A Study of Improving the Coherence in Multi-Step Ahead Forecasting

Chao-Fu Hong[1] Yung-Sheng Liao[1] Mu-Hua Lin[2] Tsai-Hsia Hong[3]

[1]Department of Information Management, Aletheia University
[2]Department of Management Information Systems National Chengchi University
[3]General Education Center, National Defense Management College, National Defense University

Abstract

The traditional multi-step ahead prediction is based on sequential algorithm to run multi-step ahead prediction and it brings error propagation problem. Furthermore, the prediction error of multi-step ahead includes both system and propagation errors. Therefore, how to decrease the propagation error has become an important issue in multi-step ahead prediction. In this study we had used the parallel algorithm to avoid the propagation error, but it brought a new problem: the incoherent learning method was used to learn the coherent time series, then, it brought an incoherent problem. Therefore, we proposed a novel parallel algorithm: after parallel algorithm, the system had to run the sequential algorithm again to avoid the incoherent problem. The experimental results evidence that the prediction error of the novel parallel algorithm was smaller than that of the parallel algorithm and the prediction error of multi-step ahead was the same as that of one-step ahead. These results imply that the prediction error of the novel parallel algorithm was approaching the system error. In addition the fractal based GP was used to learn the predicting function. The prediction error was as the radius of the trajectory line. Because the fractal was drawn by the pipe line, it indicates that the stock's price time series belonged to the non-determinate chaos.

Keywords: Chaos, Fractal, Artificial Intelligence, Evolutionary Computation

1. Introduction

How to increase multi-step ahead prediction precision has been a hot issue in forecast. Pavlidis, Tasoulis, and Vrahatis (2003, 2005) used the neural network to train the fractal based predicting function, and then, they used the predicting function to run the multi-step ahead prediction. The prediction precisions were not as good as that of one-step ahead prediction. Here, the explanation is presented in Fig. 1; the multi-step ahead prediction process follows the fractal trajectory to run sequential predictions. When it predicts the value of two-step ahead, the prediction value of one-step ahead is one of the input variables to predict two-step ahead. This process will influence two-step ahead prediction precision, which results the larger prediction error than that of one-step ahead. This kind of error propagation mechanism has been a big problem for multi-step ahead prediction.

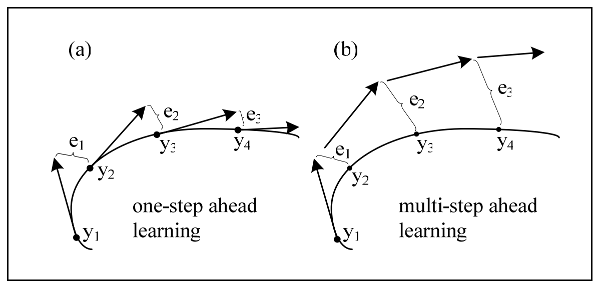

Fig. 1: (a) One-step ahead prediction. (b) Multi-step ahead prediction. (Rembrandt Bakker, 1997)

Parallel Algorithm

In multi-step ahead prediction, we followed the sequential algorithm, as presented in Fig.1*b*, and predicted the value two-step ahead. The prediction value of one-step ahead was used as one of the input variables for predicting. This mechanism resulted in the amplification of prediction error by propagation. It indicates that the propagating mechanism caused the prediction error two-step ahead larger than that one-step ahead. After few steps the prediction errors would be out of control by this propagating mechanism

In this paper, we adopted the parallel algorithm to stop the error propagation, which was only generated by the sequential algorithm. The optimal delay time τ was calculated by Gautama's method (2003), and then, τs' parallel sub-fractal time series was built, based on the delay time τ, as shown in Fig.2. The fractal based GP was used to learn the characteristic of sub-fractal time series. Finally, we got the sub-fractal predicting functions, using every sub-fractal predicting function only to run one-step ahead prediction, and ***integrating all sub-fractal prediction functions' predicting values*** to format multi-step ahead prediction. This kind of algorithm was called the ***parallel algorithm***. There was a drawback: assuming all sub-fractal time series were independent time series; but, in practice, the correlation coefficient between any two sub-fractal time series were very large. It implies that the parallel algorithm would bring the incoherence problem. Therefore, we proposed a novel parallel algorithm to overcome this problem.

	Sub-1	*Sub-2*	*Sub-3*	…	*Sub-τ*
1τ	p_1	p_2	p_3	…	p_τ
2τ	$p_{\tau+1}$	$p_{\tau+2}$	$p_{\tau+3}$	…	$p_{\tau+\tau}$
3τ	$p_{2\tau+1}$	$p_{2\tau+2}$	$p_{2\tau+3}$	…	$p_{2\tau+\tau}$
⋮	⋮	⋮	⋮	⋮	⋮

Fig. 2: sub-fractal time series.

2. Methodology and Experiment

Phase 1: all sub-fractal based GP learning prediction values integrated to format the parallel based multi-step ahead prediction

Gautama's (2003) method was used to determine the optimal delay time (τ), and then, τ was used to decompose the fractal time series for reformatting parallel τs' sub-fractals time series as presented in Fig.2. Furthermore, as shown in Fig.3 the GP and sliding window (τ) were used to extract sub-fractal predicting function from sub-fractal time series. After τs' sub-fractal based predicting functions were built to run one-step ahead prediction, all prediction values were collected after the accomplishment of τ-step ahead prediction. In this parallel algorithm, every sub-fractal predicting function only ran one-step ahead prediction, therefore, the error propagation could be avoided.

p_1	$p_{1+\tau}$	$p_{1+2\tau}$	…	$p_{1+\tau\tau}$	$p_{1+\tau(\tau+1)}$			
	$p_{1+\tau}$	$p_{1+2\tau}$	…	$p_{1+\tau\tau}$	$p_{1+\tau(\tau+1)}$	$p_{1+\tau(2\tau+1)}$		
		$p_{1+2\tau}$	…	$p_{1+\tau\tau}$	$p_{1+\tau(\tau+1)}$	$p_{1+\tau(2\tau+1)}$	$p_{1+\tau(3\tau+1)}$	
			⋮	⋮	⋮	⋮	⋮	⋮

Fig. 3: sub1-fractal GP learning with sliding window

The sub-fractal based forecasting functions are presented in the followings:

$$P_{t+1} = f(P_{t+1-1\tau}, P_{t+1-2\tau}, \ldots, p_{t+1-\tau\tau})$$
$$P_{t+2} = f(P_{t+2-1\tau}, P_{t+2-2\tau}, \ldots, p_{t+2-\tau\tau}) \quad \ldots\ldots (1)$$
$$\vdots$$
$$P_{t+\tau} = f(P_{t+\tau-1\tau}, P_{t+\tau-2\tau}, \ldots, p_{t+\tau-\tau\tau})$$

All sub-prediction values, p_{t+1}, p_{t+2}, …, $p_{t+\tau}$, were integrated to accomplish τ-step ahead prediction. Our model's residuary error was defined by Eq.2.

$$\Delta p = P_{predict} - P_{real} \quad \ldots\ldots (2)$$

The predicting future time series was $p_{predict}$ and the error term Δp was defined by the difference between $p_{predict}$ and raw time series (p_{real}).

Phase 2: a novel parallel based multi-steps ahead prediction

p_1	p_2	p_3	…	p_τ	$p_{\tau+1}$		
	p_2	p_3	…	p_τ	$p_{\tau+1}$	$p_{\tau+2}$	
		p_3	…	p_τ	$p_{\tau+1}$	$p_{\tau+2}$	$p_{\tau+3}$

Fig. 4: sequential fractal GP learning

As described above, the assumption of parallel based multi-steps ahead prediction was that there was no relationship among all sub-fractal time series. Therefore, we collected all sub-fractal prediction values and formatted multi-steps prediction. Many reports (Iba 1999, Iba 2000, Kaboudan 1998, Kaboudan 2005, Kumar 1999) indicate that the most important input variable for forecasting the p_{t+1} (next step's value) was the p_t (today's value). These reports imply that the parallel algorithm would bring the incoherent problem. In this study we proposed a novel method to avoid this incoherent problem: after the parallel based training and the integration of all sub-fractal predicting time series, we obtained predicting time series, and then, we integrated the predicting time series into the sequential fractal based GP to extract one-step ahead predicting function as shown in Fig.4. The predicting function is presented as Eq.3.

$$P_{t+1} = f(P_t, P_{t-1}, \ldots, p_{t-\tau}) \quad (3)$$

Finally, this predicting function was used to adjust multi prediction values to increase its coherent characteristic and prediction precision.

Experiments

In this study, we held two experiments: one was to evidence that the prediction precision of the novel parallel algorithm was better than that of parallel algorithm. The other was to evidence that the residuary error was as good as that of one-step ahead prediction. The experimental flow is shown in Fig.5.

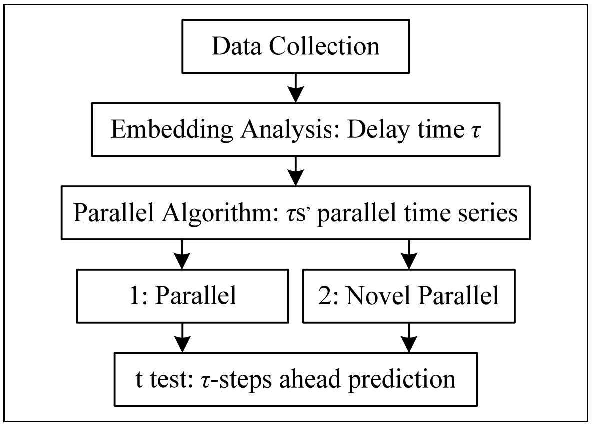

Fig. 5: The experimental flow chart

3. Experiment Results and Discussions

The stock's price data of Lemel were collected from TSEC (Taiwan Stock Exchange Corp.) from 01/02/2001 to 12/31/2003. Total 741 data were collected, and 610 data were used in training predicting function during the training period. The remaining 131 data were used to test the predicting function in testing period. All were raw data.

Parallel Algorithm

The optimal delay time of the stock price time series were determined by Gautama's entropy analysis method. The number of training data was only 610. When τ was larger than 8 (τ>8), τ was defined as 8 (τ=8). The financial time series was decomposed to 8 parallel Sub-Fractal time series as presented in Fig. 2. All stocks' delay time are shown in table 1.

Table 1: the stocks' optimal delay time (τ)

No.	Stock	τ	No.	Stock	τ
2347	Lemel	25	2349	RITEK	23
2324	Compal	24	2379	RT	21
2343	SYSTEX	25	2330	TSMC	20
n2396	PRODISC	20	2394	PREMIER	28
2313	COMPEQMFG.	28			

Phase 1: parallel based multi-steps ahead prediction

The GP, sliding window and parallel τs' sub- fractals time series were used to train τs' sub-fractals' predicting function, and then, the system could carry out predicting functions (Eq.2). After training, Eq.3 was adopted to calculate the prediction error and to format the prediction (residuary) error time series. Every sub-fractal prediction characteristics of multi-steps ahead are presented in Table 2. Even though the interval was 8 days, the prediction error was almost controlled under 10% by sub-fractal based GP.

Table 2: The prediction error by parallel based sub-fractal GP (in training period)

	Mean	Sigma	Skew.	Kurt.	SSE	MSE
F1	1.30	2.43	0.42	17.08	491.13	5.85
F2	4.16	16.84	-8.86	80.26	23,790.41	283.22
F3	1.46	2.01	-0.49	1.80	334.24	3.98
F4	2.38	3.13	-0.47	1.04	817.65	9.73
F5	1.64	2.68	0.00	7.73	595.35	7.09
F6	1.00	1.60	-0.34	2.96	212.36	2.53
F7	2.01	2.59	-0.19	1.48	562.09	6.69
F8	2.87	4.01	-0.73	2.37	1,339.43	15.95

Phase 2: a novel parallel based multi-steps ahead prediction

As described above, the prediction depending upon parallel based sub-fractal GP was calculated to obtain the prediction value and the prediction error was calculated by Eq.3. After that, the sequential training process (fractal based GP), presented in Fig.4, was used to increase the coherence of predicting time series. This predicting function was used to run the final predicting future price and to calculate the prediction error in whole period (training and testing). The fractal based GP was also used to learn one-step ahead prediction. This mechanism neither had the propagation prediction error, nor had the incoherent effect. This kind of prediction error was the system error, which was a good benchmark. The characteristics of three prediction error time series are shown in Table 3.

Table 3: comparison of both algorithms' prediction errors

Paired Samples Test						
	Paired Differences			t	df	Sig. (2-tailed)
	Mean	Std. Deviation	Std. Error Mean			
parallel - novel	-0.81392	14.01341	1.22436	-0.665	130	0.507
parallel - one day	-1.21727	13.91357	1.21563	-1.001	130	0.319
parallel - one day	-0.40336	0.64111	0.05601	-7.201	130	0.000

In table 3, the prediction error of the novel parallel method was significantly smaller than that of parallel algorithm, and one-step ahead predicting algorithm was significantly smaller than that of parallel algorithm. However, the prediction error of the novel parallel method was not significantly smaller than that of one-step ahead predicting algorithm. In Fig. 8, the red solid line was predicted by novel parallel and the blue solid line was predicted by one-step ahead. Even in the testing period the predictability of the novel parallel algorithm was as good as that of one-step ahead prediction algorithm. These results indicate that the predictability of the novel parallel algorithm could overcome the incoherent problem.

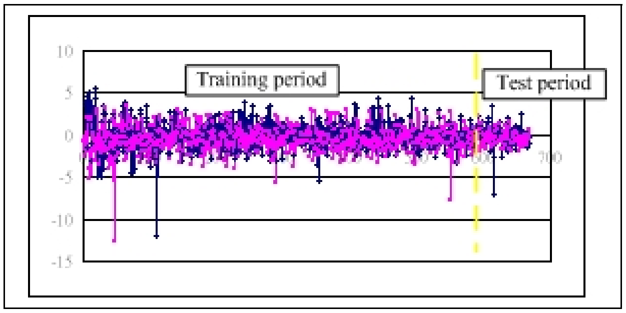

Fig. 6: the prediction error of novel parallel (red line) and one-step ahead prediction (blue line)

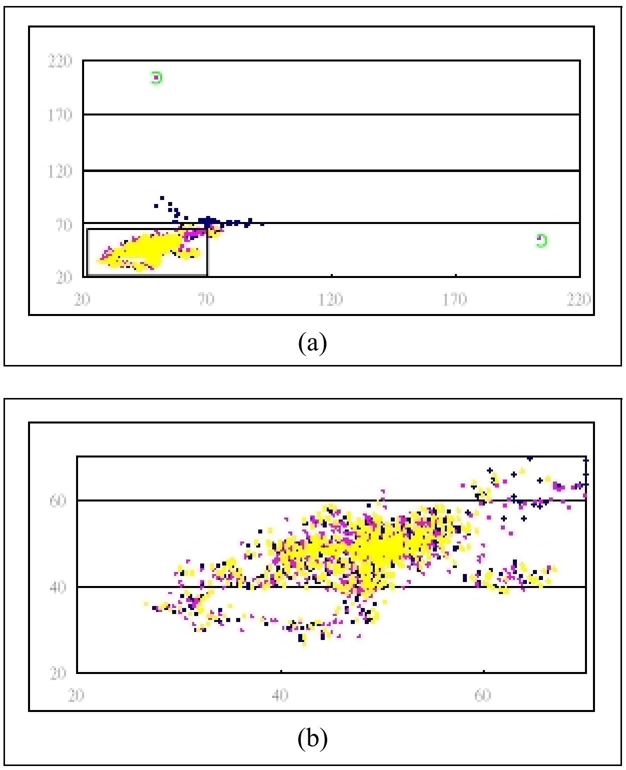

Fig. 7: Lemel's Phase Diagram the red by parallel, yellow by novel and blue by raw data. (a) Whole graph (b) partial graph

Three kinds of predicting time series were used to sketch their fractals as presented in Fig.7a. Some large prediction errors obtained by parallel algorithm (red marker) were observed in the fractal graph. In Fig.7b we found that yellow markers (novel parallel) were more close to blue markers (raw data) than the red marker (parallel algorithm). These results indicate that fractal trajectory was drawn by the novel parallel was very close to the real fractal trajectory. Furthermore, we compared these prediction error time series, and the results are presented in Fig.8. The prediction error of the parallel algorithm (blue marker) was clearly larger than those of other algorithms (the novel parallel yellow maker and one-step ahead red marker). Whether prediction errors were generated by the novel parallel algorithm or by one-step ahead algorithm were hardly separated in Fig. 8. Furthermore, the fractal was used to extract the characteristic of the time series, which implies that time series was a kind of chaotic time series as shown in Fig. 7. Prediction errors of both the novel parallel and one-step ahead are smaller than ±5%, as presented in Fig.8, which indirectly evidence that the stock's price time series belonged to the non-determinate chaos.

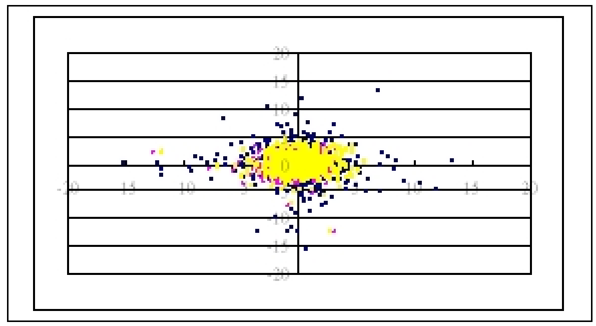

Fig. 8: The prediction error the red by one-step ahead, yellow by novel and blue by parallel.

The motivation of this paper was to decrease the prediction error in multi-step ahead prediction. The prediction error included both system and propagation errors. The parallel algorithm stopped the error propagation, but it generated a new problem: incoherence. The coherent algorithm was added into parallel algorithm to form the novel parallel algorithm. The increase of precision agrees that the novel parallel algorithm not only stopped the error propagation, but also kept the data's coherence. In addition, these merits not only happened in Lemel, but also occurred in other 6 stocks as presented in table 4. It implies that the novel parallel algorithm was a robust algorithm in stock price time series.

Table 4: the prediction error list by parallel and novel

No.	Stock	Parallel	novel	predict precision
2347	Synnex	192.8954	1.7564	99.09%
2324	Compal	467.9330	9.0074	98.08%
2343	SYSTEX	6.2917	0.1941	96.92%
2349	RITEK	10.2297	0.4383	95.72%
2379	RT	92.4137	4.1852	95.47%
2330	TSMC	129.0375	8.8028	93.18%
2396	PRODISC	15.7440	1.1700	92.57%
2313	COMPEQMFG.	12.3145	0.9240	92.50%
2394	PREMIER	21.2871	2.1192	90.04%

4. Conclusions

The prediction error has always occurred in conventional multi-step ahead prediction. The prediction error included both system and propagation errors. How to decrease the propagation error has been the important issue in multi-step ahead prediction. The parallel algorithm has always been used to stop the propagation effect, but it has brought the incoherent problem. Our novel parallel algorithm had to run the sequential algorithm after the parallel algorithm to avoid the incoherent problem. The experimental results evidence that novel parallel algorithm could significantly decrease the prediction error. This indicates that the novel parallel algorithm could overcome the incoherent effect. Finally, we compared the error distributions and found that the radius of the trajectory line was about ±5%, which indicate that the line of the trajectory was the pipe line. Therefore, the ±5% prediction error was the system error, which might not be improved. We evidence that the stock's price time series belonged to the non-determinate chaos.

References

[1] Gautama T., D.P. Mandic, and M.M. Van Hulle. 2003. A differential entropy based method for determining the optimal embedding parameters of a signal. Proceedings of the International Conference on Acoustics, Speech and Signal Processing, 29-32.

[2] Iba H. and N. I. Nikolaev. 2000. Genetic Programming Polynomial Models of Financial Data Series. Proceedings of the 2000 Congress on Evolutionary Computation CEC00, California, USA, IEEE Press, 1459-1466.

[3] Iba H. and T. Sasaki. 1999. Using genetic programming to predict financial data. Proceedings of the Congress on Evolutionary Computation, 1, 244-251.

[4] Kaboudan M., 2005. A measure of time series predictability using genetic programming applied to stock returns. Journal of Forecasting, 18, 345-357.

[5] Kaboudan.M. 1998. A GP approach to distinguish chaotic from noisy signals. Genetic Programming 1998: Proceedings of the Third Annual Conference, San Francisco, California, USA, Morgan Kaufmann Press, 187-192.

[6] Pavlidis N. G., D. K. Tasoulis, and M. N. Vrahatis 2003. Financial Forecasting Through Unsupervised Clustering and Evolutionary Trained Neural Networks, Evolutionary Computation, 2003. CEC '03. IEEE Press, 4, 2314- 2321

[7] Pavlidis N. G., D. K. Tasoulis, and M. N. Vrahatis. 2005. Time Series Forecasting Methodology for Multiple–Step–Ahead Prediction. International Conference on Computational Intelligence. CI 2005, 456-461.

[8] Rembrandt Bakker, J. C. Schouten, C. M. van den Bleek and C. Lee Giles. 1997. Neural Learning of Chaotic Dynamics: The Error Propagation Algorithm. Technical Report UMIACS-TR-97-77 and CS-TR-3843 Institute for Advanced Computer Studies University of Maryland College Park, MD 20742.