Latent Variable Models for Hippocampal Sequence Analysis

Etienne Ackermann and Caleb Kemere
Department of Electrical and Computer Engineering
Rice University, Houston, TX
Email: http://era3.web.rice.edu/contact

Kourosh Maboudi and Kamran Diba
Department of Anesthesiology
University of Michigan, Ann Arbor, MI

***Abstract*—The activity of ensembles of neurons within the hippocampus is thought to enable memory formation, storage, recall, and potentially decision making. During offline states (associated with sharp wave ripples, quiescence, or sleep), some of these neurons are reactivated in temporally-ordered sequences which are thought to enable associations across time and episodic memories spanning longer periods. However, analyzing these sequences of neural activity remains challenging. Here we build on recent approaches using latent variable models for hippocampal population codes, to detect so-called "replay events", and to build models of hippocampal sequences independent of animal behavior. We demonstrate that our approach can identify the same replay events as traditional Bayesian decoding approaches, and moreover, that it can detect nonlinear remote replay events that are difficult or impossible to detect with existing approaches.**

***Keywords*—Neural signal processing, hidden Markov models, replay, sequence analysis**

I. Introduction

The activity of ensembles of neurons within the hippocampus is thought to enable memory formation, storage, recall, and potentially decision making. In rodents, hippocampal "place cells" are known to encode an animal's location in its environment as it explores [1]. Hence, populations of these neurons fire in temporally-ordered sequences corresponding to the spatiotemporal trajectories the animals traverse. Of particular interest to us are hippocampal *replay events* in which neurons recapitulate their spatially-ordered sequences during periods of quiescence or sleep (and often associated with brief, 150–250 Hz sharp wave ripple (SWR) oscillations in the hippocampus).

Most approaches to replay detection rely on the estimation of behavioral templates during active behavior, followed by comparisons of the SWR-associated replay canditate events to the learned behavioral templates. Such an approach is critically dependent on (i) the availability of behavioral data, as well as (ii) the associated task complexity.

The requirement to have access to the behavioral correlates have caused studies of hippocampal replay to be limited to those where the behavioral correlates are well understood and easily observable (most notably that of position); it is much more difficult to identify sequences of non-spatial memories e.g., sequences of odor cues. In addition, template matching approaches quickly become prohibitive even for relatively simple tasks, and therefore more powerful, generalizable approaches have to be considered (see e.g. [2] for an alternative approach to template matching, where heuristic rules such as a maximum jump distance from one frame to the next, and a minimum end-to-end distace, were used in an open field).

Hidden Markov models (HMMs) are well suited to model this sort of sequential activity—due in part to the Markovian nature of spatial locomotion (e.g., our position at time t is constrained by our position at time $t-1$), but perhaps more importantly is the fact that, by definition, there is no observable animal behavior during candidate replay events.

Indeed, HMMs have found several uses in neuroscience (see for example [3]), and have also been used to uncover hippocampal population codes during awake [4], as well as during sleep-associated activity [5]. Here, we build on these recent approaches using latent variable models (and HMMs in particular), to detect remote replay events in a moderately complex environment/task (a continuous alternation task in a w-shaped maze [6]).

We demonstrate that our approach can identify many of the same replay events as traditional Bayesian decoding approaches, and furthermore, that it can detect nonlinear remote replay events that are difficult or impossible to detect with existing approaches. Moreover, our approach is not dependent on the underlying task complexity, and does not require (or use) the associated behavioral data, laying the groundwork for the study of non-spatial memory.

In Section II, we briefly introduce the experimental paradigm, along with the regression-based approach for remote replay detection used by [6]. Subsequently, we present our alternative HMM-based approach in Section III.

II. Awake replay of remote experiences

To showcase the use of HMMs in neural data analysis, we consider the experiment presented in [6], where it was shown that rats exhibit robust replay of remote experiences during awake periods[1].

Briefly, ensembles of principle neurons were recorded from hippocampal areas CA3 and CA1 while three rats were exposed to two w-shaped environments (Fig. 1 E1 and E2), in which the rats were rewarded at the endpoint of each arm when correctly performing a continuous alternation task. Rats were exposed to E2 for several days before the first exposure to E1, so that E2 was always more familiar to the animals than E1.

[1]The data is publically available from crcns.org [7].

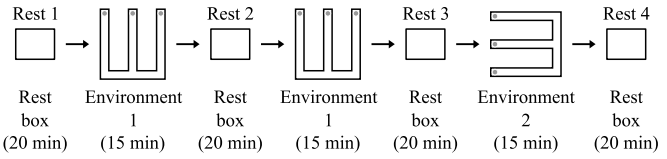


Fig. 1: Remote replay task illustration. Each day of the experiment, rats were eposed to a series of contexts, including two distinct w-shaped mazes, Environment 1 (E1) and Environment 2 (E2). Remote replay was defined as robust representations of E1 while the animal was running in E2. Adapted from [6].

Remote replay (more specifically, awake remote replay) was then defined as robust representations (during SWRs) of E1 while the animal was in E2, or robust representations of either E1 or E2 when the animal was awake and in the rest box. We will only consider remote replay of E1 while the animal was in E2 here.

A. *Regression analysis for remote replay detection*

Following [6], candidate remote replay events were identified as SWRs (recorded while the animal was in E2) during which at least 5 neurons that had place fields in E1 were active. Candidate events were then divided into 15 ms bins, and a Bayesian decoder with a uniform prior was used to decode the ensemble neural activity to distributions over positions in environment E1.

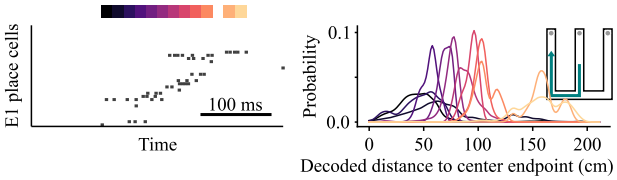
Critically, the position was first linearized in one of two ways: (i) for events where the trajectory went to or from the center arm, linearized position was defined as the distance from the reward well on the center arm, and (ii) for events that had trajectories that went from one outer arm to the other, linearized position was defined as the distance from the upper left reward well.

Candidate events were then scored by determining the R^2 value from a regression of [decoded, linearized] position over time, compared to 10,000 regressions on surrogate events where the order of the time bins were randomly permuted. A P value for each event was then calculated as the proportion of shuffled R^2 values greater than the actual R^2 value, and an event was considered significant when $P < 0.05$.

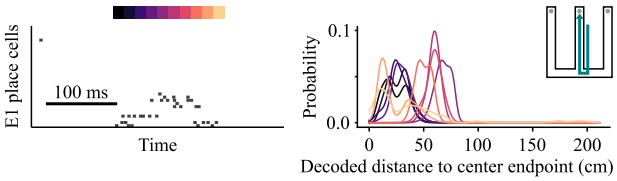
Fig. 2 shows four example candidate remote replay events, along with the spike rasters for all the place cells in E1, and the decoded distributions over linearized position for each time bin. Time bins with no spikes cannot be meaningfully decoded to position, and were not included in the regression analysis. Fig. 2.a and 2.c show clear linear movement through the environment, and both were correctly identified as remote replay by the regression analysis.

B. *Limitations of regression analysis*

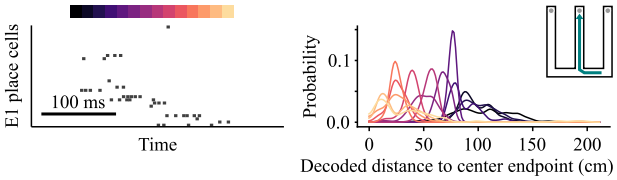
Even though this regression-based analysis is effective for finding examples of remote replay in the w-maze task, it critically relies on (i) the linearization of position, and on (ii) the position data being available. It is non-trivial, for example, to extend this analysis to less constrained or more complex environments (such as open fields), or to other behavioral



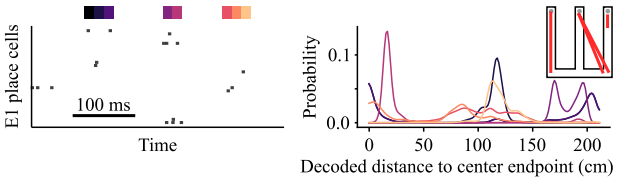
(a) Linearized remote replay candidate 1.



(b) Linearized remote replay candidate 2.



(c) Linearized remote replay candidate 3.



(d) Linearized remote replay candidate 4.

Fig. 2: (a) and (c) Examples of remote replay candidate events that were correctly classified as [significant] remote replay by the regression-based analysis. (d) An example of a candidate remote replay event that was correctly classified as non-significant. (b) An example of a remote replay event that was misclassified as nonsignificant because it exhibited nonlinear trajectory behavior. In each panel, the spike raster is shown on the left, with the decoded linearized position distribution for each time bin shown on the right, along with a cartoon representation of the decoded trajectory in the w-maze.

correlates that are not easily observable (such as sequences of odor cues, or other sequences of episodic memory).

It is with these more complex tasks in mind that we set out to develop an approach that (i) generalizes trivially to more complex environments and tasks, and (ii) that can work in the absence of observable behavioral correlates. We demonstrate that the HMM can be used to achieve both of these goals.

III. Hidden Markov Models of Neural Activity

HMMs are statistical models where the systems being modeled are Markov chains (or more generally Markov processes) with unobserved or latent states. HMMs have been widely used for sequential pattern recognition and processing in fields ranging from speech recognition to bioinformatics (see [8] for an excellent tutorial introduction), and have also been used to model neural activity in a variety of contexts [4], [9]–[14].

In this section we loosely follow the approach and notation presented in [4], [14].

A. Model specification

Let $\mathbf{y}_t$ denote the *observation* at time t, where $\mathbf{y}_t \in \mathbb{Z}^C$ is a vector of spike counts for C hippocampal pyramidal cells. It is assumed that the observations are sampled at discrete, equally-spaced time intervals, so that t can be an integer-valued index, with some associated Δt.

We further assume that the hidden state space is *discrete*. That is, $S_t \in \{1, \ldots, m\}$ can take on one of m possible states. In our spatial navigation context, each possible state can be thought of as loosely corresponding to a particular location in the environment.

To define a probability distribution over sequences of observations, we then need to specify a probability distribution over the initial state $P(S_1)$, with $\boldsymbol{\pi}_i \equiv \Pr(S_1 = i)$, the $m \times m$ state transition probability matrix, $\mathbf{A}$, with $\mathbf{A}_{ij}$ defining $P(S_t = j | S_{t-1} = i)$, and the output or emissions model defining $P(\mathbf{y}_t | S_t)$. We have found that a Poisson emissions model worked well.

More specifically, we assume Poisson firing statistics for each spike train, so that the emission probability for the ith state is modeled by a spatially varying (state-dependent) multivariate Poisson process:

$$\begin{aligned} P(\mathbf{y}_t | S_t = i; \theta) &= \prod_{c=1}^{C} P(y_{c,t} | S_t = i; \theta) \\ &= \prod_{c=1}^{C} \prod_{j=1}^{m} P(y_{c,t} | S_t = j; \theta)^{S_{t,i}} \\ &= \prod_{c=1}^{C} \prod_{j=1}^{m} \left(\frac{\exp(-\lambda_{jc}) \lambda_{jc}^{y_{c,t}})}{y_{c,t}!} \right)^{S_{t,i}} \end{aligned}$$

where $\theta = \{\boldsymbol{\pi}, \mathbf{A}, \boldsymbol{\Lambda}\}$ are the model parameters, $\boldsymbol{\Lambda} \in \mathbb{R}^{m \times C}$ are the tuning curve parameters (a spike firing rate λ for every possible state $j \in \{1 \ldots m\}$ for each pyramidal cell $c \in \{1, \ldots C\}$), and $S_{t,i} = 1$ iff $S_t = i$, and 0 otherwise.

Finally, we assume that our model is time-invariant: that is, we assume that the state transition probability matrix and the output model do not change over time.

Given a training set $\mathcal{D} = \{\mathbf{y}_{1:T_1}^{(1)}, \ldots, \mathbf{y}_{1:T_N}^{(N)}\}$, containing N sequences of observations, and since the training sequences are assumed to have been drawn independently, the complete data likelihood takes the form

$$P(\mathcal{D}, \mathbf{S} | \theta) = \prod_{n=1}^{N} P\left(\mathbf{y}_{1:T_n}^{(n)} | \theta, S_{1:T_n}^{(n)}\right) P\left(S_{1:T_n}^{(n)}\right). \quad (1)$$

The model parameters $\theta = \{\boldsymbol{\pi}, \mathbf{A}, \boldsymbol{\Lambda}\}$ can then be estimated using standard methods such as expectation maximization, variational Bayes, or Monte Carlo methods. We have used the standard iterative expectation-maximization (EM) algorithm [8] to learn the parameters in our models.

From here, we can use computationally efficient algorithms (such as the "forward-backward algorithm" [8]) to compute a "score" for any observation sequence $P(\mathbf{y}_{t=1:T})$, as well as to decode neural activity to the underlying state space.

B. Hidden Markov models for remote replay detection

We are primarily interested in determining when short bursts of neural activity during SWRs encode remote experiences or environments. To this end, we searched for evidence of sequences corresponding to environment 1 (E1) while the animal was running in environment 2 (E2), as reported in [6]. For each animal, and for each experiment day, we learned two HMMs (one for each environment), using bouts of run activity (animal speed > 3 cm/s), while excluding any SWRs. We subsequently scored all the candidate remote replay events recorded in E2, in the HMM corresponding to E1. Scoring SWR-associated sequences from E2 in the HMM from E1 allows us to determine if the sequences are consistent with those observed during active behavior in E1.

More specifically, we learned the HMMs using a time bin size of $\Delta t = 125$ ms, which is (i) short enough to capture the behavioral dynamics of the animals while running, and moreover (ii) it captures a full theta cycle (≈ 8 Hz in rodents). We arbitrarily chose $m = 30$ states for our models, but we have previously shown that the analyses are remarkably insensitive to the actual choice of the number of states [15].

Similar to the regression analysis performed in [6], we employed a time swap shuffle to determine congruence with our HMMs. In particular, for each candidate event, we synthesized $10,000$ surrogate events by randomly permuting the observation time bins, and scoring each surrogate event in the E1-HMM to form a shuffle distribution for that event. The P value for the event was then defined as the proportion of the shuffled scores that was greater than the score of the actual event. (see e.g., 4 for some example shuffle ditributions and the scores of the actual events). Significant events were defined as those that had $P < 0.05$, as in [6].

Note that the behavioral-timescale HMMs were learned with $\Delta t = 125$ ms, but the candidate SWR-associated events were binned into $\Delta t = 15$ ms bins, as in [6]. For all the results presented here, we did not apply any scaling to either the events, nor the model, when scoring the candidate events. Moreover, we found that the results were largely insensitive to any scaling that we did try (results not shown).

IV. Results

We performed both the regression-based analysis from [6] as well as our HMM-based analysis for all animals, and all experiment days (see Table II), but we show results for only a single representative session (rat 2, day 3) here.

A. HMMs capture the positional code

Since our HMMs are learned during running behavior, it is natural to expect that the latent states should somehow encode position in their abstract representations. To better understand the relationship between the latent state space and physical space, we used the latent state trajectories decoded during running with their corresponding positions to form an estimate of the likelihood as a function of location on the maze. These latent-state place fields (see Fig. 3) in many ways resembled neuronal place fields and similarly tiled the extent of the w-maze.

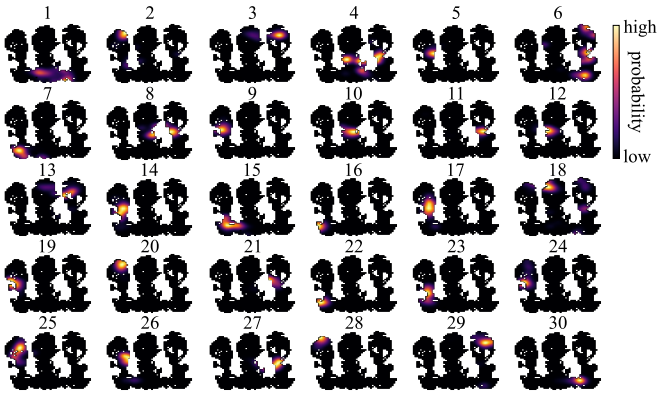


Fig. 3: Latent space place fields. We decoded neural activity during running behavior to the state space, and learned a mapping from the state space to physical position. The likelihood of the animal's position is shown for each of the $m = 30$ states in our HMM from E1.

B. HMMs can identify examples of remote replay

Following [6], we identified SWR-associated candidate events for remote replay in E2, and scored all of these events in our E1-HMMs. In our example session, and using the signficicance threshold of $P < 0.05$, we identified 26 out of 36 candidate events as significant remote replay (72%), in comparison to 23 out of 36 (64%) using the regression-based analysis (see Table I). Overall, there is a 69% agreement between the two approaches (19 + 6 out of 36).

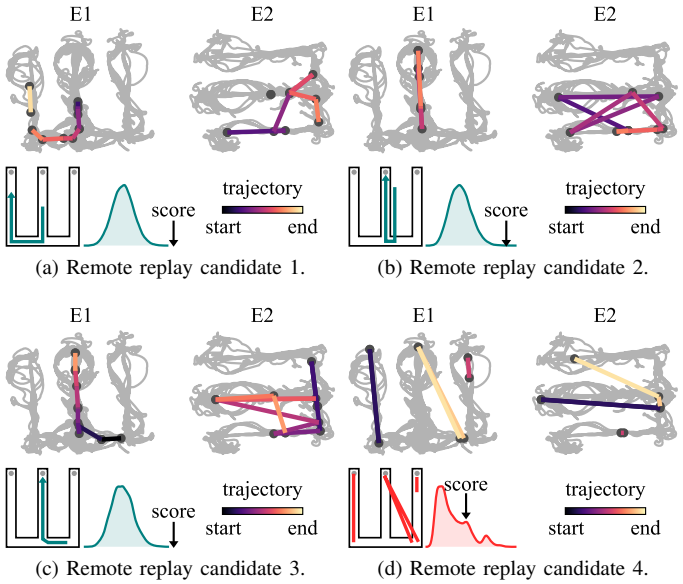
TABLE I: Comparison between the number of significant remote replay events obtained using regression-based and our HMM-based approaches. Here, R^2 denotes the regression-based analysis, '+' denotes significant events, and '-' denotes nonsignificant events.

$N = 36$	R^2 +	R^2 −	
HMM +	19	7	26
HMM −	4	6	10
	23	13	

C. HMMs can identify nonlinear examples of remote replay

Closer inspection of the remote replay events reveals some interesting differences between the two approaches. In particular, the regression-based analysis requires the position on the w-maze to be linearized (which introduces a position ambiguity in one of the arms), and then requires remote replay events to be linear traversals in this linearized space. However, using our HMM approach, which is learned completely independent of the position data, we do not need to concern ourselves with the linearization, and moreover, we can identify nonlinear remote replay events where the trajectory might back-track or exhibit some other interesting-but-consistent behavior.

One example of such a nonlinear remote replay event that was correctly classified by the HMM-based approach (and misclassified by the regression analysis) is shown in Fig. 4.b. Fig. 4 also shows three other examples of remote replay candidates, two of which are correctly classified as significant



(a) Remote replay candidate 1. (b) Remote replay candidate 2.

(c) Remote replay candidate 3. (d) Remote replay candidate 4.

Fig. 4: (a)–(c) Examples of remote replay candidate events that were correctly classified as [significant] remote replay by the HMM. (d) An example of a candidate remote replay event that was correctly classified as non-significant by the HMM. E1 and E2 denote environments one and two, respectively. In each panel, the top row shows the decoded trajectories (using place fields and a Bayesian decoder) in each environment, while the bottom row shows a cartoon representation of the trajectory in E1 (left) and the shuffle distribution obtained using the HMM for the trajectory event (right), along with the event score indicated by the arrow. (Note that the four example events are the same ones as shown earlier in Fig. 2.)

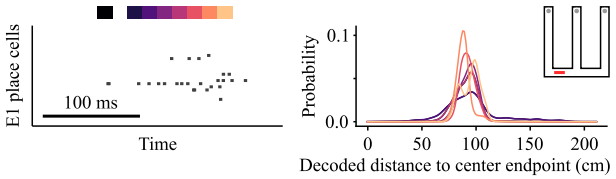


Fig. 5: Example of a remote replay candidate that was classified as significant by the regression analysis, and nonsignificant by the HMM approach. Notice that the trajectory is rather short, making it unclear whether this is, in fact, a true example of remote replay or not.

remote replay (Fig. 4.a and c) and one that was correctly classified as nonsignificant (Fig. 4.d). Notice that in all the examples shown, the candidate events are nonsensical in the current (local) environment, E2.

The HMM also seemed to have slightly fewer false positives [than the regression-based analysis], or at least to be more selective in the session considered here. For example, Fig. 5 shows an example event that was classified as significant by the regression analysis, while being rejected by the HMM. However, a comprehensive analysis of classification accuracy (which would necessarily have to rely on subjective determinations by human scorers) has not been carried out yet.

D. Aggregated results

Results pooled across all animals, and all experiment days, are summarized in Table II. In particular, we found 188 out of 623 (30%) significant remote replay events using our HMM approach, and 270 out of 623 (43%) using the regression analysis. The overall agreement between the two approaches is therefore 69% (131 + 296 out of 623).

TABLE II: Comparison between the number of significant remote replay events obtained using regression-based and our HMM-based approaches for all sessions and animals combined. Overall agreement is 69% ($P < 0.001$, Fisher's exact test, two-tailed).

$N = 623$	R^2+	R^2-	
HMM +	131	57	188
HMM −	139	296	435
	270	353	

V. Discussion

It is encouraging to see the large agreement ($\approx$ 70%) between the regression and HMM approaches for the identification of remote replay, especially considering that the HMM identified additional examples of nonlinear remote replay.

We have only considered remote replay of E1 while the animal was behaving in E2 here, but it is reasonable to expect that we may obtain similar results when looking for remote replay while the animal is in the rest box, as reported in [6].

It is worth emphasizing that even though we have learned HMMs on neural data during run bouts in both E1 and E2, we did not use any of the *behavioral* data (i.e., position) in our models. Thus, in contrast to the approach presented in [6], our HMM-based approach can be used in the absence of behavioral data or correlates.

Indeed, we have recently shown that, given sufficient training data during SWRs, we can learn HMMs on the SWR-associated events directly, and still recover sufficient information about the associated behavior [15]. In this way, it might be possible to find instances of remote replay even without recording the neural data (let alone the behavioral correlates) during the remote experience.

VI. Conclusion

We have demonstrated how the HMM be be used to detect remote replay without appealing to any behavioral correlates, which in contrast to existing approaches, means that our approach is largely independent of task complexity. Indeed, this flexibility makes it possible to analyze a whole new class of behaviors beyond spatial memory. Indeed, the HMM framework presents a powerful and attractive approach to analyze a variety of sequential tasks and phenomena in the hippocampus and beyond.

Acknowledgment

The authors would like to thank Mattias Karlsson and Loren M. Frank for making the data available on CRCNS.org.

References

[1] J. O'Keefe and J. Dostrovsky, "The hippocampus as a spatial map. preliminary evidence from unit activity in the freely-moving rat," *Brain research*, vol. 34, no. 1, pp. 171–175, 1971.

[2] B. E. Pfeiffer and D. J. Foster, "Hippocampal place-cell sequences depict future paths to remembered goals," *Nature*, vol. 497, no. 7447, pp. 74–79, 2013.

[3] B. Florian, K. Sepp, H. Joshua, and H. Richard, "Hidden markov models in the neurosciences," in *Hidden Markov Models, Theory and Applications*. InTech, 2011.

[4] S. W. Linderman, M. J. Johnson, M. A. Wilson, and Z. Chen, "A bayesian nonparametric approach for uncovering rat hippocampal population codes during spatial navigation," *Journal of neuroscience methods*, vol. 263, pp. 36–47, 2016.

[5] Z. Chen, A. D. Grosmark, H. Penagos, and M. A. Wilson, "Uncovering representations of sleep-associated hippocampal ensemble spike activity." *Scientific reports*, vol. 6, no. August, p. 32193, 2016. [Online]. Available: http://dx.doi.org/10.1038/srep32193

[6] M. P. Karlsson and L. M. Frank, "Awake replay of remote experiences in the hippocampus." *Nature neuroscience*, vol. 12, no. 7, pp. 913–8, jul 2009.

[7] M. Karlsson, M. Carr, and L. M. Frank, "Simultaneous extracellular recordings from hippocampal areas CA1 and CA3 (or MEC and CA1) from rats performing an alternation task in two W-shapped tracks that are geometrically identically but visually distinct. CRCNS.org." 2015, http://dx.doi.org/10.6080/K0NK3BZJ.

[8] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, 1989.

[9] Z. Chen and M. A. Wilson, "Deciphering neural codes of memory during sleep," *Trends in Neurosciences*, 2017.

[10] J. Deppisch, K. Pawelzik, and T. Geisel, "Uncovering the synchronization dynamics from correlated neuronal activity quantifies assembly formation," *Biological cybernetics*, vol. 71, no. 5, pp. 387–399, 1994.

[11] C. Kemere, G. Santhanam, M. Y. Byron, A. Afshar, S. I. Ryu, T. H. Meng, and K. V. Shenoy, "Detecting neural-state transitions using hidden markov models for motor cortical prostheses," *Journal of neurophysiology*, vol. 100, no. 4, pp. 2441–2452, 2008.

[12] G. Radons, J. Becker, B. Dülfer, and J. Krüger, "Analysis, classification, and coding of multielectrode spike trains with hidden markov models," *Biological cybernetics*, vol. 71, no. 4, pp. 359–373, 1994.

[13] W. Wu, J. E. Kulkarni, N. G. Hatsopoulos, and L. Paninski, "Neural decoding of hand motion using a linear state-space model with hidden states," *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, vol. 17, no. 4, pp. 370–378, 2009.

[14] E. Ackermann and C. Kemere, "Scoring sequences of hippocampal activity using hidden markov models," in *Engineering in Medicine and Biology Society (EMBC), 2016 IEEE 38th Annual International Conference of the*. IEEE, 2016, pp. 957–960.

[15] K. Maboudi, E. Ackermann, K. Diba, and C. Kemere, "Uncovering temporal structure in hippocampal output patterns," unpublished.