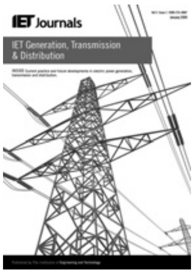
www.ietdl.org

Published in IET Generation, Transmission & Distribution
Received on 18th May 2010
Revised on 17th August 2010
doi: 10.1049/iet-gtd.2010.0355



ISSN 1751-8687

Real-time transient stability assessment model using extreme learning machine

Y. Xu[1] *Z.Y. Dong*[1] *K. Meng*[1] *R. Zhang*[1] *K.P. Wong*[1,2]

[1]Department of Electrical Engineering, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong
[2]School of Electrical, Electronic, and Computer Engineering, University of Western Australia, Perth, Australia
E-mail: zydong@ieee.org

Abstract: In recent years, computational intelligence and machine learning techniques have gained popularity to facilitate very fast dynamic security assessment for earlier detection of the risk of blackouts. However, many of the current state-of-the-art models usually suffer from excessive training time and complex parameters tuning problems, leading to inefficiency for real-time implementation and on-line model updating. In this study, a new transient stability assessment model using the increasingly prevalent extreme learning machine theory is developed. It has significantly improved the learning speed and can enable effective on-line updating. The proposed model is examined on the New England 39-bus test system, and compared with some state-of-the-art methods in terms of computation time and prediction accuracy. The simulation results show that the proposed model possesses significant superior computation speed and competitively high accuracy.

1 Introduction

Modern power systems are being pushed to operate near their security limits by the growing load demand and inadequate infrastructure investments in many nations. At the same time, the undergoing deregulation of electric utilities and large amount penetration of renewable energy have introduced increasing uncertainties and complexities to the systems. As a result, modern power systems become more vulnerable to disturbances that could trigger static and/or dynamic insecurity sequence developing into cascading failures and/or blackouts [1].

In the power system operation phase, one of the effective measures to reduce the risk of blackout and/or cascading failure is to perform real-time security assessment, through which system operators will have sufficient time to activate preventive or corrective controls to ensure system security in case of dangerous disturbances. In terms of dynamic security assessment (DSA), conventional methods like time-domain simulation, although accurate and flexible, normally entail a computationally expensive calculation procedure, making it difficult to offer timely situational awareness of the risk of dynamic insecurity.

In recent years, computational intelligence and machine learning techniques have been widely applied as independent or supplementary tools to facilitate very fast DSA because of their strong non-linear modelling capability [2–10]. With these tools, the required computation time to obtain system stability information can be significantly reduced by diminishing the needs for performing time-domain simulations. Rather, the DSA model is trained by a large data set representing the dynamic security characteristics of a power system, called knowledge base (KB), and can then be applied on-line to predict the security status at a fast speed. In the literature, neural network (NN) [4, 5, 11, 12], decision tree (DT) [6–8] and supporting vector machine (SVM) [9, 10] are among the most frequently employed techniques.

However, it has also been observed that the current state-of-the-art techniques usually suffer from excessive training time and complex parameters tuning problems when a large-sized KB is encountered, making the developed DSA tool difficult to be on-line updated using most recent operating information. As a modern power system is increasingly large, resulting into a growing larger KB, there is a pressing demand for more computationally efficient and high-accuracy DSA models that can allow more effective real-time implementation and on-line updating.

More recently, a new learning algorithm for a single hidden layer feed-forward neural network (SLF-NN) was proposed, called extreme learning machine (ELM) [13], and has shown much faster learning speed and better generalisation performance over conventional algorithms on a number of benchmark problems and engineering applications from regression and classification areas [13–16]. In this paper, ELM theory is presented and an ELM-based transient stability assessment (TSA) model is proposed. To enhance the performance of the new TSA model, a computationally efficient feature selection approach and KB generation scheme are also presented. The proposed model is examined on the New England 39-bus system, and compared with other state-of-the-art approaches in terms of computation time and prediction accuracy. Simulation results show that the new model possesses significant superior computation speed and comparable accuracy, and it therefore will enable effective on-line updating and more efficient real-time application.

The paper is organised as follows: Section 2 introduces ELM theory and proposes the ELM-based TSA model. Section 3 describes the KB generation scheme, and Section 4 presents the feature selection method. In Section 5, a case study on New-England-39-bus system is conducted. Finally, the conclusion of the paper is given in Section 6.

2 ELM for real-time TSA

ELM was proposed by Huang [13] as a new learning scheme for SLF-NNs which can overcome the insufficiently fast learning speed of the conventional learning algorithms. The basic principle of ELM theory is that the parameters of hidden nodes (the input weights and biases for additive hidden nodes or kernel parameters) need not to be traditionally tuned by time-consuming algorithms, rather, one can randomly assign the hidden nodes parameters when the activation functions in the hidden layer are infinitely differentiable and analytically determine the output weights (linking the hidden layer to the output layer) through simple generalised inverse operation of hidden layer output matrices [13].

2.1 ELM theory

For N arbitrary distinct samples $\aleph_N = \{(\boldsymbol{x}_i, t_i)|x_i \in R^n, t_i \in R^m\}_{i=1}^N$, where $\boldsymbol{x}_i$ is the $n \times 1$ input feature vector and $\boldsymbol{t}_i$ is a $m \times 1$ target vector. A standard SLN-NN which has Ñ hidden nodes and activation function $\partial(x)$ can be mathematically modelled as

$$\sum_{i=1}^{\tilde{N}} \beta_i \vartheta_i(\boldsymbol{w}_i, b_i, \boldsymbol{x}_j) = \boldsymbol{o}_j, \quad j = 1, \ldots, N \qquad (1)$$

where $\boldsymbol{w}_i = [w_{i1}, w_{i2}, K, w_{in}]^{\mathrm{T}}$ is the weight vector connecting the ith hidden node and the input nodes, is the weight vector connecting the ith hidden nod and the output nodes and b_i is $\beta_i = [\beta_{i1}, \beta_{i2}, K, \beta_{in}]^{\mathrm{T}}$ the threshold of the ith hidden node. Here $\boldsymbol{w}_i$ and $\boldsymbol{x}_j$ denote the inner products of $\boldsymbol{w}_i$ and $\boldsymbol{x}_j$. The structure of an SLF-NN is shown in Fig. 1.

If the SLF-NNs can approximate these N samples with zero error, we will have $\sum_{j=1}^{N}\|o_j - t_j\| = 0$, that is, there exist β_i, b_i, $\boldsymbol{w}_i$ such that $\sum_{i=1}^{\tilde{N}} \beta_i \vartheta_i(\boldsymbol{w}_i, b_i, \boldsymbol{x}_j) = t_j$, $j = 1, \ldots, N$.

The above N equations can be written compactly as

$$\boldsymbol{H}\beta = \boldsymbol{T} \qquad (2)$$

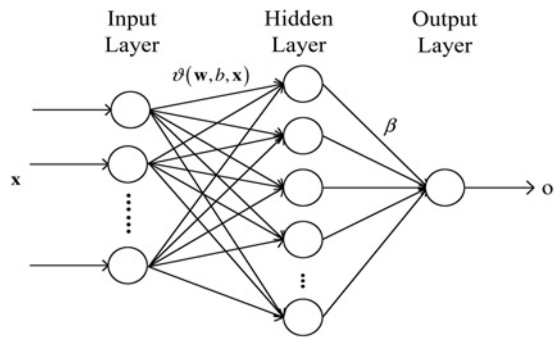


Fig. 1 *Structure of an SLF-NN*

where

$$\boldsymbol{H}(w_1, \ldots, w_{\tilde{N}}, b_1, \ldots, b_{\tilde{N}}, x_1, \ldots, x_N) = \begin{bmatrix} \vartheta(\boldsymbol{w}_i, b_i, \boldsymbol{x}_1) & & \vartheta(\boldsymbol{w}_{\tilde{N}}, b_{\tilde{N}}, \boldsymbol{x}_1) \\ \vdots & \cdots & \vdots \\ \vartheta(\boldsymbol{w}_i, b_i, \boldsymbol{x}_N) & & \vartheta(\boldsymbol{w}_{\tilde{N}}, b_{\tilde{N}}, \boldsymbol{x}_N) \end{bmatrix}_{N \times \tilde{N}} \qquad (3)$$

$$\beta = \begin{bmatrix} \beta_1^{\mathrm{T}} \\ \vdots \\ \beta_{\tilde{N}}^{\mathrm{T}} \end{bmatrix}_{\tilde{N} \times m} \quad \text{and} \quad \boldsymbol{T} = \begin{bmatrix} t_1^{\mathrm{T}} \\ \vdots \\ t_N^{\mathrm{T}} \end{bmatrix}_{N \times m} \qquad (4)$$

As named by Huang *et al.* [13], $\boldsymbol{H}$ is called the hidden layer output matrix of the NN; the ith column of $\boldsymbol{H}$ is the ith hidden node output with respect to inputs x_1, x_2, K and x_N. It is proven by Huang [13] that the input weight vectors $\boldsymbol{w}_i$ and the hidden biases b_i are in fact not necessarily tuned and the matrix $\boldsymbol{H}$ can actually remain unchanged once random values have been assigned to these parameters in the beginning of learning:

1. If the activation function ∂ is infinitely differentiable, when the number of hidden neurons is equal to the number of distinct training samples, that is, $\tilde{N} = N$, one can randomly assign the parameters of hidden nodes ($\boldsymbol{w}_i$ and b_i or kernel parameters) and based on this one can analytically calculate the output weights β by simply inverting $\boldsymbol{H}$, and therefore SLFNs can approximate $\aleph_N$ with zero error.
2. For $\tilde{N} \ll N$, which is the common case in the real world, $\boldsymbol{H}$ will be a non-square matrix and there may not exist β_i, b_i, $\boldsymbol{w}_i$. However, one specific set of $\tilde{\beta}_i$, $\tilde{b}_i$, $\tilde{\boldsymbol{w}}_i$ could be found such that

$$\|\boldsymbol{H}(\tilde{\boldsymbol{w}}_1, \ldots, \tilde{\boldsymbol{w}}_{\tilde{N}}, \hat{b}_1, \ldots, \hat{b}_{\tilde{N}})\tilde{\beta} - \boldsymbol{T}\| = \min_{\beta_i, b_i, \boldsymbol{w}_i} \|\boldsymbol{H}(\boldsymbol{w}_1, \ldots, \boldsymbol{w}_{\tilde{N}}, b_1, \ldots, b_{\tilde{N}})\beta - \boldsymbol{T}\| \qquad (5)$$

which is equivalent to minimising the cost function

$$E = \sum_{j=1}^{N}\left(\sum_{i=1}^{\tilde{N}} \beta_i \vartheta_i(\boldsymbol{w}_i, b_i, \boldsymbol{x}_j) - t_j\right)^2 \qquad (6)$$

In this case, it is proven in [13] that, for fixed $\boldsymbol{w}_i$ and b_i or kernel parameters, (2) becomes a linear system, and the output weights β can be estimated as

$$\hat{\beta} = \boldsymbol{H}^{\dagger}\boldsymbol{T} \qquad (7)$$

where $H^{\dagger}$ is the Moore–Penrose generalised inverse of matrix $\boldsymbol{H}$. Therefore one can still randomly assign the parameters $\boldsymbol{w}_i$ and b_i of the hidden nodes and calculate the output weights $\hat{\beta}$ by (7) to give a small non-zero training error $\varepsilon > 0$. There are several ways to calculate the Moore–Penrose generalised inverse of a matrix, including the orthogonal projection method, orthogonalisation method, iterative method, and singular value decomposition (SVD) etc.

It can be seen that through such learning schemes, the time-costly training is instead of a single calculation step, and as analysed in [13], it can still ensure prediction accuracy and may tend to reach better generalisation performance. In [15], the performance of ELM was compared with other algorithms, including SVM and conventional back-propagation. The results show that ELM possesses a superior performance in relative terms.

2.2 ELM-based TSA model

The general framework of computational intelligence-based DSA models has traditionally been as off-line training and on-line application, wherein the kernel part is learning algorithms. As a matter of fact, many state-of-the-art techniques usually suffer from excessive training time and complex parameter tuning problems in front of a large-sized KB, which is always encountered in today's bulk power systems.

Also, it is widely known that for supervised learning systems, the more prior knowledge they learn, the better performance they can render. However, the KB has been traditionally generated based on the forecasting of prospective operating conditions, whereas today's power system are often highly unpredictable in the context of restructuring and increasing penetration of renewable energy (especially wind power); this can make the off-line-generated KB insufficient for accurate real-time DSA. In [6], the authors proposed a periodic checking scheme to partially utilise the on-line acquired information to avoid missing any significant changes of the system. However, so far, it remains very difficult to make full use of valuable on-line operation information. The major reason can be attributed to the computationally expensive retraining which can be too time consuming to harvest practical values.

As presented above, the ELM method can undertake learning at a very high speed, and based on this an ELM-based TSA model is developed in this paper. As shown in Fig. 2, an initial KB is off-line generated based on historical TSA archives and off-line instance simulations, and then a feature selection process is performed to identify significant features as input for training. The significant features cannot only enhance the accuracy of ELMs but also provide insight into critical system parameters. In the training phase, a set of single ELMs are trained using the significant features with respect to potential contingencies, and this process can be finished very rapidly. During the real-time TSA phase, the ELM-based model takes the system snapshot as input and gives the TSA results within a very short time delay. If instable status is shown, preventive controls should be activated to retain system stability. In parallel, the current system condition measurements are utilised to generate prospective operating conditions of the upcoming period and full time-domain simulation is executed to produce new instances. When the new instances appear unexperienced by the ELMs, they should be used to enrich and/or update the KB. In order to obtain more on-line instances, distributed computation technology for time-domain simulation can be adopted in this model [17]. As new instances are on-line generated, the original KB can be continuously enriched and/or updated, which simultaneously update and/or retrain the ELMs.

It should be noted that the proposed model is distinct from traditional computational intelligence-based DSA tools for its effective on-line updating capacity which can enhance the performance in the context of unforeseen major changes in the system. In addition, it is also worth noting that the proposed model can be readily extended to other types of stability assessment, provided the problem can be converted into classification or regression form.

In the proposed TSA model, other key methodologies include KB generation and feature selection. The related methods will be presented in subsequent sections.

3 KB generation

The application of supervised learning algorithms is based on the previous knowledge which depicts the characteristics of a system. For power system TSA, the KB should cover sufficient operating points to approximate the practical power system operating point space and can withstand uncertain operating condition changes. This section presents a KB generation scheme based on an extended load profile and economic dispatching.

During the operational planning phase, load demand forecasting is conducted, of which the results will be used for unit commitment, economic dispatch, maintenance scheduling, as well as off-line DSA etc. In order to acquire an abundant KB, the forecasted load profile can be extended for the purpose of enriching operating points.

For a forecasted load profile, the extended active and reactive load demand at each bus are given by

$$P_{\mathrm{L}i} = nP^{0}_{\mathrm{L}i} \pm \sigma \tag{8}$$

$$Q_{\mathrm{L}i} = P_{\mathrm{L}i}\eta \tag{9}$$

where $P^{0}_{\mathrm{L}i}$ is the forecast load demand at bus i, n is the forecast load profile coefficient describing different load patterns in a range of time intervals, σ is the deviation index which is a set of random values representing the unforeseen load pattern changes and η is the reactive power factor.

Given a forecasted load profile, the extended load points can be plotted in Fig. 3. By such schemes, the extended load points can accommodate a range (relies on specified σ)

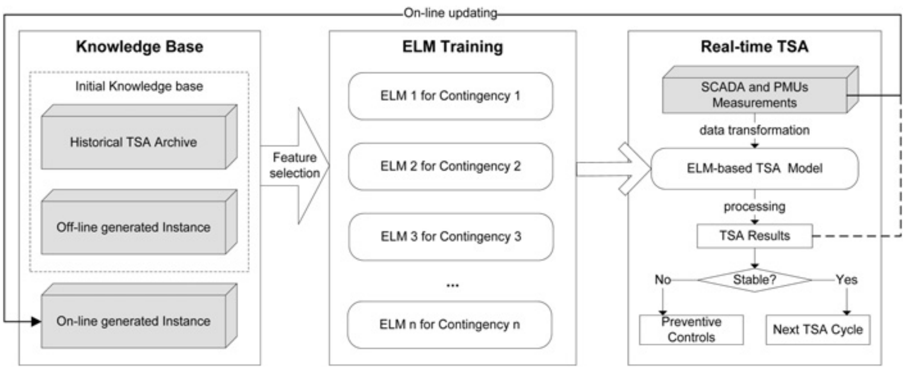


Fig. 2 *Proposed TSA model*

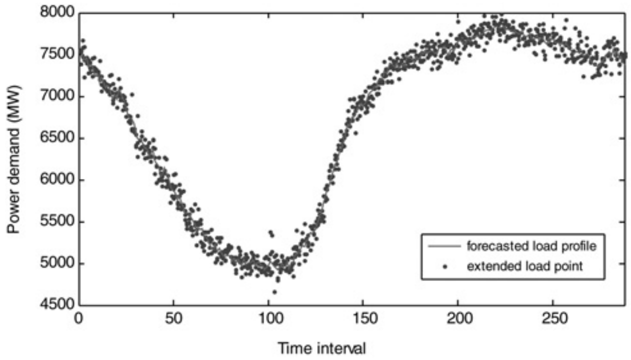


Fig. 3 *Forecasted and generated load profile*

of changes in load demands during the real-time operating phase. In addition, the resulting KB tends to be more robust.

In order to better represent the actual power system operation conditions, other variables of an operating point can be determined by an economic dispatching scheme, which is usually implemented in the modern dispatching centre. The mathematical formulations of the economic dispatching model with a generation cost minimisation objective can be given as follows

$$\text{Minimise} \quad \sum_{i\in S_{\text{G}}} (a_i + b_i P_{\text{G}i} + c_i P_{\text{G}i}^2) \tag{10}$$

Subject to

$$\begin{cases} P_{\text{G}i} - P_{\text{D}i} - V_i \sum_{j=1}^{n} V_j (G_{ij}\cos\theta_{ij} + B_{ij}\sin\theta_{ij}) = 0 \\ Q_{\text{G}i} - Q_{\text{D}i} - V_i \sum_{j=1}^{n} V_j (G_{ij}\sin\theta_{ij} - B_{ij}\cos\theta_{ij}) = 0 \end{cases} \tag{11}$$

$$\underline{P_{\text{G}i}} \le P_{\text{G}i} \le \overline{P_{\text{G}i}} \quad (i \in S_{\text{G}}) \tag{12}$$

$$\underline{Q_{\text{R}i}} \le Q_{\text{R}i} \le \overline{Q_{\text{R}i}} \quad (i \in S_{\text{R}}) \tag{13}$$

$$\underline{V_i} \le V_i \le \overline{V_i} \quad (i \in S_{\text{B}}) \tag{14}$$

$$\underline{S_{\text{L}i}} \le S_{\text{L}i} \le \overline{S_{\text{L}i}} \quad (i \in S_{\text{L}}) \tag{15}$$

where a_i, b_i and c_i are the generation cost coefficients of the ith generator, $P_{\text{G}i}$ is the active output of the ith generator in controllable generator sets S_{G}, $Q_{\text{R}i}$ is the reactive output of the ith reactive source in controllable reactive source sets S_{R}, V_i is the voltage of the ith bus in bus sets S_{B}, $S_{\text{L}i}$, is the apparent power across the ith transmission line in transmission line sets S_{L}.

Given a set of load points, the above model can be solved to determine the generation output and other variables of an operating point.

For each generated operating point, detailed time-domain simulations are executed to identify the transient stability index with respect to potential contingencies. Each operating point can be characterised by a vector of power system measurable variables, called features, and a vector of stability index under contingencies, called objects. In general, the features can be either the pre-contingency or post-contingency variables or even their combination, and the object can be denoted by numerical or nominal values.

It should be mentioned that the KB generation scheme can be used both in off-line and in on-line. For on-line instance generation, with the hourly load forecasting and unit commitment (wind power generation prediction can also be included), the prospective operating points for the next period can be generated very rapidly, then distributed computation technique can be adopted to speed up the computing efficiency of detailed time-domain simulation [17]; in this way, large number of on-line instances can be produced.

4 Feature selection for ELM

As previously mentioned, feature selection is needed to enhance the assessment accuracy. This is discussed in detail in this section. Feature selection is a data pre-processing aiming to eliminate irrelevant feature inputs by which the training data size can be significantly reduced for faster learning speed and better accuracy. For the ELM method, the learning speed is not sensitive to the training data size, but the parameters of the hidden nodes are randomly assigned even before the training data are seen; this means that all the input features are equally treated, if irrelevant or less relevant features are assigned over-contributable weights by randomness, and the prediction accuracy of ELM can be deteriorated. Therefore it is paramount to select significant features as input to ensure a high accuracy of ELM. In addition, the analytically selected features can also provide insights into system crucial parameters [3].

In the literature, principal component analysis (PCA) [11], divergence analysis [7], Fisher discrimination [4, 5] and wrapper models [18] are among frequently adopted methods for feature selection. These methods have shown satisfactory performance in eliminating redundant features; however, for wrapper models, although it can find out a feature subset that results in the best testing performance, the computation burden is always too heavy. PCA is very computationally efficient, but it only gives a linear combination of features from which crucial system parameters are difficult to identify. Fisher discrimination and convergence analysis are based on separability measurements of subsets and tend to provide potential system weak-point information.

As introduced in [4], Fisher discrimination measurement is based on Fisher's linear discrimination function $F(w)$, which is a projection from D-dimensional space onto a line in which manner the data are best separated. Given a set of nD-dimensional training samples x_1, x_2, L and x_n with n_1 samples in class C_1 and n_2 samples in class C_2, the task is to find the linear mapping, $y = w^{\text{T}}x$, that maximises

$$F(\text{w}) = \frac{|m_1 - m_2|^2}{\sigma_1^2 + \sigma_2^2} \tag{16}$$

where m_i is the mean of class C_i and σ_i^2 is the variance of C_i. Function $F(w)$ can be written as explicit function of w as

$$F(w) = \frac{w^{\text{T}}\boldsymbol{S}_{\text{B}}w}{w^{\text{T}}\boldsymbol{S}_{\text{W}}w} \tag{17}$$

where $\boldsymbol{S}_{\text{B}}$ is the between-class scatter matrix and $\boldsymbol{S}_{\text{W}}$ is within-class scatter matrix. The class separability of a feature set can be measured by

$$J_F = \text{trace}(\boldsymbol{S}_{\text{W}}^{-1}\boldsymbol{S}_{\text{B}}) \tag{18}$$

The magnitude of J_F can be an index of the linear separability of feature set; the higher the value of J_F, the more separable the data are.

In the previous works, in order to identify the optimal feature subset, the Fisher discrimination measurement is combined with a searching process [4, 5]. However, the searching process will introduce a huge computation cost when large-sized data are encountered; this can lead to inefficiency in the on-line model updating context. To overcome the drawback in computation speed, the Fisher discrimination criterion can be replaced by a simple variation as shown in (19), which is to evaluate the discrimination capability of a single feature.

For the kth individual feature, its discrimination capability can be calculated by

$$F_s(k) = \frac{\boldsymbol{S}_B^{(k)}}{\boldsymbol{S}_W^{(k)}} \tag{19}$$

where $\boldsymbol{S}_B^{(k)}$ and $\boldsymbol{S}_W^{(k)}$ are the kth diagonal elements of $\boldsymbol{S}_B$ and $\boldsymbol{S}_W$, respectively, the bigger the F_s of a single feature, the bigger discriminating capability it possesses, and the more significance it holds for classification.

For a fast feature selection, we can calculate F_s for each feature, order them in decreasing sequence of F_s values and simply select the top ones. As the F_s can indicate the separating ability of a feature in terms of transient stability index, the features with high F_s values are crucial to system stability and the preventive controls can be designed to control these variables to retain system stability.

5 Case study on the New England 39-bus test system

In this section, the proposed model is examined on the New-England 39-bus test system including ten generators and 39 buses, and its one-line diagram is shown in Fig. 4, [19]. This is a well-known test system for similar TSA studies in many previous works [11, 12].

5.1 KB generation

To examine the proposed method on the test system, 1500 operating points are generated using the scheme in Section 3, the load level of each bus ranges from 20 to 150% of base load, that is, $n \in [0.2, 1.5]$, the deviation index σ is given within ±8%, which is consistent with the mean average percentage error (MAPE) of ordinary load forecasting, and the reactive power factor η is given within 0.90–0.98. During generating operating points, the network topology changes are experienced by randomly removing single or multiple transmission lines in the system. Four severe contingencies are applied including three-phase, two-phase-to-ground and single-phase-to-ground fault with different clearing times and post-fault topologies. The detailed time-domain simulation is performed using the PSS/E package [17, 20].

In this paper, the pre-contingency steady-state variables are selected as the features and in order to accommodate the network structure changes in the KB, the topology-independent features can be picked out as the candidate features [16]. As for the object, the binary numerical value −1 and 1 is used to denote the transient unstable and stable status of an operating point following a contingency: if the maximum relative rotor angle deviation during the transient period exceeds π, the system is considered unstable (object marked as −1), otherwise the system is considered stable (object marked as 1). Accordingly, the features are normalised into $[-1, +1]$. In Figs. 5 and 6, a transient stable case and an unstable case are shown by plotting the rotor angle swing curve.

The generated KB is described in Table 1.

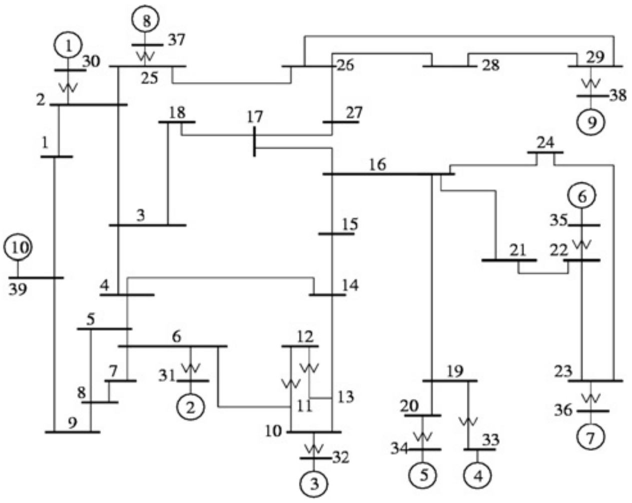


Fig. 4 *New England 39-bus system*

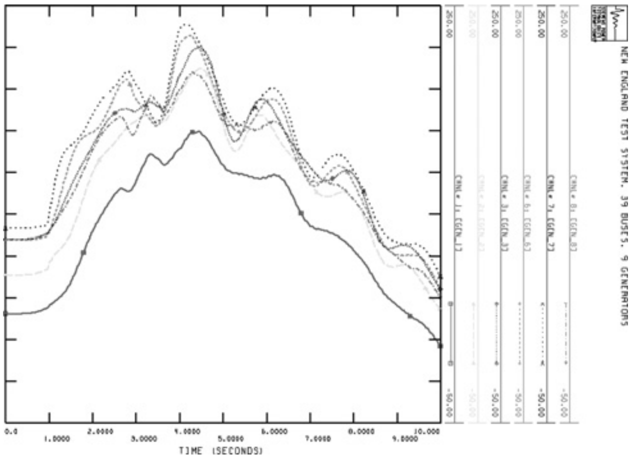


Fig. 5 *Transient stable case*

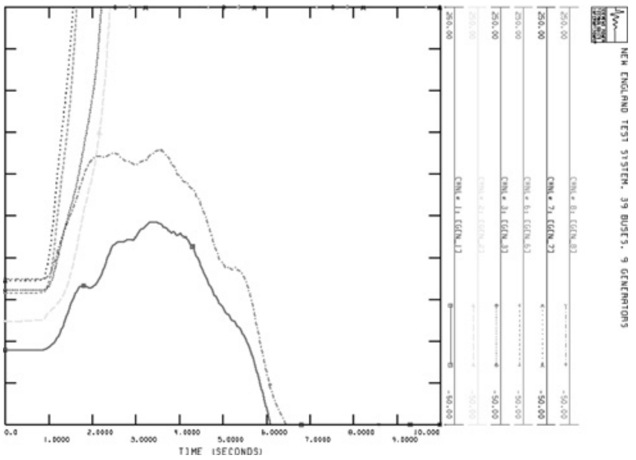


Fig. 6 *Transient unstable case*

Table 1 Generated data set description

Contingency ID	1	2	3	4
fault location	bus 29	bus 28	bus 16	bus 16
fault type	three-phase	three-phase	single-phase-to-ground	two-phase-to-ground
fault duration time	0.14 s	0.14 s	0.18 s	0.16 s
tripped line	29–26	28–29	16–17	16–21
stable OP	748	747	610	1248
unstable OP	752	753	890	252

In order to examine the proposed method without loss of generality, 500 out of 1500 generated operating points are randomly sampled with their stability indexes under the four postulated contingencies as the testing data set, and the remaining as the training data set, that is, KB.

5.2 Feature selection

As mentioned earlier, the fault-independent features can be selected as the candidate features to accommodate the network topology changes; the pre-contingency steady-state features listed in Table 2 are initially selected for the test system; all of them are basic measurement of power system and can be readily obtained in SCADA and PMU.

Fisher discrimination for a single feature is calculated on the training set, and the F_s value of all features are ordered in a descending sequence and shown in Fig. 7 where distinct F_s values can provide the indication that different features possess different separating capabilities.

According to Fig. 7, for faults 1–3, we can select the top 50 features, and the top 25 features for fault 4 as significant features considering their noticeably larger F_s values.

5.3 Optimal ELM structure determination

To train an ELM, no parameters need to be empirically pre-defined, except for the network structure (the number of hidden nodes). It is apparent that distinct data sets apply to different optimal network structures; hence for each single ELM with respect to different contingencies, the optimal structure needs to be tentatively determined. For this, the training data set is separated into two non-overlapped subsets, wherein one is for training and the other is for validation. The optimal number of hidden nodes is determined as the one which results in the highest validation accuracy.

Table 2 Initial candidate features

Notation	Description
P_{Gi}, Q_{Gi}	active and reactive power output of generator i
P_{Li}, Q_{Li}	active and reactive power load of bus i
P_{All}, Q_{All}	total active and reactive power load of the system
V_i, θ_i	voltage and angle of bus i
θ_{i-j}	angle difference of generator bus i and j

During the validation phase, the stability status assessed by ELM is determined according to its output value o_i

$$\begin{aligned} &\text{if } o_i \geq 0 \rightarrow o_i = 1 \text{ (stable)} \\ &\text{if } o_i < 0 \rightarrow o_i = -1 \text{ (unstable)} \end{aligned} \tag{20}$$

In order to show the benefits of feature selection, the whole candidate features and significant features are, respectively, used in the optimal ELM structure determination, and the validation results of ELM 1 for contingency 1 are illustrated in Figs. 8 and 9.

It can be observed that on the given data sets, the validation accuracy significantly rises with the added hidden nodes (starting from 5) before the highest validation accuracy is achieved, which is 95.44% with 155 hidden nodes and 97.94% with 96 hidden nodes, as shown in Figs. 8 and 9, respectively. Consequently, the optimal number of hidden nodes should be 155 and 96, respectively, when the whole features and significant features are used. In addition, it is important to note that the validation accuracy in Fig. 9 is generally higher than that of Fig. 8, indicating that the use of significant features can improve the accuracy.

5.4 Performance evaluation and comparison

To show the strength of the proposed method, popular computational intelligence methods are compared in terms

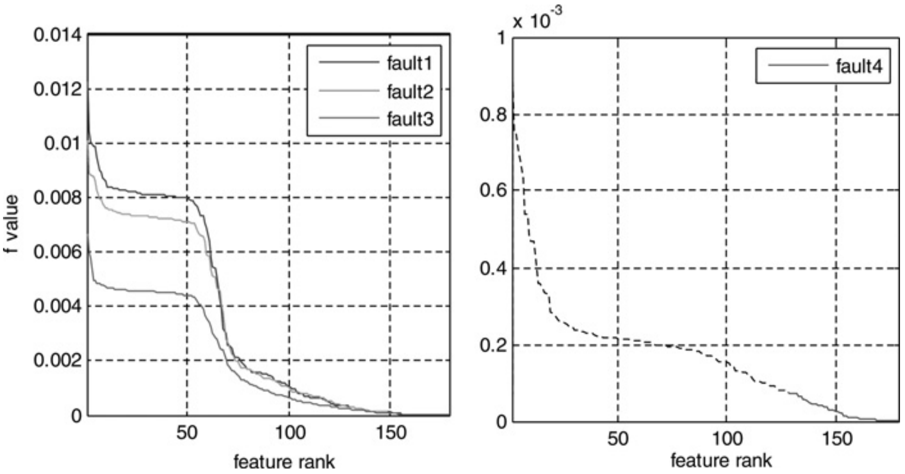


Fig. 7 *Features and their F_s values*

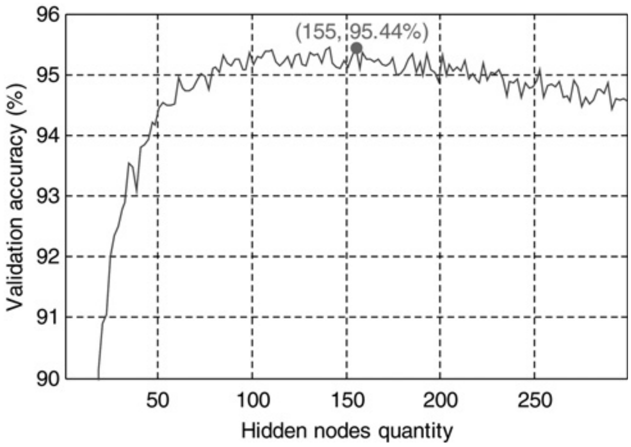


Fig. 8 *ELM validation accuracy using the whole candidate features*

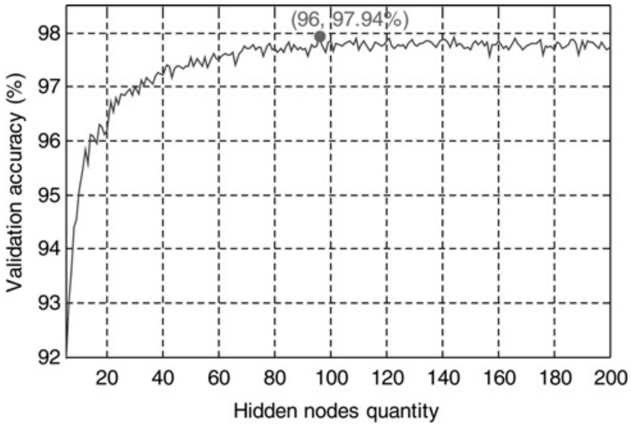


Fig. 9 *ELM validation accuracy using the significant features*

of computation time and prediction accuracy using the same training and testing data set. The methods to be compared include RBFNN, DT and SVM, which have shown satisfactory performance for DSA in the previous works as in [5, 6, 9], respectively. The simulations are implemented in MATLAB 7.0 running on an ordinary PC with 2.66 GHZ CPU.

Test 1: To demonstrate the benefit that on-line updating/enriching KB can provide, in Test 1, the four methods are trained with only 70% of generated training data and evaluated on the entire testing data.

The performance of the four methods is shown in Table 3, and note that the training time is the overall elapsed time during the training.

According to the test result, the four approaches are comparable in terms of testing accuracy (their overall testing accuracy are around 96%, although the DT method reaches the highest). However, in the computation speed aspect, the ELM method is far superior over the others: it only requires less than 0.1 s to learn the training data, whereas the others cost 1–10 min.

Test 2: In Test 2, the four methods are trained by the whole training data and applied on the same testing data as in Test 1. The performance is given in Table 4.

In the second test, the accuracy of all the four methods has been improved (overall testing accuracy around 98%, and ELM performs best in this test). When computation speed is concerned, the ELM method still shows significant superiority: the training time remains less than 0.1 s, whereas the other three methods cost more minutes. It is anticipated that for even larger data sets where more contingencies is considered and more features may be encountered, the strength of ELM in computation speed will be more valuable, with which rapid retraining with on-line-generated instances can be realised.

For visualised comparison purpose, the overall training time and testing accuracy in the two tests is plotted in Figs. 10 and 11.

5.5 Misclassification analysis and discussion

It should be noted that for the developed TSA model, despite the KB enrichment, a small quantity of misclassification is encountered on the generated data, as well as for the other three methods. In this section, the misclassified instances by the ELM method are investigated in detail and the scheme is discussed to reduce the risk of misclassification during the practical implementation.

Table 3 Test result 1: The bold values represent the best performance in each column

Methods	Overall training time, s	Testing accuracy, %				
		Fault 1	Fault 2	Fault 3	Fault 4	Overall
ELM	**0.4532**	**96.8**	96.6	94.4	**96.6**	96.1
DT	143.74	96.4	**97.2**	**95.8**	96.0	**96.35**
SVM	1613.4	96.6	96.6	92.2	95.8	95.3
RBFNN	348.98	96.6	96.6	94.4	96.0	95.9

Table 4 Test result 2: The bold values represent the best performance in each column

Methods	Overall training time, s	Testing accuracy, %				
		Fault 1	Fault 2	Fault 3	Fault 4	Overall
ELM	**0.5001**	**98.6**	**98.8**	**97.8**	**98.6**	**98.45**
DT	148.62	98.0	98.0	**97.8**	98.0	97.95
SVM	1727.65	97.2	97.4	96.8	96.6	97
RBFNN	383.64	97.6	97.0	97.0	96.4	97

doi: 10.1049/iet-gtd.2010.0355

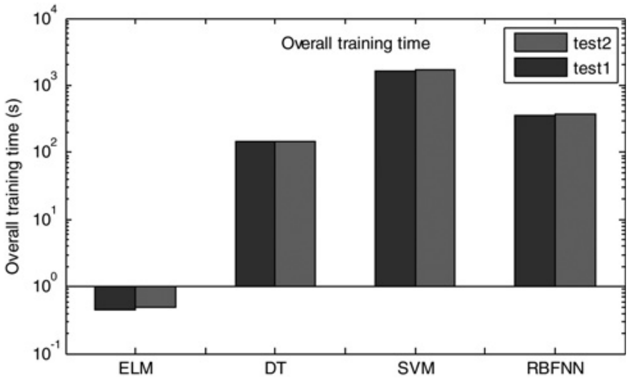


Fig. 10 *Training time of each method*

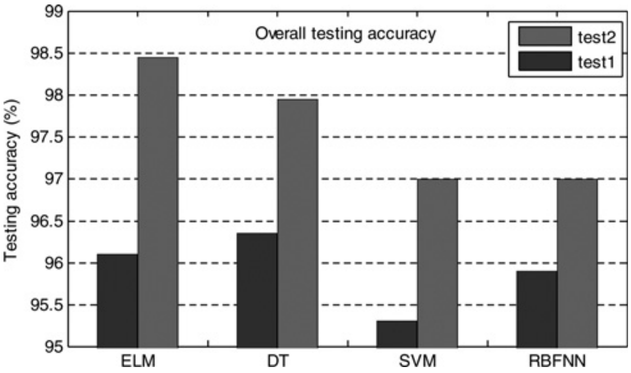


Fig. 11 *Testing accuracy of each method*

The encountered misclassified instances in the test are shown in Fig. 12 where their ELM output values are given.

It can be observed that most of the misclassified instances have an ELM output value very close to 0, which is the threshold value to determine the final stability status. Further investigation on the correctly classified instances shows that their ELM output values have an overall mean value as 0.878 and −0.732 for stable and unstable classifications, respectively. This can provide the important indication that when the ELM output is very close to the classification threshold (0 in this paper), the determination of final class can be less reliable. In order to reduce the risk of the misclassification during the practical implementation of the developed method, one of the potential methods is to apply time-domain simulation to determine an accurate stability status when such 'problematic' output is appeared. Although in this special case the computation time can be longer because of the time-domain simulation, the accuracy can be maintained and the overall computation speed remains very fast.

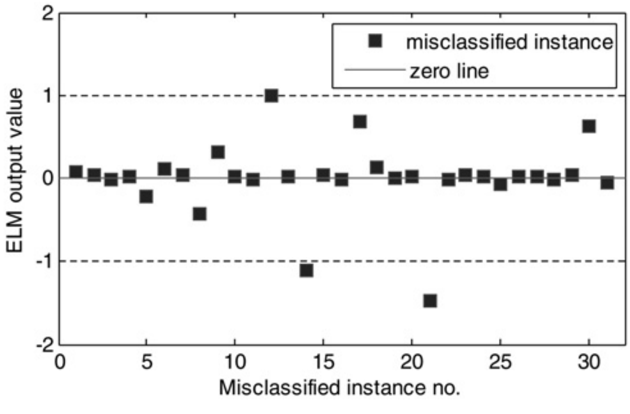


Fig. 12 *ELM output value of misclassified instances*

6 Conclusions

In this paper, the ELM theory is presented and an ELM-based TSA model is developed. By the initial training with an off-line-generated KB, the new TSA model can be implemented in real time; as its training speed is sufficiently fast, it can be on-line updated by real-time operating information by which way its performance can be kept accurate and more robust.

For an enhanced performance of ELMs, a fast feature selection method based on Fisher discrimination for a single feature is used to identify significant features as the input to ELM, and a scheme based on extended load profile and economic dispatching is employed to generate KB. The proposed model is examined on the New England 39-bus test system and compared with three state-of-the-art approaches. Simulation results show the benefit of KB enrichment and the significant superior computation speed with competitively high accuracy of the proposed model. The misclassified instances in the tests are investigated and the scheme is also discussed to reduce the risk of misclassification during the practical engineering implementation.

7 Acknowledgment

This research is partially supported by the Hong Kong Polytechnic University grant #YH55 and a Sichuan Smart Grid Key Laboratory Project on 'Data mining methods and WAMS based Power System Stability Assessment'.

8 References

1 Makarov, Y.V., Reshetov, V.I., Stroev, A., Voropai, I.: 'Blackout prevention in the United States, Europe, and Russia', *Proc. IEEE*, 2005, **93**, (11), pp. 1942–1955

2 Huang, J.A., Valette, A., Beaudoin, M., *et al.*: 'An intelligent system for advanced dynamic security assessment'. Proc. IEEE PowerCon 2002, Kunming, China, October 2002

3 Morison, K.: 'On-line dynamic security assessment using intelligent systems'. Proc. PES. General Meeting, Tampa, FL, June 2006

4 Jensen, C.A., El-Sharkawi, M.A.: 'Power system security assessment using neural networks: feature selection using Fisher discrimination', *IEEE Trans. Power Syst.*, 2001, **16**, pp. 757–763

5 Jain, T., Srivastava, L., Singh, S.N.: 'Fast voltage contingency screening using radial basis function neural network', *IEEE Trans. Power Syst.*, 2003, **18**, pp. 1359–1366

6 Sun, K., Likhate, S., Vittal, V., Kolluri, V.S., Mandal, S.: 'An online dynamic security assessment scheme using phasor measurements and decision trees', *IEEE Trans. Power Syst.*, 2007, **22**, pp. 1935–1943

7 Voumvoulakis, E.M., Gavoyiannis, A.E., Hatziargyriou, N.D.: 'Decision trees for dynamic security assessment and load shedding scheme'. Proc. 2006 IEEE Power Engineering Society General Meeting, pp. 1–7

8 Voumvoulakis, E.M., Hatziargyriou, N.D.: 'Decision trees-aided self-organized maps for corrective dynamic security', *IEEE Trans. Power Syst.*, 2008, **23**, pp. 622–630

9 Moulin, L.S., Silva, A.P.A., El-Sharkawi, M.A., Marks, R.J.: 'Support vector machines for transient stability analysis of large-scale power systems', *IEEE Trans. Power Syst.*, 2004, **19**, pp. 818–825

10 Zhao, J.H., Dong, Z.Y., Zhang, P.: 'Mining complex power networks for blackout prevention'. Proc. 13th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining, San Jose, CA, 12–15 August 2007

11 Sawhney, H., Jeyasurya, B.: 'A feed-forward artificial neural network with enhanced feature selection for power system transient stability assessment', *Electr. Power Syst. Res.*, 2006, **76**, (12), pp. 1047–1054

12 Amjady, N., Majedi, S.F.: 'Transient stability prediction by hybrid intelligent system', *IEEE Trans. Power Syst.*, 2007, **22**, pp. 1275–1283

13 Huang, G.-B., Zhu, Q.-Y., Siew, C.-K.: 'Extreme learning machine: theory and applications', *Neurocomputing*, 2006, **70**, pp. 489–501

14 Huang, G.-B., Chen, L., Siew, C.-K.: 'Universal approximation using incremental constructive feedforward networks with random hidden nodes', *IEEE Trans. Neural Netw.*, 2006, **17**, (4), pp. 879–892

15 Huang, G.-B., Zhu, Q.-Y., Siew, C.-K.: 'Extreme learning machine: a new learning scheme of feedforward neural networks'. Proc. 2004 IEEE Int. Joint Conf. Neural Networks, 2004

16 Nizar, A.H., Dong, Z.Y., Wang, Y.: 'Power utility nontechnical loss analysis with extreme learning machine method', *IEEE Trans. Power Syst.*, 2008, **23**, pp. 946–955

17 Meng, K., Dong, Z.Y., Wong, K.P., Xu, Y., Luo, F.: 'Speed-up the computing efficiency of power system simulator for engineering-based power system transient stability simulations', *IET Gener. Transm. Distrib.*, 2010, **4**, pp. 652–661

18 Wang, T., Guan, L., Zhang, Y.: 'A modified pattern recognition algorithm and its application in power system transient stability assessment'. Proc. IEEE Power Eng. General Meeting, 2008

19 Pai, M.A.: 'Energy function analysis for power system stability' (Kluwer Academic Publishers, 1989)

20 PSS/E User Support Area: www.pti-us.com/pti/software/pss/user_support.cfm

doi: 10.1049/iet-gtd.2010.0355