A REVERSIBLE JUMP SAMPLER FOR POLYNOMIAL-PHASE SIGNALS

Céline Theys, Michelle Vieira, Gérard Alengrin

I3S, CNRS/UNSA, 41 Bd. Napoléon III, 06041 Nice, France
theys@unice.fr - vieira@unice.fr - alengrin@unice.fr

ABSTRACT

We use reversible jump Markov chain Monte Carlo (MCMC) methods to address the problem of order and parameters estimation of noisy polynomial-phase signals within a Bayesian framework. As posterior distributions of the parameters are not tractable, MCMC methods are used to simulate them. Efficient model jumping is achieved by proposing model space moves from the conditional density of the polynomial coefficients, estimated with the "one variable at a time" Metropolis Hasting algorithm. This algorithm provides simultaneous order and parameters estimation from simulated marginal posterior distributions. Results on simulated data are given and discussed.

1. PROBLEM STATEMENT

The signal under study is a noisy polynomial phase signal s_n as:

$$s_n = A\exp(j\sum_{i=0}^{M} a_i n^i) + e_n \tag{1}$$

where e_n is a complex Gaussian i.i.d. noise of known variance σ^2 and zero mean.

The aim of this paper is to estimate the order M and the parameters $\mathbf{a} = \{a_0, a_1, \ldots, a_M\}$ of the polynomial phase. This problem is encountered in radar systems or vibration monitoring for example, see [4] for a detailed introduction.

The Bayesian approach consists in regarding $\mathbf{a}$ and M as being drawn from prior distributions which reflect degree of knowledge on the parameters values. The Bayesian solution is then to evaluate the posterior density of $\mathbf{a}$ and M conditional on the data $\mathrm{s} = \{s_0, s_1, \ldots, s_{N-1}\}$ and the prior information I, abbreviated as $p(M, \mathbf{a}|\mathrm{s}, I)$. As posterior distributions of the parameters are not tractable, MCMC methods are used to simulate them. Estimating M implies a change in dimensionality so a reversible jump algorithm is proposed.

The estimated values of $\mathbf{a}$ and M are then computed respectively from the mean (MMSE estimator) and the maximum (MAP estimator) of the simulated posterior densities, [3], [7].

2. POSTERIOR DENSITIES

By applying Bayes' theorem, the joint posterior probability density of the parameters, $p(M, \mathbf{a}, A, \sigma^2|\mathrm{s}, I)$, is :

$$p(M, \mathbf{a}, A, \sigma^2|\mathrm{s}, I) \propto p(M, \mathbf{a}, A, \sigma^2|I)p(\mathrm{s}|M, \mathbf{a}, A, \sigma^2, I) \tag{2}$$

where $p(M, \mathbf{a}, A, \sigma^2|I)$ is the parameter prior,

$p(\mathrm{s}|M, \mathbf{a}, A, \sigma^2, I)$ is the likelihood function. Since e_n is Gaussian, the likelihood is:

$$p(\mathrm{s}|M, \mathbf{a}, A, \sigma^2, I) = \prod_{i=0}^{N-1} p(e_i)$$
$$= \frac{1}{\pi^N \sigma^{2N}} \exp\left(-\frac{1}{\sigma^2}\sum_{k=0}^{N-1} |s_k - A\exp j\sum_{i=0}^{M} a_i k^i|^2\right) \tag{3}$$

Parameters $\mathbf{a}$ and M are assumed to be uniformly distributed to express ignorance about the value of the parameter vector in absence of data. We use Jeffrey's prior for σ, [6] :

$$p(\sigma^2/I) \propto \frac{1}{\sigma^2}. \tag{4}$$

Then the joint posterior density (2) is:

$$p(M, \mathbf{a}, A, \sigma^2|\mathrm{s}, I) \propto \frac{1}{\sigma^{2N+2}} \exp$$
$$\left(-\frac{1}{\sigma^2}\sum_{k=0}^{N-1} |s_k - A\exp j\sum_{i=0}^{M} a_i k^i|^2\right) \tag{5}$$

This form of distribution allows us to eliminate the so-called nuisance parameters A and σ^2 by integration:

$$p(M, \mathbf{a}|\mathrm{s}, I) = \int_{A=-\infty}^{\infty}\int_{\sigma^2=0}^{\infty} p(M, \mathbf{a}, A, \sigma^2|\mathrm{s}, I)dAd\sigma^2 \tag{6}$$

After computation:

$$p(M, \mathbf{a}|\mathrm{s}, I) \propto (\sum_{k=0}^{N-1} s_k^* s_k - \frac{1}{4N^2}(\sum_{k=0}^{N-1} s_k^* \exp(j\sum_{i=0}^{M} a_i k^i) +$$
$$\sum_{k=0}^{N-1} s_k \exp(-j\sum_{i=0}^{M} a_i k^i))^2)^{-(N+3/2)} \tag{7}$$

It is worth noticing that eq.(7) is highly non-linear in the coefficients $\mathbf{a}$ and that an expression of $p(M|\ldots)$ can not be obtained in closed form, even up to a normalizing constant. We develop in the following MCMC methods to estimate $p(M, \mathbf{a}|\mathrm{s}, I)$.

3. REVERSIBLE JUMP MCMC

Metropolis-Hastings algorithms produce a Markov chain which converges to a required distribution $p(\theta)$ from a convenient density, easy to simulate. Each step consists of:

- Choosing the subset of the parameters θ to update.
- Proposing new values for the subset associated with that move by drawing from an arbitrary, convenient density:

$$\theta'_u \sim q_u(\theta'_u|\theta_{-u}) \tag{8}$$

  where $-u$ denotes the complementary subset of u.
- Calculating the acceptance probability for this move, $\alpha(\theta \to \theta')$:

$$\alpha(\theta \to \theta') = \min\left(1, \frac{p(\theta'_u|\theta_{-u})q_u(\theta_u|\theta_{-u})}{p(\theta_u|\theta_{-u})q_u(\theta'_u|\theta_{-u})}\right) \tag{9}$$

- Either
  - Accepting the move and setting the parameters to the proposed values or
  - Rejecting the move, not changing any parameter value

Reversible jump MCMC (Green 1995) is a generalization which introduces moves between parameter spaces of different dimensionality. Candidates are proposed according to a set of proposal distributions. These candidates are randomly accepted following an acceptance ratio which ensures reversibility and thus invariance of the Markov chain with respect to the posterior distribution. One problem is to evaluate a ratio of probability measures between subspaces of different dimensions. To avoid this problem Green has proposed to perform reversible jumps between subspaces via proper dimension matching, see [2] for details.

If $P(k \to k')$ is the probability of proposing a move from a parameter space of dimension k to one of dimension k', and ϕ contains the parameters, then the required acceptance probability becomes:

$$\alpha((k,\theta^k) \to (k',\theta^{k'}) = \min\left(1, \frac{p(k',\theta^{k'}|\phi)P(k' \to k)q(\theta^k|k',\theta^{k'},\phi)}{p(k,\theta^k|\phi)P(k \to k')q(\theta^{k'}|k,\theta^k,\phi)} J(k \to k')\right) \tag{10}$$

where $J(k \to k')$ is the Jacobian of the transformation.

4. SAMPLING STRATEGY

The parameters to be sampled are the coefficients $\mathbf{a}$ and the order M of the polynomial phase.

4.1. Model moves

Sampling M involves a change in dimensionality, so a reversible jump move is used. We have chosen the following complementary reversible jumps to ensure reversibility:

1. increasing order: $M \to M+1$.
2. decreasing order: $M \to M-1$.

This choice is purely heuristic and has the main advantage to be very simple to implement, the choice of jumps has only an influence on the rate of convergence of the algorithm. Another scheme would be to jump from an order M to an order M' for example.

At each iteration, one of the candidate movement, $M \to M+1$ or $M \to M-1$ is randomly chosen with respective probabilities p_M and $1-p_M$ for all $1 \leq M \leq M_{\max}$. For $M=1$, decreasing order is impossible, so $1-p_{M_1}=1$; for $M=M_{\max}$, increasing order is impossible so $p_{M_{\max}}=0$. All others values of M give increasing order or decreasing order with the same probability and each time M corresponding values are generated for the M polynomial coefficients. From the resulting order M, new values of the complete vector $\mathbf{a}_M$ is generated.

4.2. Algorithm

In this section, algorithms are given in details. The main procedure is the following:

Reversible MCMC algorithm

1. **Initialization**: set $(M^0, \mathbf{a}_{M^0}) \in \{1,\dots,M_{\max}\} \times \mathcal{A}$ belonging to the support of the posterior density, (7).
2. **Iteration i** $(i=1,\dots,T)$
   - Update the parameters $\mathbf{a}^i_M$ with "*M-H one variable at a time*" algorithm.
   - Draw $u \sim \mathcal{U}_{[0,1]}$
     - if $u \leq p^i_M$ "*increasing order*".
     - else "*decreasing order*".
     - end if.
3. $\mathbf{i} \leftarrow \mathbf{i}+1$, goto 2

The main algorithm calls the following procedures:

M-H one variable at a time

1. **Iteration m** $(m=0,\dots,M^i)$
   - Draw $y_m \sim \mathcal{U}_{[-\delta_m,\delta_m]}$
   - Calculate the acceptance coefficient: $\alpha(a^i_m, y_m) = \min(1, r^i)$
     $r^i = \frac{p(y_m|a^i_0,\dots,a^i_{m-1},a^{i-1}_{m+1},\dots,a^{i-1}_M)}{p(a^i_m|a^i_0,\dots,a^i_{m-1},a^{i-1}_{m+1},\dots,a^{i-1}_M)}$
   - Draw $u \sim \mathcal{U}_{[0,1]}$
     - if $u_m \leq \alpha(a^i_m, y_m)$ then $a^i_m = y_m$.
     - else $a^i_m = a^{i-1}_m$.
     - end if.
2. $\mathbf{m} \leftarrow \mathbf{m}+1$, goto 1

increasing order

- Draw $\mathbf{y} \sim \mathcal{N}(0, \Sigma_{M^+})$
- Calculate the acceptance probability $\alpha\left((M^{i-1},\mathbf{a}),(M^+,\mathbf{y})\right) = \min(1, r^i)$
  $r^i = \frac{p(M^+,\mathbf{y})(1-p_{M^+})|\Sigma_{M^+}|^{1/2}\exp(-\mathbf{y}^t\Sigma^{-1}_{M^+}\mathbf{y})}{p(M^{i-1},\mathbf{a}^{i-1})p_{M^+}|\Lambda_{M^+}|^{1/2}\exp(-\mathbf{a}^{t(i-1)}\Lambda^{-1}_{M^+}\mathbf{a}^{(i-1)})}$

- Draw $u \sim \mathcal{U}_{[0,1]}$
  - if $u \leq \alpha\left((M^{i-1}, \mathbf{a}), (M^{+}, \mathbf{y})\right)$ then $\mathbf{a}^i = \mathbf{y}$, $M^i = M^+$.
  - else $\mathbf{a}^i = \mathbf{a}^{i-1}$, $M^i = M^{i-1}$
  - end if.

decreasing order

- Draw $\mathbf{y} \sim \mathcal{N}(0, \Lambda_{M^-})$
- Calculate the acceptance probability $\alpha\left((M^{i-1}, \mathbf{a}), (M^{-}, \mathbf{y})\right) = \min(1, r^i)$

$$r^i = \frac{p(M^-, \mathbf{y}) p_M^- |\Lambda_{M^-}|^{1/2} \exp(-\mathbf{y}^t \Lambda_{M^-}^{-1} \mathbf{y})}{p(M^{i-1}, \mathbf{a}^{i-1})(1-p_{M^-})|\Sigma_{M^-}|^{1/2} \exp(-\mathbf{a}^{t(i-1)} \Sigma_{M^-}^{-1} \mathbf{a}^{(i-1)})}$$

- Draw $u \sim \mathcal{U}_{[0,1]}$
  - if $u \leq \alpha\left((M^{i-1}, \mathbf{a}), (M^{-}, \mathbf{y})\right)$ then $\mathbf{a}^i = \mathbf{y}$, $M^i = M^-$.
  - else $\mathbf{a}^i = \mathbf{a}^{i-1}$, $M^i = M^{i-1}$
  - end if.

Notation

- T: number of iterations.
- $M^{i-1} + 1 = M^+$, $M^{i-1} - 1 = M^-$.
- $\mathcal{A}$: subset of possible values of $\mathbf{a}$.
- $u \sim p(u)$: u is distributed according to $p(u)$.
- $\mathcal{U}_{[a,b]}$: uniform distribution on $[a, b]$.
- $\mathcal{N}(m, \Sigma)$: Gaussian multivariate distribution with mean m and covariance matrix Σ.

Some comments and precisions about these algorithms:

- The Jacobian term does not appear in acceptance probability of the moves procedures since parameters are generated directly in the new parameter space.
- New values of $\mathbf{a}$ are drawn from Gaussian densities since posterior densities of the coefficients do not have a standard form.
- The M-H "one-variable at a time" algorithm combines $M + 1$ updates to draw the posterior density of the parameter vector $\mathbf{a}$. It is usually used because it is far easier to find several conditional kernels that converge to their respective conditional densities than to find one kernel that converges to the joint, see [1] for a simple exposition of the M-H algorithm.

5. SIMULATION RESULTS

In order to study performances of the proposed algorithm, computer simulations using $N = 100$ samples polynomial-phase signal of order $M = 3, 4$ and signal to noise ratio $snr = 0, 5, 10dB$ have been drawn. Parameters for $M = 3$ are $a_0 = \pi/4$, $a_1 = -0.02$ and $a_2 = 0.002$, for $M = 4$, the first parameters are the same and $a_4 = -.0001$.

Parameters of the algorithm are $M_{\max} = 6$, $\delta_0 = 0.015$, $\delta_i = 3\exp(-2(1+i))$, $\Sigma_{i,i} = 2\exp(-i)$, $\Lambda = \Sigma$. Note that the values of these parameters only have an influence on the rate of convergence of the algorithm. Initial values M^0, $\mathbf{a}_{M^0}$ are randomly chosen.

parameter true value		a_0 0.7854	a_1 -0.02	a_2 0.002
$rsb = 10dB$	mean	0.7623	-0.0186	0.002
	var	1.49e-2	1.05e-4	4.96e-8
$rsb = 5dB$	mean	0.7701	-0.0177	0.002
	var	2.38e-2	1.50e-4	2.48e-7
$rsb = 0dB$	mean	0.5444	-0.0046	0.0018
	var	6e-2	1.51e-4	5.70e-8

Table 1: Mean and standard deviation of the polynomial coefficients for $M = 3$ and $rsb = 10, 5, 0dB$

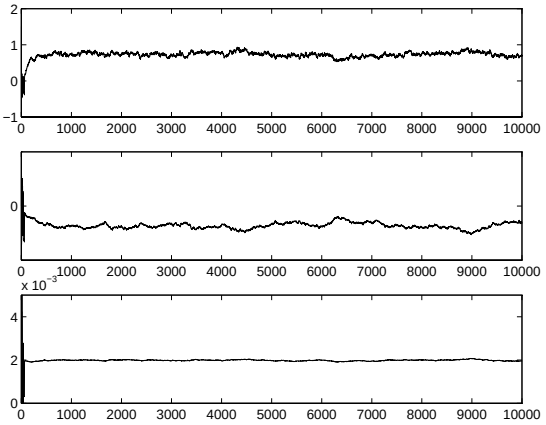

Figure 1: Generated chains for a_0, a_1 and a_2, $rsb = 10dB$ (top to bottom)

Figure (1) shows the chains generated by the algorithm for a three order phase signal and a $snr = 10dB$. Chain generated for M has not been given since it remains at the true order $M = 3$ from a number of iterations equal to 200 samples.

Figure (2) shows posterior distribution $p(M|\mathbf{y})$ of the four order phase signal for three snr.

Figures (3, 4, 5) show posterior distribution of the polynomial coefficients following the snr.

Table (1) gives mean and variance over 30 realizations of $T = 5000$ samples, ignoring the first $T_0 = 2000$ samples as a burning time period.

Let us note that results obtained show classical characteristics of polynomial-phase signals: supports of the posterior densities increase as the snr decrease, relative error and support are always the largest for $i = 0$, and decrease monotonically with i (figs. 3,4,5), [5].

In short, the order is correctly found, at more than 90% whatever is the snr, (fig. 2) and estimation of the parameters is accurate, (tab. 1). Let us insist on the fact that the algorithm provides much more information than classical approaches, particularly posterior densities of the parameters.

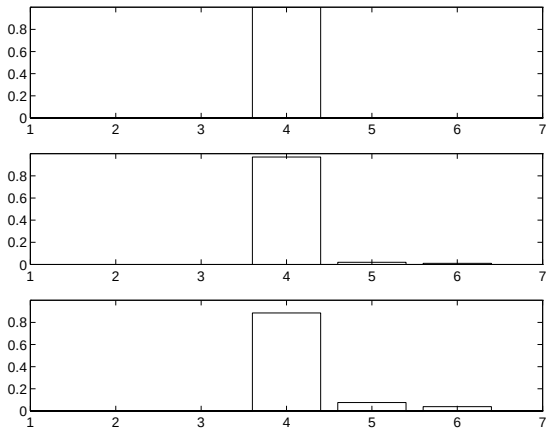

Figure 2: Posterior distribution $p(M|\mathbf{y})$, $M = 4$, for $10, 5, 0dB$ (top to bottom)

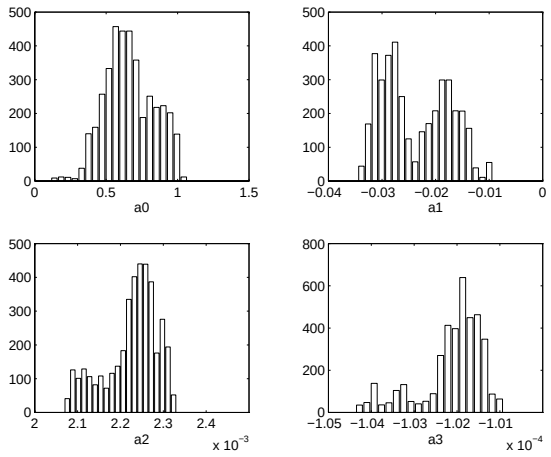

Figure 3: Estimation of the posterior distribution $p(a_i|\mathbf{y})$, $M = 4$, for $0dB$

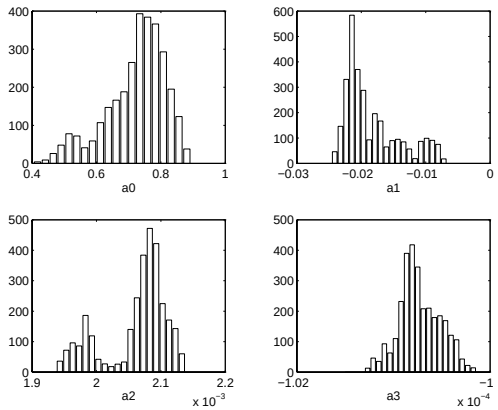

Figure 4: Estimation of the posterior distribution $p(a_i|\mathbf{y})$, $M = 4$, for $5dB$

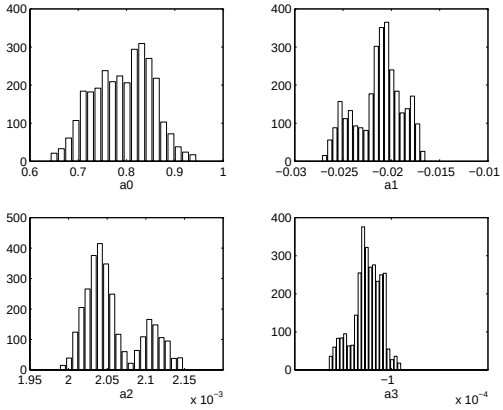

Figure 5: Estimation of the posterior distribution $p(a_i|\mathbf{y})$, $M = 4$, for $10dB$

6. CONCLUSION

A solution to order determination and parameters estimation of polynomial phase signals has been provided. Analytic posterior distribution of the parameters cannot be workable, so a solution combining M-H algorithm to estimate polynomial parameters and a reversible jump algorithm to determine the order is proposed. The algorithm easy to implement and gives simultaneous estimation of all the parameters required. Simulated results are very relevant.

7. REFERENCES

[1] S. Chib and E. Greenberg. Understanding the Metropolis-Hastings algorithm. *The American Statistician*, 49:327–335, 1995.

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

[3] S. M. Kay. *Fundamentals of Statistical Signal Processing : Estimation Theory*. Signal Processing Series. Prentice Hall, 1993.

[4] S. Peleg and B. Friedlander. The discrete polynomial-phase transform. *IEEE Transactions on Signal Processing*, 43(8):1901–1912, August 1995.

[5] S. Peleg and B. Porat. The Cramer-Rao Lower Bound for Signals with Constant Amplitude and Polynomial Phase. *IEEE Transactions on Signal Processing*, 39(3):749–752, March 1991.

[6] A.P. Quinn. *Bayesian Point Inference in Signal Processing*. PhD thesis, Cambridge University Engineering Dept., 1992.

[7] C. Theys, M. Vieira, and A. Ferrari. Bayesian Estimation of the Parameters of a Polynomial Phase Signal using MCMC methods. *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 3553–3556, 1997.