A Non-Parametric Bayesian Approach to Spike Sorting

Frank Wood[†] Sharon Goldwater[‡] Michael J. Black[†]

[†]Department of Computer Science, [‡]Department of Cognitive and Linguistic Sciences
Brown University, Providence, RI, USA

***Abstract*— In this work we present and apply infinite Gaussian mixture modeling, a non-parametric Bayesian method, to the problem of spike sorting. As this approach is Bayesian, it allows us to integrate prior knowledge about the problem in a principled way. Because it is non-parametric we are able to avoid model selection, a difficult problem that most current spike sorting methods do not address. We compare this approach to using penalized log likelihood to select the best from multiple finite mixture models trained by expectation maximization. We show favorable offline sorting results on real data and discuss ways to extend our model to online applications.**

***Index Terms*— Spike sorting, mixture modeling, infinite mixture model, non-parametric Bayesian modeling, Chinese restaurant process, Bayesian inference, Markov chain Monte Carlo, expectation maximization, Gibbs sampling.**

I. Introduction

Spike sorting is a collection of processes whose purpose is to both distinguish between multiple cells recorded together and to assign detected spiking activity to the neuron responsible for generating it [1]–[6]. This definition presumes that spikes have been detected a priori (disambiguated from background noise) by some other process. This is a narrow definition of spike sorting, and one that admittedly may not accurately describe more sophisticated approaches that combine the two steps intelligently [2], [7]. We further restrict our focus to a single channel of data, disregarding the valuable correlation and spatial information provided by closely spaced electrodes. We adopt this definition and viewpoint to highlight the model identification problem.

A central problem in spike sorting is determining how many neurons are present in a recording. This problem can be viewed more generally as a problem of statistical model identification if we adopt a mixture modeling approach, as in [3], [5], [6]. In mixture modeling the distribution of a population (all the spikes from a channel) is formulated as the additive combination of a number of hidden subpopulations, or classes (the spikes from each individual cell). Learning a parametric mixture model consists of determining the parameters of the classes as well as the proportion of each class in the full population. Traditional approaches such as expectation maximization (EM) [8] provide no explicit solution to the problem of how many classes there are (how many cells). The typical approach for determining that number is to learn many different models, one or more each for each choice of dimensionality (number of classes). Then a penalty such as the Bayesian information criteria (BIC) or the Aikaike information criteria is used to select the best model corresponding to the best penalized log-likelihood of held-out data.

This work was supported by NIH-NINDS R01 NS 50967-01 as part of the NSF/NIH Collaborative Research in Computational Neuroscience Program.
Correspondence F. Wood (fwood@cs.brown.edu)

Many solutions have been proposed to the model selection problem [9], [10]. Here we apply the infinite mixture modeling (IMM) technique of [11] to the spike sorting problem and demonstrate that it may in some ways outperform standard maximum likelihood techniques on real data while avoiding the model identification problem. We present an abbreviated introduction to IMM; for more details to reader is referred to [11]–[16] among others.

II. Methods

Suppose we have a single channel neurophysiological recording $R = [\vec{t}_1, \ldots, \vec{t}_N]$ consisting of N spike waveforms, where each waveform is a vector of n voltage samples $\vec{t}_i = [t_i^1, \ldots, t_i^n]^T \in \mathbb{R}^n$ ($n = 40$ throughout). Our goal is to 'sort' this channel by figuring out how many units (neurons) are present and which waveforms came from which units.

To both improve the numerical robustness of our approach and to aid in human verification of our results, we used a reduced dimensionality representation of the waveforms, where each $\vec{t}_i$ is represented by bases obtained via principal component analysis (PCA) such that $\vec{t}_i \approx \sum_{d=1}^{D} y_i^d \vec{u}_d$. Here $\vec{u}_d$ is the d^{th} PCA basis vector, and the y_i^d are the linear coefficients. Our spike sorting algorithm clusters the low dimensionality representation of the waveforms $\mathcal{Y} = [\vec{y}_i, \ldots, \vec{y}_N]$ rather than the full waveforms, so, for the remainder of this paper, when we write 'spike' or 'waveform' it should be read as shorthand for 'low dimensionality waveform representation'. For learning and sampling we used $D = 3$ while in Figures 1 and 2 we show $D = 2$, accounting for approximately 80% and 73% of the waveform variance respectively.

Like others we assume that the distribution of PCA coefficients of spike waveforms is well modeled by a multivariate Gaussian [5]. Under this assumption it makes sense to model a channel as a Gaussian mixture model, with one Gaussian density accounting for each hidden neuron. The corresponding generative model is

$$\begin{aligned} c_i|\vec{\pi} &\sim \text{Multinomial}(\cdot|\vec{\pi}) \\ \vec{y}_i|c_i = k, \Theta &\sim \mathcal{N}(\cdot|\theta_k) \end{aligned} \tag{1}$$

where $\mathcal{C} = \{c_i\}_{i=1}^N$ indicate which class each spike belongs to, $\Theta = \{\theta_k\}_{k=1}^K, \theta_k = \{\vec{\mu}_k, \Sigma_k\}$ are the class distribution parameters, and $\vec{\pi} = \{\pi_k\}_{k=1}^K, \pi_k = P(c_i = k)$ are the

mixture weights. To generate a spike in this model, first choose a neuron according to the multinomial $\vec{\pi}$, then sample a spike from the normal distribution for that neuron.

If we know the number of neurons, K, and the neuron responsible for each spike, we can compute the complete data likelihood

$$P(\mathcal{Y}, \mathcal{C}|\vec{\pi}, \Theta) = \prod_{i=1}^{N} \sum_{k=1}^{K} \pi_k P(\vec{y}_i|c_i = k, \Theta). \quad (2)$$

Unfortunately we know neither. To address this problem we may treat the class indicators as hidden random variables and use Expectation Maximization (EM) to find a maximum likelihood (ML) estimate of the model parameters

$$\hat{\vec{\pi}}, \hat{\Theta} = \arg\max_{\vec{\pi}, \Theta} \log(P(\mathcal{Y}, \mathcal{C}|\vec{\pi}, \Theta)). \quad (3)$$

EM is a powerful algorithm and guaranteed to converge to a local maximum of the likelihood. However, it has several problems. If the likelihood has multiple maxima (as is usually the case), the global maximum will not necessarily be found, and the choice of parameters for initialization can have a marked effect on the results. Also, since the likelihood of the data can be increased by adding more mixture components, a method for choosing the correct K must be adopted. One way is to penalize the likelihood by an information theoretic measure of model complexity such as the Bayesian information criterion BIC $= -2\log(P(\mathcal{Y}, \mathcal{C}|\vec{\pi}, \Theta)) + \nu_K log(N)$, where ν_K is the number of free parameters in a model with K hidden densities. Another way is to assume, as in [9] and [10], that K is finite but unknown.

As we will show, adopting a fully Bayesian approach allows us to avoid these problems. Bayes' rule tells us that the posterior probability of a model h given the observed data $\mathcal{Y}$ is proportional to the prior probability of h times the likelihood:

$$P(h|\mathcal{Y}) \propto P(h)P(\mathcal{Y}|h) \quad (4)$$

EM assumes that the prior probability of all hypotheses is equal, and seeks the single model with highest posterior probability (the maximum *a posteriori* (MAP) solution). Here, we use sensible priors to encode our knowledge about the problem, and seek to learn the full posterior distribution. This ultimately will allow us to avoid the model selection problem by marginalizing out the unknown model parameters (including the number of classes) when making inferences about the data.

Assume for the moment that we know the value of K. We choose conjugate priors for the model parameters:[1] Dirichlet for $\vec{\pi}$ and Normal times Inverse Wishart for θ [17], [18].

$$\begin{aligned} \vec{\pi}|\alpha &\sim \text{Dirichlet}(\cdot|\frac{\alpha}{K}, \ldots, \frac{\alpha}{K}) \quad (5) \\ \Theta &\sim \mathcal{G}_0 \end{aligned}$$

where $\Theta \sim \mathcal{G}_0$ is shorthand for

$$\begin{aligned} \Sigma_k &\sim \text{Inverse-Wishart}_{\upsilon_0}(\Lambda_0^{-1}) \\ \vec{\mu}_k &\sim \mathcal{N}(\vec{\mu}_0, \Sigma_k/\kappa_0). \end{aligned}$$

[1] A prior is conjugate if it yields a posterior that is in the same family as the prior (a mathematical convenience).

The Dirichlet hyperparameters, here symmetric $\frac{\alpha}{K}$, can encode our beliefs about how uniform or skewed the class mixture weights will be. The parameters to the Normal times Inverse-Wishart prior, $\Lambda_0^{-1}, \upsilon_0, \vec{\mu}_0, \kappa_0$, can encode our prior experience regarding the shape and position of the mixture densities. For instance $\vec{\mu}_0$ specifies where we believe the mean of the mixture densities should be, where κ_0 is the number of pseudo-observations we are willing to ascribe to our belief [17]. The hyper-parameters $\Lambda_0^{-1}, \upsilon_0$ behave similarly for the mixture density covariance.

Under this model, the posterior distribution is

$$\begin{aligned} &P(\mathcal{C}, \Theta, \vec{\pi}, \alpha|\mathcal{Y}) \quad (6) \\ &\propto \quad P(\mathcal{Y}|\mathcal{C}, \Theta)P(\Theta|\mathcal{G}_0) \prod_{i=1}^{N} P(c_i|\vec{\pi})P(\vec{\pi}\,|\alpha)P(\alpha). \end{aligned}$$

This distribution cannot be computed analytically, but one can obtain samples from it using Markov chain Monte Carlo (MCMC) methods [12]. These methods simulate a Markov chain whose equilibrium distribution is the posterior distribution in Eqn. 6. We describe our particular sampler below.

Sampling from this posterior distribution circumvents the problems with initialization and local optima that exist with ML estimation. We can avoid the problem of model selection by assuming that there are an infinite number of causal classes, but that only a finite number are ever observed in a finite amount of data. This is possible because the posterior (Eqn. 6) is well behaved as $K \to \infty$. This approach, first developed by [11], is an application of Dirichlet process mixture modeling [13], [14], [19].

The only terms in Eqn. 6 that involve K are $P(c_i|\vec{\pi})$ and $P(\vec{\pi}|\alpha)$. Further, because we chose conjugate priors, we can marginalize out $\vec{\pi}$ [15].

$$\begin{aligned} P(\mathcal{C}|\alpha) &= \int \prod_{i=1}^{N} P(c_i|\vec{\pi})P(\vec{\pi}\,|\alpha)d\vec{\pi} \\ &= \frac{\prod_{k=1}^{K} \Gamma(m_k + \frac{\alpha}{K})}{\Gamma(\frac{\alpha}{K})^K} \frac{\Gamma(\alpha)}{\Gamma(N + \alpha)}. \quad (7) \end{aligned}$$

Taking the limit as $K \to \infty$ (for details see [15]), we have

$$P(\mathcal{C}|\alpha) = \alpha^{K_+} \left(\prod_{k=1}^{K_+} (m_k - 1)! \right) \frac{\Gamma(\alpha)}{\Gamma(N + \alpha)} \quad (8)$$

where K_+ is the number of classes containing at least one item, $m_k = \sum_{i=1}^{N} I(c_i = k)$ is the number of items in class k, and $\Gamma(\alpha) = (\alpha - 1)\Gamma(\alpha - 1)$ is the Gamma function.

Ultimately, we wish to draw samples from the posterior distribution over models in Eqn. 6 when $K \to \infty$. We do so using an MCMC method known as Gibbs sampling [13], in which new values for each model parameter are repeatedly sampled conditioned on the current values of all other variables. Upon convergence, these samples will approximate the posterior distribution.

The state of our sampler consists of $\{\mathcal{C}, \Theta\}$. We sample new values for Θ according to

$$P(\theta_k|\mathcal{C}, \mathcal{Y}, \Theta_{-k}, \vec{\pi}, \alpha) \propto \prod_{i \text{ s.t. } c_i = k} P(\vec{y}_i|c_i, \theta_k)P_{\mathcal{G}_0}(\theta_k). \quad (9)$$

where $\Theta_{-k} = \{\theta_k, \ldots, \theta_{k-1}, \theta_{k+1}, \ldots, \theta_N\}$ and $P_{\mathcal{G}_0}(\theta_k)$ is the probability of θ_k under $\mathcal{G}_0$. $\mathcal{C}$ is sampled according to

$$P(c_i = k|\mathcal{C}_{-i}, \mathcal{Y}, \Theta, \vec{\pi}, \alpha) \propto P(\vec{y}_i|c_i, \Theta)P(c_i|\mathcal{C}_{-i}) \quad (10)$$

where $\mathcal{C}_{-i} = \{c_1, \ldots, c_{i-1}, c_{i+1}, \ldots, c_N\}$ can be derived from Eqn. 10 and is given by

$$P(c_i = k|\mathcal{C}_{-i}) = \begin{cases} \frac{m_k}{i-1+\alpha} & k \leq K_+ \\ \frac{\alpha}{i-1+\alpha} & k > K_+ \end{cases} \quad (11)$$

The generative process in Eqn. 11 is known as the Chinese restaurant process [15].

We sampled the remaining free parameter, α, using a Metropolis update, although it too can be updated using a Gibbs step [16].

III. Experiments

Four multi-unit hand sorted channels were selected from a multi-channel extra-cellular recording of an awake and behaving monkey performing a task. For a detailed description of the recording technique and data format see [6]. A regular subsample of the waveforms recorded on each channel was taken, corresponding to retaining every sixteenth spike (leaving 814, 299, 1360, and 838 on channels 1-4 as viewed left to right in Fig. 1). By starting at a different offset a separate held-out dataset was constructed similarly.

Fig. 1 shows modeling results for each channel using ML and infinite mixture modeling (IMM) techniques. For ML we used EM to train a model for each value of K between 1 and 15. Each model was trained 10 times per channel starting from different random seeds, and the model with overall maximum BIC score was selected. For IMM we used the sampler described above, again run 10 times per channel with different random seeds. Each run generated 5000 samples and the first 500 were discarded as burn-in. The sample with the highest posterior probability (an estimate of the MAP solution) is shown in Fig. 1. To generate these results, we used a Gamma$(1, 1)$ prior for α to encode our belief that each channel has relatively few neurons. We specified Λ_0 to be isotropic with variance equal to 0.1, encoding our belief that the clusters should be roughly spherical and tightly clustered. We set $\mu_0 = 0$. The other hyperparameters were assigned to give low weight to our prior. We believe that the clusters found by IMM more closely resemble the human-sorted 'ground truth' than do the EM clusters. In Fig. 1 one outlier was removed from channels 2 and 4. The channel 4 outlier is retained in Fig. 2.

A quantitative evaluation is shown in Table I, where we compare the log likelihood of held out data under the MAP estimates obtained from IMM and EM[2]. Note that computing the true log likelihood of held out data under the IMM requires integrating over all possible assignments of held-out spikes to neurons. Instead, we computed a lower bound on this quantity by constructing a (finite) Gaussian mixture model approximation to the infinite model, estimating $\vec{\pi}$ from the classification induced by the sample with $K = K_+$.

[2]Using only the MAP estimate from IMM is somewhat unfair to the Bayesian approach, since it discards the rest of the information contained in the posterior distribution. However, it makes evaluation simpler.

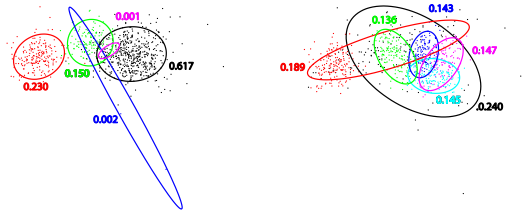

Fig. 2. Weights and 95% confidence intervals for the IMM MAP estimate (left) and the ML estimate with best BIC score (right) for channel 4.

	1	2	3	4	
IMM	-5.91 $\pm$ -0.03	-2.55 $\pm$ 0.05	-9.37 $\pm$ 0.04	-5.83 $\pm$ 0.02	$\times 10^3$
EM	-5.81 $\pm$ 0.02	-2.15 $\pm$ 0.17	-9.39 $\pm$ 0.07	-5.84 $\pm$ 0.59	$\times 10^3$

TABLE I

Per-channel average log likehood of held-out data

Given this approximation, we expected EM to outperform IMM, but their performance is virtually indistinguishable. We believe this is because the finite mixture model, lacking the influence of priors, is overfitting and doesn't generalize well. We observed that EM achieved higher log likelihood on the training data than IMM, which supports this assertion.

A benefit of using IMM is its implicit prior on class weights. This prior prefers clusterings with few prominent clusters. ML, on the other hand, implicitly assumes a uniform prior over class weights. This prior is uninformative with respect to cluster weights. Figure 2 illustrates the effects of these assumptions on clustering.

IV. Discussion and Future Work

In this work we have applied non-parametric Bayesian mixture modeling techniques to the problem of spike sorting and compared this method to maximum likelihood finite Gaussian mixture modeling. It should be noted that non-parametric Bayesian techniques in the form we have discussed suffer from a problem similar to the EM halting problem; i.e. it is difficult to determine when the Markov chain simulated by the Gibbs sampler has converged to its equilibrium distribution. In practice, convergence may not be necessary to find a good solution for a particular data set. Additionally, more sophisticated methods for initializing EM have been tried [6]. These are uncommon in general practice and, in limited experimentation, did not affect our reported results.

The primary contribution of this work is in presenting a way to avoid the problem of model selection in spike sorting. By changing the modeling assumption from 'There are K neurons on this channel.' to 'There are an infinite number of neurons that could be on this channel, how many are actually recorded?' it is possible to select MAP IMM model parameterizations that are competitive with traditional ML density estimators such as EM. This approach also allows us to address a wider range of questions than those allowed by traditional ML models. Due to space limitations, we provide only a single example: if we are interested in whether or not two spikes originate from the same neuron, the IMM yields

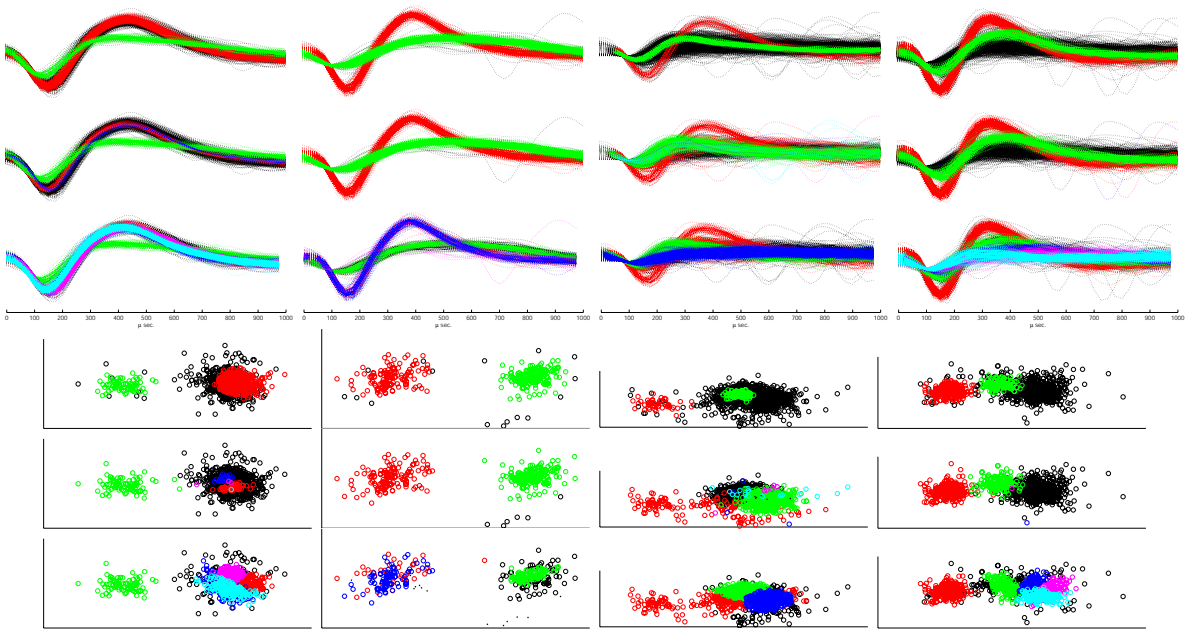

Fig. 1. Results from sorting four channels of real neural data using both maximum likelihood (ML) finite Gaussian mixture modeling (GMM) and infinite Gaussian mixture modeling (IMM). Channels 1-4 are shown left to right in columns. The first three rows show the sorted waveforms for human, IMM, and ML sorting. The last three rows show the sorted PCA waveform coefficients in the same order. The ML results are the model with highest Bayesian information criterion score. The IMM results are a single sample estimate of the MAP model drawn from the sampled posterior.

an answer that implicitly averages over an infinite number of models, avoiding the model selection problem. We believe that this is an important feature given the large amount of noise and ambiguity currently intrinsic to neural signals.

As developed in this work, the non-parametric Bayesian spike sorting approach is practical for offline analysis, yet it is also possible to extend it to online spike sorting. By estimating the posterior sequentially, an online spike sorter could be built that dynamically accomodates non-stationarity in the signal, adding neurons when sufficient evidence for their existence is recorded. This not only will solve the problem of determining when the Markov chain reaches equilibrium, but it also could be valuable for online analyses, long term neural decoding, and future clinical neuroprosthetic applications.

V. Acknowledgement

We wish to thank M. Fellows for data, S. Roth for careful editing, and T. L. Griffiths for introducing us to food-types as statistical metaphors.

References

[1] K. H. Kim and S. J. Kim, "Neural spike sorting under nearly 0-db signal-to-noise ratio using nonlinear energy operator and artificial neural-network classifier," *IEEE Trans. on Biomedical Engineering*, vol. 47, pp. 1406–1411, 2000.

[2] S. Takahashi, Y. Anzai, and Y. Sakurai, "Automatic sorting for multi-neuronal activity recorded with tetrodes in the presence of overlapping spikes," *Journal of Neurophysiology*, vol. 89, pp. 2245–2258, 2003.

[3] S. Shoham, M. R. Fellows, and R. A. Normann, "Robust, automatic spike sorting using mixtures of multivariate t-distributions," *Journal of Neuroscience Methods*, vol. 127(2), pp. 111–122, 2003.

[4] E. Hulata, R. Segev, and E. Ben-Jacob, "A method for spike sorting and detection based on wavelet packets and Shannon's mutual information," *Journal of Neuroscience Methods*, vol. 117, pp. 1–12, 2002.

[5] M. S. Lewicki, "A review of methods for spike sorting: the detection and classification of neural action potentials," *Network*, vol. 9, no. 4, pp. R53–78, 1998.

[6] F. Wood, M. Fellows, J. P. Donoghue, and M. J. Black, "Automatic spike sorting for neural decoding," in *IEEE Engineering Medicine Biological Systems*, 2004, pp. 4126–4129.

[7] M. Sahani, J. S. Pezaris, and R. A. Andersen, "On the separation of signals from neighboring cells in tetrode recordings," in *Advances in Neural Information Processing Systems 10*. MIT Press, 1998.

[8] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society, B*, vol. 39, 1977.

[9] N. Friedman, "The Bayesian structural EM algorithm," in *Proceedings of the 14th Annual conference on uncertainty in AI*, 1998, pp. 129–138.

[10] P. Green, "Reversible jump Markov chain Monte Carlo computation and Bayesian model determination," *Biometrika*, pp. 711–732, 1995.

[11] C. Rasmussen, "The infinite Gaussian mixture model," in *Advances in Neural Information Processing Systems 12*. Cambridge, MA: MIT Press, 2000.

[12] R. M. Neal, "Probabilistic inference using Markov chain Monte Carlo methods," University of Toronto, Tech. Rep. CRG-TR-93-1, 1993.

[13] ——, "Markov chain sampling methods for Dirichlet process mixture models," Department of Statistics, University of Toronto, Tech. Rep. 9815, 1998.

[14] S. MacEachern and P. Müller, "Estimating mixture of Dirichlet process models," *Journal of Computational and Graphical Statistics*, vol. 7, pp. 223–238, 1998.

[15] T. L. Griffiths and Z. Ghahramani, "Infinite latent feature models and the Indian buffet process," Gatsby Computational Neuroscience Unit, Tech. Rep. 2005-001, 2005.

[16] M. D. Escobar and M. West, "Bayesian density estimation and inference using mixtures," *Journal of the American Statistical Association*, vol. 90, pp. 577–588, 1995.

[17] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin, *Bayesian data analysis*. New York: Chapman & Hall, 1995.

[18] C. Fraley and A. E. Raftery, "Bayesian regularization for normal mixture estimation and model-based clustering," Department of Statistics, Washington University, Seattle, Washington, Tech. Rep. 05/486, 2005.

[19] R. M. Neal, "Markov chain sampling methods for Dirichlet process mixture models," *Journal of Computational and Graphical Statistics*, vol. 9, pp. 249–265, 2000.