# BAYESIAN MCMC NONLINEAR TIME SERIES PREDICTION

*Y. Nakada, T. Kurihara and T. Matsumoto*

Department of Electrical, Electronics and Computer Engineering
Waseda University; CREST, JST
3-4-1 Ohkubo, Sinjuku-ku, Tokyo, 169-8555, Japan
Tel/Fax: +81-3-5286-3377, E-mail: takashi@mse.waseda.ac.jp

## ABSTRACT

An MCMC(Markov Chain Monte Carlo) algorithm is proposed for nonlinear time series prediction with Hierarchical Bayesian framework. The algorithm computes *predictive mean* and *error bar* by drawing samples from predictive distributions. The algorithm is tested against time series generated by (chaotic) Rössler system and it outperforms quadratic approximations previously proposed by the authors.

## 1. INTRODUCTION

Given data $\{x_t\}_{t=0}^N \subset \mathbb{R}$, time series prediction problem amounts to making predictions of its future values $\{x_t\}_{t>N}$.

In many time series prediction problems,

**(i)** *nonlinearity* is behind time series data, and

**(ii)** data contains *noise*.

One way of approaching these problems is probabilistic/statistical approach with "basis" functions such as neural nets to fit time series data $\{x_t\}_{t=0}^N$ without overfitting.

Hierarchical Bayesian approach using model Markov processes with *neural net* was previously proposed by the authors [1]-[3]. In this approach, the goal is to evaluate *predictive distribution* for future values $\{x_t\}_{t>N}$. Since evaluation of predictive distribution is difficult, quadratic approximations (QAP) have been used so far [1]-[3].

This paper proposes a new scheme for time series prediction problem via MCMC(Markov Chain Monte Carlo) without QAP. For evaluation of predictive distribution for future values $\{x_t\}_{t>N}$, the proposed scheme draws samples of future values from predictive distribution. Using these samples, one can estimate

**(i)** *predictive mean* which is usually used as prediction of future values, and

**(ii)** *error bar (standard deviation)* which shows uncertainty of prediction.

In order to demonstrate validity of the proposed scheme, it is applied to chaotic time series data generated by noisy Rössler system.

## 2. FORMULATION

**Problem:**

Given data set $D := \{x_t\}_{t=0}^N \subset \mathbb{R}$ , predict $\{x_t\}_{t=N+1}^T$ .

### Hypothesis $\mathcal{H}$

In a Bayesian framework, model specification consists of the following.

1. **Architecture**:
   Specification of basis function for data fitting, e.g., three-layer perceptron with $h$ hidden units and a particular sigmoid function.

2. **Likelihood**:

$$P\left(\{x_t\}_{t=\tau}^N, \{x_{\tau-1}, \ldots, x_0\} \mid \boldsymbol{w}, \beta, \mathcal{H}\right) := \underbrace{\prod_{t=0}^{N-\tau} \frac{1}{Z_D(\beta)} \exp\left(-\beta E_D(x_{t+\tau} \mid x_{t+\tau-1}, \ldots, x_t; \boldsymbol{w})\right)}_{\text{noisy dynamics}} \times \underbrace{P(x_{\tau-1}, \ldots, x_0 \mid \mathcal{H})}_{\text{initial state uncertainty}} \quad (1)$$

$$E_D(x_{t+\tau} \mid x_{t+\tau-1}, \ldots, x_t; \boldsymbol{w}) := \frac{1}{2}(x_{t+\tau} - f(x_{t+\tau-1}, \ldots, x_t; \boldsymbol{w}))^2 \quad (2)$$

where $f(\cdot)$ is neural net output, $\boldsymbol{w} \in \mathbb{R}^k$ the weight parameters of a particular architecture, $\beta$ (unknown) uncertainty levels, $Z_D(\beta)$ the normalization constants, and $\tau$ is *embedding dimension* (the order of the dynamics) which is, in general,

unknown. Equation(1) looks at $\{x_t\}$ as a $\tau$-th order Markov process whose state transition probability density is given by the first two factors, whereas the last factor is the initial state probability density.

3. **Prior for $(\boldsymbol{w}, \boldsymbol{\alpha}, \beta, \mathcal{H})$:**
See [1]-[3].

The goal of the prediction problem is to evaluate predictive distribution

$$P(\{x_t\}_{N+1}^{T} \mid D, \mathcal{H}) = \iiint P(\{x_t\}_{N+1}^{T} \mid \boldsymbol{w}, \beta, \mathcal{H}) \times P(\boldsymbol{w}, \boldsymbol{\alpha}, \beta \mid D, \mathcal{H}) d\boldsymbol{w} d\boldsymbol{\alpha} d\beta \quad (3)$$

## 3. ALGORITHM

In order to draw samples from joint posterior of $\boldsymbol{w}$ and $(\boldsymbol{\alpha}, \beta)$, the scheme proposed in [5] consists of alternate iterations of two different operations:

(A) Hyperparameter $(\boldsymbol{\alpha}^{(j)}, \beta^{(j)})$ updated via Gibbs sampling[4].

(B) Weight parameter $\boldsymbol{w}^{(j)}$ updated via Hybrid Monte Carlo[5].

At the end of $j$th iteration, $(\boldsymbol{w}^{(j)}, \boldsymbol{\alpha}^{(j)}, \beta^{(j)})$ is considered to be a sample from the joint posterior.

### 3.1. The Hybrid Monte Carlo

Consider the case where

$$P(\boldsymbol{w} \mid D, \boldsymbol{\alpha}, \beta, \mathcal{H}) \propto \exp(-M(\boldsymbol{w})) \quad (4)$$

for some "energy" function $M(\boldsymbol{w})$. Let "Hamiltonian " function $H(\boldsymbol{w}, \boldsymbol{z})$ and "Kinetic energy " function $K(\boldsymbol{z})$ be defined by

$$H(\boldsymbol{w}, \boldsymbol{z}) := M(\boldsymbol{w}) + K(\boldsymbol{z}) \quad (5)$$

$$K(\boldsymbol{z}) := \sum_{i=1}^{k} \frac{z_i^2}{2m_i} \quad (6)$$

The Hybrid Monte Carlo considers the Hamiltonian dynamical system

$$\frac{dw_i}{ds} = +\frac{\partial H}{\partial z_i}(\boldsymbol{w}, \boldsymbol{z}) = \frac{z_i}{m_i} \quad (7)$$

$$\frac{dz_i}{ds} = -\frac{\partial H}{\partial w_i}(\boldsymbol{w}, \boldsymbol{z}) = -\frac{\partial M}{\partial w_i}(\boldsymbol{w}) \quad (8)$$

where $s$ is "time" of Hamiltonian dynamical system(7),(8). A Hamiltonian dynamical system is volume preserving, i.e., any Euclidean volume is preserved along trajectories;

$$\frac{d}{ds}\left(\det\left(\frac{\partial g(\boldsymbol{w}, \boldsymbol{z})}{\partial(\boldsymbol{w}, \boldsymbol{z})}\right)\right) \equiv 0 \quad (9)$$

where $g(\cdot)$ represents the right hand side of (7) and (8). Obviously, the density in question

$$P(\boldsymbol{w}(s), \boldsymbol{z}(s)) = \frac{1}{Z} exp\left(-H(\boldsymbol{w}(s), \boldsymbol{z}(s))\right) \quad (10)$$

is absolutely continuous with respect to the Lebesgue measure so that it is invariant under (7),(8);

$$\iint_A P(\boldsymbol{w}, \boldsymbol{z}) d\boldsymbol{w} d\boldsymbol{z} = \iint_A P\left(F^{-s}(\boldsymbol{w}', \boldsymbol{z}')\right) d\boldsymbol{w}' d\boldsymbol{z}' \quad (11)$$

for any (Lebesgue measurable) subset A of $\mathbb{R}^k \times \mathbb{R}^k$, where $F^s$ is the time $s$-map of the flow induced by (7),(8). One of the important ideas behind this is the fact that *derivative information* $\frac{\partial M}{\partial \boldsymbol{w}}$ can be incorporated so that random walk behavior can be avoided. Since (7),(8) is a deterministic dynamical system, one needs to perform other operations in order to ensure ergodic sampling. The Hybrid Monte Carlo consists of two steps:

(i) Deterministic transition via Hamiltonian dynamical system (7),(8);

(ii) Stochastic transition via occasional updates of initial condition for the auxiliary variable $z_i$, $i = 1, \cdots, k$, by performing sampling from Gaussian distribution.

Actual implementation of this scheme is more complicated than that described above, because a Hamiltonian dynamical system cannot be exactly simulated by a computer so that perfect preservation $H(\boldsymbol{w}(s), \boldsymbol{z}(s)) \equiv$ *constant* is not possible. In order to overcome this, the Hybrid Monte Carlo considers the Leapfrog discretization. Let $\Delta$ be the period of time over which the deterministic transition via Hamiltonian dynamics is to be performed, let $\epsilon > 0$ be a step size for discretization, and define $L := \Delta/\epsilon$.

The Leapfrog discretization performs the following step $L$ times supposing that $L$ is an integer:

$$\hat{z}_i\left(s + \frac{\epsilon}{2}\right) = \hat{z}_i(s) - \frac{\epsilon}{2}\frac{\partial M}{\partial w_i}(\hat{\boldsymbol{w}}(s)) \quad (12)$$

$$\hat{w}_i(s + \epsilon) = \hat{w}_i(s) - \frac{\epsilon}{m_i}\hat{z}_i\left(s + \frac{\epsilon}{2}\right) \quad (13)$$

$$\hat{z}_i(s + \epsilon) = \hat{z}_i\left(s + \frac{\epsilon}{2}\right) - \frac{\epsilon}{2}\frac{\partial M}{\partial w_i}(\hat{\boldsymbol{w}}(s + \epsilon)) \quad (14)$$

This amounts to half step size $(\epsilon/2)$ approximation for $z_i$ and full step size approximation for $w_i$, and another

half step size approximation. Half step desretizations are often used for numerical integration of differential equations, e.g., Runge-Kutta.

### 3.2. Gibbs Sampling for Hyperparameters

Samples of hyperparameters are drawn by the usual Gibbs sampling.

## 4. PREDICTION

The goal of the proposed scheme is to draw samples from predictive distributions:

$$\{x_t^{(l)}\}_{t=N+1}^T \sim P(\{x_t\}_{t=N+1}^T \mid D, \mathcal{H}) \tag{15}$$

In order to draw samples of future values $\{x_t^{(l)}\}_{t=N+1}^T$ via (15), the proposed scheme proceeds in two steps:

$$\left(\boldsymbol{w}^{(l)}, \boldsymbol{\alpha}^{(l)}, \beta^{(l)}\right) \sim P(\boldsymbol{w}, \boldsymbol{\alpha}, \beta \mid D, \mathcal{H}) \tag{16}$$

$$\{x_t^{(l)}\}_{t=N+1}^T \sim P(\{x_t\}_{t=N+1}^T \mid \boldsymbol{w}^{(l)}, \beta^{(l)}, \mathcal{H}) \tag{17}$$

The scheme first draws samples $\left(\boldsymbol{w}^{(l)}, \boldsymbol{\alpha}^{(l)}, \beta^{(l)}\right)$ from the posterior distributions (16) via MCMC described in the previous section. Secondly, the scheme uses these samples $\left(\boldsymbol{w}^{(l)}, \boldsymbol{\alpha}^{(l)}, \beta^{(l)}\right)$ to draw samples of future values $\{x_t^{(l)}\}_{t=N+1}^T$:

$$\begin{aligned} x_{N+1}^{(l)} &\sim P(x_{N+1} \mid \boldsymbol{x}_N^{(l)}, \boldsymbol{w}^{(l)}, \beta^{(l)}, \mathcal{H}) \\ x_{N+2}^{(l)} &\sim P(x_{N+2} \mid \boldsymbol{x}_{N+1}^{(l)}, \boldsymbol{w}^{(l)}, \beta^{(l)}, \mathcal{H}) \\ &\vdots \\ x_T^{(l)} &\sim P(x_T \mid \boldsymbol{x}_{T-1}^{(l)}, \boldsymbol{w}^{(l)}, \beta^{(l)}, \mathcal{H}) \end{aligned} \tag{18}$$

$\boldsymbol{x}_t := (x_t, x_{t-1}, \cdots, x_{t-\tau+1})$.

It follows from the Markov property

$$\begin{aligned} &P(\{x_t\}_{t=N+1}^T \mid \boldsymbol{w}, \beta, \mathcal{H}) \\ &\quad = \prod_{t=N+1}^T P(x_t \mid \boldsymbol{x}_{t-1}, \boldsymbol{w}, \beta, \mathcal{H}) \end{aligned} \tag{19}$$

where

$$\begin{aligned} &P(x_t \mid \boldsymbol{x}_{t-1}, \boldsymbol{w}, \beta, \mathcal{H}) \\ &\quad = \sqrt{\frac{\beta}{2\pi}} \exp\left(\frac{-\beta\{x_t - f(\boldsymbol{x}_{t-1}; \boldsymbol{w})\}^2}{2}\right) \end{aligned} \tag{20}$$

After drawing samples of future values $\{x_t^{(l)}\}_{t=N+1}^T$, predictive mean $\overline{x}_t$ and error bar $\sigma_{x_t}$ at $t$ can be estimated as follows:

$$\overline{x}_t \approx \frac{1}{S} \sum_{l=1}^L x_t^{(l)} \tag{21}$$

$$\sigma_{x_t} \approx \sqrt{\frac{1}{S-1} \sum_{l=1}^L \left(x_t^{(l)} - \overline{x}_t\right)^2} \tag{22}$$

where $S$ is the number of samples.

## 5. DEMONSTRATION:CHAOTIC TIME SERIES

$$\left\{ \begin{array}{rcl} \dot{x} &=& -y - z + \nu_t^1 \\ \dot{y} &=& x + ay + \nu_t^2 \\ \dot{z} &=& bx - cz + xz + \nu_t^3 \end{array} \right. \tag{23}$$

Equation (23) is the well-known Rössler system with noise processes ($\nu_t^1$, $\nu_t^2$, $\nu_t^3$). To avoid technical difficulties associated with stochastic process with continuous parameters, let us consider the discrete version of (24):

$$\left\{ \begin{array}{rcl} x_{(t+1)\delta} &=& f(x_{t\delta}, y_{t\delta}, z_{t\delta}) + \nu_{t\delta}^1 \\ y_{(t+1)\delta} &=& g(x_{t\delta}, y_{t\delta}, z_{t\delta}) + \nu_{t\delta}^2 \\ z_{(t+1)\delta} &=& h(x_{t\delta}, y_{t\delta}, z_{t\delta}) + \nu_{t\delta}^3 \end{array} \right. \tag{24}$$

where $f(\cdot)$, $g(\cdot)$, $h(\cdot)$ represent a numerical integration scheme , e.g., Runge-Kutta, with step size $\delta$, and $\nu_{t\delta}^i \sim i.i.d.N(0, \sigma^2), i = 1, 2, 3$.

Let $\{x_{t\delta}\}_{t\geq 0}$ be the observation. Figure. 2 shows time series data $\{x_{t\delta\eta}\}_{t=0}^{499}$ generated by discrete noisy Rössler system (24), where $\delta = 0.01$, $\eta = 70$, and $\sigma = 0.02$, embedded into $\mathbb{R}^3$. Observe that the magnitude of the right hand side of (24) is roughly $\delta = 0.01$ times that of (23).Therefore $\nu_{t\delta}^i \sim N(0, (0.02)^2)$ implies that the noise process $\nu_t^i$ in (23) is roughly 100 times larger than that of (24). The value $\eta$ stands for sampling period. In general $\eta$ and the order of Markov process $\tau$ needs to be estimated, however, in this paper we assume $\eta$ and $\tau(= 4)$ are already estimated [1]. This data was used as the training data set and the scheme dscribed in the previous section was applied. In order to demonstrate validity of the proposed scheme, we provided 5 different test data sets, and for each test data set predictive mean and error bar were estimated with various model $\mathcal{H}$. Let these 5 test data sets be denoted $test\,1, \cdots, test\,5$. For comparison proposes, QAP [1]-[3] was also applied.

Figure. 3 shows the average of cumulative squared errors for five data sets up to 80 step ($T = 80$).This indicates superiority of the prediction with MCMC.

Figure. 4 compares prediction capabilities of MCMC and that of QAP with various models for $test\,1$ where the evolutions of cumulative squared errors

$$\sum_{t=0}^T (x_{t\delta\eta} - \overline{x}_t)^2, \quad T = 1, \cdots, 80 \tag{25}$$

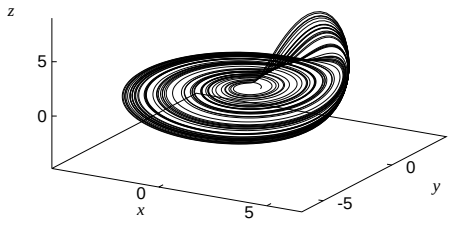


Figure 1: Rössler system
at $(a, b, c) = (0.36, 0.4, 4.5)$

is shown.

Figure. 5 shows predictive mean and error bar of the case ($test$ $2, h = 4$) which shows the best performance with respect to cumulative squared errors at 80 step. In this case, most of true values are within the range $\overline{x}_t \pm 1\sigma_{x_t}$. The algorithm appears to be fully functional.

## 6. REFERENCES


[1] T. Matsumoto, M. Saito, Y. Nakajima, J. Sugi, H. Hamagishi, "A Hierarchical Bayes Approach to Reconstruction and Prediction of Nonlinear Dynamical Systems", IEEE Workshop on Nonlinear Signal and Image Processing (NSIP '99) , vol. 1, pp.114 - 118, June 1999

[2] T. Matsumoto, M. Saito, Y. Nakajima, J. Sugi, H. Hamagishi, "Reconstruction and Prediction of Nonlinear Dynamical Systems : A Hierarchical Bayes Approach with Nueral Nets", IEEE International Confence on Accoustics, Speech, and Signal Processing (ICASSP 99) ,vol. 2, p.1057 - 1060, March 1999

[3] T. Matsumoto, M. Saito and J. Sugi, "Nonlinear Time Series Prediction Weighted by Marginal Likelihoods: A Hirearchical Bayesian Approach," the 1999 International Joint Conference on Neural Networks, Washington, D.C., USA, July 10-16, 1999.

[4] W. R. Gilks, S. Richardson, and D. J. Spiegelhalter, "Markov Chain Monte Carlo in Practice, " Chapman & Hall, London, 1996.

[5] R. M. Neal, "Bayesian Learning for Neural Networks," Springer-Verlag, New York, 1996.


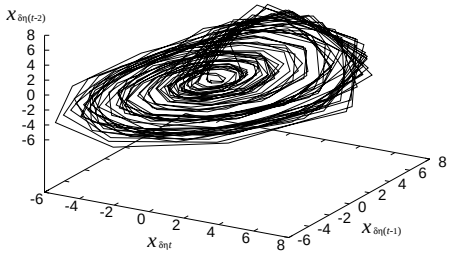


Figure 2: Training data
(Noisy Rössler system)

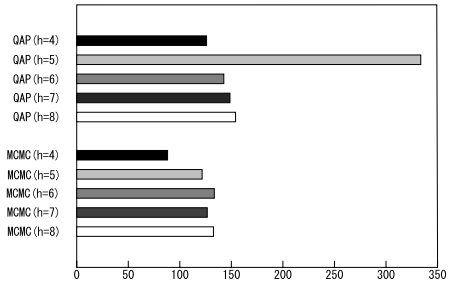


Figure 3: Average squared errors (up to 80 steps) for various models with QAP and MCMC

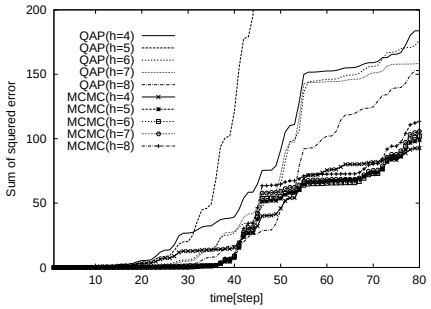


Figure 4: Evolutions of cumulative squared errors ($test$ 1)

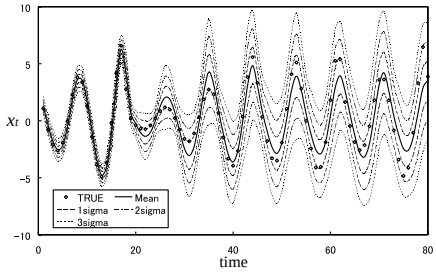


Figure 5: Predictive mean and error bar ($test$ $2, h = 4$)