Sequential MCMC for Spatial Signal Separation and Restoration From An Array of Sensors

William Ng, James P. Reilly[1], Jean-René Larocque

Department of Electrical and Computer Engineering,
McMaster University, 1280 Main St. W.,
Hamilton, Ontario,
Canada L8S 4K1

Abstract. This paper addresses the implementation of sequential Markov Chain Monte Carlo (MCMC) estimation, also known as particle filtering, to signal separation and restoration problems, using a passive array of sensors. This proposed method offers significant advantages: 1) the signals mixed at the array can be well-separated in space and restored in an online fashion, 2) the assumption of a stationary environment over the interval can be relaxed, 3) the estimated joint posterior distribution of all the unknown parameters can be used for statistical inference, and 4) the method can also be used to dynamically detect the number of signals throughout the observation period. The signals used in the simulation were mixed by a highly-nonlinear but structured steering-vector matrix. Simulation results demonstrated the effectiveness of the method in such a way that the true and restored signals were clearly separated and restored by the sequential MCMC method.

INTRODUCTION

In many practical signal processing applications, signals are collected at an array of sensors for use in radar, sonar, communications, geophysical exploration, astrophysical exploration, and biomedical signal processing. These independent sensors are placed at different points in space to “listen” to the received signal. In effect, the sensors provide a means of sampling the received signal in space. The set of sensor outputs collected at a particular instant of time constitutes a snapshot.

In any event, spatial filtering, known as *beamforming*, is used in the array systems to distinguish between the spatial properties of signals and noise, and to separate signals from different spatial locations or directions. The device used to do the beamforming is known as a *beamformer*. The objective is to estimate the desired signal arriving from a known direction in the presence of noise and interfering signals, which arrive from different spatial locations or directions. A primitive type of beamforming operation is shown in Figure 1.

In recent decades, adaptive array processing has been widely used, and the theory of adaptive beamforming has been well developed and a variety of efficient algorithms has been proposed. These algorithms may be classified into two classes: data independent and statistically optimum, depending on how the weights are chosen. Until recently, many of these algorithms were developed upon an assumption that the directions of arrival (DOAs) of the sources were stationary, i.e., they did not move throughout the entire observation period, such that temporal averaging could be incorporated. In addition to knowledge of the number of sources, knowledge of the target source as well as the locations where the algorithms place nulls might also be needed to protect the target on the one hand and to suppress the impact of the interferences on the other. Methods [1] like MVDR, GSC, Frost’s Adaptive Algorithm, etc., are just few examples that require these assumptions to be valid for acceptable performance in spatial filtering. Unfortunately, in reality these

[1] J. Reilly, corresponding author: ph: 905 525 9140 x22895, fax: 905 521 2922, email: reillyj@mcmaster.ca

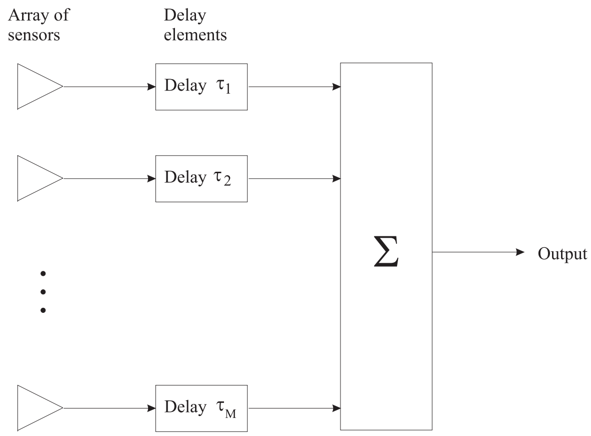

FIGURE 1. A delay and sum beamformer.

assumptions might be easily violated: the number of sources may not be available, the sources may vary from observation to observation, and there may be error in the specified DOA of the desired source, which can result in degraded performance.

To relax the assumptions about stationary sources and the knowledge of the number of sources, some methods have proposed to estimate the number of sources and to track the motions within a sliding window. The PASTd [2] algorithm assumes that the sources are stationary within the window, and estimates a subspace that must remain almost constant over the duration of the window, and thus fails to track the sources if their DOAs change quickly. On the other hand, the method proposed in [3] takes the movement of the sources into account within the window such that the beamformer for source localization and tracking in a nonstationary environment can yield useful results. However, this method can be computationally intensive, since a multi-dimensional surface must be searched to determine the global maxima that correspond to the locations of the sources. Moreover, if the surface consists of many local maxima, the performance will be degraded.

The performance of some algorithms are sensitive to the exact knowledge about the target, so their performance deteriorates significantly if the assumed DOA is different from the actual one. Some robustness [4] has been introduced to the algorithms in such a way that the main lobe width is traded off with the severe degradation caused by the deviation of the assumed and actual DOAs.

Many problems, including the problem of estimation of DOAs and amplitudes, can be modeled as *state-space models* that consist of a transition equation and an observation equation. The aim is to estimate the state process using the observations that update the posterior distribution of interest as new observations arrive. Classical methods to obtain approximations to the desired distributions include analytical approximations, such as the extended Kalman filter [5], the Gaussian sum filter [6], and deterministic numerical integration techniques [7]. The extended Kalman filter and Gaussian sum filter are computationally cheap, but fail in some difficult circumstances.

The method proposed in this paper uses the sequential MC (Monte Carlo) methods in conjunction with the MCMC (Markov Chain Monte Carlo) methods [8] [9] that have recently emerged as useful methods in the signal processing areas. They are Bayesian methods based on the idea of numerically sampling posterior distributions of interest that are difficult or impossible to handle analytically. Using the histogram so obtained from the samples, statistical inferences on parameters of interest can be made. The sequential MC methods (also known as *particle filters*) [5] [10] are suitable for estimating the state process using the observations and recursively updating the posterior distribution of interest as new observations arrive.

This paper proposes the application of particle filters to joint detection, estimation and tracking of an unknown and time-varying number of sources, using an array of sensors with the objective of estimating the waveforms of the desired sources. The proposed method is therefore an alternate approach to the beamforming

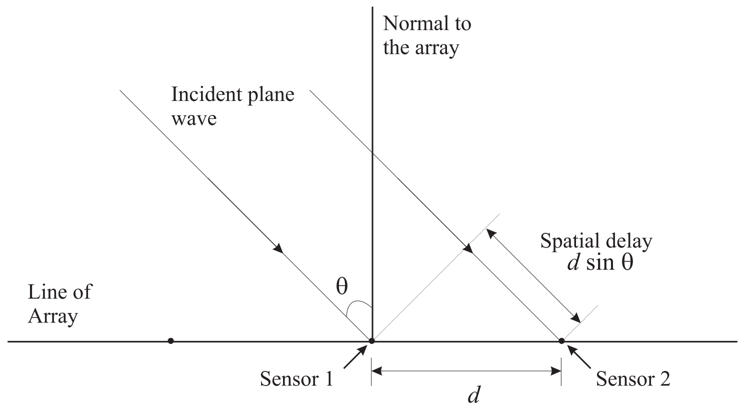

FIGURE 2. Spatial delay incurred when a plane wave impinges on a linear array.

problem. There are several advantages offered by this approach. Firstly, unlike other conventional methods, like MDL and AIC criteria, the proposed approach offers robust detection of the model order in nonstationary environments. This is in contrast to other methods which assume the model order is known. Secondly, given an appropriate model governing the motions of the DOAs, unlike other conventional beamforming approaches that require knowledge of the target DOA, the particle filtering approach can estimate all the parameters describing all incident signals. We can therefore extract information regarding any *set* of signals, rather than only the desired signal as in the beamforming case. Thirdly, the approach estimates the posterior distribution of the parameters given all past data. This distribution can be marginalized to yield the "instantaneous" posterior distribution of the desired parameters at the current time instant. Thus we need not assume stationarity. This is in contrast to other methods which involve estimation of second- or high-order statistics by temporal averaging, a process which requires stationarity over an appropriate interval. Finally, in contrast to other methods, because the MCMC techniques provide an approximation of the entire distribution of interest, one can easily make inferences on other statistics of the parameters.

This paper is organized as follows. We first present the state-space model, a brief derivation of the necessary distributions, and the derivation of the sequential update of the target posterior distribution. Description of the moves in the reversible jump MCMC that introduces statistical diversity will be given, followed by the simulation results and conclusions.

THE STATE-SPACE MODEL

In this section, most of the details about the derivation of the formulae will not be provided; readers are referred to [11] [12].

The State-Space Model

For simplicity, the transmission medium is assumed to be isotropic and nondispersive so that the radiation propagates in straight lines, and the sources are assumed *narrowband* and in the *far-field* of the array. In other words, the radiation impinging on the array is in the form of a sum of plane waves (see Figure 2).

Assume there exist $k(t)$ plane waves at time t impinging onto an M-element array from distinct directions, where $M > k(t)$. The quantity $k(t)$, which is also model order, may vary throughout the observation interval. The sources are emitted from narrowband sources. The signals are assumed to have the same known center frequency and hence the effect of a time delay on the received waveforms is simply a phase shift. Denoting a DOA vector by $\boldsymbol{\phi}(t) \in (0, 2\pi]^{k(t)}$, we define the *steering matrix* by $\boldsymbol{S}(\boldsymbol{\phi}(t)) \in \mathcal{C}^{M \times k(t)}$ as follows:

$$\boldsymbol{S}(\boldsymbol{\phi}(t)) \quad = \quad [\boldsymbol{s}(\phi_{t,1}), \boldsymbol{s}(\phi_{t,2}), \ldots, \boldsymbol{s}(\phi_{t,k})], \tag{1}$$

where $k = 1, 2, ..., k(t)$. Each column of $\boldsymbol{S}(\boldsymbol{\phi}(t))$ is the steering vector corresponding to a particular source, defined as follows:

$$\boldsymbol{s}(\phi_{t,k}) \quad = \quad [e^{-jd_1\xi_k}, e^{-jd_2\xi_k}, \ldots, e^{-jd_M\xi_k}]^T, \tag{2}$$

where

$$\xi_k = (2\pi/\lambda)\sin\phi_{t,k}, \tag{3}$$

and $\phi_{t,k}$ is the kth incident angle impinging onto the array, d_m for $m = 1, 2, \ldots, M$ is the position of the mth sensor, and λ is the wavelength. Each column $\boldsymbol{s}(\phi_{t,k})$ represents the response of the array to a plane wave from direction $\phi_{t,k}$. Denote a complex vector of observations by $\boldsymbol{y}(t) \in \mathcal{C}^M$ that represents the data received by a linear array of sensors at tth snapshot, and a complex vector of amplitudes of the sources at the tth instant by $\boldsymbol{a}(t) \in \mathcal{C}^{k(t)}$. We adopt a first order state-space hidden Markov model in the sequential sampling approach. It is assumed that the states $[\boldsymbol{\phi}(t), \boldsymbol{a}(t)]$ evolve according to the following equations:

$$\boldsymbol{\phi}(t) \quad = \quad \boldsymbol{\phi}(t-1) + \sigma_w \boldsymbol{w}(t), \tag{4}$$

$$\boldsymbol{a}(t) \quad \sim \quad \mathcal{N}(\boldsymbol{0}, \delta^2\sigma_w^2 \left[\boldsymbol{S}^H(\boldsymbol{\phi}(t))\boldsymbol{S}(\boldsymbol{\phi}(t))\right]^{-1}), \tag{5}$$

whereas the observation equation is defined as:

$$\boldsymbol{y}(t) \quad = \quad \boldsymbol{S}(\boldsymbol{\phi}(t))\boldsymbol{a}(t) + \sigma_n \boldsymbol{v}(t), \tag{6}$$

where the noise variables $\boldsymbol{v}(t) \in \mathcal{C}^M$ and $\boldsymbol{w}(t) \in \mathcal{R}^{k(t)}$ are Gaussian variables with zero mean and unit variance, the noise variances σ_w and σ_v are unknown, and the hyperparameter δ^2 is set to an a priori estimate of the SNR [13]. The model order $k(t)$ can be described by:

$$k(t) \quad = \quad k(t-1) + \epsilon_k(t), \tag{7}$$

where the $\epsilon_k(t)$ are discrete *iid* variables such that

$$\begin{aligned} P(\epsilon_k(t) = -1) &= h/2, \\ P(\epsilon_k(t) = 0) &= 1 - h, \\ P(\epsilon_k(t) = 1) &= h/2, \end{aligned} \tag{8}$$

where $h \in [0,1]$ [12].
We define the following parameter vector $\boldsymbol{\theta}_{1:t}$:

$$\boldsymbol{\theta}_{1:t} \triangleq (\{\boldsymbol{\phi}_{k(t)}\}_{1:t}, \{\boldsymbol{a}_{k(t)}\}_{1:t}, k_{1:t}, \sigma_v^2, \sigma_w^2), \tag{9}$$

where the notation $(\cdot)_{1:t}$ indicates all values of the parameters over the time interval $[1, t]$. We have the following posterior distribution:

$$\pi(\boldsymbol{\theta}_{1:t}) \quad \propto \quad p(\boldsymbol{y}_{1:t}|\boldsymbol{\theta}_{1:t})p(\boldsymbol{\theta}_{1:t}), \tag{10}$$

where $p(\boldsymbol{y}_{1:t}|\boldsymbol{\theta}_{1:t})$ is the likelihood function and $p(\boldsymbol{\theta}_{1:t})$ is the prior distributions of the parameters. Assuming the observations, given the states, are *iid*, the conditional update likelihoods of the states are also *iid*, and further assuming the distribution of the initial states is uniform, we can express the individual distributions, using the Markov properties, as follows:

$$p(\boldsymbol{y}_{1:t}|\boldsymbol{\phi}_{1:t}, \boldsymbol{a}_{1:t}, k_{1:t}, \sigma_v^2, \sigma_w^2) \quad = \quad \prod_{l=1}^{t} \mathcal{N}(\boldsymbol{S}(\boldsymbol{\phi}_l)\boldsymbol{a}_l, \sigma_w^2 \boldsymbol{I}_M), \tag{11}$$

$$
\begin{aligned}
p(\boldsymbol{\phi}_{1:t}|k_{1:t},\sigma_v^2) &= \prod_{l=1}^{t}\mathcal{N}(\boldsymbol{\phi}_{l-1},\sigma_v^2\boldsymbol{I}_{k_l}), &(12)\\
p(\boldsymbol{a}_{1:t}|\boldsymbol{\phi}_{1:t},k_{1:t},\sigma_w^2) &= \prod_{l=1}^{t}\mathcal{N}(\boldsymbol{0},\delta^2\sigma_w^2\left[\boldsymbol{S}^H(\boldsymbol{\phi}_l)\boldsymbol{S}(\boldsymbol{\phi}_l)\right]^{-1}), &(13)\\
p(k_{1:t}) &= \prod_{l=1}^{t}p(k_l|k_{l-1}) = \prod_{l=1}^{t}\epsilon_k(l). &(14)
\end{aligned}
$$

The prior distributions of the noise variances are both assumed to follow the inverse Gamma distribution, which is the conjugate distribution for the Normal distribution as follows:

$$
\begin{aligned}
p(\sigma_v^2) &\sim \mathcal{IG}(\frac{\alpha_0}{2},\frac{\gamma_0}{2}), &(15)\\
p(\sigma_w^2) &\sim \mathcal{IG}(\alpha_1,\gamma_1), &(16)
\end{aligned}
$$

where $\alpha_0,\alpha_1,\gamma_0,\gamma_1$ are hyperparameters.

It can be shown [12] that the target distribution $\pi(\boldsymbol{\theta}_{1:t})$ can be represented as follows:

$$
\begin{aligned}
\pi(\boldsymbol{\theta}_{1:t}) \propto\ & \prod_{l=1}^{t}\frac{1}{\sigma_w^{2^{k_l}}\pi^{k_l}}\exp\left[\frac{-1}{\sigma_w^2}(\boldsymbol{a}_l-\boldsymbol{m}_{\boldsymbol{a}_l})^H\boldsymbol{\Sigma}_{\boldsymbol{a}_l}^{-1}(\boldsymbol{a}_l-\boldsymbol{m}_{\boldsymbol{a}_l})\right]\\
\times\ & \prod_{l=1}^{t}\frac{1}{\sigma_w^{2^M}\delta^{2^{k_l}}}\exp\left[\frac{-1}{\sigma_w^2}\boldsymbol{y}_l^H\boldsymbol{P}_S^{\perp}(\boldsymbol{\phi}_l)\boldsymbol{y}_l\right]\\
\times\ & \prod_{l=1}^{t}\frac{|\boldsymbol{S}^H(\boldsymbol{\phi}_l)\boldsymbol{S}(\boldsymbol{\phi}_l)|}{\sigma_v^{2^{k_l/2}}(2\pi)^{k_l/2}}\exp\left[\frac{-1}{2\sigma_v^2}(\boldsymbol{\phi}_l-\boldsymbol{\phi}_{l-1})^H(\boldsymbol{\phi}_l-\boldsymbol{\phi}_{l-1})\right]\\
\times\ & \sigma_v^{2^{(-\frac{\alpha_0}{2}-1)}}\exp\left[\frac{-\gamma_0}{2\sigma_v^2}\right]\times\sigma_w^{2^{(-\alpha_1-1)}}\exp\left[\frac{-\gamma_1}{\sigma_w^2}\right]\times\prod_{l=1}^{t}p(k_l|k_{l-1}), \qquad (17)
\end{aligned}
$$

where

$$
\begin{aligned}
\boldsymbol{\Sigma}_{\boldsymbol{a}_l}^{-1} &= \boldsymbol{S}^H(\phi_l)\boldsymbol{S}(\phi_l)(1+1/\delta^2), &(18)\\
\boldsymbol{m}_{\boldsymbol{a}_l} &= \boldsymbol{\Sigma}_{\boldsymbol{a}_l}\boldsymbol{S}^H(\phi_l)\boldsymbol{y}_l, &(19)
\end{aligned}
$$

and

$$
\boldsymbol{P}_S^{\perp}(\phi_l) = \boldsymbol{I}-\frac{\boldsymbol{S}(\phi_l)[\boldsymbol{S}^H(\phi_l)\boldsymbol{S}(\phi_l)]^{-1}\boldsymbol{S}^H(\phi_l)}{(1+1/\delta^2)}, \qquad (20)
$$

Eq. (19) is a *maximum aposteriori* (MAP) estimate of the amplitudes $\boldsymbol{a}_l$, which implies that the amplitudes need not be included in the particle filter but can be estimated at each iteration after the sampling of other parameters. Finally, integrating out $\boldsymbol{a}_l$ analytically from (17) yields

$$
\begin{aligned}
\pi(\boldsymbol{\alpha}_{1:t}) \propto\ & \prod_{l=1}^{t}\frac{1}{\sigma_w^{2^M}(1+\delta^2)^{k_l}}\exp\left[\frac{-1}{\sigma_w^2}\boldsymbol{y}_l^H\boldsymbol{P}_S^{\perp}(\phi_l)\boldsymbol{y}_l\right]\\
\times\ & \prod_{l=1}^{t}\frac{1}{\sigma_v^{2^{k_l/2}}(2\pi)^{k_l/2}}\exp\left[\frac{-1}{2\sigma_v^2}(\boldsymbol{\phi}_l-\boldsymbol{\phi}_{l-1})^H(\boldsymbol{\phi}_l-\boldsymbol{\phi}_{l-1})\right]\\
\times\ & \sigma_v^{2^{(-\frac{\alpha_0}{2}-1)}}\exp\left[\frac{-\gamma_0}{2\sigma_v^2}\right]\times\sigma_w^{2^{(-\alpha_1-1)}}\exp\left[\frac{-\gamma_1}{\sigma_w^2}\right]\times\prod_{l=1}^{t}p(k_l|k_{l-1}), \qquad (21)
\end{aligned}
$$

where

$$
\boldsymbol{\alpha}_{1:t} \triangleq (\boldsymbol{\phi}_{1:t},k_{1:t},\sigma_v^2,\sigma_w^2). \qquad (22)
$$

Eq. (21) is a much simpler posterior distribution in terms of the remaining parameters. It is also possible to obtain the MAP estimators of the variance parameters as follows:

$$\sigma^2_{v,MAP} = \frac{\frac{\gamma_0}{2} + \frac{1}{2}\sum_{l=1}^{t}\left(\boldsymbol{\phi}_l - \boldsymbol{\phi}_{l-1}\right)^H\left(\boldsymbol{\phi}_l - \boldsymbol{\phi}_{l-1}\right)}{\frac{\nu_0}{2} + \frac{1}{2}\sum_{l=1}^{t} k(l) + 1}, \tag{23}$$

$$\sigma^2_{v,MAP} = \frac{\gamma_1 + \sum_{l=1}^{t}\boldsymbol{y}_l^H \boldsymbol{P}_S^{\perp}(\boldsymbol{\phi}_l)\boldsymbol{y}_l}{\nu_1 + Mt + 1}. \tag{24}$$

SEQUENTIAL MC (SMC)

Sequential Importance Sampling (SIS)

The Bayesian importance sampling can be used generate a numerical approximation corresponding to an arbitrary distribution of interest. Let's consider a function $f(\boldsymbol{x})$ for $\boldsymbol{x}$. The expected value of the function over a probability distribution function $\pi(\boldsymbol{x})$ is defined as:

$$I_f = E_{\pi(\boldsymbol{x})}(f(\boldsymbol{x})) = \int \pi(\boldsymbol{x}) f(\boldsymbol{x}) d\boldsymbol{x}. \tag{25}$$

Let's assume that we are able to draw N *iid* samples (particles) $\{\boldsymbol{x}^{(i)}; i = 1, 2, \ldots, N\}$ from the distribution $\pi(\boldsymbol{x})$. Using Monte Carlo integration, we can numerically estimate this distribution as follows:

$$\hat{\pi}_N(d\boldsymbol{x}) = \frac{1}{N}\sum_{i=1}^{N} \delta_{\boldsymbol{x}^{(i)}} d\boldsymbol{x}, \tag{26}$$

where $d\boldsymbol{x}$ is a small, finite region surrounding an $\boldsymbol{x}$ of interest and $\delta_{\boldsymbol{x}^{(i)}}$ is an indicator function defined as:

$$\delta_{\boldsymbol{x}^{(i)}}(d\boldsymbol{x}) = \begin{cases} 1, & \text{if } \boldsymbol{x}^{(i)} \in d\boldsymbol{x}, \\ 0, & \text{otherwise.} \end{cases} \tag{27}$$

As a result, we can estimate the expected value as follows:

$$\hat{I}_{f,N} = \int \hat{\pi}_N(\boldsymbol{x}) f(\boldsymbol{x}) d\boldsymbol{x} = \frac{1}{N}\sum_{i=1}^{N} f(\boldsymbol{x}^{(i)}). \tag{28}$$

According to the strong law of large numbers (SLLN) with $N \to +\infty$, $\hat{I}_{f,N}$ converges to I_f. The advantage of the MC integration method is clear. One can easily and efficiently estimate I_f and other statistical inferences based on the set of particles $\{\boldsymbol{x}^{(i)}; i = 1, 2, \ldots, N\}$. Unfortunately, it might be impossible to draw samples directly from the desired distribution $\pi(\boldsymbol{x})$. Instead, N samples are drawn from another "easy-to-sample" function $q(\boldsymbol{x})$, called "importance function[2]," whose support includes that of $\pi(\boldsymbol{x})$. The histogram of these particles approximates the distribution $q(\boldsymbol{x})$. Denote the importance weight by $w(\boldsymbol{x}^{(i)})$ as follows:

$$w(\boldsymbol{x}^{(i)}) \propto \frac{\pi(\boldsymbol{x}^{(i)})}{q(\boldsymbol{x}^{(i)})} \tag{29}$$

such that the target distribution $\pi(\boldsymbol{x})$ can be approximated by a histogram as follows:

$$\begin{aligned}\hat{\pi}^*_N(d\boldsymbol{x}) &= \frac{\sum_{i=1}^{N} w(\boldsymbol{x}^{(i)})\delta_{\boldsymbol{x}^{(i)}}(d\boldsymbol{x})}{\sum_{i=1}^{N} w(\boldsymbol{x}^{(i)})}, \\ &= \sum_{i=1}^{N} \tilde{w}(\boldsymbol{x}^{(i)})\delta_{\boldsymbol{x}^{(i)}}(d\boldsymbol{x}),\end{aligned} \tag{30}$$

[2] Because $f(\boldsymbol{x})$ is sampled nonuniformly with the density $q(\boldsymbol{x})$, some samples $\boldsymbol{x}$ have more "importance" than others as a result, and hence $q(\boldsymbol{x})$ is called importance function.

where

$$\tilde{w}(\boldsymbol{x}^{(i)}) = \frac{w(\boldsymbol{x}^{(i)})}{\sum_{i=1}^{N} w(\boldsymbol{x}^{(i)})}, \tag{31}$$

is the normalized importance weight. The expected value of the function $f(\boldsymbol{x})$ can now be estimated as follows:

$$\hat{I}^{*}_{f,N} = \int \hat{\pi}^{*}_{N}(\boldsymbol{x}) f(\boldsymbol{x}) d\boldsymbol{x} = \frac{1}{N} \sum_{i=1}^{N} w(\boldsymbol{x}^{(i)}) f(\boldsymbol{x}^{(i)}). \tag{32}$$

It can be shown [8] [13] [14] [15] that even though $\hat{\pi}^{*}_{N}(\boldsymbol{x})$ is biased, the expectation $\hat{I}^{*}_{f,N}$ of any function $f(\boldsymbol{x})$ over $\hat{\pi}^{*}_{N}(\boldsymbol{x})$ converges to I_f as $N \to +\infty$.

Sequential Update of The Posterior Distribution

The key in SMC is to approximate the posterior distribution in the form of (30) recursively and sequentially, based upon the arrival of new observations. In this section a recursive update of this approximation will be given in order to retain the previously tracked trajectories of the particles. It can be shown [12] that an *optimal* importance function that minimizes the variance of the weights and has support including that of the target distribution is given by:

$$q_{optimal}(\cdot) = q(\boldsymbol{\alpha}_t^{(i)} | \boldsymbol{\alpha}_{t-1}^{(i)}, \boldsymbol{y}_t),$$

such that the following property can be satisfied:

$$q(\boldsymbol{\alpha}_{1:t} | \boldsymbol{y}_{1:t}) = q(\boldsymbol{\alpha}_{1:t-1} | \boldsymbol{y}_{1:t-1}) q(\boldsymbol{\alpha}_t | \boldsymbol{\alpha}_{1:t-1}, \boldsymbol{y}_{1:t}). \tag{33}$$

Denoting the importance weight $w^{(i)}(t)$ by:

$$w^{(i)}(t) = \frac{\pi(\boldsymbol{\alpha}_{1:t})}{q(\boldsymbol{\alpha}_{1:t} | \boldsymbol{y}_{1:t})},$$

and using the Markov properties of the model and the *iid* assumptions on the noise variables, it can be shown [10] [16]

$$w^{(i)}(t) \quad = \quad \tilde{w}^{(i)}(t-1) \times \frac{p(\boldsymbol{y}_t | \boldsymbol{\alpha}_t^{(i)}) p(\boldsymbol{\alpha}_t^{(i)} | \boldsymbol{\alpha}_{t-1}^{(i)})}{q(\boldsymbol{\alpha}_t^{(i)} | \boldsymbol{\alpha}_{1:t-1}^{(i)}, \boldsymbol{y}_{1:t})}, \tag{34}$$

where $\tilde{w}^{(i)}(t-1)$, which absorbs the normalizing component $p(\boldsymbol{y}_t | \boldsymbol{y}_{1:t-1})$, is defined as follows:

$$\tilde{w}^{(i)}(t-1) = \frac{w^{(i)}(t-1)}{\sum_{i=1}^{N} w^{(i)}(t-1)}. \tag{35}$$

A major difficulty with the SIS procedure is that the recursion of (35) degenerates quickly after a few iterations in such a way that all but a few of the normalized weights are very close to zero. As a result, any estimate based on these very few significant particles would show a large variance. Therefore, in addition to the use of the optimal importance function it is necessary to introduce other procedures to regenerate the statistical diversity of the samples.

Resampling is an idea to eliminate the particles which have weak normalized importance weights and to multiply particles with strong importance weights. The most popular resampling scheme is Sampling Importance Resampling (SIR), which is to resample the particles according to their respective importance weights. It can be shown [17] that the resampling can be done very efficiently with order (N) operations. Unfortunately, the particles with high importance weights are statisically selected many times, limiting the true statistical diversity amongst the particles. This is the classical problem of depletion of samples, resulting in the situation where a cloud of particles may eventually collapse to a single particle.

An efficient way of limiting sample depletion consists of using a reversible jump MCMC step [8] [10] [18] on each particle at time t. The MCMC procedure provides samples from the posterior distribution of $\boldsymbol{\phi}(t)$ given the other parameters, thus introducing the desired statistical diversity. The reversible jump process is capable of exploring parameter spaces of varying dimension and thus samples model order as well as the other desired parameters.

In summary, the SMC approach is basically a combination of sequential Bayesian importance sampling, sampling importance resampling, and an MCMC step. We summarize these steps in the following table:

Sequential MC Algorithm

1. *Sequential Importance Sampling Step*
   (a) Sampling N particles of $\boldsymbol{\alpha}_t^{(i)}$ for $i = 1, 2, \ldots, N$ from the importance function as follows:

$$\boldsymbol{\alpha}_t^{(i)} \sim q(\boldsymbol{\alpha}_t^{(i)} | \boldsymbol{\alpha}_{0:t-1}^{(i)}, \boldsymbol{y}_{1:t})$$

   (b) Evaluation of the importance weights for N particles as follows:

$$w^{(i)}(t) = \tilde{w}^{(i)}(t-1) \times \frac{p(\boldsymbol{y}_t | \boldsymbol{\alpha}_t^{(i)}) p(\boldsymbol{\alpha}_t^{(i)} | \boldsymbol{\alpha}_{t-1}^{(i)})}{q(\boldsymbol{\alpha}_t^{(i)} | \boldsymbol{\alpha}_{1:t-1}^{(i)}, \boldsymbol{y}_{1:t})}, \tag{36}$$

   and hence the normalized importance weights as follows:

$$\tilde{w}^{(i)}(t) = \frac{w^{(i)}(t)}{\sum_{j=1}^{N} w^{(j)}(t)} \tag{37}$$

2. *Resampling/Selection of the Particles*
   (a) Sample a vector of index $\boldsymbol{l}$ distributed as:

$$P(l(j) = i) = w^{(i)}(t) \tag{38}$$

   (b) Resample the particles with the index vector:

$$\boldsymbol{\phi}_{0:k}^{(i)} = \boldsymbol{\phi}_{0:k}^{(l(i))} \tag{39}$$

   (c) Re-assign all the weights to $w^{(i)}(t) = \frac{1}{N}$.
3. *The Reversible Jump MCMC Step*
   Follow the update move described in the next section to introduce diversity and facilitate detection of model order.

THE REVERSIBLE JUMP MCMC DIVERSITY STEP

The reversible jump MCMC process is a variation of the Metropolis-Hasting (MH) algorithm. The algorithm inherently sets up a Markov chain whose invariant distribution corresponds to the posterior of interest. In this application, we use the MH method to sample the posterior distribution with respect to $\boldsymbol{\phi}(t)$. Assume at ith iteration of the chain we are in state $\boldsymbol{\phi}^{(i)}$. A candidate $\boldsymbol{\phi}^{\star}$ for the next state of the chain is drawn at random from a proposal distribution $q(.|.)$ to be defined later, which may be conditional on $\boldsymbol{\phi}^{(i)}$. An acceptance ratio r is computed as follows:

$$r = \frac{\pi(\boldsymbol{\phi}^{\star} | \boldsymbol{\phi}^{(i)}) q(\boldsymbol{\phi}^{(i)} | \boldsymbol{\phi}^{\star})}{\pi(\boldsymbol{\phi}^{(i)} | \boldsymbol{\phi}^{\star}) q(\boldsymbol{\phi}^{\star} | \boldsymbol{\phi}^{(i)})} \times \boldsymbol{J}, \tag{40}$$

where $\boldsymbol{J}$ is the Jacobian[3] of the transformation from $\boldsymbol{\phi}$ to $\boldsymbol{\phi}^\star$. An acceptance parameter η is then defined as

$$\eta = \min\{r, 1\}, \tag{41}$$

which can be interpreted as the probability that the candidate $\boldsymbol{\phi}^\star$ is accepted. The set of accepted candidates represents a set of samples drawn from the posterior distribution of interest. Because the model order $k_{1:t}$ is varying, the reversible jump MCMC method can be used to sample directly from the joint distribution over all model orders of interest. This method allows the sampling process to jump between subspaces of different dimensions, thus visiting all relevant model orders. To ensure the reversibility and the invariance of the Markov chain with respect to the desired posterior distribution, three moves[4] from which the candidates are sampled are used and described as follows:

- Birth Move is chosen with probability b_k for which a new source is proposed at random such that $k(t) = k(t-1)+1$.
- Death Move is chosen with probability d_k for which one of the existing source is proposed to be removed such that $k(t) = k(t-1)-1$.
- Update Move is chosen with probability $u_k = 1-b_k-d_k$ such that all particles are updated with fixed dimension $k(t) = k(t-1)$.

The selection of moves can be summarized by the following schema.

Reversible Jump MCMC

1. Current state of the chain = current state of the particles $(k(t), \boldsymbol{\phi}^{(i)}(t))$.
2. Iteration t for the ith particle, $i = 1, \ldots, N$
   - Sample $u \sim U$, where U is a uniform distribution over $[0,1]$,
   - if $(u < b_k)$ then execute a "birth move",
   - else if $(u < b_k + d_k)$ then execute a "death move",
   - else, execute an update move .
3. $t \leftarrow t+1$, goto step 2

Update Move

In this move, the model order is kept fixed, i.e., $k(t) = k(t-1)$, and the steps for such a move can be summarized as follows [12]:

1. Sample a candidate in the DOAs according to the following proposal distribution:

$$q(\boldsymbol{\phi}^{(i)}(t)|\boldsymbol{\phi}^{(i)}(t-1)) = \mathcal{N}(\boldsymbol{\phi}^{(i)}(t-1), \sigma_v^2 \boldsymbol{I}_{k(t)}), \tag{42}$$

and then evaluate the acceptance ratio by substituting (42) and (21) into (40) as follows:

$$r_{update} = \frac{\exp\left[-\frac{1}{\sigma_w^2}\boldsymbol{y}^H(t)\boldsymbol{P}_S^\perp(\phi^\star)\boldsymbol{y}(t)\right]}{\exp\left[-\frac{1}{\sigma_w^2}\boldsymbol{y}^H(t)\boldsymbol{P}_S^\perp(\phi^{(i)}(t))\boldsymbol{y}(t)\right]}, \tag{43}$$

---

[3] It can be shown that $\boldsymbol{J} = 1$ in this case.

[4] Two novel moves - *split* and *merge* - are introduced in [12]. A split move is chosen with probability s_k for which an existing source is proposed to be split into two sources, whereas a merge move, which is the reverse of a split move to maintain the reversibility condition, is chosen with probability m_k for which the existing sources are proposed to be merged. These move types are useful for handling the case where the DOA tracks of two separate sources cross each other.

and accept the candidate $\boldsymbol{\phi}^\star$ with the probability:

$$\eta_{update} = \min\{r_{update}, 1\}. \tag{44}$$

2. Estimate the amplitudes $\boldsymbol{a}(t)$ according to (19).
3. Estimate the noise variances σ_v^2 and σ_w^2 which are required in (42) and (43) according to (23) and (24), respectively.

Birth/Death moves

The birth move proposes a candidate in a higher dimension model, whereas the death move proposes a candidate in a lower dimension model. In the birth move, we assume that the current state is $(\boldsymbol{\phi}_k, k)$ and we wish to determine whether the next state is $\left(\boldsymbol{\phi}_{k+1}, k+1\right)$ at the next iteration. This involves the addition of a new source ϕ_c, which is proposed at random from the prior distribution for the directions of arrival. In [12] ϕ_c is sampled from the uniform distribution over $(0, 2\pi]$. Thus,

$$\boldsymbol{\phi}^\star_{k(t)+1} = [\boldsymbol{\phi}_k(t) | \phi_c]. \tag{45}$$

According to [12], the acceptance ratio for the birth move is:

$$r_{birth} \quad = \quad \frac{\exp\left[-\frac{1}{\sigma_w^2}\boldsymbol{y}^H(t)\boldsymbol{P}_S^\perp(\phi^\star_{k+1})\boldsymbol{y}(t)\right]}{\exp\left[-\frac{1}{\sigma_w^2}\boldsymbol{y}^H(t)\boldsymbol{P}_S^\perp(\phi^{(i)}_k)\boldsymbol{y}(t)\right]} \times \frac{1}{(1+\delta^2)(k+1)}, \tag{46}$$

where the $\boldsymbol{\phi}^\star_{k+1}$ will be accepted with a probability:

$$\eta_{birth} = \min\{r_{birth}, 1\}. \tag{47}$$

In order to maintain the invariant distribution of the reversible jump MCMC algorithm with respect to model order, the Markov chain must be *reversible* with respect to moves across subspaces of different model orders. That is, the probability of moving from model order k to $k+1$ must be equal to that of moving from $k+1$ to k. Therefore we propose a death move in which a source in the current state $\left(\boldsymbol{\phi}_{k+1}, k+1\right)$ is randomly selected to be removed such that the next state becomes $(\boldsymbol{\phi}_k, k)$ at the next iteration. A sufficient condition for reversibility with respect to model order is that the acceptance ratio for the death move be defined as:

$$r_{death} = \frac{1}{r_{birth}}, \tag{48}$$

and the new candidate of dimension k is accepted with probability:

$$\eta_{death} = \min\{\frac{1}{r_{death}}, 1\}. \tag{49}$$

The schemas for the birth and death moves are similar to those for the update move with appropriate changes, and are described in [12] in more detail.

SIMULATION RESULTS

The proposed algorithm is now applied to two sets of simulation data, generated for $K = 2$ sources with the parameters described in Tables 1 and 4, respectively. The DOAs of the sources are generated by a first-order random walk with initial values $\boldsymbol{\phi}(0)$ and variances σ_v^2, specified in these tables. Furthermore, the amplitudes used are generated by two stable[5], 7th-order autoregressive (AR) processes. The received array is circular and composed of $M = 8$ elements. An example of such an array is shown in Figure 3. Circular arrays are used more often in practice as they do not suffer from the ambiguity between the forward and backward look directions, as is inherent to linear arrays.

[5] The stable AR processes are generated by lowpass filtering a white noise process.

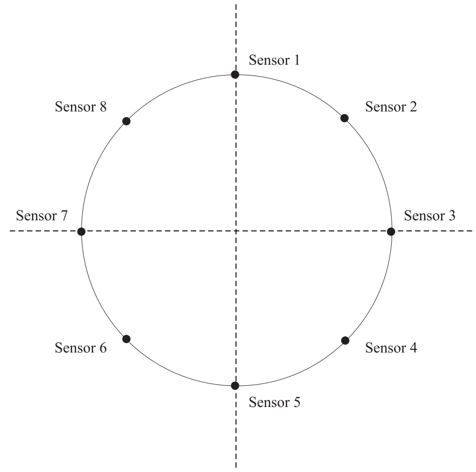

FIGURE 3. A circular array with 8 elements used in the simulation.

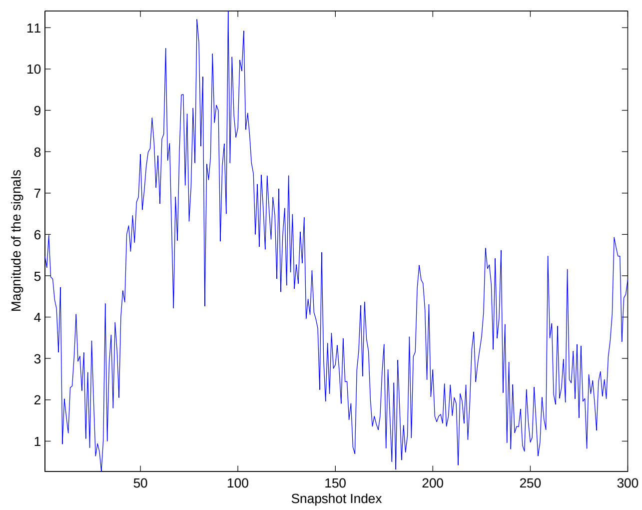

FIGURE 4. An example of the magnitude of the signals received from one sensor of the array.

Experiment 1: Widely-separated sources

In this experiment, the initial SNR is 20 dB for both sources, and 300 snapshots are generated. Figure 4 illustrates the magnitude of the signals mixed at the array input of a particular channel. The algorithm randomly initializes the unknown parameters and assigns the model order $k(1)$ to $k_{max} = M - 1 = 7$, where k_{max} is the maximum allowable model order. The particle filter uses $N = 100$ particles throughout the experiment. The entire trajectories of the estimates of the DOAs and amplitudes are shown in Figures. 6 and 7.

TABLE 1. Parameters of the state-space model for simulated data.

Parameter	M	K	σ_v^2	σ_w^2	$\boldsymbol{\phi}(0)$	$\boldsymbol{a}(0)$
Value	8	2	${5^\circ}^2$	0.083	$[80^\circ, 200^\circ]$	$[4, 5]$

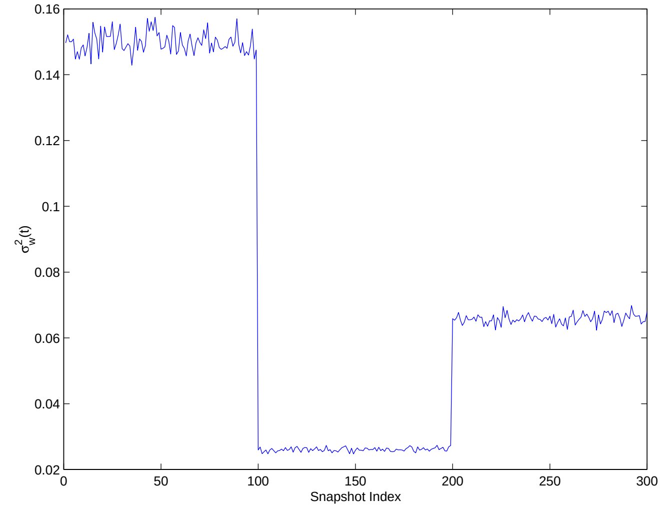

FIGURE 5. An observation noise process with time-varying variances.

TABLE 2. The average SNRs in the different time ranges.

Time range	SNR (dB)
1-100	18.56
101-200	22.32
201-300	19.78

In order to show the robustness of the algorithm in nonstationary noise environments, the observation noise is generated such that the variances are varying or nonstationary within the observation period as shown in Figure 5. The average SNRs in the different time regions are summarized in Table 2.

Figure 6 shows that the DOAs of two sources are changing quite significantly between snapshots but they never cross each other. Figure 7 shows two AR processes, one of which has more rapid fluctuations than the other. Unlike the DOAs, the amplitudes overlap over each other. Because the initial model order $k(1)$ does not correspond to the correct order, the algorithm takes certain period of time or snapshots to search over subspaces of different dimensions and to detect the correct order. The trajectory of the detection of model order is shown in Figure 8. As a result, within $t \in [1, 13]$ the DOAs and amplitudes are not well estimated, but when the order is detected correctly at $t = 14$, from then on the algorithm starts to stabilize itself and produce good estimates of the DOAs and amplitudes.

According to Figure 8, the SMCMC method converges very quickly to the correct model order. Moreover, Figures 6 and 7 reveal that the SMCMC method performs very well even though the sources are moving rapidly from one snapshot to another in a highly nonstationary environment. Not only are the DOAs of the sources well tracked, but the amplitudes mixed at the array input are well separated and restored as well. Table 3 provides a quantitative measure on the performance of the algorithm using the mean-square error (MSE) between the true and estimated DOAs and amplitudes with respect to their respective true values.

TABLE 3. The MSE between the true and estimated DOAs and amplitudes for two sources

Time range	MSE for DOAs (dB)	MSE for Amplitudes (dB)
1-13	-3.3246, -8.001	1.5323, -1.9577
14-300	-30.7568, -40.0840	-28.9063, -23.1563

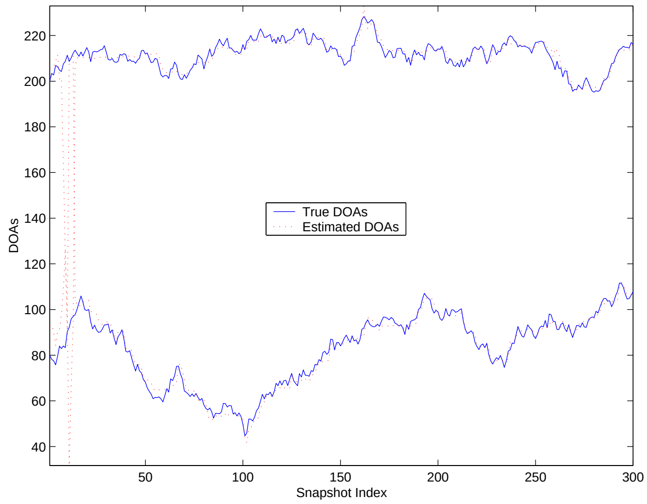

FIGURE 6. Sequential estimates of the DOAs.

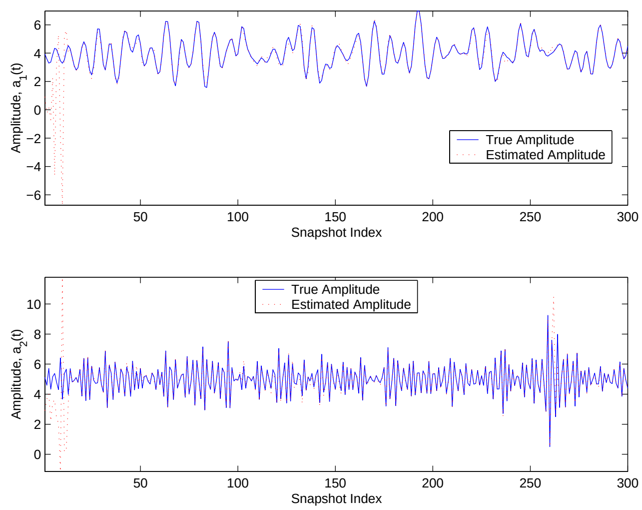

FIGURE 7. Sequential estimates of the amplitudes for the sources.

Experiment 2: Sources crossing each other

In this experiment, two sources that change rapidly between snapshots, approach and cross each other at about $t = 90$ are used (see Figure 9). As in the previous experiment, two AR processes are used for the source amplitudes as shown in Figure 10. The SNR is about 20 dB for both sources, 300 snapshots are generated and 100 particles are used for each iteration.

In this experiment, the simulated DOA trajectories evolve in three stages. Firstly, they are separated by about 50 degrees and approach each other for $t \in [1, 75]$. Secondly, for $t \in [76, 165]$ they get closer and closer

TABLE 4. Parameters of the state-space model for simulated data

Parameter	M	K	σ_v^2	σ_w^2	$\boldsymbol{\phi}(0)$	$\boldsymbol{a}(0)$
Value	8	2	$2^{\circ 2}$	0.15	$[100^\circ, 150^\circ]$	$[4, 5]$

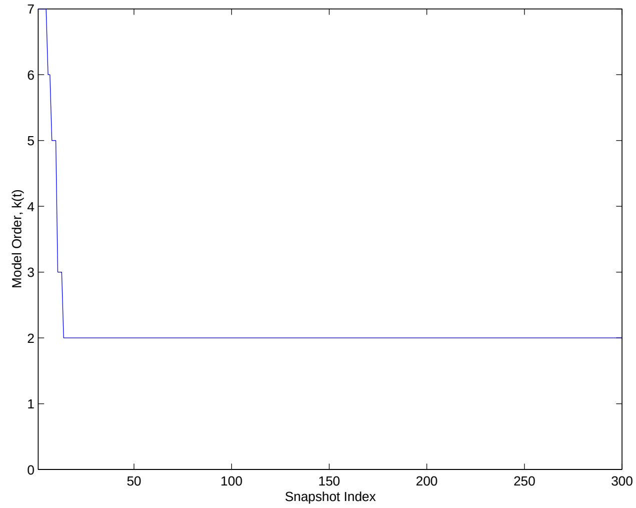

FIGURE 8. Sequential estimates of the model order.

TABLE 5. The MSE between the true and estimated DOAs for two sources

Time range	MSE for DOA Estimation
1-70	-30.0556, -34.0134
71-165	-20.9023, -5.9669
166-300	-37.3439, -33.0165

to and cross each other at $t = 90$. Eventually, they separate and widen for $t \in [166, 300]$.

Initially, the estimates in the first few iterations are off from the true values, but 10 snapshots later the system quickly stabilizes and tracks the moving DOAs properly. However, as the sources approach each other the estimation deteriorates. In the region $t \in [76, 165]$, the two sources differ by less than 20 degrees and they eventually cross each other at $t = 90$. In this region, the steering matrix $\boldsymbol{S}(\boldsymbol{\phi}(t))$ becomes poorly conditioned due to the values in $\boldsymbol{\phi}(t)$ being only slightly different, resulting in the deterioration of the DOA estimates and subsequently resulting in poor estimation of the corresponding amplitudes (see Figure (10)). Moreover, the impact of the poorly estimated amplitudes in the last iteration will be brought along to the next iteration in the estimation of the DOAs, as well as in the evaluation of the importance weights.

As the DOAs diverge beyond $t = 165$, the system stabilizes itself again due to the arrival of new data and resumes its proper track afterwards, as is evident in Figure 9. However, it takes about 5 more snapshots for the amplitude estimates to stabilize (see Figure 10). Tables 5 and 6 provide a quantitative measure on the performance of the algorithm in terms of the MSE of the estimated trajectories for these three stages. As expected, the MSE for the DOA estimation during the periods $t \in [1, 75]$ and $t \in [166, 300]$, and that for the amplitude estimation during the periods $t \in [1, 75]$ and $t \in [171, 300]$ are small, respectively. Note that, however, the MSE in the respective periods $t \in [166, 300]$ and $t \in [171, 300]$ for DOA and amplitude estimation indeed indicate how well the algorithm recovers from an ambiguous situation where the DOAs of the signals are too close during $t \in [76, 165]$. On the other hand, during $t \in [76, 165]$ the MSE for DOAs estimation on average is far better than that for the amplitude estimation process, implying that poor DOA estimation indeed causes large error in the amplitude estimation procedure.

CONCLUSION

In this paper, a particle filter that includes a reversible jump MCMC is used for joint sequential detection and estimation of an unknown number of directions of arrival. The signals that are mixed at the array input

TABLE 6. The MSE between the true and estimated amplitudes for two sources

Time range	MSE for Amplitude Estimation
1-70	-19.1120, -20.1889
71-170	0.7059, -1.0593
171-300	-26.7899, -29.7507

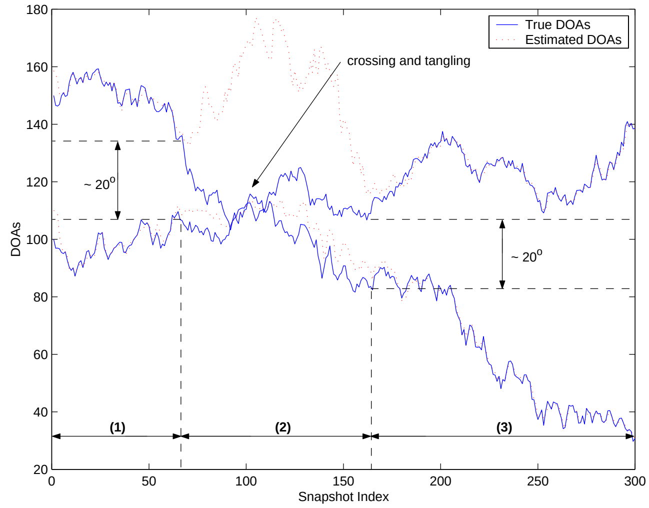

FIGURE 9. Sequential estimates of the DOAs.

can be separated and restored. The superior performance of the particle filter over other methods which use time-averaged statistics in nonstationary envirnoments has been clearly indicated.

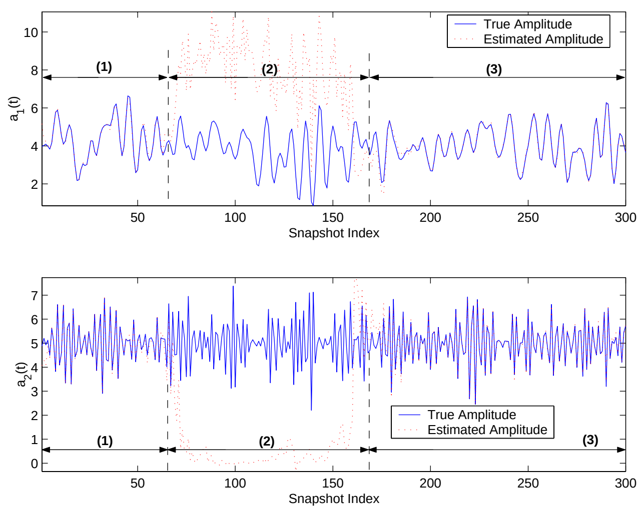

FIGURE 10. Sequential estimates of the amplitudes for the sources.

REFERENCES

1. Johnson, D., and Dudgeon, D., *Array Signal Processing: Concepts and Techniques*, PTR Prentice Hall, New Jersey, 1993.
2. Yang, B., "An Extension of the PASTd Algorithm to Both Rank and Subspace Tracking.", 1995, vol. 2, pp. 179–182.
3. Katkovnik, V., and Gershman, A., "A Local Polynomial Approximation Based Beamforming for Source Localization and Tracking in Nonstationary Environment", 2000, vol. 7.
4. Gershman, A., "Robust Aadaptive Beamforming in Sensor Arrays", 1999, vol. 53, pp. 305–314.
5. Anderson, B., and Moore, J., *Optimal Filtering*, Prentice-Hall, Englewood Cliffs, 1979.
6. Aspach, D., and Sorenson, H., "Nonlinear Bayesian Estimation Using Gaussian Sum Approximations", 1972, vol. 17, pp. 439–448.
7. Bucy, R., and Senne, K., "Digital Sysnthesis of Nonlinear Filters", 1971, vol. 7, pp. 287–298.
8. Andrieu, C., Freitas, N. D., and Doucet, A., "Sequential MCMC for Bayesian Model Selection", in *Proceedings of the Internationl Workshop on Higher Order Statistics*, Ceasarea, Israel, 1999.
9. Andrieu, C., Doucet, A., Fitzgerald, W., and Godsill, S., "An Introduction to the Theory and Applications of Simulation Based Computational Methods in Bayesian Signal Processing", in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, Seattle, WA, 1998.
10. Doucet, A., Godsill, S., and Andrieu, C., *On Sequential Monte Carlo Sampling Methods for Bayesian Filtering*, vol. 10, 2000.
11. Ng, W., Larocque, J.-R., and Reilly, J. P., "Particle filter for the tracking of an unknown number of sources", in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, Salt Lake City, UT, 2001.
12. Larocque, J.-R., Reilly, J. P., and Ng, W., "Particle filter for the tracking of an unknown number of sources", 2001, submitted for publication, also available at http://www.ece.mcmaster.ca/∼reilly.
13. Andrieu, C., and Doucet, A., "Joint Bayesian Model Selection and Estimation of Noisy Sinusoids via Reversible Jump MCMC", 1999, vol. 47, pp. 2667–2676.
14. Gilks, W., Richardson, S., and Spiegelhalter, D., *Markov Chain Monte Carlo in Practice*, Chapman and Hall, New York, 1998.
15. Ruanaidh, J. O., and Fitzgerald, W., *Numerical Bayesian Methods Applied to Signal Processing.*, Springer-Verlag, New York, 1996.
16. Larocque, J.-R., *Advanced numerical Bayesian methods for array signal processing*, Ph.D. thesis, McMaster University, Hamilton, Ontario, Canada (2001).
17. Doucet, A., On sequential simulation-based methods for Bayesian filtering., Tech. Rep. TR.310, University of Cambridge, Department of Engineering, Signal Processing Group, England (1998).
18. Green, P., "Reversible Jump Markov Chain Monte Carlo Computation and Bayesian Model Determination", 1995, vol. 82, pp. 711–732.