A REVERSIBLE JUMP SAMPLER FOR AUTOREGRESSIVE TIME SERIES

*Paul T. Troughton*and Simon J. Godsill*

Signal Processing and Communications Group, Department of Engineering,
University of Cambridge, Trumpington Street, Cambridge CB2 1PZ, England
ptt10@cam.ac.uk sjg@eng.cam.ac.uk

ABSTRACT

We use reversible jump Markov chain Monte Carlo (MCMC) methods to address the problem of model order uncertainty in autoregressive (AR) time series within a Bayesian framework. Efficient model jumping is achieved by proposing model space moves from the the full conditional density for the AR parameters, which is obtained analytically. This is compared with an alternative method, for which the moves are cheaper to compute, in which proposals are made only for new parameters in each move. Results are presented for both synthetic and audio time series.

1. INTRODUCTION

When fitting an autoregressive model to Gaussian time series data, often the correct order of the model is unknown. The model order cannot be estimated analytically by conventional Bayesian techniques when the excitation variance is unknown.

We present reversible jump MCMC methods [1] for drawing samples from the joint posterior of all the unknowns, from which Monte Carlo estimates of the quantities of interest, such as interpolated values, can be made. This has the advantage that these estimates will be based on a mixture of the probable models, rather than just the single most probable one.

Previous work on MCMC autoregressive model selection has parameterised the model using partial correlation coefficients [2, 3] or pole positions [4]. These have a simple physical interpretation for certain types of signal and allow stability to be enforced in a straightforward manner.

We use the AR parameters, $\mathbf{a}$, directly. This allows us to use the full conditional density for $\mathbf{a}$, which is available analytically, to propose efficient reversible jump MCMC moves.

While Barbieri & O'Hagan [3] also use reversible jump MCMC, Barnett *et al.* [2] and Huerta & West [4] use techniques similar to stochastic search variable selection [5], avoiding changing the dimension of the parameter vector by including all parameters, up to an arbitrary maximum order, at each iteration.

2. MODELLING FRAMEWORK

2.1. Autoregressive time series model

We model the signal $\{y_t\}$ as:

$$y_t = e_t + \sum_{i=1}^{k} a_i^{(k)} y_{t-i}$$

where $\{e_t\}$ is a white, Gaussian excitation sequence with variance σ_e^2, and $\mathbf{a}^{(k)}$ is the AR parameter vector for a kth order model. This can be rewritten in matrix-vector form as:

$$\mathbf{e} = \mathbf{A}\mathbf{y} = \mathbf{y}_1 - \mathbf{Y}^{(k)}\mathbf{a}^{(k)}$$

where $\mathbf{y}_0$ and $\mathbf{y}_1$ are formed by partitioning $\mathbf{y}$ into, respectively, the first k values and the remainder, and $\mathbf{A}$ and $\mathbf{Y}^{(k)}$ take appropriate forms.

Since the excitation sequence is Gaussian, the (approximate) likelihood takes the form [6, §A7.4]:

$$\begin{aligned} p(\mathbf{y} \mid k, \mathbf{a}^{(k)}, \sigma_e^2) &\approx p(\mathbf{y}_1 \mid \mathbf{y}_0, k, \mathbf{a}^{(k)}, \sigma_e^2) \\ &= (2\pi\sigma_e^2)^{-\frac{n_e}{2}} \exp\left(-\tfrac{1}{2\sigma_e^2}\mathbf{e}^T\mathbf{e}\right) \end{aligned}$$

where n_e is the length of $\mathbf{e}$ and $\mathbf{y}_1$.

2.2. Prior distributions

We choose simple conjugate prior distributions:

$$\begin{aligned} p(k) &= \begin{cases} \frac{1}{k_{\max}} & k \in \{0, 1, \dots k_{\max}\} \\ 0 & \text{elsewhere} \end{cases} \\ p(\mathbf{a}^{(k)} \mid k) &= \mathbf{N}\left(\mathbf{a}^{(k)} \mid \mathbf{0}, \sigma_a^2\mathbf{I}_k\right) \\ p(\sigma_a^2) &= \mathrm{IG}\left(\sigma_a^2 \mid \alpha_a, \beta_a\right) \\ p(\sigma_e^2) &= \mathrm{IG}\left(\sigma_e^2 \mid \alpha_e, \beta_e\right) \end{aligned}$$

where the inverted-gamma distribution (see *e.g.* [7]) is defined for positive parameters α and β, and positive x, as:

$$\mathrm{IG}(x \mid \alpha, \beta) \propto x^{-(\alpha+1)} \exp(-\beta/x)$$

which tends to the Jeffreys' prior as $\alpha, \beta \to 0$. k, $\mathbf{a}^{(k)}$, σ_a^2 and σ_e^2 are assumed to be *a priori* independent.

2.3. Bayesian hierarchy

The posterior density for the parameters is then:

$$p(k, \mathbf{a}^{(k)}, \sigma_a^2, \sigma_e^2 \mid \mathbf{y}) \propto \overbrace{p(\mathbf{y} \mid k, \mathbf{a}^{(k)}, \sigma_e^2)}^{\text{Likelihood}} \cdot \underbrace{p(k)\, p(\mathbf{a}^{(k)} \mid \sigma_a^2)\, p(\sigma_a^2)\, p(\sigma_e^2)}_{\text{Priors}}$$

*Supported by the Engineering & Physical Sciences Research Council.

3. REVERSIBLE JUMP MCMC

Metropolis-Hastings algorithms [8, 9] developed from molecular simulations as means to produce a Markov chain which converges to a required equilibrium distribution $p(\boldsymbol{\theta})$, without the need to sample directly from any inconvenient density.

Each step consists of choosing which subset of the parameters $\boldsymbol{\theta}$ to update, proposing new values for them by drawing from an arbitrary, convenient density $q_u(\boldsymbol{\theta}'_u \mid \boldsymbol{\theta}_{-u})$, calculating the acceptance probability for this move, $A(\boldsymbol{\theta} \rightarrow \boldsymbol{\theta}')$, then either accepting the move, setting the parameters to the proposed values, or rejecting the move, not changing any parameter values.

Reversible jump MCMC [1] is a generalisation which introduces moves between parameter spaces of different dimensionality, whilst retaining detailed balance [9], which is required for convergence, within each type of move.

If $J(k \rightarrow k')$ is the probability of proposing a move from a parameter space of dimension k to one of dimension k', and $\boldsymbol{\phi}$ contains those parameters which are present, and have the same meaning, in models of both dimensionalities, then the required acceptance probability is:

$$A\big((k, \boldsymbol{\theta}^{(k)}) \rightarrow (k', \boldsymbol{\theta}^{(k')})\big) = \min\Bigg(1, \underbrace{\frac{p(k', \boldsymbol{\theta}^{(k')} \mid \boldsymbol{\phi})}{p(k, \boldsymbol{\theta}^{(k)} \mid \boldsymbol{\phi}))}}_{\text{Target density ratio}} \underbrace{\frac{J(k' \rightarrow k)}{J(k \rightarrow k')} \frac{q(\boldsymbol{\theta}^{(k)} \mid k', \boldsymbol{\theta}^{(k')}, \boldsymbol{\phi})}{q(\boldsymbol{\theta}^{(k')} \mid k, \boldsymbol{\theta}^{(k)}, \boldsymbol{\phi})}}_{\text{Transition probability ratio}}\Bigg) \tag{1}$$

4. SAMPLING STRATEGY

The parameters to be sampled comprise k, $\mathbf{a}^{(k)}$, σ_e^2 and σ_a^2.

4.1. Model moves

Sampling k involves a change in dimensionality, so we use a reversible jump move. First, we choose to propose a move from order k to order k' by sampling k' from the distribution $J(k \rightarrow k')$, for which we use a discretised Laplacian density (see §4.3). We then sample a new parameter vector $\mathbf{a}^{(k')}$ from a proposal density:

$$\mathbf{a}^{(k')} \sim q(\mathbf{a}^{(k')} \mid k', \mathbf{a}^{(k)}, \mathbf{y}, \sigma_a^2, \sigma_e^2) \tag{2}$$

Note that σ_a^2 and σ_e^2 remain unchanged during model moves.

4.1.1. Full parameter vector proposals

As the proposal density, we can use the full conditional for the complete parameter vector $\mathbf{a}^{(k')}$, which is available analytically [10]:

$$\begin{aligned} q(\mathbf{a}^{(k')} \mid k', \mathbf{a}^{(k)}, \mathbf{y}, \sigma_a^2, \sigma_e^2) &= p(\mathbf{a}^{(k')} \mid k', \mathbf{y}, \sigma_a^2, \sigma_e^2) \\ &\propto \mathbf{N}\left(\mathbf{a}^{(k')} \mid \boldsymbol{\mu}_{\mathbf{a}^{(k')}}, \mathbf{C}_{\mathbf{a}^{(k')}}\right) \end{aligned} \tag{3}$$

where

$$\begin{aligned} \mathbf{C}^{-1}_{\mathbf{a}^{(k')}} &= \sigma_e^{-2} {\mathbf{Y}^{(k')}}^T \mathbf{Y}^{(k')} + \sigma_a^{-2} \mathbf{I}_{k'} \\ \boldsymbol{\mu}_{\mathbf{a}^{(k')}} &= \sigma_e^{-2} \mathbf{C}^T_{\mathbf{a}^{(k')}} {\mathbf{Y}^{(k')}}^T \mathbf{y}_{-[1 \ldots k']} \end{aligned}$$

Rather than drawing a value of $\mathbf{a}^{(k')}$, then simply substituting equation (3) and the likelihood and priors into equation (1), which could lead to numerical problems, we can use the 'Candidate's Identity' [11]:

$$\frac{p(k, \mathbf{a}^{(k)} \mid \mathbf{y}, \sigma_a^2, \sigma_e^2)}{p(\mathbf{a}^{(k)} \mid k, \mathbf{y}, \sigma_a^2, \sigma_e^2)} = p(k \mid \mathbf{y}, \sigma_a^2, \sigma_e^2)$$

to simplify equation (1) in this case to:

$$\begin{aligned} &A\big((k, \mathbf{a}^{(k)}) \rightarrow (k', \mathbf{a}^{(k')})\big) \\ &\qquad = \min\left(1, \frac{p(k' \mid \mathbf{y}, \sigma_a^2, \sigma_e^2)}{p(k \mid \mathbf{y}, \sigma_a^2, \sigma_e^2)} \frac{J(k' \rightarrow k)}{J(k \rightarrow k')}\right) \end{aligned} \tag{4}$$

We can express $p(k \mid \mathbf{y}, \sigma_a^2, \sigma_e^2)$ as:

$$\begin{aligned} &p(k \mid \mathbf{y}, \sigma_a^2, \sigma_e^2) \\ &\quad \propto p(k)\, p(\mathbf{y} \mid k, \sigma_a^2, \sigma_e^2) \\ &\quad = p(k) \int p(\mathbf{y}, \mathbf{a}^{(k)} \mid k, \sigma_a^2, \sigma_e^2)\, d\mathbf{a}^{(k)} \\ &\quad = p(k) \int p(\mathbf{y} \mid k, \mathbf{a}^{(k)}, \sigma_a^2, \sigma_e^2)\, p(\mathbf{a}^{(k)} \mid k, \sigma_a^2)\, d\mathbf{a}^{(k)} \end{aligned} \tag{5}$$

which allows us to simplify equation (4) to: [12]

$$\begin{aligned} &A\big((k, \mathbf{a}^{(k)}) \rightarrow (k', \mathbf{a}^{(k')})\big) \\ &\qquad = \min\Bigg(1, \frac{\sigma_a^{-k'}}{\sigma_a^{-k}} \frac{\left|\mathbf{C}_{\mathbf{a}^{(k')}}\right|^{\frac{1}{2}}}{\left|\mathbf{C}_{\mathbf{a}^{(k)}}\right|^{\frac{1}{2}}} \frac{J(k' \rightarrow k)}{J(k \rightarrow k')} \\ &\qquad\qquad \frac{\exp\left(\frac{1}{2} \boldsymbol{\mu}^T_{\mathbf{a}^{(k')}} \mathbf{C}^{-1}_{\mathbf{a}^{(k')}} \boldsymbol{\mu}_{\mathbf{a}^{(k')}}\right)}{\exp\left(\frac{1}{2} \boldsymbol{\mu}^T_{\mathbf{a}^{(k)}} \mathbf{C}^{-1}_{\mathbf{a}^{(k)}} \boldsymbol{\mu}_{\mathbf{a}^{(k)}}\right)}\Bigg) \end{aligned} \tag{6}$$

where, to eliminate dependence on the scale of the signal, the same vector $\mathbf{y}_1$, and hence length n_e, is used for both model orders being considered, *i.e.* all probabilities are conditional on the first $\max(k, k')$ values of $\mathbf{y}$. This expression is independent of $\mathbf{a}^{(k')}$, so the sampling operation of equation (2) need only be performed if the move is accepted.

4.1.2. Partial parameter vector proposals

Alternatively, we can propose only the additional AR parameters introduced by the move. This is quicker to compute than equation (3), but is likely to be susceptible to local maxima because low order parameters remain fixed in moves between higher order models. It is hence important also to sample $p(\mathbf{a}^{(k)} \mid k, \mathbf{y}, \sigma_a^2, \sigma_e^2)$ from time to time (see §4.2.1).

For partial parameter vector proposals, the acceptance probability takes different forms for 'birth' ($k' > k$) and 'death' ($k' < k$) moves:[1]

'Birth' move In this case, we are proposing

$$\mathbf{a}^{(k+n)} = \begin{bmatrix} \mathbf{a}^{(k)} \\ \mathbf{a}_u \end{bmatrix} \tag{7}$$

where $\mathbf{a}_u$ are the n new parameters drawn from the full conditional posterior density [10]:

$$\mathbf{a}_u \sim q(\mathbf{a}_u \mid k+n, \mathbf{a}^{(k)}, \mathbf{y}, \sigma_a^2, \sigma_e^2) \propto \mathbf{N}\left(\mathbf{a}_u \mid \boldsymbol{\mu}_{\mathbf{a}_u}, \mathbf{C}_{\mathbf{a}_u}\right)$$

[1] 'Life' moves ($k' = k$) need not be proposed at all.

where

$$\mathbf{C}_{\mathbf{a}_u}^{-1} = \sigma_e^{-2}\mathbf{Y}_u^{(k+n)\,T}\mathbf{Y}_u^{(k+n)} + \sigma_a^{-2}\mathbf{I}_n$$
$$\boldsymbol{\mu}_{\mathbf{a}_u} = \sigma_e^{-2}\mathbf{C}_{\mathbf{a}_u}\mathbf{Y}_u^{(k+n)\,T}\big(\mathbf{y}_1 - \mathbf{Y}_{-u}^{(k+n)}\mathbf{a}^{(k)}\big)$$

where the matrix $\mathbf{Y}$ is partitioned columnwise as $\mathbf{Y}_u$ and $\mathbf{Y}_{-u}$. Again, we can simplify equation (1) by marginalising $\mathbf{a}_u$ to give:

$$A\big((k,\mathbf{a}^{(k)}) \to (k+n, [\,{}^{\mathbf{a}^{(k)}}_{\mathbf{a}_u}\,])\big) = \min\Bigg(1, \frac{J(k+n\to k)}{J(k\to k+n)}\frac{p(k+n)}{p(k)} \sigma_a^{-n}\,|\mathbf{C}_{\mathbf{a}_u}|^{\frac{1}{2}}\exp\big(\tfrac{1}{2}\boldsymbol{\mu}_{\mathbf{a}_u}^T\mathbf{C}_{\mathbf{a}_u}^{-1}\boldsymbol{\mu}_{\mathbf{a}_u}\big)\Bigg) \quad (8)$$

'Death' move Here, no new parameters are being proposed; we merely truncate $\mathbf{a}^{(k)}$ at the k'th parameter. By definition,

$$A\big((k,\mathbf{a}^{(k)}) \to (k',\mathbf{a}^{(k')})\big) = \min(1, Q)$$
$$\Downarrow$$
$$A\big((k',\mathbf{a}^{(k')}) \to (k,\mathbf{a}^{(k)})\big) = \min(1, \tfrac{1}{Q})$$

so the calculations are similar to those for the acceptance probability for the corresponding 'birth' move.

4.2. 'Null' moves

Sampling from $\mathbf{a}^{(k)}$, σ_e^2 or σ_a^2 with fixed model order k does not involve any change of dimensionality, so the treatment is more straightforward.

4.2.1. Sampling the AR parameter vector

We can sample $\mathbf{a}^{(k)}$ directly from its full conditional (eq. 3) in a Gibbs move, for which the acceptance probability is always 1.

4.2.2. Sampling the noise variance

We can also sample σ_e^2 using a Gibbs move. To do this, we require the full conditional posterior distribution:

$$p(\sigma_e^2 \mid \mathbf{y}, k, \mathbf{a}^{(k)}, \sigma_a^2) \propto \overbrace{p(\mathbf{y} \mid k, \mathbf{a}^{(k)}, \sigma_a^2, \sigma_e^2)}^{\text{Likelihood}} \cdot \overbrace{p(\sigma_e^2)}^{\text{Prior}}$$
$$\approx \mathbf{N}\Big(\mathbf{e}^T\mathbf{e} \mid \mathbf{0}, \sigma_e^2\mathbf{I}\Big) \cdot \mathrm{IG}\big(\sigma_e^2 \mid \alpha_e, \beta_e\big)$$
$$= \mathrm{IG}\big(\sigma_e^2 \mid \alpha_{se}, \beta_{se}\big)$$

where

$$\alpha_{se} = \alpha_e + \tfrac{1}{2}n_e \qquad \text{and} \qquad \beta_{se} = \beta_e + \tfrac{1}{2}\mathbf{e}^T\mathbf{e}$$

We can sample from this inverted-gamma density directly.

4.2.3. Sampling the parameter variance

Similarly, we can use a Gibbs move to sample the hyperparameter σ_a^2:

$$p(\sigma_a^2 \mid \mathbf{y}, k, \mathbf{a}^{(k)}, \sigma_e^2) = p(\sigma_a^2 \mid \mathbf{a}^{(k)})$$
$$\propto p(\mathbf{a}^{(k)} \mid \sigma_a^2) \cdot p(\sigma_a^2)$$
$$\propto \mathbf{N}\Big(\mathbf{a}^{(k)} \mid \mathbf{0}, \sigma_a^2\mathbf{I}_k\Big) \cdot \mathrm{IG}\big(\sigma_a^2 \mid \alpha_a, \beta_a\big)$$
$$= \mathrm{IG}\big(\sigma_a^2 \mid \alpha_{sa}, \beta_{sa}\big)$$

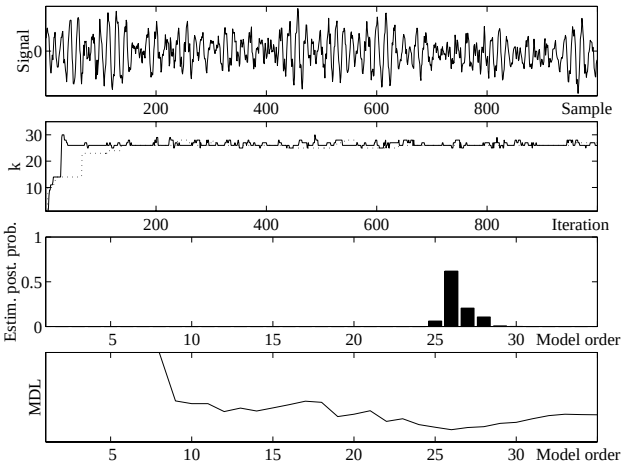

Figure 1: Orchestral recording (from top): Signal; Sampled model order values for full proposals (solid) and partial proposals (dotted); Monte Carlo estimate of $p(k \mid \mathbf{y})$; MDL values.

where

$$\alpha_{sa} = \alpha_a + \tfrac{1}{2}k \qquad \text{and} \qquad \beta_{sa} = \beta_a + \tfrac{1}{2}\mathbf{a}^{(k)\,T}\mathbf{a}^{(k)}$$

4.3. Move selection

For simplicity, a partially systematic scan was chosen: each proposed model move is followed by a σ_e^2 move and a σ_a^2 move, but sampling $\mathbf{a}^{(k)}$ is a relatively expensive move, so it is performed randomly, less frequently.

The choice of model move is determined by the function $J(k \to k')$. To ensure good convergence, we want most proposed jumps to be small, but occasional large ones to occur too. We have chosen a discretised Laplacian density:

$$J(k \to k+n) \propto \exp(-\lambda\,|n|)$$

5. RESULTS

The sampler was implemented as described for both full and partial parameter vector proposals.

5.1. Audio data

The samplers were used to fit an AR model to a block of 1000 samples from a 44.1kHz sampled orchestral music recording. Figure 1 shows the signal, along with the results of running the both samplers.

Although both versions converge to roughly the same model order distribution, it can be seen that the sampler using partial proposals appears to generate a less well mixed chain than the full proposal sampler.

The Monte Carlo estimate of the marginal posterior density $p(k \mid \mathbf{y})$ was obtained by calculating the histogram of k having discarded the values from the first 50 iterations as "burn-in" from the full parameter proposal sampler's output. Clearly, the maximum *a posteriori* estimate of the model order is 26, which agrees with the global minimum of the MDL criterion, which is plotted for comparison.

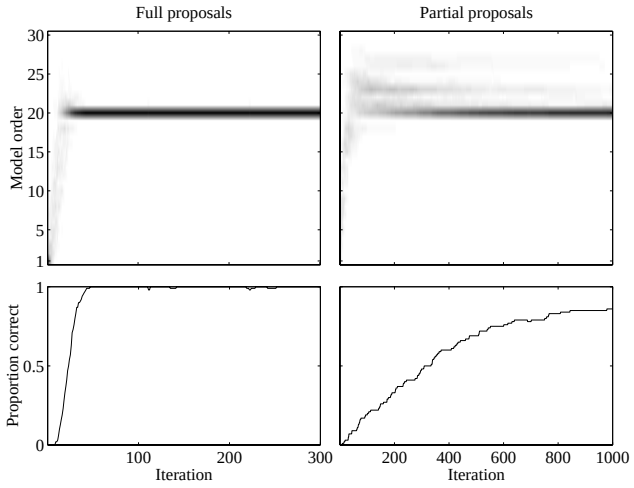

Figure 2: Convergence behaviour from 100 runs with synthetic AR(20) data: *(left)* Full proposals; *(right*) Partial proposals; *(top*) Evolution of the model order histogram – darkness represents frequency; *(bottom*) Frequency of choosing $k = 20$. Note differing x scales.

5.2. Synthetic data

To investigate convergence behaviour, 3500 samples were synthesised from an AR(20) process, and an ensemble of 100 runs were made with each sampler. The results are shown in figure 2. For each iteration, the top plots show the model order histogram, across the ensemble, and the bottom plots show the proportion of the ensemble which have the 'correct' value of model order.

It can be seen that all the runs of the full proposal sampler appear to converge within 50 iterations, whereas, even after 1000 iterations, some 15% of the partial proposal sampler runs have not converged.

6. DISCUSSION

This reversible jump sampler provides a fast, straightforward way to cope with AR model order uncertainty in an MCMC framework. Using the raw AR parameters allows the methods presented here to take advantage of the partially analytic structure of the AR model to speed convergence. The computation involved could probably be further reduced, for a given class of problems, by using a fully random scan and adjusting the move probabilities.

Proposing the full parameter vector in each move leads to a reliable sampler. Whilst proposing only part of the parameter vector makes acceptance probabilities faster to compute, the resulting Markov chain is more highly correlated.

Furthermore, whilst the two methods behave similarly for many modelling problems, in some cases, such as the AR(20) process of figure 2, the partial parameter vector proposals method is consistently very slow to converge. This is probably due to the the transition of equation (7) being quite unnatural when considered in terms of, for example, pole positions; models of different order to the correct one, but with similar values for common parameters, may have low likelihood.

We do not enforce model stability, as this is difficult to incorporate into a prior on the AR parameters. Rejection sampling could be used, but the marginalisation in equation (5) would need to be over a parameter space containing only stable models. This problem will be explored in future work.

7. REFERENCES

[1] P. J. Green. "Reversible jump Markov chain Monte Carlo computation and Bayesian model determination". *Biometrika*, 82(4):711–732, 1995.

[2] G. Barnett, R. Kohn, & S. Sheather. "Bayesian estimation of an autoregressive model using Markov chain Monte Carlo". *Journal of Econometrics*, 74(2):237–254, 1996.

[3] M. M. Barbieri & A. O'Hagan. *A reversible jump MCMC sampler for Bayesian analysis of ARMA time series.* Tech. rep., Dipartimento di Statistica, Probabilità e Statistiche Applicate, Università "La Sapienza", Roma and Department of Mathematics, University of Nottingham, 1996.

[4] G. Huerta & M. West. *Priors and component structures in autoregressive time series models.* Tech. rep., Institute of Statistics & Decision Sciences, Duke University, 1997.

[5] E. I. George & R. E. McCulloch. *Stochastic search variable selection.* In W. R. Gilks, S. Richardson, & D. J. Spiegelhalter, eds., *Markov Chain Monte Carlo in Practice: Interdisciplinary Statistics*, pp. 203–214. Chapman & Hall, 1996.

[6] G. E. P. Box, G. M. Jenkins, & G. C. Reinsel. *Time Series Analysis: Forecasting and Control*. Holden-Day, 3rd edition, 1994.

[7] N. L. Johnson & S. Kotz. *Distributions in Statistics: Continuous Univariate Distributions*. Wiley, 1970.

[8] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, & E. Teller. "Equation of state calculations by fast computing machines". *Journal of Chemical Physics*, 21(6):1087–1092, 1953.

[9] W. K. Hastings. "Monte Carlo sampling methods using Markov chains and their applications". *Biometrika*, 57(1):97–109, 1970.

[10] P. T. Troughton & S. J. Godsill. "Bayesian model selection for time series using Markov chain Monte Carlo". *Proceedings of IEEE ICASSP-97*, V:3733–3736, 1997.

[11] J. Besag. "A candidate's formula — a curious result in Bayesian prediction". *Biometrika*, 76(1):183, 1989.

[12] P. T. Troughton & S. J. Godsill. *A reversible jump sampler for autoregressive time series, employing full conditional to achieve efficient model space moves.* Tech. rep. CUED/F-INFENG/TR.304, Department of Engineering, University of Cambridge, 1997. Available from http://www-sigproc.eng.cam.ac.uk/~ptt10/papers/ar_rev.ps.gz.