



A NETWORK TRAFFIC HYBRID PREDICTION MODEL OPTIMIZED BY IMPROVED HARMONY SEARCH ALGORITHM

Z. Tian, S. Li, Y. Wang, X. Wang

Abstract: The telecommunication and Ethernet traffic prediction problem is studied. Network traffic prediction is an important problem of telecommunication and Ethernet congestion control and network management. In order to improve network traffic prediction accuracy, a network traffic hybrid prediction model was proposed by using the advantages of grey model and Elman neural network, grey model and Elman neural network predictive values were independently obtained, the different weight coefficients of two prediction models were given. In terms of weight coefficients optimization, an improved harmony search algorithm with better convergence speed and accuracy was proposed, the optimal weight coefficients of network traffic hybrid prediction model were determined through this algorithm, two prediction models results were multiplied by the weight coefficients to obtain the final prediction value. The network traffic sample data from an actual telecommunication network was collected as simulation object. The simulation results verified that the proposed network traffic hybrid prediction model based on improved harmony search algorithm has higher prediction accuracy.

Key words: *network traffic, grey model, Elman neural network, prediction, improved harmony search*

Received: September 19, 2014

DOI: 10.14311/NNW.2015.25.034

Revised and accepted: June 10, 2015

1. Introduction

The telecommunication or Ethernet network traffic is an important parameter of the network management. During designing a telecommunication or Ethernet network congestion control strategy for the case that network source is limited, an accurate network traffic prediction is very important for reducing network congestion, reasonably allocating network resource, improving the service quality of the network and finding abnormal network behavior [1, 12].

Zhongda Tian – Corresponding author, Shujiang Li, Yanhong Wang, Xiangdong Wang, Shenyang University of Technology, School of Information Science and Engineering, Shenyang, Liaoning, 110870, China, E-mail: tianzhongda@126.com, lisj2005@126.com, sshuang123456@163.com, 503563853@qq.com

Leland et al., studies found that Ethernet network traffic has self similarity characteristic [8], the self similarity can be measured by Hurst index. The study pointed out that the Hurst index of the network traffic is greater than 0.5. Network traffic has self similarity characteristic, thus the network traffic is predictable. Their study laid the foundation work for the network traffic prediction.

At present, many scholars made a lot of research work for telecommunication and Ethernet network traffic prediction. Some linear prediction models such as auto regressive moving average (ARMA) [6, 18], auto regressive integrated moving average (ARIMA) [16, 19] and fractional auto regressive integrated moving average (FARIMA) [13] were used for network traffic prediction. The literature [5] studied the above linear models, in which the prediction accuracy of each model with different time scales is performed by experiment, through the simulation, the author pointed out that the appropriate time scale of each model. But with the complexity of the network increases, network traffic characteristics are beyond the Poisson or Markov distribution. Therefore, the linear prediction models has the deficiencies in theory, it is difficult to ensure the accuracy of the prediction. The network nonlinear prediction models include least square support vector machine (LSSVM) [9], artificial neural networks [17, 11] and GM (1,1) grey model [20], etc. Although nonlinear prediction model can improve predictive accuracy more than the linear model, there are also disadvantages. Neural network is easy to fall into local optimal value, also it is difficult to determine the network structure. LSSVM prediction model needs small sample, but it is difficult to determine the key model parameters. Grey model has good prediction effect only for stationary time series. So the nonlinear prediction model of network traffic is also very difficult to guarantee the prediction accuracy.

From the current research results, single prediction model is difficult to achieve a more accurate prediction effect, prediction error is relatively large. Combing different prediction models together can compensate the shortcomings of each prediction model, and can describe the characteristics of the network traffic more accurately.

GM (1,1) grey model has good prediction accuracy for smooth time series, when the network traffic fluctuation is relatively large, the prediction precision will decreased. The Elman neural network has good prediction effect for nonlinear time series. This paper proposed a network traffic hybrid prediction model through the combination of the two models with their respective advantages. GM (1,1) grey model can weaken the randomness of the original network traffic sequence, and the Elman neural network has good prediction ability for nonlinear sequence, through hybrid model for taking better prediction effect. Two prediction models are given different weight coefficients, two prediction models predictive values are multiplied by the respective weighting coefficients to obtain a final prediction value. Weight coefficients are optimized by using an improved harmony search algorithm proposed in this paper. The actual network traffic data collected from telecommunication network is treated as simulation object. Through simulation and comparison with other common prediction models, it indicates that the proposed model has higher prediction precision and smaller prediction errors.

2. Predictability analysis of network traffic

The predictability characteristic of the time series can be measured through Hurst index that expressed as H [15]. A time series is random when H is 0.5, and it means no correlation between events. Time series is described as anti-persistent when $H \in [0, 0.5)$, and is described as persistent when $H \in [0.5, 1)$. It means there is self-similarity between events when $H \in (0.5, 1)$. H is greater then the time series similarity is greater. Therefore, the larger the Hurst index, self-similar (long associated) and nonlinearity characteristic is higher [21]. The typical Hurst index calculation method is Rescaled Range (R/S) analysis, can be expressed as:

$$(R/S)_n = An^H, \quad (1)$$

wherein, n is the number of time series samples, R is rescaling range and be expressed as

$$R = \max(X_{t,n} - \min X_{t,n}), t = 1, 2, \dots, n, \quad (2)$$

S is standard deviation, H is Hurst index, and A is a constant. Drawing the diagram $\lg(R/S)_n$ corresponding to $\lg n$, calculating slope of data points using least square method, that is Hurst index.

In this paper, 500 groups of network traffic data from a core 3G communication router in Liaoning branch of China Unicom are collected, the sampling period of network traffic is 10 minutes. Fig. 1 shows the sample data. The Hurst index of network traffic time series is obtained by rescaled R/S method, as shown in Fig. 2. The Hurst index of network traffic time series have 0.8525, apparently satisfied $0.5 < H < 1$, so the network traffic time series has predictability, self similar, nonlinear and long-related characteristics. Therefore, how to select the appropriate prediction model is very important, the paper will discuss this problem in the following sections.

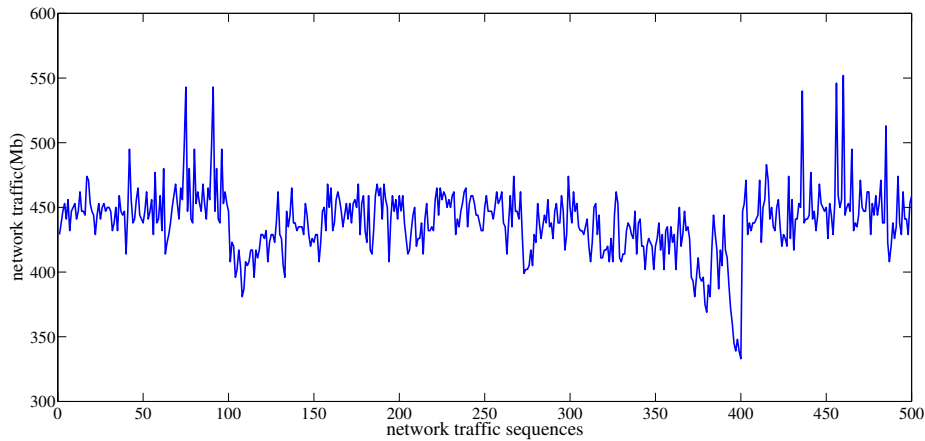


Fig. 1 Network traffic sample data.

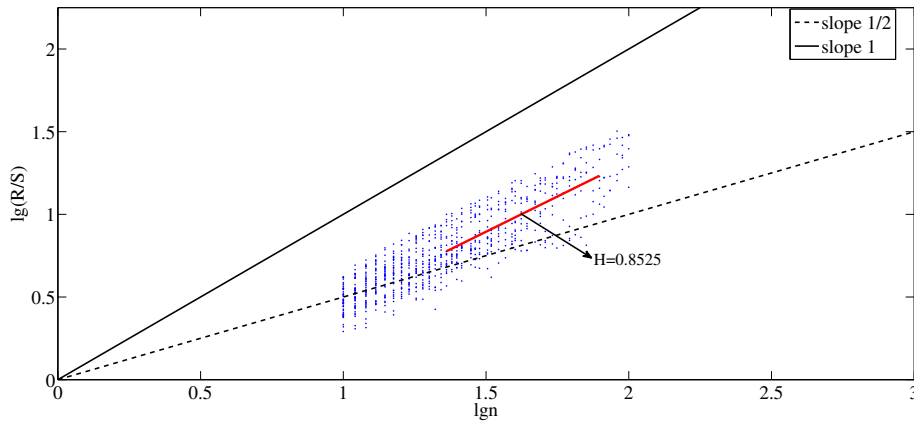


Fig. 2 The Hurst index of network traffic time series.

3. Network traffic hybrid prediction model

The prediction model has a great influence on the prediction accuracy. According to the characteristics of network traffic discussed in Section 2, this paper proposed grey model and Elman neural network hybrid prediction model. The grey model can predict the general trend of network traffic. The Elman neural network model can reduce the influence caused by nonlinear factors. The final predicted value is not obtained through a simple addition, but through an appropriate optimization algorithm to obtain different weight coefficients. The proposed hybrid prediction model combines the advantages of grey model and Elman neural network, so it has good prediction results.

3.1 Grey model

GM (1,1) prediction model is the basic model of grey model, the network traffic sequence D^0 as

$$\{d^0(k)\}, k = 1, 2, \dots, n. \tag{3}$$

Because the network traffic sequence is random variation, it is unable to build a prediction model for network traffic sequence directly. Therefore sequence D^1 is generated by one-accumulate of sequence D^0 .

$$\{d^1(k)\}, k = 1, 2, \dots, n, \tag{4}$$

where

$$d^1(k) = \sum_{k=1}^n d^0(k). \tag{5}$$

Then

$$d^0(k) + ad^1(k) = b. \tag{6}$$

The Eq. (6) is the original form of GM (1,1) prediction model. The basic form of GM (1,1) model is analyzed from network traffic sequence with an average idea, it can replace $d^1(k)$ by $z^1(k) = 0.5(d^1(k) + d^1(k - 1))$. Generating the mean consecutive neighbors of D^1 . Thereby the following form can be obtained as

$$d^0(k) + az^1(k) = b, \quad (k = 2, 3, \dots, n). \tag{7}$$

After differentiating the Eq. (7) it can be obtained

$$\frac{dd^1(t)}{dt} + ad^1(t) = b. \tag{8}$$

Parameter identification of Eq. (8) with the least square method is as follows:

$$\hat{\mathbf{a}} = [a \ b]^T = (\mathbf{Z}^T \mathbf{Z})^{-1} \mathbf{Z}^T \mathbf{Y}, \tag{9}$$

where

$$\mathbf{Y} = \begin{bmatrix} d^0(2) \\ d^0(3) \\ \vdots \\ d^0(n) \end{bmatrix}, \tag{10}$$

$$\mathbf{Z} = \begin{bmatrix} -z^1(2) & 1 \\ -z^1(3) & 1 \\ \vdots & \vdots \\ -z^1(n) & 1 \end{bmatrix}. \tag{11}$$

Thus GM (1,1) prediction model can be obtained as following two equations:

$$\hat{d}^1(k + 1) = (d^0(1) - b/a)e^{-ak} + b/a, \tag{12}$$

$$\hat{d}^0(k + 1) = \hat{d}^1(k + 1) - \hat{d}^1(k). \tag{13}$$

3.2 Elman neural network prediction model

Elman neural network is a two layer BP neural network with feedback structure. Elman neural network add a feedback layer on the basic structure of the traditional BP neural network, so the system has the ability to adapt to the time-varying characteristics. It is able to approach to any continuous nonlinear function if appropriate numbers of neurons are selected, at the same time, the convergence speed is much faster than the BP network. So Elman neural network is very suitable for nonlinear time series prediction. The structure of Elman neural network is shown as Fig. 3.

Elman neural network adopts dynamic back propagation learning algorithm. The gradient descent method is employed to train the network. The target of algorithm is to minimize the mean square error of sample output and system recognition output by adjusting the weight of each layer of network. $\mathbf{u}(t)$ is network external

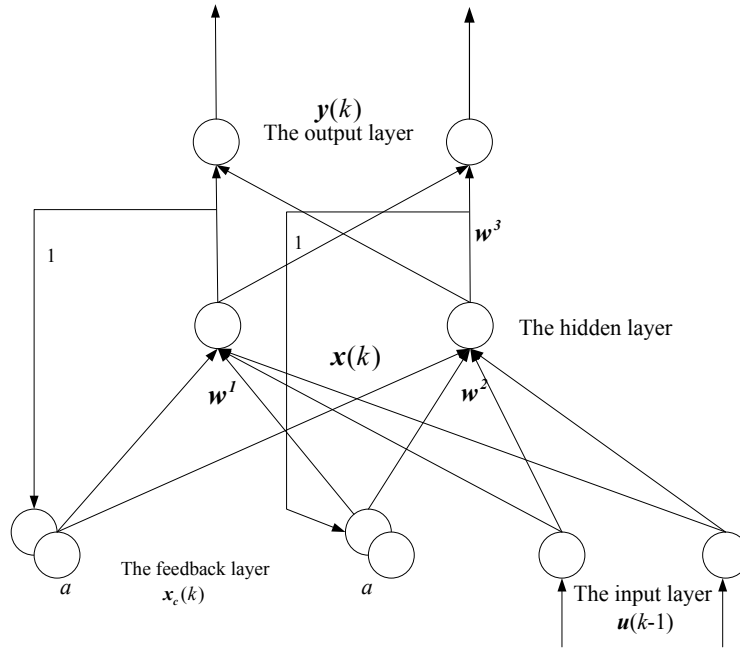


Fig. 3 The structure of Elman neural network.

input sequence, $\mathbf{y}_c(t)$ is feedback layer output, $\mathbf{y}(t)$ is network output, \mathbf{w}^1 is connection weight matrix from feedback layer to the hidden layer, \mathbf{w}^2 is connection weight matrix from input layer to the hidden layer, \mathbf{w}^3 is connection weight matrix from hidden layer to the output layer. The specific learning algorithm is

$$E(K) = \frac{1}{2}(\mathbf{y}_d(k) - \mathbf{y}(k))^T(\mathbf{y}_d(k) - \mathbf{y}(k)), \quad (14)$$

$$\Delta w_{ij}^3 = \eta_3 \delta_j^0 \mathbf{x}_j(k), \quad (15)$$

$$\Delta w_{jq}^2 = \eta_2 \delta_j^h \mathbf{u}_q(k-1), \quad (16)$$

$$\Delta w_{jl}^1 = \eta_1 \sum_{i=1}^m (\delta_i^0 w_{ij}^3) \frac{\partial \mathbf{x}_j(k)}{\partial w_{jl}^1}, \quad (17)$$

$$\delta_i^0 = (\mathbf{y}_{di}(k) - \mathbf{y}_i(k))g'(\cdot), \frac{\partial \mathbf{x}_j(k)}{\partial w_{jl}^1} = f'_j(\cdot)\mathbf{x}_l(k-1) + a \frac{\partial \mathbf{x}_j(k-1)}{\partial w_{jl}''}, \quad (18)$$

$$\delta_j^h = \sum_{i=1}^m (\delta_i^0 w_{ij}^3) f'_j(\cdot), \quad (19)$$

wherein, $i = 1, 2, \dots, m$; $j = 1, 2, \dots, n$; $q = 1, 2, \dots, r$; η_1, η_2, η_3 is learning step of $\mathbf{w}^1, \mathbf{w}^2$ and \mathbf{w}^3 ; $f(\cdot)$ is a nonlinear function, and taken as sigmoid function

$$f(x) = \frac{1}{1 + e^{-x}}. \quad (20)$$

3.3 Hybrid prediction model

The network traffic hybrid prediction model in this paper is shown in Fig. 4.

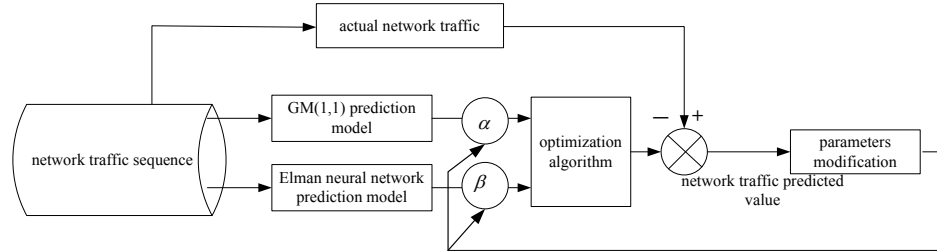


Fig. 4 The network traffic hybrid prediction model.

The prediction steps of the model in this paper are described below. The modeling phase:

Step 1: Give the network traffic sample sequence as $t(k-m), t(k-m+1), \dots, t(k)$, k is sampling time, m is embedding dimension.

Step 2: Use $t(k-m), t(k-m+1), \dots, t(k)$ as model input, $t(k+1)$ as output, modeling and training GM (1,1) model according to Section 3.1, modeling and training Elman neural network according to Section 3.2, continuous updating k , obtained model parameters.

Step 3: Define the GM(1,1) prediction model output as $\hat{t}_1(i)$, the Elman neural network prediction model output as $\hat{t}_2(i)$, the hybrid prediction model output as $\hat{t}(i)$, $\hat{t}(i)$ satisfies the following Eq. (22), α and β is weighting coefficient.

$$\hat{t}(i) = \alpha\hat{t}_1(i) + \beta\hat{t}_2(i), \quad -1 < \alpha < 1, -1 < \beta < 1. \quad (21)$$

Step 4: Use the root mean square error of actual and prediction value of network traffic as fitness function, expressed as follows:

$$\text{fitness}(j) = \frac{1}{N} \sqrt{\sum_{k=1}^N (t_k - \hat{t}_k)^2}, \quad (22)$$

wherein N is sequence length, j is the number of iterations. The improved harmony search algorithm proposed in the next section is used for the optimization of α and β .

The prediction phase:

Step 1: The current sampling time is k , the input sequence as $t(k-m), t(k-m+1), \dots, t(k)$, the GM(1,1) model prediction output as $\hat{t}_1(k+1)$, the Elman neural network prediction output as $\hat{t}_2(k+1)$.

Step 2: The prediction output multiplied by the respective weighting coefficient, the final prediction value $\hat{t}(k+1)$ can be obtained according to Eq. (22).

Step 3: Let $k = k + 1$, $\hat{t}(k+1)$ obtained in Step2 into sequence $t(k-m), t(k-m+1), \dots, t(k), \hat{t}(k+1)$ replaced $t(k)$, $t(k-m)$ is removed, go to Step1 until prediction process end.

Different weight coefficients α and β will affect the prediction precision, so how to determine the most suitable α and β is the key part of hybrid prediction model. In this paper, the optimized weight coefficients can be determined by the improved harmony search algorithm proposed in the next section.

4. Improved harmony search algorithm

4.1 Standard harmony search algorithm

Harmony search (HS) algorithm was proposed in 2001 [22], it is a swarm intelligence optimization algorithm coming from imitation of music creation process. HS algorithm generated a new individual by all individual cooperation, do not rely on the initial conditions, it has simple structure, easy to realize and fast convergence speed and other good characteristics. Literature [10, 7] showed the optimization performance of HS algorithm is better than genetic algorithm, simulated annealing algorithm, etc. HS algorithm has been widely applied to power system optimization and energy saving [2], economic cost optimization [14], multi-objective flexible job shop scheduling problem [4], architectural design [3] and other optimization of the actual problems.

The following is the standard HS algorithm optimization process. For a given optimization problem

$$\min f(\mathbf{X}), \text{ s.t. } x_i \in [L_i, U_i], \tag{23}$$

where \mathbf{X} is a u -dimensional real-valued vector and $f(\mathbf{X})$ is a real-valued continuous function of optimization problem.

Step 1: The HS algorithm first initializes the following parameters: harmony memory size (HMS), harmony memory consideration rate (HMCR), pitch adjusting rate (PAR), number of iteration (NI) and range of decision variables $[L_i, U_i]$.

Step 2: Random create harmony memory as follows:

$$\mathbf{HM} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \vdots \\ \mathbf{X}_m \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix}, \quad m = HMS. \tag{24}$$

Step 3: Generate a candidate solution $\mathbf{X}_{\text{new}} = (x_{\text{new}}(1), x_{\text{new}}(2), \dots, x_{\text{new}}(j))$, $x_{\text{new}}(j)$ is as the following steps:

$$x_{\text{new}}(j) = \begin{cases} x_{md(i),j} & \text{if } r_1 < HMCR, \\ x_{\text{new}}(j) \in [L_i, U_i] & \text{if } r_1 \geq HMCR, \end{cases} \quad (25)$$

where $x_{md(i),j}$ is a randomly selected component of J-th column component in \mathbf{HM} , $x_{\text{new}}(j)$ is a random value of J-th component, r_1 is a uniformly distributed random number between $[0, 1]$. If $x_{\text{new}}(j)$ is solution of the selected component of \mathbf{HM} , then according to the following trimming:

$$x_{\text{new}}(j) = \begin{cases} x_{\text{new}}(j) \pm r_3 \times BW & \text{if } r_2 < PAR, \\ x_{\text{new}}(j) & \text{if } r_2 \geq PAR, \end{cases} \quad (26)$$

where r_2 and r_3 are the uniformly distributed random numbers between $[0, 1]$.

Step 4: Update the harmony memory, if $f(\mathbf{X}_{\text{new}}) < f(\mathbf{X}_w)$, \mathbf{X}_w is the worst solution of harmony memory, then $\mathbf{X}_w = \mathbf{X}_{\text{new}}$.

Step 5: Repeat Step 3 and Step 4, until the number of iterations reaches NI .

4.2 Improved harmony search algorithm

The speed and convergence precision of HS algorithm are related to the algorithm parameters. In order to improve the algorithm search efficiency, and to overcome the shortcomings of the standard HS algorithm, the following is improved in this paper.

The parameter $HMCR$ should be dynamically regulated from large to small values, this ensures HM algorithm will fully search the harmony memory at first, then go to search external harmony memory in later during iterative search. In order to improve the diversity of population, adjustment method is as follows:

$$HMCR(t) = \begin{cases} HMCR(t-1) \times \rho & HMCR(t) > HMCR_{\text{max}}, \\ HMCR_{\text{min}} & HMCR(t) \leq HMCR_{\text{min}}, \\ HMCR_{\text{max}} & t = 0. \end{cases} \quad (27)$$

In the early of the HS algorithm, smaller PAR can be good at searching a better region. In the late of HS algorithm, the larger PAR is conducive to make the algorithm jump out of local optimal value. So PAR is changed from small to large. PAR change strategy in this paper is

$$PAR(t) = \frac{PAR_{\text{max}} - PAR_{\text{min}}}{\sqrt{NI}} \times \sqrt{t} + PAR_{\text{min}}. \quad (28)$$

As parameter BW , in the early of HS algorithm, the larger BW is good at searching in a wide range. Later in the search algorithm, smaller BW is beneficial

to the precise search within a small area. Therefore, BW should be changed from large to small, the changing strategy is as follows:

$$BW(t) = BW_{\min} + (BW_{\max} - BW_{\min}) \times e^{-t}. \quad (29)$$

The new solution x_{new} is generated through by selecting a component randomly in memory when the standard HS algorithm updates memory. This will cause the algorithm search direction uncertain with larger randomness. This paper reference to crossover idea of genetic algorithm, a new solution x_{new1} is generated according to the standard HS algorithm, then the second new solution x_{new2} is generated by randomly single-point crossover between x_{new1} and the other column component. If $f(x_{\text{new1}}) < f(x_{\text{new2}})$, x_{new1} is reserved, otherwise x_{new2} is reserved.

In order to validate the performance of the improved HS algorithm, the following Sphere function expressed as Eq. (31), Rosenbrock function expressed as Eq. (32) and Rastrigin function expressed as Eq. (33) are used to test.

$$f_1(x) = \sum_{i=1}^n x_i^2, \quad (30)$$

$$f_2(x) = \sum_{i=1}^n 100(x_{i+1}^2 - x_i)^2 + (1 - x_i)^2, \quad (31)$$

$$f_3(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]. \quad (32)$$

In order to illustrate the effect of the improved harmony search algorithm in this paper, standard harmony search algorithm is compared. The standard HS algorithm parameters are the same as literature [1]: $HMCR = 0.9$, $PAR = 0.3$, $BW = 0.01$. The improved HS algorithm parameters in the paper are: $HMCR_{\max} = 1$, $HMCR_{\min} = 0.4$, $\rho = 0.97$, $PAR_{\max} = 0.9$, $PAR_{\min} = 0.4$, $BW_{\min} = 0.0001$, $BW_{\max} = 1$, cross-factor $u = 0.8$. The number of iterations is set to 5000, $HMS = 6$. The Tab. I shows the three test functions dimension, the global optimal value and initialization parameters.

Functions	Dimensions	Range of parameters	Optimal value
Sphere	4	[-100, 100]	0
Rastrigin	30	[-5.12, 5.12]	0
Rosenbrock	30	[-2.048, 2.048]	0

Tab. I Function parameters.

In order to eliminate the influence of random, all the algorithms are run for 20 times and the average value is chosen as the optimization results. Figs. 5, 6 and 7 are fitness convergence curve of the three functions in one test process. For the convenience of the display, the horizontal ordinate is recording a fitness value per 50 iterations, the vertical ordinate is the fitness takes on logarithm of 10. It can be seen from the figure the improved HS algorithm in this paper is better than the standard HS algorithm with faster convergence, and better fitness value. Tab. II

gives the comparison result of HS and IHS algorithms, including the best fitness value and the average fitness value. The best fitness value and average fitness value reflects the convergence of the algorithm, the mean value of fitness reflects the robustness of the algorithm. As can be seen from Tab. II, the best fitness, the success rate and other performance indexes of improved HS are better than standard HS algorithm.

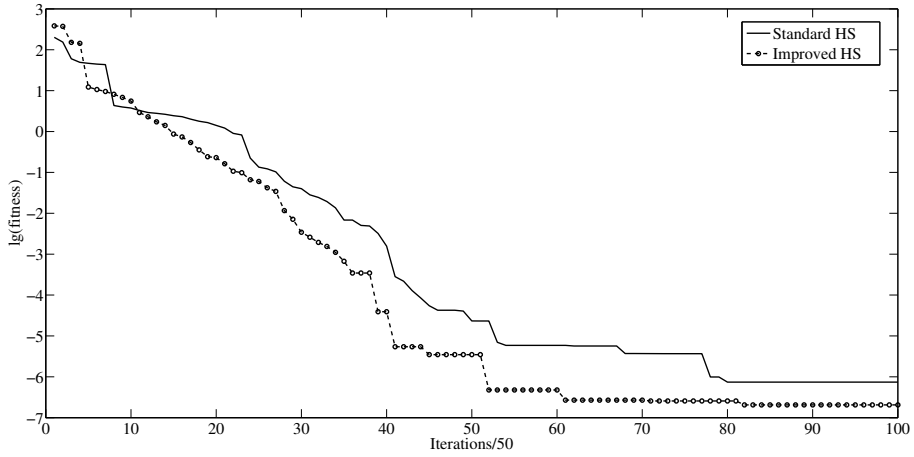


Fig. 5 Sphere function convergence curve.

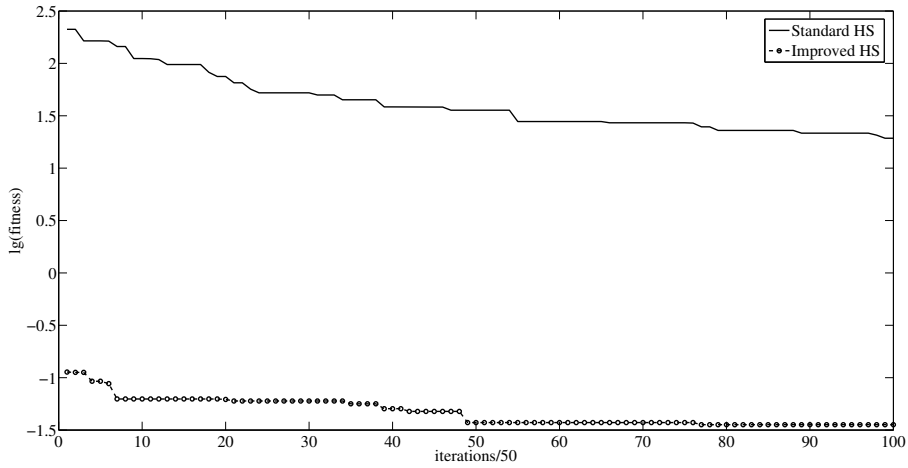


Fig. 6 Rastrigin function convergence curve.

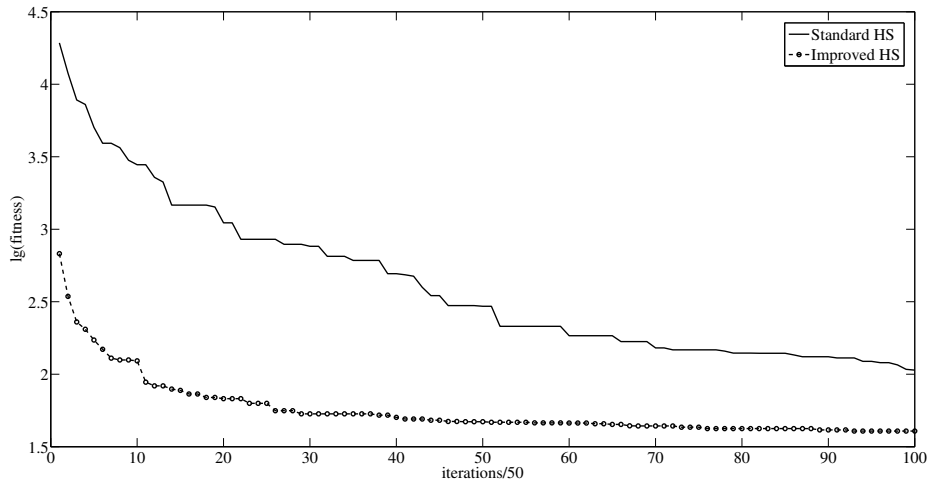


Fig. 7 Rosenbrock function convergence curve.

Functions	Average fitness	Best fitness	Standard deviation	Success rate [%]
Sphere by HS	1.3272	0.6125	0.3157	76
Sphere by IHS	0.0321	0.0011	0.0004	100
Rastrigin by HS	13.6480	11.3287	2.0176	55
Rastrigin by IHS	0.2939	-0.2435	0.0623	100
Rosenbrock by HS	15.1525	10.5421	2.5652	20
Rosenbrock by IHS	0.1236	0.0952	0.0463	100

Tab. II Algorithm simulation results.

This section verifies that the convergence speed and accuracy of improved HS algorithm are better than the standard HS algorithm. In the simulation, the improved HS algorithm is applied to the weight coefficients optimization of network traffic hybrid prediction model.

5. Simulation

The network traffic data for simulations is the same with Section 2. The former 400 groups for training and modelling network traffic hybrid prediction model, the remaining 100 groups of data are used to validate the predictive accuracy of the model. Network traffic queues embedding dimension m is chosen as 20. After trained, the GM (1,1) prediction model parameters are $b = 147.8073$, $a = -0.0093$.

Elman neural network input layers are the same with embedding dimension m , are taken to be 20, maximum number of iterations is 3000, iteration goal is 0.0001, the hidden layer number is 10, the output layer number is 1, the multi step

prediction can be obtained through iteration. When GM (1,1) model and Elman neural network model are trained, the network traffic predicted value is used for weight coefficient optimization according to the Eq. (22). The improved harmony search algorithm proposed in this paper is used as optimization algorithm. The fitness function uses the root mean square error between the network traffic actual value and predictive value. Parameters of the improved HS algorithm are used the same values as described in Section 4. Through the optimization search, the optimal weight coefficient is obtained as $\alpha = 0.3018$, $\beta = 0.7216$. Fitness convergence curve as shown in Fig. 8.

Fig. 9 is GM (1,1) prediction model comparison between predicted value and actual value of the 100 groups network traffic. Fig. 10 is Elman neural network

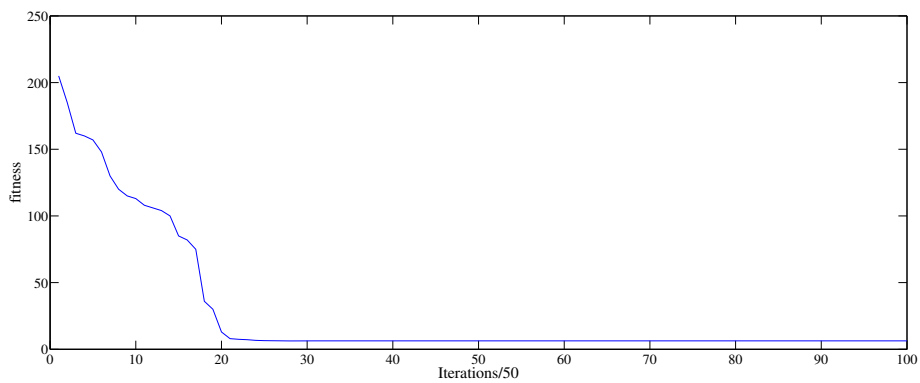


Fig. 8 RMSE fitness convergence curve.

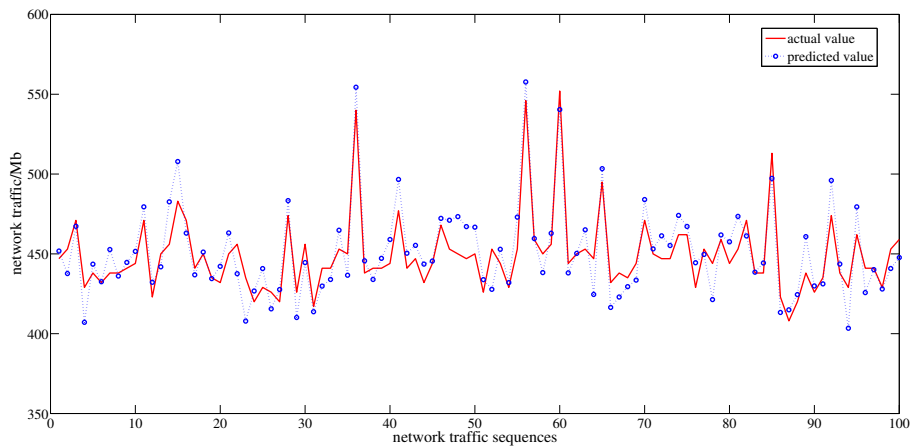


Fig. 9 The GM(1,1) prediction model comparison between predicted and actual value.

prediction model comparison between predicted value and actual value of the 100 groups network traffic. Fig. 11 is hybrid prediction model comparison between predicted value and actual value of the 100 groups network traffic.

In order to further verify prediction precision of network traffic hybrid prediction model. Fig. 12 shows the comparison between predicted value and actual value of network traffic by ARMA model in literature [18], Fig. 13 shows the comparison

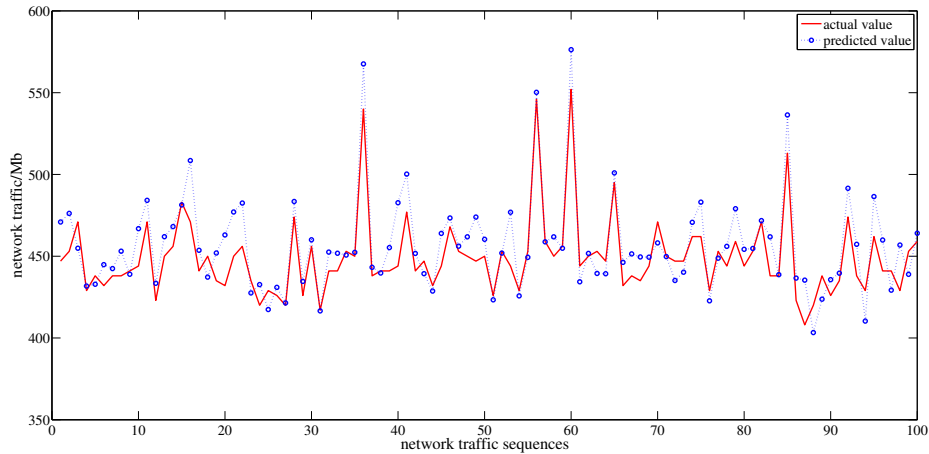


Fig. 10 *The Elman neural network prediction model comparison between predicted and actual value.*

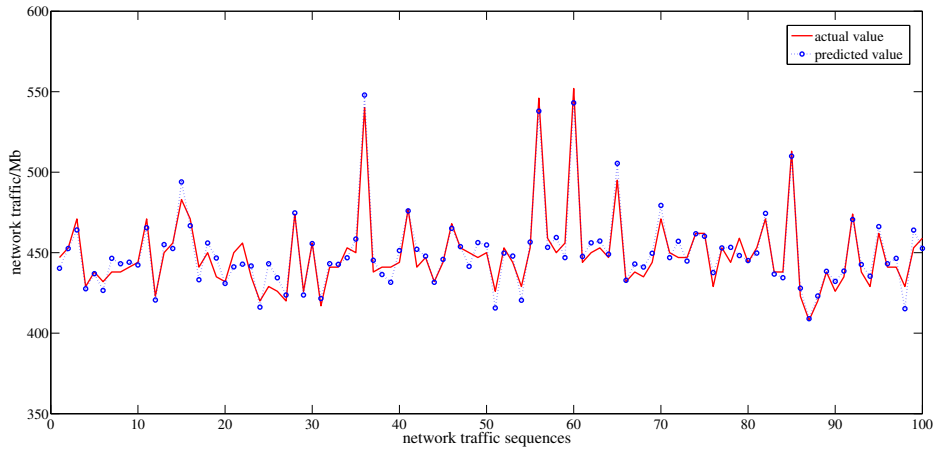


Fig. 11 *The hybrid prediction model comparison between predicted and actual value.*

between predicted value and actual value of network traffic by LSSVM in literature [9]. The ARMA model parameters are determined according to the Akaike information criterion (AIC), it is $p = 2$, $q = 1$. The LSSVM parameters obtained by grid cross validation model, the parameters as $\gamma = 32.03$, $\sigma^2 = 4.361$.

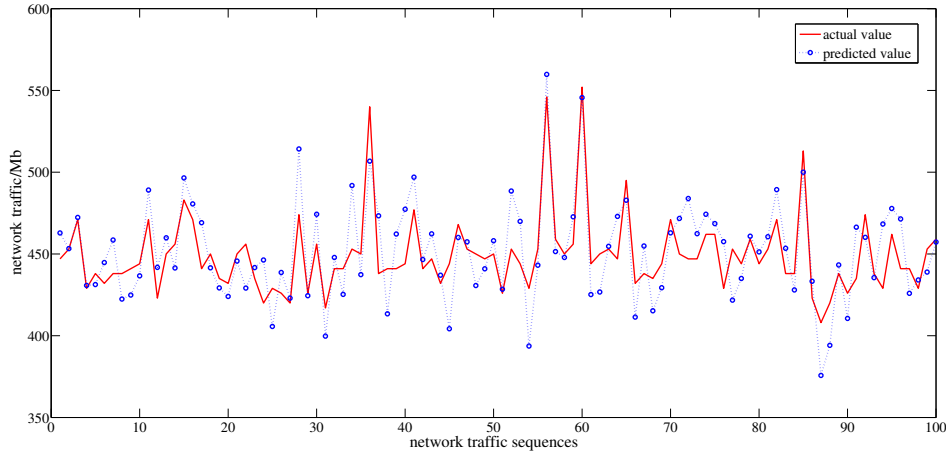


Fig. 12 The ARMA prediction model comparison between predicted and actual value.

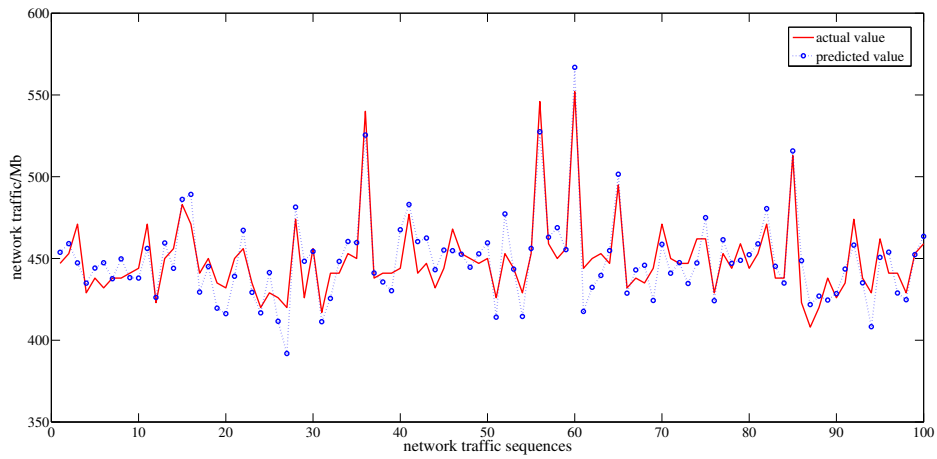


Fig. 13 The LSSVM prediction model comparison between predicted and actual value.

Prediction model	RMSE	MAPE
Hybrid prediction model	6.2722	1.035
GM(1,1)	12.7614	2.402
Elman neural network	16.2809	3.424
LSSVM	12.0303	2.253
ARMA	18.9903	3.503

Tab. III The prediction error comparison of five models.

Fig. 14 shows the prediction error distribution of five prediction models, the prediction error can be observed from this figure that the prediction model proposed in this paper has smaller prediction error and more uniform error distribution.

Tab. III is root mean square error (RMSE) and mean absolute percentage error (MAPE) comparison of five prediction models.

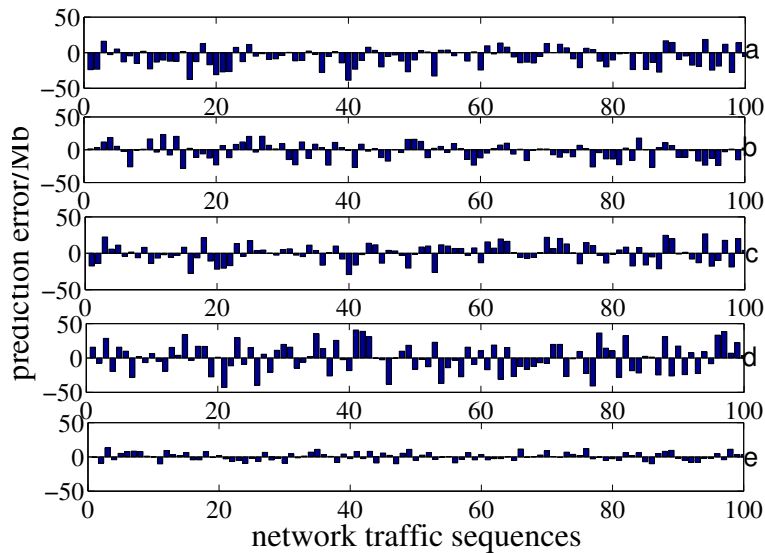


Fig. 14 The prediction error distribution of five prediction models (a: Elman b: GM(1,1) c: LSSVM d: ARMA e: Hybrid prediction model).

Through comparison between the predictive value and actual value of network traffic from Fig. 9 to Fig. 13, the prediction error contrast from Fig. 14, the RMSE and MAPE contrast from Tab. III, it can be seen the network traffic hybrid prediction model has higher prediction accuracy than other common single prediction models. The main reason of the prediction accuracy improvement is that two models are used to predict, thus can overcome the shortcomings of the single prediction model. GM (1,1) grey model can weaken the randomness of the network traffic original sequence, and the Elman neural network has good prediction ability for nonlinear sequence, through hybrid model can achieve better prediction effect. At the same time, two prediction model weight coefficients are optimized by the

improved HS algorithm, the prediction error can complement each other at each moment, so the prediction precision is improved.

5.1 Conclusions

In this paper, in order to increase the prediction accuracy of network traffic, a hybrid prediction model combining grey model and Elman neural network is proposed. The optimal weight coefficients of hybrid prediction model are optimized by an improved HS algorithm, it can achieve the best prediction results. The simulation was performed using the actual data collected from the actual telecommunication network. Simulation results demonstrate that the improved HS algorithm has higher convergence speed and better fitness value. The simulation also shows network traffic hybrid prediction model has higher prediction accuracy and smaller prediction error than other prediction models.

Acknowledgement

This work was supported by the Liaoning Province Doctor Startup Fund under Grant 20141070.

References

- [1] ALARCON-AQUINO V., BARRIA J.A. Multiresolution FIR neural-network-based learning algorithm applied to network traffic prediction. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*. 2006, 36(2), pp. 208–220, doi: [10.1109/TSMCC.2004.843217](https://doi.org/10.1109/TSMCC.2004.843217).
- [2] ARUL R., RAVI G., VELUSAMI S. Solving optimal power flow problems using chaotic self-adaptive differential harmony search algorithm. *Electric Power Components and Systems*. 2013, 41(8), pp. 782–805, doi: [10.1080/15325008.2013.769033](https://doi.org/10.1080/15325008.2013.769033).
- [3] DE MEDEIROS G.F., KRIPKA M. Optimization of reinforced concrete columns according to different environmental impact assessment parameters. *Engineering Structures*. 2014, 59, pp. 185–194, doi: [10.1016/j.engstruct.2013.10.045](https://doi.org/10.1016/j.engstruct.2013.10.045).
- [4] GAO K.Z., SUGANTHAN P.N., PAN Q.K., CHUA T.J., CAI T.X., CHONG C.S. Pareto-based grouping discrete harmony search algorithm for multi-objective flexible job shop scheduling. *Information Sciences*. 2014, 289, pp. 76–90, doi: [10.1016/j.ins.2014.07.039](https://doi.org/10.1016/j.ins.2014.07.039).
- [5] JIANG M., WU C.M., ZHANG M., HU D.J. Research on the comparison of time series models for network traffic prediction. *Acta Electronica Sinica*. 2009, 37(11), pp. 2353–2358, doi: [037222112\(2009\)1122353206](https://doi.org/037222112(2009)1122353206).
- [6] LANER M., SVOBODA P., RUPP M. Parsimonious fitting of long-range dependent network traffic using ARMA models. *IEEE Communications Letters*. 2013, 17(12), pp. 2368–2371, doi: [10.1109/LCOMM.2013.102613.131853](https://doi.org/10.1109/LCOMM.2013.102613.131853).
- [7] LEE K.S., GEEM Z.W. A new meta-heuristic algorithm for continuous engineering optimization, harmony search theory and practice. *Computer Methods in Applied Mechanics and Engineering*. 2005, 194(36/37/38), pp. 3902–3933, doi: [10.1016/j.cma.2004.09.007](https://doi.org/10.1016/j.cma.2004.09.007).
- [8] LELAND W.E., TAQQU M.S., WILLINGER W., WILSON D.V. On the self-similar nature of Ethernet traffic (extended version). *IEEE/ACM Transactions on Networking*. 1994, 2(1), pp. 1–15, doi: [10.1109/90.282603](https://doi.org/10.1109/90.282603).
- [9] LIAO W.J., BALZEN Z. LSSVM network flow prediction based on the self-adaptive genetic algorithm optimization. *Journal of Networks*. 2013, 8(2), pp. 507–512, doi: [10.4304/jnw.8.2.507-512](https://doi.org/10.4304/jnw.8.2.507-512).

- [10] MAHDAVI M., FESANGHARY M., DAMANGIR E. An improved harmony search algorithm for solving optimization problems. *Applied Mathematics and Computation*. 2007, 188(2), pp. 1567–1579, doi: [10.1016/j.amc.2006.11.033](https://doi.org/10.1016/j.amc.2006.11.033).
- [11] PARK D.C. Structure optimization of BiLinear Recurrent Neural Networks and its application to Ethernet network traffic prediction. *Information Sciences*. 2013, 237, pp. 18–28, doi: [10.1016/j.ins.2009.10.005](https://doi.org/10.1016/j.ins.2009.10.005).
- [12] RASPALL F. Efficient packet sampling for accurate traffic measurements. *Computer Networks*. 2012, 56(6), pp. 1667–1684, doi: [10.1016/j.comnet.2011.11.017](https://doi.org/10.1016/j.comnet.2011.11.017).
- [13] REN X.Y., YANG Y., ZHANG J.F., MA X.D. Parameter estimation and application of time-varying FARIMA model. *International Journal of Advancements in Computing Technology*. 2011, 3(3), pp. 89–94, doi: [10.4156/ijact.vol3.issue3.8](https://doi.org/10.4156/ijact.vol3.issue3.8).
- [14] SAYAH S., HAMOUDA A., BEKRAR A. Efficient hybrid optimization approach for emission constrained economic dispatch with nonsmooth cost curves. *International Journal of Electrical Power and Energy Systems*. 2014, 56, pp. 127–139, doi: [10.1016/j.ijepes.2013.11.001](https://doi.org/10.1016/j.ijepes.2013.11.001).
- [15] SOARES H.C., DA SILVA L., LOBAO D.C., CAETANO D.P., HUGUENIN J.A.O. Hurst exponent analysis of moving metallic surfaces. *Physica A: Statistical Mechanics and its Applications*. 2013, 292(21), pp. 5307–5312, doi: [10.1016/j.physa.2013.07.005](https://doi.org/10.1016/j.physa.2013.07.005).
- [16] TAVERNIER W., PAPADIMITRIOU D., COLLE D., PICKAVET M., DEMEESTER P. Packet loss reduction during rerouting using network traffic analysis. *Telecommunication Systems*. 2013, 52(2), pp. 861–879, doi: [10.1007/s11235-011-9581-6](https://doi.org/10.1007/s11235-011-9581-6).
- [17] WANG J.S., WANG J.K., ZENG M.H., WANG J.J. Prediction of internet traffic based on Elman neural network. In: *Proceedings of 21th Chinese Control and Decision Conference (CCDC 2009)*, Piscataway, NJ, USA. IEEE, 2009, pp. 1248–1252, doi: [10.1109/CCDC.2009.5192639](https://doi.org/10.1109/CCDC.2009.5192639).
- [18] XU S.N., ZENG B.Q. Network traffic prediction model based on auto-regressive moving average. *Journal of Networks*. 2014, 9(3), pp. 653–659, doi: [10.4304/jnw.9.3.653-659](https://doi.org/10.4304/jnw.9.3.653-659).
- [19] YADAV R.K., BALAKRISHNAN M. Comparative evaluation of ARIMA and ANFIS for modeling of wireless network traffic time series. *EURASIP Journal on Wireless Communications and Networking*. 2014, 2, pp. 1–13, doi: [10.1186/1687-1499-2014-15](https://doi.org/10.1186/1687-1499-2014-15).
- [20] YANG Z.J., YANG L.H., QING C.M. An oblique-extrema-based approach for empirical mode decomposition. *Digital signal processing*. 2010, 20(3), pp. 699–714, doi: [10.1016/j.dsp.2009.08.013](https://doi.org/10.1016/j.dsp.2009.08.013).
- [21] ZHUANG J.J., NING X.B., YANG X.D., HOU F.Z., HUO C.Y. Decrease in Hurst exponent of human gait with aging and neurodegenerative diseases. *Chinese Physics B*. 2008, 17(3), pp. 852–856, doi: [10.1088/1674-1056/17/3/021](https://doi.org/10.1088/1674-1056/17/3/021).
- [22] ZONG W.G., KIM J.H., Loganathan G.V. A new heuristic optimization algorithm: Harmony search. *Simulation*. 2001, 76(2), pp. 60–68, doi: [10.1177/003754970107600201](https://doi.org/10.1177/003754970107600201).