Model Learning for Switching Linear Systems with Autonomous Mode Transitions

Lars Blackmore and Stephanie Gil and Seung Chung and Brian Williams

***Abstract*— We present a novel method for model learning in hybrid discrete-continuous systems. The approach uses approximate Expectation-Maximization to learn the Maximum-Likelihood parameters of a switching linear system. The approach extends previous work by 1) considering *autonomous* mode transitions, where the discrete transitions are conditioned on the continuous state, and 2) learning the effects of control inputs on the system. We evaluate the approach in simulation.**

I. Introduction

Stochastic hybrid discrete-continuous models have been used to represent a large number of physical and biological systems[1], [2], [3]. A great deal of recent work has proposed methods for control and state estimation with such systems, for example [4], [5], [6], [2], [7], [8]. These approaches typically rely on accurate models of the hybrid system, however specifying these models manually is often challenging. We must therefore determine hybrid system models from observed data. Since the discrete and continuous dynamics of such models are coupled, partially observable and stochastic, this is a very challenging problem. One approach in dealing with hybrid-discrete systems is to first map continuous data to discrete data to use for the purposes of learning as in [9], [10]; alternatively, our approach is to deal directly with continuous observations.

In the case of linear systems with only continuous dynamics, previous work ([11], [12]) developed methods to determine the Maximum-Likelihood (ML) model parameters using an Expectation-Maximization(EM)[13] approach. This approach guarantees convergence to a local maximum of the likelihood function. More recent work extended this approach to Jump Markov Linear Systems(JMLS)[14], [15]. These systems have linear continuous dynamics and Markovian discrete transitions; in this case the discrete dynamics are independent of the continuous state. [14] used an 'approximate' EM approach to learn the parameters of JMLS. Due to the approximation introduced in the Expectation Step (E-Step), this approach does *not* have guaranteed convergence. [15] used an approach inspired by EM to guarantee convergence; this approach iterates between calculating the maximum likelihood discrete model and the maximum likelihood continuous model. This is analogous to 'hard' EM, where instead of using distributions over the unknown variables, only the most likely values are used.

Lars Blackmore is a PhD student, MIT. larsb@mit.edu
Stephanie Gil is an SM student, MIT. sgil@mit.edu
Seung Chung is a PhD student, MIT. chung@mit.edu
Brian Williams is an associate professor, MIT. williams@mit.edu

In this paper we present a new approach to hybrid model learning that extends this work in three ways. First, we consider systems where transitions in the discrete state *do* depend on the continuous state; we call these *autonomous* mode transitions. Second, we extend the work of [14], [15], [16] to learn explicitly the dependence of control inputs on the system dynamics. Finally, we consider 'soft' EM, where a distribution over the hidden variables is used, rather than just the most likely values.

In Section II we define a Linear Probabilistic Hybrid Automaton and state the hybrid model learning problem. In Section III we review general Expectation-Maximization [13], and in Section IV we give an overview of the new hybrid model learning approach before describing its key components in Sections V and VI. Finally, in Section VII we provide simulation results.

II. Problem Statement

Prior work defined the Probabilistic Hybrid Automaton (PHA)[2]. In this paper we are concerned with a restricted type of Probabilistic Hybrid Automaton, which we refer to as a Linear Probabilistic Hybrid Automaton (LPHA). The continuous dynamics for a LPHA are given by:

$$\begin{aligned} \mathbf{x}_{t+1} &= A(\mathbf{m}_t)\mathbf{x}_t + B(\mathbf{m}_t)\mathbf{u}_t + \omega_t \\ \mathbf{y}_{c,t+1} &= C(\mathbf{m}_t)\mathbf{x}_{t+1} + D(\mathbf{m}_t)\mathbf{u}_t + \nu_t, \end{aligned} \tag{1}$$

where $\mathbf{x} \in \Re^{n_x}$ is the continuous state and $\mathbf{m} \in \mathcal{X}_d$ is the discrete state, or *mode*. We use the subscript notation $\mathbf{x}_t$ to denote the value of variable $\mathbf{x}$ at time t, and use $\mathbf{x}_{t_1}^{t_2}$ to denote the sequence $\mathbf{x}_{t_1}, \ldots, \mathbf{x}_{t_2}$. The variables ω and ν are process and observation noise respectively, which we restrict to be zero-mean, Gaussian white noise with covariance $Q(\mathbf{m}_t)$ and $R(\mathbf{m}_t)$, respectively. The initial distribution $p(\mathbf{x}_0, \mathbf{m}_0)$ is a sum-of-Gaussians, where $p(\mathbf{x}_0|\mathbf{m}_0)$ is a Gaussian with mean $\mu(\mathbf{m}_0)$ and covariance $V(\mathbf{m}_0)$.

The evolution of the discrete state is described by a number of *guard conditions* $c_i \in \mathcal{G}$. Each guard condition has an associated *guard region* $\mathcal{C}_i \subset \Re^{n_x}$ and a *transition probability matrix* T_i such that $T_i(m,n) = p(\mathbf{m}_{t+1} = m|\mathbf{m}_t = n, \mathbf{x}_t \in \mathcal{C}_i)$. The guard regions form a partition of the space $\Re^{n_x}$. For clarity we consider guard regions over the continuous state only, however dealing with guards defined over control inputs or output variables is straightforward since these are fully observable.

The *continuous model parameters* $\theta_c(\mathbf{m})$ are defined for each mode $\mathbf{m} \in \mathcal{X}_d$ of a LPHA as the set $\langle A(\mathbf{m}), B(\mathbf{m}), C(\mathbf{m}), D(\mathbf{m}), Q(\mathbf{m}), R(\mathbf{m}), V(\mathbf{m}), \mu(\mathbf{m})\rangle$. The *discrete model parameters* $\theta_d(\mathbf{m})$ are defined for each guard condition $c_i \in \mathcal{G}$ of a LPHA as the transition probability matrix T_i. The *hybrid model parameters* for a LPHA $\mathcal{P}$ are defined as the continuous model parameters $\theta_c(\mathbf{m})$ for every mode $\mathbf{m} \in \mathcal{X}_d$ and the discrete model parameters $\theta_d(\mathbf{m})$ for each guard condition $c_i \in \mathcal{G}$. The hybrid model learning problem is then:

> *Given a finite sequence of observations $\mathbf{y}_1^{T+1}$, and a finite sequence of control inputs $\mathbf{u}_0^T$, determine the hybrid model parameters θ that maximize the likelihood $p(\mathbf{y}_1^{T+1}|\theta)$.*

Note that the hybrid model parameters do not include the guard regions; learning of guard regions is a topic for future research.

III. Review of Expectation-Maximization

In this section we review the general Expectation-Maximization (EM) algorithm[13]. More thorough reviews have been carried out by [17] and [18].

EM is an iterative approach for finding a local maximum to a function, which is often the Maximum Likelihood probability of some matrix of observations $\mathbf{Y}$ given a vector of parameters θ, i.e. $p(\mathbf{Y}|\theta)$. The maximizer of this value is also the maximizer of the function:

$$g(\theta) = \log p(\mathbf{Y}|\theta). \tag{2}$$

EM addresses the situation when the likelihood value $p(\mathbf{Y}|\theta)$ is not readily evaluated. An example of this is when there are *hidden* variables so that only the distribution $p(\mathbf{Y}, \mathbf{X}|\theta)$ is known explicitly. In this case we express the likelihood using a marginalization over the hidden variables:

$$g(\theta) = \log \int_{\mathbf{X}} p(\mathbf{Y}, \mathbf{X}|\theta) d\mathbf{X}. \tag{3}$$

In this case $\mathbf{X}$ takes continuous values; in the case of discrete-valued $\mathbf{X}$, the integral is replaced by a summation. The key difficulty in maximizing $g(\theta)$ is that it involves a logarithm over an integral (or a large summation), which is difficult to deal with in many cases[18]. It is, however, possible to create a lower bound to $g(\theta)$ that instead involves an integral or sum of logarithms, which *is* tractable. In EM, Jensen's inequality is used to give the lower bound:

$$\begin{aligned} g(\theta) &= \log \int_{\mathbf{X}} p(\mathbf{Y}, \mathbf{X}|\theta) d\mathbf{X} \\ &\geq \int_{\mathbf{X}} p(\mathbf{X}|\mathbf{Y}, \theta^k) \log \frac{p(\mathbf{Y}, \mathbf{X}|\theta)}{p(\mathbf{X}|\mathbf{Y}, \theta^k)} d\mathbf{X} := h(\theta|\theta^k), \end{aligned} \tag{4}$$

where θ^k is a guess for the value for the parameters θ at iteration k of the EM algorithm. This bound can be written in terms of an expectation over the hidden state X, and an 'entropy' term denoted $\mathcal{H}$ that does not depend on θ.

$$\begin{aligned} h(\theta|\theta^k) &= \int_{\mathbf{X}} p(\mathbf{X}|\mathbf{Y}, \theta^k) \log \frac{p(\mathbf{Y}, \mathbf{X}|\theta)}{p(\mathbf{X}|\mathbf{Y}, \theta^k)} d\mathbf{X} \\ &= \int_{\mathbf{X}} p(\mathbf{X}|\mathbf{Y}, \theta^k) \log p(\mathbf{Y}, \mathbf{X}|\theta) d\mathbf{X} \\ &\quad - \int_{\mathbf{X}} p(\mathbf{X}|\mathbf{Y}, \theta^k) \log p(\mathbf{X}|\mathbf{Y}, \theta^k) d\mathbf{X} \\ &= E_{\mathbf{X}|\mathbf{Y}, \theta^k} \left[\log p(\mathbf{Y}, \mathbf{X}|\theta) \right] + \mathcal{H}. \end{aligned} \tag{5}$$

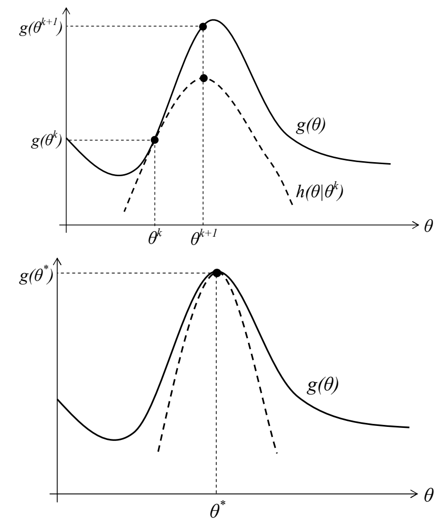

Fig. 1. Expectation-Maximization finds a local maximum of the function $g(\theta)$ through iterative lower bound maximization. **Top:** At each iteration a lower bound $h(\theta|\theta^k)$ is constructed, which touches $g(\theta)$ at the current guess θ^k. Any value of θ^{k+1} that increases $h(\theta|\theta^k)$ must also increase $g(\theta)$, as shown. **Bottom:** EM converges when θ^k is at a local maximum of $h(\theta|\theta^k)$ denoted θ^*, which, under smoothness assumptions, is guaranteed to be a local maximum of $g(\theta)$.

The lower bound $h(\theta|\theta^k)$ is the tightest possible bound[17], and in particular at the current guess θ^k, the bound touches the objective function $g(\theta)$.

The key idea behind the EM algorithm is to construct the tractable bound $h(\theta|\theta^k)$ given the current guess θ^k and then to maximize the bound with respect to θ to give θ^{k+1}. The bound $h(\theta|\theta^k)$ involves an integral over logarithms rather than the logarithm of an integral, which makes it tractable for optimization in many cases. Because the bound $h(\theta|\theta^k)$ touches the objective function $g(\theta)$ at the current guess, maximizing the bound with regard to θ guarantees finding a value θ that increases the true objective function $g(\theta)$ unless θ^k is a local maximizer of $g(\theta)$; in this case the local optimum has been found[17]. EM therefore proceeds as follows:

1) **Initialization:** Set $k = 1$. Initialize θ^k to initial guess.
2) **Expectation Step:** Given θ^k, calculate bound $h(\theta|\theta^k)$.
3) **Maximization Step:** Set θ^{k+1} to value of θ that maximizes bound $h(\theta|\theta^k)$.
4) **Convergence Check:** Evaluate $g(\theta^{k+1})$. If $g(\theta)$ has converged, stop. Otherwise set $k = k+1$ and go to 2).

For each k we have $g(\theta^{k+1}) \geq g(\theta^k)$, with equality only if θ^k is a local maximizer. Hence EM guarantees that θ^k will converge to a local maximizer of $g(\theta)$, and does so by iteratively maximizing a *tractable* lower bound on $g(\theta)$. This is illustrated in Fig. 1. In order for this convergence property to apply, it is necessary that the bound $h(\theta|\theta^k)$ touches the objective function. Previous work has, however, shown that 'approximate' EM can be effective in practice despite lacking an analytic convergence guarantee[14], [19]. In the following sections we show that exact EM is intractable in the hybrid model learning problem, but propose a tractable, approximate

EM approach that is effective in practice.

IV. Overview of Approach

In this section we outline the new approach to hybrid model learning for LPHA. The new approach uses EM as described in Section III, except for the key difference that the Expectation Step is approximated in order to ensure tractability.

In order to solve the hybrid model learning problem defined in Section II, we must maximize the value $f(\theta) = p(\mathbf{y}_1^{T+1}|\theta)$. Using EM to do so as described in Section III we first calculate the bound $h(\theta|\theta^k)$ (Expectation Step) and then maximize the bound (Maximization Step). In the case of hybrid model learning, the hidden data X is comprised of both the hidden continuous state sequence $\mathbf{x}_0^{T+1}$ and the hidden discrete mode sequence $\mathbf{m}_0^T$. The observed data consists of the observation sequence $\mathbf{y}_1^{T+1}$. The bound is therefore:

$$h(\theta|\theta^k) = E\left[\log p(\mathbf{y}_1^{T+1}, \mathbf{x}_0^{T+1}, \mathbf{m}_0^T|\theta)\right] + \mathcal{H}, \quad (6)$$

where the expectation is calculated over the hybrid state distribution $p(\mathbf{x}_0^{T+1}, \mathbf{m}_0^T|\mathbf{y}_1^{T+1}, \theta^k)$. Calculating this expectation is not, in the general case, possible in closed form. However in Section V we show how the structure of LPHA can be exploited to make this possible. In doing so we enable existing results for Linear Time Invariant(LTI) systems to be used in the Maximization step, as described in Section VI. These results make it possible for the maximum of the bound (6) to be found analytically. Both the Expectation Step and the Maximization step are intractable in practice, however, because the number of mode sequences $\mathbf{m}_0^T$ is exponential in the number of time steps T. In Section V we therefore introduce an approximation to the bound $h(\theta|\theta^k)$ that makes the Expectation and Maximization Steps tractable.

V. Expectation Step for Hybrid Model Learning

In order to calculate the bound (6) analytically we first use the law of iterated expectations to write the bound in terms of an expectation over the continuous state, conditioned on the discrete mode sequence, and an expectation over discrete mode sequences.

$$\begin{aligned} h(\theta|\theta^k) &= E_{\mathbf{m}_0^T|\mathbf{y}_1^{T+1},\theta^k}\left[E_{\mathbf{x}_0^{T+1}|\mathbf{m}_0^T,\mathbf{y}_1^{T+1},\theta^k}\left[l_c\right]\right] + \mathcal{H} \\ l_c &= \log p(\mathbf{y}_1^{T+1}, \mathbf{x}_0^{T+1}, \mathbf{m}_0^T|\theta). \end{aligned} \quad (7)$$

Writing the expectations out in full gives:

$$\begin{aligned} h(\theta|\theta^k) = \sum_{\mathbf{m}_0^T} \Bigg(& p(\mathbf{m}_0^T|\mathbf{y}_1^{T+1}, \theta^k) \int p(\mathbf{x}_0^{T+1}|\mathbf{m}_0^T, \mathbf{y}_1^{T+1}, \theta^k) \\ & * \log p(\mathbf{y}_1^{T+1}, \mathbf{x}_0^{T+1}, \mathbf{m}_0^T|\theta) d\mathbf{x}_0^{T+1} \Bigg) + \mathcal{H}. \end{aligned} \quad (8)$$

In order to construct this bound we require two key values. The first value is $p(\mathbf{y}_1^{T+1}, \mathbf{x}_0^{T+1}, \mathbf{m}_0^T|\theta)$, also known as the 'completed-data' probability. This describes the joint distribution over observation sequences and state sequences given the model parameters. The second value is $p(\mathbf{x}_0^{T+1}, \mathbf{m}_0^T|\mathbf{y}_1^{T+1}, \theta^k)$, which is the distribution over the hidden hybrid state *given* both the observed observation sequence and the *current guess* for the model parameters. The calculation of these distributions is described in Sections V-A and V-B.

A. Completed-Data Probability

In order to calculate the complete-data probability $p(\mathbf{y}_1^{T+1}, \mathbf{x}_0^{T+1}, \mathbf{m}_0^T|\theta)$, we first use the Markov properties of LPHA to write the probability as:

$$\begin{aligned} & p(\mathbf{y}_1^{T+1}, \mathbf{x}_0^{T+1}, \mathbf{m}_0^T|\theta) = \\ & \quad p(\mathbf{x}_0, \mathbf{m}_0|\theta) \prod_{t=0}^{T-1} p(\mathbf{m}_{t+1}|\mathbf{x}_t, \mathbf{m}_t, \theta) \\ & \quad * \prod_{t=0}^{T} p(\mathbf{y}_{t+1}|\mathbf{x}_{t+1}, \mathbf{m}_t, \theta) p(\mathbf{x}_{t+1}|\mathbf{x}_t, \mathbf{m}_t, \theta). \end{aligned} \quad (9)$$

We can calculate the individual terms in (9) by extending standard results from LTI systems[12] to switching systems. Given the definition of the process noise ω, we have:

$$\begin{aligned} p(\mathbf{x}_{t+1}|\mathbf{x}_t, \mathbf{m}_t, \theta) &= (2\pi)^{-\frac{n_x}{2}} |Q(\mathbf{m}_t)|^{-\frac{1}{2}} e^{-\frac{1}{2}\delta_p' Q^{-1}(\mathbf{m}_t)\delta_p} \\ \delta_p &= \mathbf{x}_{t+1} - A(\mathbf{m}_t)\mathbf{x}_t - B(\mathbf{m}_t)\mathbf{u}_t. \end{aligned} \quad (10)$$

Similarly, given the definition of the observation noise ν, we have:

$$\begin{aligned} p(\mathbf{y}_{t+1}|\mathbf{x}_{t+1}, \mathbf{m}_t, \theta) &= (2\pi)^{-\frac{n_y}{2}} |R(\mathbf{m}_t)|^{-\frac{1}{2}} e^{-\frac{1}{2}\delta_o' R^{-1}(\mathbf{m}_t)\delta_o} \\ \delta_o &= \mathbf{y}_{t+1} - C(\mathbf{m}_t)\mathbf{x}_{t+1} - D(\mathbf{m}_t)\mathbf{u}_t. \end{aligned} \quad (11)$$

The initial probability distribution $p(\mathbf{x}_0, \mathbf{m}_0)$ is given by:

$$\begin{aligned} & p(\mathbf{x}_0, \mathbf{m}_0|\theta) = p(\mathbf{m}_0)(2\pi)^{-\frac{n_x}{2}} |V(\mathbf{m}_0)|^{-\frac{1}{2}} * \\ & \quad e^{-\frac{1}{2}\left[\mathbf{x}_0 - \mu(\mathbf{m}_0)\right]' V^{-1}(\mathbf{m}_0)\left[\mathbf{x}_0 - \mu(\mathbf{m}_0)\right]}. \end{aligned} \quad (12)$$

The hybrid model parameters θ define the distribution of all the necessary values in (9) through (12). Hence the completed-data probability $p(\mathbf{y}_1^{T+1}, \mathbf{x}_0^{T+1}, \mathbf{m}_0^T|\theta)$ can be evaluated in closed form, as required.

B. Hybrid State Distribution

1) Exact Hybrid State Estimation: Calculation of the distribution $p(\mathbf{x}_0^{T+1}, \mathbf{m}_0^T|\mathbf{y}_1^{T+1}, \theta^k)$ is a problem of hidden state estimation for hybrid systems. Different variations of this problem have received a great deal of attention in recent years[1], [2], [20], [21]. The state estimation problem addressed by [1], [2] among others, is to estimate the *current hybrid state* given all observations up to the current time step. These approaches build on early filtering work by [22]. We extend the work of [2] in order to determine $p(\mathbf{x}_0^{T+1}, \mathbf{m}_0^T|\mathbf{y}_1^{T+1}, \theta^k)$, the hybrid state distribution over the *entire sequence*. The key idea is to perform a forward-backward, or 'smoothing', Kalman Filter recursion for each possible mode sequence[23]: as noted by [22], in a non-real time approach, smoothing is preferable to a filtering algorithm such as [22]. This recursion determines $p(\mathbf{x}_0^{T+1}|\mathbf{y}_1^{T+1}, \theta^k, \mathbf{m}_0^T)$, the probability distribution over continuous state sequences conditioned on a particular mode sequence. In addition the forward Kalman Filter residuals are used to determine the observation likelihood conditioned on a mode sequence, $p(\mathbf{y}_1^{T+1}|\mathbf{m}_0^T,\theta)$. Using this likelihood we evaluate $p(\mathbf{m}_0^T|\mathbf{y}_1^{T+1},\theta)$, the posterior probability of the discrete mode sequence using Bayes' Rule:

$$p(\mathbf{m}_0^T|\mathbf{y}_1^{T+1},\theta) = \frac{p(\mathbf{y}_1^{T+1}|\mathbf{m}_0^T,\theta)p(\mathbf{m}_0^T|\theta)}{\sum_{\mathbf{m}_0^T} p(\mathbf{y}_1^{T+1},\mathbf{m}_0^T|\theta)p(\mathbf{m}_0^T|\theta)}. \quad (13)$$

The desired joint distribution over mode sequences and continuous state *sequences* $p(\mathbf{x}_0^{T+1},\mathbf{m}_0^T|\mathbf{y}_1^{T+1},\theta^k)$ is then given by the expression:

$$p(\mathbf{x}_0^{T+1}|\mathbf{y}_1^{T+1},\mathbf{m}_0^T,\theta^k)p(\mathbf{m}_0^T|\mathbf{y}_1^{T+1},\theta^k). \quad (14)$$

2) Approximate Hybrid State Estimation: The calculation of the distribution (14) is intractable in practice, since it requires performing a forward-backward Kalman Filter recursion for every possible discrete mode sequence $\mathbf{m}_0^T$; the number of such mode sequences is exponential in T. In a similar spirit to [2], we therefore approximate the distribution (14) by performing the Kalman Filter recursion only for mode sequences in the restricted set $\mathcal{S}$. This introduces approximation in (14) in two distinct, but important ways. First, the probability of mode sequences that are not in $\mathcal{S}$ are assigned to zero. Second, the posterior probability of each mode sequence in $\mathcal{S}$ cannot be evaluated exactly; the sum in (13) over all possible mode sequences is no longer possible. This means that $p(\mathbf{m}_0^T|\mathbf{y}_1^{T+1},\theta)$ can only be calculated accurate to a factor; this unknown factor is the same for each mode sequence. This is a well-known problem in approximate inference; a standard approach is to choose the factor so that the approximated posteriors, denoted $\tilde{p}(\mathbf{m}_0^T|\mathbf{y}_1^{T+1},\theta)$, sum to one:

$$\tilde{p}(\mathbf{m}_0^T|\mathbf{y}_1^{T+1},\theta) = \frac{1}{c}p(\mathbf{m}_0^T|\mathbf{y}_1^{T+1},\theta), \quad (15)$$

where c is a normalization constant given by:

$$c = \sum_{\mathbf{m}_0^T \in \mathcal{S}} p(\mathbf{m}_0^T|\mathbf{y}_1^{T+1},\theta^k). \quad (16)$$

Denoting the approximate *joint* distribution as $\tilde{p}(\mathbf{x}_0^{T+1},\mathbf{m}_0^T|\mathbf{y}_1^{T+1},\theta^k)$, we have:

$$\begin{aligned}&\tilde{p}(\mathbf{x}_0^{T+1},\mathbf{m}_0^T|\mathbf{y}_1^{T+1},\theta^k)\\ &= \begin{cases} p(\mathbf{x}_0^{T+1}|\mathbf{y}_1^{T+1},\theta^k,\mathbf{m}_0^T)\tilde{p}(\mathbf{m}_0^T|\mathbf{y}_1^{T+1},\theta^k) & \mathbf{m}_0^T \in \mathcal{S} \\ 0 & \mathbf{m}_0^T \notin \mathcal{S}. \end{cases}\end{aligned} \quad (17)$$

Given this approximation of $p(\mathbf{x}_0^{T+1},\mathbf{m}_0^T|\mathbf{y}_1^{T+1},\theta^k)$, we can write the approximation of the bound $h(\theta|\theta^k)$ as:

$$\tilde{h}(\theta|\theta^k) = \sum_{\mathbf{m}_0^T \in \mathcal{S}} \Bigg(\tilde{p}(\mathbf{m}_0^T|\mathbf{y}_1^{T+1},\theta^k) \int p(\mathbf{x}_0^{T+1}|\mathbf{m}_0^T,\mathbf{y}_1^{T+1},\theta^k) * \log p(\mathbf{y}_1^{T+1},\mathbf{x}_0^{T+1},\mathbf{m}_0^T|\theta) d\mathbf{x}_0^{T+1} \Bigg) + \tilde{\mathcal{H}}. \quad (18)$$

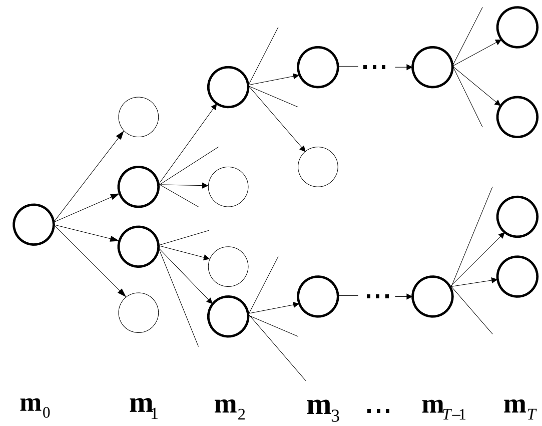

Fig. 2. N-best enumeration. At each time step t in the sequence, N sequences are stored. The successors of these sequences are enumerated, and the N partial sequences $\mathbf{m}_0^{t+1}$ with the greatest posterior probability $p(\mathbf{m}_0^{t+1}|\mathbf{y}_1^{t+2},\theta^k)$ are retained. In this example $N=4$ and the sequences retained at time T are shown in bold.

Note that $h(\theta|\theta^k)$ is a special case of $\tilde{h}(\theta|\theta^k)$ for which the set $\mathcal{S}$ comprises all possible mode sequences $\mathbf{m}_0^T$.

The technical challenge in performing the approximation (17) is to choose the set of mode sequences $\mathcal{S}$. Previous work in hybrid state estimation[2] introduced the idea of capturing as much of the probability mass of the true distribution $p(\mathbf{x}_0^{T+1},\mathbf{m}_0^T|\mathbf{y}_1^{T+1},\theta^k)$ as possible in the approximate distribution, by choosing the *a posteriori most likely* mode sequences. In the Expectation-Maximization approach for hybrid learning we successively maximize a lower bound on the log likelihood $g(\theta)$. The exact bound $h(\theta|\theta^k)$ is the tightest bound possible[17], hence we choose the approximated bound $\tilde{h}(\theta|\theta^k)$ to be as close as possible to the exact bound $h(\theta|\theta^k)$. In order to do so we use the same heuristic as for approximate hybrid state estimation, namely to capture as much of the probability mass of the true distribution $p(\mathbf{x}_0^{T+1},\mathbf{m}_0^T|\mathbf{y}_1^{T+1},\theta^k)$ as possible in the approximate Expectation Step. We therefore aim to include in $\mathcal{S}$ the N most likely mode sequences.

Calculation of the N most likely mode sequences is a challenging problem in itself. We use the iterative approach introduced by [2]. This approach calculates for each time step $t=1,\ldots,T$ the N partial sequences $\mathbf{m}_0^t$ that maximize $\tilde{p}(\mathbf{m}_0^t|\mathbf{y}_1^{t+1},\theta^k)$, restricting the search to sequences that are successors of the N partial sequences $\mathbf{m}_0^{t-1}$ stored at time step $t-1$. The algorithm is illustrated in Fig. 2. Although this algorithm does not guarantee finding the most likely N sequences, in most cases it is sufficient and, most importantly, the algorithm is linear in the number of time steps.

In summary, we have introduced an approach for performing an approximate Expectation Step of the EM algorithm for hybrid model learning. The approach, described in full in Table I, makes calculation of an approximation to the bound $h(\theta|\theta^k)$ tractable by restricting the number of discrete mode sequences considered to the set $\mathcal{S}$. In Section VII we show how the number of sequences N affects the performance of the learning algorithm.

1) **Initialization.** Initialize the set $\mathcal{S}$ of N-best partial sequences to contain the initial mode $\mathbf{m}_0$ and assign $t = 0$.
2) **K-best Enumeration.** Find the N partial sequences $\mathbf{m}_0^{t+1}$ that maximize the probability $\tilde{p}(\mathbf{m}_0^{t+1}|\mathbf{y}_1^{t+2}, \theta^k)$. Evaluation of this probability requires performing a forward Kalman Filter step as described in [2]. Assign the set $\mathcal{S}$ to contain only these sequences.
3) If $t < T$, set $t = t + 1$ and go to 2). Otherwise go to 4).
4) **Backward Kalman Filter Recursion.** For each complete mode sequence $\mathbf{m}_0^T$ in $\mathcal{S}$, perform a backward Kalman Filter recursion to determine the probability $\tilde{p}(\mathbf{x}_0^{T+1}, \mathbf{m}_0^T|\mathbf{y}_1^{T+1}, \theta^k)$. The forward Kalman Filter part of this process has already been performed in 2).

TABLE I
APPROXIMATE EXPECTATION STEP FOR LPHA

VI. MAXIMIZATION STEP FOR HYBRID MODEL LEARNING

In the Maximization Step the bound $\tilde{h}(\theta|\theta^k)$ must be maximized over θ. It is only strictly necessary for $\tilde{h}(\theta|\theta^k)$ to be increased at each iteration, rather than maximized, for convergence of the EM algorithm; however in this section we show that for hybrid model learning in LPHA, the maximum of $\tilde{h}(\theta|\theta^k)$ can be found in closed form. In Section VI-A we find the optimal continuous parameters θ_c, while in Section VI-B we find the optimal discrete parameters θ_d.

A. Maximization Step for Continuous Model Parameters

The key insight in estimating the continuous model parameters is that for a given mode sequence, existing results for LTI systems[12] can be used to find the maximum. We express the full bound (8) as a summation over mode sequences. To find the maximum of $\tilde{h}(\theta|\theta^k)$ we set its derivative with respect to θ_c to zero. We can write this derivative as a summation over the derivative for each mode sequence:

$$\frac{\partial \tilde{h}(\theta|\theta^k)}{\partial \theta_c} = \sum_{\mathbf{m}_0^T \in \mathcal{S}} \Bigg(\tilde{p}(\mathbf{m}_0^T|\mathbf{y}_1^{T+1}, \theta^k) * \frac{\partial}{\partial \theta_c} \int_{\mathbf{x}_0^{T+1}} p(\mathbf{x}_0^{T+1}|\mathbf{m}_0^T, \mathbf{y}_1^{T+1}, \theta^k) * \log p(\mathbf{y}_1^{T+1}, \mathbf{x}_0^{T+1}, \mathbf{m}_0^T|\theta) d\mathbf{x}_0^{T+1} \Bigg) = 0. \tag{19}$$

The optimal values for $A(\mathbf{m})$ and $B(\mathbf{m})$ are found by summing the LTI results from [12] over the mode sequences in $\mathcal{S}$ to give the following equations:

$$\sum_{\mathbf{m}_0^T \in \mathcal{S}} \Bigg(\tilde{p}(\mathbf{m}_0^T|\mathbf{y}_1^{T+1}, \theta^k) \sum_{t \in \mathcal{F}(\mathbf{m}_0^T)} P_{t+1,t}(\mathbf{m}_0^T) \Bigg) = A^*(\mathbf{m}) \sum_{\mathbf{m}_0^T \in \mathcal{S}} \Bigg(\tilde{p}(\mathbf{m}_0^T|\mathbf{y}_1^{T+1}, \theta^k) \sum_{t \in \mathcal{F}(\mathbf{m}_0^T)} P_t(\mathbf{m}_0^T) \Bigg) + B^*(\mathbf{m}) \sum_{\mathbf{m}_0^T \in \mathcal{S}} \Bigg(\tilde{p}(\mathbf{m}_0^T|\mathbf{y}_1^{T+1}, \theta^k) \sum_{t \in \mathcal{F}(\mathbf{m}_0^T)} \mathbf{u}_t \hat{\mathbf{x}}_t{}'(\mathbf{m}_0^T) \Bigg) \tag{20}$$

$$\sum_{\mathbf{m}_0^T \in \mathcal{S}} \Bigg(\tilde{p}(\mathbf{m}_0^T|\mathbf{y}_1^{T+1}, \theta^k) \sum_{t \in \mathcal{F}(\mathbf{m}_0^T)} \hat{\mathbf{x}}_{t+1} \mathbf{u}_t{}' \Bigg) = A^*(\mathbf{m}) \sum_{\mathbf{m}_0^T \in \mathcal{S}} \Bigg(\tilde{p}(\mathbf{m}_0^T|\mathbf{y}_1^{T+1}, \theta^k) \sum_{t \in \mathcal{F}(\mathbf{m}_0^T)} \hat{\mathbf{x}}_t(\mathbf{m}_0^T) \mathbf{u}_t' \Bigg) + B^*(\mathbf{m}) \sum_{\mathbf{m}_0^T \in \mathcal{S}} \Bigg(\tilde{p}(\mathbf{m}_0^T|\mathbf{y}_1^{T+1}, \theta^k) \sum_{t \in \mathcal{F}(\mathbf{m}_0^T)} \mathbf{u}_t \mathbf{u}_t{}' \Bigg), \tag{21}$$

where $\mathcal{F}(\mathbf{m}_0^T)$ is the set of time steps in the sequence $\mathbf{m}_0^T$ for which the mode is $\mathbf{m}$. Members of $\mathcal{F}(\mathbf{m}_0^T)$ are integers in the range $[0, T]$. Solving the set of linear equations (20), (21) yields the optimal values for $A(\mathbf{m})$ and $B(\mathbf{m})$. Similarly the optimal values for $C(\mathbf{m})$ and $D(\mathbf{m})$ are found by performing a weighted sum over the LTI results from [12] to give the system of linear equations:

$$\sum_{\mathbf{m}_0^T} \Bigg(\tilde{p}(\mathbf{m}_0^T|\mathbf{y}_1^{T+1}, \theta^k) \sum_{t \in \mathcal{F}(\mathbf{m}_0^T)} \mathbf{y}_{t+1} \hat{\mathbf{x}}_{t+1}{}'(\mathbf{m}_0^T) \Bigg) = C^*(\mathbf{m}) \sum_{\mathbf{m}_0^T \in \mathcal{S}} \Bigg(\tilde{p}(\mathbf{m}_0^T|\mathbf{y}_1^{T+1}, \theta^k) \sum_{t \in \mathcal{F}(\mathbf{m}_0^T)} P_{t+1}(\mathbf{m}_0^T) \Bigg) + D^*(\mathbf{m}) \sum_{\mathbf{m}_0^T \in \mathcal{S}} \Bigg(\tilde{p}(\mathbf{m}_0^T|\mathbf{y}_1^{T+1}, \theta^k) \sum_{t \in \mathcal{F}(\mathbf{m}_0^T)} \mathbf{u}_t \hat{\mathbf{x}}_{t+1}{}'(\mathbf{m}_0^T) \Bigg) \tag{22}$$

$$\sum_{\mathbf{m}_0^T \in \mathcal{S}} \Bigg(\tilde{p}(\mathbf{m}_1^T|\mathbf{y}_1^{T+1}, \theta^k) \sum_{t \in \mathcal{F}(\mathbf{m}_0^T)} \mathbf{y}_{t+1} \mathbf{u}_t{}' \Bigg) = C^*(\mathbf{m}) \sum_{\mathbf{m}_0^T \in \mathcal{S}} \Bigg(\tilde{p}(\mathbf{m}_0^T|\mathbf{y}_1^{T+1}, \theta^k) \sum_{t \in \mathcal{F}(\mathbf{m}_0^T)} \hat{\mathbf{x}}_{t+1}(\mathbf{m}_0^T) \mathbf{u}_t{}' \Bigg) + D^*(\mathbf{m}) \sum_{\mathbf{m}_0^T \in \mathcal{S}} \Bigg(\tilde{p}(\mathbf{m}_0^T|\mathbf{y}_1^{T+1}, \theta^k) \sum_{t \in \mathcal{F}(\mathbf{m}_0^T)} \mathbf{u}_t \mathbf{u}_t{}' \Bigg). \tag{23}$$

Using the optimal values for $A(\mathbf{m})$, $B(\mathbf{m})$, $C(\mathbf{m})$ and $D(\mathbf{m})$ we obtain the optimal covariance matrices for the noise processes:

$$Q^*(\mathbf{m}) = \sum_{\mathbf{m}_0^T \in \mathcal{S}} \Bigg(\frac{\tilde{p}(\mathbf{m}_0^T|\mathbf{y}_1^{T+1}, \theta^k)}{|\mathcal{F}(\mathbf{m}_0^T)|} \sum_{t \in \mathcal{F}(\mathbf{m}_0^T)} \Big(P_{t+1}(\mathbf{m}_0^T) - A^*(\mathbf{m}) P_{t,t+1}(\mathbf{m}_0^T) - B^*(\mathbf{m}) \mathbf{u}_t \hat{\mathbf{x}}_{t+1}{}'(\mathbf{m}_0^T) \Big) \Bigg) \tag{24}$$

$$R^*(\mathbf{m}) = \sum_{\mathbf{m}_0^T \in \mathcal{S}} \Bigg(\frac{\tilde{p}(\mathbf{m}_0^T|\mathbf{y}_1^{T+1}, \theta^k)}{|\mathcal{F}(\mathbf{m}_0^T)|} \sum_{t \in \mathcal{F}(\mathbf{m}_0^T)} \Big(\mathbf{y}_{t+1} - C^*(\mathbf{m}) \hat{\mathbf{x}}_{t+1}(\mathbf{m}_0^T) - D^*(\mathbf{m}) \mathbf{u}_t \Big) \mathbf{y}_{t+1}{}' \Bigg). \tag{25}$$

Finally, the optimal parameters for the initial continuous distribution are given by:

$$\mu^*(\mathbf{m}) = \sum_{\{\mathbf{m}_0^T|\mathbf{m}_0=\mathbf{m}\}} \tilde{p}(\mathbf{m}_0^T|\mathbf{y}_1^{T+1},\theta^k)\hat{\mathbf{x}}_0{}'(\mathbf{m}_0^T)$$
$$V^*(\mathbf{m}) = \sum_{\{\mathbf{m}_0^T|\mathbf{m}_0=\mathbf{m}\}} \tilde{p}(\mathbf{m}_0^T|\mathbf{y}_1^{T+1},\theta^k)P_{0,0}(\mathbf{m}_0^T), \quad (26)$$

where the summation is over the discrete mode sequences $\mathbf{m}_0^T$ for which the initial mode $\mathbf{m}_0$ is the same as $\mathbf{m}$.

In (20) through (26) we use the following definitions:

$$\hat{\mathbf{x}}_t(\mathbf{m}_0^T) = E\big[\mathbf{x}_t|\mathbf{m}_0^T,\mathbf{y}_1^{T+1},\theta^k\big]$$
$$P_{t_1,t_2}(\mathbf{m}_0^T) = E\big[\mathbf{x}_{t_1}\mathbf{x}_{t_2}{}'|\mathbf{m}_0^T,\mathbf{y}_1^{T+1},\theta^k\big]. \quad (27)$$

We have therefore shown that for LPHA, calculation of the optimal continuous parameters in the maximization step can be carried out in closed form. By choosing the set $\mathcal{S}$ to contain a subset of the possible discrete mode sequences we make this calculation tractable despite the exponential number of such sequences.

B. Maximization Step for Discrete Model Parameters

We now maximize the bound $\tilde{h}(\theta|\theta^k)$ with respect to the discrete parameters θ_d. For each guard condition c_i we must find the optimal value of the transition matrix T_i. It is not sufficient simply to set the derivative of $\tilde{h}(\theta|\theta^k)$ with respect to each element of T_i to zero; in order to ensure that T_i describes a valid transition probability matrix, we must impose the constraint that each column in T_i has elements that sum to one. We therefore perform the following constrained optimization for every mode $\mathbf{m} \in \mathcal{X}_d$ and guard $i \in \mathcal{C}$:

$$\text{Maximize over } T_i(j,\mathbf{m}),\ \ j=1,\ldots,|\mathcal{X}_d| : \quad (28)$$
$$\tilde{h}(\theta|\theta^k)$$
$$\text{Subject to:}$$
$$\sum_{j=1}^{|\mathcal{X}_d|} T_i(j,\mathbf{m}) = 1. \quad (29)$$

We use a Lagrange multiplier approach to solve this optimization. The Lagrangian L is given by:

$$L = \tilde{h}(\theta|\theta^k) + \lambda\bigg(\sum_{j=1}^{|\mathcal{X}_d|} T_i(j,\mathbf{m}) - 1\bigg), \quad (30)$$

where λ is a Lagrange multiplier. We must set the derivative of L with respect to each term $T_i(j,\mathbf{m})$ as well as λ to zero. In order to do so we must evaluate the expression $\frac{\partial \tilde{h}(\theta|\theta^k)}{\partial T_i(j,\mathbf{m})}$:

$$\frac{\partial \tilde{h}(\theta|\theta^k)}{\partial T_i(j,\mathbf{m})} = \frac{\partial}{\partial T_i(j,\mathbf{m})} \sum_{\mathbf{m}_0^T\in\mathcal{S}} \tilde{p}(\mathbf{m}_0^T|\mathbf{y}_1^{T+1},\theta^k) *$$
$$\sum_{t=1}^{T}\int p(\mathbf{x}_{t-1}|\mathbf{y}_1^{T+1},\mathbf{m}_0^T,\theta^k)\log p(\mathbf{m}_t|\mathbf{m}_{t-1},\mathbf{x}_{t-1},\theta)d\mathbf{x}_{t-1}. \quad (31)$$

Evaluating the derivative in this expression is made challenging by the dependence of mode transitions on the continuous state. However since the distribution $p(\mathbf{m}_t|\mathbf{m}_{t-1},\mathbf{x}_{t-1},\theta)$ has a constant value for each guard condition c_i, we can rewrite the integral over $\mathbf{x}_{t-1}$ as a sum over guard conditions:

$$\frac{\partial \tilde{h}(\theta|\theta^k)}{\partial T_i(j,\mathbf{m})} = \frac{\partial}{\partial T_i(j,\mathbf{m})} \sum_{\mathbf{m}_0^T\in\mathcal{S}} \tilde{p}(\mathbf{m}_0^T|\mathbf{y}_1^{T+1},\theta^k)$$
$$* \sum_{t=1}^{T}\sum_{c_i\in\mathcal{G}} p_{c_i}(\mathbf{m}_0^T)\log T_i(\mathbf{m}_t,\mathbf{m}_{t-1}), \quad (32)$$

where $p_{c_i}(\mathbf{m}_0^T)$ is the probability that guard condition c_i is satisfied given the discrete mode sequence $\mathbf{m}_0^T$, the observation sequence $\mathbf{y}_1^{T+1}$ and the current guess of the parameters θ^k. This probability can be written as:

$$p_{c_i}(\mathbf{m}_0^T) = \int_{\mathcal{C}_i} p(\mathbf{x}_{t-1}|\mathbf{y}_1^{T+1},\mathbf{m}_0^T,\theta^k)d\mathbf{x}_{t-1}, \quad (33)$$

where $\mathcal{C}_i$ is the region of $\mathbf{x}_{t-1}$ for which the guard c_i is satisfied. Evaluating (33) requires integrating a Gaussian over $\mathcal{C}_i$. Prior work has shown that for special classes of guard conditions c_i such as rectangular and linear guards, the integral can be evaluated efficiently using a lookup of Gaussian cumulative distribution functions[2]. Using this approach we can evaluate the derivative in (32) with respect to $T_i(j,\mathbf{m})$:

$$\frac{\partial \tilde{h}(\theta|\theta^k)}{\partial T_i(j,\mathbf{m})} = \sum_{\mathbf{m}_0^T\in\mathcal{S}}\bigg(\tilde{p}(\mathbf{m}_0^T|\mathbf{y}_1^{T+1},\theta^k)\sum_{t\in\mathcal{F}(\mathbf{m}_0^T)}\frac{p_{c_i}(\mathbf{m}_0^T)}{T_i(j,\mathbf{m})}\bigg), \quad (34)$$

where $\mathcal{F}(\mathbf{m}_0^T)$ contains the time steps in $\mathbf{m}_0^T$ for which $\mathbf{m}_{t-1} = \mathbf{m}$ and $\mathbf{m}_t = j$. Using the expression (34) we now set to zero the derivative of the Lagrangian (30) with respect to $T_i(j,\mathbf{m})$ to yield:

$$\lambda\cdot T_i(j,\mathbf{m}) = -\sum_{\mathbf{m}_0^T\in\mathcal{S}} \tilde{p}(\mathbf{m}_0^T|\mathbf{y}_1^{T+1},\theta^k) *$$
$$\sum_{t\in\mathcal{F}(\mathbf{m}_0^T)} p_{c_i}(\mathbf{m}_0^T). \quad (35)$$

We now sum over each target mode j to yield:

$$\lambda\sum_{j=1}^{|\mathcal{X}_d|} T_i(j,\mathbf{m}) = -\sum_{j=1}^{|\mathcal{X}_d|}\bigg(\sum_{\mathbf{m}_0^T\in\mathcal{S}} \tilde{p}(\mathbf{m}_0^T|\mathbf{y}_1^{T+1},\theta^k) *$$
$$\sum_{t\in\mathcal{F}(\mathbf{m}_0^T)} p_{c_i}(\mathbf{m}_0^T)\bigg). \quad (36)$$

Setting the derivative of L with respect to λ to zero yields the original constraint (29). Substituting this into (36) gives the optimal value for λ. We substitute this back into (35) to give the following expression for the optimal value of $T_i(j, \mathbf{m})$:

$$T_i^*(j, \mathbf{m}) = \frac{\sum_{\mathbf{m}_0^T \in \mathcal{S}} \tilde{p}(\mathbf{m}_0^T | \mathbf{y}_1^{T+1}, \theta^k) \sum_{t \in \mathcal{F}(\mathbf{m}_0^T)} p_{c_i}(\mathbf{m}_0^T)}{\sum_{\mathbf{m} \in \mathcal{X}_d} \left(\sum_{\mathbf{m}_0^T \in \mathcal{S}} \tilde{p}(\mathbf{m}_0^T | \mathbf{y}_1^{T+1}, \theta^k) \sum_{t \in \mathcal{F}(\mathbf{m}_0^T)} p_{c_i}(\mathbf{m}_0^T) \right)}. \tag{37}$$

Using a similar Lagrange multiplier method to determine the optimal initial mode distribution, we arrive at the following optimal parameters:

$$p^*(\mathbf{m}_0 = \mathbf{m}) = \frac{\sum_{\mathbf{m}_0^T \in \mathcal{Q}} \tilde{p}(\mathbf{m}_0^T | \mathbf{y}_1^{T+1}, \theta^k)}{\sum_{\mathbf{m} \in \mathcal{X}_d} \left(\sum_{\mathbf{m}_0^T \in \mathcal{Q}} \tilde{p}(\mathbf{m}_0^T | \mathbf{y}_1^{T+1}, \theta^k) \right)}, \tag{38}$$

where $\mathcal{Q}$ is the set of mode sequences $\mathbf{m}_0^T$ for which the initial mode $\mathbf{m}_0 = \mathbf{m}$. In other words, the optimal initial discrete mode distribution is found by taking the *weighted fraction of initial modes* in the sequence set $\mathcal{S}$; the weighting is the posterior probability of each mode sequence given the current guess for the parameters.

In summary, we have shown that the Maximization Step for LPHA can be performed analytically, by providing a closed form expression for the hybrid parameters θ that maximize the bound $\tilde{h}(\theta|\theta^k)$. In the case where $\tilde{h}(\theta|\theta^k) = h(\theta|\theta^k)$, i.e. a complete Expectation Step has been carried out, the Maximization step is typically intractable due to the exponential number of mode sequences in the set $\mathcal{S}$. However, by restricting the size of the set $\mathcal{S}$, the Maximization Step is made tractable. Note that the Maximization Step is still guaranteed to find the maximum of the bound $\tilde{h}(\theta|\theta^k)$.

VII. Simulation Results

In this section we demonstrate the new model learning approach using a simulated planetary rover example. We consider the subsystem consisting of a motor and a wheel. An intermittent fault causes the wheel to 'stick' at random, and the probability of the wheel sticking is different depending on whether the wheel is being driven forwards or backwards. When stuck, the wheel experiences increased friction.

The wheel subsystem is modeled as a LPHA with two modes. In Mode 1 the wheel operates normally, while Mode 2 the wheel is stuck. The hidden continuous state $\mathbf{x}$ is $\left[i\ \dot{\theta}\right]'$ where i is the current in the motor and $\dot{\theta}$ is the angular velocity of the wheel. Noisy observations $\mathbf{y}$ of the wheel velocity are available through an encoder. The input $\mathbf{u}$ is the voltage applied to the driver circuit.

The true continuous parameters are given by:

$$A(1) = \begin{bmatrix} -0.0044 & -0.0203 \\ 0.0366 & 0.1665 \end{bmatrix} \quad B(1) = \begin{bmatrix} 0.92 \\ 0.81 \end{bmatrix}$$
$$A(2) = \begin{bmatrix} -0.0032 & -0.0142 \\ 0.0256 & 0.1106 \end{bmatrix} \quad B(2) = \begin{bmatrix} 0.93 \\ 0.71 \end{bmatrix}$$
$$C(1) = C(2) = \begin{bmatrix} 0 & 1 \end{bmatrix} \quad C(1) = D(2) = 0. \tag{39}$$

The true guard conditions are given by:

$$\mathcal{C}_1 = [-\infty\ 0] \quad T_1 = \begin{bmatrix} 0.9 & 0.2 \\ 0.1 & 0.8 \end{bmatrix}$$
$$\mathcal{C}_2 = [0\ \infty] \quad T_2 = \begin{bmatrix} 0.5 & 0.1 \\ 0.5 & 0.9 \end{bmatrix}, \tag{40}$$

where the guard regions $\mathcal{C}_1$ and $\mathcal{C}_2$ are defined over $\dot{\theta}$.

While we would like to evaluate the performance of the new algorithm against the likelihood $p(\mathbf{y}_1^{T+1}|\theta)$, evaluation of this is intractable. The change in the lower bound $\tilde{h}(\theta|\theta^k)$ can, however, be evaluated. We therefore analyze the convergence of the algorithm in terms of this change. Fig. 3 shows the change in the lower bound over the course of a typical learning run. The change in $\tilde{h}(\theta|\theta^k)$ is always positive, and decreases almost monotonically as learning proceeds, indicating convergence of the learned parameters. As has been previously demonstrated for LTI systems[12] the continuous model parameters typically do not converge to the true parameters. This is because, first, there are an infinite number of equivalent LTI systems, and second, because the EM algorithm is a local optimization approach. In many cases, however, the discrete parameters *do* converge to values close to the true ones. A typical case is shown in Fig 4.

Fig. 5 shows the effect of the number of tracked mode sequences on the performance of the algorithm. We cannot use the likelihood $p(\mathbf{y}_1^{T+1}|\theta)$ to evaluate performance, but since we are often interested in estimating the hidden mode, we use the fraction of Maximum A Posteriori (MAP) mode estimate errors at convergence as a performance criterion. Counterintuitively, the number of tracked mode sequences N has essentially no impact on the performance of the learning algorithm. Extensive testing also showed that increasing N does not, on average, increase the number of iterations to convergence. This is a surprising empirical result which warrants further investigation. These results motivate the use of 'hard' EM, where the single most likely trajectory is tracked, in order to minimize computation time.

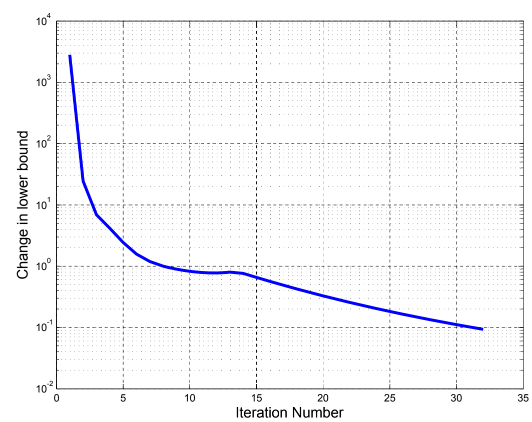

Fig. 3. Convergence of the approximate EM algorithm for hybrid model learning for a typical run, with $N = 10$ and $T = 100$. The change in the bound $\tilde{h}(\theta|\theta^k)$ decreases almost monotonically as EM proceeds. This shows convergence of the learned parameters.

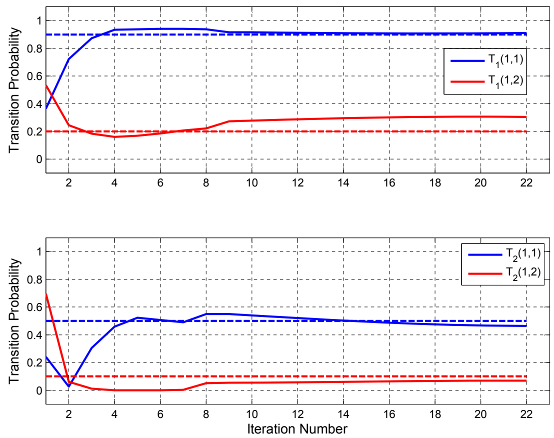

Fig. 4. Convergence of the discrete parameters for a typical run, with $N = 10$ and $T = 100$. The transition probabilities for guard condition $\mathcal{C}_1$ (top) and for guard condition $\mathcal{C}_2$ (bottom) converge to values close to the true ones (shown dashed).

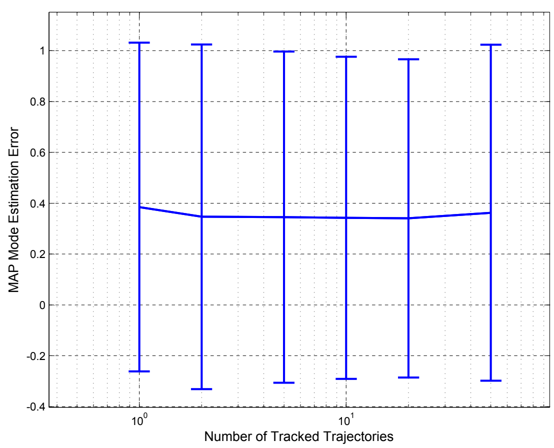

Fig. 5. Fraction of MAP mode estimation errors at convergence against number of mode sequences tracked (N) averaged over 20 random runs. Convergence was defined as a change of less than 0.1 in $\tilde{h}(\theta|\theta^k)$. For each run, the initial continuous parameters were chosen randomly by perturbing their true values by up to 50%, and the discrete transition probabilities were chosen from a uniform distribution between 0 and 1. The error bars represent two standard deviations. N has no almost effect on the MAP error.

VIII. Conclusion

We have presented a new approximate Expectation-Maximization approach for learning the parameters of dynamic systems with hybrid discrete-continuous state. The new method can handle transitions in the discrete state that are conditioned on the continuous state, and can learn the effects of control inputs on the system. Approximation is introduced by tracking a subset of the discrete mode sequences in the Expectation step. Simulation results showed that the approach converges in practice, and that the number of tracked mode sequences does not affect the performance of the algorithm.

References

[1] L. Blackmore, S. Funiak, and B. C. Williams, "Combining stochastic and greedy search in hybrid estimation," in *Proceedings of 20th National Conference on Artificial Intelligence (AAAI05)*, 2005.

[2] M. Hofbaur and B. C. Williams, "Mode estimation of probabilistic hybrid systems," in *Intl. Conf. on Hybrid Systems: Computation and Control*, May 2002.

[3] S. M. Oh, J. M. Rehg, T. Balch, and F. Dallaert, "Learning and inference in parametric switching linear dynamic systems," in *Proc. IEEE Int. Conf. Comp. Vision*, 2005.

[4] O. L. V. Costa, M. D. Fragoso, and R. P. Marques, *Discrete-Time Markovian Jump Linear Systems*. Springer-Verlag, 2005.

[5] E. K. Boukas, P. Shi, and K. Benjelloun, "Robust stochastic stabilization of discrete-time linear systems with markovian jumping parameters," *Journal of Dynamic Systems, Measurement, and Control*, vol. 121, pp. 331–334, 1999.

[6] A. N. Vargas, W. Furloni, and J. B. R. do Val, "Constrained MPC of jump linear systems with noise and non-observed markov state," in *Proc. ACC*, 2006.

[7] L. Blackmore, A. Bektassov, M. Ono, and B. C. Williams, "Robust, optimal predictive control of jump markov linear systems using particles," in *Hybrid Systems: Computation and Control, HSCC 2007*, ser. Lecture Notes in Computer Science, A. Bemporad, A. Bicchi, and G. Buttazzo, Eds. Springer Verlag, 2007, vol. 4416, pp. 104–117.

[8] U. Lerner, B. Moses, M. Scott, S. McIlraith, and D. Koller, "Monitoring a complex physical system using a hybrid dynamic bayes net," in *Proc. 18th Annual Conf. on Uncertainty in Artificial Intelligence*, Edmonton, Canada, August 2002.

[9] M. Fox, M. Ghallab, G.Infantes, and D.Long, "Robot introspection through learned hidden markov models," *Artifical Intelligence*, vol. 170, no. 2, pp. 59–113, Feb. 2006.

[10] S. Dong, "Unsupervised learning and recognition of physical activity plans," Master's thesis, Massachusetts Institute of Technology, Cambridge, 2007.

[11] Z. Ghahramani and G. E. Hinton, "Parameter estimation for linear dynamical systems," Department of Computer Science, University of Toronto, Tech. Rep. CRG-TR-96-2, Feb 1996. [Online]. Available: citeseer.ist.psu.edu/ghahramani96parameter.html

[12] S. Cheng and P. N. Sabes, "Modeling sensorimotor learning with linear dynamical systems," *Neural Computation*, vol. 18, no. 4, pp. 760–793, 2006.

[13] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the e m algorithm," *J. Royal Statistical Society, Series B*, vol. 39, pp. 1–38, 2006.

[14] Z. Ghahramani and G. E. Hinton, "Variational learning for switching state-space models," *Neur. Comp.*, vol. 12, no. 4, pp. 831–864, 2000.

[15] H. Balakrishnan, I. Hwang, J. Jang, and C. Tomlin, "Inference methods for autonomous stochastic linear hybrid systems," in *Hybrid Systems: Computation and Control, HSCC 2007*, ser. Lecture Notes in Computer Science, G. Goos, J. Hartmanis, and J. van Leeuwen, Eds. Springer Verlag, 2004, vol. 2993, pp. 64–79.

[16] M. Henry, "Model-based estimation of probabilistic hybrid automata," Master's thesis, Massachusetts Institute of Technology, Cambridge, 2002.

[17] T. Minka, "Expectation-maximization as lower bound maximization," 1998. [Online]. Available: citeseer.ist.psu.edu/minka98expectationmaximization.html

[18] F. Dellaert, "The expectation maximization algorithm," College of Computing, Georgia Institute of Technology, Tech. Rep. GIT-GVU-02-20, 2002.

[19] R. Neal and G. Hinton, "A view of the em algorithm that justifies incremental, sparse, and other variants," in *Learning in Graphical Models*, M. I. Jordan, Ed. Kluwer, 1998.

[20] A. Doucet, J. de Freitas, K. Murphy, and S. Russell, "Rao-Blackwellised particle filtering for dynamic Bayesian networks," in *Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI)*, 2000.

[21] U. Lerner, R. Parr, D. Koller, and G. Biswas, "Bayesian fault detection and diagnosis in dynamic systems," in *Proc. of the 17th National Conference on AI*, July 2000. [Online]. Available: citeseer.nj.nec.com/lerner00bayesian.html

[22] H. A. P. Blom and Y. Bar-Shalom, "The interacting multiple model agorithm for systems with markovian switching coefficients," vol. 33, pp. 780–783, Aug. 1988.

[23] R. H. Shumway and D. S. Stoffer, "An approach to time series smoothing and forecasting using the em algorithm," *Journal of Time Series Analysis*, vol. 3, no. 4, pp. 253–264, 1982.