Estimating structure of multivariate systems with genetic algorithms for nonlinear prediction

Tomoya Suzuki,[1] Yuta Ueoka,[2] and Haruki Sato[3]
[1]*Department of Intelligent Systems Engineering, College of Engineering, Ibaraki University, 4-12-1 Nakanarisawa-cho, Hitachi, Ibaraki 316-8511, Japan*
[2]*Graduate School of Engineering, Doshisha University, 1-3 Tatara-Miyakodani, Kyotanabe-shi, Kyoto 610-0321, Japan*
[3]*Department of Information System Design, Faculty of Science and Engineering, Doshisha University, 1-3 Tatara-Miyakodani, Kyotanabe-shi, Kyoto 610-0321, Japan*
(Received 4 June 2009; published 17 December 2009)

Although we can often observe time-series data of many elements, these elements do not always interact with each other. This paper proposes a scheme to estimate the interdependency among observed elements only by time-series data, which is useful for selecting essential elements to optimize multivariate prediction model. Because this estimation is a sort of combinatorial optimization problems, we applied the genetic algorithm as a method to moderate this problem. Through some simulations, we confirmed performance of our method, which can identify interaction of multivariate system and can improve its prediction accuracy. Especially, our method can be applied to predict real foreign-exchange markets even if system has nonstational property and its structure changes dynamically.

DOI: 10.1103/PhysRevE.80.066208 PACS number(s): 05.45.Tp, 89.75.Fb, 05.10.−a

I. INTRODUCTION

In the real world, there are many complex systems whose elements interact with each other like a network and constitute multivariate systems. The behavior of these systems is too complex to be controlled and to be predicted. In this paper, we discuss how to predict such complex multivariate systems effectively.

If we predict the behavior of element i $(i=1,2,\ldots,N)$ constituting a system, we can use some behavior of observed elements to make a prediction model. However, we are not sure which observed elements relate to the element i. If we use elements disrelated with the element i, prediction model becomes complex and its generalization ability declines because of overfitting for learning data [1–3], and then prediction accuracy worsens. Namely, we have to constitute a prediction model only by the set of elements interacting with element i, $\{j(i)\}(j(i)=1,2,\ldots,N_i)$, and we need the method to estimate such elements $\{j(i)\}$ from all elements. Here, elements $\{j(i)\}$ are explanatory variables for the objective variable i. If we can estimate $\{j(i)\}$, we can predict future behavior of i with the time-series data of $\{j(i)\}$ as learning data. Moreover, estimated results can be clue to analyze complex systems.

For this motivation, several statistical measures—the correlation coefficient, the partial correlation coefficient, etc. could be useful to estimate interaction from the viewpoint of similarity of each element's behavior. However, because real systems often have nonlinearity, linear statistics are not always useful. In this study, we identity interactions from the viewpoint of nonlinear prediction accuracy of previous behavior. However, because this numerical cost grows exponentially according to the number of elements, this identification is a sort of combinatorial optimization problems. Although it is almost impossible to find the exact optimum solution, there are several metaheuristic searches [4] to find very good near-optimum solution in a reasonable computational time. In this paper, we applied the genetic algorithms (GA) as a more popular method to moderate this optimization problem.

Next, we predict the future behavior of element i with the optimized nonlinear prediction model based on estimated interactions. Here, there are so many nonlinear prediction methods [5–10] that can be categorized into two classes: global and local approximations of relationship between element i and elements $\{j(i)\}$. In particular, global approximations are often performed by neural network or radial basis functions [7]. In our proposed scheme, simple prediction method is more desirable because we have to consider the numerical cost of GA. Therefore, our scheme is based on simple local linear approximation method [5] proposed by Lorenz.

To confirm the efficiency of our scheme to estimate the structure and predict its multivariate system, we perform some simulations with a numerical model of complex systems, the coupled map lattice (CML) [11], and evaluate the accuracy of estimated structure and that of predicted behavior. In addition, we compare prediction methods between the optimized prediction model by our scheme and a lazy prediction model based on all observed elements, that is, without any selection of elements. By comparing each prediction accuracy, we evaluate the efficiency of selecting elements to optimize prediction model.

Moreover, as application to real systems, we predict foreign-exchange markets composing a multivariate system. Because it is possible that the mechanism of real systems is dynamically changed by external effects such as international accidents and politicians' interviews, we have to iterate the optimization of prediction model with GA. In order to confirm the importance of this dynamical optimization, we perform simulations to predict a real financial system and numerical complex systems.

II. ESTIMATION OF INTERACTIONS AMONG ELEMENTS BY THE BEST PREDICTION

A. Prediction model based on direct interactions

We denote time-series data of element i by $x_i(t)$ and denote the behavior of a multivariate system by

$$\boldsymbol{v}(t) = \{x_1(t), x_2(t), \ldots, x_i(t), \ldots, x_N(t)\}, \tag{1}$$

which means that we observed the behavior of N elements. In this study, to predict future behavior of $\boldsymbol{v}(t)$, we use the local linear approximation method [5]. If we perform prediction of $\boldsymbol{v}(t)$ by all elements, we first find some near neighbors $\boldsymbol{v}(t_n)$, $n=1,\ldots,m$, from the previous behavior, that is, $t-L \leq t_n < t$ where L is the length of learning data. Then, we predict one-step future behavior of $\boldsymbol{v}(t)$ by

$$\tilde{\boldsymbol{v}}(t+1) = \frac{1}{m}\sum_{n=1}^{m} \boldsymbol{v}(t_n+1). \tag{2}$$

This prediction method approximates temporal displacement of the present state $\boldsymbol{v}(t)$ linearly with local near neighbors. However, from the viewpoint of the whole of approximated state space, this approximation globally works as nonlinear prediction because each set of neighbors and each approximation are different locally according to state of $\boldsymbol{v}$. Then, the number of neighbors m should be set so as to keep local property of $\boldsymbol{v}(t)$ and to surround $\boldsymbol{v}(t)$ with $\boldsymbol{v}(t_n)$. Therefore, if we denote the dimension of $\boldsymbol{v}$ as d, we set $m=d+1$.

If we aim to predict element i, we can obtain its predicted value $\tilde{x}_i(t+1)$ from the i-th element of $\tilde{\boldsymbol{v}}(t+1)$. However, this prediction model does not work well because all observed elements do not always compose the same system like Eq. (1). Namely, there is no guarantee that these elements interact with element i that we aim to predict. If element i is caused only by elements i_1, i_2, and i_3, the structure is denoted as

$$x_i(t+1) = \boldsymbol{F}[x_i(t); x_{i_1}(t), x_{i_2}(t), x_{i_3}(t)], \tag{3}$$

where $\boldsymbol{F}$ is a dynamics of system, element i means the objective variable, and element $i_{1\sim 3}$ means explanatory variables. In this case, we have to modify Eq. (1) into

$$\boldsymbol{v}(t) = \{x_i(t), x_{i_1}(t), x_{i_2}(t), x_{i_3}(t)\}, \tag{4}$$

and perform one-step prediction approximating the dynamics $\boldsymbol{F}$ locally by Eq. (2). This prediction model can select better neighbors $\boldsymbol{v}(t_n)$ and can improve prediction accuracy more than using Eq. (1), which causes overfitting to the structure disrelated with element i, and which decreases generalization ability [1–3] for predictions after learning data.

B. Applying GA algorithm to search the best prediction model

If we can find the best combination optimizing prediction accuracy, it is possible to estimate elements $\{j(i)\}$ interacting with element i. On the contrary, if we do not use suitable elements to make $\boldsymbol{v}(t)$ like Eq. (4), $\boldsymbol{v}(t)$ cannot reconstruct the dynamics of multivariate system properly, and prediction accuracy must decline because suitable neighbors $\boldsymbol{v}(t_n)$ cannot be selected for the local linear prediction. Here, the number of combinatorial pairs C for making $\boldsymbol{v}(t)$ grows exponentially to the number of observed elements N, that is, $C=2^N$. We have to search the optimum prediction model from these combinatorial pairs. Namely, to identify interactions, that is, to search the set of elements $\{j(i)\}$ is a sort of combinatorial optimization problems. In this paper, we apply the genetic algorithm as a method to moderate this problem. Moreover, we make prediction models by using each combinatorial pair of elements, and estimate each fitting accuracy of learning data as the validity of prediction model and interaction among elements.

First, we initialize each genotype of GA as random binary series like $g_i=\{11001\cdots\}$. In our study, we prepared 30 genotypes. Here, $g_i(j)=0$ means that element j is not used to predict element i as $\boldsymbol{v}(t)$ of Eq. (4), and $g_i(j)=1$ means that element j is used. Then, to evaluate the goodness of each genotype g_i, we estimate its fitting accuracy by predicting behavior of element i. Here, to avoid misestimation by overfitting for learning data and to evaluate generalization ability of prediction model, we used the K-fold cross-validation (CV) method [3,12], which is one of the resampling schemes and is applied for time-series prediction [13–15]; we divided learning data $x(T)$, $t-L \leq T \leq t$, into K subsamples equally. Here, a single set $x(T_k)$, $k=1,2,\ldots,K$, was used as testing data to estimate modeling accuracy, and the remaining $K-1$ subsamples $x(T_{K-1})$ were used as training data to estimate local flow of the dynamics $\boldsymbol{F}$. Namely, we constituted $\boldsymbol{v}(T_{K-1})$ based on each genotype g_i, and can approximate $\boldsymbol{F}$ linearly with local near neighbors $\boldsymbol{v}(T_n) \in \boldsymbol{v}(T_{K-1})$. Then, $\boldsymbol{v}(T+1) \in \boldsymbol{v}(T_K)$ is predicted from the previous data $\boldsymbol{v}(T) \in \boldsymbol{v}(T_k)$ according to the approximated $\tilde{\boldsymbol{F}}$. And then, we estimated a prediction accuracy $e_{k,i}$ with the correlation coefficient between the true testing data $x_i(T_k)$ and the predicted values $\tilde{x}_i(T_k)$. Similarly, this process is repeated K times with each of the subsamples used exactly once as the testing data. Finally, we calculated the mean value of the set $\{e_{k,i}\}$ as e_i, which means a general fitting error considering generalization ability of prediction model without overfitting learning data $x(T)$. Then, we considered this fitting error as the goodness e_i of each genotype g_i. In our study, we set $K=2$ for saving numerical cost.

Next, by GA's algorithms such as the crossover and the mutation, each genotype g_i is evolved through generalizations. At each generalization step, because better genotypes having higher goodness are easier to survive, we preferentially select genotypes at random with replacement so that better genotypes are selected more frequently according to e_i. The number of selected genotypes is the same as that of initial genotypes. Then, we mix two genotypes with the crossover algorithm to breed new genotypes; if two genotypes are $g_i'=\{00000\}$ and $g_i''=\{11111\}$, new genotypes become like $g_i'=\{10101\}$ and $g_i''=\{01010\}$. This crossover is called multipoints crossover, whose crossover points are decided randomly. Moreover, the mutation algorithm is used to avoid that genotypes are close to each other. Mutation points are selected randomly, and each gene of selected mutation points is changed; if the value is 1, it is replaced by 0. In particular, the selected genotypes having the highest goodness are treated as elite genotypes, which can survive without

any modification. In our study, we treated 10 (%) of selected genotypes as elite genotypes, and divided the other genotypes for the crossover and for the mutation randomly at the ratio of 9:1. By iterating these algorithms, we can evolve genotypes, and can obtain good near-optimum solution in a reasonable computational time. We set the number of generation to 500. After the optimized genotype g_i^* and its goodness e_i^* are obtained, we record $e_i^* \times g_i^*$ in the i-th line of the matrix $M_{i\leftarrow j}$ as the plausibility of the interaction from element j to element i. By iterating above process and changing a predicted element from $i=1$ to $i=N$, we can estimate the matrix $M_{i\leftarrow j}$ showing directed and weighted interactions among elements. In the case of undirected estimation, we calculate

$$M_{i,j} = \frac{M_{i\leftarrow j} + M_{j\leftarrow i}}{2}, \tag{5}$$

where $M_{j\leftarrow i}$ is calculated by $M_{i\leftarrow j}^T$

C. Evaluating the efficiency of the proposed estimation

To confirm the efficiency of our estimation presented above, we performed numerical simulations by using the following coupled map lattice [11] modeling complex systems:

$$x_i(t+1) = \boldsymbol{F}\left((1-\epsilon)x_i(t) + \frac{\epsilon}{N_i} \sum_{j \in \{j(i)\}(p)}^{N_i} x_j(t) \right), \tag{6}$$

where $x_i(t)$ is time-series data of the i-th element ($i=1,2,\ldots,N$), N_i is the number of elements interacting with the i-th element, ϵ is the strength of interactions, and $\{j(i)\}(p)$ is a set of elements interacting with the i-th element.

Although the regular CML [11] has a ring lattice where each element interacts only both neighbors, this CML of Eq. (6) is modified by the Watts-Strogatz (WS) model [16] so that the topology of interactions can be changed according to p. As the definition of WS model, we first prepare a ring lattice where each element is connected with the nearest κ neighbors, that is, we prepare a regular network [16], and the total number of edges is $\frac{N\kappa}{2}$. In our study, we set $N=100$ and $\kappa=4$. Next, we rewire each edge according to the rewiring probability p, by cutting one side of edge and randomly connecting it to some element. Besides, once rewired edges are fixed. If we set $p=0$, the topology is the regular network. Moreover, we can realize the small-world network [16] by setting $0<p<1$ and the random network [16] by setting $p=1$. In our study, we set $p=1$ because the topology of the random network is the most complex. Moreover, as dynamics $\boldsymbol{F}$, we used the logistic map: $\boldsymbol{F}(x)=1-ax^2$, which is widely analyzed as a sort of chaotic maps. Especially, when $a=2$, each element moves chaotically by the dynamics of the logistic map. Therefore, we set $a=2$ to derive complex behavior. Then, because the system has only undirected interactions, we estimate $M_{i,j}$ by Eq. (5).

As the accuracy of identified interacting elements E, we calculate

$$E = \frac{\sum_{i=1}^{N} |\{\tilde{j}(i)\} \cup \{j(i)\}|}{\sum_{i=1}^{N} |\{j(i)\}|}, \tag{7}$$

where $\{j(i)\}$ corresponds to the true elements interacting with element i, and $\{\tilde{j}(i)\}$ corresponds to the estimated ones. That is, $E=1$ means perfect estimation of interactions.

In addition, in our previous study [17,18], we estimate interactions by using the correlation coefficient $C_{i,j}$, the partial correlation coefficient $P_{i,j}$, the mutual information $I_{i,j}$, and the transfer entropy [19] $T_{i,j}$ between time-series data of observed elements. The mutual information and the transfer entropy can estimate nonlinear interactions, and each statistics are defined as follows:

$$I_{i,j} = \sum_t p[x_i(t),x_j(t)] \log \frac{p[x_i(t),x_j(t)]}{p[x_i(t)]p[x_j(t)]},$$

$$T_{i\to j} = \sum_t p[x_j(t+1),x_j(t),x_i(t)] \log \frac{p[x_j(t+1)|x_j(t),x_i(t)]}{p[x_j(t+1)|x_j(t)]},$$

$$T_{i,j} = \frac{1}{2}(T_{i\to j} + T_{j\to i}),$$

where $T_{i,j}$ is used to estimate an undirected interaction. In this study, these measures $C_{i,j}$, $P_{i,j}$, $I_{i,j}$, and $T_{i,j}$ are used as comparison with our estimation $M_{i,j}$. Then, because each component of these measures shows the possibility of interaction between elements i and j, we detected the set of interacting elements $\{\tilde{j}(i)\}$ according to larger components of each measured matrix. Here, the number of detected interactions is limited to that of the true interactions, that is, $\Sigma_{i=1}^N |\{\tilde{j}(i)\}| = \Sigma_{i=1}^N |\{j(i)\}|$. If a component (i,j) of a matrix is large enough for the detection, it is concluded that elements i and j interact with each other and one element is an explanatory element of the other objective element. Then, we calculate each estimation accuracy comparing $\{\tilde{j}(i)\}$ and $\{j(i)\}$ by Eq. (7). In the case of using each measure: $C_{i,j}$, $P_{i,j}$, $I_{i,j}$, and $T_{i,j}$, each estimation accuracy is denoted as E_C, E_P, E_I, and E_T, respectively.

Figure 1 shows our proposed method changing model parameters p and ϵ of Eq. (6). As shown in shaded regions of Fig. 1(a), there are many cases where the proposed method $M_{i,j}$ is the optimum method. Moreover, even if our proposed method is not the optimum method, the difference between each best accuracy $E^*(p,\epsilon)$ shown in Fig. 1(b) and $E_M(p,\epsilon)$ by our proposed method is small as shown in Fig. 1(c). These results are almost the same if we change the parameters—the number of elements N and the total length of testing and learning data L for the cross-validation method [3].

From results, we confirm that our method can estimate structure of systems and can identify interacting elements $\{j(i)\}$. Namely, our method might be able to improve a prediction model like Eq. (4) against Eq. (1). This possibility is

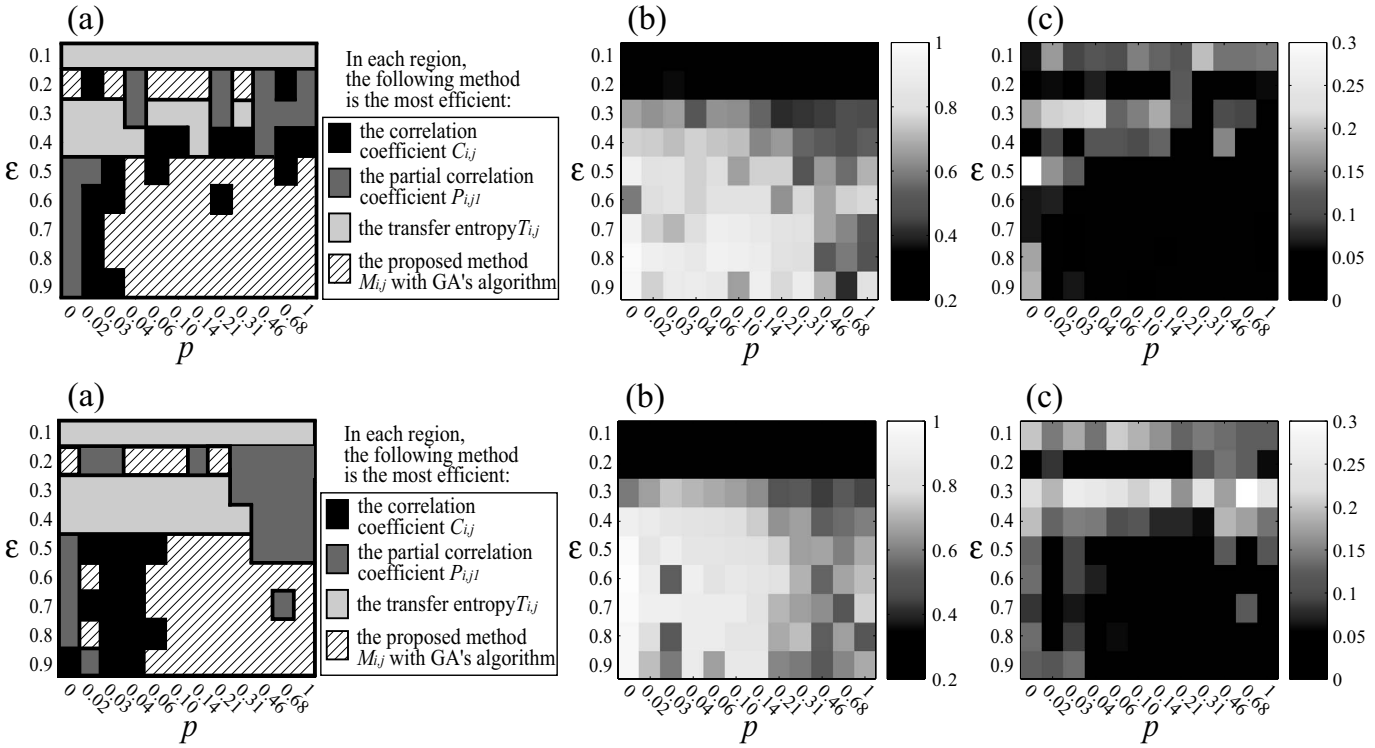

FIG. 1. (a) Phase diagram of the optimum method which maximizes the estimation accuracy of interacting elements $E(p,\epsilon)$ of Eq. (7). (b) The best accuracy $E^*(p,\epsilon)$ estimated by each optimum method shown in Figs. (a). (c) The difference between the best accuracy $E^*(p,\epsilon)$ shown in Figs.(b) and the accuracy $E_M(p,\epsilon)$ by our proposed method $M_{i,j}$, that is, $E^*(p,\epsilon)-E_M(p,\epsilon)$ is shown. Upper figures show the case that $N=30$ and $L=100$, and lower figures show the case that $N=50$ and $L=200$.

verified in Sec. II D. However, Fig. 1(b) shows that it becomes more difficult to estimate interactions as ϵ becomes smaller or p becomes larger. When p is large, a system verges to the random network whose elements are eager to synchronize each other because the shortest path length between elements becomes small [16]. Because of these weak interactions and synchronized system, each set of time-series data obtained through elements is very similar, and it becomes difficult to extract enough information to estimate interactions among elements.

D. Nonlinear prediction based on the estimated interactions

To confirm the improvement of prediction model based on the estimated interactions $\{\tilde{j}(i)\}$ by $M_{i,j}$, we perform predicting future behavior of elements. Besides, for comparison, we also perform lazy predictions based on Eq. (1) with all elements without any selection. In these simulations, to estimate prediction accuracy, we calculate the correlation coefficient between the true time-series data $x_i(t)$ and the predicted time-series data $\tilde{x}_i(t)$. Here, we denote G_i as the prediction accuracy of element i by using the modified $V(t)$ based on $\{\tilde{j}(i)\}$ estimated by $M_{i,j}$, and we denote A_i as that by using the $\boldsymbol{v}(t)$ of Eq. (1) composed by all observed elements.

Figures 2(a) and 2(b) show each mean value $\langle G_i(p,\epsilon)\rangle_i = \frac{1}{N}\Sigma_{i=1}^{N} G_i(p,\epsilon)$ and $\langle A_i(p,\epsilon)\rangle_i = \frac{1}{N}\Sigma_{i=1}^{N} A_i(p,\epsilon)$. We can confirm $\langle G_i(p,\epsilon)\rangle_i > \langle A_i(p,\epsilon)\rangle_i$ in almost all situations. Namely, the prediction accuracy by the proposed method is better than that by using all observed elements, that is, lazy prediction.

Moreover, to verify the advantage of the proposed method carefully, we performed the Wilcoxson's sign rank sum test. Here, we consider null hypothesis as $\{G_i\} \leq \{A_i\}$, and perform one-side test by significant level $\alpha=0.01$. If the null hypothesis is rejected, we can insist the advantage of the proposed method, that is, $\{G_i\} > \{A_i\}$ aggressively. As shown in Fig. 2(c), we can confirm rejections even by $\alpha=0.01$ in almost all situations.

III. DYNAMICAL OPTIMIZATION FOR PREDICTING REAL MULTIVARIATE SYSTEMS

As an application of the proposed method, we tried predicting real foreign-exchange markets in 1996 [20]. This system is composed by 25 types of foreign exchanges [21], and shows us very complex behavior because exchange markets interact with each other. Exchange price of each market is recorded every 30 minutes, and corresponds to $x_i(t)$ as behavior of each element in a multivariate system, so $N=25$ in this case. As presented in Sec. I, predicting real systems accurately is very hard because real systems are often affected by gusty external effects and change their structure dynamically. Therefore, in this section, we dynamically optimized prediction model by GA iterating our proposed method.

To examine the advantage of this dynamical optimization, we compared several prediction methods: the first method (Method 1) uses all elements like Eq. (1), the second method

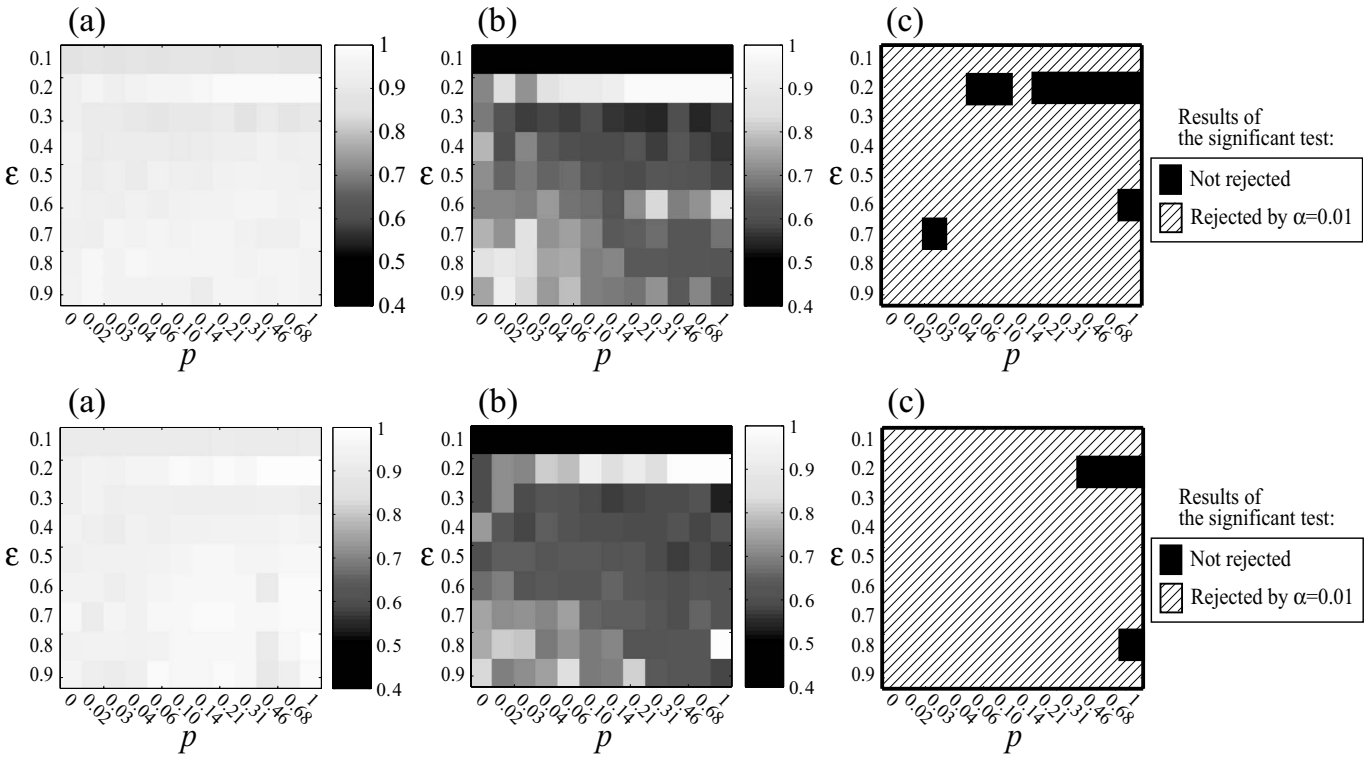

FIG. 2. (a) The prediction accuracy $\langle G_i(p,\epsilon)\rangle_i$ by using the modified $\boldsymbol{v}(t)$ based on the interacting elements $\{\tilde{j}(i)\}$ estimated by the proposed method $M_{i,j}$. (b) The prediction accuracy $\langle A_i(p,\epsilon)\rangle_i$ by using the $V(t)$ of Eq. (1) composed by all observed elements. (c) The result of the significant test of $\{G_i(p,\epsilon)\}>\{A_i(p,\epsilon)\}$ by significant level $\alpha=0.01$. If the null hypothesis $\{G_i(p,\epsilon)\}\leq\{A_i(p,\epsilon)\}$ is rejected, the proposed method is shown as shaded region. Upper figures show the case that $N=30$ and $L=100$ and lower figures show the case that $N=50$ and $L=200$.

(Method 2) uses the only elements selected by the GA g_i^* as introduced in Sec. II B, but the optimization is performed only once at the first prediction and the same selected elements are reused iteratively. Besides, as the last method (Method 3), we performed the dynamical optimization every prediction, and used new elements g_i^* selected by the GA to modify $\boldsymbol{v}(t)$ of Eq. (1).

Moreover, because it is difficult to predict real systems accurately, in this section, we simply predicted whether one-step's future behavior rises or falls. That is, if $\hat{x}_i(t+1)-x_i(t)>0$, we predict that future behavior will rise, or if $\hat{x}_i(t+1)-x_i(t)<0$, we predict that future behavior will fall. Then, as comparison, we also predicted the modified CML of Eq. (6) whose structure does not change unlike real systems.

TABLE I. The results of predicting real foreign-exchange markets. The symbol P_i means prediction accuracy of i-th market, $\langle P_i\rangle_i=\frac{1}{N}\Sigma_{i=1}^{N}P_i$, and R means the percentage of improved elements whose P_i becomes better by Method 2 or Method 3 than P_i of Method 1. Each significance test examines the advantage of Method 2 or 3 against Method 1. For more details, see Sec. III.

	$\langle P_i\rangle_i$ (%)	R (%)	The significance test
Method 1	56		
Method 2	56	44	Not rejected
Method 3	60	96	Rejected by $\alpha=0.01$

The results are shown in Tables I and II. Here, to evaluate the performance of each method, we estimated two measures P_i $(i=1,2,\ldots,N)$ and R. The measure P_i means the percentage of correctly predicting whether i-th market's future movements rise or fall, R means the percentage of improved elements whose P_i becomes better by using Method 2 or Method 3 than P_i of Method 1. Moreover, we performed the Wilcoxson's sign rank sum test whose null hypothesis is that $\{P_i\}$ of Method 2 or Method 3 is not more than that of Method 1, and performed one-side test by the significant level $\alpha=0.01$.

As results, we can confirm the efficiency of the optimization by the GA. Especially, dynamical optimization, Method 3, shows the best prediction performance. Moreover, in real financial system, Table I shows that Method 2 has no difference with Method 1. These results support the fact that this system changes dynamically. Therefore, even Method 2 is insufficient because the optimization is performed only once

TABLE II. The same as Table I, but the modified CML of Eq. (6) was predicted.

	$\langle P_i\rangle_i$ (%)	R (%)	The significance test
Method 1	84		
Method 2	88	73	Rejected by $\alpha=0.01$
Method 3	92	80	Rejected by $\alpha=0.01$

at the first prediction. Namely, we have to optimize prediction model dynamically for predicting real systems.

Table II also shows the efficiency of optimizing prediction model for the modified CML. However, because this system does not change dynamically, not only Method 3 but also Method 2 can improve prediction accuracy. Moreover, the difference between P_i of Method 2 and that of Method 3 cannot be confirmed by the significance test. Therefore, Method 2 is adequate to predict systems having invariant structure and is effective from the viewpoint of numerical cost.

IV. CONCLUSIONS

In this paper, we discussed the method to select capital elements for predicting multivariate systems properly. As the procedure, we identify interactions among elements of a multivariate system, and optimize prediction model. Because the identification and the optimization are considered as a sort of combinatorial optimization problems, we applied the genetic algorithm as a method to moderate the problem. By some numerical simulations comparing with the lazy prediction using all observed elements without any selection, we can conclude as follows:

(i) we can optimize prediction model with previous data by applying GA algorithm.

(ii) We can identify interactions of multivariate systems by referring to the optimized prediction model.

(iii) By using the optimized prediction model, we can also predict new time-series data more accurately.

(iv) In the case that the structure of system dynamically changes like real systems, we can improve prediction accuracy by optimizing prediction model iteratively every prediction.

ACKNOWLEDGMENTS

This research was partially supported by the Grant-in-Aid for Young Scientists (B) under Grant No. 18800061 from the Ministry of Education, Culture, Sports, Science and Technology, and by the Grant-in-Aid for Physicists from the Ken Millennium Corporation.

[1] H. Akaike, in *2nd International Symposium on Information Theory* (Akademiai Kiado, Hungary, 1973), pp. 267–281.
[2] J. Rissanen, *Stochastic Complexity in Statistical Inquiry* (World Scientific, Singapore, 1989).
[3] R. Kohavi, Proceedings of the 14th International Joint Conference on Artificial Intelligence, 1995 (unpublished), Vol. 2, No. 12, pp. 1137–1143.
[4] C. Blum and A. Roli, ACM Comput. Surv. **35**, 268 (2003).
[5] E. N. Lorenz, J. Atmos. Sci. **26**, 636 (1969).
[6] J. D. Farmer and J. J. Sidorowich, Phys. Rev. Lett. **59**, 845 (1987).
[7] M. Casdagli, Physica D **35**, 335 (1989).
[8] G. Sugihara and R. M. May, Nature (London) **344**, 734 (1990).
[9] D. Kugiumtzis, Physica D **112**, 344 (1998).
[10] M. Ragwitz and H. Kantz, Phys. Rev. E **65**, 056201 (2002).
[11] K. Kaneko, in *Formation, Dynamics, and Statistics of Patterns*, edited by K. Kawasaki *et al.* (World Scientific, Singapore, 1990), Vol. I.
[12] R. R. Picard and R. D. Cook, J. Am. Stat. Assoc. **79**, 575 (1984).
[13] R. M. Kunst, Austrian J. Stat. **37**, 271 (2008).
[14] T. P. Lim and S. Puthusserypady, Chaos **16**, 013106 (2006).
[15] T. Suzuki, T. Ikeguchi, and M. Suzuki, Phys. Rev. E **76**, 046202 (2007).
[16] D. Watts and S. Strogatz, Nature (London) **393**, 440 (1998).
[17] Y. Ueoka, T. Suzuki, T. Ikeguchi, and Y. Horio, in Proceedings of 2008 International Symposium on Nonlinear Theory and its Applications (NOLTA), Hungary, 2008 (unpublished), pp. 1–4
[18] Y. Ueoka, T. Suzuki, and S. Yamamoto, in Proceedings of 2009 International Workshop on Nonlinear Circuit and Signal Processing (NCSP'09), USA, 2009 (unpublished), pp. 558–561.
[19] T. Schreiber, Phys. Rev. Lett. **85**, 461 (2000).
[20] We bought the set of historical data from OLSEN Corp. (http://www.olsendata.com/).
[21] The data set includes following 25 exchange markets: USD-DEM, USD-JPY, USD-CHF, GBP-USD, USD-FRF, USD-ITL, USD-NLG, AUD-USD, CAD-USD, USD-SEK, USD-DKK, USD-FIM, USD-ZAR, USD-ESP, USD-BEF, USD-XEU, USD-SGD, USD-MYR, DEM-JPY, DEM-SEK, DEM-ESP, DEM-ITL, DEM-FIM, DEM-FRF, GBP-DEM, AUD-USD.