2011 American Control Conference
on O'Farrell Street, San Francisco, CA, USA
June 29 - July 01, 2011

An Adaptive Chaotic PSO for Parameter Optimization and Feature Extraction of LS-SVM Based Modelling

Weijian Cheng, Jinliang Ding, Weijian Kong, Tianyou Chai, and S.Joe Qin

Abstract—**While training an LS-SVM model, two main challenges are parameter optimization and input feature extraction. The main purpose of this article is to address these two problems. Commonly used tools are PSO and BPSO, but they are not suitable for the optimization issues of many local optima owing to its random numbers used to update velocities. In this paper, an adaptive chaotic particle swarm optimization (cPSO) algorithm is proposed to enhance its global searching capability and local searching capability. The practicality of the proposed scheme is demonstrated by application to mineral process for the prediction models between production rate of the concentrated ore and the technical indexes. Compared with the original methods of grid search+PCA, GA+PCA, PSO+PCA as well as PSO+BPSO, the proposed strategy outperforms these existing methods in terms of convergence accuracy.**

I. Introduction

SUPPORT vector machine (SVM) [1] is a novel method for solving problems in nonlinear classification and regression. Unlike most of the traditional methods which implement the empirical risk minimization principle, it is introduced in the context of statistical learning theory and structural risk minimization, eventually resulting in better generalization performance. LS-SVM [2] is reformulations to standard SVM, which leads to solving linear KKT systems, and it is also proposed to address a variety of classification and regression problems [3], [4].

The main challenge of LS-SVM lies in continuous parameter optimization. In this procedure, kernel parameter σ^2 and penalty coefficient γ are two critical parameters for model selection, and have a significant influence on the performance of the regression model obtained in the final. Therefore, these two parameters' tuning attracts much attention. Typical continuous optimization method adopted is a grid search algorithm [5], but the complete grid search unavoidably brings a high computational burden, frequently excluding their application to large-scaled problems. In addition, heuristic strategies such as GA [6], PSO [7] and cPSO [8] are also introduced to optimize parameters of LS-SVM, however, GA and PSO easily trap into local optima, meanwhile, stochastic nature-inspired search algorithms can not assure good results in the different runs. To tackle these problems, we propose a bi-population approach in our former research [8], in which one population searches in the manner of PSO with excellent convergence ability, while the other performs adaptive lattice search with outstanding global exploration ability to ensure a good balance between exploration and exploitation, which doubles the optimization process, thus it is time consuming. In this paper, we utilize adaptive lattice search to create lattice particles to replace the worst particles in PSO only when the fitness value of *gbest* (global optima) stays constant for a certain period, which reduces the computational load without ruining the optimal solutions.

Another difficulty of modelling based on LS-SVM is input feature extraction. Due to the coupling among input variables and computational problems high-dimensional input space causes, some linear methods such as PCA [9], PLS [10], ICA [11] etc., are proposed to address these issues. However, input space with nonlinear coupling limits their application to this issue. Thus, nonlinear methods, for example, KPCA [12], KPLS [13] based on kernel functions are also adopted. The original input space is firstly projected into high dimensional subspace, and then extract feature space with linear PCA or PLS, but these methods are criticized for their computational inefficiency. In addition, the discrete binary particle swarm optimization (BPSO) [14] algorithm has lately gained much attention for solving discrete feature extraction. In [15] the authors apply BPSO to find optimal feature space, through experiment they compare BPSO with GA as well as SA and conclude that BPSO is more efficient for feature selection. However, it is known that BPSO easily traps into local optima. Therefore, we introduce a novel BPSO with chaotic operator to enhance its searching capability.

Aiming at solving problems of continuous parameter optimization and discrete input feature extraction of LS-SVM, we propose two novel strategies to modify standard PSO. Firstly, chaotic operators generated by chaotic map are introduced to

Manuscript received September 22, 2010. This work was supported in part by the National Nature Science Foundation of China under Grant 61020106003, 60821063 and 60904079, by the National Basic Research Program of China under Grant 2009CB320601,and the 111 project (B08015)

W.J.Cheng, J.L.Ding, W.J.Kong are with State Key Laboratory of Integrated Automation for Process Industry (Northeastern University), China. (e-mail: chengweijian1986@gmail.com, jlding@mail. neu. edu.cn, weijian. kong9@gmail.com).

T.Y. Chai is with State Key Laboratory of Integrated Automation for Process Industry (Northeastern University) and Research Center of Automation, China. (e-mail: tychai@mail.neu.edu.cn).

S. J. Qin is with the Mork Family Department of Chemical Engineering and Materials Science, the Ming Hsieh Department of Electrical Engineering, and the Daniel J. Epstein Department of Industrial and Systems Engineering, University of Southern California, Los Angeles, CA 90089-1211 USA. (e-mail: sqin@usc.edu)

replace random numbers. These operators can iteratively generate ergodic, non-repeated and pseudorandom solutions, which guarantee global search capability and convergence. Besides, we put forward a novel adaptive search strategy, in which we optimize continuous parameters in a manner of Clerc's constriction PSO [16] unless the *gbest* fitness lessens in certain iterations. Otherwise, an adaptive lattice search will be introduced to create lattice particles to substitute the worst particles in PSO, which enables us to obtain more accurate local solutions.

The remainder of this paper is organized as follows. Section II describes LS-SVM algorithm in brief. Section III introduces the proposed adaptive cPSO and its' application to LS-SVM model selection. Section IV describes the mineral processing and compares the experimental results with grid search+PCA, PSO+PCA, GA+PCA, PSO+BPSO in the mineral process. Conclusions are finally given in Section V.

II. LEAST SQUARE SUPPORT VECTOR MACHINE

The formulation of LS-SVM is introduced as follows. It is necessary to minimize a cost function containing a penalized regression error as follows:

$$\min J(w,\zeta)=\frac{1}{2}\|w\|^2+\frac{1}{2}\gamma\sum_{j=1}^{l}\zeta_j^2$$
$$s.t.\ y_j=\langle w,\phi(x_j)\rangle+b+\zeta_j \quad j=1,\ldots,l \tag{1}$$

where, γ and ζ_j represent the relative weight and regression error respectively.

To solve this optimization problem, Lagrange function is constructed as follows.

$$L(w,\zeta,b,\alpha)=\frac{1}{2}\|w\|^2+\frac{1}{2}\gamma\sum_{j=1}^{l}\zeta_j^2 -\sum_{j=1}^{l}\alpha_j\left[\langle w,\phi(x_j)\rangle+b+\zeta_j-y_j\right] \tag{2}$$

where, α_j is Lagrange multiplier. The solution of (1) can be obtained by Karush-Kuhn-Tucker (KKT) with respect to ω, b, ζ_j and α_j. Eliminate ω and ζ_j, and then get linear equality:

$$\begin{bmatrix}0 & H^T\\ 1 & \Omega+I/\gamma\end{bmatrix}\begin{bmatrix}b\\ \alpha\end{bmatrix}=\begin{bmatrix}0\\ y\end{bmatrix} \tag{3}$$

where, $y=[y_1,y_2,\ldots,y_l]$, $H=[1,1,\ldots,1]$, $\alpha=[\alpha_1,\alpha_2,\ldots,\alpha_l]$.

Apply Mercer's condition:

$$\Omega_{jk}=\langle\phi(x_k)^T,\phi(x_j)\rangle=K(x_k,x_j)\quad j,k=1,\ldots,l \tag{4}$$

In this study, we focus on RBF kernel, its equation is demonstrated as follow:

$$K(x_j,x_k)=\exp\left(-\frac{\|x_j-x_k\|^2}{2\delta^2}\right) \tag{5}$$

which leads to the following nonlinear regression function:

$$f(x)=\sum_{j=1}^{l}\alpha_jK(x,x_j)+b \tag{6}$$

To achieve a good generalization and accurate model, it should be stressed to do a careful model selection, i.e. tuning σ^2 and γ. In addition, the feature extraction problem should also be taken into account. The algorithm proposed to address these two problems is demonstrated in the next section.

III. ADAPTIVE CHAOTIC PSO (ADAPTIVE CPSO) AND ITS' APPLICATION TO LS-SVM MODEL

A. Adaptive cPSO Algorithm

1) Standard PSO and discrete BPSO

Suppose that the searching space is n-dimensional and continuous. The number of particles is *Num*, the *ith* particle represents an n-dimensional vector X_i. It means that the *ith* particle locates at $X_i=(x_{i1},x_{ij,\ldots},x_{in})$ in the search space. The position of each particle is a potential solution. The particles' velocity and position are updated by the following equations:

$$v_{ij}(t+1)=wV*v_{ij}(t)+c_1rand_{1j}(t)(pbest_{ij}(t)-x_{ij}(t)) +c_2rand_{2j}(t)(gbest_j(t)-x_{ij}(t)) \tag{7}$$

$$x_{ij}(t+1)=x_{ij}(t)+v_{ij}(t+1) \tag{8}$$

where, v_{ij}, x_{ij} represent velocity and position of *ith* particle, $pbest_i$ is the local best position of *ith* particle, *gbest* is the global best solution. c_1 and c_2 are acceleration coefficients, $rand_{1j}$ and $rand_{2j}$ are two random numbers between 0 and 1.

Binary particle swarm optimization (BPSO) [14] is proposed in 1997 by Kennedy and Eberhart. It could be effectively utilized to discrete binary sequence optimization problems. In the BPSO technique, the probability of the particle being as 0 or 1 is specified by the velocity value using sigmoid function. This determination of the position is performed using the following formula:

$$x_{ij}(t+1)=\begin{cases}1 & rand>f(v_{ij}(t+1))\\ 0 & rand\le f(v_{ij}(t+1))\end{cases} \tag{9}$$

where, *rand* is the random numbers uniformly distributed between 0 and 1, $f(\cdot)$ is sigmoid function and it is given as follows.

$$f(v_{ij}(t+1))=\frac{1}{1+e^{-v_{ij}(t+1)}} \tag{10}$$

2) Chaotic map

Chaos is a deterministic dynamic system and is very sensitive dependence on its initial conditions and parameters. Though the nature of chaos is apparently random and unpredictable, but it is characterized by ergodicity and regularity, and a chaotic map can iteratively generate ergodic, non-repeated, and deterministic solutions [18]. Based on special property of regularity, a more efficient pseudorandom search algorithm can be designed to replace random numbers in every run of standard PSO [19].

Logistic map and tent map are the most frequently used chaotic behavior, but both have the disadvantage of trends to assemble to some range in higher probability after certain iteration. Hence, we introduce a novel chaotic map model utilizing the tent map perturbed by the logistic map [8]. The slight disturbance of the logistic map can eliminate the fixed points of the tent map and almost do not change the amplitude of the tent map. The new chaotic map model is formulated as follows.

$$\begin{cases} u_{k+1} = 4 * u_k (1 - u_k) & 0 \le u_k \le 1 \\ v_{k+1} = \begin{cases} 1/1.001 * (2 * v_k + 0.001 * u_{k+1}) & 0 \le v_k \le 0.5 \\ 1/1.001 * (2 * (1 - v_k) + 0.001 * u_{k+1}) & 0.5 < v_k \le 1 \end{cases} \end{cases} \quad (11)$$

In this study, we set initial point u_0 and x_0 to 0.1, and the designed chaotic operator generates 30000 iterative points, in which the most appearing times of a point are 328, the least appearing times of a point are 254, and the average appearing times are 300, so the constructed chaotic map is good at distribution. Therefore, a chaotic operator *chaotic_operator (k)* = v_k is designed, which can get completely different track while slightly tuning the initial value of u_0 and x_0.

3) adaptive cPSO

The major drawback of standard PSO lies in its premature convergence, especially while handling problems with many local optima, in this situation, the solution is mot available for some issues for the sake of accurate global solution. In this paper, chaotic operators generated from proposed chaotic map substitute random numbers in standard PSO. By this way, it is intended to improve the global convergence and to prevent to trap into local optima. Besides, we also propose adaptive lattice search to enhance its accuracy of local solution. Both contribute to a more accurate global solution. The details are explained as follows.

Based on the standard PSO, a novel chaotic operator is introduced with the expectation of keeping the local diversity, as well as enhancing the reliability of the algorithm. The velocity of each particle is updated by the following equation:

$$\begin{aligned} v_{ij}(t+1) = wV * v_{ij}(t) + c_1 * chaotic\_operator_1(t) * \ldots \\ (pbest_{ij}(t) - v_{ij}(t)) + c_2 * \ldots \quad (12) \\ chaotic\_operator_2(t) * (gbest_j(t) - v_{ij}(t) \end{aligned}$$

where, *chaotic_operator* is an iterative value as chaotic mapping.

In addition, a novel PSO embedded with lattice search is introduced to keep a good balance between exploration and exploitation. If the *gbest* fitness lessens in certain iterations, we search continuous parameters in a manner of PSO, otherwise, adaptive lattice search will be introduced to fulfill another subspace search in the neighborhood of *gbest*. This algorithm creates lattice particles in place of the worst PSO particles.

The adaptive lattice search is described as following. Provide an n-dimensional decision space Ω^n and the *ith* dimension variable range is $[d_{imin}, d_{imax}]$. Initially partition each dimension of the decision space to p_i lattices equally, so the range of initial lattice is $[d_{imin} + \delta_i * (j\text{-}1),\ d_{imin} + \delta_i * j]$, and its width is $\delta_i = (d_{imax} - d_{imin})/p_i$. Hence, there will be $n_d = \prod p_i$ lattices in the decision space Ω^n. According to *gbest* in last generation, we locate the position of *gbest* in a range of a initial lattice $[d_{imin} + \delta_i * (aim\text{-}1),\ d_{imin} + \delta_i * aim]$, $aim = 1, \ldots, p_i$, and then tune the variable range to $[d_{imin} + \delta_i * (aim\ \text{-}1),\ d_{imin} + \delta_i * aim]$ to search the global optima in a more accurate range. Update the position of each particle according to the following equation:

$$\begin{aligned} x_{ij}(k+1) = chaotic\_operator * \delta_i^{'} + \delta_i^{'} * (j-1) \\ + d_{i\min}(k) \qquad k = 0, 1, \ldots, \max gen - 1 \end{aligned} \quad (13)$$

where, $d_{imin}(k) = d_{imin} + \delta_i * (aim\ \text{-}1)$, and its width is $\delta_i^{'} = \delta_i / p_i$, and *chaotic_operator* is an iterative value as chaotic mapping, *maxgen* is overall generations. The construction of the two-dimensional 4*4 adaptive lattice search is showed in Fig. 1.

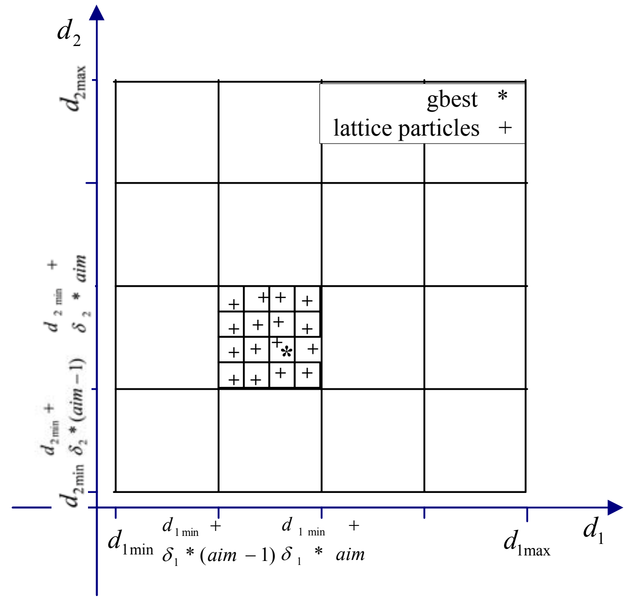

Fig.1. Construction of adaptive lattice search

The main steps of adaptive cPSO are described as follows, and its' mechanism is shown in Fig. 2.

Step1. Set parameters of proposed algorithm. The population of particles *Num* is 30, and acceleration coefficient c_1=2.05, c_2=2.05. The maximum iterative number *maxgen* =200, and the inertia weight *wV* is the linearly decreasing weight [20], given by equation 14. By this way, balance between the global and local searching abilities of the swarm is guaranteed effectively.

$$\begin{aligned} wV(t+1) = wV(t) - (0.9 - 0.4) / \max gen \\ wV(0) = 0.9, k = 1, 2, \ldots, \max gen - 1. \end{aligned} \quad (14)$$

Step2. Initialize the population of PSO randomly.

Step3. Calculate the fitness value and update the local best position of *ith* particle $pbest_i$ and global best position *gbest.*

Step4. Check whether the fitness value lessens in the past 10 iterations, if it changes, go to step 6, otherwise go to step 5.

Step5. Rank all the individuals based on fitness values in descending order, and select the best particles (account for 70% of the sum) to Step7, while the rest are replaced by lattice particles produced by adaptive lattice search. First tune adaptive lattices range according to the best solution *gbest,* and then create lattice particles by (14). Finally, set their velocity to 0 at next generation.

Step6. Update the particles' velocity and position of the population through (8), (9), (12).

Step7. Evaluate the termination criterion. If it reaches the termination criterion, stop the iteration, and *gbest* is the optimal solution. Otherwise, go to step 3.

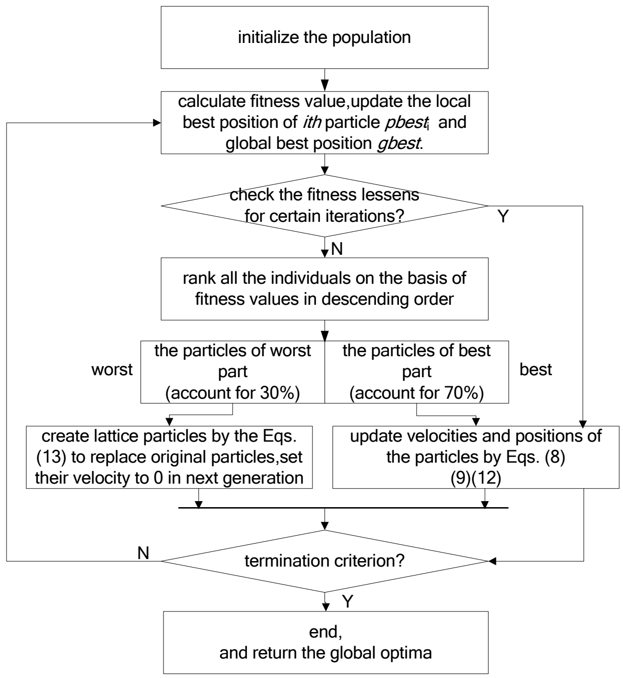

Fig.2. Flow chart of adaptive cPSO for parameter optimization and feature extraction

B. LS-SVM Parameters Optimization Based on Adaptive cPSO

To improve model accuracy and generalization of LS-SVM, adaptive cPSO is proposed to optimize the continuous parameters and discrete feature space of LS-SVM model. The structure of the modeling process is shown in Fig. 3.

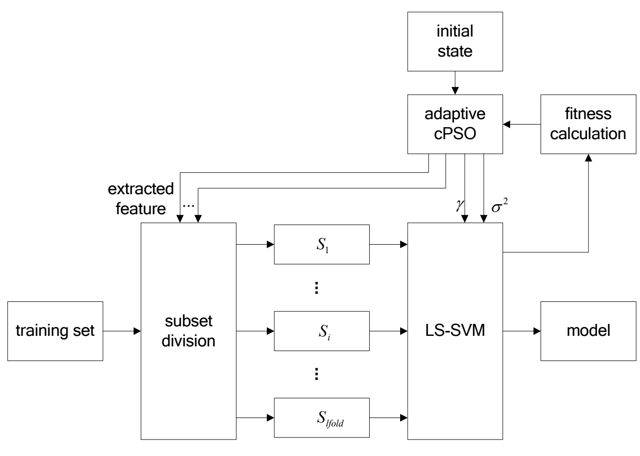

Fig.3. Architecture of modelling process

To assess the performance of the training process comprehensively, we introduce cross validation (CV) [21] algorithm to calculate model error. The training set is split into *lfold* subsets equally, denote $S_1,S_2,\ldots,S_{lfold}$, for each parameter setting and input feature space, and using it to train the LS-SVM model *lfold* times during which S_i is held out while the remaining subsets serve as training set to train the model, and then the trained model is validated using the held-out subset S_i. That enables each subset to take a turn as the testing data, and then calculate the predicting output based on the regression model and corresponding root mean square error (RMSE) of the entire training set. That is a fitness function of the proposed reliable and efficient PSO given by the following equation:

$$fitness = \sqrt{\frac{\sum_{i=1}^{lfold}\sum_{j=1}^{\dim}(f(x_{ij})-y_{ij})^2}{l}} \tag{15}$$

where, y_{ij} is the *jth* actual output of S_i, $f(x_{ij})$ is the *jth* predicting output of S_i, and *dim* represents the dimension of S_i.

As shown in the modelling flow, firstly, we initialize the particles' position and velocity, in which position represents two parameters and the input feature. In each generation the adaptive cPSO generates new particles, by which predictive model with LS-SVM and CV algorithm is trained, and then calculates the predictive output and the corresponding fitness value. Based on the fitness of these particles, we update the local and global best position until the termination condition is satisfied. Finally, we get the global optima and corresponding best predictive model.

IV. EXPERIMENT RESULTS IN MINERAL PROCESS

A. Mineral Process Description

Mineral processes, sometimes called ore dressing, are a complex process with many uncertain factors and multi-variable coupling. And it has the characteristics in terms of large-range continuity, nonlinearity and large-time delay, so there is no reported explicit physical model so far. How- ever, the rich real-time and laboratory analysis data makes it possible to establish a data-driven nonlinear model by LS-SVM.

Based on the production process of a mineral processing plant in western China, the ore-dressing process is composed of the raw ore screening, the shaft furnace roasting, grinding, and the magnetic separation(with strong and weak magnetic field) and concentrate generation [17], which is illustrated in Fig.4.

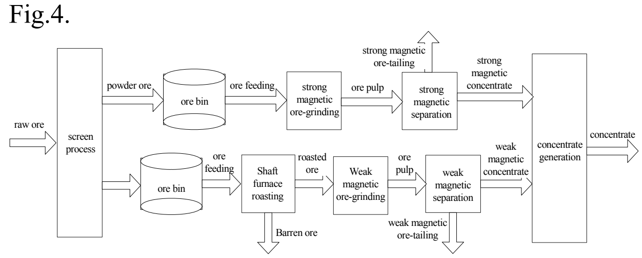

Fig.4. The production process of mineral process.

To establish the predictive model, we extract the related factors to be taken as the input set of the regression model according to the process mechanism, that is $\boldsymbol{X} = [\varepsilon, G_1, G_2, P_1, P_2, Q_1, Q_2, \beta_1, \beta_2, \zeta_1, \zeta_2, \psi_1, \psi_2, \rho]$, in which, ε, G_1, G_2, P_1, P_2, Q_1, Q_2, β_1, β_2, ζ_1, ζ_2, ψ_1, ψ_2 and ρ represent magnetic tube recovery rate, grade of particle ore, grade of roasted ore, particle size of strong magnetic ore, particle size of weak magnetic ore, production rate per hour of grinding of strong magnetic mill, production rate per hour of grinding of weak magnetic mill, grade of strong magnetic concentrate, grade of weak magnetic concentrate, grade of strong magnetic tailing, grade of weak magnetic tailing, running time of grinding of strong magnetic ore, running time of grinding of weak magnetic ore, and grade of gangue respectively.

B. Experiments Results

For the industrial mineral processing data, 575 samples are split into training set with 500 samples and the remaining is test set. To obtain optimum parameters and feature space, the search process is constructed based on 3-fold CV error of the training set. Grid search, GA, PSO and adaptive cPSO are adopted to optimize two parameters, whilst PCA, BPSO and BPSO with chaotic operator are employed to fulfill the task of feature extraction. In every generation, we choose 30 particles to search the best parameters and feature space. When the *gbest* fitness value doesn't change for certain iterations, a 3*3 adaptive lattice search will be adopted to search in the neighborhood of *gbest*. After 200 iterations, we get the best parameters σ^2 =20, γ= 1037.6 and the input feature space *feature* = 1 4 5 8 9 13 14.

The simulation results of proposed method are shown in Fig.5, representing the performance of the model based on adaptive cPSO, in which 500 data are used to train the model and 75 are used to test the model. At the same time, we calculate the output error between the actual and predicted output, the corresponding error autocorrelation is shown in Figs. 6-7. From Figs. 5-7, it is can be seen that the proposed method is of high accuracy and generalization.

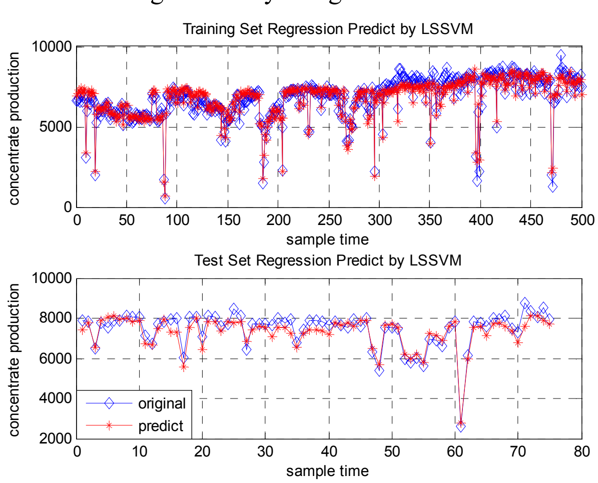

Fig. 5.The performance of obtained model for predicting output of train and test set

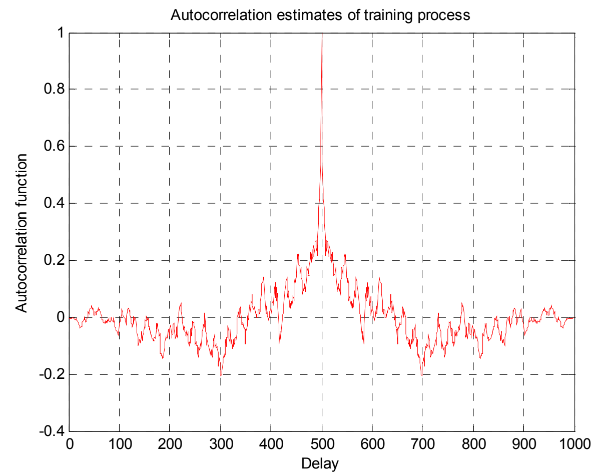

Fig.6. Error autocorrelation digram of training set

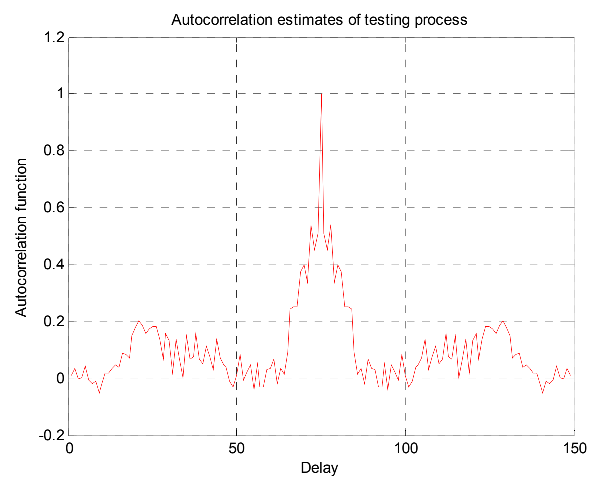

Fig.7. Error autocorrelation digram of test set

In addition, the root mean square error (RMSE) and mean square correlation coefficient r^2 are used as the criterion to evaluate the performance of proposed methods, given by the following equations:

$$RMSE = \sqrt{\sum_{j=1}^{\dim}(f(x_j)-y_j)^2 \Big/ \dim} \tag{16}$$

$$r^2 = \frac{(\dim \sum_{i=1}^{\dim} f(x_i)y_i - \sum_{i=1}^{\dim} f(x_i) \sum_{i=1}^{\dim} y_i \quad)}{(\dim \sum_{i=1}^{\dim} f(x_i)^2 - (\sum_{i=1}^{\dim} f(x_i))^2)(\dim \sum_{i=1}^{\dim} y_i^2 - (\sum_{i=1}^{\dim} y_i)^2)} \tag{17}$$

where, $f(x_j)$ represents the estimated value, y_j denotes the actual value, and *dim* is the number of input.

The training and test set are adopted to validate the performance of LS-SVM predictive model based on optimized by grid search+PCA, PSO+PCA, GA+PCA, PSO +BPSO and adaptive cPSO respectively. The performance indexes are listed in table I. From this table, it can be seen that the model established by LS-SVM with the proposed algorithm has a better regression result on indexes of RMSE and r^2 on training set. Even though indexes on test set is not as good as those obtained by other algorithms, it also is of high accuracy. On the other hand, it reflects the good performance of the proposed algorithm due to the parameter optimization and feature extraction executed on training set, and those are not suitable for test set owing to the difference between the two sets to some extent. Hence, the proposed method is an efficient algorithm for parameter optimization and feature extraction.

TABLE I
PERFORMANCE INDEXES ON OBTAINED MODEL

Algorithm	Performance indexes			
	RMSE of training set	r^2 of training set	RMSE of test set	r^2 of test set
Grid search+PCA	414.0111	0.8926	342.6216	0.8817
GA+PCA	413.2958	0.8930	340.8386	0.8837
PSO+PCA	410.4893	0.8926	342.5434	0.8823
PSO+BPSO	403.3789	0.8981	332.4582	0.8874
Adaptive cPSO	359.4959	0.9198	356.0067	0.8786

V. CONCLUSIONS

This study aims to design a more efficient algorithm that addresses the issues of parameter optimization and feature extraction in modelling based on LS-SVM. A novel chaotic operator is proposed to replace random numbers in standard PSO, which gives similar effect in offering diversity for search algorithm and offers an ergodic, non-repeated, and deterministic solution, leading to a global result. In addition, an adaptive lattice search is introduced to strengthen cPSO' local exploration ability to obtain a more accurate local result in the neighborhood of *gbest*, which contributes to a more accurate global solution. The proposed method is utilized to establish the predictive model between the production rate and the technical indexes of the production procedures in the mineral processing plant. The experimental results show that the proposed adaptive cPSO performs better than other methods regarding convergence accuracy.

REFERENCES

[1] V. Vapnik, "The Nature of Statistical Learning Theory," New York: Springer-Verlag, 1995.

[2] J. A. K. Suykens, J. Vandewalle, "Least squares support vector machine classifiers," *Neural Processing Letters*, vol.9, no.3, pp.293-300, 1999.

[3] Muhsin Tunay Gencoglu,Murat Uyar, "Prediction of flashover voltage of insulators using least squares support vector machines," *Expert Systems with Applications*, vol.36, pp.10789–10798, 2009.

[4] T.V.Gestel, J.A.K. Suykens, D.-E. Baestaens, A. Lambrechts, "Financial time series prediction using least squares support vector machines within the evidence framework," *IEEE Transactions on Neural Networks*, vol. 12, no. 4, pp. 809-821, 2001.

[5] Olivier Devos, Cyril Ruckebusch, Alexandra Durand, "Support vector machines (SVM) in near infrared (NIR) spectroscopy: Focus on parameters optimization and model interpretation, " *Chemometrics and Intelligent Laboratory Systems*, vol. 96, pp.27–33, 2009.

[6] Hoi-Ming Chi, Herbert Moskowitz, OkanK.Ersoy, "Machine learning and genetic algorithms in pharmaceutical development and manufacturing processes," *Decision Support Systems*, vol.48, issue.1, pp.69–80, 2009.

[7] Jorge E.Hurtado, Diego A. Alvarez, "An optimization method for learning statistical classifiers in structural reliability," *Probabilistic Engineering Mechanics*, vol.25, pp.26-34, 2010.

[8] W.J. Kong, W.J. Cheng, J.L. Ding, T.Y. Chai, "A Reliable and Efficient Hybrid PSO Algorithm for Parameter Optimization of LS-SVM for Production Index Prediction Model," in *Third International Symposium on Computational Intelligence and Design,* vol.2, pp.140-143,2010.

[9] J EDWARD, JACKSON, "Principle Components and Factor Analysis; Part1 Principle Components," *Journal of Quality Technology*, vol.12, no.4, 1980.

[10] H.Wold, "Estimation of principal components and related models by iterative least squares," in *Multivariate Analysis*, New York: Academic press, 1966, pp. 391-420.

[11] Oh-Wook Kwon, Te-Won Lee, "Phoneme recognition using ICA-based feature extraction and transformation," *Signal Processing*, vol.84, issue.6, pp.1005-1019, 2004.

[12] J.A.K. Suykens, T. Van Gestel, J. Vandewalle, B.DeMoor, "A support vector machine formulation to PCA analysis and its Kernel version", *IEEE Transactions on Neural Networks* , vol.14,no.2, pp. 447 – 450, 2003.

[13] J. Arenas-Garcia, G. Camps-Valls, "Efficient Kernel Orthonormalized PLS for Remote Sensing Applications," *IEEE Transactions on Geoscience and Remote Sensing*, vol.46, no.10, 2008.

[14] J. Kennedy, R.C. Eberhart, "A discrete binary version of the particle swarm algorithm,"in *Proceedings of 1997 conference systems man and cybernetics*, vol. 5, pp.4104–4108, 1997.

[15] C.K. Zhang,H. Hu, "Using PSO algorithm to evolve an optimum input subset for a SVM in time series forecasting ,"*IEEE International Conference on Systems, Man and Cybernetics* , vol. 4, pp.3793 – 3796, 2005.

[16] M. Clerc, J. Kennedy, "The particle swarm-explosion, stability, and convergence in a multidimensional complex space," *IEEE Transactions on Evolutionary Computation*, vol.6, no.1, pp.58–73, 2002.

[17] X.L. Huang, T.Y. Chai, "Contemporary Integrated Manufacturing System based on ERP/MMES/PCS in Ore Dressing [C]," in *Eighth International Conference on Control, Automation, Robotics and Vision*, vol.3, pp.1879-1884, 2004.

[18] B. Hao, "Starting with parabolas: an introduction to chaotic dynamics," Shanghai :Shanghai Science and Technology Education Press,1993

[19] lal Alatas, Erhan Akin,A. Bedri Ozer, "Chaos embedded particle swarm optimization", *Chaos,Solitons&Fractals,* vol. 40, no.4, pp. 1715 - 1734, 2009.

[20] Y. Shi, R.C. Eberhart, "Empirical study of particle swarm optimization," in *Proceedings of Congress on Evolutionary Computation,* vol.3, pp. 1945-1949, 1999.

[21] Salzberg, S. L. , " On comparing classifiers: pitfalls to avoid and a recommended approach.," *Data Mining and Knowledge Discovery,* vol. 1, pp.317–327, 1997.