Transient Stability Assessment Via Decision Trees and Multivariate Adaptive Regression Splines

Matin Rahmatian[a], Yu Christine Chen[a], Atefeh Palizban, Ali Moshref[b], William G. Dunford[a]

[a]*Department of Electrical and Computer Engineering, The University of British Columbia, Vancouver, Canada*
[b]*Power Systems and Testing, BBA Inc., Vancouver, Canada*

Abstract

This paper focuses on the practical implementation of online transient stability assessment (TSA) tools that employ, in conjunction with high-speed synchronized phasor measurements obtained from phasor measurement units (PMUs), classification and regression trees (CART) and multivariate adaptive regression splines (MARS) models. To build CART and MARS models that are amenable to real-time applications, pertinent transient stability-related system characteristics are identified; these include voltage and current phasors, deviations from the centre-of-inertia angle and speed, and potential- and kinetic-energy related quantities. These characteristic quantities are evaluated using PMU measurements and then leveraged to train CART and MARS models for the full Western Electricity Coordinating Council (WECC) system. The resultant models are tested and validated with the full WECC system using credible contingency scenarios in the BC Hydro subsystem. High prediction accuracy rates are observed for both CART and MARS methods, making them attractive options for real-time TSA.

Keywords: Classification and regression trees, decision trees, multivariate adaptive regression splines, phasor measurement units, real-time transient stability assessment, synchrophasors

1. Introduction

Real-time dynamic security assessment (DSA) is a challenging issue in modern electric power systems. Uncertainties arising from renewable electric sources, increased electricity demand, and recent blackouts all highlight the need for improved tools to monitor dynamic security in real time [1]. As an enabling technology for accomplishing this goal, phasor measurement units (PMUs) provide synchronized voltage and current phasor measurements with high sampling rate in the milliseconds range [2]. These phasor measurements are collected locally with phasor data concentrators and then transmitted to a central decision maker. Such an architecture, along with fast communication infrastructure, has the potential of realizing accurate and efficient real-time DSA [3]. The broad class of DSA includes assessment of system security with respect to various types of stability, including voltage, transient, and small-signal stability. The focus of this paper is on transient stability assessment (TSA), which is related to the time evolution of a dynamic power system trajectory when subject to a large perturbation, e.g., a fault in a transmission line [4]. (We interchangeably refer to faults as contingencies.) The ability to conduct real-time TSA would enable power systems to operate closer to their limits since the effects of unexpected faults could be accurately assessed in a timely manner.

Given the obvious utility of real-time TSA tools, numerous approaches have been explored in the literature. These approaches can be divided into three main categories: time-domain simulations, direct approaches, and automatic learning methods. The main drawback of time-domain simulation approaches is that they are computational burdensome [5]. In direct approaches, energy functions are difficult to construct for practical large-scale systems [6–8]. On the other hand, recent work has revealed automatic learning methods to be promising for practical real-time TSA [9–15]. For example, in [9], a comprehensive database of the Hydro-Québec power system along with relevant contingency cases are constructed and a decision tree (DT) is used to classify stable vs. unstable scenarios. A random forest classifier is shown to operate efficiently in the presence of small changes in the network topology in [16]. In [10], regression trees are used to indicate the number of overloaded lines and buses with voltage magnitude violations following a fault. In [11], application of multiple DTs leads to improved reliability for determining stable versus unstable scenarios. The support vector machine method is used as the classifier in [14] where energy-based power system features are applied, and accurate prediction rates are demonstrated, albeit for small-scale systems (New England 39-bus power system model) only. In [12] and [17], in order to tailor to online applications, ensemble learning machine is used to decrease the required training and decision-making time.

In this paper, we focus on the *practical* implementation of classification and regression models to assess transient stability in real time using PMU measurements obtained from the system. The classification model is developed using classification and regression trees (CART), which is a well-known DT algorithm [18]. We then use the developed model to classify test cases as either stable or unstable with high degree of accuracy. In addition to CART, which returns discrete-valued stable versus unstable classifications, we propose to use mul-

tivariate adaptive regression splines (MARS) [19] in conjunction with a continuous-valued transient stability index (TSI) to assess the severity of the contingency and the system's proximity to instability. Application of this method can be useful to determine probable corrective actions. Development of both CART and MARS models proceeds as follows. First, we identify system characteristics that are pertinent to TSA, including voltage and current phasors, deviations from centre-of-inertia angle and speed, and potential- and kinetic-energy related quantities. Next, these characteristic quantities, called "indices" or "features", are evaluated using synthetic measurements obtained from time-domain simulations of the full Western Electricity Coordinating Council (WECC) system. These evaluated quantities are used to build classification and regression models, which are then tested and verified on the WECC system, with emphasis on transient stability issues in the BC Hydro power system that may emerge due to high power exchange rates with the Bonneville Power Administration (BPA) system. We also show the necessity and practicality of updating the obtained models after topology changes or in the presence of high penetration of renewable resources.

To validate the proposed CART and MARS methods in a practical setting, we conduct test cases on the full WECC system model, with emphasis on the BC Hydro power system. It is worth noting that this is the same detailed model that BC Hydro uses for its planning and operations studies. Mindful of practical field implementations of the proposed methods, our studies assume limited number of PMUs installed, in accordance with existing infrastructure in the BC Hydro system. Similarly, we assume only large generating plants of the neighbouring regions are equipped with PMUs. As a consequence of the limitation in the number of installed PMUs, some features, such as the location of the fault highlighted in [20], may not be readily available. Moreover, certain features proposed in the literature may not be available in *real time*, such as the duration of the fault and the time to loss of synchronism considered in [21]. Since this paper focuses on the practical implementation of CART and MARS methods, features used in our studies are carefully selected to ensure they are available in real time while being cognizant of existing infrastructure limitations. In addition to considering practical limitations, we also take advantage of recent advances in PMU technology and assume that synchrophasor measurements are available at the control centre through wide-area monitoring system (WAMS) at every electrical cycle [22].

This paper builds upon our work in [23] and extends it in several directions. First, the modified data set includes 200 power-flow scenarios and 3 different topologies leading to 9474 different simulation cases, which is more comprehensive than the data set used in [23] and highlights the necessity for an adaptive framework. Second, MARS is used to fit a closed-form equation for the transient stability margin of the system. Furthermore, we test the resulting CART and MARS models with data obtained at sampling times other than used to train the models in order to uncover their sensitivity to erroneous data fed from a later or earlier sampling time. Finally, we demonstrate the sensitivity of MARS model prediction accuracy with respect to two relevant parameters.

The remainder of the paper is organized as follows. Section 2 outlines TSA indices used to build CART and MARS models. In Section 3, fundamentals of CART and MARS model building procedures are described. Section 4 presents WECC system specifications, data set generation, PMU measurement sampling, and model training. Simulation results are reported and the necessity for updating models is discussed in Section 5. Finally, Section 6 offers concluding remarks and directions for future work.

2. Transient Stability Assessment Indices

The first step to building classification and regression models for TSA is to identify system characteristics that are relevant to determining transient stability. With regard to this, numerous features have been used in the literature, including direct phasor measurements and computed quantities involving these measurements [9–12, 16, 21]. (We interchangeably refer to features as indices.) In this section, we enumerate and describe TSA indices used in our work.

2.1. Direct Measurement Indices

Most commonly used indices are time-synchronized measurements of, e.g., generator rotor angles, rotor speeds, bus relative voltage phase angles and magnitudes, and current flows [10–12, 21]. Note that generator rotor and voltage phase angles are obtained as relative angles with respect to a system reference. Other basic indices are obtained from simple calculations involving voltage and current phasors, such as active and reactive generation, active and reactive load, and voltage drop across transmission lines [10–12].

Based on the practical availability of WAMS, we select several synchronized measurements from the pre- and post-fault systems and use them as TSA indices. In a practical power system, major power plants with large generation capacity as well as critical transmission corridors are candidates for PMU installations and real-time monitoring [24]. Under pre-fault normal operating conditions, we collect the total and individual active-power generation of major power plants in each balancing authority region (with respect to the WECC system, the balancing authority regions are BC Hydro, BPA, and Alberta power systems). We also use the relative voltage phase angles of buses connected to these major power plants and those of buses that interface between two regions. Furthermore, we make use of the active-power flows across inter-ties connecting the regions and along critical long transmission lines. Under post-fault conditions, we use only measurements of relative voltage phase angles of buses connected to major power plants and of buses that interface between two regions.

2.2. Centre-of-inertia-referred Indices

Since deviation from the centre-of-inertia (COI) angle can indicate system stress, below, we introduce COI-referred system characteristics. Consider an interconnected power system in which the monitored region has $K-1$ neighbouring regions.

Assume we monitor region k which contains N buses. Also assume region k contains G major synchronous generators that are equipped with PMUs. Denote, by $P_{mi}(t)$ and $P_{ei}(t)$, the mechanical-power input [p.u.] and the electrical-power output [p.u.], respectively, of generator i at time t. Let $\delta_i(t)$ and $\omega_i(t)$ denote the rotor angle [rad] and speed [rad/s], respectively, of generator i at time t; and let M_i denote the inertia constant [s^2/rad] of generator i. Then, the COI angle of the monitored region is defined as [4]

$$\delta_{\text{COI},k}(t) := \frac{1}{M_{\text{total}}} \sum_{i=1}^{G} M_i \delta_i(t), \tag{1}$$

where $M_{\text{total}} = \sum_{i=1}^{G} M_i$; and the COI rotor speed is analogously defined as [4]

$$\omega_{\text{COI},k}(t) := \frac{1}{M_{\text{total}}} \sum_{i=1}^{G} M_i \omega_i(t). \tag{2}$$

Using (1) and (2), define the COI-referred angle and speed of generator i in region k as $\widetilde{\delta}_i(t) = \delta_i(t) - \delta_{\text{COI},k}(t)$ and $\widetilde{\omega}_i(t) = \omega_i(t) - \omega_{\text{COI},k}(t)$, respectively. We next describe COI-referred indices that are used to develop CART and MARS models.

First, we compare the pre- and post-fault conditions and propose to use the following centre-deviation index:

$$\Delta\delta(t) = \frac{\delta_{\text{COI},k}(t) - \delta_{\text{COI},k}^{\text{pre-fault}}}{\delta_{\text{COI},k}^{\text{pre-fault}}}, \tag{3}$$

where $\delta_{\text{COI},k}^{\text{pre-fault}}$ represents the pre-fault COI angle of the monitored region k. Moreover, based on energy function concepts, products involving post-fault power mismatches and angle and speed deviations are good indicators of system stress [25]. Thus, we also apply the following indices [9, 25]:

$$\alpha_k(t) = \sum_{i=1}^{G} f_i(t)\widetilde{\omega}_i(t), \tag{4}$$

where $f_i(t) = P_{mi}(t) - P_{ei}(t) - \frac{M_i}{M_{\text{total}}} P_{\text{COI},k}(t)$, with $P_{\text{COI},k}(t) = \sum_{i=1}^{G}(P_{mi}(t) - P_{ei}(t))$,

$$\beta_k(t) = \sum_{i=1}^{G} f_i(t)\widetilde{\delta}_i(t),\ \gamma_k(t) = \sum_{i=1}^{G} \widetilde{\omega}_i(t)\left(\widetilde{\delta}_i(t) - \widetilde{\delta}_i^{\text{pre-fault}}\right), \tag{5}$$

where $\widetilde{\delta}_i^{\text{pre-fault}}$ denotes the pre-fault COI-referred angle of generator i. In addition to the indices described in (3)–(5), we modify $\gamma_k(t)$ in (5) to reflect the behaviour of the COI angle and speed, and include the following index:

$$\sigma_k(t) = \omega_{\text{COI},k}(t)\left(\delta_{\text{COI},k}(t) - \delta_{\text{COI},k}^{\text{pre-fault}}\right). \tag{6}$$

Not captured by indices in (3)–(6), we may wish to penalize generators whose rotor angles deviate further from the COI angle. Thus, aimed at an index that is sensitive to loss of synchronism, we make use of the integral square generator angle (ISGA) index, which is expressed as [26]

$$J = \frac{1}{T} \int_0^T \frac{1}{M_{\text{total}}} \sum_{i=1}^{G} M_i \left(\delta_i(t) - \delta_{\text{COI},k}(t)\right)^2 dt, \tag{7}$$

where T represents the observation time window. Note that, in (7), the ISGA index is normalized with respect to both M_{total} as well as T. Thus, in the case of disconnection of any generator, its contribution to both M_{total} and $\delta_{\text{COI},k}(t)$ must be removed from (7).

To assess transient stability, it is not only important to consider characteristics of the monitored region, but to also account for interactions between the monitored region and its neighbours. To this end, we consider the centre-of-angle stability index, expressed as [27]

$$\Delta\delta_{k,j}(t) = \left| \frac{\delta_{\text{COI},k}(t) - \delta_{\text{COI},j}(t)}{\delta_{\text{COI},k}^{\text{pre-fault}} + \delta_{\text{COI},j}^{\text{pre-fault}}} \right| \times 100, \tag{8}$$

which can be interpreted as the angle deviation between regions k and j, $k \neq j$.

Rotor angles and speeds are very important indicators of power system transient stability [4]. The phase angle and frequency differences between two buses are combined in [28] to calculate the rms-coherency between those buses in order to partition a system. Inspired by this, to assess the coherency between region k and its neighbouring regions following a fault, we consider COI angle and rotor speed of regions k and j and calculate the rms-coherency between them, as follows:

$$\psi_{k,j} = \left[\frac{1}{T} \int_0^T \Big(\left(\delta_{\text{COI},k}(t) - \delta_{\text{COI},j}(t)\right)^2 + \left(\omega_{\text{COI},k}(t) - \omega_{\text{COI},j}(t)\right)^2 \Big) dt \right]^{1/2}, \tag{9}$$

where T represents the observation time window.

To summarize, we consider COI-referred indices in (3)–(9) in training CART and MARS models for real-time TSA.

2.3. Energy Function-based Indices

Assuming the remainder of the system can be modelled as an infinite bus, the dynamics of generator i can be expressed using its swing equation [4]

$$M_i \frac{d^2\delta_i}{dt^2} = P_{mi} - P_{ei}. \tag{10}$$

From (10), we recover the kinetic energy function $V_i(t)$ of generator i as [4, 29]

$$V_i(t) = \frac{1}{2} M_i \left(\frac{d\delta_i}{dt}\right)^2, \tag{11}$$

and its potential energy function $U_i(t)$ as

$$U_i(t) = \int_0^t \left(-P_{mi} + P_{ei}(\tau)\right) \frac{d\delta_i}{d\tau} d\tau. \tag{12}$$

In Section 4, our simulations assume that $P_{ei}(\tau)$ measurements can be calculated from PMU measurements, from which $U_i(t)$ can be computed by numerical integration.

Based on (12), as an indicator of transient stability in region k, we propose to use the following total potential energy index:

$$U_{\text{total},k}(t) = \sum_{i=1}^{G} U_i(t), \tag{13}$$

where $U_i(t)$ is computed numerically. Similarly, from (11), we obtain the total kinetic energy in region k as

$$V_{\text{total},k}(t) = \sum_{i=1}^{G} V_i(t). \tag{14}$$

As discussed in Section 2.2, the interactions between the monitored region and its neighbours are also critical for transient stability assessment. To this end, we propose the kinetic energy separation index, formulated as

$$\Delta V_{k,j}(t) = |V_{\text{total},k}(t) - V_{\text{total},j}(t)|, \tag{15}$$

to be used in our proposed TSA algorithms.

In summary, we consider energy-based features in (13)–(15) in training CART and MARS models for real-time TSA.

3. Data Mining Fundamentals

In this section, we describe classification and regression tree (CART) as the classification method and multivariate adaptive regression splines (MARS) as the regression method used in the paper. In order to build and test the models from these methods, S time-domain simulation cases are obtained by combining different faults and operating conditions. For each case $i = 1, \ldots, S$, we obtain or compute numerical values for R TSA features or indices. As described in Section 2, in the context of this work, these include direct measurement, COI-referred, and energy function-based indices. Collect feature values for each simulation case in the $S \times R$ matrix $X = [x_{ir}]$, where x_{ir} represents the value of feature r for case i. Denote the entire data set by $\mathcal{D} = \{(e_i^{\mathrm{T}} X, y_i), i = 1, 2, \ldots, S\}$, where $e_i \in \mathbb{R}^S$ denotes a column vector of all zeros except in the ith entry, which is equal to 1, and y_i is a target variable related to the stability of simulation case i (we elaborate on this later). To build a CART or MARS model, the data set $\mathcal{D}$ is divided into two subsets. The training set, denoted by $\mathcal{T}$ consisting of $S_{\mathcal{T}}$ simulation cases, is used to train the CART and MARS models. The remainder partition $\mathcal{D} \setminus \mathcal{T}$, called the test set, is used to ensure the validity of the CART and MARS models built using $\mathcal{T}$.

3.1. Classification and Regression Tree

In order to build the CART model, denote the target variable for case i by y_i, which takes the value of either 0 or 1, representing unstable or stable case, respectively. The objective is to construct a decision tree model from the training set and subsequently, use it for prediction of new unseen simulation cases involving other credible contingencies. Such a tree is constructed in three stages: (i) tree growing, (ii) tree pruning, and (iii) optimal tree selection. These stages are summarized below. Interested readers may refer to [18] for more details.

3.1.1. Tree Growing

Beginning from a root node, a feature, called the "splitting feature", is selected and a value for that feature, called the "splitting value", is chosen to divide the training into two subsets. The result is two child nodes, either internal or terminal. An internal child node becomes a parent node, at which another splitting feature and corresponding splitting value are selected. This sequential splitting procedure eventually leads to terminal nodes called "leaf nodes". Each leaf node is classified as either stable or unstable. A path from the root node to a leaf node is characterized by a sequence of yes/no questions. In this manner, the CART model is simple and transparent to interpret.

A node is considered "pure" when a majority of cases corresponding to that node are associated with one class label, either stable or unstable. The partitioning process described above should lead to increase in nodes' purity. To accomplish this aim, node impurity functions in (16) and (17) are defined in such a way to be maximized if the cases belonging to a node include an equal number of stable and unstable cases, making it difficult to classify the node as "stable" or "unstable" with any confidence. Conversely, the impurity function is minimized if all cases belonging to the node are either stable or unstable, so that the node can be classified with certainty. Based on the intuition above, we describe two possibilities for the node impurity function, one of which is based on the Gini diversity index and the other on the Entropy function [30]. In the context of this work, there are only two possible classes: stable (1) and unstable (0). Thus, denote the probability of stable cases corresponding to node z as $p(z) = \Pr(1|z)$. Then, the impurity function based on the Gini diversity index at node z is [30]

$$i(z) = 2p(z)(1 - p(z)). \tag{16}$$

On the other hand, the impurity function for node z, based on the Entropy function for two classes, is [30]

$$i(z) = -p(z) \log p(z) - (1 - p(z)) \log(1 - p(z)). \tag{17}$$

Indeed, both candidate impurity functions in (16)–(17) are maximized with $p = 0.5$ and minimized with $p = 0$ or 1.

3.1.2. Tree Pruning

Intuitively, the tree's prediction accuracy rate should increase as the number of the nodes increases. However, if the growing process is continued until no two simulation cases belong to the same terminal node, the tree would likely be overfitting the training set, and consequently, would not be a good classifier for the test set. Thus, the tree obtained from the process described in Section 3.1.1 must be pruned. We begin by introducing terminology and variables that are relevant to the pruning process. Let Z represent the classification tree obtained from the tree-growing process described in Section 3.1.1. A node z' is a descendant of node z if there exists a connected path down the tree leading from z to z'. A branch Z_z of Z consists of the node z and all descendants of z, and branch Z_z has z as its root node. Pruning a branch Z_z from a tree Z consists of deleting, from Z, all descendants of z, while retaining node z, which becomes a leaf node of the pruned tree. Suppose Z' is obtained from Z by successively pruning off branches, then $Z' \preceq Z$ is a pruned subtree of Z.

The pruning process identifies branches that contribute least to the model accuracy rate and eliminates them. In order to do so, the misclassification rate needs to be estimated. One of

the most common estimation methods of misclassification rate is the "resubstitution estimate". At node z, the resubstitution estimate, denoted by $\rho(z)$, for a two-class data set, is expressed as [30]

$$\rho(z) = \min\left(p(z), 1 - p(z)\right), \tag{18}$$

where $p(z) = \Pr(1|z)$. Now, let $\widetilde{Z} = \{z_1, z_2, \ldots, z_L\}$ denote the set of $L(Z)$ terminal nodes of classification tree Z and $\pi(z_l)$ the proportion of all simulation cases that belong to node z_l. Based on (18), the resubstitution estimate of the misclassification rate for Z is expressed as

$$\varrho(Z) = \sum_{l=1}^{L(Z)} \rho(z_l)\pi(z_l). \tag{19}$$

The method of *minimum cost-complexity pruning* creates a set of subtrees from the original tree. In this method, the resubstitution estimate in (19) is modified to include a complexity parameter, $\alpha \geq 0$. Then, for any subtree $Z' \prec Z$ the cost-complexity pruning rate is

$$\varrho_\alpha(Z') = \varrho(Z') + \alpha L(Z'), \tag{20}$$

where $\alpha L(Z')$ represents a penalty term for the size of subtree Z', which is not considered in $\varrho(Z')$.

For each value of α, the subtree $Z^*(\alpha)$ of Z that minimizes $\varrho_\alpha(Z')$ is selected as $Z^*(\alpha) = \arg\min_{Z' \preceq Z} \varrho_\alpha(Z')$. The value of α determines the size of the tree. If α is small, then the penalty term in (20) would be small, causing $\varrho(Z')$ to dominate. In turn, the size of the subtree $Z^*(\alpha)$ would be large. Conversely, for sufficiently large values of α, $Z^*(\alpha)$ would consist of the root node z only. It is worth noting that although α is defined on the continuous interval $[0, \infty)$, the number of subtrees of Z is finite. Thus, for a sequence of complexity parameters $0 < \alpha_0 < \alpha_1 < \cdots < \alpha_M$, nested subtrees are obtained as

$$Z \succ Z^*(\alpha_0) \succ Z^*(\alpha_1) \succ \cdots \succ Z^*(\alpha_M). \tag{21}$$

3.1.3. Optimal Tree Selection

From the set of subtrees generated in the pruning procedure, namely $Z^*(\alpha_0), \ldots, Z^*(\alpha_M)$, the subtree with the minimum misclassification rate (as computed from (19)) is selected as the final CART model [30].

3.2. Multivariate Adaptive Regression Splines

Recall that the $S \times R$ matrix X contains the values of each feature $r = 1, \ldots, R$ associated with each case $i = 1, \ldots, S$. Denote feature r as variable w_r, and collect all feature variables into $w = [w_1, \ldots, w_R]^{\mathrm{T}}$. Instead of considering a discrete output of 0 or 1, as in the CART model, here, we use MARS to map the feature variable w to a continuous-valued target variable, which is indicative of system stability. Particularly, we use the transient stability index (TSI) defined as

$$\mathrm{TSI} := \frac{360 - \Delta\delta_{\max}}{360 + \Delta\delta_{\max}} \times 100, \tag{22}$$

where $\Delta\delta_{\max}$ denotes the maximum angle separation between any two generators in the system. Stable and unstable conditions are indicated by $\mathrm{TSI} > 0$ and $\mathrm{TSI} < 0$, respectively [31].

Let y_i denote the TSI metric computed in simulation case i. Then, the goal of MARS is to fit this output observation to a function of features, i.e., $y_i = f(w) + \epsilon$, where $f(w) : \mathbb{R}^R \to \mathbb{R}$, while minimizing ϵ. The MARS method results in a model of the form $f(w) = \sum_{q=0}^{Q} \beta_q h_q(w)$, where $h_q(w)$ denotes a basis function (BF) or a product of two or more BFs (we elaborate on these later), β_q represents a constant multiplicative coefficient, Q denotes the number of additive terms in the final model, and for case i, w is evaluated as $w = [x_{i1}, x_{i2}, \ldots, x_{iR}]$. Similar to the CART algorithm, the MARS method first creates an overfit model and then prunes it back. This process is composed of two stages—forward and backward, which we summarize below [30]. Interested readers may refer to [19] for details.

3.2.1. Forward Procedure

The MARS method first collects pairs of BFs for each feature w_r. The BF may be a constant value or a hinge function. A hinge function takes the form of $\max(0, w_r - x_{ir})$ for $w_r \geq x_{ir}$, and $\max(0, x_{ir} - w_r)$ for $w_r < x_{ir}$, where the constant x_{ir} is called a "knot" and is the value of feature r for a particular simulation case i in the training set. With the above in mind, the pair of BFs for the r^{th} feature and i^{th} knot value is expressed as the set

$$C^{r,i} = \{\max(0, w_r - x_{ir}), \max(0, x_{ir} - w_r)\}. \tag{23}$$

Given R features and $S_{\mathcal{T}}$ training cases, by taking the union of all enumerated basis function sets, $2RS_{\mathcal{T}}$ BFs in $C = \cup_{r=1}^{R} \cup_{i=1}^{S_{\mathcal{T}}} C^{r,i}$ are established. With the set of BFs C in place, let the model obtained at the end of the forward procedure be $f_{Q_{\max}}(w) = \sum_{q=0}^{Q_{\max}} \beta_q h_q(w)$ with $Q_{\max}$ denoting the total number of the added terms in the model. The forward procedure is iterative and begins with a constant function $f_0(w) = h_0(w) = c$ which is set as the mean of the training cases' output, i.e., $c = \frac{1}{S_{\mathcal{T}}} \sum_{i=1}^{S_{\mathcal{T}}} y_i$, in order to minimize ϵ.

At each iteration $q = 1, \ldots, Q_{\max}$, for each $r = 1, \ldots, R$ and simulation case $i = 1, \ldots, S_{\mathcal{T}}$ in the training set, the following function is constructed:

$$h_q^{r,i} = \beta_{q1}^{r,i} h_{q-1} \max(0, w_r - x_{ir}) + \beta_{q2}^{r,i} h_{q-1} \max(0, x_{ir} - w_r) \tag{24}$$

from which the function

$$f_q^{r,i}(w) = f_{q-1}(w) + h_q^{r,i}(w) \tag{25}$$

is formed. In (24), the coefficients $\beta_{q1}^{r,i}$ and $\beta_{q2}^{r,i}$ are calculated based on least-squares criterion as

$$\beta_{q1}^{r,i}, \beta_{q2}^{r,i} = \arg\min \sum_{j=1}^{S_{\mathcal{T}}} \left| y_j - f_q^{r,i}(w)\big|_{w=[x_{j1},\ldots,x_{jR}]^{\mathrm{T}}} \right|^2, \tag{26}$$

where, with slight abuse of notation, $\beta_{q1}^{r,i}$ and $\beta_{q2}^{r,i}$ denote the values that minimize the objective function above. Repeatedly solving the optimization in (26) for each value of r and i yields a pool of candidate functions, $f_q^{r,i}$, each containing the minimizing $\beta_{q1}^{r,i}$ and $\beta_{q2}^{r,i}$. Next, from this pool, $r^\star$ and $i^\star$ are chosen so that $f_q^{r^\star,i^\star}$ leads to the minimum least-squares error, as defined in (26), when compared against all other candidate functions.

Finally, set $f_q(w) = f_{q-1}(w) + h_q^{r^\star, i^\star}(w)$ and remove the corresponding basis functions in the set $C^{r^\star, i^\star}$ from consideration in the next iteration, i.e., set $C = C \setminus C^{r^\star, i^\star}$. The iterative forward procedure is continued until the change in the residual squared error is

$$\sum_{j=1}^{S_\mathcal{T}} \left| y_j - f_q(w)\big|_{w=[x_{j1},\ldots,x_{jR}]^\mathrm{T}} \right|^2 < \epsilon_0, \tag{27}$$

where $\epsilon_0 > 0$ is a predetermined threshold, or until the preset maximum iterations $Q_{\max}$ is reached [19]. This forward procedure is hierarchical, i.e., the terms from the collection C are multiplied by the terms already involved in the model to build a multiway product. In this way, we can identify higher-order interactions between the features [30].

3.2.2. Backward Procedure

The large model obtained at the end of the forward process typically overfits the data set, so, a backward procedure is necessary. In this procedure, terms in $f_{Q_{\max}}$ are removed to successively improve the accuracy of the final resulting MARS model [30]. The iterative backward procedure begins with $f_{Q_{\max}}(w)$ obtained from the forward procedure and initializes $\hat{f}_{\lambda+1}(w) = f_{Q_{\max}}(w)$. At each iteration, $\lambda = Q_{\max}, \ldots, 1$, $\lambda + 1$ functions, each denoted by $\hat{f}_\lambda^\ell(w)$, are formed by removing the ℓ^{th} term from $\hat{f}_{\lambda+1}(w)$, where $\ell = 0, \ldots, \lambda$. Among the $\lambda + 1$ functions, $\hat{f}_\lambda^{\ell^\star}(w)$ is selected, such that

$$\ell^\star = \arg \min_{\ell=0,\ldots,\lambda} \sum_{i=1}^{S_\mathcal{T}} \left| y_i - \hat{f}_\lambda^\ell(w)\big|_{w=[x_{i1},\ldots,x_{iR}]^\mathrm{T}} \right|^2. \tag{28}$$

At the end of each iteration λ, set $\hat{f}_{\lambda+1}(w) = \hat{f}_\lambda^{\ell^\star}(w)$ and repeat the process above. Finally, from the pool of candidate $\hat{f}_\lambda$ functions, $\lambda = 1, \ldots, Q_{\max}$, the final MARS model is chosen as $\hat{f}_{\lambda^\star}(w)$ where

$$\lambda^\star = \arg \min_{\lambda=1,\ldots,Q_{\max}} \frac{\sum_{i=1}^{S_\mathcal{T}} (y_i - \hat{f}_\lambda(w))^2}{(1 - M(\lambda)/S_\mathcal{T})^2}, \tag{29}$$

$M(\lambda)$ can be interpreted as a penalty factor for greater model complexity. Specifically, suppose the model includes F linearly independent BFs, then $M(\lambda)$ is obtained as [30]

$$M(\lambda) = F + c_1(Q_{\max} - 1), \tag{30}$$

where the penalty coefficient c_1 is a user-defined constant. Interested readers may refer to [19] for more details.

4. Training

We consider the full WECC bulk transmission system model from summer 2013, which includes 17724 buses, 22005 transmission lines and transformers, and 170 GW generation capacity [32]. We monitor the BC Hydro (BCH) system, which is connected to two neighbouring regions, the Alberta electricity system (AES) and the Bonneville Power Administration (BPA) system. The interconnection between BCH and BPA is the Northern Intertie, which consists of a 500-kV double-circuit and a 230-kV transmission line, and transfers around 2500 MW between BCH and BPA. In addition, BCH is connected to AES on the east, via two interties, a 500-kV and a 230-kV transmission line, enabling BCH and AES to exchange up to 800 MW.

According to the North American Synchrophasor Initiative, PMUs ought to be installed at major power plants with high generation capacity as well as along critical transmission corridors [24]. The present study assumes that PMU measurements are available at major power plants with high generation capacity in the three power systems (5 in BCH, 12 in BPA, and 7 in AES), the interties between these three regions, and a major transmission corridor transferring power southward from northern BC. This assumption is consistent with the existing infrastructure in the BCH system. Consequently, power flows, angles, voltage magnitudes, and frequencies at these locations listed above are accessible at the control centre.

To cover a wide range of operating conditions, the data set consists of 9474 time-domain simulation cases that include three possible topologies, 200 WECC power-flow base cases from summer 2013, and 48 credible $N - k$ ($k = 1, \ldots, 4$) contingencies in the 500-kV and 230-kV levels. As a standard in the BCH power system, the faults occurring on these transmission voltage levels are cleared after three cycles [33]. Therefore, each simulation assumes that a fault occurs at $t = 100$ ms and lasts for three cycles, i.e., 50 ms, before it is cleared. We conduct these simulations in TSAT [31] and find 1098 cases to be unstable. Based on the simulations, values for TSA indices or features discussed in Section 2 are obtained for different faults under various power flow patterns. Synthetic time-synchronized measurements in the present study are obtained from these time-domain simulations in each of five cycles after fault clearance, which is possible with P-class PMUs [34]. Each electric cycle lasts 16.666 ms, which is the measurement sampling time in our study.

The synchronized measurements of the first five cycles after fault-clearance time, along with pre-fault measurements described in Section 2.1, are used to train five CART and MARS models via the procedures described in Sections 3.1 and 3.2, respectively. The training set consists of 70% of the full data set. The training set is used to train CART and MARS models for the first five cycles of the post-fault instants. The remaining 30% of the data set is applied for testing the accuracy of the trained CART and MARS models. We make use of existing commercial software SPM [35, 36] to build the CART and MARS models from the training set data.

5. Case Study Results

In this section, CART and MARS models from the training stage are used to predict transient stability for the test set data. Relevant results are presented and discussed.

5.1. CART Results

As mentioned in Section 4, five CART models are trained using synthetic measurements of each of the first five cycles after fault clearance, along with pre-fault measurements described in

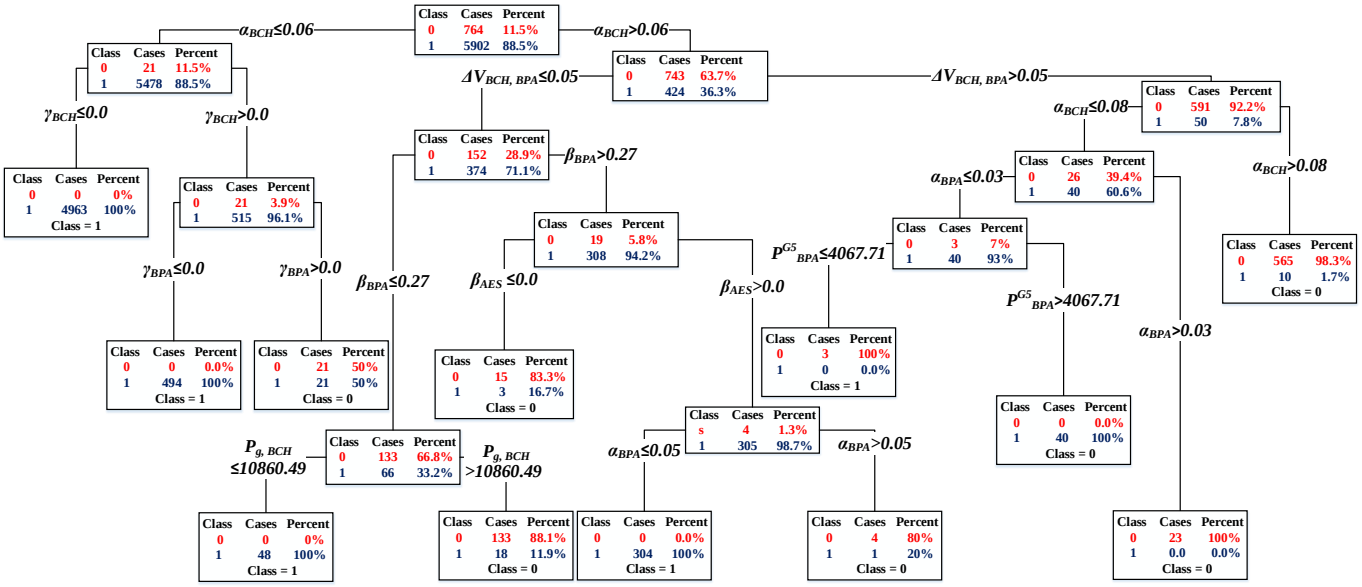

Figure 1: CART model trained and built using synthetic measurements obtained in the fourth cycle after fault clearance.

Table 1: TSA indices selected in CART models built using data obtained in 5 cycles after fault clearance.

Level	CART 1	CART 2	CART 3	CART 4	CART 5
1	α_{BCH}	α_{BCH}	α_{BCH}	α_{BCH}	α_{BCH}
2	σ_{BCH}	$\Delta V_{BCH,BPA}$, $V_{\text{total},BPA}$.	$\Delta V_{BCH,BPA}$.	$\Delta V_{BCH,BPA}$, γ_{BCH}.	J_{AES}, $\Delta\delta_{BCH,BPA}$.
3	α_{BCH}, $V_{\text{total},BCH}$, β_{AES}.	α_{BCH}, β_{AES}.	α_{BCH}, β_{AES}.	α_{BCH}, β_{BPA}, γ_{BPA}.	$\Delta\delta_{BCH}$, α_{BPA}, $\Delta V_{BCH,BPA}$.

Table 2: Accuracy results of obtained CART models when tested using data from various time instants.

Test Data	Trained CART Model				
	CART 1	CART 2	CART 3	CART 4	CART 5
Cycle 1	99.50%	93.98%	80.15%	85.49%	74.87%
Cycle 2	93.38%	99.18%	84.58%	84.58%	82.38%
Cycle 3	96.78%	96.62%	99.61%	91.85%	90.39%
Cycle 4	96.89%	96.71%	95.52%	98.97%	92.90%
Cycle 5	96.41%	94.46%	94.41%	96.04%	99.43%

Table 3: Computational times required for various CART models.

Test Data	Computation Time [s]				
	CART 1	CART 2	CART 3	CART 4	CART 5
Cycle 1	12	16	17	21	20
Cycle 2	18	12	17	21	19
Cycle 3	19	17	12	20	20
Cycle 4	18	18	19	12	19
Cycle 5	17	19	19	19	11

Section 2.1. The CART model obtained from the fourth-cycle measurements is shown in Fig. 1, which is typical of resulting trees from the other four cycles. In Table 1, we summarize the selected features for the top three levels in each of the five CART models. The root node in all five models is the index α_{BCH}. The index $\Delta V_{BCH,BPA}$ emerges in the second level in three of the five models. Moreover, the indices σ_{BCH}, γ_{BCH}, $\Delta\delta_{BCH,BPA}$, and $V_{\text{total},BPA}$ are selected in the second level. All the indices selected in the first three levels are either COI-referred or energy function-based ones. This observation indicates the importance of these indices in TSA. Moreover, indices related to the BCH system play an essential role in the trees' structures.

We report accuracy rates of the resulting CART models in Table 2, in which the CART model built using the indices from each cycle is tested against the data obtained at all five cycles. In other words, the diagonal entries in Table 2 indicate the accuracy of each CART model when data from the expected sampling time is fed to it, while the off-diagonal entries indicate the model accuracy when data from a later or earlier time are used. For accuracy rates other than those shown on the diagonal, the CART models are tested using the whole data set of the other cycle (instead of only the test set). We note that, in Table 2, the accuracy rates of the corresponding cycles (diagonal elements) are about 99%. When the trained CART model is tested against data obtained from one cycle before or after the corresponding cycle, the accuracy rates are around 90% and in some cases close to 95%. These off-diagonal entries are relevant for cases in which fault-clearance time is not detected properly by the PMUs and the data is fed into the wrong tree. Also, considering the off-diagonal entries in Table 2, the decrease in accuracy is particularly noticeable when the training data belongs to cycles 4 and 5, and the testing data belongs to the earlier cycles 1 and 2. Thus, our results suggest that the obtained models are more sensitive when they are trained using data obtained from the later cycles and tested with data from earlier cycles.

The computational times required to obtain the CART models are reported in Table 3. As evident from the table, the time required to obtain a CART model using the data set from the same cycle (diagonal entries) is around 12 s, and it increases to around 19 s in the case of using data from different cycles (off-diagonal entries). This increase is the result of the increase

Table 4: TSA indices selected in MARS models built using data obtained in 5 cycles after fault clearance.

Model	Transient Stability Assessment Indices
MARS 1	$\alpha_{BCH}, \beta_{BCH}, \gamma_{BCH}, \sigma_{BCH}, \alpha_{BPA}, \delta^{BCH,BPA}_{intercon,1}, P^{G2}_{BPA}$.
MARS 2	$\alpha_{BCH}, \beta_{BCH}, \sigma_{BCH}, \omega_{COI,BCH}, V_{\text{total},BCH}, \delta^{BCH,BPA}_{intercon,1}, P^{G3}_{BPA}$.
MARS 3	$\alpha_{BCH}, \sigma_{BCH}, \omega_{COI,BCH}, V_{\text{total},BCH}, U_{\text{total},BCH}, \delta^{BCH,BPA}_{intercon,1}, \psi_{BCH,BPA}$.
MARS 4	$\alpha_{BCH}, \beta_{BCH}, \sigma_{BCH}, U_{\text{total},BCH}, \Delta V_{BCH,BPA}, \beta_{AES}, \delta^{BCH,BPA}_{intercon,1}, \delta^{G3,Pre}_{BPA}$.
MARS 5	$\alpha_{BCH}, \beta_{BCH}, \gamma_{BCH}, \sigma_{BCH}, \delta^{BCH,BPA}_{intercon,1}, \delta^{BCH,BPA}_{intercon,2}$.

in the size of training and testing data sets. Note that the commercial software SPM provides the total time required to train and test the models.

5.1.1. Comparison with the Existing Methods

Here, we compare our CART results in Table 2 with existing studies in the literature. In [15] and [20], authors demonstrate accuracy rates of 97–98% using only pre-fault features. However, all probable faults are pre-defined and used as features to train and test DT models. Similarly, in [37], the fault location is used as a feature to obtain prediction accuracy rate of 97%. Unsurprisingly, the fault type and location are very effective features for assessing transient stability. However, in practical implementation with limited number of PMUs installed, it is impossible to determine the type and location of all credible faults in real time. As shown in Table 2, models developed in this paper lead to similar prediction accuracy rates *without* knowledge of the particular fault type and location.

In [9] and [21], authors use post-fault features obtained 1–2 s after fault clearance. This time is reduced to 150–300 ms in [38]. The accuracy rates reported in these works are in the range of 93% with simple DTs and 99% with ensemble DTs. However, duration of simulation until normal end (in stable cases) or loss of synchronism (in unstable cases) is used as a feature, which is not available in real-time TSA. Models developed in this paper take advantage of new PMU technologies that are capable of transmitting post-fault measurements immediately after fault clearance. As shown in Table 2, prediction accuracy rates resulting from simple DTs are comparable to prior work using the more complicated ensemble DTs *without* knowledge of offline simulation duration.

5.1.2. Adaptive Training

As mentioned in Section 4, we train the CART model based on the synthetic measurements obtained from three different topologies. This can be viewed as having updated the model with two additional topologies. In order to showcase the necessity of updating the CART model when new topologies arise, we train a CART model using only data from one topology and subsequently test this model using data obtained from another topology. In this case, the prediction accuracy rate decreases to 94.62%. This result clearly shows the necessity for an adaptive framework with the capability of updating the classification or regression model after topology changes. Such a framework is also important in the case of high penetration of renewable resources that leads to more frequent changes in operating conditions. To account for these aspects, the trained models can be updated using the adaptive framework described in [20].

Table 5: Accuracy results of obtained MARS models when tested using data from various time instants.

Test Data	Trained MARS Model				
	MARS 1	MARS 2	MARS 3	MARS 4	MARS 5
Cycle 1	5.323	10.72	9.167	10.04	9.786
Cycle 2	5.326	4.998	6.944	7.714	7.645
Cycle 3	8.102	7.678	5.333	6.479	7.034
Cycle 4	11.16	9.657	6.747	5.026	6.672
Cycle 5	15.40	12.30	7.030	7.544	4.834

Table 6: Computational times required for various MARS models with 15 maximum basis functions and 1 maximum interaction.

Test Data	Computation Time [s]				
	MARS 1	MARS 2	MARS 3	MARS 4	MARS 5
Cycle 1	8	13	14	14	14
Cycle 2	13	7	14	14	13
Cycle 3	13	14	7	14	14
Cycle 4	13	14	14	7	13
Cycle 5	13	14	14	14	8

5.2. MARS Results

Similar to CART models, five different MARS models are obtained for the first five cycles after fault clearance. Also in this case, since there are synthetic measurements obtained from three different topologies, we can assume that we have updated the obtained model based on the first topology for two more topologies. First, we select a simple model with a maximum of 15 possible BFs and no interactions between the indices, i.e., multiplication of different BFs. As an example, the model obtained for the first cycle is shown below:

$$\begin{aligned}\text{MARS 1} = &-19.54 - 1.925 \cdot \max(0, 7.709 - \sigma_{BCH}) \\ &- 97.74 \cdot \max(0, \alpha_{BCH} - 0.319) \\ &+ 60.04 \cdot \max(0, 0.319 - \alpha_{BCH}) \\ &- 7.497 \cdot \max(0, \delta^{BCH,BPA}_{intercon,1} - 6.219) \\ &+ 0.665 \cdot \max(0, 6.219 - \delta^{BCH,BPA}_{intercon,1}) \\ &- 10397 \cdot \max(0, \gamma_{BCH} - 0.0138) \\ &+ 890.3 \cdot \max(0, 0.0138 - \gamma_{BCH}) \\ &+ 9.315 \cdot \max(0, \beta_{BCH} + 1.138) \\ &- 60.89 \cdot \max(0, -1.138 - \beta_{BCH}) \\ &+ 50.73 \cdot \max(0, \alpha_{BPA} + 0.050) \\ &+ 3453 \cdot \max(0, -0.050 - \alpha_{BPA}) \\ &- 22.41 \cdot \max(0, \sigma_{BPA} - 8.368) \\ &- 0.057 \cdot \max(0, P^{G2}_{BPA} - 1339.1). \end{aligned} \tag{31}$$

The TSA indices selected in all five MARS models are shown in Table 4. The COI-referred and energy function-based indices have the most participation in the models, especially those related to BCH. Also, $\delta^{BCH,BPA}_{intercon,1}$, the relative angle of the 500 kV bus connecting BCH to BPA, emerges in all five models. On

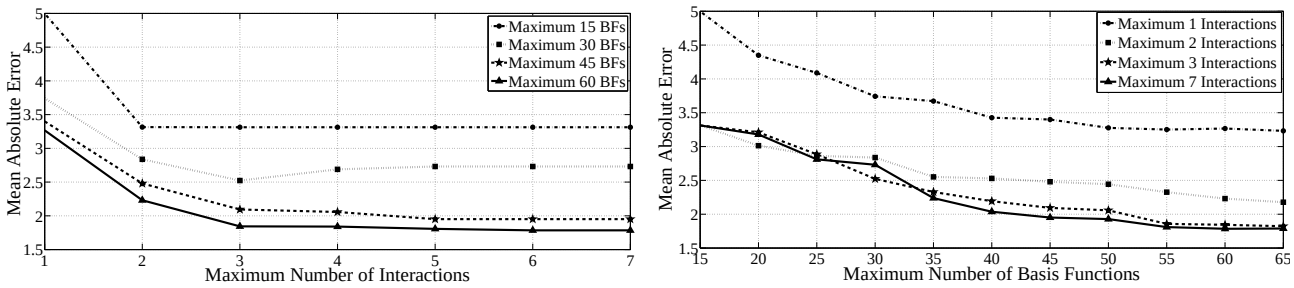

Figure 2: Sensitivity analysis of MARS model trained using synthetic measurements obtained in the second cycle after fault clearance to maximum number of interactions and maximum number of basis functions.

Table 7: Accuracy results of the obtained MARS models with 60 maximum basis functions and 6 maximum interactions.

Model	MARS 1	MARS 2	MARS 3	MARS 4	MARS 5
MAE	2.479	1.784	1.725	2.004	1.982

Table 8: Computational times required for various MARS models with 60 maximum basis functions and 6 maximum interactions.

Model	MARS 1	MARS 2	MARS 3	MARS 4	MARS 5
Computational Time [s]	53	53	49	50	50

the whole, we conclude that COI-referred and energy function-based indices from Sections 2.2 and 2.3 act as important TSA characteristics in both classification and regression, more than direct measurements from Section 2.1.

The MARS model accuracy is evaluated by comparing the TSI values obtained from MARS models on the benchmark computed via (22). The mean absolute errors (MAEs) obtained for each of the five models based on the test set data are reported in Table 5. Also, MAEs are reported for cases in which the test data from a cycle other than the cycle used to train the MARS model is fed to it. As is evident from Table 5, the error is around 6 for all the models when the test data is from the correct cycle. When the test data is from one cycle prior or after, MAE usually increases to not more than 8. According to Table 6, the computational time required to build MARS models with 15 maximum BFs and 1 maximum interaction is approximately 8 s using the data set from the same cycle (diagonal entries). However, the computational time increases to about 14 s in the case of using training and testing data from different cycles (off-diagonal entries). Note, again, that SPM provides the total time required for the training and testing stages.

In an effort to reduce MAEs in the MARS model, we have the option of increasing the maximum number of BFs and the maximum number of interactions. Increasing both of these parameters in the models makes them more complex, but reduced MAEs can be achieved. We vary the maximum number of BFs in the range of 15 to 65 and the maximum number of interactions in the range of 1 to 7 and obtain MARS models for each case. In Fig. 2, we plot the sensitivity of model accuracy to the maximum number of interactions and BFs for the MARS model trained using data obtained from the second cycle after fault clearance. According to Fig. 2, as the maximum number of interactions increases to 3 or more and the maximum number of BFs increases to 50 or more, the MAEs decrease to less than 2, significantly lower than the MAEs resulting from the simple models reported in Table 5. In Table 7, we report MAEs for MARS models obtained with 60 BFs and 6 interactions. Also, referring to Table 8, we find that the computational time required to obtain the MARS models with 60 maximum BFs and 6 maximum interactions is approximately 50 s.

Table 9: Accuracy results of obtained CART models when white Gaussian noise is added to data set.

SNR [dB]	CART 1	CART 2	CART 3	CART 4	CART 5
No noise	99.50%	99.18%	99.61%	98.92%	99.43%
40	99.00%	98.93%	98.72%	98.58%	99.04%
30	98.93%	98.11%	98.33%	98.65%	98.47%
20	98.43%	98.08%	97.01%	98.33%	97.40%
10	96.72%	96.79%	95.98%	97.93%	96.55%

Table 10: Accuracy results of obtained MARS models when white Gaussian noise is added to data set.

SNR [dB]	MARS 1	MARS 2	MARS 3	MARS 4	MARS 5
No noise	5.323	4.998	5.333	5.026	4.834
40	6.297	5.464	5.347	5.309	5.315
30	6.015	5.501	5.566	5.328	5.493
20	7.357	7.249	6.710	6.577	5.975
10	8.768	8.405	7.878	7.569	7.499

5.3. Sensitivity to Noise

In order to assess the proposed methods' sensitivity to noisy measurements, we add white Guassian noise to the data set with different levels of signal-to-noise ratio (SNR). The obtained results are shown in Tables 9 and 10. As expected, as the SNR decreases, the accuracy rates in the CART models decrease and the MAEs in the MARS models increase. However, the models are quite robust against injected measurement noise as compared to rates reported in the existing literature. For example, in [38], accuracy rates decrease to 81% when models are trained and tested in the presence of noise with 30 dB SNR.

6. Concluding Remarks

Using PMU measurements obtained from the system, we develop *practical* CART and MARS models to assess transient

stability in real time. In order to train these models, we apply direct measurement, COI-referred, and energy function-based indices, which are chosen based on their availability in real time. Via case studies involving the full WECC system, we show that the trained CART and MARS models predict stable and unstable cases with high accuracy rates and provide a transient stability margin of the system based on the applied features, respectively. We observe that COI-referred and energy function-based features have more impact in the classification and regression models. Moreover, we study the sensitivity of the trained models to the data obtained from earlier or later sampling times. Avenues for future work include analyzing the robustness of the obtained models to missing and bad data which are probable occurrences in a practical WAMS. Another compelling topic for future work is the use of trained models to assess the fidelity or quality of real-time measurements.

Acknowledgement

The authors would like to thank Giuseppe Stanciulescu at Powertech Labs and Helen Whittaker at Technology Planning and Performance, BC Hydro for their help with model and data acquisition.

References

[1] G. Andersson, P. Donalek, R. Farmer, N. Hatziargyriou, I. Kamwa, P. Kundur, N. Martins, J. Paserba, P. Pourbeik, J. Sanchez-Gasca, et al., Causes of the 2003 major grid blackouts in north america and europe, and recommended means to improve system dynamic performance, IEEE Transactions on Power Systems 20 (4) (2005) 1922–1928.

[2] D. Novosel, V. Madani, B. Bhargava, K. Vu, J. Cole, Dawn of the grid synchronization, IEEE Power and Energy Magazine 1 (6) (2008) 49–60.

[3] M. Kezunovic, S. Meliopoulos, V. Venkatasubramanian, V. Vittal, Application of Time-synchronized Measurements in Power System Transmission Networks, Springer, 2014.

[4] P. W. Sauer, Power system dynamics and stability, Prentice Hall, 1998.

[5] S. Zadkhast, J. Jatskevich, E. Vaahedi, A multi-decomposition approach for accelerated time-domain simulation of transient stability problems, IEEE Transactions on Power Systems 30 (5) (2015) 2301–2311.

[6] H.-D. Chiang, Direct methods for stability analysis of electric power systems: theoretical foundation, BCU methodologies, and applications, John Wiley & Sons, 2011.

[7] D. P. Wadduwage, C. Q. Wu, U. Annakkage, Power system transient stability analysis via the concept of lyapunov exponents, Electric Power Systems Research 104 (2013) 183–192.

[8] A. Xue, F. F. Wu, Y. Ni, S. Mei, Q. Lu, Power system transient stability assessment based on quadratic approximation of stability region, Electric power systems research 76 (9) (2006) 709–715.

[9] I. Kamwa, S. Samantaray, G. Joos, Development of rule-based classifiers for rapid stability assessment of wide-area post-disturbance records, IEEE Transactions on Power Systems 24 (1) (2009) 258–270.

[10] R. Diao, V. Vittal, N. Logic, Design of a real-time security assessment tool for situational awareness enhancement in modern power systems, IEEE Transactions on Power Systems 25 (2) (2010) 957–965.

[11] M. He, J. Zhang, V. Vittal, A data mining framework for online dynamic security assessment: Decision trees, boosting, and complexity analysis, in: 2012 IEEE PES Innovative Smart Grid Technologies (ISGT), 2012, pp. 1–8.

[12] Y. Xu, Z. Y. Dong, J. H. Zhao, P. Zhang, K. P. Wong, A reliable intelligent system for real-time dynamic security assessment of power systems, IEEE Transactions on Power Systems 27 (3) (2012) 1253–1263.

[13] A. Kaci, I. Kamwa, L. Dessaint, S. Guillon, et al., Synchrophasor data baselining and mining for online monitoring of dynamic security limits, IEEE Transactions on Power Systems 29 (6) (2014) 2681–2695.

[14] J. Geeganage, U. Annakkage, T. Weekes, B. Archer, et al., Application of energy-based power system features for dynamic security assessment, IEEE Transactions on Power Systems 30 (4) (2015) 1957–1965.

[15] M. He, V. Vittal, J. Zhang, Online dynamic security assessment with missing PMU measurements: A data mining approach, IEEE Transactions on Power Systems 28 (2) (2013) 1969–1977.

[16] I. Kamwa, S. Samantaray, G. Joos, Catastrophe predictors from ensemble decision-tree learning of wide-area severity indices, IEEE Transactions on Smart Grid 1 (2) (2010) 144–158.

[17] Y. Xu, Z. Y. Dong, K. Meng, R. Zhang, K. P. Wong, Real-time transient stability assessment model using extreme learning machine, IET generation, transmission & distribution 5 (3) (2011) 314–322.

[18] L. Breiman, J. Friedman, C. J. Stone, R. Olshen, Classification and Regression Trees, Wadsworth, 1984.

[19] J. H. Friedman, Multivariate adaptive regression splines, The annals of statistics (1991) 1–67.

[20] M. He, J. Zhang, V. Vittal, Robust online dynamic security assessment using adaptive ensemble decision-tree learning, Power Systems, IEEE Transactions on 28 (4) (2013) 4089–4098.

[21] S. Samantaray, I. Kamwa, G. Joos, Ensemble decision trees for phasor measurement unit-based wide-area security assessment in the operations time frame, IET Generation, Transmission & Distribution 4 (12) (2010) 1334–1348.

[22] V. Madani, J. Giri, D. Kosterev, D. Novosel, D. Brancaccio, Challenging changing landscapes: Implementing synchrophasor technology in grid operations in the wecc region, IEEE Power and Energy Magazine 13 (5) (2015) 18–28.

[23] M. Rahmatian, W. G. Dunford, A. Palizban, A. Moshref, Transient stability assessment of power systems through wide-area monitoring system, in: Proceeding of 2015 IEEE PES General Meeting (PESGM), 2015.

[24] North American SynchroPhasor Initiative (NASPI) RITT Report, Guidelines for siting phasor measurement units (Jun 2011).
URL `https://www.naspi.org/documents`

[25] C. Fu, A. Bose, Contingency ranking based on severity indices in dynamic security analysis, IEEE Transactions on Power Systems 14 (3) (1999) 980–985.

[26] S. M. Rovnyak, Integral square generator angle index for stability assessment, in: Proceedings of the IEEE Power Engineering Society Winter Meeting, 2001.

[27] M. Rahmatian, W. G. Dunford, A. Moshref, PMU based system protection scheme, in: Proceeding of the 14th Electrical Power and Energy Conference (EPEC), 2014.

[28] I. Kamwa, A. K. Pradhan, G. Joos, S. Samantaray, Fuzzy partitioning of a real power system for dynamic vulnerability assessment, IEEE Transactions on Power Systems 24 (3) (2009) 1356–1365.

[29] A. A. Al-Taee, M. A. Al-Taee, W. Al-Nuaimy, Augmentation of transient stability margin based on rapid assessment of rate of change of kinetic energy, Electric Power Systems Research 140 (2016) 588–596.

[30] T. Hastie, R. Tibshirani, J. Friedman, The elements of statistical learning, Vol. 2, Springer, 2009.

[31] TSAT, Transient Stability Assessment Tool, http://www.dsatools.com.

[32] WECC planing services, https://www.wecc.biz/PlanningServices/Pages/Default.aspx.

[33] BC Hydro, Protection and Control Planning Department, Bulk transmission circuit fault clearing time, internal report (April 2015).

[34] IEEE standard for synchrophasor measurements for power systems, IEEE Std C37.118.1-2011 (Revision of IEEE Std C37.118-2005) (2011) 1–61doi:10.1109/IEEESTD.2011.6111219.

[35] CART, Classification And Regression Tree, http://www.salford-systems.com.

[36] MARS, Multivariate Adaptive Regression Splines, http://www.salford-systems.com.

[37] K. Sun, S. Likhate, V. Vittal, V. S. Kolluri, S. Mandal, An online dynamic security assessment scheme using phasor measurements and decision trees, IEEE Transactions on Power Systems 22 (4) (2007) 1935–1943.

[38] I. Kamwa, S. Samantaray, G. Joós, On the accuracy versus transparency trade-off of data-mining models for fast-response pmu-based catastrophe predictors, IEEE Transactions on Smart Grid 3 (1) (2012) 152–161.