J. Cent. South Univ. (2012) 19: 3311−3319
DOI: 10.1007/s11771-012-1409-3

Springer

Deformation prediction of tunnel surrounding rock mass using CPSO-SVM model

LI Shao-jun(李邵军)[1], ZHAO Hong-bo(赵洪波)[2], RU Zhong-liang(茹忠亮)[2]

1. State Key Laboratory of Geomechanics and Geotechnical Engineering (Institute of Rock and Soil Mechanics, Chinese Academy of Sciences), Wuhan 430071, China;
2. School of Civil Engineering, Henan Polytechnic University, Jiaozuo 454003, China

Abstract: A new method integrating support vector machine (SVM), particle swarm optimization (PSO) and chaotic mapping (CPSO-SVM) was proposed to predict the deformation of tunnel surrounding rock mass. Since chaotic mapping was featured by certainty, ergodicity and stochastic property, it was employed to improve the convergence rate and resulting precision of PSO. The chaotic PSO was adopted in the optimization of the appropriate SVM parameters, such as kernel function and training parameters, improving substantially the generalization ability of SVM. And finally, the integrating method was applied to predict the convergence deformation of the Xiakeng tunnel in China. The results indicate that the proposed method can describe the relationship of deformation time series well and is proved to be more efficient.

Key words: deformation prediction; tunnel; chaotic mapping; particle swarm optimization; support vector machine

1 Introduction

The excavation of tunnels and other underground openings causes the deformation in the surrounding rock mass. Evaluation of the deformation behavior of surrounding rock mass is an important aspect of the safety assessment for underground engineering in complex conditions, contributing to the design of tunnel excavation and support. Large deformation may significantly reduce tunnel diameter and damage the existed supports, leading to re-construction and dramatic increase of the engineering costs [1−2]. For this reason, deformation of surrounding rock mass is required to be estimated before tunnel excavation and be measured continuously during and after the excavation. A complete deformation analysis of surrounding rock mass can be computed based on numerical simulation by means of finite element, finite difference, discrete element method [3], etc. Other methods such as the calculation of the 'characteristic line' have been used to obtain a preliminary estimation of the expected tunnel deformation [4]. On the basis of the field measurements of rock mass deformation by some special instruments, such as extensometer, and sliding micrometer, it is important to compare the predicted and observed deformations to understand the stability of tunnel surrounding rock mass and take measures to guarantee the tunnel construction safer and under control. A key problem in this process is to extract the relationship between the time series of deformation based on the observed data, which shows great complexity and non-linearity and is difficult to model by traditional mathematical methods.

The method of time series analysis is a good approach to address this kind of problems in geotechnical engineering. For example, gray method and fuzzy set theory have been applied to build the model of the deformation time series [5−6]. But, it is difficult to present the complex, nonlinear relationship between deformations. The predicted results are not satisfactory. In addition, neural networks (NN) are capable of learning non-linear functional mapping and are suitable for adapting to complex functions. It has been used for the analysis of ground surface settlements due to tunnel excavation [7−10]. NN model has some inherent drawbacks such as slow convergence speed, local minimization and over-fitting. Support vector machine (SVM) has shown a comparable generalization performance to NN. They were applied to model non-linear displacement time series of the high slope of the permanent shiplock of the Three Gorges

Foundation item: Project(NCET-08-0662) supported by Program for New Century Excellent Talents in University of China; Project(2010CB732006) supported by the Special Funds for the National Basic Research Program of China; Projects(51178187, 41072224) supported by the National Natural Science Foundation of China
Received date: 2012−01−09; **Accepted date:** 2012−05−15
Corresponding author: LI Shao-jun, Associate Professor, PhD; Tel: +86−27−87198805; E-mail: sjli@whrsm.ac.cn

Project and a large scale landslide in China [11−12]. More applications of SVM in geotechnical engineering were also reported by ZHAO et al [13], ANTHONY and GOH [14], LI et al [15], etc.

However, one disadvantage of the SVM is that its performance is strongly dependent on the selection of its internal parameters. To overcome this problem, recently, particle swarm optimization (PSO) has been used to search the optimal SVM and proved to be effective. Since chaotic mapping is featured by certainty, ergodicity and stochastic property, in this work, we introduce chaos mapping into the PSO algorithm. And finally, the new model integrating the chaotic system, PSO and SVM (CPSO-SVM) is presented and an example of predicting the deformation of surrounding rock masses of a tunnel is given.

2 Chaotic particle swarm optimization

2.1 Simple particle swarm optimization

The particle swarm optimization (PSO) was originally designed by KENNEDY and EBERHART [16] and has been compared to genetic algorithms for efficiently seeking optimal or near-optimal solutions in large search spaces. The technique involves simulating social behavior among individuals (particles) "flying" through a multidimensional search space, with each particle representing a single intersection of all search dimensions. The particles evaluate their positions relative to a goal (fitness) at each iteration. Particles in a local neighborhood share memories of their "best" positions, and then use those memories to adjust their own velocities and thus subsequent positions. The original formula developed by KENNEDY and EBERHART [16] was improved by SHI and EBERHART [17] with the introduction of an inertia parameter, which increases the overall performance of PSO.

The original PSO formula defines each particle as a potential solution to a problem in D-dimensional space, with particle i represented by $X_i = (x_{i1}, x_{i2}, \cdots, x_{iD})$. Each particle also maintains a memory of its previous best position, $P_i=(p_{i1}, p_{i2}, \cdots, p_{iD})$, and a velocity along each dimension, represented by $V_i=(v_{i1}, v_{i2}, \cdots, v_{iD})$. At each iteration, the $\boldsymbol{P}$ vector of the particle with the best fitness in the local neighborhood, designated g, and the $\boldsymbol{P}$ vector of the current particle are combined to adjust the velocity along each dimension, and that velocity is then used to compute a new position for the particle. The portion of the adjustment to the velocity influenced by the individual's previous best position ($\boldsymbol{P}$) is considered as the cognition component, and the portion influenced by the best position in the neighborhood is the social component.

As to minimum problem, $f(\boldsymbol{X})$ is supposed to be the objection function, $X_i=(x_{i1}, x_{i2}, \cdots, x_{in})$ is the current position of particle, $V_i=(v_{i1}, v_{i2}, \cdots, v_{in})$ is the current speed of particle, and $P_i=(p_{i1}, p_{i2}, \cdots, p_{in})$ is the best position which particle flied, then the best position of particle i can be computed based on the following formula:

$$P_i(t+1)=\begin{cases} P_i(t) \text{ if } f(x_i(t+1) \ge f(P_i(t))) \\ X_i(t+1) \text{ if } f(x_i(t+1) < f(P_i(t))) \end{cases} \tag{1}$$

If the population is s, and $P_{\mathrm{g}}(t)$ is global best position which all particles flied, then

$$P_{\mathrm{g}}(t) \in \{P_0(t), P_1(t), \cdots, P_{\mathrm{s}}(t)\} \mid f(P_{\mathrm{g}}(t)) = \min\{f(P_0(t)), f(P_1(t)), \cdots, f(P_{\mathrm{s}}(t))\} \tag{2}$$

According to the theory of particle swarm optimization, the following equations represent the evolutionary process:

$$v_i(t+1) = wv_i(t) + c_1r_1(t)(p_{ij}(t) - x_i(t)) + c_2r_2(t)(p_{\mathrm{g}}(t) - x_i(t)) \tag{3}$$

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1) \tag{4}$$

where v_i, the velocity for particle i, represents the distance to be traveled by this particle from its current position; x_{ij} represents the position of particle i; p_{ij} is the best previous position of particle i; p_{g} represents the best position among all particles in the population; r_1 and r_2 are two independent uniformly distributed random variables within the range of [0, 1]; c_1 and c_2 are positive constant parameters called acceleration coefficients which control the maximum step size; w is called the inertia weight that controls the impact of previous velocity of particle on its current one. In the standard PSO, Eq. (3) is used to calculate the new velocity based on its previous velocity and the distances from both its own best historical position and its neighbors' best position to its current position. Generally, the value of each component in v_i can be clamped to the range of $[-v_{\max}, v_{\max}]$ to control excessive roaming of particles outside the search space. Then, the particle flies toward a new position based on Eq. (4). This process is repeated until a user-defined stop criterion is reached.

2.2 Chaotic particle swarm optimization

In simple PSO, parameters of PSO (inertia weight factor, etc) are crucial in searching the optimum solution efficiently. The performance of PSO depends greatly on its parameters. Many scholars think that the parameters w, r_1 and r_2 (in Eq. (3)) are the key factors that affect the convergence of the PSO. The inertia weight w is the modulus that controls the impact of previous velocity on the current one. The larger scale contributes to searching for the global optimal solution in an expansive area, but its precision is not good because of the rough search. The smaller scale improves the precision of the optimal solution, but the algorithm may be trapped in a local optimization. So, the balance between exploration and exploitation in PSO is dictated by w. Thus, proper control of the inertia weight is very important to search the optimal solution accurately and efficiently. In simple PSO, inertia weight cannot ensure optimization ergodicity entirely in phase space, because it is random. Parameters r_1 and r_2 cannot ensure the optimization ergodicity entirely in search space.

In order to overcome this problem, the well-known logistic equation is incorporated into the simple PSO. The logistic equation is defined as

$$x_{n+1} = \mu \cdot x_n(1 - x_n), 0 \le x_0 \le 1 \tag{5}$$

where μ is the control parameter, x is a variable and n=0, 1, 2, ⋯. Although the logistic equation is deterministic, it exhibits chaotic dynamics when μ=4 and x_0 is not 0, 0.25, 0.5, 0.75 and 1. The track of chaotic variable can travel ergodically over the whole search space. In general, the chaotic variable has special characteristics, i.e. ergodicity, pseudo-randomness and irregularity.

Furthermore, we incorporate chaotic mapping with certainty, ergodicity and stochastic property into the simple PSO so as to improve the global convergence. The parameters w, r_1 and r_2 in Eq. (3) are controlled by the following equations:

$$w(t+1) = 4.0 \times w(t)(1 - w(t)),\ 0 < w(t) < 1 \tag{6}$$

$$r_i(t+1) = 4.0 \times r_i(t)(1 - r_i(t)),\ 0 \le r_i(t) \le 1,\ i = 1, 2 \tag{7}$$

3 CPSO based support vector machine

The SVM was proposed by VAPNIK [18] and is illustrated in Fig. 1 [19]. It is used to train nonlinear relationships based on the structural risk minimization principle that seeks to minimize an upper bound of the generalization error rather than to minimize the empirical error implemented in neural networks. Merit of the SVM is that training is a uniquely solvable quadratic optimization problem.

The SVM uses nonlinear mapping based on an internal integral function to transform an input space to a high dimension space and then seeks a nonlinear relationship between inputs and outputs in that space. The SVM not only has theoretical support but also can find global optimum solutions for those problems with small training samples, high dimensions, nonlinear and local optima. A wide variety of applications such as pattern recognition and nonlinear regression, have empirically shown the SVM's ability to be generalized. But performances of SVM are controlled by its parameters. The integrating method of CPSO-SVM uses chaotic particle swarm optimization to search the kernel function and its training parameters with the input of the training sample set. The tentative SVMs are tested by the testing sample set. The training process of the SVM will be completed when the identified SVM can give good generalized predictions for testing samples.

The algorithm can be summarized as follows:

Step 1: Collect a set of monitored deformation time series to construct a training SVM sample set. A calibration sample set is built by selecting randomly

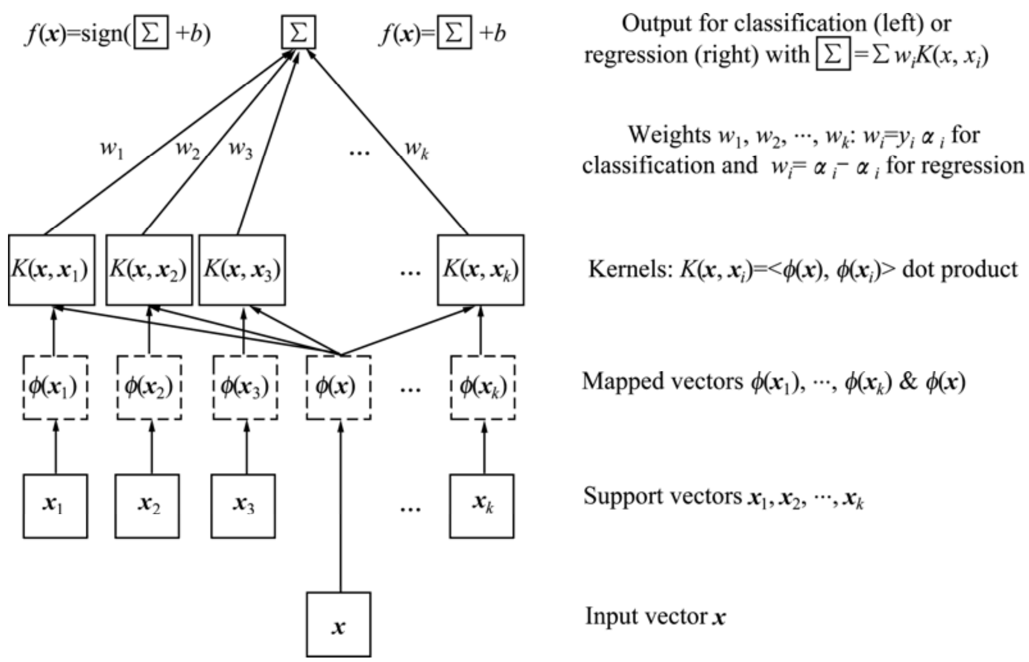


Fig. 1 Support vector machine for classification and regression

from the monitored deformation time series, which may not be included in the training sample set.

Step 2: Initialize parameters for PSO such as the number of population size, range of the kernel function and its parameters including C and σ.

Step 3: Select randomly a kernel function from common examples of kernel functions such as polynomials, Gaussian radial base, and Sigmoid. Produce randomly a set of C, and σ in the given ranges. Every created kernel function and its parameters such as C and σ are regarded as an individual of the tentative SVM.

Step 4: Solve the quadratic programming problems, including every tentative SVM individual, to obtain their support vectors.

Step 5: The selected parameters and the obtained support vectors represent a SVM model. Use the calibration samples to test the prediction ability of the SVM models. The applicability of the model is identified by fitness:

$$f=\sqrt{\frac{\sum_{i=1}^{n}(x_i-x_i')^2}{n}} \tag{8}$$

where x_i and x'_i are the monitored and predicted deformations for the i-th calibration sample series. n is the number of the calibration samples.

Step 6: If the fitness is accepted, then the training procedure of the SVM is completed. Otherwise, use Eq. (4) to create a new SVM individual.

Step 7: If all new individuals of population size are generated, then return to **Step 4**; otherwise, return to **Step 6**.

4 Deformation prediction of surrounding rock mass by CPSO-SVM methods

4.1 SVM representation of deformation time series

The deformation of surrounding rock mass can be regarded as a nonlinear displacement time series, and deformation time-dependent series $\{x_i\}=\{x_1, x_2, \cdots, x_N\}$ can be obtained by in situ monitoring. Modeling the nonlinear deformation series means finding the relationship between the deformation x_{i+p} at time $i+p$ and its deformation $x_i, x_{i+1}, \cdots, x_{i+p-1}$ at the next p time steps, i.e. $x_{i+p}=f(x_i, x_{i+1}, \cdots, x_{i+p-1})$. As a nonlinear function, $f(\cdot)$ expresses the nonlinear relationship of the deformation time series.

According to the theory of SVM, the nonlinear relationship mentioned above can be obtained by learning the measured deformation. For example, the nonlinear relationship of the deformation time series can be obtained by learning the deformation behavior at $n-p$ time steps $x_i, x_{i+1}, \cdots, x_{i+p-1}$, $i=1, \cdots, n-p$, and it can be expressed as

$$f(x_{n+m})=\sum_{i=1}^{n-p}(\alpha_i-\alpha_i^*)K(\boldsymbol{X}_{n+m},\boldsymbol{X}_i)+b \tag{9}$$

where $f(x_{n+m})$ is the deformation at time $(n+m)$; $\boldsymbol{X}_{n+m}$ is the vector of deformation at the next p time steps, $\boldsymbol{X}_{n+m}=(x_{n+m-p}, x_{n+m-p+1}, \cdots, x_{n+m-1})$; $\boldsymbol{X}_i$ is the deformation vector of training samples, $\boldsymbol{X}_i=(x_i, x_{i+1}, \cdots, x_{i+p-1})$; $K(\cdot)$ is the kernel function; p is the number of historical points; α, α^* and b are obtained by solving the following quadratic programming problem:

$$\text{Max } W(\alpha,\alpha^*)=-\frac{1}{2}\sum_{i,j=1}^{n-p}(\alpha_i-\alpha_i^*)(\alpha_j-\alpha_j^*)K(\boldsymbol{X}_i\cdot\boldsymbol{X}_j)+\sum_{i=1}^{k}x_{i+p}(\alpha_i-\alpha_i^*)-\varepsilon\sum_{i=1}^{n-p}(\alpha_i+\alpha_i^*) \tag{10}$$

$$\text{Subject to}\begin{cases}\sum_{i=1}^{n-p}(\alpha_i-\alpha_i^*)=0,\\ 0\le\alpha_i,\alpha_i^*\le C, i=1,2,\cdots,n-p\end{cases} \tag{11}$$

4.2 In situ monitored data

Xiakeng Tunnel in Wan-Gan Railway is taken here as an example [20] for the verification of the proposed method. The tunnel has the length of 65 m and is excavated in seriously weathered phyllite strata. The horizontal convergence deformation of surrounding rock mass in one section DK2946+07 were perfectly monitored from July to September, as listed in Table 1.

4.3 Parameters of model

The surrounding rock mass deformation of this tunnel has been predicted using the proposed model. The history time step is important for prediction efficiency and affects the performance of SVM. In this work, the deformation of surrounding mass was predicted using different history time steps, and the history time steps were set to be 3, 4, 5, 6 and 7. The parameters of chaotic particle swarm optimization are: population size of 50, and $c_1=c_2=2.0$. The searching range of SVM parameters is 0.000 1−10 000.

4.4 Comparison of predicting results

Based on the CPSO-SVM method described above, the prediction on deformation time series was carried out. The final results are shown in Figs. 2−6. According to these results, it can be seen the CPSO-SVM model performs acceptable prediction. When history time step is 6, the performance of CPSO-SVM is the best. The values of Lagrange multipliers of SVM (history time step of 6) are listed in Table 2.

Table 1 Monitored value of convergence deformation

Monitored date (mm−dd)	Deformation value/mm
07−31	0
08−01	1.33
08−02	3.1
08−03	4.72
08−04	6.5
08−05	7.88
08−06	9.25
08−07	11.01
08−08	12.76
08−09	14.52
08−10	15.77
08−11	17.6
08−12	19.48
08−13	21.76
08−14	22.55
08−15	24.08
08−16	24.84
08−17	25.45
08−18	26.12
08−19	26.77
08−20	27.42
08−21	27.45
08−22	28.32
08−23	29.22
08−24	28.62
08−25	28.82
08−26	29.02
08−27	29.55
08−28	29.38
08−29	29.21
08−30	29.72
08−31	30.03
09−01	30.72
09−02	29.83
09−03	30.08
09−04	30.41
09−05	30.74
09−06	30.96
09−07	30.38

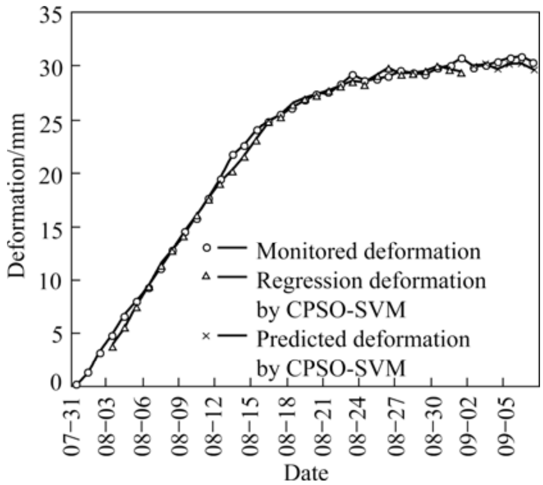


Fig. 2 Prediction of surrounding rock mass deformation using CPSO-SVM (History time step of 3)

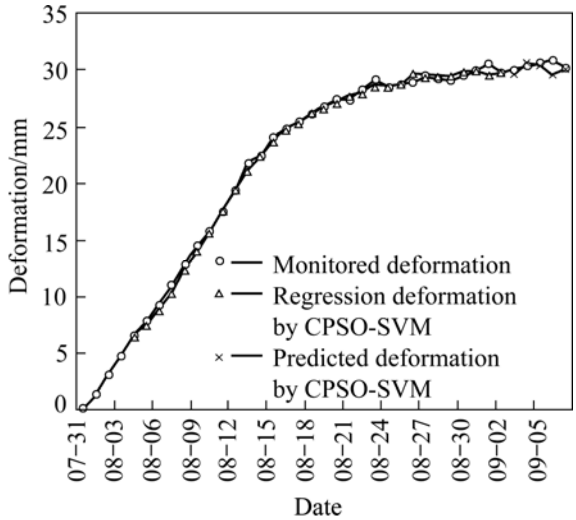


Fig. 3 Prediction of surrounding rock mass deformation using CPSO-SVM (History time step of 4)

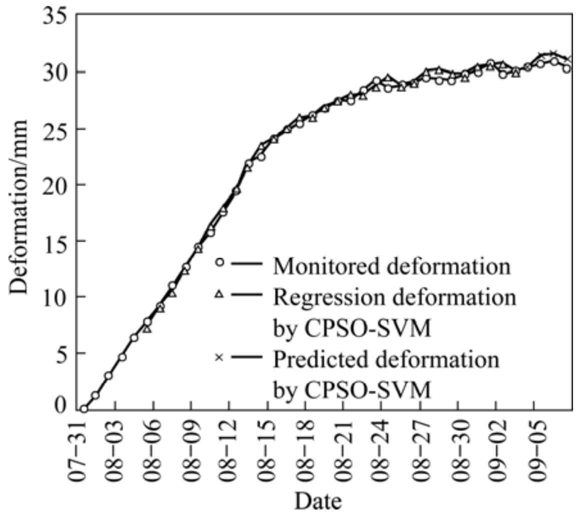


Fig. 4 Prediction of surrounding rock mass deformation using CPSO-SVM (History time step of 5)

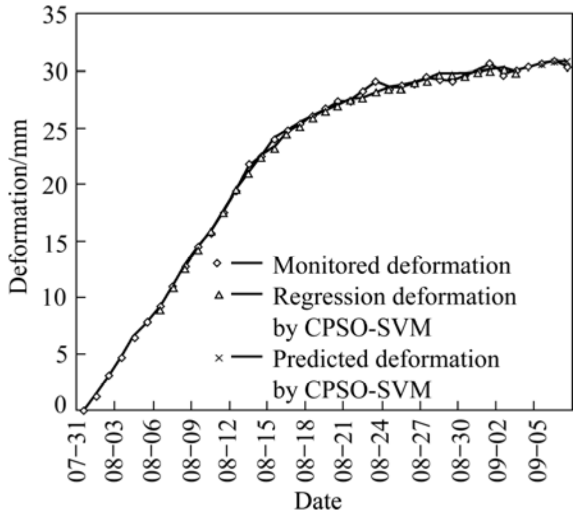


Fig. 5 Prediction of surrounding rock mass deformation using CPSO-SVM (History time step of 6)

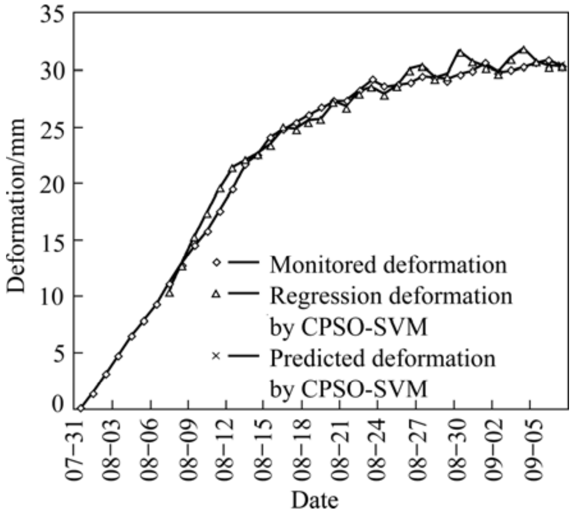


Fig. 6 Prediction of surrounding rock mass deformation using CPSO-SVM (History time step of 7)

Table 2 Values of Lagrange multipliers of SVM

Number of learning sample	α	α*
1	0.000 0	752.642 3
2	2 074.834 8	0.000 0
3	0.000 0	2 368.574 8
4	0.000 0	26.285 0
5	3 230.030 6	0.000 0
6	0.000 0	1 603.219 8
7	0.000 0	1 182.834 2
8	529.481 5	0.000 0
9	0.000 0	3 089.980 7
10	4 912.686 5	0.000 0
11	0.000 0	1 586.499 9
12	1 611.596 0	0.000 0
13	19.643 1	0.000 0
14	0.000 0	4 442.608 6
15	2 699.395 5	0.000 0
16	0.000 0	4 912.686 5
17	4 912.686 5	0.000 0
18	0.000 0	4 912.686 5
19	4 912.686 5	0.000 0
20	4 912.686 5	0.000 0
21	0.000 0	1 102.723 1
22	0.000 0	789.741 2
23	0.000 0	4 912.686 5
24	0.000 0	2 895.108 4
25	3 110.208 8	0.000 0
26	0.000 0	4 912.686 5
27	4 037.823 5	0.000 0
28	545.574 3	0.000 0
29	4 912.686 5	0.000 0
30	0.000 0	2 931.056 6

5 Discussions

To evaluate predicted results, some evaluation criteria are defined in the following formula:

$$\begin{cases} E_{\mathrm{MB}}=\dfrac{\sum\limits_{i=1}^{n}(x_i-x_i')}{n} \\ E_{\mathrm{RMS}}=\sqrt{\dfrac{\sum\limits_{i=1}^{n}(x_i-x_i')^2}{n}} \\ E_{\mathrm{MR}}=\dfrac{\sum\limits_{i=1}^{n}\left(\dfrac{x_i-x_i'}{x_i}\right)}{n}\times 100\% \end{cases} \tag{12}$$

where x_i and x_i' are the monitored and predicted deformations of the surrounding rock mass of a tunnel, respectively; n is the number of formation time series.

The performance of CPSO-SVM using different history time step was studied. The results are shown in Table 3 and Table 4. It is indicated that the history time step has great influence on the performance of the model, resulting in the different prediction results. In this case study, when history time step is 6, the performance of model is better than others. From Table 4, it can be seen that the proposed method is reliable (E_{MB}=−0.084 3 mm and E_{MR}=1.54%).

In order to verify the performance of CPSO-SVM, further study based on EASVM is carried out to solve the same problem. The method EASVM is proposed by the authors before [11], in which genetic algorithm is adopted to determine the optimal SVM parameters. The predicted results are shown in Fig. 7 and Fig. 8. The performance comparison between CPSO-SVM and EASVM method is listed in Table 5. It can be seen that the maximum relative errors of CPSO-SVM are less than those of EASVM under the same computing conditions, thus the performance of CPSO-SVM is better than EASVM. The efficiency of algorithm is important to deformation prediction. The converge processes of CPSO-SVM and EASVM are shown in Fig. 9 and Fig. 10, respectively. Obviously, the convergence speed of CPSO-SVM is faster, which proves that the method of CPSO-SVM has the higher efficiency.

6 Conclusions

1) The method of CPSO-SVM for deformation prediction of tunnel surrounding rock mass is proposed.

Table 3 Performance of CPSO-SVM using different history time step in learning data

Performance index	Number of historical time steps				
	3	4	5	6	7
Mean bias error (MBE)	0.166 967	0.073 183	−0.206 24	0.073 356	0.342 07
Root-mean-square error (RMSE)	0.571 998	0.345 895	0.459 797	0.347 104	0.941 724

Table 4 Performance of CPSO-SVM using different history time step in testing data

Performance index	Number of historical time steps				
	3	4	5	6	7
Mean bias error (MBE)	0.349 3	0.371 6	0.495 8	−0.084 3	0.079 6
Root-mean-square error (RMSE)	0.509 4	0.636 7	0.567 5	0.284 1	0.305 2
Maximum relative error (MRE)/%	2.31	4.25	2.47	1.54	1.21

Table 5 Performance comparison between CPSO-SVM and EASVM

Method	Performance index	Calibration		Validation	
		n=4	n=6	n=4	n=6
CPSO-SVM	Mean bias error (MBE)	0.073 2	0.073 4	0.371 6	−0.084 3
	Root-mean-square error (RMSE)	0.345 9	0.347 1	0.636 7	0.284 1
	Maximum relative error (MRE)/%	5.39	3.27	4.25	1.54
EASVM	Mean bias error (MBE)	−0.715 0	−0.524 5	−4.407 2	0.949 8
	Root-mean-square error (RMSE)	1.228 6	0.765 3	3.629 6	1.121 4
	Maximum relative error (MRE)	7.50%	11.82%	1.66%	5.73%

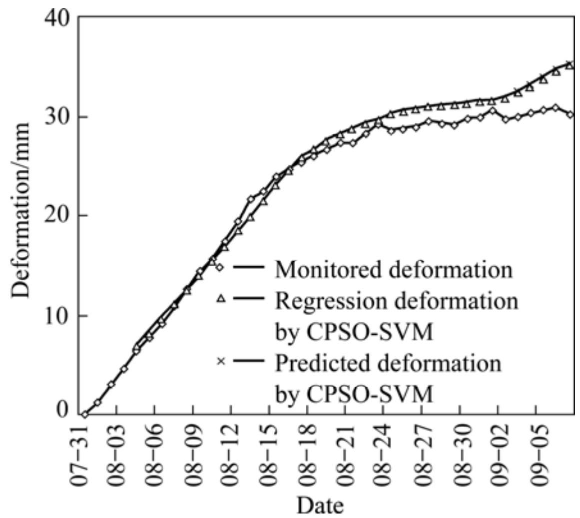


Fig. 7 Prediction of surrounding rock mass deformation using EASVM (History time step of 4)

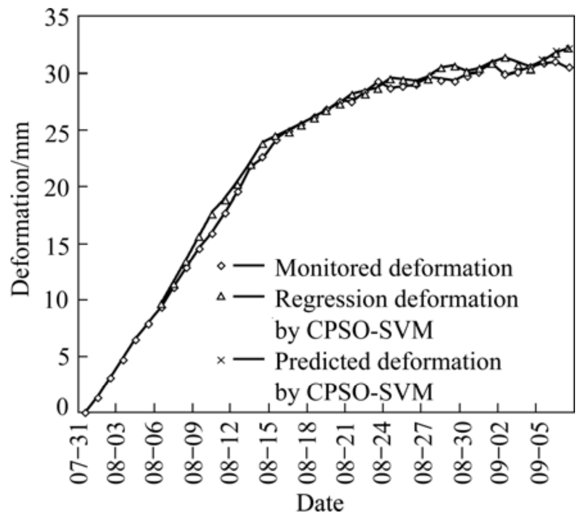
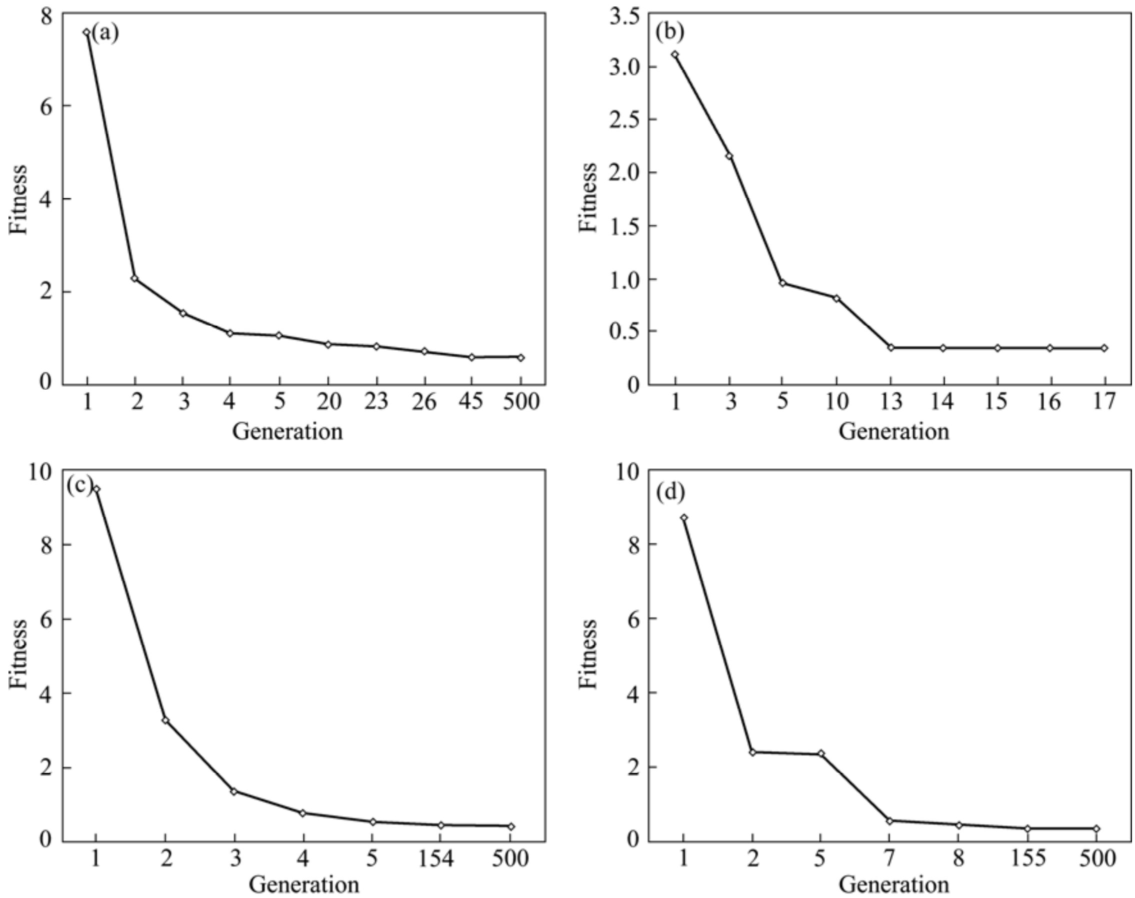


Fig. 8 Prediction of surrounding rock mass deformation using EASVM (History time step of 6)

It can also be applied to some other underground openings. The monitored deformation is regarded as a time series and used as the learning samples of SVM. The relationship among deformation time series is built by SVM, and the parameters of SVM are searched by CPSO.

2) The results obtained from the practical example show that the proposed method can predict the deformation of tunnel surrounding rock mass efficiently. It has a good generalization performance.

3) It should be also noted that the capability of the method in making accurate predictions depends on the quantity of data used for SVM training to some degree. If the data are not representative or sample training is inadequate, the proposed method should be treated with cautions. Therefore, the arrangement, collection and



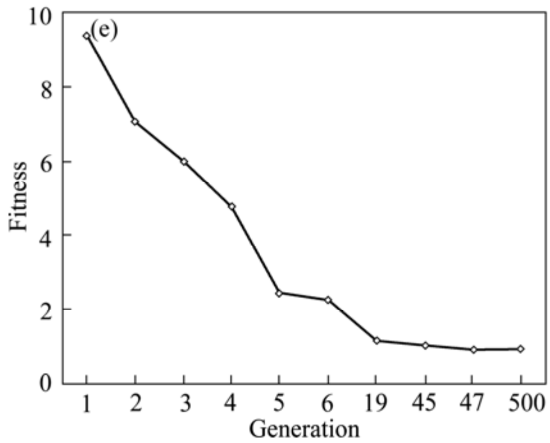


Fig. 9 Convergence process using CPSO-SVM at different history time: (a) Step of 3; (b) Step of 4; (c) Step of 5; (d) Step of 6; (e) Step of 7

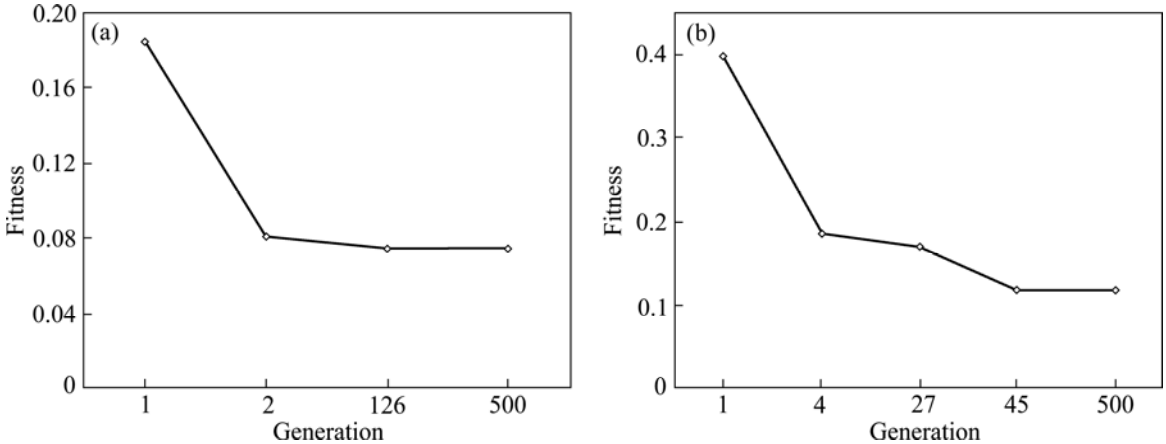


Fig. 10 Convergence process using EASVM at different history time: (a) Step of 4; (b) Step of 6

analysis of the monitored data should be carefully carried out.

References

[1] KONTOGIANNI V A, STIROS S C. Predictions and observations of convergence in shallow tunnels: Case histories in Greece [J]. Engineering Geology, 2002, 63: 333−345.

[2] FENG X T, HUDSON J A. Rock engineering design [M]. CRC Press, 2011.

[3] JIAO Y Y, FAN S C, ZHAO J. Numerical investigation of joint effect on shock wave propagation in jointed rock masses [J]. Journal of Testing and Evaluation, 2005, 33(3): 197−203.

[4] BROWN E T, BRAY J W, LADANYI B, HOEK E. Ground response curves for rock tunnels [J]. Journal of Geotechnical Engineering, ASCE, 1983, 109(1): 15−39.

[5] LI W X, LI H N. Fuzzy system models (FSMs) for analysis of rock mass displacement caused by underground mining in soft rock strata [J]. Expert Systems with Applications, 2009, 36(3): 4637−4645.

[6] HUANG L C, XU Z S, PENG L M. Deformation prediction in soft wall rock tunnel based on the equal- dimension gray filling model [J]. Journal of Railway Science and Engineering, 2009, 6(6): 13−17. (in Chinese)

[7] SUWANSAWAT S, EINSTEIN H H. Artificial neural networks for predicting the maximum surface settlement caused by EPB shield tunneling [J]. Tunnelling and Underground Space Technology, 2006, 21(2): 133−150.

[8] SHI J, ORTIGAO J A R, BAI J J. Modular neural networks for predicting settlements during tunneling [J]. Journal of Geotechnical and Geoenviroment Engineering, 1998, 124(5): 389−395.

[9] KIM C Y, BAE G J, HONG S W, et al. Neural network based prediction of ground surface settlements due to tunneling [J]. Computers and Geotechnics, 2001, 28: 517−547.

[10] NEAUPANE K M, ADHIKARI N R. Prediction of tunneling-induced ground movement with the multi-layer perceptron [J]. Tunnelling and Underground Space Technology, 2006, 21: 151−159.

[11] FENG X T, ZHANG Z Q, SHENG Q. Estimating mechanical rock mass parameters relating to the Three Gorges Project permanent shiplock using an intelligent displacement back analysis method [J]. International Journal of Rock Mechanics and Mining Sciences, 2000, 37(7): 1039−1054.

[12] FENG X T, ZHAO H B, LI S J. Modeling non-linear displacement time series of geo-materials using evolutionary support vector machines [J]. International Journal of Rock Mechanics and Mining Sciences, 2004, 41(7): 1087−1107.

[13] ZHAO H B, RU Z L, YIN S D. Updated support vector machine for seismic liquefaction evaluation based on the penetration tests [J]. Marine Georesources & Geotechnology, 2007, 25(3): 209−220.

[14] ANTHONY T C G, GOH S H. Support vector machines: Their use in geotechnical engineering as illustrated using seismic liquefaction data [J]. Computers and Geotechnics, 2007, 34(5): 410−421.

[15] LI S J, FENG X T, ZHAO H B, et al. Forecast analysis of monitoring data for high slopes based on three-dimensional geological information and intelligent algorithm [J]. International Journal of Rock Mechanics and Mining Sciences, 2004, 41(3): 519−520.

[16] KENNEDY J, EBERHART R C. Particle swarm optimization [C]// Proceedings of IEEE international conference on neural networks. Perth, Australia: IEEE Service Center, 1995: 1 942−1 948.

[17] SHI Y, EBERHART R C. Empirical study of particle swarm optimization [C]// Proceedings of the Congress on Evolutionary Computation (CEC '99), Washington D C, USA, 1999, 3: 1945−1950.

[18] VAPNIK V N. The nature of statistical learning theory [M]. Springer: New York, 1995.

[19] SHEN J D, SYAU Y R, LEE E S. Support vector fuzzy adaptive network in regression analysis [J]. Computers and Mathematics with Applications, 2007, 54: 1353−1366.

[20] JING S T, ZHU Y Q, SONG Y X. Reliability analysis of tunnel structure [M]. Beijing: Chinese Rail Road Press, 2002. (in Chinese)

(Edited by YANG Bing)