ARTIFICIAL SATELLITES, Vol. 50, No. 1 – 2015
DOI: 10.1515/arsa-2015-0002

EXTREME LEARNING MACHINE FOR THE PREDICTIONS OF LENGTH OF DAY

Yu Lei[1, 2, 3], Danning Zhao[1, 3], Hongbing Cai[1, 2]
[1] National Time Service Center, Chinese Academy of Sciences, Xi'an 710600, China
[2] Key Laboratory of Primary Time and Frequency Standards, Chinese Academy of Sciences, Xi'an 710600, China
[3] University of Chinese Academy of Sciences, Beijing 100049, China

ABSTRACT. This work presents short- and medium-term predictions of length of day (LOD) up to 500 days by means of extreme learning machine (ELM). The EOP C04 time-series with daily values from the International Earth Rotation and Reference Systems Service (IERS) serve as the data basis. The influences of the solid Earth and ocean tides and seasonal atmospheric variations are removed from the C04 series. The residuals are used for training of the ELM. The results of the prediction are compared with those from other prediction methods. The accuracy of the prediction is equal to or even better than that by other approaches. The most striking advantages of employing ELM instead of other algorithms are its noticeably reduced complexity and high computational efficiency.

Keywords: Earth orientation parameters (EOP); length of day (LOD); predictions; artificial neural networks (ANN); extreme learning machine (ELM)

1. INTRODUCTION

Today, daily or even subdaily time-series of the Earth orientation parameters (EOP) are available with high accuracy from the International Earth Rotation and Reference Systems Service (IERS). The EOP are essential for many researches and applications in astronomy and geodesy since they provide the time-varying transformation between the celestial and terrestrial reference frames (CRF and TRF). Advanced space-geodetic techniques, e.g., Very Long Baseline Interferometry (VLBI), Global Navigation Satellite System (GNSS) and Satellite Laser Ranging (SLR), enable determination of the EOP with high accuracy up to 5-10μs in the case of universal time (UT1-UTC) data and 50-100μas in the case of pole coordinates [Kalarus M. et al. 2010]. However, the EOP estimates cannot be published in real time due to the delay caused by computation procedures. Therefore, it is necessary to predict the EOP at least over a few days for many real-time applications including the tracking and navigation of

interplanetary spacecrafts. EOP predictions can also be valuable for geophysical studies on time scales ranging from a few hours to decades.

Regularly generated EOP predictions are published by several national and international services, e.g., the IERS Rapid Service/Prediction Center (RS/PC) [Johnson T. et al. 2005], or the EOP Service of the Institute for Astronomy and Astrophysics (IAA) in Saint Petersburg, Russia [Malkin Z., Skurikhina E. 1996]. Nevertheless, all prediction algorithms should be continually improved since the prediction errors of UT1-UTC and pole coordinates even for a few days in the future are several times greater than their observational accuracy. Out of the five EOP, UT1–UTC or its first derivative, length of day (LOD), which represents the variations in the Earth's rotation rate, are the most difficult to predict. Particularly the greatest difficulties in UT1-UTC or LOD predictions are owing to the occurrence of extremes in the LOD signal induced by the collapse of the tropical easterly winds during an EI Niňo event [Gross RS. et al. 1996]. Consequently, precise predictions of UT1-UTC or LOD are an ongoing challenge. This study focuses on LOD predictions.

Up to now, various prediction methods and techniques have been developed to improve the prediction accuracy of LOD time-series, e.g., least-squares (LS) extrapolation of the harmonic model [Niedzielski T., Kosek W. 2008], autocovariance (AC) prediction [Koesk W. et al. 1998], autoregressive (AR) prediction [Niedzielski T., Kosek W. 2008], artificial neural networks (ANN) [Schuh H. et al. 2002], [Zhang XH. et al. 2012], fuzzy inference systems (FIS) [Akyilmaz O., Kutterer H. 2004], [Akyilmaz O., Kutterer H. 2005], fuzzy-wavelet [Akyilmaz O. et al. 2011] and combined solutions [Xu XQ. et al. 2012]. A list of most contemporary methods and their comparison can be found in [Kalarus M. et al. 2010]. Most of the prediction methodologies use a combined model consisting of the deterministic part, which is either known or estimated by means of the least-squares (LS) method, and of the part to be forecasted, which can be stochastic or non-stochastic. [Schuh H. et al. 2002], [Akyilmaz O., Kutterer H. 2004], [Akyilmaz O., Kutterer H. 2005] and [Zhang XH. et al. 2012] have employed a combination of an a priori deterministic model and ANN so as to make a precise prediction of short- and long-term LOD. As a first step, parameters of an a priori model are estimated by the LS approach and the resulting residual time-series are used in ANN prediction. At the second step, the residuals forecasted by ANN are then added to the corresponding a priori model value to provide LOD for the respective day. The further discussed approach is similar to those as considered in [Schuh H. et al. 2002], [Akyilmaz O., Kutterer H. 2004], [Akyilmaz O., Kutterer H. 2005] and [Zhang XH. et al. 2012]. As usual, we first use, for LS extrapolation, an a prior model consisting of periodic effects such as the impacts of the solid Earth tides and the ocean tides on LOD and seasonal variations. Then we attempt to enhance near-term predictions by applying the extreme learning machine (ELM) to the residuals after subtracting the a prior model from actual LOD. This work concludes with a comparison of the new results and those obtained by other prediction methods and techniques.

The ELM used here will be explained in Section 2. It is an efficient learning algorithm for single-hidden layer feed-forward neural networks (SLFN) proposed by [Huang GB. et al. 2004] and [Huang GB. et al. 2006]. In the ELM algorithm, the input weights (connecting the input layer and the hidden layer) and hidden layer biases are randomly chosen, and the output weights (linking the hidden layer to the output layer) are analytically determined by employing the

Moore–Penrose (MP) generalized inverse. The ELM not only learns much faster with higher generalization performance than traditional gradient-based learning algorithms like back-propagation neural networks (BPNN), but it also avoids many difficulties faced by gradient-based solutions such as stopping criteria, local minima and the over-tuned problems. Methods from the ELM have found numerous applications in different disciplines over the past few years and offer an alternative to conventional multi-hidden layer feed-forward neural networks (MLFN).

2. EXTREME LEARNING MACHINE

For N arbitrary distinct samples $(\boldsymbol{x}_i, \boldsymbol{y}_i)$, where $\boldsymbol{x}_i = [x_{i1}, x_{i2}, \cdots, x_{in}]^{\mathrm{T}} \in \boldsymbol{R}^n$ and $\boldsymbol{y}_i = [y_{i1}, y_{i2}, \cdots, y_{im}]^{\mathrm{T}} \in \boldsymbol{R}^m$, if a standard SLFN with $\tilde{N}$ hidden neurons and active function vectors $g(x)$ can approximate these N samples with zero errors, i.e., $\sum_{j=1}^{N} \left\| \boldsymbol{o}_j - \boldsymbol{y}_j \right\| = 0$, where $\boldsymbol{o}$ is the actual outputs of the SLFN, there exist $\boldsymbol{\beta}_i$, $\boldsymbol{w}_i$ and b_i such that

$$\sum_{i=1}^{\tilde{N}} \boldsymbol{\beta}_i g(\boldsymbol{w}_i \cdot \boldsymbol{x}_j + b_i) = \boldsymbol{y}_j, \quad j = 1, 2, \cdots, N. \tag{1}$$

where $\boldsymbol{w}_i = [w_{i1}, w_{i2}, \cdots, w_{in}]^{\mathrm{T}}$ is the weight vector linking the ith hidden node to the input nodes, $\boldsymbol{\beta}_i = [\beta_{i1}, \beta_{i2}, \cdots, \beta_{im}]^{\mathrm{T}}$ is the weight vector connecting the ith hidden node and the output nodes, and b_i is the threshold of the ith hidden neuron. $\boldsymbol{w}_i \cdot \boldsymbol{x}_j$ denotes the inner product of $\boldsymbol{w}_i$ and $\boldsymbol{x}_j$.

The above N equations can be written compactly as

$$\boldsymbol{H}\boldsymbol{\beta} = \boldsymbol{Y}. \tag{2}$$

where

$$\boldsymbol{H}(\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots, \boldsymbol{w}_{\tilde{N}}, b_1, b_2, \cdots, b_{\tilde{N}}, \boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_N) = \begin{bmatrix} g(\boldsymbol{w}_1 \cdot \boldsymbol{x}_1 + b_1) & \cdots & g(\boldsymbol{w}_{\tilde{N}} \cdot \boldsymbol{x}_1 + b_{\tilde{N}}) \\ \vdots & \cdots & \vdots \\ g(\boldsymbol{w}_1 \cdot \boldsymbol{x}_N + b_1) & \cdots & g(\boldsymbol{w}_{\tilde{N}} \cdot \boldsymbol{x}_N + b_{\tilde{N}}) \end{bmatrix}_{N \times \tilde{N}}, \tag{3}$$

$$\boldsymbol{\beta} = \begin{bmatrix} \boldsymbol{\beta}_1^{\mathrm{T}} \\ \vdots \\ \boldsymbol{\beta}_{\tilde{N}}^{\mathrm{T}} \end{bmatrix}_{\tilde{N}\times m} \quad \text{and} \quad \boldsymbol{Y} = \begin{bmatrix} \boldsymbol{y}_1^{\mathrm{T}} \\ \vdots \\ \boldsymbol{y}_N^{\mathrm{T}} \end{bmatrix}_{N\times m} . \tag{4}$$

$\boldsymbol{H}$ is called the hidden layer output matrix of the neural network; the i th column of $\boldsymbol{H}$ is the i th hidden neuron output with respect to inputs $\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_N$.

In the ELM solution, the input weights and hidden layer biases are randomly assigned instead of tuned. Thereby, the estimates of the output weights are as simple as finding the LS solution to the given linear system. The minimum norm LS solution to the linear system (1) is

$$\hat{\boldsymbol{\beta}} = \boldsymbol{H}^{\dagger}\boldsymbol{Y}. \tag{5}$$

where $\hat{\boldsymbol{\beta}}$ is used as the estimated value of $\boldsymbol{\beta}$ and $\boldsymbol{H}^{\dagger}$ is the Moore–Penrose generalized inverse of matrix $\boldsymbol{H}$. The minimum norm LS solution is unique and has the smallest norm among all the LS solutions.

The solving process of the ELM algorithm can be summarized as follows.

(1) Give a training dataset $(\boldsymbol{x}_i, \boldsymbol{y}_i)$, $i = 1, 2, \cdots, N$ and $\tilde{N}$ hidden nodes.

(2) Randomly assign threshold b_j and input weights $\boldsymbol{w}_j$, $j = 1, 2, \cdots, \tilde{N}$.

(3) Compute the hidden layer output matrix $\boldsymbol{H}$.

(4) Estimate the output weights $\hat{\boldsymbol{\beta}} = \boldsymbol{H}^{\dagger}\boldsymbol{Y}$.

3. METHODOLOGY

Daily time-series of LOD used in this work are collected from the IERS EOP 05 C04 series. As discussed in [Schuh H. et al. 2002], [Akyilmaz O., Kutterer H. 2004], [Akyilmaz O., Kutterer H. 2005] and [Zhang XH. et al. 2012], the original LOD time-series should be reduced before training the neural network in order to avoid the error coming from the extrapolation problem. In this paper, a simple reduction procedure has been applied to the daily values of observed LOD time-series. After reduction of the original time-series by removing the a prior model, training patterns are formed out of the residuals. These patterns are employed for training the ELM network. The subsequently forecasted residuals are then added to the a prior model so as to gain the predicted values of LOD.

In the following section the generation of the a prior model, formation of the training patterns and building of the ELM model are described in detail.

3.1. GENERATION OF THE A PRIOR MODEL

The model of LOD contains several well-known components, such as the effects of zonal Earth tides with periods from 5 days up to 18.6 years, diurnal and semi-diurnal variations due to the ocean tides [Petit G., Luzum B. 2004], annual and semi-annual oscillations. In accordance with the above-described deterministic components of LOD data, we fit the function

$$f_{\mathrm{LOD}}(t) = a_0 + a_1 t + A_a \sin(\omega_a t + \Phi_a) + A_{sa} \sin(\omega_{sa} t + \Phi_{sa}) + \text{tidal terms}. \quad (6)$$

where $\omega_a = 2\pi / 365.24$ and $\omega_{sa} = 2\pi / 182.62$, and bias ($a_0$) and drift ($a_1$) of the linear term, amplitudes (A_a, A_{sa}) and phases (Φ_a, Φ_{sa}) of the annual and semi-annual oscillations have to be estimated from the observations.

The estimated deterministic model is subsequently used for two purposes: (1) to forecast the deterministic components of the signal (extrapolation) and (2) to obtain stochastic residuals (the difference between the data themselves and the deterministic model). The stepwise reduction of LOD in the time domain using the IERS EOP 05 C04 series is shown in Fig. 1. The amplitude of the residuals (plot (e)) is small in contrast with that of the original time-series (plot (a)). This indicates that the deterministic model represents the actual LOD time-series rather well. The derived stochastic residuals are used for training the neural network.

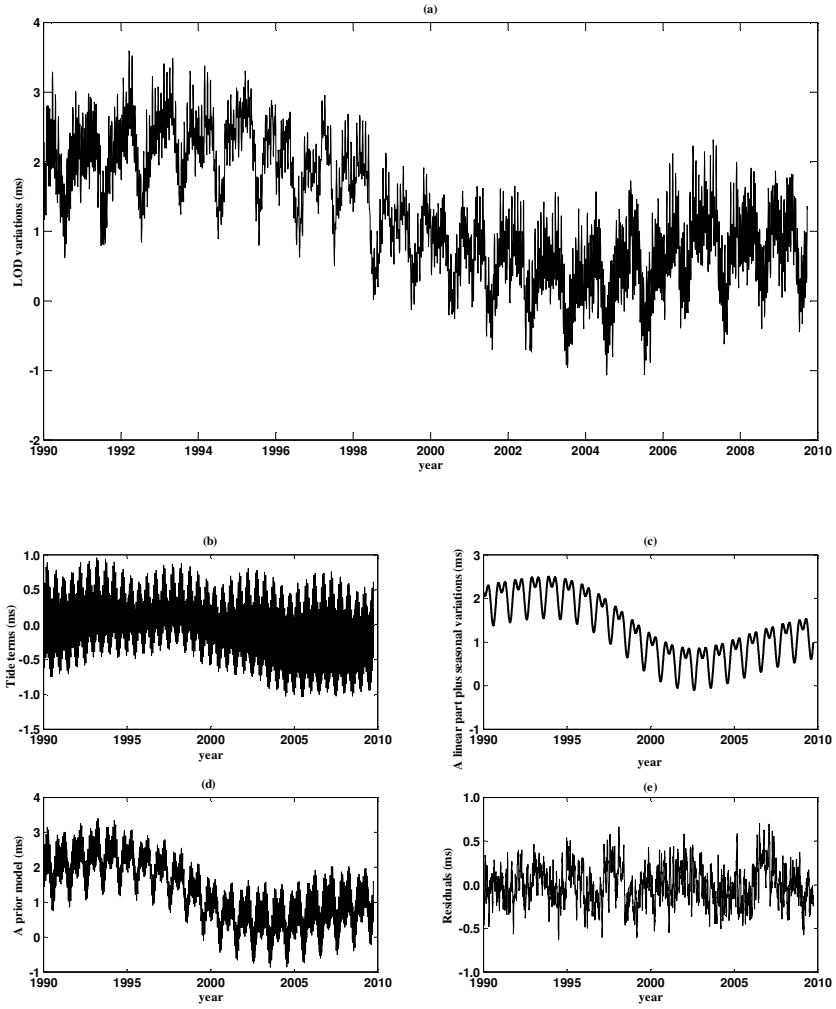

Fig. 1. The observed LOD (a), the solid Earth and ocean tide terms (b), a linear trend plus the seasonal variations including annual and semi-annual oscillations (c), the a prior model of LOD (d), and the stochastic residuals (e)

3.2. FORMATION OF THE TRAINING PATTERNS

After the LOD time-series have been reduced, the training patterns are generated. A first possibility is to utilize the variable time t as the only input for feeding the network. However, [Schuh H. et al. 2002] had shown that using the time t as the only input for feeding the network does not result in a precise prediction in the case of ANN. It turns out that the closer the observational data is near to the day to be forecasted, the greater impact on the prediction is, especially for LOD data which vary rapidly and unpredictably in time. Hence a more sophisticated strategy is to utilize previous values as inputs of the network and future values as outputs. In this contribution, a multitude of pattern pairs are formed as follows.

$$\boldsymbol{X} = \left[\boldsymbol{x}_1 \; \boldsymbol{x}_2 \cdots \boldsymbol{x}_N \; \right]^{\mathrm{T}} = \begin{bmatrix} \xi(1), & \xi(2), & \xi(3), & \xi(4), & \xi(5) \\ \xi(2), & \xi(3), & \xi(4), & \xi(5), & \xi(6) \\ & & \vdots & & \\ \xi(L-5), & \xi(L-4), & \xi(L-3), & \xi(L-2), & \xi(L-1) \end{bmatrix}. \tag{7}$$

$$\boldsymbol{Y} = \left[y_1 \; y_2 \cdots y_N \; \right]^{\mathrm{T}} = \begin{bmatrix} \xi(6) \\ \xi(7) \\ \vdots \\ \xi(L) \end{bmatrix}. \tag{8}$$

where $\boldsymbol{X}$ and $\boldsymbol{Y}$ are the pattern matrices, $\xi(t)$, $t = 1, 2, \cdots, L$, is the residual data, and the values of the residual time-series of the last five days are selected as inputs and the day to be forecasted is selected as output. This strategy is based both on theoretical considerations concerning the quasi-periodic and irregular variations in LOD residuals and on practical trials.

After the neural network has been trained, the new input vector for the prediction of the first day into the future is formed as

$$\boldsymbol{x} = \left[\xi(L-4), \xi(L-3), \xi(L-2), \xi(L-1), \xi(L) \right]. \tag{9}$$

and then are put into the network. The output will be $y = \hat{\xi}(L+1)$, where $\hat{\xi}(L+1)$ is the one-step-ahead prediction value. After the first day, the forecasted values are used as inputs in the already existing model composed for the first day's prediction to compute the corresponding prediction values for the days 2, 3, $\cdots$. No individual model has been composed for the prediction of each day after the first day since the prediction errors increase rapidly with a linear trend in the case of updating models. The inputs and output for the predictions of the 2nd, 3rd, … days into the future are given below.

$$\begin{cases} \boldsymbol{x} = \left[\xi(L-3), \xi(L-2), \xi(L-1), \;\; \xi(L), \;\; \hat{\xi}(L+1) \right], \; y = \hat{\xi}(L+2), \; d = 2 \\ \boldsymbol{x} = \left[\xi(L-2), \xi(L-1), \xi(L), \;\; \hat{\xi}(L+1), \;\; \hat{\xi}(L+2) \right], \; y = \hat{\xi}(L+3), \; d = 3 \\ \qquad\qquad \vdots \\ \boldsymbol{x} = \left[\xi(L), \;\; \hat{\xi}(L+1), \hat{\xi}(L+2), \hat{\xi}(L+3), \hat{\xi}(L+4) \right], \; y = \hat{\xi}(L+5), \; d = 5 \\ \boldsymbol{x} = \left[\hat{\xi}(L+1), \hat{\xi}(L+2), \hat{\xi}(L+3), \hat{\xi}(L+4), \hat{\xi}(L+5) \right], \; y = \hat{\xi}(L+6), \; d = 6 \\ \qquad\qquad \vdots \end{cases}. \tag{10}$$

where d represents the prediction horizon.

3.3. MODEL BUILDING

Next a neural network has to be designed and then built to provide predictions of the LOD residuals. In practice, the optimum network configuration relies on the composed training patterns, number of hidden neurons and type of activation function. The three factors have to be considered while trying to find the best network configuration. In this study, the patterns as

composed in the previous paragraph are utilized to train the network, and the sigmoid function is employed as activation function of ELM, that is, $g(x)=1/(1+e^{-x})$.

Actually, the optimum configuration solely depends on the number of hidden neurons, since the training patterns and type of activation function can be chosen in advance. In this work, in order to find the optimum number of hidden nodes and thus improve the generalization performance of ELM, the given training dataset are divided into two parts, one for training and the other for validation. Hidden neurons are generated k times. Among k ELM predictors, the predictor which yields the minimum validation errors is finally selected. A detailed description of the improved ELM algorithm is explained as follows.

(1) Give the training dataset $\aleph_{\text{training}}$ and validation dataset $\aleph_{\text{validation}}$.

(2) For $\tilde{N}=1:k$

(a) Randomly designate parameters of hidden neurons $(b_j^{\tilde{N}}, \boldsymbol{w}_j^{\tilde{N}})$, $j=1, 2, \cdots, \tilde{N}$.

(b) Compute the output matrix of the hidden layer on the training dataset $\aleph_{\text{training}}$: $\boldsymbol{H}_{\text{training}}^{\tilde{N}}$.

(c) Compute the output weights $\hat{\boldsymbol{\beta}}^{\tilde{N}}=\boldsymbol{H}_{\text{training}}^{\tilde{N}\dagger}\boldsymbol{Y}_{\text{training}}$, where $\boldsymbol{Y}_{\text{training}}$ is the target matrix of the training dataset $\aleph_{\text{training}}$.

(d) Compute the validation errors $\boldsymbol{E}^{\tilde{N}}=\boldsymbol{H}_{\text{validation}}^{\tilde{N}}\hat{\boldsymbol{\beta}}^{\tilde{N}}-\boldsymbol{Y}_{\text{validation}}$, where $\boldsymbol{Y}_{\text{validation}}$ and $\boldsymbol{H}_{\text{validation}}^{\tilde{N}}$ is the target output matrix of the validation dataset $\aleph_{\text{validation}}$ and the hidden layer's output matrix on the dataset $\aleph_{\text{validation}}$, respectively.

end for.

(3) Let $\tilde{N}^*=\left\{\tilde{N}\left|\min_{1\le\tilde{N}\le k}\left\|\boldsymbol{E}^{\tilde{N}}\right\|\right.\right\}$. Set $\tilde{N}^*$ and ($b_j^{\tilde{N}}$, $\boldsymbol{w}_j^{\tilde{N}}$) as the optimum number and parameters of hidden neurons, respectively.

In this study, k is set as 50.

4. PREDICTION RESULTS AND COMPARISON WITH OTHER METHODS

Daily time-series of the IERS EOP 05 C04 series, which span the time interval from 1/1/1990 to 31/12/2001, are used for modeling and evaluation. The whole dataset is split up into two parts in such a way that the time-series from 1/1/1990 to 31/12/1999 are employed for network training and the remaining part between 1/1/2000 and 31/12/2001 for the model assessment. The training patterns as described in the previous paragraph have been composed, and then

used to train the network. Note that it is very important to divide the training dataset into two segments during the network training. Herein the chronologically first 90% shape the training dataset and the remaining 10% represent the validation dataset. After the network has been trained, the well-trained network model is used to produce a predicted set of residuals for the future 1~360 days. Then the resulting forecasted value of the residuals for any particular day is added to the corresponding value of the a prior model to obtain the actual prediction value of LOD. The results of the ELM prediction are compared with those of BPNN [Schuh H. et al. 2002], modified BPNN [Zhang XH. et al. 2012], general regression neural networks (GRNN) [Zhang XH. et al. 2012] and FIS [Akyilmaz O., Kutterer H. 2004], [Akyilmaz O., Kutterer H. 2005] which had performed equally or even better than former methods (Table 1). The error measure is calculated according to [Schuh H. et al. 2002] as

$$RMS_d = \sqrt{\frac{1}{p}\sum_{i=1}^{p}(\hat{l}_d^i - l_d^i)^2}\ . \tag{11}$$

where $\hat{l}_d^i$ is the forecasted value of the ELM network for day d, l_d^i is the actual value of the IERS C04 series, and p is the number of predictions made for the respective day. Approximately 365 predictions starting at different days are made for each day to calculate the RMS errors, i.e. $p = 365$.

Table 1. Comparison of ELM, BPNN, modified BPNN, GRNN and FIS RMS prediction errors (in units of ms).

Prediction day	ELM	BPNN	Modified BPNN	GRNN	FIS
1	0.027	0.019	0.027	0.037	0.017
2	0.057	0.049	0.073	0.074	0.045
3	0.078	0.074	0.093	0.097	0.067
4	0.098	0.097	0.110	0.117	0.088
5	0.110	0.121	0.131	0.134	0.115
6	0.121	0.142	0.148	0.151	0.139
7	0.131	0.159	0.162	0.164	0.153
8	0.142	0.174	0.170	0.174	0.170
9	0.151	0.184	0.176	0.179	0.182
10	0.157	0.193	0.185	0.187	0.188
15	0.179	0.246	0.221	0.204	0.251
20	0.192	0.251	0.217	0.210	0.259
25	0.198	0.249	0.215	0.211	0.267
30	0.201	0.245	0.219	0.217	0.275
60	0.222	0.292	0.219	0.222	—
90	0.242	0.306	0.231	0.226	—
120	0.238	0.314	0.229	0.226	—
150	0.212	0.330	0.237	0.233	—
180	0.217	0.361	0.234	0.234	0.296
210	0.229	0.397	0.241	0.236	—
240	0.232	0.377	0.236	0.236	—
270	0.228	0.386	0.231	0.240	0.313
300	0.227	0.402	0.249	0.247	—
330	0.224	0.372	0.262	0.254	—
360	0.232	0.347	0.245	0.250	0.303

In order to make the comparison illustrative, the RMS prediction errors obtained by different machine learning (ML) methods are shown in Fig. 2. As can be seen in Fig. 2 and Table 1, the RMS errors of the ELM prediction are smaller than that of other ML methods. Note that the RMS errors given there are obtained by testing the prediction algorithms over different prediction spans. Despite utilizing the same equation for computing the RMS error and the same LOD reference series (C04 of the IERS), this might have affected the results of the other authors. Thereby, it is not directly comparable with the other approaches. A final picture of the prediction performance of different methods could only be attained by a kind of contest where prediction period and evaluation strategy are clearly specified in advance. Fortunately, the EOP prediction comparison campaign (EOP PCC) lasting from October 2005 until February 2008 provided an opportunity to compare the performance of different prediction methods and techniques directly. Therefore, we have carried out a comparison with the EOP PCC for the purpose of evaluating the prediction accuracy of the proposed method. The LOD time-series spanning the time interval from 30/9/1995 until 30/9/2005 are selected as the data base to forecast the LOD values for the future 1～500 days during the period from 1/10/2005 to 28/2/2008 (the same prediction period as that of the EOP PCC). A graphical comparison of the ELM results with the EOP PCC regarding the mean-absolute-error (MAE) error measure is given in Fig. 3～5. The MAE error measure is calculated in agreement with the following equation.

$$MAE_d = \frac{1}{p}\sum_{i=1}^{p}\left|\hat{l}_d^i - l_d^i\right|. \tag{12}$$

A list of participants who supported the LOD predictions during the EOP PCC can be found in [Kalarus M. et al. 2010]. What can be said with the information available from the comparison is that the MAE errors of the ultra short-term (up to 10 days) predictions by the ELM are larger than those by Kalman filter with AAM forecasts (top. 1) and LS+AR (top. 2), but remarkably smaller than those by other prediction methods and techniques. For the short-term (up to 30 days) predictions the accuracy of the ELM prediction is inferior to Kalman filter with AAM forecasts, which is the best presently available prediction method found in [Gross RS. et al. 1998]. In Fig. 5, it can be seen that the developed strategy can offer long-term predictions which are better than those of other methods and techniques, indicating that the presented algorithm has very good extrapolation performance.

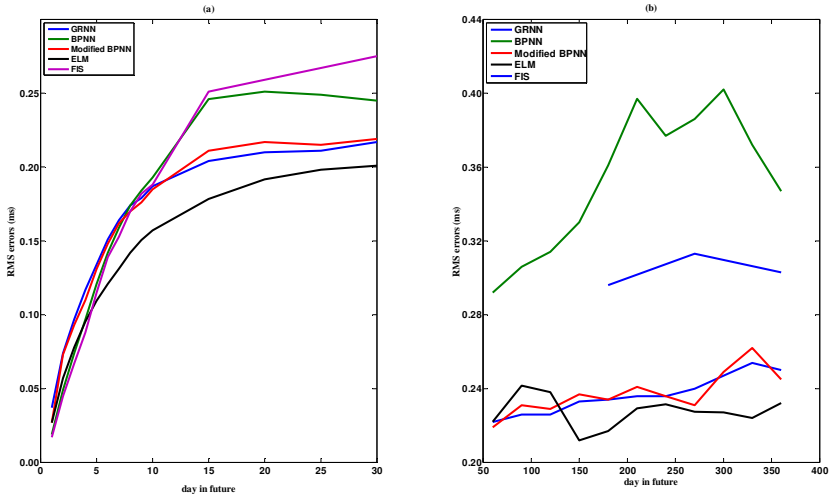

Fig. 2. Comparison of the RMS prediction errors of different ML methods: (a) the short-term (up to 30 days) and medium-term (up to 360 days) predictions

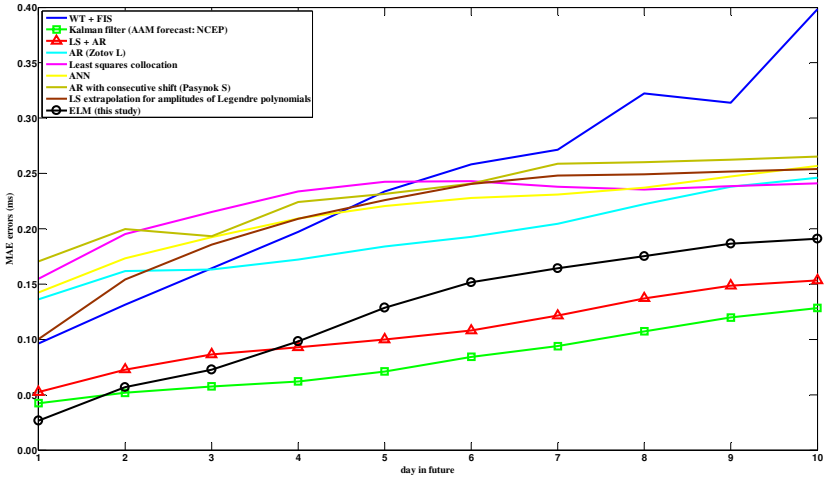

Fig. 3. Comparison of the MAE errors of ultra short-term (up to 10 days) predictions by the ELM and EOP PCC

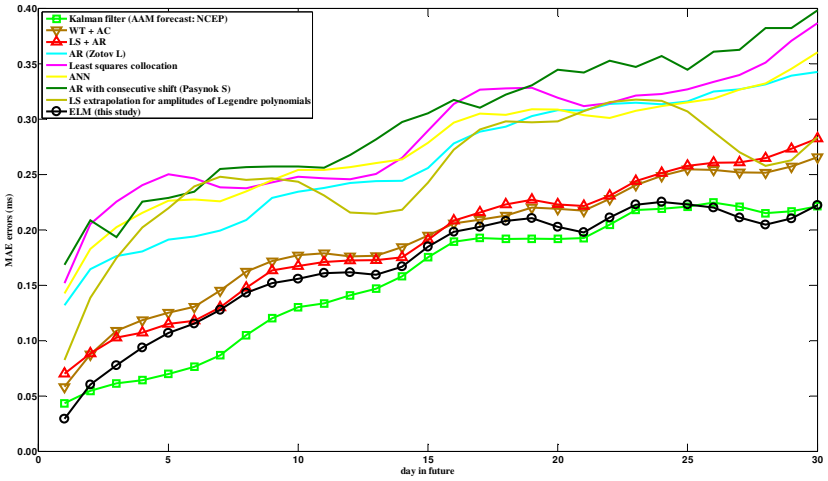

Fig. 4. Comparison of the MAE errors of short-term (up to 30 days) predictions by the ELM and EOP PCC

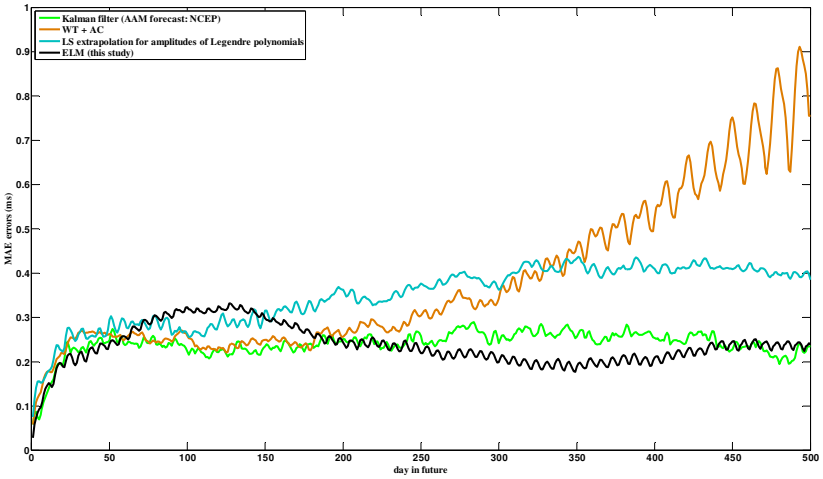

Fig. 5. Comparison of the MAE errors of medium-term (up to 500 days) predictions by the ELM and EOP PCC

5. CONCLUSIONS

The comparison of ELM-derived results with those from other approaches clearly demonstrates that ELM is a promising tool to predict LOD. Precise predictions for both the short- and medium-term are possible by employing a same network for each day of prediction. Although the predicted values are used as inputs for the next days to be predicted after the first day, the prediction errors do not increase rapidly. For long-term predictions, e.g., several and more years into the future, more input variables could be utilized in the ELM model because computer run time becomes a minor restriction. Nevertheless, this case is not considered in present work.

The comparison among different ML algorithms including BPNN, modified BPNN, GRNN, FIS and ELM regarding complexity and the resulting predictions is illustrative. Traditional ANN algorithms such as BPNN, GRNN provide a very good prediction but it is difficult to find the optimum network configuration. In addition, training of traditional ANN algorithms is time-consuming. According to our experience, training of those ANN algorithms takes much longer than that of ELM: training of ELM only takes several minutes, while training of two-hidden layer neural networks with five inputs may take hours or even days on the same computer used for ELM. As a future work, values of atmospheric and oceanic angular momentum (AAM and OAM) may be added into the ELM network so as to improve the prediction quality.

A promising study can be the use of artificial intelligence (AI) algorithms for the determination of optimal number of hidden neurons, such as genetic algorithms (GA), particle swarm optimization (PSO). Additionally training patterns for different prediction days should be formed and compared in order to optimize the network solution.

ACKNOWLEDGEMENTS

The authors are grateful to the IERS for providing the LOD data, to M. Kalarus for providing the data of the EOP PCC, and to the anonymous reviewers for their remarks and advice on modifying the manuscript. This work is supported by the West Light Foundation of Chinese Academy of Sciences.

REFERENCES

Akyilmaz O., Kutterer H. (2004) Prediction of Earth rotation parameters by fuzzy inference systems. Journal of Geodesy, Vol. 78, No. 1-2, 2004, pp. 82-93.

Akyilmaz O., Kutterer H., Shum CK. and Ayan T. (2011) Fuzzy-wavelet based prediction of Earth rotation parameters. Applied Soft Computing, Vol. 11, No. 1, 2011, pp. 837-841.

Gross RS., Marcus SL., Eubanks TM., Dickey JO. and Keppenne CL. (1996) Detection of an ENSO signal in seasonal length-of-day variations. Geophysical Research Letters, Vol. 23, No. 23, 1996, pp. 3373-3376.

Gross RS., Eubanks TM., Steppe JA., Freedman AP., Dickey JO. and Runge TF. (1998) A Kalman-filter-based approach to combining independent Earth-orientation series. Journal of Geodesy, Vol. 72, No. 4, 1998, pp. 215-235.

Huang GB., Zhu QY. and Siew CK. (2004) Extreme learning machine: a new learning scheme of feedforward neural networks. Proceedings of 2004 IEEE International Joint Conference on Neural Networks, Budapest, Hungary, pp. 985-990.

Huang GB., Zhu QY. and Siew CK. (2006) Extreme learning machine: theory and applications, Neurocomputing, Vol. 70, No. 1-3, 2006, pp. 489-501.

Johnson T., Luzum BJ., Ray JR. (2005) Improved near-term Earth rotation predictions using atmospheric angular momentum analysis and forecasts. Journal of Geodynamics, Vol. 39, No. 3, 2005, pp. 209-221.

Kalarus M., Schuh H., Kosek W., et al. (2010) Achievements of the Earth orientation parameters prediction comparison campaign. Journal of Geodesy, Vol. 84, No. 10, 2010, pp. 587-596.

Kosek W., McCarthy DD. and Luzum BJ. (1998) Possible improvement of Earth orientation forecast using autocovariance prediction procedures. Journal of Geodesy, Vol. 72, No. 4, 1998, pp. 189-199.

Malkin Z., Skurikhina E. (1996) On prediction of EOP, Communications of IAA, No. 93.

Niedzielski T., Kosek W. (2008) Prediction of UT1–UTC, LOD and AAM by combination of least-squares and multivariate stochastic methods. Journal of Geodesy, Vol. 82, No. 2, 2008, pp. 83-92.Petit G., Luzum B. (2010) IERS Conventions (2010). Verlag des Bundesamts für Kartographie und Geodasie, Frankfurt am Main, pp. 123-131.

Schuh H., Ulrich M., Egger D., Müller J. and Schwegmann W. (2002) Prediction of Earth orientation parameters by artificial neural networks. Journal of Geodesy, Vol. 76, No. 5, 2002, pp. 247-258.

Xu XQ., Zotov L. and Zhou YH. (2012) Combined prediction of Earth orientation parameters. Chinese Satellite Navigation Conference (CSNC) 2012 Proceedings, Lecture Notes in Electrical Engineering 160, pp. 361-369.

Zhang XH., Wang QJ., Zhu JJ. and Zhang H. (2012) Application of general regression neural network to the prediction of LOD change. Chinese Astronomy and Astrophysics, Vol. 36, No. 1, 2012, pp. 86-96.

Received: 2014-11-14,
Reviewed: 2015-02-09,
Accepted: 2015-02-27.