MCMC METHODS FOR RESTORATION OF NONLINEARLY DISTORTED AUTOREGRESSIVE SIGNALS

Paul T. Troughton[*], *Simon J. Godsill*

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

ABSTRACT

We approach the problem of restoring distorted autoregressive (AR) signals by using a cascade model, in which the observed signal is modelled as the output of a nonlinear AR process (NAR) excited by the linear AR signal we are attempting to recover.

The Volterra expansion of the NAR model has a very large number of possible terms even when truncated at fairly small maximum orders and lags. We address the problem of subset selection and uncertainty in the nonlinear stage and model length uncertainty in the linear stage through a hierarchical Bayesian approach, using reversible jump Markov chain Monte Carlo (MCMC) and Gibbs sampling.

We demonstrate the method using synthetic AR data, and extend the approach to process a long distorted audio time series, for which the source model cannot be considered to be stationary.

1. INTRODUCTION

Autoregressive processes can be used to model a variety of signals, including audio. We consider the problem of reconstructing such a signal from a nonlinearly distorted version of it, when neither the form of the nonlinearity nor the maximum lag of the AR process are known.

The Volterra expansion [1] is a generalisation of the Taylor polynomial expansion into the time domain, and can approximate a wide variety of nonlinearities. We use it in a NAR model [2] to model the distortion process.

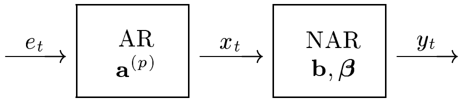
In making a Volterra expansion to the order and lag necessary to model adequately a given nonlinearity, there can be a very large number of possible terms. Incorporating all of these terms would lead to severe overfitting. Hence there is a need to select a subset.

The previous approach to this problem [3] assumed knowledge of the maximum lag of the AR process, and used a coarse grid search to reduce the number of candidate nonlinear terms first arbitrarily, then using the Akaike information criterion [4], followed a stepwise regression procedure to select a subset.

We present a fully Bayesian approach, evaluated using MCMC methods [5], which has the advantage, in the case of model uncertainty, that model mixing can be used, in which the reconstruction is based on all the models, according to their probabilities, rather than just the single most probable one.

2. MODEL FRAMEWORK

We use a cascade model [3] consisting of an AR source model driving a NAR channel model:



$$x_t = e_t + \sum_{i=1}^{p} a_i^{(p)} \, x_{t-i}$$

$$\begin{aligned} y_t = x_t &+ \sum_{i=1}^{\eta_b} \sum_{j=1}^{i} b_{i,j} \, \beta_{i,j} \, y_{t-i} \, y_{t-j} \\ &+ \sum_{i=1}^{\eta_b} \sum_{j=1}^{i} \sum_{k=1}^{j} b_{i,j,k} \, \beta_{i,j,k} \, y_{t-i} \, y_{t-j} \, y_{t-k} \\ &+ \text{higher order terms} \end{aligned} \qquad (1)$$

where $\{e_t\}$ is a zero-mean i.i.d. Gaussian excitation sequence, $\{x_t\}$ is the original signal, $\{y_t\}$ is the distorted signal we observe, $\{a_i^{(p)}\}$ are the parameters of the AR process with (unknown) maximum lag p, $\{b_{i,j}, b_{i,j,k}, \dots\}$ are the parameters of the NAR distortion process which is purely nonlinear, *i.e.* it only has terms of order 2 and above, and η_b is the maximum lag of the NAR model.

To represent a subset of the nonlinear terms, we introduce binary indicators, $\boldsymbol{\beta}$, such that if $\beta_{i,j} = 1$ then the term with parameter $b_{i,j}$ is included in the model. Note that, for simplicity, we have concatenated the different order nonlinear parameters into a single vector:

$$\mathbf{b} = \begin{bmatrix} b_{1,1} & b_{2,1} & \cdots & b_{\eta_b,\eta_b,\eta_b,\cdots\eta_b} \end{bmatrix}^T$$

We can extend this matrix-vector form to express equation (1) as:

$$\begin{aligned} \mathbf{e} = \mathbf{A}\mathbf{x} = \mathbf{x}_1 - \mathbf{X}^{(p)}\mathbf{a}^{(p)} \\ \mathbf{x} = \mathbf{y}_1 - \mathbf{Y}(\mathbf{b} \circ \boldsymbol{\beta}) \end{aligned} \qquad (2)$$

where $\circ$ denotes the Hadamard (elementwise) product, $\mathbf{x}_1$ omits the first k terms of $\mathbf{x}$, $\mathbf{y}_1$ omits the first η_b terms of $\mathbf{y}$, $\mathbf{A}$ and $\mathbf{X}$ are matrices containing elements from $\mathbf{a}^{(p)}$ and $\mathbf{x}$, respectively [6], and $\mathbf{Y}$ is a block diagonal matrix containing products of elements from $\mathbf{y}$.

[*] Supported by the Engineering & Physical Sciences Research Council.

2.1. Likelihoods

The output of the AR section, $\mathbf{x}$, is coloured zero-mean Gaussian noise which is completely described by:

$$\mathbf{x} \sim \mathbf{N}(\mathbf{x} \mid \mathbf{0}, \mathbf{C}_{\mathbf{x}}) \qquad \text{where} \qquad \mathbf{C}_{\mathbf{x}}^{-1} = \frac{\mathbf{A}^T\mathbf{A}}{\sigma_e^2}$$

where σ_e^2 is the variance of $\mathbf{e}$. The approximate likelihood for $\mathbf{y}$ can hence also be expressed as a multivariate Gaussian:

$$\begin{aligned} p(\mathbf{y} \mid \mathbf{b}, \boldsymbol{\beta}, \mathbf{a}^{(p)}, k, \sigma_e^2) &\approx p(\mathbf{y} \mid \mathbf{y}_0, \mathbf{b}, \boldsymbol{\beta}, \mathbf{a}^{(p)}, k, \sigma_e^2) \\ &= \mathbf{N}(\mathbf{y}_1 - \mathbf{Y}_{\boldsymbol{\beta}}\mathbf{b}_{\boldsymbol{\beta}} \mid \mathbf{0}, \mathbf{C}_{\mathbf{x}}) \end{aligned}$$

where $\mathbf{y}_0$ is the first η_b elements of $\mathbf{y}$ and $\mathbf{Y}_{\boldsymbol{\beta}}$ and $\mathbf{b}_{\boldsymbol{\beta}}$ are partitioned such that $\mathbf{Y}_{\boldsymbol{\beta}}\mathbf{b}_{\boldsymbol{\beta}} = \mathbf{Y}(\mathbf{b} \circ \boldsymbol{\beta})$.

Similarly, the approximate likelihood for $\mathbf{x}$ is:

$$p(\mathbf{x} \mid \mathbf{a}^{(p)}, k, \sigma_e^2) \approx \mathbf{N}\Big(\mathbf{x}_1 - \mathbf{X}^{(p)}\mathbf{a}^{(p)} \mid \mathbf{0}, \sigma_e^2\mathbf{I}\Big)$$

2.2. Priors

We choose simple Bernoulli and bounded uniform priors for the NAR indicators and AR model length:

$$\begin{aligned} p(\boldsymbol{\beta}) &= \prod_{i=1}^{n_b}\big(c\delta(1-\beta_i) + (1-c)\delta(\beta_i)\big) \\ p(k) &= \begin{cases} \frac{1}{k_{\max}} & k \in \{0, 1, \dots k_{\max}\} \\ 0 & \text{elsewhere} \end{cases} \end{aligned}$$

and conjugate priors for the model parameters and hyperparameters:

$$\begin{aligned} p(\mathbf{a}^{(p)} \mid \sigma_a^2) &= \mathbf{N}\big(\mathbf{a}^{(p)} \mid \mathbf{0}, \sigma_a^2\mathbf{I}\big) \\ p(\sigma_a^2) &= \mathrm{IG}\big(\sigma_a^2 \mid \alpha_a, \beta_a\big) \\ p(\sigma_e^2) &= \mathrm{IG}\big(\sigma_e^2 \mid \alpha_e, \beta_e\big) \\ p(\mathbf{b}_{\{m\}} \mid \sigma_b^2) &= \mathbf{N}\big(\mathbf{b}_{\{m\}} \mid \mathbf{0}, \tfrac{\sigma_b^2}{s_{\{m\}}}\mathbf{I}\big) \\ p(\sigma_b^2) &= \mathrm{IG}\big(\sigma_b^2 \mid \alpha_b, \beta_b\big) \end{aligned}$$

where $\mathbf{b}_{\{m\}}$ are the m-th order NAR parameters. This partitioning is required because we expect *a priori* the values of the NAR parameters of different orders to be of different magnitudes. We use $s_{\{m\}} = E(|y_t^m|)$ to scale the NAR parameter values to be comparable.

2.3. Bayesian hierarchy

We wish to reconstruct the signal, $\mathbf{x}$. Doing this using equation (2) requires knowledge of $\mathbf{b}$ and $\boldsymbol{\beta}$, the posterior for which is:

$$\begin{aligned} &p(\mathbf{b}, \boldsymbol{\beta} \mid \mathbf{y}, \mathbf{a}^{(p)}, k, \sigma_b^2, \sigma_e^2) \\ &\qquad \propto p(\mathbf{y} \mid \mathbf{b}, \boldsymbol{\beta}, \mathbf{a}^{(p)}, k, \sigma_e^2)\, p(\boldsymbol{\beta})\, p(\mathbf{b} \mid \sigma_b^2)\, p(\sigma_b^2) \end{aligned}$$

This, however, is dependent on $\mathbf{a}^{(p)}$ and k, which are also unknown, and have posterior:

$$\begin{aligned} &p(\mathbf{a}^{(p)}, k \mid \mathbf{x}, \sigma_a^2, \sigma_e^2) \\ &\qquad \propto p(\mathbf{x} \mid \mathbf{a}^{(p)}, k, \sigma_e^2)\, p(k)\, p(\mathbf{a}^{(p)} \mid \sigma_a^2)\, p(\sigma_a^2) \end{aligned}$$

which is dependent on $\mathbf{x}$.

3. MARKOV CHAIN MONTE CARLO

Since we cannot evaluate the required marginal distributions analytically, we take the MCMC approach [5], in which we generate samples from the joint posterior of all the parameters, from which we can then make Monte Carlo estimates. We cannot sample from the joint distribution directly, but we can use Gibbs sampling [7] or the Metropolis-Hastings algorithm [8] in order to sample only from conditional distributions of subsets of the parameters, and yet generate a Markov chain which converges in the limit to the correct joint distribution.

3.1. Gibbs moves for subset model

In plain Gibbs sampling, each parameter is sampled from its full conditional. Where there is strong interdependence between parameters, such as between b_i and β_i, convergence will tend to be slow [9]. We hence exploit the analytic structure of the model to marginalise b_i, to allow us to sample b_i and β_i jointly, and we further speed convergence by sampling parameters and indicators for small random groups of terms jointly [10]:

$$\mathbf{b}_u, \boldsymbol{\beta}_u \sim p(\mathbf{b}_u, \boldsymbol{\beta}_u \mid \mathbf{y}, \mathbf{b}_k, \boldsymbol{\beta}_k, \mathbf{a}^{(p)}, k, \sigma_b^2, \sigma_e^2)$$

which can be performed in two steps:

$$\begin{aligned} \boldsymbol{\beta}_u &\sim p(\boldsymbol{\beta}_u \mid \mathbf{y}, \mathbf{b}_k, \boldsymbol{\beta}_k, \mathbf{a}^{(p)}, k, \sigma_b^2, \sigma_e^2) \\ \mathbf{b}_u &\sim p(\mathbf{b}_u \mid \mathbf{y}, \boldsymbol{\beta}_u, \mathbf{b}_k, \boldsymbol{\beta}_k, \mathbf{a}^{(p)}, k, \sigma_b^2, \sigma_e^2) \end{aligned} \tag{3}$$

where the subscript $(\cdot)_u$ denotes the partition containing the element(s) we are sampling, and $(\cdot)_k$ is the remainder. Note that equation (3) is a discrete distribution, and independent of $\mathbf{b}_u$. These distributions are:

$$\begin{aligned} &p(\boldsymbol{\beta}_u = \boldsymbol{\beta}_c \mid \mathbf{y}, \mathbf{b}_k, \boldsymbol{\beta}_k, \mathbf{a}^{(p)}, k, \sigma_b^2, \sigma_e^2) \\ &\qquad \propto p(\boldsymbol{\beta}_u)\sigma_b^{-n_c}\, |\mathbf{C}_{s\mathbf{b}_c}|^{\frac{1}{2}} \exp\big(\tfrac{1}{2}\boldsymbol{\mu}_{s\mathbf{b}_c}^T \mathbf{C}_{s\mathbf{b}_c}^{-1} \boldsymbol{\mu}_{s\mathbf{b}_c}\big) \end{aligned}$$

and

$$\begin{aligned} &p(\mathbf{b}_c \mid \mathbf{y}, \boldsymbol{\beta}_c, \mathbf{b}_k, \boldsymbol{\beta}_k, \mathbf{a}^{(p)}, k, \sigma_b^2, \sigma_e^2) \\ &\qquad\qquad \propto \mathbf{N}\big(\mathbf{b}_c \mid \boldsymbol{\mu}_{s\mathbf{b}_c}, \mathbf{C}_{s\mathbf{b}_c}\big) \end{aligned}$$

where

$$\begin{aligned} \mathbf{C}_{s\mathbf{b}_c}^{-1} &= \mathbf{Y}_c^T \mathbf{C}_{\mathbf{x}}^{-1} \mathbf{Y}_c + \sigma_b^{-2}\mathbf{I} \\ \boldsymbol{\mu}_{s\mathbf{b}_c} &= \mathbf{C}_{s\mathbf{b}_c} \mathbf{Y}_c^T \mathbf{C}_{\mathbf{x}}^{-1}(\mathbf{y}_1 - \mathbf{Y}_k\mathbf{b}_k) \end{aligned} \tag{4}$$

where $\mathbf{Y}_c$ is those columns of $\mathbf{Y}$ which correspond to 1s in $\boldsymbol{\beta}_c$, the value of $\boldsymbol{\beta}_u$ for which the distributions are being evaluated, and n_c is the number of 1s in $\boldsymbol{\beta}_c$.

3.2. Reversible jump moves for AR model length

The parameter vector $\mathbf{a}^{(p)}$ is of unknown dimension, as the value of p is unknown. We hence use reversible jump MCMC [11], a generalisation of the Metropolis-Hastings algorithm which allows jumps between parameter spaces of different dimensionality.

Since the full conditional for the parameter vector $\mathbf{a}^{(p)}$ is available analytically, we can use it to propose efficient model moves. We can also marginalise the parameter values from the acceptance probability to make it easy to compute

[12]. For a jump from AR length p to length p', the acceptance probability is:

$$A(p \to p') = \min\left(1, \frac{\sigma_a^{-p'}}{\sigma_a^{-p}} \frac{\left|\mathbf{C}_{s\mathbf{a}(p')}\right|^{\frac{1}{2}}}{\left|\mathbf{C}_{s\mathbf{a}(p)}\right|^{\frac{1}{2}}} \frac{J(p' \to p)}{J(p \to p')} \frac{\exp\left(\frac{1}{2}\boldsymbol{\mu}^T_{s\mathbf{a}(p')}\mathbf{C}^{-1}_{s\mathbf{a}(p')}\boldsymbol{\mu}_{s\mathbf{a}(p')}\right)}{\exp\left(\frac{1}{2}\boldsymbol{\mu}^T_{s\mathbf{a}(p)}\mathbf{C}^{-1}_{s\mathbf{a}(p)}\boldsymbol{\mu}_{s\mathbf{a}(p)}\right)}\right)$$

where $J(p \to p')$ is the probability of choosing to propose a move between these model lengths, for which we choose a discretised Laplacian distribution, and other terms are defined in a similar manner to equation (4).

3.3. Gibbs moves for other parameters

The remaining parameters and hyperparameters, σ_a^2, σ_b^2 and σ_e^2 can all be sampled in Gibbs moves from their full conditionals, which are Inverse Gamma distributions.

It also speeds convergence to sample occasionally the complete parameter vectors $\mathbf{b}_{\boldsymbol{\beta}}$ and $\mathbf{a}^{(p)}$, whose full conditionals are multivariate Gaussians, so Gibbs moves can again be used.

3.4. Sampling strategy

$\mathbf{C}_{\mathbf{x}}^{-1}$ and $\mathbf{x}$ are relatively expensive to compute, so we separate our sampling moves into those which affect $\mathbf{C}_{\mathbf{x}}^{-1}$ and those which affect $\mathbf{x}$. We can then reduce computation by, within each iteration, making multiple moves from one of these groups before recomputing the affected variable and starting on the other group.

4. SYNTHETIC DATA

8000 samples were generated from a synthetic AR-NAR process with 6 AR lags and 6 nonlinear terms:

$$b_{2,2} = -0.20 \qquad b_{4,1} = 0.18 \qquad b_{5,4,1} = -0.16$$
$$b_{5,5,3} = 0.16 \qquad b_{5,5,5} = 0.20 \quad b_{2,2,2,1} = -0.10$$

Figure 1 shows the result of running the sampler for 2000 iterations with 82 candidate nonlinear terms (second and third order to lag 6 and fourth order to lag 2). Indicators were sampled in random triples, and 8 reversible jump moves were proposed each iteration. It was initialised with an empty model and arbitrary values for σ_e^2, σ_a^2 and σ_b^2.

It can be seen that the sampler converged very quickly to the correct number of AR lags. The 6 terms which appear most frequently in the sampler output are the correct nonlinear terms; that subset accounts for over 50% of the iterations after a 'burn-in' of 1000 iterations.

Figure 2 shows Monte Carlo estimates of posterior distributions of parameter values, produced from those post-burn-in iterations which selected the most popular model. It can be seen that the estimated distributions have significant mass close to the known true values. The scatter plot in figure 3 shows that the AR parameters were also accurately estimated.

5. LONG SIGNALS

A possible application of this method is the restoration of distorted audio signals. AR models are widely used in processing audio; a conventional approach is to break the audio signal into blocks which are sufficiently small, *i.e.* fractions of a second, that an assumption of stationarity is reasonable, and process each block independently.

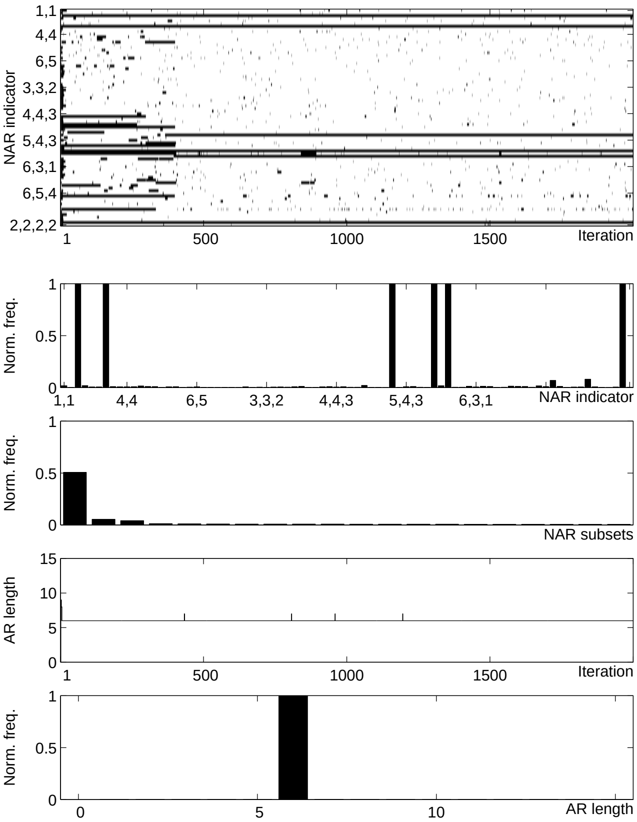


Figure 1: Identifying synthetic AR-NAR data *(from top)*: Raw β values — black areas represent 1s; Frequency of appearance of each NAR model term; Frequency of appearance of the most popular NAR subsets; Raw p values; Frequency of appearance of each p value.

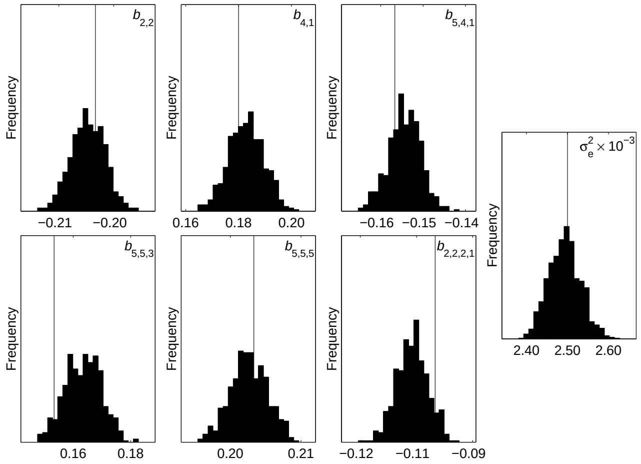


Figure 2: Identifying synthetic AR-NAR data: Histograms of σ_e^2 and NAR parameter values (true values marked by lines).

With many distortion problems, however, we can expect the distortion process to remain unchanged over a much longer period, perhaps minutes. This can be exploited by estimating $\mathbf{b}$ and $\boldsymbol{\beta}$ over many (not necessarily contiguous) blocks of audio, $\mathbf{y}_{[i]}$, $i \in \{1 \dots I\}$, the source model for each

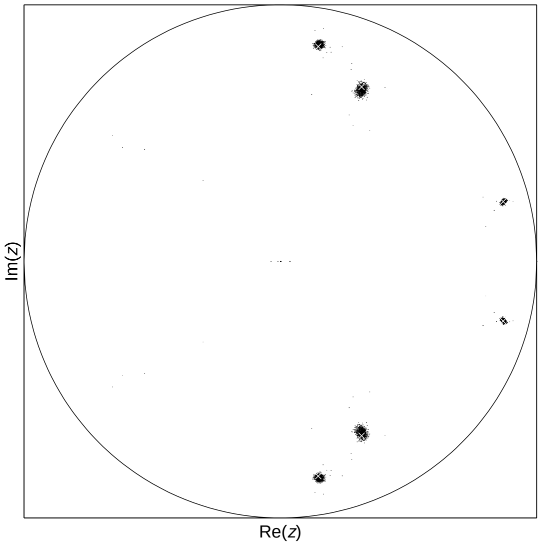


Figure 3: Identifying synthetic AR-NAR data: Sampled pole positions (true values marked by white crosses).

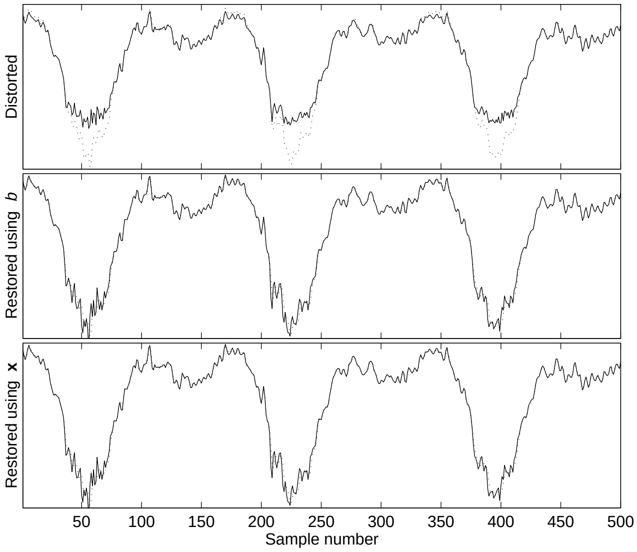


Figure 4: Restoring NAR distorted audio: Part of original signal *(dotted)* with *(solid, from top)*: Distorted signal $\mathbf{y}$; Restoration using estimates of $\mathbf{b}$ and $\boldsymbol{\beta}$; Restoration by direct Monte Carlo estimate of $\mathbf{x}$.

of which has separate parameters $\mathbf{a}_{[i]}^{(p)}$, $p_{[i]}$ and ${\sigma_e}^2{}_{[i]}$. This leads to $\mathbf{A}$ having a block Toeplitz form, which can be exploited to speed the evaluation of $\mathbf{C}_{\mathbf{x}}^{-1}$. The computation required increases only linearly with the number of blocks.

6. AUDIO DATA

70 000 samples from a 44.1kHz sampled pop music recording were artificially distorted by an NAR process containing third, fifth and sixth order terms. 10 randomly-chosen blocks of between 700 and 1000 samples from the signal were used for analysis. The sampler was run for 300 iterations.

Figure 4 shows part of the original and distorted waveforms, together with two attempts at correcting the distortion. The first uses a nonlinear moving average (NMA) filter, which is the inverse of the NAR process, with parameters $\mathbf{b}$ found by Monte Carlo estimation from the sampler's output for the most frequently appearing value of $\boldsymbol{\beta}$, having discarded the first 100 iterations as 'burn-in'. The second exploits model mixing by making a Monte Carlo estimate of $\mathbf{x}$ directly. It can be seen that, while the distorted version differs markedly from the original, both restorations match the original signal closely.

7. DISCUSSION

The MCMC method presented here jointly estimates the structure and parameters of a cascade AR-NAR model, and allows for model mixing. Based on our previous work [10, 12], we have exploited the partially analytic structure of both parts of the model to make joint moves, and efficient proposals, to speed the convergence of the Markov chain.

Future work will include the application of this method to the restoration of physically distorted audio. It is anticipated that the effects of bandlimiting by the recording media may need to be incorporated into the model [13].

8. REFERENCES

[1] M. Schetzen. *The Volterra and Wiener Theories of Nonlinear Systems.* John Wiley & Sons, 1980.

[2] H. Tong. *Non-linear Time Series: A Dynamical System Approach.* Oxford Statistical Science Series. Oxford University Press, 1990.

[3] K. J. Mercer. *Identification of Distortion Models.* PhD thesis, University of Cambridge, 1993.

[4] H. Akaike. "A new look at statistical model identification". *IEEE Transactions on Automatic Control*, AC-19:716–723, 1974.

[5] W. R. Gilks, S. Richardson, & D. J. Spiegelhalter, eds. *Markov Chain Monte Carlo in Practice: Interdisciplinary Statistics.* 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] S. Geman & D. Geman. "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6(6):721–741, 1984.

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

[9] L. Tierney. "Markov chains for exploring posterior distributions". *Annals of Statistics*, 22(4):1701–1762, 1994. With discussion.

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

[12] P. T. Troughton & S. J. Godsill. *A reversible jump sampler for autoregressive time series, employing full conditionals to achieve efficient model space moves.* Tech. rep. CUED/F-INFENG/TR.304, Department of Engineering, University of Cambridge, 1997.

[13] D. Preis & H. Polchlopek. "Restoration of nonlinearly distorted magnetic recordings". *Journal of the Audio Engineering Society*, 32(1):26–30, 1984.