I/O HSMM: Learning Behavioral Dynamics of a Cognitive Wireless Network Node From Spectrum Sensing

Silvija Kokalj-Filipovic, Joel Goodman, Crystal Bertoncini Acosta, George Stantchev

Naval Research Laboratory
{silvija.kokalj-filipovic, crystal.acosta, joel.goodman, george.stantchev}@nrl.navy.mil

***Abstract*—We introduce a generative model, dubbed *I/O HSMM*, for learning the bi-modal behavioral dynamics of a network of cognitive radios (CRs). Each of the two modes of the CRs is represented as a Hidden Semi-Markov model (HSMM), where the states, state durations and emissions, transition probabilities between states, and transitions between modes are uncovered based solely on RF spectrum sensing. The learning of the CR dynamics is non-parametric and derived from the Hierarchical Dirichlet Process (HDP), with the switching between the two modes modeled as a latent variable that is estimated as a part of the learning process. The non-parametric model provides flexibility in handling unknown communication protocols. We evaluate the quality of learning against ground truth, and demonstrate that this approach is promising and merits extension to more complex models.**

I. Introduction

Although the concept of a *cognitive radio* (CR) was introduced almost two decades ago [1], it has recently become feasible to physically realize an operational CR network due to the spread of software defined radios. Fundamentally, a CR is a device with the ability to adaptively select a frequency for transmission in a part of the spectrum that is unutilized in a specific geographic location. Efficient communications may require changes in parameters other than frequency, such as modulation, error-control coding, transmit power or activity pattern. As cognitive networks penetrate application space [2], accurate modeling of other (cognitive) spectrum users becomes essentially important for prediction of available "spectrum holes". That is why we do not limit ourselves to the usual CR model involving primary user (PU) and secondary user (SU) [3]. We refer to nodes of the CR network that is sensing the environment as *observers*, while active radios in other networks that are transmitting, receiving and adapting their parameters as CRs. Each observer can share information with other radios within its network for source localization. This enables observers to estimate the interference that the CRs are subject to in other networks [4].

The use of machine learning for a CR to characterize the environment and adaptively alter its transmission parameters is not new [5]. However, in this paper, learning of the CR behavior is performed by an observer that is not a part of the CR network. From an observer perspective, we adopt a generative model to represent a CR state machine, and estimate the conditional probability of the CR adapting its transmission parameters contingent upon its current (hidden) protocol state, as well as on inputs from the environment in the form of binary-quantized interference. We assume that the node is switching between two state-machines when evaluation of the most-recent window of binary interference indicators (BIIs) dictates so. We model the evaluation of the BIIs by a function referred to as the *switching function (SF)*. Note that we assume the CR and the observer perceive the same BIIs, and the SF is known to the observer. Because the observer has no *apriori* knowledge of the CR state machine, we employ Bayesian non-parametric learning in the form of an HDP-HSMM [6]. The semi-Markov model aspect of the HDP-HSMM [7] is well-suited to characterize the amount of sojourn time in any particular state, which for CRs may represent the transmission length. For example, the acknowledgments (*acks*) may have distinctly shorter durations than those of data packets, and potentially be transmitted with higher power to avoid re-transmission of the sequence of much longer packets preceding the *ack* in case it gets lost. Because we do not parametrically model the structure of the HSMM, and in particular the number of states or their semantics, a state is identified based on the distinct statistical description of its emission parameters [8].

As an example, consider a CR with two modes; the first mode is 'nominal', in the absence of interference, in which the CR transmits at a lower power level in each state, while the second mode is 'active' in which in every state the CR transmits at a higher power level to overcome interference. The objective of the observer then is to learn the HDP-HSMM for each of the individual modes as well as the conditions under which the CR switches modes. Because the interference estimate (BII) is used by the observer to infer the probability of a CR switching modes, we construct an extension of the conventional HSMM to *Input/Output HSMM (I/O HSMM)*, whereby state dynamics are conditioned on the input (BII).

The rest of this paper is organized as follows. In Section II we explain the learning model preliminaries and introduce our system model based on an example. Section III depicts the I/O HSMM extension to HDP HSMM. Section IV presents the simulation results of I/O HSMM performance, and Section V concludes with a brief summary.

II. System Model through an Example

A. Basic HSMM Model

A hidden Markov model [9] is a doubly stochastic process, with the underlying process being a discrete-time finite-state homogeneous Markov chain. The state sequence $\{s_t\}$ is not observable, hence, hidden. Here t denotes discrete time, and $s_t \in S$, $|S| < \infty$. The second stochastic process is producing a sequence of observations $\{y_t\}$, conditioned on the hidden state sequence s.t. emission probability density which describes state i observations is $f_i(y_t) = p(y_t|s_t = i)$. Due to the non-zero probability of self-transition of a (non-absorbing) state at each discrete time instant, the state duration of an HMM implicitly follows a geometric distribution. This limits the use of the HMM in some applications. It can be overcome by allowing the underlying process to be a semi-Markov chain. This modified model is known as *Hidden semi-Markov model (HSMM)*. In HSMM each state i is associated with a distinct sojourn time (duration) d, which is a random variable with probability distribution (PD) $p_i(d)$. The state-duration is commonly identified with the number of observations produced while in that state. Various degrees of dependency between state durations and successive state-transition probabilities are defined in [7]. We adopt the HSMM variant in which state duration is assumed to be independent of the previous states, and the transition probability $a^{i,d'}(j,d)$ to state i of duration d' after a sojourn of length d in state j is $a^{i,d'}(j,d) = a^i(j,d) = a_{ij}p_j(d)$, where $p_j(\bullet)$ is the duration PD of state j, and $a_{ij} = P(s_{t+1} = i|s_t = j)$, is the HMM-style transition probability from j to i (independent of durations). In view of the difference in the number of observations produced per state, we refer to a state in HSMM as the super-state to distinguish it from an HMM state. Sequence of super states is denoted by $\{z_k\}$, where k denotes the sequence index, and not discrete time. Fig. 1 presents two instances of the HSMM model (one for each behavioral mode). Note that super-state z_1 has a duration equivalent to D_1 discrete time ticks, producing D_1 emissions which are drawn i.i.d. from $f_{z_1}(\bullet)$. For a comprehensive overview of the HSMM see [7].

B. Cognitive Radio Ground-Truth and Observation Model

In this paper we implicitly focus on ad-hoc wireless network nodes, and our examples with one CR reflect this assumption. However, the approach is equally applicable to infrastructural wireless networks. To present and analyze the learning mechanism, we use an example of the CR *ground truth (GT)* model consisting of protocol states and modes, and which involves per-packet power adjustment for different types of traffic [10] and for distinct interference levels. For the sake of simplicity, other parameters are not modified. We assume that control packets are more important for the functioning of the entire network, and that they are transmitted with a higher power to ensure a better average SNR across the network. Moreover, control packets can be medium access requests, traffic control handshaking (ACKs), network discovery, routing signaling and

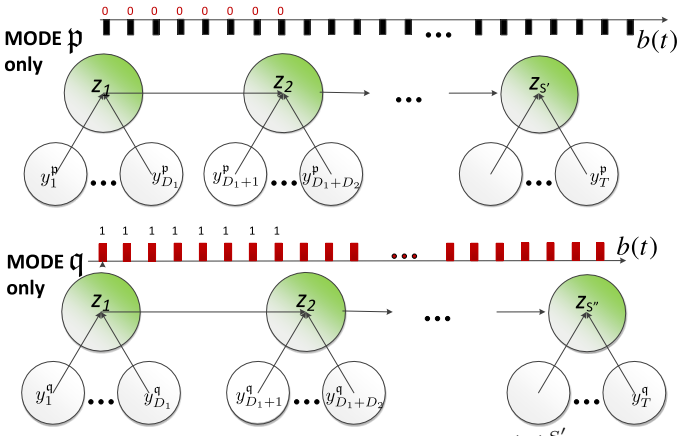

Fig. 1. HSMM: Markov chain on a set of super-states $(z_i)_{i=1}^{S'}$ belonging to mode $\mathfrak{p}$ only or mode $\mathfrak{q}$ only, as the binary switch is trivial. Number of leaf nodes D_i associated with a super-state i follows *state-and-mode-specific* (duration) distribution. Values of leaf nodes (observations) follow *state-and-mode-specific* (emission) distribution

many more. We assume that each particular family of control or data packets $\mathcal{F}_i$ is transmitted with a different power P_i^t. The observer, who performs the learning task, is receiving the node's transmissions over a log-normal channel. Hence, each transmission from $\mathcal{F}_i$ is perceived by the observer in the log domain as a normal variable R_i whose mean is a known function of P_i^t.

Let us assume that the received power is sampled at regular intervals *(time ticks)*. Consequently, according to the adopted HSMM model, the observed node is emitting a sequence of Gaussian random variables $\{y_t\}$ from an unknown number of super-states $\{z_j\}$, each drawn from Gaussian pdf with unknown parameters $\{\mu_i, \sigma_i\}$, conditional on $z_j = i$. Each super-state i has an unknown initial probability π_i^0, and its transition probabilities A_{ij} to other super-states are unknown. In addition, each super-state's duration is described by a Poisson random variable denoting the number of time ticks, and whose parameter λ_i is distinct. Hence, we aim to learn the posterior probability of the following set of parameters $\Theta = \{\theta_i = (\mu_i, \sigma_i), \lambda_i, \pi_i, A_{ij}, j \in S\}$. For a fixed mode, i.e. without any external interference, Θ describes fully the behavioral dynamics of the CR node.

Fig. 1 illustrates this scenario, where the BII sequences are either all zeros, or all ones, and therefore the CR node is perpetually either in mode $\mathfrak{p}$ or in mode $\mathfrak{q}$. Recall that BII $b(t)$ is a quantized interference measurement at each time tick t observed by both the CR and the observer. State unfolding in both examples is represented as a trajectory of super-states $\left\{z_t^{(k)}\right\}$, for a fixed $k \in \{\mathfrak{p}, \mathfrak{q}\}$ where each super state i persists for D_t^i successive time ticks, and where D_t^i is a Poisson random variable of intensity $\lambda_i^{(k)}$. The transition probabilities between super states i and j in mode k are denoted $A_{ij}^{(k)}$, where $A_{ii}^{(k)} = 0$. Note that (in both modes) the super-state is intended to model a distinct time-span of node's emissions, which also may have an interpretation related to the protocol state machine. For instance D_t^i consecutive samples from distribution $\mathcal{N}(\mu_i, \sigma_i)$ may correspond to a packet of a certain

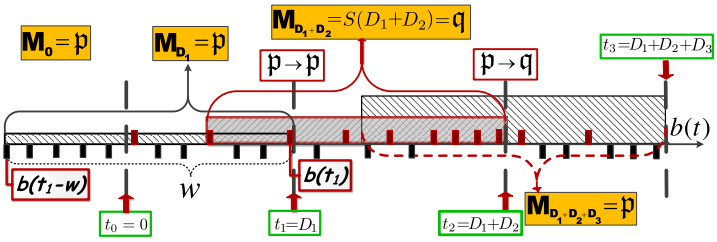

Fig. 2. Illustration of the SF that evaluates a window of indicators $b(t-w+1:t)$ at each state boundary $t=\sum_{i=1}^{\ell} D_i, \ell=1,2,\cdots$

duration, transmitted with a certain power.

By using a GT example we do not intend to narrow the scope of the proposed learning. Let us illustrate this on an example different from the GT. As already mentioned, data packets are usually followed by acknowledgment (*ack*) packets. As *acks* are very short, it is justified to apply a less efficient modulation, provided it works well in very low signal-to-noise ratio (SNR) conditions. Such robust modulation prevents 'loosing' a large data packet due to a loss of its *ack* in case of channel deterioration. That is why WiFi *acks* (and other control packets) are commonly BPSK-modulated, while higher-order modulation is applied to data packets. Hence, if we sample at a symbol rate, we will collect different phase histograms (constellations) during *ack* and data transmission. Also, given cognition, the CR may detect interference and decide that its data packets be modulated with some low-order QAM instead of the nominal high-order modulation. Using phases as features, the learning mechanism should be able to recognize each of these packet transmissions as separate hidden states. The interpretation of the states is outside of the learning scope here, although learning the HSMM parameters is certainly the first step in recognizing domain-relevant states and modes.

Mode switching as a result of the evaluation of the SF is equivalent to switching between super-states of two distinct HSMMs. Such a switch is associated with an additional set of *intermode* transition probabilities A_{ij}^{kl}, with $k \neq l$, indicating a transition from super-state i in mode k to superstate j in mode l. We learn these transition probabilities implicitly through the switching function. Note that we know *apriori* that the GT system flips between two such modes, and we know the criterion for flipping, but not the flipping times. Knowing the switching criterion makes our approach semi-parametric.

The switching function (SF) in the GT model is evaluated at boundaries of network protocol states $t_\ell = \sum_{i=1}^{\ell} D_i, \ell = 1,2,\cdots$ (red arrows in Fig. 2), assuming that the network protocol design does not allow asynchronous evaluation of the interference. The value $S(t)$ of the SF defines the current mode

$$M_{t_\ell:t_{\ell+1}} = S(t_\ell). \tag{1}$$

We now introduce the SF as a composite mapping $S(t) = Q_\tau(f(t))$, where t is the time expressed in ticks. $Q_\tau(r)$ is a binary quantizing function with domain $r \in [0,1]$, and parameterized by threshold τ:

$$Q_\tau(r) = \begin{cases} \mathfrak{p} & \text{if } r <= \tau \\ \mathfrak{q} & \text{o.w.} \end{cases}$$

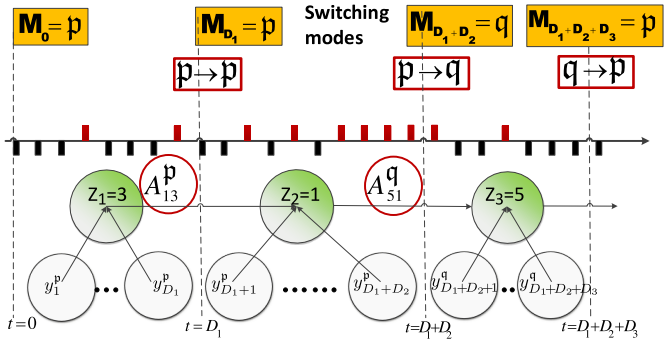

Fig. 3. I/O HSMM state trajectory includes states from both modes according to the SF evaluated over the latest window of BIIs

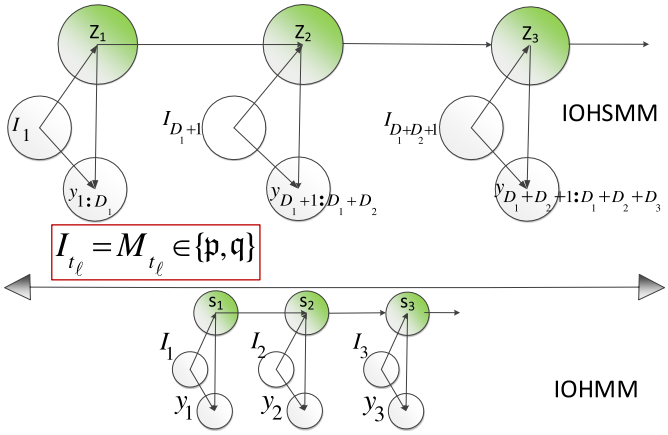

Fig. 4. I/O HSMM vs I/O HMM (bottom)

The threshold τ in our example is based on the average occurrence of ones over *the entire binary sequence*. Now, we monitor the average occurrence of interference indicators *over the most recent window* w of BIIs. The function $f(t)$ calculates the percentage of ones in the window at time t by evaluating BIIs $b(u)$ preceding t :

$$f(t) = \frac{\sum_{u=t-w+1}^{t} b(u)}{w}. \tag{2}$$

The window size is considered fix *apriori* but can be obtained as side information, e.g, through a separate learning mechanism that performs the analysis of the binary sequence prior to the main learning task. Our ultimate goal is to learn jointly w and the HSMM parameters. Fig. 3 presents our model which switches between two modes based on the SF evaluated at the states boundaries.

III. Learning Model

Our model, the *I/O HSMM*, is an extension of the input-output HMM model (IOHMM) [11]. The extension incorporates state durations, following the HSMM paradigm, and the flexibility of non-parametric Bayesian learning [8]. The original IOHMM model [11] extended the HMM by conditioning the latent variables $s(t)$ on additional input time series $I(t)$,

and demonstrated how the *Expectation-Maximization (EM)* and the Viterbi algorithm can be modified to allow the learning of an IOHMM's model and the latent sequence, by using the following joint likelihood function:

$$p(I_{1:T_s}, y_{1:D}, z_{1:T_s}|\Theta) = \quad (3)$$
$$p(z_1|\Theta, I_1) \prod_{t=2}^{t=T_s} p(z_t|z_{t-1}, I_t, \Theta) \prod_{u=1}^{u=D_t} p(y_v|z_t, I_t, \Theta),$$

where Θ are the I/O HSMM parameters, D_t is the number of samples emitted in state z_t and $D = \sum_{t=1}^{T_s} D_t$ is the total number of observed samples. Also, note that I_t denotes input observables, which in our model is equivalent to M_t, (1). The bottom insert in figure 4 presents the IOHMM model, while the centerpiece depicts the version of the input-output HMM that we propose, in which the HMM is replaced by HSMM (note that the first super state of length D_1 is producing D_1 emissions $y_{1:D_1}$ drawn from the same distribution, each conditioned on the same latent variable s_1 and the input I_1. To learn I/O HSMM, we modify an existing non-parametric Bayesian learning model (HDP HSMM [6]) s.t. it reflects the generative model of the cognitive node, i.e., the binary-modulated Markov Chain shown in Fig. 3.

The learning model segments the time series of high-resolution node and interference-specific observations ($\{y_t\}$ and $\{b_t\}$), attempting to recognize the latent sources of those segments as node's activity states and modes. Recall that the mode M_t at time t is based on the last output of the SF $S(t_\ell)$, $t_\ell \leq t$, and hence is a deterministic function of the binary sequence $b(t_\ell)$. Hence, the complete joint likelihood (of modes, observations and states) $p(M, y, s|\Theta)$ has the form given in (3). Since the complete joint likelihood can be intractable, we use Gibbs sampling to iteratively resample our model parameters [6]. Gibbs sampling is a generic MCMC method that relies on knowing only the conditional marginal probabilities of the unknown parameters (as in Fig. 5). When the infinite state-model is approximated by its finite counterpart, our learning model also implements a version of the EM algorithm, i.e., a procedure similar to the Baum-Welch algorithm [12] used to train standard HMMs. In any case, we modify *backward message passing* to incorporate the notion of modes with super-states and their associated durations.

Weak-Limit Gibbs Sampler for the HDP-HSMM

Learning in the I/O HSMM model is carried out via modification of the HDP-HSMM [6] to allow for modulation of the state-models by an external function (M_t). Hence, I/O HSMM is learned via a Bayesian non-parametric generative model defined by the following equations,

$$\begin{aligned} B^{(k)} &\sim GEM(\gamma), \\ \pi^{(k)} &\sim DP(\alpha, B^{(k)}),, \qquad k \in \{\mathfrak{p}, \mathfrak{q}\} \\ (\theta_i^{(k)}, \lambda_i^{(k)}) &\sim (H^{(k)} \times G^{(k)}), \qquad i = 1, 2, \cdots, \\ z_s &\sim \pi^{(k)}(z_{s-1}), \end{aligned}$$

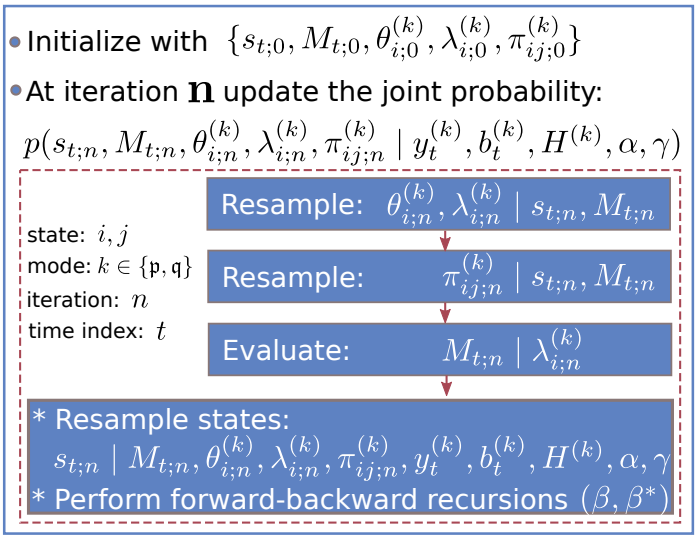

Fig. 5. Gibbs Sampling of I/O HSMM

where GEM denotes the Griffiths-Engen-McCloskey stick-breaking process, which provides a prior for the discrete measure π used in the Dirichlet Process (DP) construction [6]. In our case $\pi^{(k)}(i)$ denotes the probability distribution of transitions from state i to any other state in mode k (equivalent to a row of the transition probability matrix in finite Markov models), and all $\pi(\bullet)$ have the same prior B. We denote observation and duration prior distributions with $H^{(k)}$ and $G^{(k)}$ respectively. The superscript (k) throughout all equations is used to distinguish the mode.

We use the weak-limit sampler of the HDP-HSMM from [6], which constructs a finite approximation (with finite K-dimensional Dirichlet distributions) to the HDP transitions prior. This allows block sampling of the entire state label sequence at once, resulting in greatly accelerated mixing of the Markov chain, i.e., faster learning. Please see [6] for a discussion on the justification for this approximation, and for further references on how the approximation becomes exact as K grows. A common approach to sampling Dirichlet process based mixture models is a marginalization known as the Chinese Restaurant Process which gives an elegant analogy of the incremental sampling of an unknown number of components. Chinese Restaurant Franchise (CRF) represents its extension to state-based models (such as HDP HMM) [8]. Hence, we could use the CRF to converge to the most likely number of states and other I/O HSMM parameters, but the weak limit approximation is computationally more appropriate for our model which includes not only state but mode transitions too. In the next subsection we redefine backward messages used by the blocked Gibbs sampler in [6] s.t. mode variables be included in the forward-backward recursions of our Gibbs sampling algorithm (Fig. 5).

Backwards Messages

In the I/O HSMM, similar to [13], backward messages are split into β and β^* components for convenience. Note that s_t in the definitions of β and β^* denotes the state, not the super-state. As the input to SF is random, so is its output,

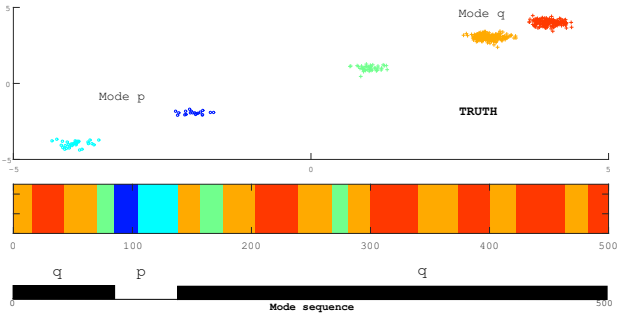

Fig. 6. Our GT example with the sequence of colored bars in the middle representing the sequence of state labels (from both modes) $s_1 : s_{500}$, corresponding to observation colors in the top pane, and the mode sequence for the first 500 samples at the bottom

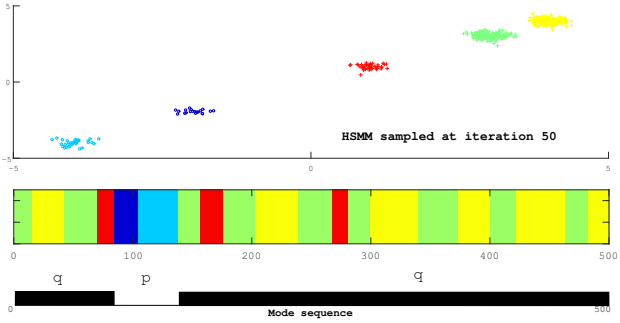

Fig. 7. Our CR toy model: learned state and mode trajectories for the first 500 samples (different color code, same meaning as in the GT)

hence the capital letter notation for the mode M_t, while m_t is the observation of M_t. The expressions are independent of the SF definition. Conditional probabilities are based on the assumption that the new super-state begins at time $t+1$, which is denoted by $(F_t = 1)$, following the notation from [13]

$$\begin{aligned}\beta_t(i) &= p(y_{t+1:T}|s_t = i, M_{t+1:T} = m_{t+1:T}, F_t = 1) = \\ &= \sum_j p(y_{t+1:T}|s_{t+1} = j, M_{t+1:T} = m_{t+1:T}, F_t = 1) \\ &\quad p(s_{t+1} = j|s_t = i, M = m_{t+1}) \\ &= \sum_j \beta_t^*(j) p(s_{t+1} = j, M = m_{t+1}|s_t = i) \\ &= \sum_j \beta_t^*(j) A_{ij}^{m_{t+1}} \end{aligned} \tag{4}$$

$$\begin{aligned}\beta_t^*(i) &= p(y_{t+1:T}|s_{t+1} = i, M_{t+1:T} = m_{t+1:T}, F_t = 1) = \\ &= \sum_{d=1}^{T-t} p(D_{t+1} = d|s_{t+1} = i, M = m_{t+1}) \\ &\quad p(y_{t+1:t+d}|s_{t+1} = i, M = m_{t+1}) \beta_{t+d}(i) + C_t(i), \end{aligned} \tag{5}$$

where $C_t(i)$, as in [14], is the contribution of super-states that run off the end of the provided observations (censored), which depends on the survival function of the duration distribution evaluated for the current mode $M = m_{t+1}$

$$\begin{aligned} C_t(i) &= p(y_{t+1:T}|s_{t+1} = i, M = m_{t+1}, D_{t+1} > T - t) \\ &\quad p(D_{t+1} > T - t|s_{t+1} = i, M = m_{t+1}). \end{aligned} \tag{6}$$

Finally, $\beta_T(i) = 1$.

IV. Simulation Results

For the convenience of visualization, in our simulations we use two-dimensional Gaussian observations instead of one-dimensional Gaussian samples (representing received dB power). The ground truth system switches between 2 behavioral modes, each with 4 protocol states, distinguished by Gaussian parameters in the following way: mode $\mathfrak{p}$'s states have emission means $\mu_i = (-i, -i),\ i \in \{1, \cdots, 4\}$ and equal standard deviations, while mode $\mathfrak{q}$'s states have emission means $\mu_i = (i, i),\ i \in \{1, \cdots, 4\}$. We placed mean values of the alternative modes in two different quadrants for better visual perception (Fig. 8). Figures 6 and 7 demonstrate visually the simulated GT system, and the learned system after resampling the I/O HSMM model 50 times, with 2000 samples observed. Note that only 2 states appear in the state trajectory of mode $\mathfrak{p}$, and 3 states in the state trajectory of mode $\mathfrak{q}$, since the sequence of synthesized samples is too short to capture all system states. Note that state labeling, and hence, color coding does not carry any meaning. However, based on associated observations, we see that the red cluster in true mode $\mathfrak{q}$ (top 3 clusters in Fig. 6) is colored yellow in the learned system. Hence, if the learning is successful, the sequence of colored bars in the middle of Fig. 7, depicting the time series of states, will have the yellow bars where the red bars are in Fig. 6. With this in mind, visual similarity is obvious, suggesting successful learning. Also, note the black mask applied in the bottom pane of figures 6 and 7 whenever the SF was evaluated as $\mathfrak{q}$.

We must point out that clusters do not overlap, making the learning task easier. We tested our learning algorithm on completely random, and on bursty BIIs with different burstiness patterns. The *quality of learning (QoL)* was similar, in the sense that convergence to the GT happened in most cases, and the learning was better with more samples. To quantify QoL, we introduce the following metrics

- **Hamming distance between GT and learned model (LM) state sequence (per mode)**
- **Frobenius norm of the difference matrix between GT and LM transition matrices (per mode)**.

We also calculated Kullback-Leibler (KL) distances both between state emission distributions and state duration distributions (with respect to GT), and all those metrics exhibited good convergence, indicting behaviors matching the GT. We do not present how KL distances converge with iterations since there are too many of them, and since the convergence rate is similar to Hamming distance convergence.

Fig. 9 shows how the relative Hamming distance (by mode) depends on the learning iteration for another example (the case of 20K samples observed from the same GT model, presented in Fig. 8) and three different input sequence types. The top plots are for the random binary input sequence with probability of one $p_1 = 0.3$. The middle plots are for alternating bursts of

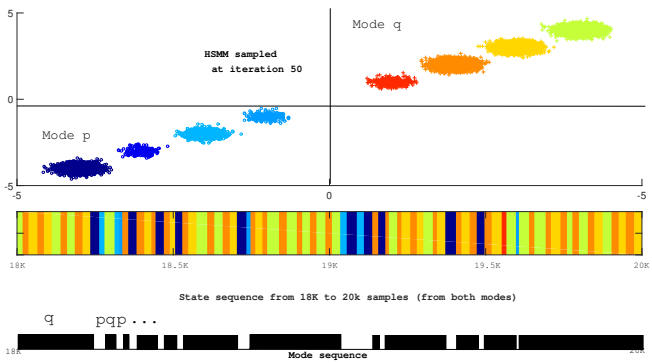

Fig. 8. Learned Model 20K (close up at the last 2K samples)

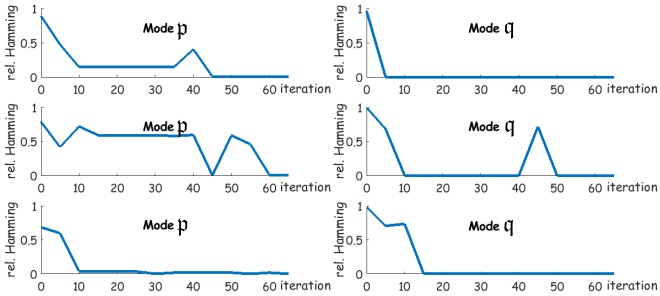

Fig. 9. Hamming Dist. for the 20K samples over 65 iterations(top: uniform BII sequence with $p_1 = 0.3$; middle: alternating bursts of 0s and 1s of random length up to 20; bottom: random bursts of 0s and bursts of 20 1s

zeros and ones of random length up to 20, and the bottom plots are for random burst of zeros (up to 20) followed by burst of 20 ones. When the sequence is order of magnitude shorter (like in Fig. 7) this QoL metric visibly deteriorates since the system is non-stationary, and small number of samples does not capture complete state and mode dynamics. The same stands for the Frobenius metric (Fig. 8). With shorter sequences, QoL metrics still indicate learning improvement with more iterations, up to a saturation point, which is random. However, we only present metrics calculated at 20K samples, due to space limit. As a practical guideline, we suggest increasing the number of samples until QoL metric start converging with respect to the previous posterior model.

V. Conclusions

We proposed and demonstrated a model which combines non-parametric Bayesian approach of the HDP HSMM [6] with the generative structure of the IO HMM [11], to semi-parametrically learn behavioral dynamics of a cognitive RF device communicating within a non-specific wireless ad-hoc network, and whose unknown state machine is bi-modal, based on a switching function (SF) that reacts to interference. The model was also successfully tested on multinomial features representing packet lengths, which underlines the value of non-parametric learning in addressing multiple time scales in CR dynamics.

Although the knowledge of the switching criterion suggests that our model is oracle-aided, note that prior statistical analysis of the sequence of BIIs could be used to learn the SF.

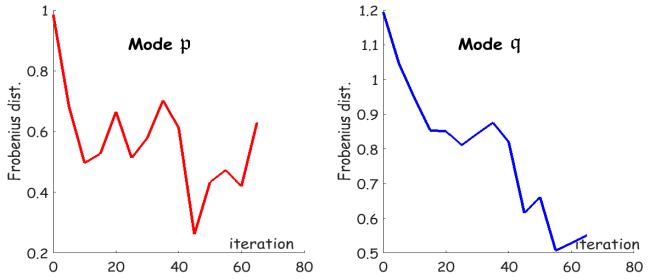

Fig. 10. Frobenius distance between GT and LT transition matrices vs learning iteration, for the BIIs as in Fig 9 bottom, and 20K samples

Moreover, joint learning of the SF and the node's dynamics is the natural extension of this work. In [3] the learning target is the SF itself, representing a PU's activity which cognitive node is trying to estimate or predict, based on a bivariate HMM and online learning setup. An interesting prior work to point out to is the paper [15] in which human activity learning is conducted using a model dubbed *switching HSMM*, which is a hierarchical Hidden Markov model with the lower-level states (atomic activities) being modeled by an HSMM. Even though this model does not have an input sequence, it motivates an extension to our model in which the modulating influence of the environment could be cast as another HMM hierarchy.

References

[1] J. Mitola and J. G.Q. Maguire, "Cognitive radio: making software radios more personal," *IEEE Personal Communications*, vol. 6, no. 4, Aug 1999.

[2] Y. Zhang, R. Yu, M. Nekovee, Y. Liu, S. Xie, and S. Gjessing, "Cognitive machine-to-machine communications: visions and potentials for the smart grid," *IEEE Network*, vol. 26, no. 3, pp. 6–13, 2012.

[3] S. Y. S., B. Mark, and Y. Ephraim, "Online parameter estimation for temporal spectrum sensing," *IEEE Trans. on Wireless Comms,*, vol. 14, no. 8, pp. 4105–4114, Aug 2015.

[4] J. Goodman, K. Rudd, and T. Clancy, "Blind multiuser localization in cognitive radio networks," *IEEE Communications Letters*, vol. 16, no. 7, p. 1018âĂŞ1021, 2012.

[5] M. Bkassiny, Y. Li, and S. Jayaweera, "A Survey on Machine-Learning Techniques in Cognitive Radios," *IEEE Communications Surveys Tutorials*, vol. 15, no. 3, 2013.

[6] M. J. Johnson and A. S. Willsky, "Bayesian Nonparametric Hidden Semi-Markov Models," *Journal of Machine Learning Research*, vol. 14, pp. 673–701, Feb 2013.

[7] S. Yu, "Hidden semi-markov models," *Artificial Intelligence*, vol. 174, no. 2, 2010.

[8] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei, "Hierarchical Dirichlet processes," *Journal of the American Statistical Association*, vol. 101, no. 476, 2006.

[9] L. R. Rabiner and B. H. Juang, "An introduction to hidden markov models," *IEEE ASSp Magazine*, 1986.

[10] S.-L. Wu, Y.-C. Tseng, and J.-P. Sheu, "Intelligent medium access for mobile ad hoc networks with busy tones and power control," *IEEE Journal on Sel. Areas in Communications*, vol. 18, no. 9, pp. 1647–1657, 2000.

[11] Y. Bengio and P. Frasconi, "An Input Output HMM Architecture," in *Neural Information Processing Systems*, 1994.

[12] S. E. Levinson, L. Rabiner, and M. Sondhi, "An introduction to the application of the theory of probabilistic functions of a Markov process to automatic speech recognition," *Bell System Tech. Jour.*, vol. 64, 1983.

[13] K. Murphy, "Hidden Semi-Markov models (segment models)," UC Berkeley, Tech. Rep., Nov. 2002.

[14] Y. Guedon, "Exploring the state sequence space for Hidden Markov and semi-Markov chains," *Comput. Statistics and Data Analysis*, 2007.

[15] T. Duong, H. Bui, D. Phung, and S. Venkatesh, "Activity recognition and abnormality detection with the switching hidden semi-Markov model," in *IEEE Computer Soc. Conf. on Computer Vision and Pattern Recognition*, June 2005.