MONTE CARLO MIXTURE KALMAN FILTER AND ITS APPLICATION TO SPACE-TIME INVERSION

Tomoyuki Higuchi *,[1] **Jun'ichi Fukuda** **

* *The Institute of Statistical Mathematics, 4-6-7 Minami-Azabu, Minato-ku, Tokyo 106-8569, Japan*
** *Earthquake Research Institute, University of Tokyo, Yayoi 1-1-1, Bunkyo-ku, Tokyo 113-0032, Japan*

Abstract: It is important to precisely know the whole time history of various types of fault slip events to understand the physics of earthquake generation. We develop a new time dependent inversion method for imaging transient fault slips from geodetic data. Past studies employed a linear Gaussian state space model and applied Kalman filter. The Kalman filter based methods, however, do not allow any variation to the temporal smoothness (or roughness) of fault slips. In the present study, we develop/apply a new filtering scheme, Monte Carlo mixture Kalman filter (MCMKF), to the time dependent inversion. MCMKF allows variation to the temporal smoothing of slips in the following scheme; (1) we prepare a finite number of competing state space models, each of which follows a different state space model, (2) we introduce a switching structure among these competing models.

Keywords: Monte Carlo calculation, Kalman filter, Filtering techniques, State-space model, Inverse dynamic problem

1. INTRODUCTION

Recently continuous measurements of surface deformation with dense Global Positioning System (GPS) network have revealed that transient crustal deformations with a time scale of hours up to years play a very important role in seismic cycles. Accurate estimates of these spatio-temporal variation of such slow events provides us with an opportunity to understand an earthquake mechanism. From geodetic view point, it is therefore important to investigate detailed spatio-temporal process of slow events and dense GPS array record provides the most suitable data for this end (Heki *et al.*, 1997; Ozawa *et al.*, 2001).

Several studies have tried to image the spatio-temporal variation of transient fault slip (Segall and Matthews, 1997). One efficient way to retrieve slip distribution is that a space-time history of fault slip is modeled by using linear Gaussian state space model, i.e., state space model (Anderson and Moore, 1979; Kitagawa and Gersch, 1996) and estimated by Kalman filter (Kitagawa and Gersch, 1996). This method is referred to as the Network Inversion Filter (NIF) and recognized as one of the standard techniques in the application domain.

In the NIF framework, the temporal smoothness of the fault slip is controlled by a scaling parameter of the employed stochastic model. This scaling parameter, often referred to as hyper-parameter, is determined with maximum likelihood method. In the NIF framework, the scaling parameter is held fixed over the observation period. The constancy of the scaling parameter seems to obscure the causal relationship among multiple events, and hence motivated us to explore a new approach to the time dependent inversion such that the scaling parameter is variable in time.

[1] Partially supported by the MEXT, Japan

In the present study, we develop a new filtering algorithm, called as Monte Carlo mixture Kalman filter (MCMKF) for imaging time-dependent fault slip from geodetic data. In section 2 a new model to identify a temporal variation of the scaling parameter is proposed and formulated by the conditional dynamic linear model (CDLM). A basic idea of MCMKF is explained in section 3. In section 4, the recursive calculations, prediction and filtering, are derived. Section 5 describes a procedure for the state estimation based on the model averaging principle. Section 6 gives a summary of the MCMKF procedure and brief discussion on the numerical experiments.

2. CONDITIONAL DYNAMIC LINEAR MODEL (CDLM)

In order to allow temporal variation of the hyper-parameter for slip acceleration, we first prepare a finite number of competing different state space models, each of which has different hyper-parameter value. Then, we realize the temporal change of the value by introducing a switching structure among them. For the limited space, we cannot give a detailed explanation for the system and the observation models in the NIF framework (Fukuda *et al.*, 2003). In stead, we remark that a departure of our model from the NIF is a time dependency of system noise variances. Namely, our model can be rewritten as follows:

$$\boldsymbol{x}_n = F_n \boldsymbol{x}_{n-1} + \boldsymbol{v}_n, \qquad \boldsymbol{v}_n \sim N(\mathbf{0}, Q_n(I_n)) \tag{1}$$

$$\boldsymbol{y}_n = H \boldsymbol{x}_n + \boldsymbol{w}_n, \qquad \boldsymbol{w}_n \sim N(\mathbf{0}, R_n). \tag{2}$$

In this case only Q_n among the four matrices is dependent on I_n. Thus a value of I_n specifies a system model with a certain value of the hyper-parameter. The dimension of the state vector depends on a way of how a displacement region of interest is numerically represented. It usually exceeds over one hundred, and then a direct approach of the particle filter (Doucet *et al.*, 2001) to combine many generalized state space model (Kitagawa, 1998; Higuchi and Kitagawa, 2000; Higuchi, 2001) cannot deal with this problem.

This generalization to incorporate a time dependency of hyperparameters can be formulated in the conditional dynamic linear model (CDLM) (Chen and Liu, 2000; Chen *et al.*, 2000; Liu *et al.*, 2001). The CDLM can be defined as:

System model

$$\boldsymbol{x}_n = F_n(I_n)\boldsymbol{x}_{n-1} + \boldsymbol{v}_n, \tag{3}$$

Observation model

$$\boldsymbol{y}_n = H_n(I_n)\boldsymbol{x}_n + \boldsymbol{w}_n \tag{4}$$

where $\boldsymbol{v}_n \sim N(\mathbf{0}, Q_n(I_n))$ and $\boldsymbol{w}_n \sim N(\mathbf{0}, R_n(I_n))$. The indicator vector I_n is a discrete latent variable which takes an integer value between $1 \sim M$. Usually a number of models treated in the Mixture Kalman filter is about $2 \sim 3$, but we consider a problem of dealing with a large number of models, $M \simeq 100$. Given I_n, F_n, H_n, Q_n, and R_n are known matrices of appropriate dimension. The CDLM is a direct generalization of the dynamic linear model (DLM) and retain a capability of dealing with outliers, sudden jumps, clutters, and other nonlinear features (Liu *et al.*, 2001). The CDLM includes other types of generalization of DLM, e.g., Partial non-Gaussian state space model (Shephard, 1994; Bergman *et al.*, 2001), Markov switching state space model, (Kim and Nelson, 1999; Frühwirth-Schnatter, 2001) and Dynamic liner models with switching (Shumway and Stoffer, 1991).

3. BASIC IDEA OF MONTE CARLO MIXTURE KALMAN FILTER (MCMKF)

In this section, we introduce a new filtering scheme and call it as Monte Carlo mixture Kalman filter (MCMKF) that allows us to choose the optimal model from many candidates or to average over many models.

3.1 Model switching structure

The MCMKF algorithm requires a stochastic model which describes a time-dependent structure for I_n. In this study, I_n is assumed to follow a stationary Markov process, i.e.,

$$p(I_n|\boldsymbol{I}_{1:n-1}) = p(I_n|I_{n-1}) \tag{5}$$

where $\boldsymbol{I}_{i:j} = (I_i, I_{i+1}, \cdots, I_j)$ and $p()$ denotes probability density function. An evolution of I_n is realized by Markov switching model with transition probability given by

$$\pi_{ij} = \Pr(I_n = j | I_{n-1} = i) \tag{6}$$

where Pr denotes realization probability. In the following subsections, we present an algorithm that determines time evolution of I_n.

3.2 Monte Carlo mixture Kalman filter

The MCMKF algorithm consists of two steps. First, temporal variation of the probability distribution of indicator variable I_n is determined. Second, temporal variation of the probability distribution of the state vector $\boldsymbol{x}_n$ is estimated following the history of I_n.

Let $\boldsymbol{y}_{i:j}$ and $\boldsymbol{I}_{i:j}$ be a set of data vectors and indicator variable from time t_i to time t_j, respectively, i.e., $\boldsymbol{y}_{i:j} = (\boldsymbol{y}_i, \boldsymbol{y}_{i+1}, \cdots, \boldsymbol{y}_j)$ and $\boldsymbol{I}_{i:j} = (I_i, I_{i+1}, \cdots, I_j)$. In MCMKF, two conditional joint distributions of $\boldsymbol{I}_{1:n}$: (i) predictive distribution $p(\boldsymbol{I}_{1:n}|\boldsymbol{y}_{1:n-1})$ and (ii) filter distribution $p(\boldsymbol{I}_{1:n}|\boldsymbol{y}_{1:n})$, are approximated by many "particles" that can be considered as independent realizations from each distribution. Let $\boldsymbol{I}^{(j)}_{1:i|k} =$

$(I_{1|k}^{(j)}, I_{2|k}^{(j)}, \cdots I_{i|k}^{(j)})$ be the jth realization of the conditional distribution $p(\boldsymbol{I}_{1:i}|\boldsymbol{y}_{1:k})$. Each distribution is approximated by N_p ($N_p \gg 1$) realizations as follows:

$$\left\{\boldsymbol{I}_{1:n|n-1}^{(1)}, \boldsymbol{I}_{1:n|n-1}^{(2)}, \cdots, \boldsymbol{I}_{1:n|n-1}^{(N_p)}\right\} \sim p(\boldsymbol{I}_{1:n}|\boldsymbol{y}_{1:n-1}) \tag{7}$$

$$\left\{\boldsymbol{I}_{1:n|n}^{(1)}, \boldsymbol{I}_{1:n|n}^{(2)}, \cdots, \boldsymbol{I}_{1:n|n}^{(N_p)}\right\} \sim p(\boldsymbol{I}_{1:n}|\boldsymbol{y}_{1:n}) \tag{8}$$

where

$$\Pr\left(\boldsymbol{I}_{1:n} = \boldsymbol{I}_{1:n|n-1}^{(j)}|\boldsymbol{y}_{1:n-1}\right) = \frac{1}{N_p}, \tag{9}$$

$$\Pr\left(\boldsymbol{I}_{1:n} = \boldsymbol{I}_{1:n|n}^{(j)}|\boldsymbol{y}_{1:n}\right) = \frac{1}{N_p}. \tag{10}$$

In this study, we refer to $\{\boldsymbol{I}_{1:n|n-1}^{(1)}, \boldsymbol{I}_{1:n|n-1}^{(2)}, \cdots, \boldsymbol{I}_{1:n|n-1}^{(N_p)}\}$ and $\{\boldsymbol{I}_{1:n|n}^{(1)}, \boldsymbol{I}_{1:n|n}^{(2)}, \cdots, \boldsymbol{I}_{1:n|n}^{(N_p)}\}$ as "approximated predictive distribution" and "approximated filter distribution", respectively. Given realizations of I_n, $I_n^{(j)} = I_{n|k}^{(j)}$, following CDLM is satisfied for each $I_n^{(j)}$ $(j = 1, \cdots, N_p)$:

$$\boldsymbol{x}_n^{(j)} = F_n(I_n^{(j)})\boldsymbol{x}_{n-1}^{(j)} + \boldsymbol{v}_n^{(j)}, \quad \boldsymbol{v}_n^{(j)} \sim N(\boldsymbol{0}, Q_n(I_n^{(j)})) \tag{11}$$

$$\boldsymbol{y}_n = H_n(I_n^{(j)})\boldsymbol{x}_n^{(j)} + \boldsymbol{w}_n^{(j)}, \quad \boldsymbol{w}_n^{(j)} \sim N(\boldsymbol{0}, R_n(I_n^{(j)})). \tag{12}$$

Using (11) and (12), we will later show that a set of particles approximating the predictive distribution and the filter distribution is obtained recursively by two steps:

$$\text{Prediction:} \quad \left\{\boldsymbol{I}_{1:n-1|n-1}^{(1)}, \cdots, \boldsymbol{I}_{1:n-1|n-1}^{(N_p)}\right\} \longrightarrow \left\{\boldsymbol{I}_{1:n|n-1}^{(1)}, \cdots, \boldsymbol{I}_{1:n|n-1}^{(N_p)}\right\}, \tag{13}$$

$$\text{Filtering:} \quad \left\{\boldsymbol{I}_{1:n|n-1}^{(1)}, \cdots, \boldsymbol{I}_{1:n|n-1}^{(N_p)}\right\} \longrightarrow \left\{\boldsymbol{I}_{1:n|n}^{(1)}, \cdots, \boldsymbol{I}_{1:n|n}^{(N_p)}\right\}. \tag{14}$$

4. RECURSIVE CALCULATION

4.1 Prediction

In this subsection, we show that an approximated predictive distribution at time t_n $\{\boldsymbol{I}_{1:n|n-1}^{(1)}, \cdots, \boldsymbol{I}_{1:n|n-1}^{(N_p)}\}$ is obtained from an approximated filter distribution at time t_{n-1} $\{\boldsymbol{I}_{1:n-1|n-1}^{(1)}, \cdots, \boldsymbol{I}_{1:n-1|n-1}^{(N_p)}\}$. We assume that $\{\boldsymbol{I}_{1:n-1|n-1}^{(1)}, \cdots, \boldsymbol{I}_{1:n-1|n-1}^{(N_p)}\}$ and $\boldsymbol{y}_{1:n-1}$ are given. Then the probability $\Pr(\boldsymbol{I}_{1:n} = \boldsymbol{I}_{1:n|n-1}^{(j)}|\boldsymbol{y}_{1:n-1})$ is manipulated as

$$\begin{aligned}
&\Pr(\boldsymbol{I}_{1:n} = \boldsymbol{I}_{1:n|n-1}^{(j)}|\boldsymbol{y}_{1:n-1}) \\
&= \Pr(I_n = I_{n|n-1}^{(j)}, \boldsymbol{I}_{1:n-1} = \boldsymbol{I}_{1:n-1|n-1}^{(j)}|\boldsymbol{y}_{1:n-1}) \\
&= \Pr(I_n = I_{n|n-1}^{(j)}|\boldsymbol{I}_{1:n-1} = \boldsymbol{I}_{1:n-1|n-1}^{(j)}, \boldsymbol{y}_{1:n-1}) \\
&\cdot \Pr(\boldsymbol{I}_{1:n-1} = \boldsymbol{I}_{1:n-1|n-1}^{(j)}|\boldsymbol{y}_{1:n-1}) \\
&= \Pr(I_n = I_{n|n-1}^{(j)}|\boldsymbol{I}_{1:n-1} = \boldsymbol{I}_{1:n-1|n-1}^{(j)}) \\
&\cdot \Pr(\boldsymbol{I}_{1:n-1} = \boldsymbol{I}_{1:n-1|n-1}^{(j)}|\boldsymbol{y}_{1:n-1}) \\
&= \Pr(I_n = I_{n|n-1}^{(j)}|I_{n-1} = I_{n-1|n-1}^{(j)})\frac{1}{N_p}.
\end{aligned} \tag{15}$$

(15) indicates that $\left\{\boldsymbol{I}_{1:n|n-1}^{(1)}, \cdots, \boldsymbol{I}_{1:n|n-1}^{(N_p)}\right\}$ is obtained by sampling a realization $I_{n|n-1}^{(j)}$ with probability or weight $\Pr(I_n = I_{n|n-1}^{(j)}|I_{n-1} = I_{n-1|n-1}^{(j)})$, and setting $\boldsymbol{I}_{1:n|n-1}^{(j)} = (\boldsymbol{I}_{1:n-1|n-1}^{(j)}, I_{n|n-1}^{(j)})$. Note that $\Pr(I_n = I_{n|n-1}^{(j)}|I_{n-1} = I_{n-1|n-1}^{(j)})$ is given by the Markovian transition probability defined by (6).

4.2 Filtering

In this subsection, we show that an approximated filter distribution at time t_n $\{\boldsymbol{I}_{1:n|n}^{(1)}, \cdots, \boldsymbol{I}_{1:n|n}^{(N_p)}\}$ is obtained from an approximated predictive distribution at time t_n $\left\{\boldsymbol{I}_{1:n|n-1}^{(1)}, \cdots, \boldsymbol{I}_{1:n|n-1}^{(N_p)}\right\}$. Given the observation $\boldsymbol{y}_n$, the probability $\Pr(\boldsymbol{I}_{1:n} = \boldsymbol{I}_{1:n|n-1}^{(j)}|\ \boldsymbol{y}_{1:n-1})$ is updated as follows:

$$\begin{aligned}
&\Pr(\boldsymbol{I}_{1:n} = \boldsymbol{I}_{1:n|n-1}^{(j)}|\boldsymbol{y}_{1:n}) \\
&= \Pr(\boldsymbol{I}_{1:n} = \boldsymbol{I}_{1:n|n-1}^{(j)}|\boldsymbol{y}_{1:n-1}, \boldsymbol{y}_n) \\
&= p(\boldsymbol{y}_n|\boldsymbol{I}_{1:n} = \boldsymbol{I}_{1:n|n-1}^{(j)}, \boldsymbol{y}_{1:n-1}) \\
&\cdot \Pr(\boldsymbol{I}_{1:n} = \boldsymbol{I}_{1:n|n-1}^{(j)}|\boldsymbol{y}_{1:n-1})/p(\boldsymbol{y}_n|\boldsymbol{y}_{1:n-1}) \\
&= \{p(\boldsymbol{y}_n|\boldsymbol{I}_{1:n} = \boldsymbol{I}_{1:n|n-1}^{(j)}, \boldsymbol{y}_{1:n-1}) \\
&\cdot \Pr(\boldsymbol{I}_{1:n} = \boldsymbol{I}_{1:n|n-1}^{(j)}|\boldsymbol{y}_{1:n-1})\}/ \\
&\{\sum_{j=1}^{N_p} p(\boldsymbol{y}_n|\boldsymbol{I}_{1:n} = \boldsymbol{I}_{1:n|n-1}^{(j)}, \boldsymbol{y}_{1:n-1}) \\
&\cdot \Pr(\boldsymbol{I}_{1:n} = \boldsymbol{I}_{1:n|n-1}^{(j)}|\boldsymbol{y}_{1:n-1})\} \\
&= \frac{w_n^{(j)}\frac{1}{N_p}}{\sum_{j=1}^{N_p} w_n^{(j)}\frac{1}{N_p}} = \frac{w_n^{(j)}}{\sum_{j=1}^{N_p} w_n^{(j)}}
\end{aligned} \tag{16}$$

where

$$w_n^{(j)} = p(\boldsymbol{y}_n|\boldsymbol{I}_{1:n} = \boldsymbol{I}_{1:n|n-1}^{(j)}, \boldsymbol{y}_{1:n-1}). \tag{17}$$

Equation (16) means that the filter distribution $p(\boldsymbol{I}_{1:n}|\boldsymbol{y}_{1:n})$ is approximated by giving weight proportional to $w_n^{(j)}$ to the jth particle of approximated predictive distribution. For the next prediction step, it is necessary to represent the approximated filter distribution with equally weighted particles $\left\{\boldsymbol{I}_{1:n|n}^{(1)}, \cdots, \boldsymbol{I}_{1:n|n}^{(N_p)}\right\}$. This is achieved by generating N_p particles $\left\{\boldsymbol{I}_{1:n|n}^{(1)}, \cdots, \boldsymbol{I}_{1:n|n}^{(N_p)}\right\}$ by resampling $\left\{\boldsymbol{I}_{1:n|n-1}^{(1)}, \cdots, \boldsymbol{I}_{1:n|n-1}^{(N_p)}\right\}$ with probability proportional to $\left\{w_n^{(1)}, \cdots, w_n^{(N_p)}\right\}$.

4.3 Recursive calculation for the state vector estimation

From (11) and (12), $p(\boldsymbol{x}_{n-1}|\boldsymbol{I}_{1:n-1} = \boldsymbol{I}_{1:n-1|n-1}^{(j)}, \boldsymbol{y}_{1:n-1})$ and $p(\boldsymbol{x}_n|\boldsymbol{I}_{1:n} = \boldsymbol{I}_{1:n|n-1}^{(j)}, \boldsymbol{y}_{1:n-1})$ becomes Gaussian distributions. Let us define mean vectors and covariance matrices of the two distributions as follows:

$$
\begin{aligned}
& p(\boldsymbol{x}_{n-1}|\boldsymbol{I}_{1:n-1} = \boldsymbol{I}_{1:n-1|n-1}^{(j)}, \boldsymbol{y}_{1:n-1}) \\
& \sim N(\boldsymbol{x}_{n-1|n-1}^{(j)}, V_{n-1|n-1}^{(j)}) \qquad (18) \\
& p(\boldsymbol{x}_n|\boldsymbol{I}_{1:n} = \boldsymbol{I}_{1:n|n-1}^{(j)}, \boldsymbol{y}_{1:n-1}) \\
& \sim N(\boldsymbol{x}_{n|n-1}^{(j)}, V_{n|n-1}^{(j)}) \qquad (19)
\end{aligned}
$$

Since $\boldsymbol{I}_{1:n-1|n-1}^{(j)}$ is assumed to be given, the CDLM (11) and (12) reduces to a linear Gaussian state space model and thus $\boldsymbol{x}_{n-1|n-1}^{(j)}$ and $V_{n-1|n-1}^{(j)}$ are calculated by Kalman filter. $\boldsymbol{x}_{n|n-1}^{(j)}$ and $V_{n|n-1}^{(j)}$ are also calculated by Kalman filter using $\boldsymbol{x}_{n-1|n-1}^{(j)}$, $V_{n-1|n-1}^{(j)}$ and $I_{n|n-1}^{(j)}$ as follows:

$$
\begin{aligned}
\boldsymbol{x}_{n|n-1}^{(j)} &= F_n(I_{n|n-1}^{(j)})\boldsymbol{x}_{n-1|n-1}^{(j)} \qquad (20) \\
V_{n|n-1}^{(j)} &= F_n(I_{n|n-1}^{(j)})V_{n-1|n-1}^{(j)}F_n^T(I_{n|n-1}^{(j)}) \\
&\quad + Q_n(I_{n|n-1}^{(j)}). \qquad (21)
\end{aligned}
$$

Note that $I_{n|n-1}^{(j)}$ is obtained by the prediction scheme of the MCMKF.

From (12), the predictive distribution of data also becomes a Gaussian:

$$
\begin{aligned}
& p(\boldsymbol{y}_n|\boldsymbol{I}_{1:n} = \boldsymbol{I}_{1:n|n-1}^{(j)}, \boldsymbol{y}_{1:n-1}) \\
& \qquad \sim N(\boldsymbol{y}_{n|n-1}^{(j)}, W_{n|n-1}^{(j)}) \qquad (22)
\end{aligned}
$$

where

$$
\begin{aligned}
\boldsymbol{y}_{n|n-1}^{(j)} &= H_n(I_{n|n-1}^{(j)})\boldsymbol{x}_{n|n-1}^{(j)} \qquad (23) \\
W_{n|n-1}^{(j)} &= H_n(I_{n|n-1}^{(j)})V_{n|n-1}^{(j)}H_n^T(I_{n|n-1}^{(j)}) \\
&\quad + R_n(I_{n|n-1}^{(j)}). \qquad (24)
\end{aligned}
$$

The left hand side of (22) is the weight $w_n^{(j)}$ defined in (17). Thus $w_n^{(j)}$ follows the Gaussian distribution with mean $\boldsymbol{y}_{n|n-1}^{(j)}$ and covariance matrix $W_{n|n-1}^{(j)}$ as follows:

$$
\begin{aligned}
w_n^{(j)} &= (2\pi)^{-N_d/2}\left|W_{n|n-1}^{(j)}\right|^{-1/2} \\
\exp & \left[-\frac{1}{2}(\boldsymbol{y}_n - \boldsymbol{y}_{n|n-1}^{(j)})^T W_{n|n-1}^{(j)-1}(\boldsymbol{y}_n - \boldsymbol{y}_{n|n-1}^{(j)})\right] \qquad (25)
\end{aligned}
$$

where $\left|W_{n|n-1}^{(j)}\right|$ is the absolute value of the determinant of $W_{n|n-1}^{(j)}$. Because $\boldsymbol{x}_{n|n-1}^{(j)}$ and $V_{n|n-1}^{(j)}$ are given in (20) and (21), the weight $w_n^{(j)}$ is obtained using (23), (24) and (25).

By using the prediction and the filtering algorithm recursively, we finally obtain N_p particles $\{\boldsymbol{I}_{1:N_e|N_e}^{(1)}, \cdots, \boldsymbol{I}_{1:N_e|N_e}^{(N_p)}\}$ that approximate $p(\boldsymbol{I}_{1:N_e}|\boldsymbol{y}_{1:N_e})$, the posterior distribution of $\boldsymbol{I}_{1:N_e}$ conditioned on all of available data. Here, N_e is the number of observation epochs. $p(\boldsymbol{I}_{1:N_e}|\boldsymbol{y}_{1:N_e})$ is called smoother distribution of $\boldsymbol{I}_{1:N_e}$. A sequence of each particle, $\boldsymbol{I}_{1:N_e|N_e}^{(j)} = [I_{1|N_e}^{(j)}, I_{2|N_e}^{(j)}, \cdots I_{N_e|N_e}^{(j)}]$, is called the trajectory.

This filtering algorithm is conceptually similar to the storing state vector algorithm in the Monte Carlo filter proposed by Kitagawa (1996). He applied the Monte Carlo approximation directly to the distribution of the state, whereas we apply the approximation to the distribution of the indicator variable. He showed that in the Monte Carlo filter the repetition of resampling gradually decreases the number of different realizations of state vector as time passes because the number of realizations is finite. Therefore the shape of the distribution of the state deteriorates as time passes. Kitagawa (1996) showed that this difficulty can be eliminated by employing fixed L-lag smoother rather than fixed interval smoother (Anderson and Moore, 1979). Although we apply the Monte Carlo approximation to the indicator variable instead of the state, this situation also applies to the MCMKF. Thus following Kitagawa (1996), we modify the MCMKF filtering algorithm as follows:
For fixed L, generate N_p particles $\{\boldsymbol{I}_{n-L:n|n}^{(1)}, \boldsymbol{I}_{n-L:n|n}^{(2)}, \cdots, \boldsymbol{I}_{n-L:n|n}^{(N_p)}\}$ by the resampling of $\{\boldsymbol{I}_{n-L:n|n-1}^{(1)}, \boldsymbol{I}_{n-L:n|n-1}^{(2)}, \cdots, \boldsymbol{I}_{n-L:n|n-1}^{(N_p)}\}$ with probability proportional to $\{w_n^{(1)}, \cdots, w_n^{(N_p)}\}$ defined in (17).
It is recommended to take L not so large (say, 10 or 20 at the largest 50) (Kitagawa, 1996; Higuchi and Kitagawa, 2000). We adopt $L = 20$ in our application study shown in Section 6.

5. MODEL AVERAGING

5.1 State Estimation

We present here an algorithm to estimate the state using all the N_p trajectories $\{\boldsymbol{I}_{1:N_e|N_e}^{(1)}, \cdots, \boldsymbol{I}_{1:N_e|N_e}^{(N_p)}\}$. In this case, $F_n(I_n^{(j)})$, $Q_n(I_n^{(j)})$, $H_n(I_n^{(j)})$ and $R_n(I_n^{(j)})$ $(j = 1, \cdots, N_p)$ in (11) and (12) reduce to sets of known matrices which have different time evolutions corresponding to trajectories. Thus the CDLM defined by (11) and (12) reduces to the conventional linear Gaussian state space model to which Kalman filter is applicable for state estimation:

$$\boldsymbol{x}_n^{(j)} = F_n^{(j)}\boldsymbol{x}_{n-1}^{(j)} + \boldsymbol{v}_n^{(j)}, \qquad \boldsymbol{v}_n^{(j)} \sim N(\mathbf{0}, Q_n^{(j)}) \quad (26)$$

$$\boldsymbol{y}_n = H_n^{(j)}\boldsymbol{x}_n^{(j)} + \boldsymbol{w}_n^{(j)}, \qquad \boldsymbol{w}_n^{(j)} \sim N(\mathbf{0}, R_n^{(j)}) \quad (27)$$

where $F_n^{(j)} = F_n(I_{n|N_e}^{(j)})$, $Q_n^{(j)} = Q_n(I_{n|N_e}^{(j)})$, $H_n^{(j)} = H_n(I_{n|N_e}^{(j)})$ and $R_n^{(j)} = R_n(I_{n|N_e}^{(j)})$. Let

$$\boldsymbol{x}_{i|k}^{(j)} = \mathrm{E}(\boldsymbol{x}_i|\boldsymbol{y}_{1:k}, \boldsymbol{I}_{1:N_e} = \boldsymbol{I}_{1:N_e|N_e}^{(j)}) \quad (28)$$

$$V_{i|k}^{(j)} = \mathrm{Cov}(\boldsymbol{x}_i|\boldsymbol{y}_{1:k}, \boldsymbol{I}_{1:N_e} = \boldsymbol{I}_{1:N_e|N_e}^{(j)}) \quad (29)$$

be the conditional mean and the covariance matrix of the state at time t_i given the data $\boldsymbol{y}_{1:k}$ for jth trajectory. $\left\{\boldsymbol{x}_{n+1|n}^{(j)}, V_{n+1|n}^{(j)}\right\}_{j=1}^{N_p}$, $\left\{\boldsymbol{x}_{n|n}^{(j)}, V_{n|n}^{(j)}\right\}_{j=1}^{N_p}$ and $\left\{\boldsymbol{x}_{n|N_e}^{(j)}, V_{n|N_e}^{(j)}\right\}_{j=1}^{N_p}$ are recursively obtained by Kalman filter. Given $\left\{\boldsymbol{x}_{n|N_e}^{(j)}, V_{n|N_e}^{(j)}\right\}_{j=1}^{N_p}$, distribution of the final estimate for $\boldsymbol{x}_n$, $p(\boldsymbol{x}_n|\boldsymbol{y}_{1:N_e})$, is written as

$$\begin{aligned} p(\boldsymbol{x}_n|\boldsymbol{y}_{1:N_e}) &= \sum_{j=1}^{N_p} p(\boldsymbol{x}_n, \boldsymbol{I}_{1:N_e} = \boldsymbol{I}_{1:N_e|N_e}^{(j)}|\boldsymbol{y}_{1:N_e}) \\ &= \sum_{j=1}^{N_p} p(\boldsymbol{x}_n|\boldsymbol{I}_{1:N_e} = \boldsymbol{I}_{1:N_e|N_e}^{(j)}, \boldsymbol{y}_{1:N_e}) \\ &\quad \Pr(\boldsymbol{I}_{1:N_e} = \boldsymbol{I}_{1:N_e|N_e}^{(j)}|\boldsymbol{y}_{1:N_e}) \\ &= \frac{1}{N_p}\sum_{j=1}^{N_p} N(\boldsymbol{x}_{n|N_e}^{(j)}, V_{n|N_e}^{(j)}). \end{aligned} \quad (30)$$

In the 3-rd equality, (10), (28) and (29) are used. Therefore $p(\boldsymbol{x}_n|\boldsymbol{y}_{1:N_e})$ is non-Gaussian distribution with mean

$$\boldsymbol{x}_{n|N_e} = \frac{1}{N_p}\sum_{j=1}^{N_p} \boldsymbol{x}_{n|N_e}^{(j)}. \quad (31)$$

Estimation of standard deviation error bounds for $\boldsymbol{x}_{n|N_e}$ is not straightforward because $p(\boldsymbol{x}_n|\boldsymbol{y}_{1:N_e})$ is non-Gaussian distribution. In this study, error bounds for $\boldsymbol{x}_{n|N_e}$ are approximately obtained as follows:

(1) Generate N_s realizations of $N(\boldsymbol{x}_{n|N_e}^{(j)}, V_{n|N_e}^{(j)})$, $\mathscr{X}_{n,1}^{(j)}$, $\mathscr{X}_{n,2}^{(j)}$, $\cdots$, $\mathscr{X}_{n,N_s}^{(j)}$, for each trajectory, $j = 1, 2, \cdots, N_p$.
(2) Estimate covariance matrix of $p(\boldsymbol{x}_n|\boldsymbol{y}_{1:N_e})$, $V_{n|N_e}$, by

$$V_{n|N_e} = \frac{1}{N_pN_s - 1}\sum_{j=1}^{N_p}\sum_{k=1}^{N_s} \left[\mathscr{X}_{n,k}^{(j)} - \bar{\mathscr{X}}_{n,k}^{(j)}\right]\left[\mathscr{X}_{n,k}^{(j)} - \bar{\mathscr{X}}_{n,k}^{(j)}\right]^T \quad (32)$$

where

$$\bar{\mathscr{X}}_{n,k}^{(j)} = \frac{1}{N_pN_s}\sum_{j=1}^{N_p}\sum_{k=1}^{N_s} \mathscr{X}_{n,k}^{(j)}. \quad (33)$$

The procedure for state estimation using all trajectories described above is computationally massive both in calculation time and in memory. More efficient algorithm is implemented by reducing number of trajectories to which Kalman filter is applied. This is done by sampling N_p' ($N_p' < N_p$) trajectories randomly from N_p trajectories $\{\boldsymbol{I}_{1:N_e|N_e}^{(1)}, \cdots, \boldsymbol{I}_{1:N_e|N_e}^{(N_p)}\}$. Once N_p' trajectories are selected, the procedure for state estimation is identical to the case using all N_p trajectories. The distribution of the final estimate for $\boldsymbol{x}_n$, $p(\boldsymbol{x}_n|\boldsymbol{y}_{1:N_e})$, and its mean vector, $\boldsymbol{x}_{n|N_e}$, are obtained by replacing N_p in (30) and (31) with N_p', respectively.

5.2 *Likelihood of the meta-model*

In this subsection, we present a formula for the log-likelihood of the model. Let $\boldsymbol{\theta}$ be a vector that contains temporally invariable hyper-parameters. Given $\boldsymbol{\theta}$, the likelihood of the model is expressed by

$$\begin{aligned} L(\boldsymbol{\theta}) &= p(\boldsymbol{y}_{1:N_e}|\boldsymbol{\theta}) \\ &= \prod_{n=1}^{N_e} p(\boldsymbol{y}_n|\boldsymbol{y}_{1:n-1}, \boldsymbol{\theta}). \end{aligned} \quad (34)$$

If we use all N_p trajectories for state estimation, $p(\boldsymbol{y}_n|\boldsymbol{y}_{1:n-1}, \boldsymbol{\theta})$ in (34) is given by

$$\begin{aligned} &p(\boldsymbol{y}_n|\boldsymbol{y}_{1:n-1}, \boldsymbol{\theta}) = \\ &\sum_{j=1}^{N_p} p(\boldsymbol{y}_n, \boldsymbol{I}_{1:n} = \boldsymbol{I}_{1:n|N_e}^{(j)}|\boldsymbol{y}_{1:n-1}, \boldsymbol{\theta}) \\ &= \sum_{j=1}^{N_p} p(\boldsymbol{y}_n|\boldsymbol{y}_{1:n-1}, \boldsymbol{I}_{1:n} = \boldsymbol{I}_{1:n|N_e}^{(j)}, \boldsymbol{\theta}) \\ &\quad \Pr(\boldsymbol{I}_{1:n} = \boldsymbol{I}_{1:n|N_e}^{(j)}|\boldsymbol{y}_{1:n-1}, \boldsymbol{\theta}) \\ &= \frac{1}{N_p}\sum_{j=1}^{N_p} p(\boldsymbol{y}_n|\boldsymbol{y}_{1:n-1}, \boldsymbol{I}_{1:n} = \boldsymbol{I}_{1:n|N_e}^{(j)}, \boldsymbol{\theta}). \end{aligned} \quad (35)$$

Combining (34) and (35) yields the following formula for the log-likelihood of the model $l(\boldsymbol{\theta})$. If we use N_p' trajectories randomly sampled from N_p trajectories, the log-likelihood of the model is obtained by replacing N_p with N_p'.

The goodness of the model is evaluated by the Akaike information criterion (AIC) (Akaike, 1974). The AIC is defined as

$$\mathrm{AIC} = -2l(\boldsymbol{\theta}) + 2(\text{number of unknown parameters}). \quad (36)$$

6. SUMMARY

6.1 *An algorithm for MCMKF*

The MCMKF algorithm that we propose in this study is summarized as follows:

(1) Initialization: For $j = 1, \ldots, N_p$,
    (a) Sample $I_{0|0}^{(j)} \sim p(I_{0|0})$.
    (b) Set $(\boldsymbol{x}_{1|0}^{(j)}, V_{1|0}^{(j)})$.
(2) For $n = 1, \ldots, N_e$,
    (a) For $j = 1, \ldots, N_p$,

(i) Sample $I^{(j)}_{n|n-1} \sim \Pr(I_n = I^{(j)}_{n|n-1} | I_{n-1} = I^{(j)}_{n-1|n-1})$.

(ii) Set $\boldsymbol{I}^{(j)}_{1:n|n-1} = (\boldsymbol{I}^{(j)}_{1:n-1|n-1}, I^{(j)}_{n|n-1})$.

(iii) Compute $w^{(j)}_n = p(\boldsymbol{y}_n | \boldsymbol{I}_{1:n} = \boldsymbol{I}^{(j)}_{1:n|n-1}, \boldsymbol{y}_{1:n-1})$.

(iv) Update $(\boldsymbol{x}^{(j)}_{n-1|n-1}, V^{(j)}_{n-1|n-1})$ to obtain $(\tilde{\boldsymbol{x}}^{(j)}_{n|n}, \tilde{V}^{(j)}_{n|n})$ using Kalman filter.

(b) Obtain $\{(\boldsymbol{I}^{(j)}_{1:n|n}, \boldsymbol{x}^{(j)}_{n|n}, V^{(j)}_{n|n})\}_{j=1}^{N_p}$ by the resampling of $\{(\boldsymbol{I}^{(j)}_{1:n|n-1}, \tilde{\boldsymbol{x}}^{(j)}_{n|n}, \tilde{V}^{(j)}_{n|n})\}_{j=1}^{N_p}$ with probability proportional to $w^{(j)}_n$.

(3) Obtain the distribution of the final estimate for $\boldsymbol{x}_n$, $p(\boldsymbol{x}_n | \boldsymbol{y}_{1:N_e})$, based on N_p trajectories $\left\{\boldsymbol{I}^{(1)}_{1:N_e|N_e}, \cdots, \boldsymbol{I}^{(N_p)}_{1:N_e|N_e}\right\}$.

6.2 Application Result

A temporally invariable scaling parameter as in the NIF could not trace abrupt changes because optimized scaling parameter would be too small to allow such a sudden change of fault slip, and vice versa. As a result, estimated slip evolution would be flattened during the event and oscillatory in steady-state period, and hence, it would be hardly possible to identify the initiation of events. In order to overcome this difficulties, we propose the CDLM and apply the MCMKF for its state estimation. We apply this space-time inversion method to simulated data which are generated by an infinitely long strike slip fault. Results show that the proposed method can reproduce rapidly accelerating and decelerating fault slip and coseismic slip as well as slow variation of fault slip rate, even in a case that noise level is so high that signal is invisible. We confirmed that a benefit of applying our approach is maximized when deformation rate varies rapidly or coseismic deformation exists, and signal-to-noise ratio is low. In addition we address that the MCMKF is designed to deal with the CDLM and then can be applicable to a wide variety of the nonlinear non-Gaussian state space models. The MCMKF allows us to integrate various type of time series models and to generate a flexible time series model automatically.

REFERENCES

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Trans. Autom. Control* **AC-19**, 716–723.

Anderson, B.D.O. and J.B. Moore (1979). *Optimal Filtering*. Prentice-Hall. New Jersey.

Bergman, N., A. Doucet and N. Gordon (2001). Optimal estimation and Cramér-Rao bounds for partial non-Gaussian state space models. *Annals of Institute of Statistical Mathematics* **53**, 97–122.

Chen, R. and J.S. Liu (2000). Mixture Kalman filters. *J. R. Statist. Soc. B.* **62**, 493–508.

Chen, R., X. X. Wang and J.S. Liu (2000). Adaptive joint detection and decoding in flat-fading channels via mixture Kalman filtering. *IEEE Trans. on Information Theory* **46**, 2079–2094.

Doucet, A., J.F.G. de Freitas and N.J. Gordon (2001). *Sequential Monte Carlo Methods in Practice*. Springer-Verlag. New York.

Frühwirth-Schnatter, S. (2001). Fully Bayesian analysis of switching Gaussian state space models. *Annals of Institute of Statistical Mathematics* **53**, 31–49.

Fukuda, J., T. Higuchi, S. Miyazaki and T. Kato (2003). A new approach to time dependent inversion of geodetic data using Monte Carlo mixture Kalman filter. *Geophysical Journal International (submitted)*.

Heki, K., S. Miyazaki and H. Tsuji (1997). Silent fault slip following an interplate thrust earthquake at the Japan trench. *Nature* **386**, 595–598.

Higuchi, T. (2001). Evolutionary time series model with parallel computing. In: *The Third JAPAN-US Joint Seminar on Statistical Time Series Analysis*. pp. 183–190.

Higuchi, T. and G. Kitagawa (2000). Knowledge discovery and self-organizing state space model. *IEICE Transactions on Information and Systems* **E83-D**, 36–43.

Kim, C.-J. and Ch.R. Nelson (1999). *State space models with regime switching*. MIT press. Cambridge.

Kitagawa, G. (1996). Monte Carlo filter and smoother for non-Gaussian nonlinear state space model. *Journal of Computational and Graphical Statistics* **5**, 1–25.

Kitagawa, G. (1998). Self-organizing state space model. *Journal of the Americal Statistical Association* **93**, 1203–1215.

Kitagawa, G. and W. Gersch (1996). *Smoothness Priors Analysis of Time Series*. Springer-Verlag. New York.

Liu, J.S., R. Chen and T. Logvinenko (2001). A theoretical framework for sequential importance sampling with resampling. In: *Sequential Monte Carlo Methods in Practice* (A. Doucet, J.F.G. de Freitas and N.J. Gordon, Eds.). pp. 225–246. Springer-Verlag. New York.

Ozawa, S., M. Murakami, M. Kaidzu, T. Tada, T. Sagiya, Y. Hatanaka, H. Yarai and T. Nishimura (2001). Detection and monitoring of ongoing aseismic slip in the tokai region, central Japan. *Science* **298**, 1009–1012.

Segall, P. and M. Matthews (1997). Time dependent inversion of geodetic data. *Journal of Geophysical Research* **102**, 22391–22409.

Shephard, N. (1994). Partial non-Gaussian time series models. *Biometrika* **81**, 115–131.

Shumway, R.H. and D.S. Stoffer (1991). Dynamic liner models with switching. *Journal of the American Statistical Association* **86**, 763–769.