RAO-BLACKWELLISED GIBBS SAMPLING FOR SWITCHING LINEAR DYNAMICAL SYSTEMS

A-V.I. Rosti and M.J.F. Gales

Cambridge University Engineering Department
Trumpington Street, Cambridge, CB2 1PZ, United Kingdom
e-mail: {avir2, mjfg}@eng.cam.ac.uk

ABSTRACT

This paper describes the application of Rao-Blackwellised Gibbs sampling (RBGS) to speech recognition using switching linear dynamical systems (SLDSs). The SLDS is a hybrid of standard hidden Markov models (HMMs) and linear dynamical systems. It is an extension of the stochastic segment model (SSM) as it relaxes the assumption of independent segments. SLDSs explicitly take into account the strong co-articulation present in speech. Unfortunately, inference in SLDS is intractable unless the discrete state sequence is known. RBGS is one approach that may be applied for both improved training and decoding for this form of intractable model. The theory of SLDS and RBGS is described, along with an efficient proposal mechanism. The performance of the SLDS using RBGS for training and inference is evaluated on the ARPA Resource Management task.

1. INTRODUCTION

Currently the most popular acoustic model for speech recognition is the hidden Markov model (HMM). However, HMMs are based on a series of assumptions some of which are known to be poor. In particular successive speech frames are assumed to be conditionally independent given the state that generated them. To overcome this limitation, segment models [1] have been proposed. These model whole segments of frames rather than individual frames. One example is the stochastic segment model (SSM). This uses a standard linear dynamical system (LDS) to model the sequence of observations within a segment. The LDS should provide both better spatial and temporal correlation model compared to HMM.

For the stochastic segment model, segments are assumed to be independent. The state vectors are thus initialised at the segment boundaries using the initial state vector distribution in the LDS. This is a poor assumption for speech due to co-articulation between the modelling units. The more states there are in a SSM system the closer its structure is to a factor analysed HMM [2]. In SLDS the posterior distribution of the state vector is propagated over the segment boundaries. Unfortunately, exact inference for SLDS is intractable, as the likelihood at any time depends on the entire discrete state sequence. Therefore, parameter optimisation using the standard EM algorithm, and inference using the Viterbi algorithm, is not feasible. Recently in the speech literature approximate decoding schemes for similar state space models have also been investigated. For example, the interacting multiple model approximation was investigated for both inference and training in [3].

A-V.I. Rosti is funded by an EPSRC studentship and Tampere Graduate School in Information Science and Engineering. He received additional support from Jenny and Antti Wihuri Foundation. This work made use of equipment kindly supplied by IBM under an SUR award.

The approximate inference scheme examined in this paper is based on Markov chain Monte Carlo methods [4]. Rather than modifying the model structure, or removing any dependencies in the state history, a sampling approach is adopted. Furthermore for efficiency, instead of sampling from the joint discrete and continuous state space, algorithm based on Rao-Blackwellisation is used. Rao Blackwellised Gibbs sampling has previously been applied for example to tracking of moving target in [5]. Here, the state and observation space dimensionalities range from 2 to 4 and the number of discrete states is 3 at most. In this paper the RBGS and methods to apply it in speech recognition are presented. In speech recognition, the dimensionalities of the state and observation space typically range between 13 and 39, and the number of discrete states in the thousands; i.e., dramatically larger than in the previous applications. Also the dynamic range of the continuous space statistics is much larger. In addition to the inference algorithm a parameter optimisation scheme based on maximum likelihood state sequence is proposed. This is the first study the authors are aware of where the SLDS has been applied to speech recognition without approximations that remove some model dependencies.

This paper is organised as follows. The next section describes the state space models in the generative model framework. In Section 3, the Rao-Blackwellised Gibbs sampling with application to speech is presented. The experiments and the results are described in the fourth section. Section 5 concludes the paper.

2. STATE SPACE MODELS

The models presented in this paper can be viewed as general state space models with N_s hidden discrete Markov states. In speech recognition applications the discrete state normally represents a phone, or context dependent phone. A hidden k-dimensional state vector, $\boldsymbol{x}_t$, is generated by the state evolution process. This continuous state vector can be viewed as an intermediate time evolving representation of the observation vectors. Every time instant, a p-dimensional observation vector, $\boldsymbol{o}_t$, is generated by a linear observation process. For all the models in this paper, the observation process is based on factor analysis.

2.1. Generative Models

The simplest state evolution process is a discrete state dependent vector of Gaussian distributed noise. This model is called the factor analysed HMM (FAHMM) [2]. Instead of generating the observation vectors, the underlying HMM generates vectors of latent variables for the factor analysis observation process. The generative model of FAHMM can be seen on the left hand side of Figure 1 where the discrete state sequence, $Q = \{q_1, q_2, \dots, q_T\}$, is defined by a set of transition probabilities, $a_{ij} = P(q_t = j|q_{t-1} = i)$. The state noise, $\boldsymbol{w}_j$, and the observation noise, $\boldsymbol{v}_j$, are distributed according to Gaussian distributions, $\mathcal{N}(\boldsymbol{\mu}_j^{(x)}, \boldsymbol{\Sigma}_j^{(x)})$ and $\mathcal{N}(\boldsymbol{\mu}_j^{(o)}, \boldsymbol{\Sigma}_j^{(o)})$, respectively. The observation matrices, $\boldsymbol{C}_j$, depend on the discrete state although any parameter in FAHMM can be arbitrarily tied. It is also possible to use Gaussian mixture models (GMMs) for both the state and observation noise sources.

$$
\begin{aligned}
q_t &\sim P(q_t|q_{t-1}) \\
\boldsymbol{x}_t &= \boldsymbol{w}_{q_t} \\
\boldsymbol{o}_t &= \boldsymbol{C}_{q_t}\boldsymbol{x}_t + \boldsymbol{v}_{q_t}
\end{aligned}
\qquad
\begin{aligned}
q_t &\sim P(q_t|q_{t-1}) \\
\boldsymbol{x}_t &= \boldsymbol{A}_{q_t}\boldsymbol{x}_{t-1} + \boldsymbol{w}_{q_t} \\
\boldsymbol{o}_t &= \boldsymbol{C}_{q_t}\boldsymbol{x}_t + \boldsymbol{v}_{q_t}
\end{aligned}
$$

Fig. 1. Generative models of a factor analysed HMM and a SLDS.

In SLDS the state vectors evolve according to a Gauss-Markov process. The generative model for a SLDS can be seen on the right hand side of Figure 1 where both the state transition matrices, $\boldsymbol{A}_j$, and the observation matrices, $\boldsymbol{C}_j$, are chosen by the discrete state and the state evolution and observation noises are Gaussian distributed as in FAHMM. The initial continuous state is also Gaussian distributed, $\boldsymbol{x}_1 \sim \mathcal{N}(\boldsymbol{\mu}_{q_1}^{(i)}, \boldsymbol{\Sigma}_{q_1}^{(i)})$. GMMs may also be used for all the noise sources. In comparison, the SSM would reset the continuous state vector, $\boldsymbol{x}_t$, according to the initial distribution every time the discrete state switches, $q_{t-1} \neq q_t$. For further discussion on the differences between FAHMM, SLDS and SSM including dynamic Bayesian network representations, see [6].

2.2. Inference and Training

For the FAHMM the inference is simple due to the conditional independence assumption. Both the standard Viterbi and forward-backward algorithms for the HMMs can be easily implemented for FAHMMs in $O(T)$ by modifying the likelihood calculations [2]. The parameter optimisation can be carried out using the EM algorithm. The inference for the SSM is more complicated since the position in the continuous state space depends on the number of frames spent in the current segment. However, standard optimisation methods are feasible [1], but at a cost of $O(T^2)$.

For the SLDS, the current position in the continuous state space depends on the entire history of the discrete states and the marginalisation becomes prohibitive. Exact computation of the observation or the posterior likelihood of the hidden variables given the observation sequence has to be carried out over $O(N_s^T)$ paths. However, given the discrete state sequence SLDS becomes tractable and the traditional Kalman filtering and smoothing algorithms can be used for inference, and EM algorithm for optimising the model parameters [6]. The intractable inference also renders any standard decoding algorithm inadmissible. Instead of full decoding, evaluation may be done if the segmentations of a number of hypotheses were known. The segmentations for training and N-best rescoring may be obtained from a tractable system such as the FAHMM.

3. APPROXIMATE INFERENCE FOR SLDS

Using segmented training data and N-best hypotheses may not be optimal for SLDS since the alignments must be produced by a tractable model with very different state evolution process. Deterministic algorithms to search the alternatives are not feasible due to the vast amount of possible segmentations in any realistic utterance. Instead, a stochastic approach may be adopted. Monte Carlo [4] simulation methods concentrate the search on areas with high probability reducing the waste of computing power. Markov chain Monte Carlo (MCMC) methods are based on drawing samples from proposal mechanisms with Markovian dependencies. The MCMC methods, such as Gibbs sampling, are especially suitable for inference in models with Markovian dependencies. Other sampling approaches would have to remove some dependencies in the model although they may have to be used for sequential processing. Since the entire utterances are available for the training and N-best rescoring, MCMC is the optimal choice.

3.1. Rao-Blackwellised Gibbs sampling

The efficiency of the Gibbs sampling algorithm depends on the initialisation and the size of the state space the samples are drawn from. For SLDS the initial alignments produced by a FAHMM system may be used as reasonable initialisations. To reduce the size of the state space, Rao-Blackwellisation may be employed. Instead of drawing the samples directly from the joint posterior of the discrete and continuous states, the tractable substructures in SLDS are utilised. In RBGS for SLDS, the samples are drawn from the proposal distribution for the discrete state and given the estimated discrete state, the continuous state space statistics can be computed using standard methods. The sampling algorithm for SLDS can be summarised as follows

1. initialise the discrete state sequence $\{q_1^{(1)}, \dots, q_T^{(1)}\}$;
2. for iteration $n > 1$
   - draw samples $q_t^{(n)} \sim P(q_t|\boldsymbol{O}, q_{-t}^{(n)})$, where $q_{-t}^{(n)} = \{q_1^{(n)}, \dots, q_{t-1}^{(n)}, q_{t+1}^{(n-1)}, \dots, q_T^{(n-1)}\}$
   - estimate statistics $\hat{\boldsymbol{x}}_t^{(n)} = E\{\boldsymbol{x}_t|\boldsymbol{O}, Q^{(n)}\}$ and $\hat{\boldsymbol{R}}_t^{(n)} = E\{\boldsymbol{x}_t\boldsymbol{x}_t'|\boldsymbol{O}, Q^{(n)}\}$.

where $Q^{(n)}$ denotes the entire discrete state sequence after iteration n, and $\hat{\boldsymbol{x}}_t^{(n)}$ and $\hat{\boldsymbol{R}}_t^{(n)}$ are the standard Kalman smoother statistics given the discrete state sequence, $Q^{(n)}$. Once all N iterations are finished, the final estimates can be approximated by

$$\gamma_j(t) \approx \frac{1}{N}\sum_{n=1}^{N}\delta(j - q_t^{(n)}) \tag{1}$$

$$\hat{\boldsymbol{x}}_t \approx \frac{1}{N}\sum_{n=1}^{N}\hat{\boldsymbol{x}}_t^{(n)} \tag{2}$$

$$\hat{\boldsymbol{R}}_t \approx \frac{1}{N}\sum_{n=1}^{N}\hat{\boldsymbol{R}}_t^{(n)} \tag{3}$$

where $\delta(\cdot)$ denotes the Dirac delta function. The statistics in Equations 1-3 can be shown to converge almost surely [5] toward the true posterior statistics $\gamma_j(t) = P(q_t = j|\boldsymbol{O})$, $\hat{\boldsymbol{x}}_t = E\{\boldsymbol{x}_t|\boldsymbol{O}\}$ and $\hat{\boldsymbol{R}}_t = E\{\boldsymbol{x}_t\boldsymbol{x}_t'|\boldsymbol{O}\}$. The proposal distribution for the Gibbs sampling is given in Equation 4 where $\boldsymbol{x}_{t|t-1}$, $\boldsymbol{\Sigma}_{t|t-1}$, $\boldsymbol{x}_{t|t}$ and $\boldsymbol{\Sigma}_{t|t}$ are the Kalman predictor and filter statistics, respectively. The statistics $\boldsymbol{P}_{t|t+1}^{-1}\boldsymbol{m}_{t|t+1}$ and $\boldsymbol{P}_{t|t+1}^{-1}$ may be obtained using the backward information filter defined by the following recursions

$$
\begin{aligned}
\boldsymbol{P}_{t|t}^{-1} &= \boldsymbol{C}_t' \boldsymbol{\Sigma}_t^{(o)-1} \boldsymbol{C}_t + \boldsymbol{P}_{t|t+1}^{-1} \\
\boldsymbol{P}_{t-1|t}^{-1} &= \boldsymbol{A}_t' \big(\boldsymbol{P}_{t|t}^{-1} \boldsymbol{\Sigma}_t^{(x)} + \boldsymbol{I}\big)^{-1} \boldsymbol{P}_{t|t}^{-1} \boldsymbol{A}_t \\
\boldsymbol{P}_{t|t}^{-1} \boldsymbol{m}_{t|t} &= \boldsymbol{P}_{t|t+1}^{-1} \boldsymbol{m}_{t|t+1} + \boldsymbol{C}_t' \boldsymbol{\Sigma}_t^{(o)-1} (\boldsymbol{o}_t - \boldsymbol{\mu}_t^{(o)}) \\
\boldsymbol{P}_{t-1|t}^{-1} \boldsymbol{m}_{t-1|t} &= \boldsymbol{A}_t' \big(\boldsymbol{P}_{t|t}^{-1} \boldsymbol{\Sigma}_t^{(x)} + \boldsymbol{I}\big)^{-1} \boldsymbol{P}_{t|t}^{-1} (\boldsymbol{m}_{t|t} - \boldsymbol{\mu}_t^{(x)})
\end{aligned}
$$

where $\boldsymbol{P}_{T|T+1}^{-1} = \mathbf{0}$. For detailed derivation of the proposal distribution and the backward information filter, see [6]. Compared to forms in [5], the recursions above include the state evolution and observation noise mean vectors for generality. The introduction of the backward information filter guarantees that the complexity of the Gibbs sampling algorithm is $O(T)$ per iteration. A straightforward implementation of the proposal distribution using the traditional Kalman filtering and RTS smoothing algorithms would result in a complexity of $O(T^2)$ per iteration.

The Gibbs sampling algorithm can be easily modified to support multiple component noise sources. The mixture components have to be initialised for the first iteration with the discrete states, and the Kalman and the backward information filters have to be run along the fixed components. Only modification to the proposal distribution in Equation 4 is to multiply it by the component priors. The mixture indicator sequence may also be initialised using a multiple component FAHMM.

The efficiency of the Rao-Blackwellised Gibbs sampling for SLDS can be further increased by taking advantage of the pronunciation restrictions. To keep the transcriptions valid, the correct order of phones in an utterance has to be retained during the sampling process. The utterance has to start in the first phone and end in the last phone in the transcription. Instead of drawing the samples from the entire set of phones in an utterance, samples from at most two discrete states have to be drawn at a time instant. Thus, no samples have to be drawn apart from the immediate proximity of a discrete state boundary.

3.2. Maximum Likelihood State Sequence Training

In the Monte Carlo EM [7] (MCEM) algorithm the continuous state posterior estimates given by Equations 2 and 3 are used in the standard update formulae for the LDS parameters. These formulae are based on the assumption that the continuous state posteriors are Gaussian. However, for the SLDS the posteriors may be mixture distributed and using them in the parameter estimation is not valid. Using only the first and second order statistics to estimate non-Gaussian distributions cannot be guaranteed to increase the log-likelihood of the training data. Furthermore, the convergence of the MCEM can only be established for very simple models.

Instead of the MCEM, a maximum likelihood state sequence (MLSS) scheme may be employed. In the experiments, alignments obtained using a FAHMM system were used in training the SLDS systems. In MLSS, Gibbs sampling is used to find a number of segmentations using the FAHMM alignments as an initialisation. The standard sufficient statistics for the LDS parameters are collected along the discrete state sequence that yields the highest log-likelihood. Given the discrete state sequence the continuous state posteriors are Gaussian distributed. Thus, the standard LDS update formulae are valid. For recent derivations see [6].

4. RESULTS

The ARPA Resource Management Corpus was used for the experiments. The training data set comprised 3990 utterances. All 1200 test utterances (feb89, oct89, feb91, sep92), `tst`, and a 300 utterance subset of the training data, `trn`, with a simple word-pair grammar were used for evaluation. A three state triphone FAHMM system clustered at the model level was built using standard methods [2]. All parameters of FAHMM, apart from the state space mean vectors, were then tied at the model level. By tying in this fashion the FAHMM is closely related to single state SLDS. By generating initial state alignments and N-best lists using this closely related model, rather than the standard HMM, the cross-system effects should be reduced. For these experiments context dependent models were used as the SLDS only uses a limited first order state evolution process. For examining the multiple mixture component performance, the observation noise distribution was split into a mixture of two Gaussians. The baseline FAHMMs were used to produce initial forced alignments of the training data and the 50-best hypotheses on the evaluation data for both the mixture configurations considered. The observation process parameters of an SLDS system were initialised based on the baseline FAHMM. A single set of LDS parameters per triphone was used. The initial continuous state vector distribution was initialised to the parameters of the first emitting state of the alignment FAHMM. The state evolution noise mean vectors were set to zeroes and the variances equal to the initial state variances. The state transition matrices, $\boldsymbol{A}_{q_t}$, were all initialised to identity matrices.

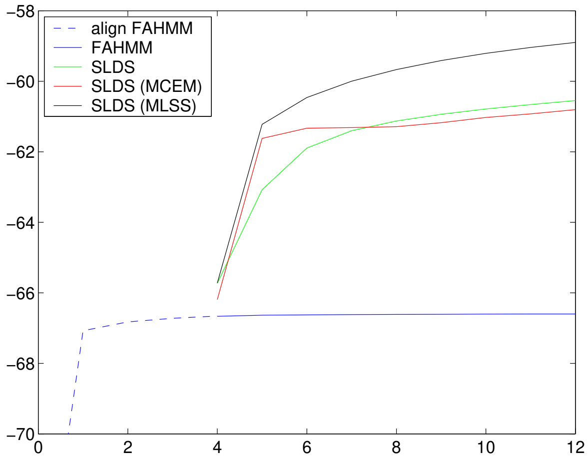

Fig. 2. Average log-likelihood of the training data against the number of iterations.

The model aligned training data was used to train the SLDS and FAHMM systems. The FAHMM system was initialised to the baseline FAHMM and the Baum-Welch algorithm was used to infer the state alignment holding the model alignment fixed. The average log-likelihoods of the training data against the number of iterations are shown in Figure 2. The first four iterations correspond to the baseline FAHMM training with full Baum-Welch algorithm and the last nine iterations correspond to the model aligned training. For the SLDS with fixed training alignment (SLDS) the log-likelihood slowly increased. Using MCEM the log-likelihood always increased but yielded a lower final log-likelihood than the fixed alignment training. As discussed in Section 3.2, the MCEM

$$P(q_t|\boldsymbol{O},q_{-t}) \propto P(q_{t+1}|q_t)P(q_t|q_{t-1})\mathcal{N}(\boldsymbol{o}_t;\boldsymbol{C}_t\boldsymbol{x}_{t|t-1}+\boldsymbol{\mu}_t^{(o)},\boldsymbol{C}_t\boldsymbol{\Sigma}_{t|t-1}\boldsymbol{C}_t'+\boldsymbol{\Sigma}_t^{(o)})\big|\boldsymbol{\Sigma}_{t|t}\boldsymbol{P}_{t|t+1}^{-1}+\boldsymbol{I}\big|^{-\frac{1}{2}}$$
$$\times\exp\Big\{\boldsymbol{x}_{t|t}'\boldsymbol{P}_{t|t+1}^{-1}\boldsymbol{m}_{t|t+1}-\frac{1}{2}\boldsymbol{x}_{t|t}'\boldsymbol{P}_{t|t+1}^{-1}\boldsymbol{x}_{t|t}+\frac{1}{2}(\boldsymbol{m}_{t|t+1}-\boldsymbol{x}_{t|t})'\boldsymbol{P}_{t|t+1}^{-1}\big(\boldsymbol{P}_{t|t+1}^{-1}+\boldsymbol{\Sigma}_{t|t}^{-1}\big)^{-1}\boldsymbol{P}_{t|t+1}^{-1}(\boldsymbol{m}_{t|t+1}-\boldsymbol{x}_{t|t})\Big\} \quad (4)$$

is not guaranteed to increase the log-likelihood. The state posteriors for this data were highly non-Gaussian. Initially, the MCEM gave significantly worse performance than other forms of training and was not investigated further. The MLSS training log-likelihoods with 5 iterations of Gibbs sampling are also shown. MLSS training clearly finds alignments with higher log-likelihood than using the fixed alignments. It was found that 5 iterations was enough to find the highest log-likelihood for most of the utterances. Larger number of iterations did not significantly increase the log-likelihoods or improve the rescoring results.

The full decoding word error rates for the $M = 1$ and $M = 2$ component baseline FAHMMs are shown in Table 1 in the column marked FAHMM. As a reference, word error rates for state-clustered FAHMM systems are 3.67% and 1.85% with 5 mixture components. Due to the non-standard model clustering, tying schemes and small number of mixture components, the baseline FAHMM results are far from the best achievable. The baseline FAHMMs were used to generate the 50-best lists for rescoring. To give an idea of the range of these N-best lists the oracle (best) error rate on the `tst` data was 1.24% and the "idiot" (worst) error rate was 52.05% for the single component system, and 1.23% and 50.93% for 2 component system. These are the bounds on subsequent rescoring results. The FAHMM trained using the model aligned data had the same performance as the baseline FAHMM.

M	Task	FAHMM	SLDS		SLDS-RBGS	
			N_0	N_5	N_0	N_5
1	`tst`	9.56	9.55	12.08	9.73	12.18
	`trn`	1.54	1.36	1.73	1.36	1.77
2	`tst`	8.90	10.32	11.57	11.26	11.57
	`trn`	0.83	1.32	1.62	1.58	1.77

Table 1. Rescoring word error rates for the baseline FAHMM and SLDSs trained with fixed alignments and MLSS training.

The rescoring results for the SLDS systems trained with the fixed alignments and MLSS with 5 Gibbs sampling iterations marked SLDS-RBGS are also shown in Table 1. Fixed alignment (N_0) and best of 5 Gibbs sampling iterations (N_5) were used. The highest log-likelihoods were again obtained during 5 iterations. Unfortunately, the results were disappointing. The SLDS system outperformed the baseline only when using the aligned training data. Even though the log-likelihoods in the training and rescoring were consistently higher, the model showed no improvements in the word error rates. Similar results were also found using just 13-dimensional front-end to see if the difference in the state and observation space dimensionality was an issue. The performance gap was even worse in 100-best rescoring when multiple states per model were used. Also the performance of the SSM using fixed alignments as well as RBGS was found to be inferior to that of an FAHMM. Further results and analysis are presented in [6].

5. CONCLUSIONS

This paper has introduced a new method to train and evaluate switching linear dynamical systems and stochastic segment models for speech recognition. The new scheme is based on MCMC simulation of the discrete state space which takes advantage of the tractable sub-structures in the models. Various implementation and efficiency issues for implementing Rao-Blackwellised Gibbs sampling for speech recognition have been described.

The performance of the SLDS and FAHMM were compared. RBGS was successfully applied to SLDS for both training and decoding, in terms of increasing log-likelihoods. However, the rescoring results were disappointing. The error rates were typically worse than the baseline FAHMM that was used to generate the N-best lists. Furthermore the performance became worse as "better" state alignments were used. Only the fixed alignment trained models showed any performance gain over the highly simplified alignment FAHMM. This error rate is still significantly worse than a standard HMM, or FAHMM. The simple linear state evolution assumption seems to render the model useless for recognition.

In this work RBGS was applied to SLDS. However, it is a general technique that may be applied to a variety of switching state space models. It gives an alternative approach to either simple approximations in inference and training, or simplifying the model structure as in variational Bayes methods.

6. REFERENCES

[1] M. Ostendorf, V. Digalakis, and O. Kimball, "From HMM's to segment models: A unified view of stochastic modeling for speech recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 4, no. 5, pp. 360–378, 1996.

[2] A-V.I. Rosti and M.J.F. Gales, "Factor analysed hidden Markov models for speech recognition," *Computer Speech and Language*, 2003, To appear.

[3] J.Z. Ma and L. Deng, "Efficient decoding strategy for conversational speech recognition using state-space models for vocal-tract-resonance dynamics," in *Proc. Eurospeech*, 2001, pp. 603–606.

[4] C.P. Robert and G. Casella, *Monte Carlo Statistical Methods*, Springer-Verlag, 1999.

[5] A. Doucet and C. Andrieu, "Iterative algorithms for state estimation of jump Markov linear systems," *IEEE Transactions on Signal Processing*, vol. 49, no. 6, pp. 1216–1227, 2001.

[6] A-V.I. Rosti and M.J.F. Gales, "Switching linear dynamical systems for speech recognition," Tech. Rep. CUED/F-INFENG/TR.461, Cambridge University Engineering Department, 2003, `http://mi.eng.cam.ac.uk/reports/`.

[7] G.C.G. Wei and M.A. Tanner, "A Monte Carlo implementation of the EM algorithm and the poor man's data augmentation algorithms," *Journal of the American Statistical Association*, vol. 85, pp. 699–704, 1990.