IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 23, 2018, accepted November 5, 2018, date of publication November 19, 2018,
date of current version December 18, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2880273

A Learning Framework for Size and Type Independent Transient Stability Prediction of Power System Using Twin Convolutional Support Vector Machine

ALIREZA BASHIRI MOSAVI[1], ALI AMIRI[1], AND SEYED HADI HOSSEINI[2]
[1]Department of Computer Engineering, University of Zanjan, Zanjan 45371- 38791, Iran
[2]Department of Electrical Engineering, University of Zanjan, Zanjan 45371- 38791, Iran

Corresponding author: Ali Amiri (a_amiri@znu.ac.ir)

ABSTRACT Real-time transient stability assessment (TSA) of power systems is an important real world problem in electrical energy engineering and pattern recognition scope. The definition of most discriminative trajectory features and proper supervised trajectory-based classifier has remained a motivational challenge for scholars vis-à-vis real-time TSA. In addition, increase in the consumption of electrical energy along with constraints such as amortization of network equipment induces electric power system inadequacy risk. The retrieval of power system adequacy involves network expansion planning such as installing new power plants for the network. This policy affects the structure and electrical specification of the network significantly. Furthermore, due to sudden or the scheduled tripping of network equipment stemming from action of protection devices or maintenance procedures, the network must undergo shallow structural changes. The different level of changes in network specification is becoming a potential barrier for network analysis tools like real-time TSA platform. In fact, the lack of consideration of the incompatibility of TSA tool with expansion planning affects the performance of TSA learning model that is trained using the pre-expansion network. However, this paradoxical problem can be solved by generalized learning for power system size & type independent (PSs&tInd) real-time TSA. For this purpose, first, we used a set of PSs&tInd trajectory features. Next, we presented a trajectory-based deep neuro classifier to eliminate kernel functions weaknesses plugged into the hyperplane-based classifier. Finally, experimental comparisons were conducted to assess the efficacy of the proposed framework. The results showed that the proposed technique offered high-generalization capacity on real-time TSA during network expansion.

INDEX TERMS Transient stability assessment (TSA), size and type independent trajectory features (s&tIF), convolutional neural network (CNN).

I. INTRODUCTION

In today's electric power system environment in which different uncertainties like power interruption affects interconnected utilities of the electrical grid, stability analysis is a basic function of the power system reliability issue. In fact, power system stability monitoring in terms of its operation positioning near the stability margins is the most significant obligatory item, which is guaranteed to enhance the social indicators (e.g., economic factor) as well as the efficiency of the power grid. Consequently, system stability control is becoming a potential option in solving future challenges associated with power supply in order to achieve secure and economic power supply. For this purpose, transient stability assessment (TSA) has been introduced as one of the categories of power system stability analysis. In fact, transient stability refers to the ability of the power system to maintain synchrony during large and sudden disturbances [1]. There are several techniques for measuring transient stability status. Time-domain simulation and transient energy function are traditional approaches of TSA [2], [3]. However, in comparison with traditional approaches that have weakness such as obtaining exact parameters and information on network components, data mining techniques is considered as a state-of-the-art solution on real-time TSA [4]–[9]. Currently, the development-oriented role of information technology (IT) has caused the staking of data with a large volume of various

power grid elements. Consequently, data mining provides proper analytical tools for knowledge discovery involving methods in the intersection of machine learning and statistics [10].

Different purposes of real-time TSA, namely preventive control, and corrective control can be studied employing data mining. Due to the unpredictable property of the events in power system, network operators can adjust the operating point of the network in a way to be able to optimize control actions in the preventive mode to prevent power system instability. However, it is clear that such a mechanism will induce non-optimal exploitation of the network. Taking into cognizance this point, transient stability status prediction based on post-fault dynamic response (corrective control) is very imperative. To collect post-fault dynamic response of power systems, wide area monitoring system (WAMS) e.g. phasor measurement units (PMUs) [11] is one of the fundamental equipment employed in real-time TSA.

Many different data mining techniques have been proposed for real-time TSA. For example, decision tree (DT) [12], [13] is one of the practical algorithms for predicting power system transient stability. In addition, ensemble DT (Random Forest) has been considered for transient stability analysis [14]. Support vector machine (SVM) is one of the useful methods employed in real-time TSA [15]. Artificial neural networks (ANNs) have attracted the interest of researchers lately and has been widely considered to improve the performance of transient stability status prediction [16], [17].

In this paper, three significant problems are considered in the performance of real-time TSA: 1) The first concern is the selection of proper trajectory features that have close relevance to its performance. Several features have been employed in earlier studies. For example, in [18]–[21] selected terminal voltage amplitude as a predictor. Also, the post-fault dynamic behavior of generator rotor angle was selected as predictors for real-time TSA in [22] and [23]. More focus on discriminative trajectory features that reflect key information in real-time TSA requires further study. 2) The next issue is related to an interconnected relationship between power grid sharp expansion and performance of real-time TSA learning framework. The power grid is experiencing growing demand in power supply along with electrical constraints (e.g. amortization of network equipment). These issues certainly affect power system adequacy. Under this circumstance, power grid must inevitably be designed on structural growth path and this policy can be formulated on network management agenda in two ways: a) deep expansion planning, which include installation of new power plants, transmission line and substations, and b) short-term plans, namely shallow expansion planning, which is conducted based on the sudden or scheduled tripping of network equipment stemming from action of protection devices or system maintenance procedures. Due to the different level of power system specification expansion for retrieving power system adequacy, machine learning approaches to power system dynamic security analysis such as real-time TSA learning procedure that is trained employing pre-expansion system is not applicable in the expanded power grid. However, solving this issue involves generalization of real-time TSA learning model, which defines power system size & type independent trajectory features as cornerstone of this solution. In this way, indirectly, the first problem is also solved. 3) The third problem is related to how to decide the best classifier on time series predictors. Due to the importance of swing curve mining, application of suitable classifier with more intuitive comparison between trajectory features in the learning process is very important. Although some researchers utilized kernel-based classifier for this purpose, future challenges of kernel selection as well as setting of parameters in hyperplane-based predictive approach affects the performance of the classifier on real-time TSA. Consequently, application of proper supervised trajectory-based classifier without requiring kernel trick to possess high performance prediction is a challenge of this topic.

The main contribution of the paper to address the abovementioned concerns are summarized as follows:

a) Since generalization of learning model is considered in our proposed methodology, we used a set of size and type independent trajectory features (s&tIFs) that measures suitable awareness level of the network status and its distance from instability.

b) Improving the performance of the classifier on the multivariate trajectory for power system transient data classification is an urgent need that helps data mining engineer to possess high-performance knowledge discovery. For this purpose, we present a convolutional neural network (CNN) framework with novel companion objective function, called twin convolutional support vector machine (TWCSVM) for mining the internal structure of trajectory features appropriately.

c) Several experiments for evaluating the generalizability of the proposed learning model based on strict strategy is considered. Furthermore, an experimental comparison of kernel-based predictive approach on real-time TSA and TWCSVM are performed.

The rest of the paper is organised as follows: In Section II, we explained the structure of the proposed methodology in order to achieve power system size & type independent real-time TSA. Section III describes the numerical experiments used to evaluate the performance of our proposed methodology. Finally, Section IV presents the conclusion of the paper.

II. METHODOLOGY

The proposed methodology for power system size & type independent transient stability prediction is related to the issues raised in this paper, as shown in Fig. 1. Our goals are considered in three step interconnected components as follow: 1) database generation module, 2) feature calculation module and 3) trajectory-based deep neural predictive module. First, time domain simulation is conducted on the various power grid (step 1) based on the desired basic attributes

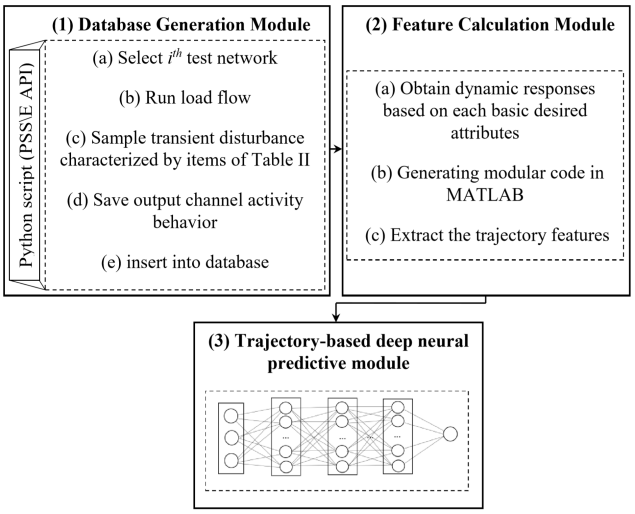

FIGURE 1. Overall process of PSs&tInd real-time TSA methodology.

TABLE 1. Definition of s&tIF.

s&tIF (1:8): Math formula [a]	s&tIF (1:8): Descriptions
$F_1^{t_m} = \frac{\sum_{i=1}^{N_{busgen}} PELEC_i}{\sum_{i=1}^{N_{busgen}} P\max_i}$	Proportion of PELEC to total of max PELEC
$F_2^{t_m} = Max([\frac{PELEC_i}{P\max_i}]^{i=1:N_{genbus}})$	Maximum [proportion of PELEC to max PELEC]
$F_3^{t_m} = Min([\frac{PELEC_i}{P\max_i}]^{i=1:N_{genbus}})$	Minimum [proportion of PELEC to max PELEC]
$F_4^{t_m} = Max\ ([VOLT_i]^{i=1:N_{bus}})$	Maximum [bus voltage]
$F_5^{t_m} = Gradient\ [Max([\frac{PELEC_i}{P\max_i}]^{i=1:N_{genbus}})]$	Derivative of maximum [proportion of PELEC to max PELEC]
$F_6^{t_m} = Gradient\ [Max([VOLT_i]^{i=1:N_{bus}}$	Derivative of maximum [bus voltage]
$F_7^{t_m} = Gradient\ [Var([VOLT_i]^{i=1:N_{bus}})]$	Derivative of variance [bus voltage]
$F_8^{t_m} = Gradient[Mean(abs([VANGLE_i - VANGLE_j]^{i,j=1:N_{bus}}))$	Derivative of mean [voltage phase angle difference on lines]

[a] Symbol: $N_{bus\ gen}$ = number of bus generator in test case, PELEC= machine electrical power (pu), P_{max}= maximum amount of machine electrical power, Volt= bus pu voltages, N_{bus}= number of buses in test case, VANGLE= voltage phase angle, Var= variance, Max= maximum, Min= minimum.

(e.g., active power). Second we apply feature calculation module to extract s&tIFs based on basic attributes (step 2). Finally, a trajectory-based classifier without kernel tricks is presented to achieve high performance prediction of real-time TSA (step 3).

A. Database generation scheme for PSs&tInd TSA

In this paper, offline simulation is conducted to generate a dataset that consists of dynamic responses of various power systems with different topology. Unlike other studies on real-time TSA, the desired case is the same for training and testing phase; we consider a strict strategy for evaluating the generalizability of the proposed learning model. For strict evaluation of learning model generalizability, contingency samples of power networks that are not topologically similar to training phase power networks are used in the testing phase. In fact, the use of power networks that are topologically different from training phase power networks in the testing phase gave the assurance that if the proposed model achieves a high performance prediction against contingency samples of different types of network topologies, it will also hold a high generalization capacity during network expansion. Taking into account the above strategy, we need a set of trajectory features that could help us to achieve power system size & type independent transient stability prediction. To this end, a set of PSs&tInd trajectory features are employed as shown in TABLE I. Basic reasons involved in the definition of 8 s&tIF are summarized as follows. In fact, application of stability limit in inducing a change and fault in the power grid precipitates transient instability. Therefore, awareness of instability occurrence would be possible via measurement of the condition of network exploitation, which states the range of distance of the operation point of the network with stability limit. Network exploitation conditions or operation point characteristics is measurable by measuring features such as total amount of active power, voltage phase angle of buses and terminal voltage magnitude. However, in order to extract power system s&tIF that measures suitable awareness level of network status and its distance from instability, these features should be normalized. In other words, a feature like the amount of active power expression nominal capacity of the operation point for a network. Nevertheless, these features should be normalised according to the total capacity of the installed power plants in comparison with the networks of various size. Furthermore, settlement of part of the whole network at stability limits will induce transient instability. In order to be aware of similar conditions, geometric attribute (e.g., maximum, minimum or variance value) according to feature should be evaluated. Since transient instability occurs because of problems stemming from the active power supply, most of the features determined are related to this feature. Variations in active power operation induced differences in voltage phase angle. Therefore, a feature is defined in order to measure voltage phase angle in the feature set. Also, voltage magnitude as an operational feature affecting transient instability is added to other features. For more details, refer to [24].

B. Proposed trajectory-based deep neural classifier

Sequential (or time series) predictors are one of the data types in most real world problems. The analysis of this data type, called multivariate data analysis, requires specific theory in pattern matching and traditional similarity metrics like Euclidean distance, which are not ready to be used in confrontation of these challenges. For this purpose, elastic distance like dynamic time warping (DTW) has been introduced as a distance measure to find optimal matching

between two trajectory data [25]. Because of the application of non-linear (elastic) alignment in DTW, it outperforms Euclidean distance (or Manhattan distance) in most cases. Therefore, choice of distance measure plugged into kernel machines like SVM is important. Due to the fact that DTW produce rich similar score, it motivates pattern recognition researchers to apply DTW in Gaussian radial basis functions (RBF) kernel. However, research work presented in [26], showed that DTW was not a positive definite symmetric (PDS) in some cases and their experiments further revealed that Euclidean distance outperformed DTW in kernel machines. Although, application of elastic distance in kernel function was fruitless, construction of PDS kernel from elastic distance continued, consequently, [27] presented a positive definite recursive elastic kernel called recursive time warp kernel (RTWK). The question is whether the kernel selection problem has been solved by PDS kernel. The answer to this question is no. Future challenges for the set of parameters in kernel functions have remained. In addition, in some cases, the performance of elastic kernel in pattern matching is dependent on the characteristics of data. According to what was mentioned on elastic kernels weaknesses, presentation of a trajectory-based classifier that does not require kernel tricks is an essential task of our proposed methodology for real-time TSA. For this purpose, we introduced a trajectory-type deep neural network classifier, called twin convolutional support vector machine (TWCSVM). TWCSVM consist of the convolutional neural network (CNN) for mining the internal structure of trajectory features (feature mapping module) without kernel tricks, and twin support vector machine (TWSVM) as a novel companion has an objective function that is associated with CNN (classification module). CNN is one of the trajectory-based deep neuro strategies presented in [28] research work that had good performance in pattern recognition and image processing. Utilizing the concept of CNN, [29] presented a novel CNN framework for time series classification. In fact, this approach was developed to reduce weaknesses of time series classification techniques like model based, distance based and feature based schemes. The main layers of the proposed CNN by [29] are comprised of: 1) input layer, 2) convolutional layer, 3) pooling layer, 4) feature layer and 5) output layer. In CNN architecture, the feature mapping procedure was done from input layer to pooling layer and then, the objective function was fed from feature layer. In our proposed classifier, deep layers of CNN perform the role of the kernel trick in hyperplane-based predictive approach and it also eliminates future challenges in kernel function selection. In the classification module of TWCSVM, unlike common objective function (mean-square error), application of the primal problem formulations of TWSVM, which showed good classification performance under comprehensive benchmark experiment [30], precipitated an optimal solution for structural risk minimization of the classification problem. Appendix A offers a principle of TWSVM for readers at a glance. According to preliminaries mentioned above about

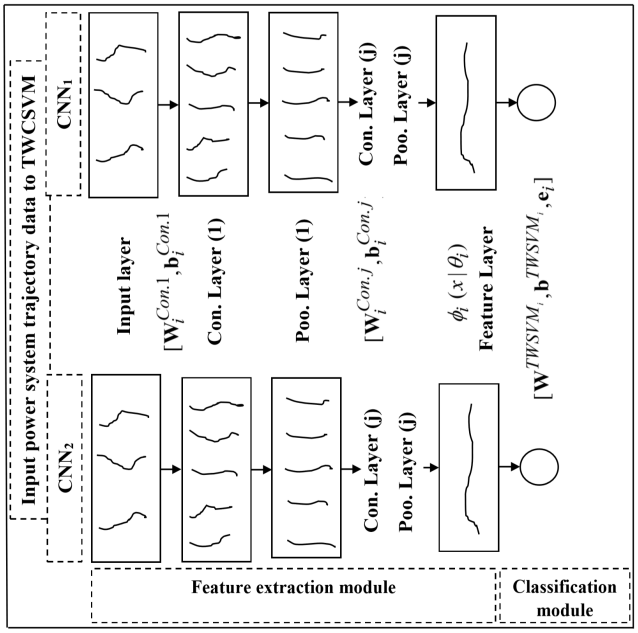

FIGURE 2. classifier for real-time TSA.

TWCSVM architecture, the framework of TWCSVM consists of two convolutional neural networks (CNNs): CNN_1 and CNN_2. Each CNNs contains the input layer, hidden layers (convolutional layer and pooling layer) and the output layer. Fig. 2 illustrates the architecture of our proposed classifier.

Some details about the TWCSVM classifier are as follows: In feature extraction module of each CNNs, feature mapping procedure as kernel function, which continued from the input layer to last hidden layer (convolutional and pooling layer), is presented as $\phi(x|\theta)$, where $\theta = \{W^{Con.j}, b^{Con.j}, 1 \leq j \leq c\}$. Classification module of CNNs include TWSVM objective function, which is fed from $\phi(x|\theta)$. The weight vector and bias of the classification module for each CNNs is represented by W^{TWSVM}, b^{TWSVM}. Generally, parallel training of each CNN was conducted according to the following steps:

Step 1) Set CNNs parameters like number of convolutional layer, filter number of convolutional layer, filter size of the convolutional layer, and pooling method. The weights matrix and bias vector per CNN between the convolutional layer and pooling layer was initialized with a small random number. Next, activation function should be determined. The sigmoid function due to its special properties was selected [31].

$$f(x) = sigmoid(x) = \frac{1}{1 + e^{-x}} \tag{1}$$

Step 2) All multivariate training set was selected in the batch form.

Step 3) The output of convolutional and pooling layer per CNN was calculated as described in [29]. The output of output layer can be written as (2):

$$out_i = W^{WTSVM_i}.\phi_i(x|\theta_i) + b^{TWSVM_i}, 1 \leq i \leq 2 \tag{2}$$

Step 4) The mean square error (MSE) was displaced by TWSVM primal problems as novel companion objective function. Therefore, we formulated the objective functions as follows:

a) Objective function of CNN_1:

$$obj_{CNN_1} = \min_{\bar{\theta}_1} \frac{1}{2}||\phi(x^+|\theta_1).W^{TWSVM_1} + e_1 b^{TWSVM_1}||^2 + C_1 e_2^T \sum_{k=1}^{N^-} [\max(e_2 + (\phi(x^-|\theta_1)W^{TWSVM_1} + e_2 b^{TWSVM_1}), 0)]^2 \quad (3)$$

where $\bar{\theta}_1 = \theta_1 \cup \{W^{TWSVM_1}, b^{TWSVM_1}\}, \theta_1 = \{W_1^{Con.j}, b_1^{Con.j}\}$.

b) Objective function of CNN_2:

$$obj_{CNN_2} = \min_{\bar{\theta}_2} \frac{1}{2}||\phi(x^-|\theta_2).W^{TWSVM_2} + e_2 b^{TWSVM_2}||^2 + C_2 e_1^T \sum_{i=1}^{N^+} [\max(e_1 - (\phi(x^+|\theta_2)W^{TWSVM_2} + e_1 b^{TWSVM_2}), 0)]^2 \quad (4)$$

where $\bar{\theta}_2 = \theta_2 \cup \{W^{TWSVM_2}, b^{TWSVM_2}\}, \theta_2 = \{W_2^{Con.j}, b_2^{Con.j}\}$. Max function in (3) and (4) was defined as $\max(a, b) = \begin{cases} aa \geq b \\ 0a \leq b \end{cases}$.

Step 5) The weights and bias updating procedure per CNN was conducted by gradient descent algorithm:

$$\bar{\theta}_i(t+1) = \bar{\theta}_i(t) - \eta \frac{\partial obj_{CNN_i}}{\partial \bar{\theta}_i}|_{\bar{\theta}_i(t)} \quad (5)$$

where $\bar{\theta}_i = \{W_i^{Con.j}, b_i^{Con.j}, W^{TWSVM_i}, b^{TWSVM_i}\}, 1 \leq i \leq 2, 1 \leq j \leq c$ and η is a learning rate.

For example, the fine tuning of the TWCSVM weight vectors in classification module (W^{TWSVM_1}, W^{TWSVM_2}) by (5) are as follows:

a) Update W^{TWSVM_1} vector:

$$cons = e_2 + (\phi_1(x^-|\theta_1) * W^{TWSVM_1} + e_2 * b^{TWSVM_1})$$
$$\Lambda = 2C_1[\max(cons, 0)]^2 \quad (6)$$
$$\frac{\partial obj_{CNN_1}}{\partial W^{TWSVM_1}} = \left(\overset{+}{(out_1)} * \phi_1(x^+|\theta_1) + \left(\Lambda * \phi_1(x^-|\theta_1)\right) \right)^T \quad (7)$$

b) Update W^{TWSVM_2} vector:

$$cons = e_1 - (\phi_2(x^+|\theta_2) * W^{TWSVM_2} + e_1 * b^{TWSVM_2})$$
$$\Lambda = -2C_2[\max(cons, 0)]^2 \quad (8)$$
$$\frac{\partial obj_{CNN_2}}{\partial W^{TWSVM_2}} = \left(\overset{-}{(out_2)} * \phi_2(x^-|\theta_2)) + \left(\Lambda * \phi_2(x^+|\theta_2)\right) \right)^T \quad (9)$$

where $*$ and $(.)^T$; matrix multiplication and matrix transpose, respectively.

Step 6) Another round of training continued with step 2 according to maximum epoch number where it was initialized at the beginning of the learning process.

TABLE 2. test systems.

Test case	Description
Brazilian 7-Bus equivalent model	Reduced order equivalent model of the South-Southeastern Brazilian system configuration [32].
New England /New York Interconnection (NETS-NYPS) system	The 68-Bus, 16-Machine, 5-Area system is a reduced order equivalent of the inter-connected New England test system and New York power system [32].
Two-Area, 4-Generator system (T-A)	The two-area system test case is from Prabha Kundur's textbook power system stability and control [32], [33].
SAVNW test system	Test case that is provided with the PSS/E software package [34].
IEEE-24 Bus system	The IEEE 24-bus test system was developed by the IEEE and published in 1979 as a benchmark for testing reliability analysis methods [35].

If a power system utility wishes to implement the proposed algorithm to classify test sample in the desired power system, the basic steps described in Appendix B should be followed.

III. EXPERIMENTAL DESIGN

A. Electric grid test cases

The test systems chosen in this paper to evaluate the performance of our proposed methodology is shown in TABLE II. The contingency samples of SAVNW, IEEE24 and Brazilian test systems were used for training process, and contingency samples of NETS-NYPS and T-A system were used for evaluation of the learning model generalizability based on s&tIF features (See TABLE I).

B. Summary of simulation procedure Database

We executed offline contingency simulation procedure on a programming platform (See Fig. 3) employing a Surface Pro 4 with an Intel Core i5-6300 2.5 GHz processor and 4 GB RAM. First, we coded contingencies associated with different load characteristics, type of disturbance and fault duration times (See TABLE III) by the Python script, which is capable of accessing simulation functions of SIEMENS PSS®E planning tools (PSS/E). In this stage, output channel activity behavior of all test cases were extracted based on basic attributes (e.g., active power, voltage magnitude, etc.). The required parameters per contingency sample using PSS/E application program interface (API) routine was determined. Accordingly, fault duration time was set at 0.16 s and 0.23 s. The fault clearing time was set after the end of each fault duration time. Also, the test system was simulated for 5 s after the fault clearance. For more details on the sampling frequency of measurement refer to [22]. When all contingency samples of all test systems were obtained; then, the data file was sent to feature calculation module in MATLAB to extract trajectory features, which were previously defined in TABLE I. The dataset form for real-time TSA is shown in Fig. 4. After contingency samples gathering, the study on dynamic behavior of the trajectory data is considered as a preprocessing step of the learning procedure. Focusing

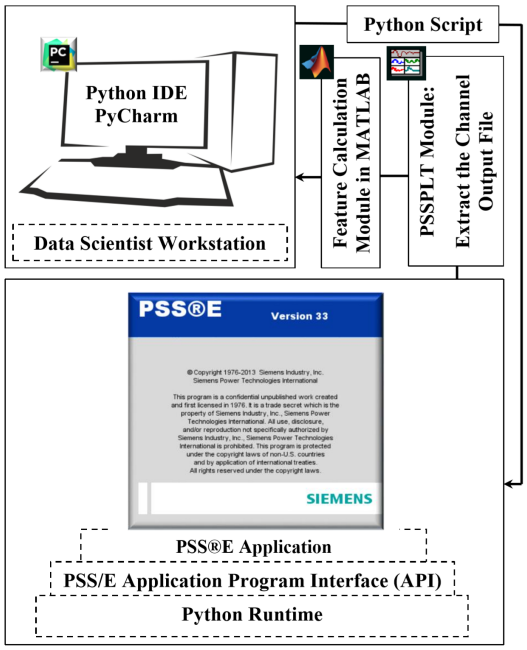

FIGURE 3. Contingency simulation procedure.

on the dynamic behavior of trajectory data using distance measure like DTW showed this fact that feature excursions of some contingency samples have the same behavior. In fact, we confronted with data redundancy in power system contingency dataset. Data redundancy causes increasing the size of the database unnecessarily; and as a result, increasing the time complexity of learning model procedure. Therefore, for filtering unique samples, redundant data was removed from dataset. Consequently, 414 simulation samples from all samples in the available database as a training and testing set was selected. Each contingency sample was labeled as ''stable'' or ''unstable'' according to network response in the solution of dynamic equations. Also, the proportion of the two classes (stable or unstable) in the available dataset is balanced. As can be seen in Fig. 4, stable samples (e.g., green-face trajectory) and unstable samples (e.g., yellow-face trajectory) of NETS-NYPS, show variation in the proportion of PELEC compared to the total of max PELEC (F_1).

C. Experimental results

1) TWCSVM PARAMETERS SETTING

One of the important aspects in designing our experiments on power system multivariate data was the determination of various parameters per CNN in TWCSVM architecture. In our proposed classifier, number of convolutional layer, filter number of convolutional layer, filter size of the convolutional layer and type of pooling strategy were the significant parameter that could affect the performance of TWCSVM. At the beginning of the TWCSVM design, each CNN of TWCSVM was designed by one convolutional layer as regards the number of convolutional layers parameter. To evaluate the impact of the other parameters on the performance of TWCSVM,

TABLE 3. Different uncertainties to generate dynamic responses.

Parameter	Parameter elements
Convert Constant MVA loads	LOADIN[a] (1): is the percent of active power load to be converted to the constant current characteristic, [0-100] (%).
	LOADIN (2): is the percent of active power load to be converted to the constant admittance characteristic, [0-100] (%).
	LOADIN (3): is the percent of active power load to be converted to the constant power characteristic, [0-100] (%).
	LOADIN (4): is the percent of reactive power load to be converted to the constant current characteristic, [0-100] (%).
	LOADIN (5): is the percent of reactive power load to be converted to the constant admittance characteristic, [0-100] (%).
	LOADIN (6): is the percent of reactive power load to be converted to the constant power characteristic, [0-100] (%).
Type of disturbance	Outage [substation, generator, line, random combination of generator and line outages]
Fault duration time	[0.16, 0.23] (second) [b]

[a] is an array of six elements; [b] New relays are capable of detecting faults and issuing commands at very fast times. In this paper, the value of the fault duration time was considered greater than the relay operating time. The reason of considering these values in the contingency simulation procedure is to generate the contingency samples that with their occurrence, conditions of instability occur on the test network while by considering relay operation with short function time, the number of unstable samples would be limited.

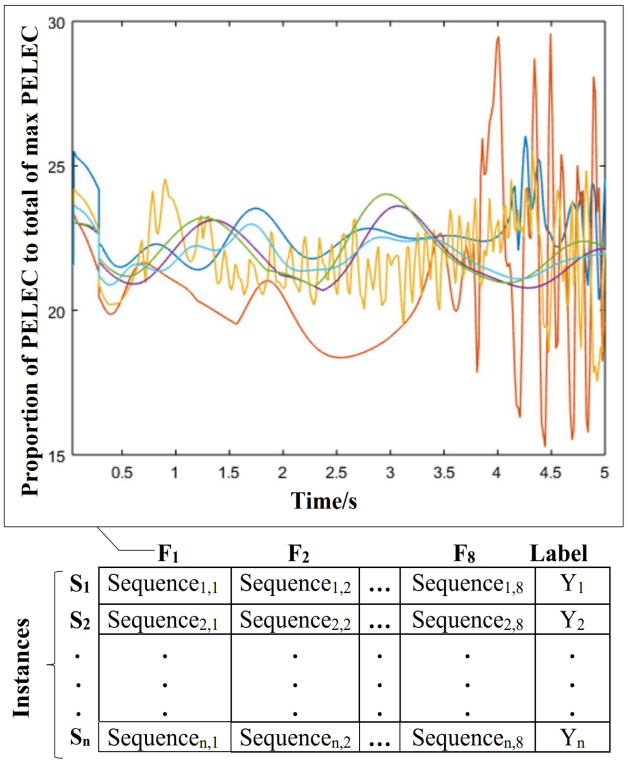

FIGURE 4. Dataset form for real-time TSA.

several experiments were conducted on the test cases. SAVNW, Brazilian and IEEE-24 test systems were employed for training process, and NETS-NYPS and T-A system were

TABLE 4. Impact of parameters on TWCSVM performance (NETS-NYPS).

TWCSVM classifier			
Parameters setting	CNN_1	CNN_2	Acc/ TPR [a]/ TNR [b]/ PT [c]
1: Filter number of the convolutional layer (F.n.)	2	3	75/ 100/ 57.14/ 767 ms [d]
	3	**3**	**97.22/ 100/ 95.24/ 808 ms**
	5	5	94.44/ 86.67/ 100/ 957 ms
	6	5	91.66/ 80/ 100/ 1003 ms
	9	8	88.88/ 100/ 80.95/ 258 ms
2: Filter sizes of the convolutional layer (F.s.)	312	312	94.44/ 86.67 / 100
	310	312	91.66 / 93.33/ 90.48
	310	311	91.66/ 93.33/ 90.48
	309	308	86.11/ 80/ 90.48
	307	306	86.11/ 66.67/ 100

[a] True Positive Rate, [b] True Negative Rate, [c] Prediction Time, [d] millisecond

used as a test set to assess the generalizability of the proposed model based on s&tIF features. Also, to assess the feasibility of learning model generalization for PSs&tInd transient stability prediction, observation window with length 5 s ($c = 300$: length of trajectory feature F_i in cycles c) was considered for training and testing procedure (Section $C.1$ and $C.2$). The accuracy (Acc) index (10) was used for evaluation of classifier performance.

$$Acc = \frac{TruePositive(TP) + TrueNegative(TN)}{TP+TN+FalsePositive(FP) + FalseNegative(FN)} \tag{10}$$

Where *Positive*: identified stable sample, *Negative*: identified unstable sample, *TP*: correctly identified stable sample, *TN*: correctly identified unstable sample, *FP*: incorrectly identified stable sample and *FN*: incorrectly identified unstable sample. Furthermore, sensitivity (11) and specificity (12) was used to measure the proportion of actual positives (or negatives) that are correctly identified.

$$Sensitivity(TPrate) = \frac{TP}{TP + FN} \tag{11}$$

$$Specificity(TNrate) = \frac{TN}{TN + FP} \tag{12}$$

TABLE IV shows the classification performance of TWCSVM for NETS-NYPS test system based on important parameter. The best values for TWCSVM parameters that precipitated the high performance prediction of power system transient stability was shown by bold-face. More analysis of TABLE IV shows the fact that TWCSVM can get proper classification performance when the value of convolutional layer parameters increases to a certain extent. For example, when the value of the filter number of the convolutional layer became greater than 3 for CNN_1 and CNN_2, in addition to increasing prediction time of TWCSVM to classify the test contingency sample, the performance of the classifier based on TPR and Acc reduced (Also Fig. 5 confirms this issue). An important point is that we chose the accuracy obtained based on parameter 1 in row 3 (underline face) to evaluate the impact of filter size (parameter 2) on

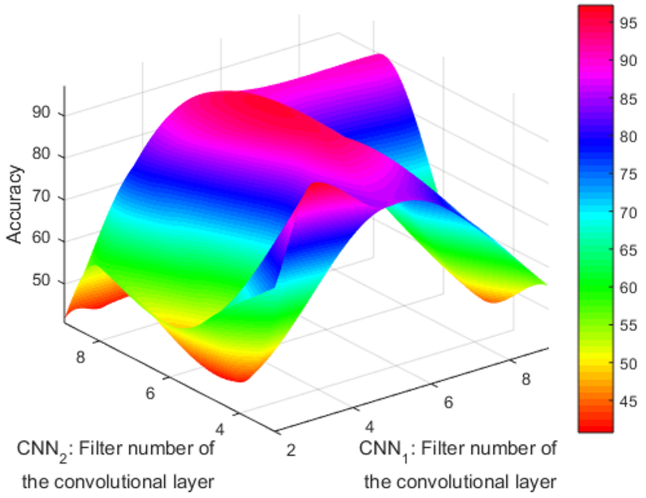

FIGURE 5. Impact of filter number of the convolutional layer on TWCSVM performance.

TABLE 5. The convergence of primal problems (NETS-NYPS; $CNN_{1,2} = 3$).

Minimization rate	Epoch[1]	Epoch[2]	Epoch[3]	Epoch[4]	Epoch[5]
$TWSVM_1$	0.0719	0.0693	0.0674	0.0660	0.0648
$TWSVM_2$	2.9484	0.0214	0.0213	0.0213	0.0212

TABLE 6. Classification performance of TWCSVM (T-A test system).

	TWCSVM	Acc/ TPR/ TNR
	CNN_1 parameters	
T-A test case	$N_{co.layer}$/ F.n./ F.s./ pooling strategy/ η/ C 1/ 5/ 312/ mean/ 1.0e-05/ 32	87.27/100 /83.72
	CNN_2 parameters	
	$N_{co.layer}$/ F.n./ F.s./ pooling strategy/ η/ C 1/ 5/ 312/ mean/ 1.0e-07/ 32	

classifier performance. Also, mean-pooling for both CNN_1 and CNN_2 was an optimal choice in terms of pooling strategy even when the different values of parameters (filter size and filter number of the convolutional layer) were selected according to TABLE IV. The variation in the minimization of primal problems of TWCSVM (e.g., $CNN_{1,2} = 3$) when the number of iteration grows to train TWCSVM is shown in TABLE V. The results show the convergence of primal problems minimization. An important point to note is that when TWCSVM framework was developed by two convolutional layers per CNN, not only that classification performance did not improve but learning procedure increased. To evaluate the performance of TWCSVM on T-A system, a set of experiment was conducted based on the different values of important parameters. The best values for TWCSVM parameters that induced a high performance for transient stability prediction of T-A test system are shown in TABLE VI.

2) COMPARISON OF EXPERIMENTAL METHODS: TWCSVM VS. KERNEL METHOD

The main purpose of this section is to compare the performance of TWCSVM with the kernel-based predictive

TABLE 7. Comparing classification performance of TWCSVM with SVM-kernel.

	Classifiers			
	TWCSVM	SVM-RTWK	SVM-RBF (DTW distance)	SVM-RBF (Euclidean distance)
Measure	NE-NY [a]/ T-A	NE-NY/ T-A	NE-NY/ T-A	NE-NY/ T-A
Acc	**97.22**/ **87.27**	91.66/ 86.27	58.33/ 76.47	94.44/ 86.27
TPR	**100**/ **100**	88.89/ 91.3	6.66/ **100**	**100**/ **100**
TNR	**95.24**/ **83.72**	94.44/ 82.14	**95.24**/ 70.73	90.47/ **83.72**

[a] NETS-NYPS

approach on real-time TSA. Regardless of future challenges vis-à-vis kernel function selection issues in SVM; overall, non-linear SVM had acceptable performance on transient stability status prediction. Hence, three efficient kernels for experimental comparison of kernel-based predictive approach on real-time TSA and TWCSVM are as follows: 1) recursive time warp kernel, called RTWK [27], 2) DTW distance in Gaussian RBF kernel [36] and 3) standard Gaussian RBF kernel [37]. Consequently, by applying changes to SVM formulations and acceptance of trajectory data and plugging of kernels into it, classification in trajectory feature space was performed. As can be seen in TABLE VII, TWCSVM outperformed kernel-based approaches on test cases regarding our strict strategy in the testing phase. More analysis based on comparison of the experimental methods is as follows: According to what has been mentioned in [32], NETS-NYPS is larger in both number of areas and installed types of equipment from T-A test system. The obtained result showed that when a small topology test network was displaced by a large topology system like NETS-NYPS, SVM-RBF (DTW distance) had lower generalization capacity for real-time TSA with such a size of topology, while TWCSVM, SVM-RTWK and SVM-RBF (Euclidean distance) had high generalization capacity. Another issue was dependency ratio of the methods on the type of test cases and effect of this issue on the performance of the techniques as shown in TABLE VIII. To determine which method does not depend on test case type, both ''change rate'' and ''impact on rank'' items should be evaluated. The ''change rate'' indicates the amount of difference in prediction accuracy of the classifier when we switched from X system to Y system. According to TABLE VIII, in terms of ''change rate'', TWCSVM, SVM-RTWK and SVM-RBF (Euclidean distance) had a lower change rate on accuracy than SVM-RBF (DTW distance) (underline face) when we switched from T-A system to NETS-NYPS system and vice versa. This result means that alteration in test case type (or type of topology) does not have a significant impact on the performance of these techniques. Next item is ''impact on rank'' which was derived from the first item. In fact, test case switching created a change in the accuracy of the classifier and may reduce its performance ranking. According to ''impact on rank'' column, TWCSVM is situated at top

TABLE 8. Effect of test case type on the performance of learning methods.

Classifier	Kernel	Items				
		From	To	Change rate	Rate type	Impact on rank (From: To)
TWCSVM	None	NE-NY	T-A	9.95	Neg.	**1:1**
		T-A	NE-NY	9.95	Pos.	**1:1**
SVM^{Kernel}	RTWK	NE-NY	T-A	5.39	Neg.	3:2
		T-A	NE-NY	5.39	Pos.	2:3
	RBF^{DTW}	NE-NY	T-A	18.14	Pos.	4:3
		T-A	NE-NY	18.14	Neg.	3:4
	$RBF^{Euclidean}$	NE-NY	T-A	8.17	Neg.	2:2
		T-A	NE-NY	8.17	Pos.	2:2

TABLE 9. The prediction time of classifiers.

	TWCSVM	SVM-RTWK	SVM-RBF (DTW distance)	SVM-RBF (Euclidean distance.)
Prediction time	0.808 s	864 s	5.625 s	0.105 s

rank even when test case type is switched and thus works better than SVM-RTWK and SVM-RBF (Euclidean distance) (bold-face). As a final report (See TABLE IX) based on experiments, SVM-RBF (Euclidean distance) is an algorithm with low prediction time, that gained the first rank while our proposed classifier ranked 2nd.

3) ASSESSING THE PERFORMANCE OF PREDICTION MODELS WITH TIMELY OBSERVATION WINDOW (OW)

Since transient stability is always a fast dynamic phenomenon, it is crucial to quickly detect instability. Using large observation window is one of the main factors that induce accurate classification on PSs&tInd transient stability prediction in previous sections, but it is an untimely assessment. Therefore, using timely OW for PSs&tInd transient stability prediction is an important issue. Taking into cognizance this point, assessing the performance of the learning models with a smaller size window was considered in this section. As can be seen in TABLE IX, $SVM\text{-}RBF^{Euclidean}$ and our proposed classifier (TWCSVM) has lower prediction time than other classification techniques. Therefore, training and testing procedures with length of trajectory features in 10 cycles were conducted using these algorithms. According to TABLE X, TWCSVM outperformed $SVM\text{-}RBF^{Euclidean}$ on Acc and TNR for PSs&tInd transient stability prediction on T-A test system. However, $SVM\text{-}RBF^{Euclidean}$ has a lower processing time (234 milliseconds) than TWCSVM (755 milliseconds). Therefore, the processing time difference between TWCSVM and $SVM\text{-}RBF^{Euclidean}$ ($755 - 234 = 521$ milliseconds) was added to the length of trajectory features in $SVM\text{-}RBF^{Euclidean}$ learning model (in row 3; underline face). Although increasing size of OW from 10 to 41 affects the classification performance of $SVM\text{-}RBF^{Euclidean}$ on Acc and TNR in a positive manner; overall, TWCSVM outperforms

TABLE 10. Training databases with smaller size window of trajectory features (T-A test system).

Classifier	Length of features in cycle (c)	Length of features in second (t_c)	Acc/ TNR/ TPR/ PT[a]
TWCSVM	10	0.167	86.27/ 88.37/ 75/ 167 ms[b]+588 ms
SVM-RBF (Euclidean distance)	10	0.167	54.9/ 46.51/ 100/ 167 ms+67 ms
	41	0.688	62.74/ 55.81/ 100/ 688 ms+69 ms

[a] Processing Time=OW+prediction time, [b] millisecond

SVM-RBF$^{\text{Euclidean}}$ for PSs&tInd transient stability prediction on T-A test system.

IV. CONCLUSION AND FUTURE WORK

This study was aimed at surmounting some concerns raised in this paper on real-time TSA. For this purpose, first, a set of trajectory features that were independent of power system size & type (s&tIF) were extracted based on the basic reasons associated with their definition. Definition of this type of features allowed us to achieve learning model generalizability on real-time TSA as an urgent need for network expansion planning in today's power system environment. As seen in the second part of the proposed methodology, we presented a twin convolutional support vector machine as supervised trajectory-based deep neural classifier, which can help us to eliminate the computational complexity of kernel trick. Finally, simulations were applied to test networks with different topology. The results showed that the classification accuracy of the proposed method with a larger size window for each test systems exceeded 87% and it outperformed kernel-based approaches on test cases. More focus on obtained results showed that TWCSVM, SVM-RTWK and SVM-RBF (Euclidean distance) had high generalization capacity based on Acc measures (+9.95, +5.39, +8.17; respectively), while and SVM-RBF (DTW distance) had low generalization capacity (-18.14) for PSs&tInd real-time TSA. After obtaining positive results on PSs&tInd transient stability prediction based on larger observation window, assessing the effect of length of trajectory features on the performance of learning models was considered. Since SVM-RBF$^{\text{Euclidean}}$ and our proposed classifier (TWCSVM) had lower prediction time than other classification techniques, training and testing procedures with smaller observation window were conducted using these algorithms. The obtained result showed this fact that our proposed methodology with the length of trajectory features in 10 cycles had good performance (Acc: 86.27 %, processing time: 755 ms) for PSs&tInd transient stability prediction.

In future studies, unseen condition in the real environment of power systems like information missing due to the communication failure (unavailability) and lack of quality of power system dynamic responses (noisy data) should be considered. Also, assessing the model generalization performance on PSs&tInd transient stability prediction under system operating condition requires further study.

APPENDIX A

Reference [30] presented non-parallel planes classifier, called TWSVM, that searched for hyperplane close to points of one class and at a distance from points of the other class. In fact, unlike primal problem of SVM in which we faced a major optimization problem, TWSVM formulation had two smaller primal problems and each constraint had certain rules as part of the patterns. Consider training vectors belonging to two separate classes:

$$D = \{(x_i, y_i) | x_i \in \mathbb{R}^n, y_i \in \{+1, -1\}, i = 1, \ldots, p+q\} \quad (13)$$

Let $P = (x_1, \ldots, x_p)^T \in \mathbb{R}^{p \times n}$, $Q = (x_{p+1}, \ldots, x_{p+q})^T \in \mathbb{R}^{q \times n}$.

TWSVM classifier without kernel trick is obtained by solving the following pair of primal problems:

$$TWSVM1: \begin{cases} \min\limits_{w_1, b_1, \xi_2} \dfrac{1}{2}||Pw_1 + e_1 b_1||^2 + c_1 e_2^T \xi_2 \\ s.t. -(Qw_1 + e_2 b_1) + \xi_2 \geq e_2, \xi_2 \geq 0 \end{cases} \quad (14)$$

$$TWSVM2: \begin{cases} \min\limits_{w_2, b_2, \xi_2} \dfrac{1}{2}||Qw_2 + e_2 b_2||^2 + c_2 e_1^T \xi_1 \\ s.t. (Pw_2 + e_1 b_2) + \xi_1 \geq e_1, \xi_1 \geq 0 \end{cases} \quad (15)$$

where C_1, C_2 are penalty parameters, e_1 and e_2 are vectors of ones, ξ_1 and ξ_2 are slack vectors, $e_1, \xi_1 \in \mathbb{R}^p, e_2, \xi_2 \in \mathbb{R}^q$. To construct nonlinear TWSVM, primal problems are extended by defining kernel functions [30]. Also, to eliminate problem constraints of TWSVM, (14) and (15) is equivalent to the following minimization problems:

$$\min_{w_1, b_1} \frac{1}{2}||Pw_1 + e_1 b_1||^2 + c_1 e_2^T \sum_{i=1}^{q} [\max(e_2 + (Qw_1 + e_2 b_1), 0)]^2 \quad (16)$$

$$\min_{w_2, b_2} \frac{1}{2}||Qw_2 + e_2 b_2||^2 + c_2 e_1^T \sum_{i=1}^{q} [\max(e_1 - (Pw_2 + e_1 b_2), 0)]^2 \quad (17)$$

APPENDIX B

If a power system utility wishes to implement the proposed algorithm in their power system, the following steps should be considered: 1) power system utility must extract the trajectory features according to TABLE I for the desired power system. 2) the parameters like weight matrix and bias vector in TWCSVM classifier (for more information on learning parameters refer to 6 steps in the learning process of TWCSVM) which are trained during the learning model process and stored under the title of "trained parameters package" is used for the testing procedure. 3) In the testing procedure, for example, according to steps of TWCSVM, unseen sample is entered to TWCSVM for determining the class label. When the unseen sample is entered to the classifier, the parameters of the classifier is set to the values

of the trained parameters (which was stored in the trained parameters package after performing training procedure).

REFERENCES

[1] P. Kundur *et al.*, "Definition and classification of power system stability," *IEEE Trans. Power Syst.*, vol. 19, no. 2, pp. 1387–1401, Aug. 2004.

[2] P. Bhui and N. Senroy, "Real-time prediction and control of transient stability using transient energy function," *IEEE Trans. Power Syst.*, vol. 32, no. 2, pp. 923–934, Mar. 2017.

[3] E. V. de LorencI, A. C. Z. de Souza, and B. I. L. Lopes, "Energy function applied to voltage stability studies—Discussion on low voltage solutions with the help of tangent vector," *Electr. Power Syst. Res.*, vol. 141, pp. 290–299, Dec. 2016.

[4] L. S. Moulin, A. P. A. da Silva, M. A. El-Sharkawi, and R. J. Marks, II, "Support vector machines for transient stability analysis of large-scale power systems," *IEEE Trans. Power Syst.*, vol. 19, no. 2, pp. 818–825, May 2004.

[5] A. D. Rajapakse, F. Gomez, K. Nanayakkara, P. A. Crossley, and V. V. Terzija, "Rotor angle instability prediction using post-disturbance voltage trajectories," *IEEE Trans. Power Syst.*, vol. 25, no. 2, pp. 947–956, May 2010.

[6] F. R. Gomez, A. D. Rajapakse, U. D. Annakkage, and I. T. Fernando, "Support vector machine-based algorithm for post-fault transient stability status prediction using synchronized measurements," *IEEE Trans. Power Syst.*, vol. 26, no. 3, pp. 1474–1483, Aug. 2011.

[7] C.-W. Liu, M.-C. Su, S.-S. Tsay, and Y.-J. Wang, "Application of a novel fuzzy neural network to real-time transient stability swings prediction based on synchronized phasor measurements," *IEEE Trans. Power Syst.*, vol. 14, no. 2, pp. 685–692, May 1999.

[8] Y. Li and Z. Yang, "Application of EOS-ELM with binary Jaya-based feature selection to real-time transient stability assessment using PMU data," *IEEE Access*, vol. 5, pp. 23092–23101, Oct. 2017.

[9] M. M. Eladany, A. A. Eldesouky, and A. A. Sallam, "Power system transient stability: An algorithm for assessment and enhancement based on catastrophe theory and FACTS devices," *IEEE Access*, vol. 6, pp. 26424–26437, May 2018.

[10] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, 3rd ed. Waltham, MA, USA: Morgan Kaufmann, 2011.

[11] S. Das, S. P. Singh, and B. K. Panigrahi, "Transmission line fault detection and location using wide area measurements," *Electr. Power Syst. Res.*, vol. 151, pp. 96–105, Oct. 2017.

[12] Y. J. Lin, "Comparison of CART- and MPL-based power system transient stability preventive control," *Int. J. Elect. Power Energy Syst.*, vol. 45, no. 1, pp. 129–136, Feb. 2013. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0142061512005169?via%3Dihub, doi: 10.1016/j.ijepes.2012.08.066.

[13] Q. Gao and S. M. Rovnyak, "Decision trees using synchronized phasor measurements for wide-area response-based Control," *IEEE Trans. Power Syst.*, vol. 26, no. 2, pp. 855–861, May 2011.

[14] H. Supreme, L.-A. Dessaint, I. Kamwa, and A. Heniche-Oussédik, "Development of new predictors based on the concept of center of power for transient and dynamic instability detection," *IEEE Trans. Smart Grid*, vol. 9, no. 4, pp. 3605–3615, Jul. 2018.

[15] D. You, K. Wang, L. Ye, J. Wu, and R. Huang, "Transient stability assessment of power system using support vector machine with generator combinatorial trajectories inputs," *Int. J. Elect. Power Energy Syst.*, vol. 44, no. 1, pp. 318–325, Jan. 2013.

[16] A. N. AL-Masri, M. Z. A. A. Kadir, H. Hizam, and N. Mariun, "A novel implementation for generator rotor angle stability prediction using an adaptive artificial neural network application for dynamic security assessment," *IEEE Trans. Power Syst.*, vol. 28, no. 3, pp. 2516–2525, Aug. 2013.

[17] S. A. Siddiqui, K. Verma, K. R. Niazi, and M. Fozdar, "Real-time monitoring of post-fault scenario for determining generator coherency and transient stability through ANN," *IEEE Trans. Ind. Appl.*, vol. 54, no. 1, pp. 685–692, Jan. 2018.

[18] J. Zhao, J. Li, X. Wu, K. Men, C. Hong, and Y. Liu, "A novel real-time transient stability prediction method based on post-disturbance voltage trajectories," in *Proc. Int. Conf. Adv. Power Syst. Automat. Protection*, Beijing, China, Oct. 2011, pp. 730–736.

[19] H. Deng, J. Zhao, Y. Zhang, G. Xu, and X. Wu, "Real-time transient instability detection based on perturbed voltage trajectories," *Int. Trans. Elect. Energy Syst.*, vol. 25, no. 6, pp. 1041–1058, Jun. 2015.

[20] P. K. Modi, S. P. Singh, and J. D. Sharma, "Voltage stability evaluation of power system with FACTS devices using fuzzy neural network," *Eng. Appl. Artif. Intell.*, vol. 20, no. 4, pp. 481–491, Jun. 2007.

[21] J. J. Q. Yu, A. Y. S. Lam, D. J. Hill, and V. O. K. Li, "Delay aware intelligent transient stability assessment system," *IEEE Access*, vol. 5, pp. 17230–17239, Aug. 2017.

[22] T. Guo and J. V. Milanović, "Online identification of power system dynamic signature using PMU measurements and data mining," *IEEE Trans. Power Syst.*, vol. 31, no. 3, pp. 1760–1768, May 2016.

[23] S. Wei, Y. Zhou, and Y. Huang, "Synchronous motor-generator pair to enhance small signal and transient stability of power system with high penetration of renewable energy," *IEEE Access*, vol. 5, pp. 11505–11512, Jun. 2017.

[24] A. B. Mousavi, H. Hosseini, and A. Amiri, "Learning model generalizability on power system size & type independent transient stability prediction," *Electr. Power Compon. Syst.*, to be published.

[25] M. Müller, *Information Retrieval for Music and Motion*, 1st ed. Berlin, Germany: Springer, 2007.

[26] H. Lei and B. A. Sun, "A study on the dynamic time warping in kernel machines," in *Proc. Int. IEEE Conf. Signal-Image Technol. Internet-Based System*, Shanghai, China, Dec. 2007, pp. 839–845.

[27] P.-F. Marteau and S. Gibet, "On recursive edit distance kernels with application to time series classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 6, pp. 1121–1133, Jun. 2015.

[28] Y. LeCun, P. Haffner, L. Bottou, and Y. Bengio, "Object recognition with gradient-based learning," in *Shape, Contour and Grouping in Computer Vision*, D. A. Forsyth, Ed. Berlin, Germany: Springer-Verlag, 1999, pp. 319–345.

[29] B. Zhao, H. Lu, S. Chen, J. Liu, and D. Wu, "Convolutional neural networks for time series classification," *J. Syst. Eng. Electron.*, vol. 28, no. 1, pp. 162–169, Feb. 2017.

[30] Jayadeva, R. Khemchandani, and S. Chandra, "Twin support vector machines for pattern classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 5, pp. 905–910, May 2007. [Online]. Available: http://ctech.iitd.ac.in/jayadeva.html and https://ieeexplore.ieee.org/document/4135685/authors#authors

[31] J. Han and C. Moraga, "The influence of the sigmoid function parameters on the speed of backpropagation learning," in *From Natural to Artificial Neural Computation*, vol. 930. Berlin, Germany: Springer, 1995, pp. 195–201.

[32] R. A. Ramos, "Benchmark systems for small-signal stability analysis and control," IEEE Power Energy Soc., Piscataway, NJ, USA, Tech. Rep. PES-TR18, Aug. 2015, pp. 1–390.

[33] P. Kundur, *Power System Stability and Control*, 1st ed. New York, NY, USA: McGraw-Hill, 1994.

[34] N. Samman *et al.*, "Dynamic contingency analysis tools—Phase 1," Pacific Northwest Nat. Lab., Richland, WA, USA, Tech. Rep. PNNL-24843, Nov. 2015.

[35] C. Grigg *et al.*, "The IEEE reliability test system-1996. A report prepared by the reliability test system task force of the application of probability methods subcommittee," *IEEE Trans. Power Syst.*, vol. 14, no. 3, pp. 1010–1020, Aug. 1999.

[36] H. Shimodaira, K.-I. Noma, M. Nakai, and S. Sagayama, "Support vector machine with dynamic time-alignment kernel for speech recognition," in *Proc. 7th Eur. Conf. Speech Commun. Technol. EUROSPEECH Scandinavia*, Aalborg, Denmark, Jan. 2001, pp. 1841–1844.

[37] C. Cortes and V. Vapnik, "Support vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Sep. 1995.

ALIREZA BASHIRI MOSAVI was born in Zanjan, Iran, in 1989. He received the B.Sc. degree in computer engineering from the Azad University of Zanjan, Zanjan, in 2010, and the M.S. degree in information technology engineering from the University of Qom, Qom, Iran, in 2013. He is currently pursuing the Ph.D. degree in computer engineering (artificial intelligence and robotics) with the University of Zanjan, Zanjan. His current research interests include data mining, machine learning, pattern recognition, and power system transient stability analysis.

ALI AMIRI received the B.Sc. degree in computer engineering from the Azad University of Zanjan, Zanjan, Iran, in 2004, and the M.Sc. degree in computer engineering and the Ph.D. degree in computer engineering (artificial intelligence and robotics) from the Iran University of Science and Technology, Tehran, Iran, in 2006 and 2011, respectively. He held several positions, such as the Vice Dean of education with the Engineering Faculty, University of Zanjan. In 2008, he joined the University of Zanjan, Zanjan, where he is currently an Assistant Professor. He has authored two books and more than 14 articles. His current research interests include data mining, pattern recognition, image processing, and parallel algorithm.

SEYED HADI HOSSEINI received the B.Sc. degree in power engineering from the University of Tehran, Tehran, Iran, in 1997, and the M.Sc. and Ph.D. degrees in power engineering from Tarbiat Modares University, Tehran, in 2000 and 2005, respectively. He is currently an Assistant Professor with the University of Zanjan, Zanjan, Iran. He is also the Vice President of students' affairs with the University of Zanjan. He has authored nine journals. He has supervised nine research projects in the field of the power system in Iran. His current research interests include planning the development of power grids, design, troubleshooting and the monitoring of power transformers, and power system reliability.