

## Research Article

# Crude Oil Price Prediction Based on a Dynamic Correcting Support Vector Regression Machine

Li Shu-rong and Ge Yu-lei

College of Information and Control Engineering, China University of Petroleum, Qingdao, Shandong 266580, China

Correspondence should be addressed to Li Shu-rong; lishuron@upc.edu.cn

Received 10 December 2012; Accepted 28 January 2013

Academic Editor: Fuding Xie

Copyright © 2013 L. Shu-rong and G. Yu-lei. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A new accurate method on predicting crude oil price is presented, which is based on  $\varepsilon$ -support vector regression ( $\varepsilon$ -SVR) machine with dynamic correction factor correcting forecasting errors. We also propose the hybrid RNA genetic algorithm (HRGA) with the position displacement idea of bare bones particle swarm optimization (PSO) changing the mutation operator. The validity of the algorithm is tested by using three benchmark functions. From the comparison of the results obtained by using HRGA and standard RNA genetic algorithm (RGA), respectively, the accuracy of HRGA is much better than that of RGA. In the end, to make the forecasting result more accurate, the HRGA is applied to the optimize parameters of  $\varepsilon$ -SVR. The predicting result is very good. The method proposed in this paper can be easily used to predict crude oil price in our life.

## 1. Introduction

In recent years, crude oil prices have experienced four jumps and two slumps. The fluctuation of crude oil price adds more changes to the development of world economy. Grasping the change of oil price can provide guidance for economic development [1]. Therefore, it is very important to predict the crude oil price accurately.

The predicting methods can be divided into two aspects. One is from the qualitative angle [2]; the other is from quantitative angle, such as econometric model and statistical model [3, 4]. And the latter method is adopted by most scholars. But it is a difficult job to predict crude oil price, since the price is nonlinear and nonstationary time series [5]. The traditional predicting methods such as  $AR(p)$  model,  $MA(q)$  model, and  $ARMA(p,q)$  model, base on linear model. They are only suitable for linear prediction and cannot be applied to model and predict nonlinear time series [6]. Wang got the predicting model by using time series and artificial neural network in 2005 [7], Xie proposed a new method for crude oil price forecasting based on support vector machine (SVM) in 2006 [8], Mohammad proposed a hybrid artificial intelligence model for crude oil price forecasting by means of feed-forward neural networks and genetic algorithm in

2007 [9], and Guo proposed a hybrid time series model on the base of GMTD model in 2010 [10]. The experimental results tell us that the prediction accuracy of these methods is better than traditional models. But the results is still existing biggish errors especially when the crude oil price is fluctuating violently.

Neural network technique provides a favorable tool for nonlinear time series forecasting. But the predictive ability of conventional neural network is low, because of the problems such as the local minimum, over learning, and the lacking of theoretical direction for selecting the hidden layer nodes. The SVM was proposed in the 1990s [11]; it can get the optimal results on the basis of the current information. The basic idea of SVM is that it fits the sample capacity of functions on the basis of regulating the upper bound of the minimum VC dimension, which also means the numbers of support vector. Compared with neural network [12, 13], SVM has strong generalization ability of learning small samples and with the inferior dependence on quantity. But the prediction performance of SVM is very sensible to parameter selection. On the other hand, the research on parameter optimization of SVM is very few at the moment. The parameters are usually determined on experience or trial method. In this way, if the parameters are not suitably chosen, the SVM will lead to poor

prediction performance. So, it is important to find one good method to get the optimal parameters of SVM.

In this paper, an  $\varepsilon$ -support vector regression machine with dynamic correction factor is proposed. And a novel hybrid RNA genetic algorithm (HRGA) is proposed to obtain the optimal parameters for a SVM. The HRGA is from the development of biological science and technology; the structure and information of RNA molecular are known profoundly. To improve the optimal performance of genetic algorithm, one genetic algorithm which bases on coding and biological molecular operation has been widely concerned [14]. This method improves the search efficiency and optimization performance through coding the individuals into biological molecules by use of bases [15, 16]. The appropriate mutation operator can improve the population diversity and prevent premature. While the mutation operator of classical RNA genetic algorithm (RGA) is fixed, so we need to find a suitable method to determine the mutation operator. In 2003, Kennedy did some improvement on particle swarm optimization (PSO) and proposed the bare bones particle swarm algorithm [17].

In the proposed HRGA, the position displacement idea of bare bones PSO is applied to change the mutation operator. The nucleotide base encoding, RNA recoding operation, and protein folding operation are reserved in the new algorithm. Thus, the strong global search capability is kept. At the same time, to make sure of the directivity of local searching, the optimal experience of the whole population and the historical experience of individuals are used. The convergence speed and solution precision are improved. Furthermore, to test the validity of HRGA, three benchmark functions are adopted. The mean value of optimum of HRGA is smaller than that of traditional RNA genetic algorithm.

Once the support vector regression machine is designed optimally, it can be used to predict crude oil price. Dynamic correction factor is brought in to improve the predictive effect and can strengthen the robustness of systems. In order to test the performance of the proposed predicting method, we provided the predicting results by using a back propagation neural network and a traditional support vector regression machine which are also improved with dynamic correction factor [7, 8]. The results show that our predicting method obtains greater accuracy than that of the other two in this paper.

The paper is organized as follows. Section 2 discusses the support vector regression machine with dynamic correction factor. Section 3 presents HRGA based on bare bones PSO, and some testing examples are applied to verify the effectiveness of the algorithm. Section 4 applies the dynamic correcting  $\varepsilon$ -SVR to predict the crude oil price. Section 5 concludes the paper.

## 2. Support Vector Regression Machine with Dynamic Correction Factor

Consider the training sample set  $(x_i, y_i)$ ,  $i = 1, 2, \dots, n$ ,  $x_i \in R^n$ , as the input variable and  $y_i \in R$  as the output variable.

The basic idea of SVM is to find a nonlinear mapping  $\phi$  from input space to output space [18–20]. Data  $x$  is mapped to a high-dimensional characteristic space  $F$  on the basis of the nonlinear mapping. The estimating function of linear regression in characteristic space  $F$  is as follows:

$$\begin{aligned} f(x) &= [\omega \times \phi(x)] + b, \\ \phi: R^n &\longrightarrow F, \omega \in F, \end{aligned} \quad (1)$$

where  $b$  denotes threshold value.

Function approximation problem is equal to the following function:

$$R_{\text{reg}}(f) = R_{\text{emp}}(f) + \lambda \|\omega\|^2 = \sum_{i=1}^s C(e_i) + \lambda \|\omega\|^2, \quad (2)$$

where  $R_{\text{reg}}(f)$  denotes the objective function,  $R_{\text{emp}}(f)$  denotes the empirical risk function,  $s$  denotes the sample quantity,  $\lambda$  denotes adjusting constant, and  $C$  denotes the error penalty factor.  $\|\omega\|^2$  reflects the complexity of  $f$  in the high-dimensional characteristic space.

Since linear  $\varepsilon$  insensitive loss function has better sparsity, we can get the following loss function:

$$|y - f(x)|_{\varepsilon} = \max\{0, |y - f(x) - \varepsilon|\}. \quad (3)$$

The empirical risk function is as follows:

$$R_{\text{emp}}^{\varepsilon}(f) = \frac{1}{n} \sum_{i=1}^n |y - f(x)|_{\varepsilon}. \quad (4)$$

According to the statistical theory, we bring in two groups of nonnegative slack variable  $\{\xi_i\}_{i=1}^n$  and  $\{\xi_i^*\}_{i=1}^n$ . Then, the question can be converted to the following nonlinear  $\varepsilon$ -support vector regression machine ( $\varepsilon$ -SVR) problem:

$$\begin{aligned} \min_{(\xi_i, \xi_i^*)} & \left\{ \frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^n (\xi_i + \xi_i^*) \right\}, \\ y_i - \omega \cdot \phi(x) - b & \leq \varepsilon + \xi_i^*, \\ \omega \cdot \phi(x) + b - y_i & \leq \varepsilon + \xi_i, \\ \xi_i, \xi_i^* & \geq 0, \end{aligned} \quad (5)$$

where  $\varepsilon$  denotes the insensitive loss function.  $C$  is used to balance the complex item and the training error of the model.

We bring into Lagrange multipliers  $\alpha_i$  and  $\alpha_i^*$ , then the convex quadratic programming problem above can be changed into the below dual problem:

$$\begin{aligned} \max_{(\alpha_i, \alpha_i^*)} & \left[ -\frac{1}{2} \sum_{i,j=1}^n (\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j) K(X_i, X_j) \right. \\ & \left. + \sum_{i=1}^n \alpha_i^* (y_i - \varepsilon) - \sum_{i=1}^n \alpha_i (y_i - \varepsilon) \right], \\ \sum_{i=1}^n (\alpha_i - \alpha_i^*) & = 0, \\ 0 \leq \alpha_i, \alpha_i^* & \leq C, \quad i = 1, 2, \dots, n, \end{aligned} \quad (6)$$

where  $K(X_i, X_j)$  denotes the inner product kernel satisfying Mercer theorem.

We can get the  $\varepsilon$ -SVR function through solving the above dual problem:

$$f(x) = \sum_{i=1}^n (\alpha_i - \alpha_i^*) K(X_i, X) + b. \quad (7)$$

When  $\varepsilon$ -SVR is used on prediction, it may have a certain error since the data fluctuates violently such as the crude oil price. To reduce the error in some certain as possible as we can, we bring in the dynamic correction factor  $\varepsilon^*$ . The main idea of the dynamic correction factor is that we use the error of back step with multiplying  $\varepsilon^*$  to revise the current predicting results. Thus, we can reduce the current predicting error. The dynamic correcting SVR can be defined as follows:

$$Y_d(i+1) = Y_1(i+1) + \varepsilon^* [Y(i) - Y_d(i)], \quad (8)$$

where  $Y$  denotes the real results,  $Y_d$  denotes the final prediction results,  $Y_1$  denotes the initial predicting results,  $\varepsilon^*$  denotes the dynamic correction factor, and  $i$  denotes the prediction steps.

In order to make the predicting results more accurate, the optimal value of  $\varepsilon^*$  and the parameters of  $\varepsilon$ -SVR involving  $C, \delta$  (the variable in gauss kernel function) should be designed (in (8)). To this end, an HRGA is studied below to optimize the following problem:

$$\min_{(\varepsilon^*, C, \delta)} \sum_i^n [Y_d(i) - Y(i)]^2. \quad (9)$$

### 3. HRGA Based on Bare Bones PSO

Assuming that population size is  $N$ , the dimension of particle is  $m$ . The position of particle  $i$  on generation  $t$  is  $X_i(t) = (x_{i1}(t), \dots, x_{ij}(t), \dots, x_{im}(t)), i = 1, 2, \dots, N$ . The speed of particle  $i$  on generation  $t$  is  $V_i(t) = (v_{i1}(t), \dots, v_{ij}(t), \dots, v_{im}(t))$ . The historic optimal value of individuals is  $PBest_i(t) = (pbest_{i1}(t), \dots, pbest_{ij}(t), \dots, pbest_{im}(t))$ .

Let the global optimal value be  $GBest(t) = (gbest_1(t), \dots, gbest_j(t), \dots, gbest_m(t))$ .

As to standard particle swarm, the position and speed are updated as

$$\begin{aligned} v_{ij}(t+1) &= \omega v_{ij}(t) + c_1 \cdot r_{1j} \cdot (pbest_{ij}(t) - x_{ij}(t)) \\ &\quad + c_2 \cdot r_{2j} \cdot (gbest_j - x_{ij}(t)), \\ x_{ij}(t+1) &= x_{ij}(t) + v_{ij}(t+1), \end{aligned} \quad (10)$$

where  $\omega$  denotes the inertia weight [21],  $c_1$  and  $c_2$  denote the accelerating operators, and  $r_{1j}$  and  $r_{2j}$  are uniform distributed random numbers in  $[0, 1]$ .

In the bare bones particle swarm optimization (PSO), (10) is replaced by (11) as the evolution equation of particle swarm algorithm:

$$\begin{aligned} x_{ij}(t+1) &= N \left( 0.5 (pbest_{ij}(t) + gbest_j(t)) \right. \\ &\quad \left. \times |pbest_{ij}(t) - gbest_j(t)| \right). \end{aligned} \quad (11)$$

The position of particle is some random numbers which are gotten from the Gauss distribution. The distribution has the mean value of  $(pbest_{ij}(t) + gbest_j(t))/2$  and the standard deviation of  $|pbest_{ij}(t) - gbest_j(t)|$ .

RNA genetic algorithm is on the basis of base coding and biological molecules operation. Since in the biological molecule, every three bases compose one amino acid. In other words, the bases' length of individuals must be divided exactly by 3. When RNA recoding and protein folding [22], to reduce calculation and to control population size, we assume that the protein folding operation only occurs on the individuals without RNA recoding. Then the most important work is to change the mutation probability [23, 24].

Angeline told us that the essence of particle swarm's position updating was one mutation operation in 1998 [25]. Traditional RNA genetic algorithm mutates as the fixed mutation probability, and the mutation is random with one direction. However HRGA can reflect the historic information of individuals and the sharing information of the population. HRGA can make every individual do directional mutation and improve search efficiency. Moreover, HRGA ensures the strong global search capability, since it does not change the selection and crossover operator.

The procedure of HRGA based on bare bones particle swarm algorithm to optimize the  $\varepsilon$ -SVR parameters and the dynamic correction factor is as follows.

*Step 1.* Get one group of  $\varepsilon$ -SVR parameters, and the dynamic correction factor randomly, code every parameter, and get the initial RNA population with  $N$  individuals, crossover probability  $P_c$ , and mutation probability  $P_m$ . Assign values for every  $PBest_i$  (individual's historic optimal solution) and  $GBest$  (population's global optimal solution).

*Step 2.* Compute its error function and get the fitness function. Comparing it with corresponding fitness value of  $PBest_i$  and  $GBest$ , then update  $PBest_i$  and  $GBest$ .

*Step 3.* Execute the selection operation. Get current generation through coping  $N$  individuals from the initial or the last generation.

*Step 4.* Decide whether the value meets the RNA recoding condition or not. If  $Y$ , recode RNA, then go to Step 6. If  $N$ , go to Step 5.

*Step 5.* Decide meet the protein mutual folding condition or not. If  $Y$ , execute the protein mutual folding operation. If  $N$ , execute the protein own folding operation.

*Step 6.* Execute the mutation operation as (11) for all the crossover individuals, on the basis of the  $PBest_i$  and  $GBest$ , which have been gotten from Step 2.

*Step 7.* Repeat Step 2 to Step 6 until the training target meets the condition. At last, we get the optimal parameters of  $\varepsilon$ -SVR and the dynamic correction factor.

TABLE 1: Benchmark functions.

Function	Formula	Global minimum
Sphere	$f_1(x) = \sum_{i=1}^n x_i^2, x_i \in [-100, 100].$	$f_1^*(x) = 0, x^* = (0, 0, \dots, 0)$
Rosenbrock	$f_2 = \sum_{i=1}^{n-1} (100 \times (x_{i+1} - x_i^2)^2) + (x_i - 1)^2, x_i \in [-30, 30].$	$f_2^*(x) = 0, x^* = (1, 1, \dots, 1)$
Griewank	$f_3 = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1, x_i \in [-600, 600].$	$f_3^*(x) = 0, x^* = (0, 0, \dots, 0)$

TABLE 2: Testing results of HRGA and standard RGA.

Function	Hunting zone	Dimension	Iterative times	RGA		HRGA	
				Parameter selection	Mean value of optimum	Parameter selection	Mean value of optimum
Sphere	(-100, 100)	2	250	$P_c = 0.9$ $P_m = 0.1$	$8.3421e - 6$	$P_c = 0.85$ $P_m = 0.3$	$1.9824e - 125$
	(-100, 100)	10	1000	$P_c = 0.9$ $P_m = 0.1$	0.0437	$P_c = 0.85$ $P_m = 0.3$	$1.4597e - 67$
	(-100, 100)	20	1500	$P_c = 0.9$ $P_m = 0.1$	0.1012	$P_c = 0.85$ $P_m = 0.3$	$1.5775e - 31$
Rosenbrock	(-30, 30)	2	250	$P_c = 0.85$ $P_m = 0.1$	0.5311	$P_c = 0.6$ $P_m = 0.3$	0.0195
	(-30, 30)	10	1000	$P_c = 0.85$ $P_m = 0.1$	15.0766	$P_c = 0.6$ $P_m = 0.3$	3.0922
	(-30, 30)	20	1500	$P_c = 0.85$ $P_m = 0.1$	124.2468	$P_c = 0.6$ $P_m = 0.3$	8.0576
Griewank	(-600, 600)	2	250	$P_c = 0.8$ $P_m = 0.15$	0.0062	$P_c = 0.5$ $P_m = 0.4$	0.0044
	(-600, 600)	10	1000	$P_c = 0.8$ $P_m = 0.15$	0.1468	$P_c = 0.5$ $P_m = 0.4$	0.0132
	(-600, 600)	20	1500	$P_c = 0.8$ $P_m = 0.15$	0.0191	$P_c = 0.5$ $P_m = 0.4$	$2.0109e - 3$

The flowchart of HRGA to optimize the  $\varepsilon$ -SVR parameters and the dynamic correction factor is shown in Figure 1.

**3.1. HRGA Testing.** Three classical benchmark functions shown in Table 1 are used to test the property of HRGA.

In addition, among the three functions, Sphere is unimodal function, and the other two are multimodal function.

With the population size  $N = 50$ , and other parameters determined by multiple test for each function. Each function is tested by HRGA and standard RGA in different dimensions. Each experience is carried on 100 times. Record the mean value of target function's optimum (shown in (12)). The result is displayed in Table 2:

$$\text{MVO} = \frac{1}{N} \sum_{i=1}^N f_i'(x). \quad (12)$$

In this equation, MVO denotes the mean value of target function's optimum;  $f_i'(x)$  denotes the optimum of benchmark functions in every experiment.

As to the experimental results, with different dimensions having the same iterative times, the mean value of optimum of HRGA is smaller than that of RGA for the three benchmark functions. The average performance of HRGA is closer to

the optimum. We can increase the mutation probability appropriately and enhance the convergence speed, since the mutation operator of HRGA has directional local search.

#### 4. Crude Oil Price Prediction Based on a Dynamic Correcting $\varepsilon$ -SVR

In this paper, we get the crude oil price from the US Energy Information Administration Web [26]. Since the oil price fluctuates violently, in order to facilitate the processing and decrease the error, we adopt the Cushing, OK WTI Spot Price FOB (dollars per barrel) monthly from January 1986 to now. We take the one hundred data from January 1986 to April 1994 as the test sample. And give the next 20-month dynamic predicting data from May 1994 to December 1995. The relative error of forecasting is shown in Table 2. The prediction effect figure of HRGA and  $\varepsilon$ -SVR with dynamic correction factor is shown in Figure 2. We use Gauss function as the kernel function of  $\varepsilon$ -SVR, which is given as follows:

$$k(x, y) = \exp\left(-\frac{\|x - y\|^2}{2 \cdot \sigma^2}\right). \quad (13)$$

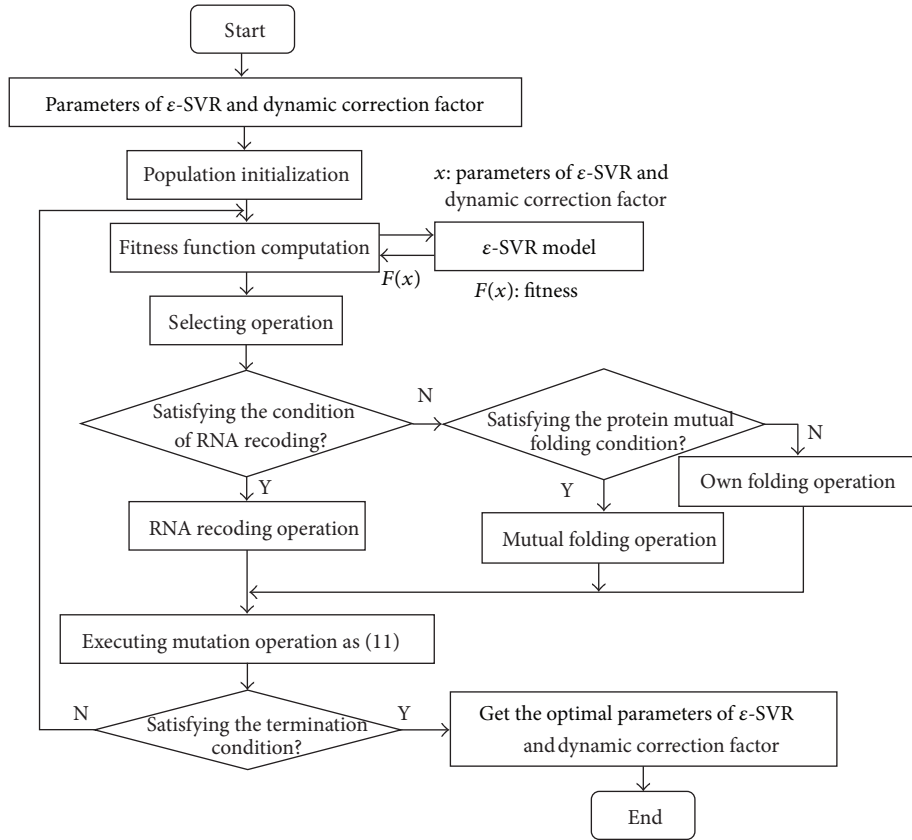


FIGURE 1: The flowchart of HRGA.

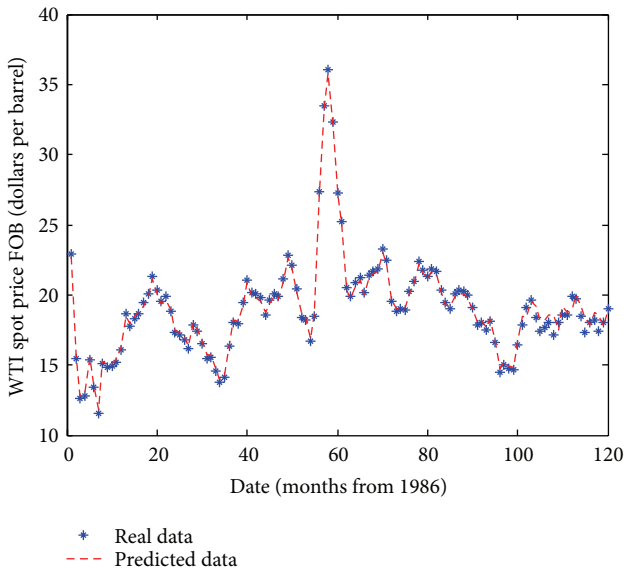


FIGURE 2: The prediction curve of HRGA- $\epsilon$ -SVR.

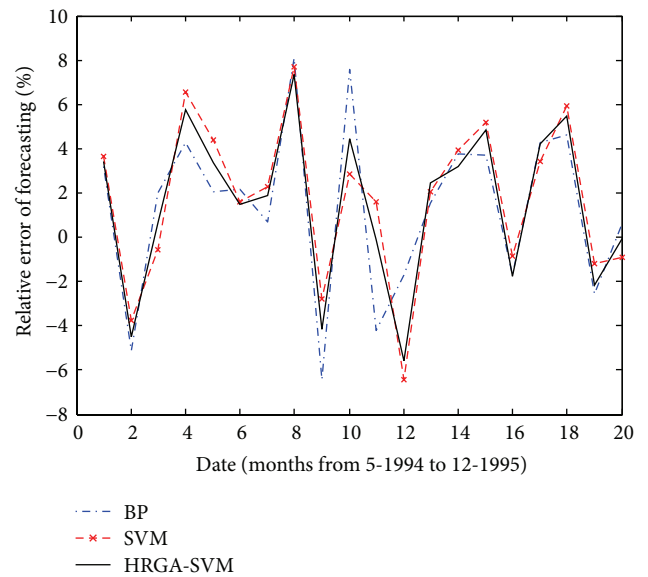


FIGURE 3: Errors analysis with different models.

Parameter setting of HRGA- $\epsilon$ -SVR is with population size being 100, maximum evolution generation being 150, coding length of  $C$  being 9, coding length of  $\epsilon$  being 8, coding length of  $\sigma$  being 13, coding length of  $\epsilon^*$  being 8,  $P_c$  being 0.8, and  $P_m$  being 0.1.

The optimization interval is set to be

$$\begin{aligned}
 1 \leq C \leq 10000, & \quad 0.0001 \leq \epsilon \leq 0.1, \\
 0.01 \leq \sigma \leq 500, & \quad 0.5 \leq \epsilon^* \leq 2.
 \end{aligned} \tag{14}$$

TABLE 3: Analysis results of forecasting error.

Date	BP/%	$\varepsilon$ -SVR%	HRGA- $\varepsilon$ -SVR /%
5-1994	3.22	3.64	3.42
6-1994	-5.1	-3.78	-4.48
7-1994	1.97	-0.57	0.61
8-1994	4.27	6.55	5.77
9-1994	2.05	4.41	3.38
10-1994	2.19	1.59	1.51
11-1994	0.69	2.3	1.87
12-1994	8.08	7.68	7.35
1-1995	-6.46	-2.79	-4.16
2-1995	7.58	2.87	4.44
3-1995	-4.23	1.61	-0.13
4-1995	-1.69	-6.46	-5.57
5-1995	1.56	2.08	2.48
6-1995	3.75	3.94	3.2
7-1995	3.7	5.19	4.83
8-1995	-1.69	-0.85	-1.75
9-1995	4.27	3.4	4.19
10-1995	4.64	5.94	5.45
11-1995	-2.58	-1.17	-2.14
12-1995	0.62	-0.92	-0.03
$\delta$	4.09	3.96	3.87

When analyzing the results, we define the evaluation index:

$$E_r = \frac{x_i - y_i}{x_i} \times 100\%, \quad \sigma = \sqrt{\frac{1}{n} \sum_{i=1}^n \left( \frac{x_i - y_i}{x_i} \right)^2}. \quad (15)$$

The forecasting error analysis results are shown in Figure 3. In this figure, SVM refers to  $\varepsilon$ -SVR. The BP neural network and  $\varepsilon$ -SVR are with dynamic correction factor which differs them to the traditional method. From Figure 2, we can know that the prediction result is very close to the real value. The HRGA- $\varepsilon$ -SVR can be used to predict the crude oil price. Table 3 tells us the WTI crude oil price predicting relative errors of twenty months. Among the three methods in twenty months, the biggest absolute value of relative error of HRGA- $\varepsilon$ -SVR is the smallest, which is 7.35%, and the smallest root-mean-square of relative error is 3.87%. As to Figure 3, the fluctuation range of HRGA- $\varepsilon$ -SVR is smaller than those of the other two methods obviously. This means that HRGA- $\varepsilon$ -SVR is the best one among the three methods.

## 5. Conclusions

In this paper, we have presented a novel method on predicting crude oil price. This method bases on an  $\varepsilon$ -support vector regression machine with dynamic correction factor correcting predicting errors. We also proposed the HRGA, with the position displacement idea of bare bones PSO changing the mutation operator, to optimize the parameters in an  $\varepsilon$ -SVR. The predicting result of crude oil price shows the validity of

the proposed method. Thus, the  $\varepsilon$ -SVR model can also be applied to predict tendency in other practical areas.

## Acknowledgments

The research was partially supported by Grant no. 60974039 from the National Science Foundation of China and by Grant no. ZR2011FM002 from the Natural Science Foundation of Shandong Province.

## References

- [1] B. Hunt, P. Isard, and D. Laxton, "The macroeconomic effects of higher oil prices," IMF Working Paper No.wp/01/14, 2001.
- [2] Y. Fan, K. Wang, Y. J. Zhang et al., "International crude oil market analysis and price forecast in 2009," *Bulletin of Chinese Academy of Sciences*, vol. 4, no. 1, pp. 42–45, 2009.
- [3] C. Morana, "A semiparametric approach to short-term oil price forecasting," *Energy Economics*, vol. 23, no. 3, pp. 325–338, 2001.
- [4] S. Mirmirani and H. Cheng Li, "A comparison of VAR and neural networks with genetic algorithm in forecasting price of oil," *Advances in Econometrics*, vol. 19, pp. 203–223, 2004.
- [5] Z. J. Ding, Q. Min, and Y. Lin, "Application of ARIMA model in forecasting prude oil price," *Logistics Technology*, vol. 27, no. 10, pp. 156–159, 2008.
- [6] J. P. Liu, S. Lin, T. Guo, and H. Y. Chen, "Nonlinear time series forecasting model and its application for oil price forecasting," *Journal of Management Science*, vol. 24, no. 6, pp. 104–112, 2011.
- [7] S. Y. Wang, L. Yu, and K. K. Lai, "Crude oil price forecasting with TEI@ I methodology," *Journal of Systems Sciences and Complexity*, vol. 18, no. 2, pp. 145–166, 2005.
- [8] W. Xie, L. Yu, S. Xu, and S. Wang, "A new method for crude oil price forecasting based on support vector machines," *Lecture Notes in Computer Science*, vol. 3994, pp. 444–451, 2006.
- [9] R. A. N. Mohammad and A. G. Ehsan, "A hybrid artificial intelligence approach to monthly forecasting of crude oil price time series," in *The Proceedings of the 10th International Conference on Engineering Applications of Neural Networks*, pp. 160–167, 2007.
- [10] S. Guo and P. Lai, "The time series mixed model and its application in price prediction of international crude oil," *Journal of Nanjing University of Information Science & Technology*, vol. 2, no. 3, pp. 280–283, 2010.
- [11] Y. B. Hou, J. Y. Du, and M. Wang, *Neural Networks*, Xidian University Press, Xi'an, China, 2007.
- [12] H. Zhu, L. Qu, and H. Zhang, "Face detection based on wavelet transform and support vector machine," *Journal of Xi'an Jiaotong University*, vol. 36, no. 9, pp. 947–950, 2002.
- [13] R. Feng, C. L. Song, Y. Z. Zhang, and H. H. Shao, "Comparative study of soft sensor models based on support vector machines and RBF neural networks," *Journal of Shanghai Jiaotong University*, vol. 37, pp. 122–125, 2003.
- [14] J. Tao and N. Wang, "DNA computing based RNA genetic algorithm with applications in parameter estimation of chemical engineering processes," *Computers & Chemical Engineering*, vol. 31, no. 12, pp. 1602–1618, 2007.
- [15] K. Wang and N. Wang, "A protein inspired RNA genetic algorithm for parameter estimation in hydrocracking of heavy oil," *Chemical Engineering Journal*, vol. 167, no. 1, pp. 228–239, 2011.

- [16] K. Wang and N. Wang, "A novel RNA genetic algorithm for parameter estimation of dynamic systems," *Chemical Engineering Research & Design*, vol. 88, no. 11, pp. 1485–1493, 2010.
- [17] D. Bratton and J. Kennedy, "Defining a standard for particle swarm optimization," in *Proceedings of the IEEE Swarm Intelligence Symposium (SIS '07)*, pp. 120–127, April 2007.
- [18] N. Y. Deng and Y. J. Tian, *A New Method of Data Mining and Germany: Support Vector Machines*, Science Press, Beijing, China, 2004.
- [19] U. Thissen, R. Van Brakel, A. P. De Weijer, W. J. Melssen, and L. M. C. Buydens, "Using support vector machines for time series prediction," *Chemometrics and Intelligent Laboratory Systems*, vol. 69, no. 1-2, pp. 35–49, 2003.
- [20] K. J. Kim, "Financial time series forecasting using support vector machines," *Neurocomputing*, vol. 55, no. 1-2, pp. 307–319, 2003.
- [21] Y. Shi and R. Eberhart, "Modified particle swarm optimizer," in *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 519–523, 1998.
- [22] D. P. Clark, *Molecular Biology: Understanding the Genetic Revolution*, Academic Press, New York, NY, USA, 2005.
- [23] J. Lis, "Genetic algorithm with the dynamic probability of mutation in the classification problem," *Pattern Recognition Letters*, vol. 16, no. 12, pp. 1311–1320, 1995.
- [24] M. Serpell and J. E. Smith, "Self-adaptation of mutation operator and probability for permutation representations in genetic algorithms," *Evolutionary Computation*, vol. 18, no. 3, pp. 491–514, 2010.
- [25] P. J. Angeline, "Evolutionary optimization versus PSO: philosophy and performance differences," *Evolutionary Programming*, vol. 7, pp. 601–610, 1998.
- [26] <http://www.eia.gov/dnav/pet/hist/LeafHandler.ashx?n=PET&s=RWTC&f=M>.



# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

