Bootstrap Prediction Intervals for Nonlinear Time-Series

Daisuke Haraki[1], Tomoya Suzuki[2], and Tohru Ikeguchi[1]

[1] Graduate School of Science and Engineering, Saitama University,
225 Shimo-Ohkubo, Sakura-ku, Saitama-city 338-8570, Japan
daisuke@nls.ics.saitama-u.ac.jp
[2] Department of Information Systems Design, Doshisya University,
1-3 Tatara Miyakodani, Kyotanabe-city 610-0394, Japan

Abstract. To evaluate predictability of complex behavior produced from nonlinear dynamical systems, we often use normalized root mean square error, which is suitable to evaluate errors between true points and predicted points. However, it is also important to estimate prediction intervals, where the future point will be included. Although estimation of prediction intervals is conventionally realized by an ensemble prediction, we applied the bootstrap resampling scheme to evaluate prediction intervals of nonlinear time-series. By several numerical simulations, we show that the bootstrap method is effective to estimate prediction intervals for nonlinear time-series.

1 Introduction

Several prediction methods have been proposed for analyzing complex, possibly chaotic, time-series, for example, the Lorenz' method of analogues[1], the Jacobian matrix estimation[2], the Bootstrap nonlinear prediction[3], and the kNN technique[4]. In the field of nonlinear time-series analysis, these prediction methods are important not only to predict future values of the time-series, but also to analyze long-term unpredictability and short-term predictability, or one of the essential characteristics of deterministic chaos[5].

To evaluate the prediction accuracy, we usually use a normalized root mean square error between true points and predicted points. In such a case, prediction must be described as a point even if its predictability is unreliable due to small size data or noise in the data. In this paper, we evaluate prediction accuracy from another viewpoint: we estimate prediction intervals in which the future point would fall.

In statistical literatures, several attempts have been made on evaluation of prediction intervals. For example, in Ref.[6], the bootstrap method is used to evaluate prediction intervals for autoregressive models. In Ref.[7], a parametric bootstrap method is applied to financial time-series. In Ref.[8], the endogenous lag order method of Kilian[9] applied to sieve bootstrap prediction intervals. However, an application of the bootstrap method is usually evaluated through a linear model such as ARMA.

E. Corchado et al. (Eds.): IDEAL 2006, LNCS 4224, pp. 155–162, 2006.

Although a nonlinear model for a stochastic volatility is used in Ref.[7], however, there is no application of the bootstrap method to more general class of nonlinear dynamical systems, possibly producing chaotic response. In this paper, we apply the concept to the prediction problem of nonlinear, possibly chaotic, time-series.

The concept of the prediction intervals does not directly evaluate the prediction accuracy but statistically estimates a spatial distribution of the future point which might be included in a bounded interval. One of the conventional methods for evaluating such a prediction interval is the ensemble prediction[10]. However, because the ensemble prediction needs large size data to evaluate ensemble properties, estimated prediction intervals[6,7] might be unreliable in the case that we cannot use large size data. In Ref.[3], we have already reported effectiveness of the bootstrap resampling scheme[11] for nonlinear prediction even if the data size is small. This paper [3] showed that the bootstrap resampling method improves prediction accuracy for nonlinear chaotic dynamical systems.

Thus, it is expected that the concept of the bootstrap method would also work well for the present issue. In this paper, we compared the performances of the ensemble prediction and the bootstrap method on the basis of accuracy or efficiency of estimated prediction intervals. By several numerical simulations, we reveal that the bootstrap method has advantages on the evaluation of prediction intervals for nonlinear time-series.

2 Local Linear Prediction Methods

Although there are several local linear prediction methods[1,2,4,5], we introduced two basic methods[1,2] which are used for estimating prediction intervals in the following sections. At first, let us consider a nonlinear dynamical system:

$$\boldsymbol{x}(t+1) = \boldsymbol{f}(\boldsymbol{x}(t)), \tag{1}$$

where $\boldsymbol{f}$ is a k-dimensional nonlinear map, $\boldsymbol{x}(t)$ is a k-dimensional state at time t. To estimate the Jacobian matrix of $\boldsymbol{f}$, we linearize Eq.(1) as follows:

$$\delta\boldsymbol{x}(t+1) = \boldsymbol{Df}(\boldsymbol{x}(t))\delta\boldsymbol{x}(t), \tag{2}$$

where $\boldsymbol{Df}(\boldsymbol{x}(t))$ is the Jacobian matrix at $\boldsymbol{x}(t)$, and $\delta\boldsymbol{x}(t)$ is an infinitesimal deviation at $\boldsymbol{x}(t)$. To evaluate $\boldsymbol{Df}(\boldsymbol{x}(t))$ only with local information at $\boldsymbol{x}(t)$, we first extract a near-neighbor set of $\boldsymbol{x}(t)$. Let us denote the i-th near neighbor of $\boldsymbol{x}(t)$ by $\boldsymbol{x}(t_{k_i})$, $(i = 1, 2, \ldots, M)$. Here, M is the total number of near neighbors. After temporal evolution, displacement vectors can be denoted as $\boldsymbol{y}_i = \boldsymbol{x}(t_{k_i}) - \boldsymbol{x}(t)$ and $\boldsymbol{z}_i = \boldsymbol{x}(t_{k_i}+1) - \boldsymbol{x}(t+1)$. Here, $\boldsymbol{y}_i$ corresponds to $\delta\boldsymbol{x}(t)$, and $\boldsymbol{z}_i$ corresponds to $\delta\boldsymbol{x}(t+1)$ in Eq.(2). If the norms of $\boldsymbol{y}_i$ and $\boldsymbol{z}_i$ and the corresponding temporal evolution is small enough, we can approximate the relation between $\boldsymbol{z}_i$ and $\boldsymbol{y}_i$ by the linear equation: $\boldsymbol{z}_i = \boldsymbol{G}(t)\boldsymbol{y}_i$, where the matrix $\boldsymbol{G}(t)$ is an estimated Jacobian matrix $\boldsymbol{Df}(\boldsymbol{x}(t))$ in Eq.(2). Then, we estimate

$\boldsymbol{G}(t)$ by the least-square-error fitting which minimizes the average square error $S = \frac{1}{M}\sum_{i=1}^{M}|\boldsymbol{z}_i - \boldsymbol{G}(t)\boldsymbol{y}_i|$. In other words, we can estimate $\boldsymbol{G}(t)$ by the following equations: $\boldsymbol{G}(t)\boldsymbol{W} = \boldsymbol{C}$, where $\boldsymbol{W}$ is the variance matrix of $\boldsymbol{y}_i$, and $\boldsymbol{C}$ is the covariance matrix between $\boldsymbol{y}_i$ and $\boldsymbol{z}_i$. If $\boldsymbol{W}$ has its inverse matrix, we can obtain $\boldsymbol{G}(t)$ from $\boldsymbol{G}(t) = \boldsymbol{C}\boldsymbol{W}^{-1}$[2,12].

Because we do not know a future value of $\boldsymbol{x}(t)$, we cannot use dynamical information of $\boldsymbol{z}_i$, and then cannot have direct information of $\boldsymbol{G}(t)$. To solve the problem, we use the information of the nearest neighbor $\boldsymbol{x}(t_{k_0})$ of $\boldsymbol{x}(t)$. Then, we calculate a displacement vector $\boldsymbol{y}' = \boldsymbol{x}(t) - \boldsymbol{x}(t_{k_0})$. Next, we can estimate the Jacobian matrix $\boldsymbol{G}(t_{k_0})$ at $\boldsymbol{x}(t_{k_0})$ by the above procedure. If we define $\hat{\boldsymbol{x}}(t+1)$ as the predicted future value of $\boldsymbol{x}(t)$, we can denote the predicted displacement vector $\hat{\boldsymbol{z}}' = \hat{\boldsymbol{x}}(t+1) - \boldsymbol{x}(t_{k_0}+1)$ by $\hat{\boldsymbol{z}}' = \boldsymbol{G}(t_{k_0})\boldsymbol{y}'$. Then, we can predict $\hat{\boldsymbol{x}}(t+1)$ as follows: $\hat{\boldsymbol{x}}(t+1) = \boldsymbol{G}(t_{k_0})(\boldsymbol{x}(t) - \boldsymbol{x}(t_{k_0})) + \boldsymbol{x}(t_{k_0}+1)$. Repeating the scheme for p time iteratively, we can predict the p step future of $\boldsymbol{x}(t)$[13].

We introduced another prediction method to estimate a dynamical system $\boldsymbol{f}$[1,5]. First, in this method, we search for the near neighbors $\boldsymbol{x}(t_{k_i})(i = 0, 1, \ldots, K)$ of $\boldsymbol{x}(t)$ on the reconstructed attractor. Then, we calculate a future value of $\boldsymbol{x}(t)$ as

$$\hat{\boldsymbol{x}}(t+1) = \sum_{i=0}^{K}\exp(-d_i)\boldsymbol{x}(t_{k_i}+1)\Big/\sum_{i=0}^{K}\exp(-d_i),$$

where $d_i = |\boldsymbol{x}(t_{k_i}) - \boldsymbol{x}(t)|$. This method is called a weighted average prediction.

3 Estimating Prediction Intervals

As a conventional measure to evaluate the prediction accuracy, we can use a normalized root mean square error:

$$E = \sqrt{\sum_{t=1}^{N}\sum_{d=1}^{k}(x_d(t+1) - \hat{x}_d(t+1))^2}\Bigg/\sqrt{\sum_{t=1}^{N}\sum_{d=1}^{k}(x_d(t+1) - \bar{x}_d)^2},$$

where N is the data length, $\bar{x}_d(t+1)$ is the d-th variable of a predicted point, and $\bar{x}_d = \dfrac{1}{N}\displaystyle\sum_{t=1}^{N}x_d(t)$. For evaluating the prediction accuracy, E is basic and essential. However, any prediction methods cannot predict future points perfectly. Instead, it is important to offer a prediction interval in which the true future point might be included.

3.1 Ensemble Prediction

Generally, the prediction intervals can be generated by the ensemble prediction[10] whose technical procedure is described as follows: first, we select several ensemble samples $\boldsymbol{x}^{(m)}(t)(m = 0, 1, 2, \ldots, M)$ from near neighbors $\boldsymbol{x}(t_{k_i})$. In this

paper, we set $M = K$, where K is the number of near neighbor data. We predict a future state of $\boldsymbol{x}^{(m)}(t)$ as $\hat{\boldsymbol{x}}^{(m)}(t+1)$ with estimating $\boldsymbol{f}$ by some prediction method, and we calculate the future of $\boldsymbol{x}(t)$ as

$$\hat{\boldsymbol{x}}(t+1) = \frac{1}{M}\sum_{m=1}^{M} \hat{\boldsymbol{x}}^{(m)}(t+1).$$

Next, to perform multi-steps prediction, we predict the futures of $\hat{\boldsymbol{x}}^{(m)}(t+1)$, respectively. Then, repeating this scheme for p times, we can predict the p step future as $\hat{\boldsymbol{x}}^{(m)}(t+p)$.

The prediction intervals can be calculated from the spatial distribution of $\hat{\boldsymbol{x}}^{(m)}(t+p)$. In this paper, the prediction intervals $\boldsymbol{E}(t+p)$ is defined by an ellipse, the center of which is $\bar{\boldsymbol{x}}(t+1)$. Then, the equation of the ellipse is defined by

$$\boldsymbol{E}(t+p) = \sum_{i=1}^{d} \frac{\hat{z}_i^2(t+p)}{\sigma_i^2} - 1$$

where $\hat{z}_i^(t+p)$ is the i-th principle component, obtained by the application of PCA to the data set $\hat{\boldsymbol{x}}^{(m)}(t+p), (m = 1, 2, \ldots, M)$, and σ_i corresponds to the variance.

3.2 Prediction Intervals by Using the Bootstrap Method

Because the ensemble prediction needs large data sets to evaluate ensemble properties, estimated prediction intervals might be unreliable in the case that we cannot use large size data. In Ref.[3], we have already reported effectivity of the bootstrap method[11] for nonlinear prediction problem even if data size is small. Thus, to evaluate more accurate prediction intervals and to perform stable prediction, it is expected that the bootstrap method may work as well.

The bootstrap resampling scheme[11] is described as follows. First, we selected near-neighbor points of $\boldsymbol{x}(t)$ to predict $\boldsymbol{x}(t)$. The data set is denoted by $\boldsymbol{D} = \{\boldsymbol{x}(t_{k_0}), \boldsymbol{x}(t_{k_1}), \cdots, \boldsymbol{x}(t_{k_L})\}$, where L is the number of near-neighbor points to make a predictor. In this paper, we set $L = M$. Next, we performed a sampling with replacement of $\boldsymbol{D}$ to obtain a new near-neighbor data set $\boldsymbol{D}^{(1)} = \{\boldsymbol{D}^{(1,1)}, \boldsymbol{D}^{(1,2)}, \cdots, \boldsymbol{D}^{(1,l)}, \cdots, \boldsymbol{D}^{(1,L)}\}$. Here, $\boldsymbol{D}^{(1,l)}$ means the l-th sampling with replacement at the first bootstrap trial. Then, we estimate a predictor $\tilde{\boldsymbol{f}}^{(1,l)}$ on each $\boldsymbol{D}^{(1,l)}$, and we predict a future point of $\boldsymbol{x}(t)$ by $\tilde{\boldsymbol{x}}^{(1,l)}(t+1) = \tilde{\boldsymbol{f}}^{(1,l)}(\boldsymbol{D}^{(1,l)})$.

We repeated such bootstrap estimates for B times. Namely, the b-th bootstrap predicted point is described by $\tilde{\boldsymbol{x}}^{(b,l)}(t+1) = \tilde{\boldsymbol{f}}^{(b,l)}(\boldsymbol{D}^{(b,l)})$, where $b = 1, 2, \ldots, B$. Next, we predict the future of each bootstrap predicted point $\{\tilde{\boldsymbol{x}}^{(b,1)}(t+1), \cdots, \tilde{\boldsymbol{x}}^{(b,L)}(t+1)\}$. Therefore, predicted two-steps futures of bootstrap estimates was described as $\tilde{\boldsymbol{x}}^{(b,l)}(t+2)$ whose size is $(B \cdot L)^2$. That is, this dimension is $(B \cdot L)^p$. If

we perform a p-steps prediction with the bootstrap method, the computational load would grow exponentially. Thus, in this paper, at the second prediction step, we randomly select B elements from $\{\tilde{\boldsymbol{x}}^{(b,l)}(t+2)\}$ to prevent the number of bootstrap samples from exploding exponentially. Repeating this scheme for p times, we can predict the p-step future of bootstrap estimates $\{\tilde{\boldsymbol{x}}^{(b,l)}(t+p)\}$, whose size is B^2. Finally, we decided the prediction interval $\boldsymbol{R}(t+p)$ in the same way as the ensemble prediction as described in 3.1.

4 Numerical Simulations

To confirm the validity of estimating prediction intervals by the proposed method, we applied the proposed method to an example test: the Ikeda map[14], which is described as follows:

$$\begin{cases} x(t+1) = a + b\big(x(t)\cos(\theta(t)) - y(t)\sin(\theta(t))\big) \\ y(t+1) = \quad b\big(x(t)\sin(\theta(t)) + y(t)\cos(\theta(t))\big), \end{cases}$$

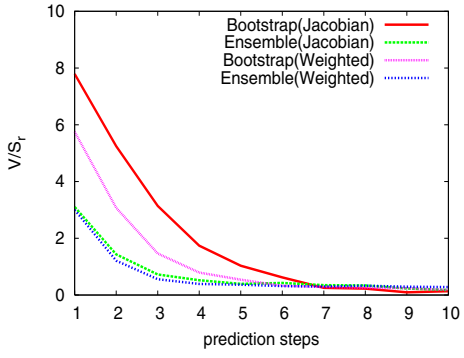
$$\theta(t) = \kappa - \alpha/(1 + x^2(t) + y^2(t)),$$

where a, b, κ, and α are parameters. The Ikeda map is suitable to check the validity of the proposed method because it has higher order nonlinearity. In simulations, the parameters were set as $a = 1.0$, $b = 0.9$, $\kappa = 0.4$ and $\alpha = 6.0$, the data length of $x(t)$ and $y(t)$ is 1,000, respectively. Then, we disturbed the system both by observational and dynamical noise. In this paper, the noise level is quantified by the signal-to-noise ratio, which is calculated by SNR[dB] $= 10\log_{10}\sigma_o^2/\sigma_\eta^2$, where σ_o^2 is the variance of the original data and σ_η^2 is the variance of Gaussian observational/dynamical noise.

For estimating prediction accuracy, we introduced a measure to evaluate the prediction interval: we counted how many times the true point were included in the prescribed prediction interval. The number is denoted by V, which is averaged on several trials. We also introduced the size of prediction interval S_r, defined by $S_r(p) = \sqrt{\sum_{d=1}^{k}\hat{\sigma}_d^2(t+p)}$ in the case of the ensemble prediction, or $S_r(p) = \sqrt{\sum_{d=1}^{k}\tilde{\sigma}_d^2(t+p)}$ in the case of the bootstrap method, and p is the prediction step.

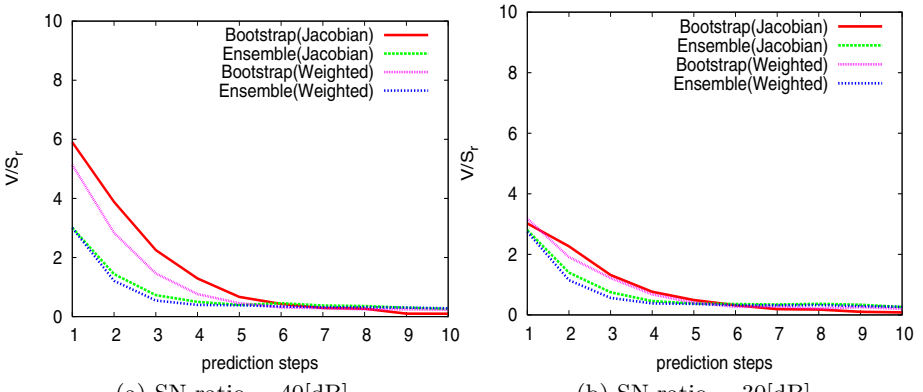
We compared two prediction methods: the Jacobian matrix estimation, and the weighted average prediction. Figs.1–3 show the results of the comparisons among four cases to estimate prediction intervals: the bootstrap method with the Jacobian prediction or the Lorenz' method of analogues, or the ensemble prediction with the same local prediction methods. Figure 1 shows the result of the noiseless data, Fig.2 shows the case that the data are disturbed by observational noise and Fig.3 shows the case that the data are disturbed by dynamical noise.

To evaluate applicability of the proposed method to real data, the data might be produced from a nonlinear chaotic dynamical system. we applied our method



(a) Noiseless data

Fig. 1. Comparisons of four methods to estimate prediction intervals for noiseless data. The horizontal axis shows the prediction steps p, and the vertical axis shows the estimation accuracy $V(p)/S_r(p)$.



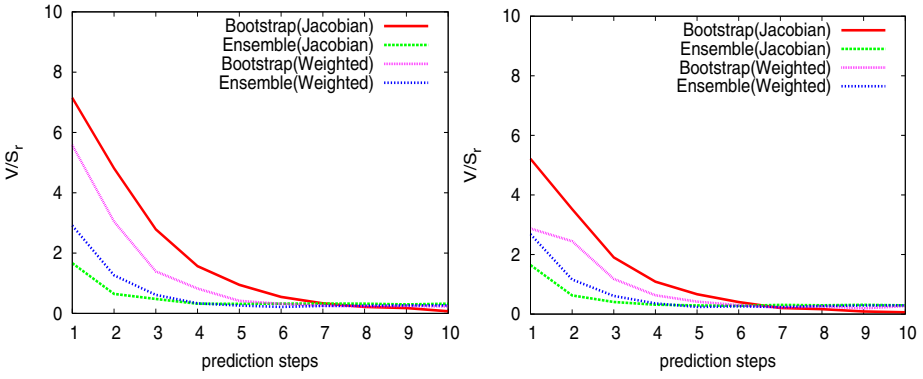
(a) SN ratio = 40[dB] (b) SN ratio = 30[dB]

Fig. 2. The same as Fig.1, but to the observational noise data

to the Japanese vowels /a/ which is example of a real time-series. This data is suitable for benchmark tests because this data has been analyzed and discussed in several studies[14-16].

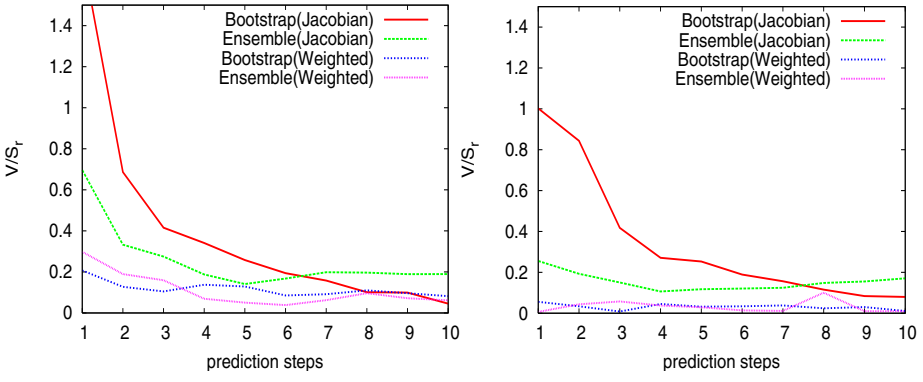
5 Discussion

These results show that the bootstrap method is more reasonable to make efficient prediction intervals if p is less than six. Namely, the bootstrap prediction method adjusts intervals size more accurately and efficiently than the ensemble prediction method. The performance of V/S_r of the bootstrap method and the ensemble method is almost the same as the observational/dynamical noise level



(a) SN ratio = 40[dB] (b) SN ratio = 30[dB]

Fig. 3. The same as Fig.2, but to the dynamical noise data



(a) embedding dimension = 3 (b) embedding dimension = 4

Fig. 4. Results of the Japanese vowels /a/. The axis is same as Fig.1.

becomes larger. In the case of noiseless data, the bootstrap method with the Jacobian prediction showed the best performance.

6 Conclusion

In this paper, we proposed a new framework for estimating the prediction intervals by using the bootstrap method with local-linear prediction methods. In particular, the proposed framework improves short-term predictability comparing to the conventional ensemble prediction. Moreover, the bootstrap method adjusts the size of the prediction intervals effectively according to the difficulty of prediction. The authors would like to thank Dr. Hiroki Hashiguchi for his valuable comments and discussions. The research of TI was partially supported by Grant-in-Aids for Scientific Research (C) (No.17500136) from JSPS.

References

1. E. N. Lorenz. Atmospheric predictability as revealed by naturally occurring analogues. *J. Atmospheric Sciences*, Vol.26, pp.636–646, 1969.
2. M. Sano and Y. Sawada. Measurement of the lyapunov spectrum from a chaotic time series. *Physical Review Letters*, Vol.55, No.10, pp.1082–1085, 1985.
3. D. Haraki, T. Suzuki, and T. Ikeguchi. *Bootstrap Nonlinear Prediction. submitted to Physical Review E*, 2005.
4. A. T. Lora, J. M. R. Santos, J. C. Riquelme, A. G. Expósito, and J. L. M. Ramos. Time-series prediction: Application to the short-term electric energy demand. *Lecture Notes in Computer Science*, Vol.3040, pp.577–586, 2004.
5. G. Sugihara and R. M. May. Nonlinear forecasting as a way of distinguishing chaos from measurement error in time series. *Nature*, Vol.344, pp.734–741, 1990.
6. M. Grigoletto. Bootstrap prediction intervals for autoregressions: Some alternatives. *International Journal of Forecasting*, Vol.14, pp.447–456, 1998.
7. Yun-Huan Lee and Tsai-Hung Fan. Bootstrapping prediction intervals on stochastic volatility models. *Applied Economic Letters*, Vol.13, pp.41–45, 2006.
8. A. M. Alonso, D. Pẽna, and J. Romo. Introducing model uncertainty in time series bootstrap. *Statistica Sinica*, Vol.14, pp.155–174, 2004.
9. L. Kilian. Accounting for lag uncertainty in autoregressions: The endogenous lag order bootstrap algorithm. *Journal of Time Series Analysis*, Vol.19, pp.531–548, 1998.
10. T. Hurukawa and S. Sakai. Ensemble prediction. *Tokyo-doh Press*, 2004, in Japanese.
11. B. Efron and R. J. Tibshirani. *An Introduction to the Bootstrap. Chapman and Hall*, 1993.
12. J. P. Eckmann, S. O. Kamphorst, D. Ruelle, and S. Ciliberto. Lyapunov exponents from time series. *Physical Review A*, Vol.34, No.6, pp.4971–4979, 1986.
13. J. D. Farmer and J. J. Sidorowich. Predicting chaotic time series. *Physical Review Letters*, Vol.59, No.8, pp.845–848, 1987.
14. K. Ikeda. Multiple-valued stationary state and its instability of the transmitted light by a ring cavity system. *Optics Communications*, Vol.30, No.2, pp.257–261, 1979.