distribution of the state $x(t+1)$ and the output $y(t)$ is replaced by a joint distibution where the combination of the most likely state estimate for $x(t)$ and the observed output u_t has probability 1, while all other combinations of state symbols and output symbols have probability 0.

By defining matrices $\hat{X}^{\mathrm{y}}_{i_1+1}$, $\forall \mathrm{y} \in \mathbb{Y}$

$$\hat{X}^{\mathrm{y}}_{i_1+1} = \begin{bmatrix} \hat{x}^{\mathrm{y}}(i_1+2) \\ \hat{x}^{\mathrm{y}}(i_1+3) \\ \vdots \\ \hat{x}^{\mathrm{y}}(T+1) \end{bmatrix},$$

equation (3) can be written as

$$\begin{aligned} &\begin{bmatrix} \hat{X}^{\mathrm{y}_1}_{i_1+1} & \hat{X}^{\mathrm{y}_2}_{i_1+1} & \dots & \hat{X}^{\mathrm{y}_{|\mathbb{Y}|}}_{i_1+1} \end{bmatrix} = \\ &\qquad \hat{X}_{i_1} \begin{bmatrix} \hat{\Pi}(\mathrm{y}_1) & \hat{\Pi}(\mathrm{y}_2) & \dots & \hat{\Pi}(\mathrm{y}_{|\mathbb{Y}|}) \end{bmatrix}, \end{aligned} \quad (4)$$

where the true system matrices are replaced by $\hat{\Pi}(\mathrm{y})$, $\mathrm{y} \in \mathbb{Y}$ as the true state distribution was replaced by most likely state estimates. By solving (4) for $\hat{\Pi}(\mathrm{y})$, $\mathrm{y} \in \mathbb{Y}$ in least squares sense, we find

$$\begin{aligned} &\begin{bmatrix} \hat{\Pi}(\mathrm{y}_1) & \hat{\Pi}(\mathrm{y}_2) & \dots & \hat{\Pi}(\mathrm{y}_{|\mathbb{Y}|}) \end{bmatrix} = \\ &\qquad (\hat{X}_{i_1})^\dagger \begin{bmatrix} \hat{X}^{\mathrm{y}_1}_{i_1+1} & \hat{X}^{\mathrm{y}_2}_{i_1+1} & \dots & \hat{X}^{\mathrm{y}_{|\mathbb{Y}|}}_{i_1+1} \end{bmatrix} \end{aligned}$$

where $(\hat{X}_{i_1})^\dagger = (\mathrm{diag}(e^\top \hat{X}_{i_1}))^{-1}(\hat{X}_{i_1})^\top$ is the Moore-Penrose pseudo-inverse of $\hat{X}_{i_1}$. It is easy to see that the matrices $\hat{\Pi}(\mathrm{y})$ obtained in this way are elementwise nonnegative. In addition it holds that $(\sum_{\mathrm{y}} \hat{\Pi}(\mathrm{y}))e = e$.

The initial state distribution $\hat{\pi}(1)$ is taken equal to the normalised left eigenvector of $\sum_{\mathrm{y}} \hat{\Pi}(\mathrm{y})$ corresponding to the eigenvalue 1.

VI. SIMULATION EXAMPLE

In this simulation example, we are given an output string $u_1 \ldots u_{1000}$ generated with $\lambda_{\rm true} = (\{1,2\},\{1,2\},\Pi_{\rm true},\pi_{\rm true}(1))$ where

$$\begin{aligned}
\Pi_{\rm true}(1) &= \begin{bmatrix} 0.20 & 0.40 \\ 0.00 & 0.20 \end{bmatrix}, \\
\Pi_{\rm true}(2) &= \begin{bmatrix} 0.10 & 0.30 \\ 0.80 & 0.00 \end{bmatrix}, \\
\pi_{\rm true} &= \begin{bmatrix} 0.53 & 0.47 \end{bmatrix}.
\end{aligned}$$

In fact this model is unknown, but we give it here the check the performance of our algorithm. We now use our proposed method as well as the Baum-Welch algorithm to find a hidden Markov model corresponding to the given output sequence. The model found with our method with $i_1 = i_2 = 3$ is given by $\lambda_{\rm SS} = (\{1,2\},\{1,2\},\Pi_{\rm SS},\pi_{\rm SS}(1))$ with

$$\begin{aligned}
\Pi_{\rm SS}(1) &= \begin{bmatrix} 0.0699 & 0.2574 \\ 0.5651 & 0.0000 \end{bmatrix}, \\
\Pi_{\rm SS}(2) &= \begin{bmatrix} 0.1342 & 0.5386 \\ 0.4349 & 0.0000 \end{bmatrix}, \\
\pi_{\rm SS} &= \begin{bmatrix} 0.5568 & 0.4432 \end{bmatrix},
\end{aligned}$$

while the model found with Baum-Welch (after convergence) is given by $\lambda_{\rm BW} = (\{1,2\},\{1,2\},\Pi_{\rm BW},\pi_{\rm BW}(1))$ where

$$\begin{aligned}
\Pi_{\rm BW}(1) &= \begin{bmatrix} 0.0736 & 0.0986 \\ 0.5311 & 0.1415 \end{bmatrix}, \\
\Pi_{\rm BW}(2) &= \begin{bmatrix} 0.0751 & 0.7526 \\ 0.2424 & 0.0850 \end{bmatrix}, \\
\pi_{\rm BW} &= \begin{bmatrix} 0 & 1 \end{bmatrix}.
\end{aligned}$$

The check the quality of both estimated models, we need a distance measure between the estimated model and the true model. A popular distance measure between $\lambda_{\rm true}$ and its approximation $\lambda_{\rm approx}$ is the Kullback-Leibler divergence defined as

$$D(\lambda_{\rm true}||\lambda_{\rm approx}) = \sum_{{\rm y}\in\mathbb{Y}^*} \mathcal{P}({\rm y}|\lambda_{\rm true}) \log \frac{\mathcal{P}({\rm y}|\lambda_{\rm true})}{\mathcal{P}({\rm y}|\lambda_{\rm approx})},$$

where $\mathcal{P}({\rm y}|\lambda)$ denotes the string probability of the string y for the model λ. To be able to calculate this distance in practice, we need to take only a finite selection of the strings instead of all strings of finite length.

If we take the Kullback-Leibler divergence for string probabilities of strings up to length 8, we find in this example

$$\begin{aligned}
D(\lambda_{\rm true}||\lambda_{\rm SS}) &= 0.3876, \\
D(\lambda_{\rm true}||\lambda_{\rm BW}) &= 1.8955,
\end{aligned}$$

from which we conclude that our method performs much better than the popular Baum-Welch approach, but still leaves room for further improvement.

VII. CONCLUSION

In this paper we considered the approximate identification problem for hidden Markov models, i.e. given a finite-valued output string generated by an unknown hidden Markov model, find an approximation for the underlying model. We proposed an identification method consisting of two steps. In the first step a state sequence corresponding to the given output sequence is calculated directly from data. In the second step the system matrices are estimated using the obtained state sequence and the given output sequence. We applied our method to a simulation example and compared the results with the classical Baum-Welch identification approach.

ACKNOWLEDGEMENTS

Bart Vanluyten is a research assistant with the Fund for Scientific Research Flanders (FWO-Vlaanderen). Jan Willems is a professor and Bart De Moor a full professor with the Katholieke Universiteit Leuven. The SISTA research program is supported by: Research Council KUL: GOA AMBioRICS, CoE EF/05/006 Optimization in Engineering, several PhD/postdoc and fellow grants; Flemish Government: FWO: PhD/postdoc grants, projects, G.0407.02 (support vector machines), G.0197.02 (power islands), G.0141.03 (Identification and cryptography), G.0491.03 (control for intensive care glycemia), G.0120.03 (QIT), G.0452.04 (new quantum algorithms), G.0499.04 (Statistics), G.0211.05 (Nonlinear), G.0226.06 (cooperative systems and optimization), G.0321.06 (Tensors), G.0302.07 (SVM/Kernel, research communities (ICCoS, ANMMM, MLDM); IWT: PhD Grants, McKnow-E, Eureka-Flite2; Belgian Federal Science Policy Office: IUAP P6/04 (Dynamical systems, control and optimization, 2007-2011); EU: ERNSI.

REFERENCES

[1] D. Blackwell and L. Koopmans, On the identifiability problem for functions of finite Markov chains, *Annals of Mathematical Statistics*, 28, 1011-1015, 1957.

[2] D. Lee and S. Sueng, Learning the parts of object by nonnegative matrix factorization, *Nature*, vol. 401, 1999, pp 788-791.

[3] D. Lee and S. Sueng, Algorithms for nonnegative matrix factorization, *Advances in Neural Information Processing Systems*, vol. 13, 2001, pp 556-562.

[4] L. Rabiner and B. Juang, An introduction to hidden Markov models, *IEEE ASSP Magazine*, vol. 3, 1986, pp 4-16.

[5] P. Van Overschee and B. De Moor, *Subspace Identification for Linear Systems: Theory, Implementation, Applications*, Kluwer Academic Publishers, 1996, 254 p.

[6] D. Ho and P. van Dooren, Nonnegative matrix factorizations with fixed row and column sums, *submitted for publication*, 2006.