Applying Different Wide-Area Response-Based Controls to Different Contingencies in Power Systems

1st Shahrzad Iranmanesh
Department of Electrical and Computer Engineering
Indiana University-Purdue University Indianapolis
Indianapolis, United States
shahrzad.iranmanesh@gmail.com

2nd Steven M. Rovnyak
Department of Electrical and Computer Engineering
Indiana University-Purdue University Indianapolis
Indianapolis, United States
srovnyak@iupui.edu

***Abstract*—Electrical disturbances in the power system can threaten stability. One-shot control is an effective method for stabilizing some events. In this paper, predetermined amounts of loads are increased or decreased around the network. Determining the amounts of loads, and the location for shedding is crucial. This paper is completed in two different sections. First, finding the effective control combinations, and second, finding an algorithm for applying different control combinations to different contingencies in real time. The particle swarm optimization (PSO) algorithm is used to find the effective control combinations. Next, decision trees (DT) are trained to assess the benefits of applying each of the three most effective control combinations found by PSO method. The DT outputs are combined into an algorithm for selecting the best control in real time. Finally, the algorithm is evaluated using a test set of contingencies. The results reveal a 46% improvement in comparison to previous studies.**

***Index Terms*—Decision tree (DT), phasor measurement unit (PMU), particle swarm optimization (PSO), transient angle stability, wide-area control.**

I. Introduction

Providing reliable and stable electrical power is one of the crucial subjects in the operation of the electrical systems. Because of the electrical faults in power stations, damages to electric transmission lines or loss of transmission equipment, the power supply faces many difficulties. On occasion the intensity of some disturbances are high enough to cause the generators to lose their synchronization, so a black-out may happen. In some situations, cascading outages may happen in the electrical grid.

One of the important issues in the electrical system is how to devise techniques for fault detection, and stabilizing them using proper control method. Therefore, after fault detection, selecting an effective combination of control actions and applying it to the system is very important to avoid the spread of faults through the electrical network. Numerous studies have been done with the purpose of detecting disturbances and applying a variety of control methods to stabilize them. Many authors used pattern recognition methods to address the concerns related to electrical disturbances in the power systems.

Pattern recognition method has been proposed in [1], [2] for stability prediction. Rovnyak et al. in [1] used Decision Tree (DT) as a pattern recognition method. The DT predictors in [1] are R and Rdot, which are apparent resistance and its rate of change measured near the electrical center of Pacific AC Inter-tie. They created a DT that could be used for response based control but control was not tested in the paper. In [2], the real-time classification was done with Recurrent Neural Networks (RNN), the long-term dependencies were resolved by Long Short Term Memory (LSTM). The pattern recognition methods in [1], [2], however, do not include any control action. In some studies, pattern recognition methods are applied to predict islanding in the power system [3]. Diao [3] used the DTs and synchronized phasor measurement to detect loss of synchronism and separate the network into pre-defined islands. A different approach is used for training of the DT in [3]; in fact, one DT is trained for each contingency instead of training one DT for all of the contingencies. Diao [3] used the voltage phase angle measurements of high voltage buses, and for each phase angle variable, they defined six features.

Some of the studies proposed islanding control method after instability prediction to maintain the frequency [4], [5]. The island management method proposed in [5] can maintain synchronism within each island. The feasible islanding interval is studied in [4] to apply island control method. The island control method can be considered as a backup for the control method in the current study.

In some other studies, pattern recognition is used to order control that keeps synchronization and avoids the need for islanding. Gao and Rovnyak [6] used two different approaches for DT construction process, and one of the methods resulted in a smaller region of feature space that is judged to be stable. The smaller region of space that is stable results in earlier detection of instability. Gao and Rovnyak [6] used 68 features as predictors calculated or measured using the Phasor Measurement Unit (PMU). One of the main contributions of [6]

This work was supported by the National Science Foundation under Grant No. ECCS-1711521.

is using the one-shot control to avoid the loss of synchronism caused by the events rather than splitting the electrical grid into islands. The algorithm in [6] orders control that keeps synchronization and avoids the need for islanding. Mei et al. [7] suggested a method to develop response-based DTs to activate control combination for stabilizing the events. The control used in [6], [7] is a fixed combination of power changes in four buses, but in the current study, the algorithm can select among multiple control options. In [8], the authors used phasor measurements and the combination of separate event detection and control DTs for transient stability control. Their control actions included disconnection of costly generation and load. Moreover, to train the DT, they applied some old and new indices. The results show a higher rate of success stabilizing events using one shot control than in [6].

A novel Under Frequency Load Shedding (UFLS) algorithm is used in [9]. In [9] the authors proposed a three stage scheme as a new centralized adaptive load shedding. The first stage includes analyzing the required data and sizing the reactive power. In the second step, the optimal amount of loads and their locations are specified. Finally, the third stage includes determining the event type.

A new control strategy is proposed in [10], which can choose between two sets of control rules. In addition to DTs for event detection and instability prediction, the author used a third DT to apply one of several one-shot control combinations, so the number of stabilized cases was improved. One of the drawbacks with this project is that they found the control options by trial and error method.

One-shot controls include discontinuous actions such as disconnection of generation and load. In contrast to [1]-[5], the present method can sometimes prevent loss of synchronism rather than just predicting loss of synchronism or allowing the system to break up into islands that are not synchronized with each other. In contrast to [6]-[8], this paper presents a new method that can apply different control options to different contingencies. In contrast to [10], the current method uses numerical optimization methods to find different control combinations that can be used on different contingencies instead of finding the different control combinations by hand.

The present method uses Particle Swarm Optimization (PSO) offline to find the most effective combination of one shot controls for each of a large set of contingencies. MATLAB programming is employed for developing the algorithm, and TSAT is used for transient analysis. The second step is to apply the best combination of controls for each contingency to all the other contingencies. Based on these results, we find that choosing one out of a small subset of the many control combinations can stabilize almost as many of the contingencies compared to applying the individually optimized combination of controls to each contingency. In the last step, a pattern recognition method has been applied to create decision criteria for deciding to actuate control and select one of a small subset of control combinations. Similar to the authors' previous work, DTs are used as the pattern recognition method.

II. Feature Extraction and Indices

A variety of indices or predictors can be used as inputs to pattern recognition tools. Some of the previous studies used only two predictors since using a two-dimensional feature space that can be visualized like R-Rdot application in [7]is helpful for explaining the method. In another study [6], voltage angle, voltage magnitude and their rates of change are used as the predictors. In this project, additional predictors are calculated from the measurements.

The first set of variables are bus voltage magnitudes and bus voltage angles that can be measured by PMU installed on a subset of the buses in the network. For each PMU bus, there are the voltage magnitude and angle variables plus the derivative of each of them. If there are PMUs installed in N buses, then 4N elements can be added to the input vector of the classification method. In this study, N=17. Derivatives of voltage angles and magnitudes can be calculated from differences between successive samples.

The average and variance of bus magnitude are two other indices that can be calculated using (1) and (2).

$$BMavg[k] = \sum_i \frac{|V_i[k]|}{17} \tag{1}$$

$$BMvar[k] = \sum_i \frac{(|V_i[k]| - BMavg[k])^2}{17} \tag{2}$$

Then derivatives of BMavg and BMvar can be calculated from point to point differences between samples 30 times per second.

One of the effective indices that can be applied for the classification objective is Integral Square Generator angle (ISGA) [11]. This index is a coherency based index that can be used to judge the severity of stable and unstable events in the simulations.

In real time, it is not possible to measure generator angles directly. Instead, this paper uses bus voltage angles from PMUs so the new index is Integral Square Bus Angles (ISBA). Indices like ISGA and ISBA represent the overall stress on the system [8] calculated over a period of time. In this paper, we do not perform the integration step and the index is Square Bus Angle (SBA) (3).

$$SBA[k] = \sum_i M_i(\Theta_i[k] - \Theta_{coa}[k])^2 \tag{3}$$

where M_i is chosen to weight angles from different locations, $\Theta_i[k]$ represents the bus angles measured by PMUs and Θ_{coa} is weighted average of the all the bus angles. We set all the Mi equal to one so the sum in (3) is just divided by 17.

Another index, which is used in this study is the derivative of the SBA that is

$$SBAdot[k] = 30(SBA[k] - SBA[k-1]) \tag{4}$$

III. PARTICLE SWARM OPTIMIZATION METHOD FOR ONE-SHOT CONTROL

Particle Swarm Optimization (PSO) is an optimization technique for exploring the search space and minimizing/maximizing a particular objective [12]. The main idea starts with initiating a random population of potential solutions in the search space. Then the objective function is evaluated for each agent, i.e. member of the population, and the best value is selected among them. In this project, the objective function used in the offline optimization is ISGA. After finding the control combination with minimum ISGA, the rest of the agents are trying to move toward the location of the best agent in addition to movements that are random [13].

In each iteration of running the PSO algorithm, the population is updating by a velocity vector that can be calculated using the equation (5) [13].

$$V_i(t+1) = \omega V_i(t) + c_1 r_1[\hat{X}_i(t) - X_i(t)] + c_2 r_2[g(t) - X_i(t)] \tag{5}$$

The index of each particle at every iteration is represented by i. $V_i(t)$ is the velocity of particle i at time t and $X_i(t)$ is the position of particle i at time t. c_1 and c_2 are two constant numbers between 0 and 2, and they are selected 2 in this research. $r_1$and r_2 are two random number between 0 and 1. $\hat{X}_i(t)$ is the best solution in each iteration. In this project, the solution related to the minimum ISGA in each iteration is selected as $\hat{X}_i(t)$. $g(t)$ is the global best candidate solution up to the iteration t. ω is a parameter decreasing by increment of the number of iteration and it is calculated by 6 [13].

$$\omega = 0.2 + \frac{(0.9 - 0.2)}{(1 - I_M)}(I_C - I_M) \tag{6}$$

In 6, I_M is the maximum number of the iteration, I_C is the current iteration number.

PSO was used to determine amounts of load changes on 27 buses, each having a load amount equal to 500 MW or greater. ISGA is the objective function of the PSO algorithm. The maximum value of the load change on each bus was selected 500 MW, and the minimum value was selected -500 MW. A negative load change represents increasing the load on a bus which could be accomplished by disconnecting generation or using a braking resistor. The number of particles in the PSO algorithm is also selected as 50. Therefore, the dimension of the search space is a 50×27. The maximum number of iterations is chosen as 20 since usually the algorithm could return the solution in less than 6 iterations. The maximum number of iterations is the only parameter that we tuned and the remainder such as the parameters of the velocity were selected from the previous publications in PSO, so they are system-independent such as c_1 and c_2 [13]. Running the PSO algorithm 100 different times for 100 unstable events produces 100 different control combinations, each involving different amounts of power changes on the 27 buses. From these 100 events, 42 were successfully stabilized by the control combination found by PSO.

IV. DECISION TREES FOR CONTROL SELECTION

The main idea of this section is to find an algorithm that can select from different control combinations for stabilizing various events rather than applying the same combination to every event as in [6] - [8]. Different artificial intelligence methods can be employed for this purpose, like Neural Network (NN), and DTs. The main advantage of DTs over other pattern recognition tools is the training time. Another advantage is that with a large number of predictor variables available, a small subset of these variables is normally selected in the DT training process [14].

A. Control Combinations

Previous studies applied the same combniation of one shot controls to any event for which control was ordered [6]–[8]. This project applies different control combinations to different events. Using the optimization results from Section III, three control combinations are selected. The proper application of one of these three control combinations can stabilize as many of the events as applying the custom control combination found by PSO to each event.

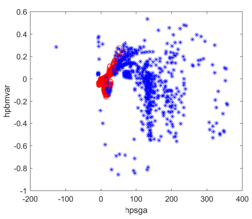
Control combination 1, for example, applies 500 MW fast load power increases on two buses (MONTANA and CA230) and 500 MW load shedding on three other buses: MIDWAY, NAVAJO, and MOHAVE. A common feature of controls combinations that can stabilize a transient event in progress is that they reduce overall angle differences in the AC network [6] and [8]. The process of selecting between three control sets is done through decision tree algorithms.

B. Data Sets

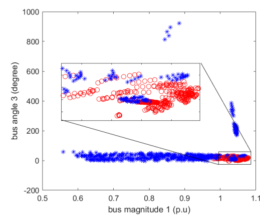
Our DTs are created offline from training data sets where each data point consists of an input vector along with a target value, which represents the class of that sample. For the combination of DTs that select between the three control combinations, the target is one of the following: Control 1, Control 2, Control 3. The training set includes data from 385 six-second simulations on a 176 bus model of the WECC. The events include short circuit to ground faults on 40 transmission lines in the WECC model. The test set includes 960 events containing 480 1-phase short circuit faults and 480 3-phase short circuit faults.

To obtain the data sets, TSAT software is run many times by a MATLAB script which compiles the resulting data. In each time step for every event, TSAT software provides bus voltage angles and magnitudes recorded at 17 locations where PMUs are located. Then bus frequencies, bus magnitude variation, SBA, and derivatives of the composite indices are calculated. From 17 PMUs there are a total of 77 features. Using the same stability condition applied in [6] and [8], an event is unstable if it has a maximum generator angle difference greater than 300 degrees.

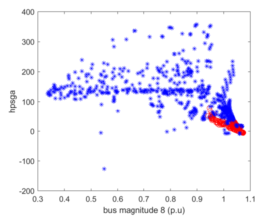
We train a separate DT for each control combination. The target value for each individual DT is determined by applying its associated control to the event. If an event can be stabilized by control combination 1, the target value in the training set for DT 1 is set to 1, otherwise, the target value is 0. Therefore, the input vectors of the training sets for the three separate DTs are identical but the target values are different. Test data is also evaluated using the same method for training data. Figure 1 represents the scattering plot for two features related to Test data for control combination 1, 2, and 3.



(a) Control combination 1.



(b) Control combination 2.



(c) Control combination 3.

Fig. 1: Testing data.

C. Implementation of the DT method

The problem for training the algorithm of each control combinations is a Boolean classification problem since the output is stable/unstable or 1/0. Every node in the DT can be represented by a variable or a feature. Eventually, the leaf nodes show the target value of the input vectors.

The classification algorithm used in this study consists of several steps. The root node receives the entire training data set as input. Usually, all nodes are asking a true-false question about one of the features. Two types of question can be asked based on the type of features in the data set greater equal $>=$ or less equal $<=$. Greater equal $>=$ is used for the questions asked in this project. In response to this question, the data set is split or divided into two subsets. The new subsets are the input to the two child nodes. The goal of the questions is dividing the labels as far as possible. The tree grows down to find the purest possible distribution of the labels at each node, or when there is no uncertainty about the type of the label. To quantify how much a question unmixed the labels a metric called Gini impurity is used in the current project [15]. To evaluate how much a question reduces the amount of uncertainty, a concept called information gain was used. There are many types of equations for calculating the Gini impurity and information gain. In this project, 7 shows the Gini function, and equation 8 shows the Impurity gain.

$$Gini = 1 - \sum_i P_i^2 \tag{7}$$

$$IG = CU - P_{left} * Gini(Left) - P_{right} * Gini(right) \tag{8}$$

In (7), P_i is the probability of the labels. As we only have two labels, the maximum of i is equal to 2. In (8), IG shows the information gain, CU shows the current uncertainty, P_{left} and P_{right} show the probability of the left and right node respectively in each iteration. Current uncertainty in the root node equals the Gini impurity of that node and as the tree proceed the uncertainty of the DT in each iteration is calculated using (8). Using Gini impurity function and information gain, the best question can be selected at each node. Then we continue recursively to build the tree on each of the new nodes. The data is continuously dividing until there is no question to ask.

V. Results

The study model in this project is a simplified representation of the Western Electricity Coordinating Council (WECC). Different types of 1-phase and 3-phase disturbances are simulated using TSAT software, and the data are analyzed using MATLAB. Various types of features are calculated as described above. Separate DTs are trained for detecting that an event has occurred and for deciding whether to apply one of the three control combinations. In addition, two sets of Training and Test events are are simulated to train and test the control algorithm technique.

The proposed scheme is tested on 960 additional contingencies simulated on the 176 bus model of the WECC. The results for the prediction accuracy of the three DTs used for control selection are shown in Table I. Each sample of measurements from each of the 960 contingencies is classified by each of the three DTs and compared to the target output values for the respective DTs. The accuracy was evaluated by counting the number of samples that were correctly classified divided by the number of samples.

TABLE I: The Accuracy results for learning and testing of the DTs.

Control combination	Test (%)	Train (%)
1	88.71	93.94
2	86.22	89.39
3	93.94	93.59

The control scheme requires two separate types of DTs. One DT is trained to detect that an event has occurred and its details are described in [8]. After an event is detected, five subsequent samples are processed by the three DTs that decide which if any of the three controls should be applied. As mentioned earlier, the output value 1 means the event is predicted to be stabilized by the control. A score is calculated for each DT by adding the five output values for each DT to obtain a number between 0 - 5. The control with the largest score that exceeds a threshold is applied to the event. If all the scores are equal and over the threshold, for example 5,5,5, the control combination 1 is selected.

Table II illustrates the results after applying the new algorithm for the 960 events. The results in Table II require every event for which control is ordered to be simulated again with the control action occurring 0.1 second after the control was ordered by the DT. This means we are not just training DTs on 960 simulations and reporting their accuracy. We are performing additional time domain simulation to determine the effect of the control scheme. In contrast to results presented in Table II, the results in Table I do not require running the simulation over again with the control ordered by the DTs.

The columns of Table II show the number of events controlled, the number of events that were controlled unnecessarily, the number of events stabilized by applying the control, average control time in seconds, and SR is the success rate of the algorithm. Every contingency in the training and test sets included a fault that was cleared at 0.67 seconds so an average control time of 0.77 seconds indicates the DTs ordered control, on average, 0.1 second after the fault clearing time.

TABLE II: Performance of 3 DTs for control selection for 1 phase and 3 phase faults.

Test set	Controlled	Unnecessary	Stabilized	Tavg (s)	SR
1-phase	96	6	36	0.77	0.375
3-phase	113	13	36	0.91	0.318
Total	209	19	72	0.85	0.3465

Figure 2(a) shows the generator rotor angles for 29 generators of the model during a transient event for 6 seconds. A 3 phase fault occurred at 0.55 second on the line between Hanford and John Day buses and cleared at 0.67 seconds. Figure 2(b) shows the simulation of generator rotor angles for the same fault after applying the algorithm. This event is stabilized by control combination 2. The algorithm can effectively identify the appropriate control combination and stabilize it using 3 DTs.

The corresponding results in [8] for the same test set have the total number of stabilized events as 49, and the success rate as 0.236. Therefore, the method in this paper successfully stabilizes 46 percent more events than the method in [8], and its success rate is 46 percent higher than the method in [8].

The results in this paper are significant for two reasons. One reason is that our method provides an improvement over the status quo, which is essentially not having any wide-area response-based one-shot control scheme to stabilize an unstable transient angle swing in progress. The other reason is that when the proposed scheme is not successful at stabilizing a transient event in progress then it almost never causes any harm.

VI. Conclusion

In this project, the Particle Swarm Optimization (PSO) algorithm is used to find the best combination of controls for a large set of contingencies. A subset of three of those controls are selected and three DTs are trained based on them. In the final step, an algorithm is developed with the ability to decide between the three control combinations and choose one of them in real-time. The final results showed the algorithm stabilizes 46% more events that the method in [8] and 30% more events than the method in [10]. These results are significant because of the enormous cost of blackouts, which can be billions of dollars, and the fact that there is no comparable wide-area response-based, one-shot control scheme that is currently deployed.

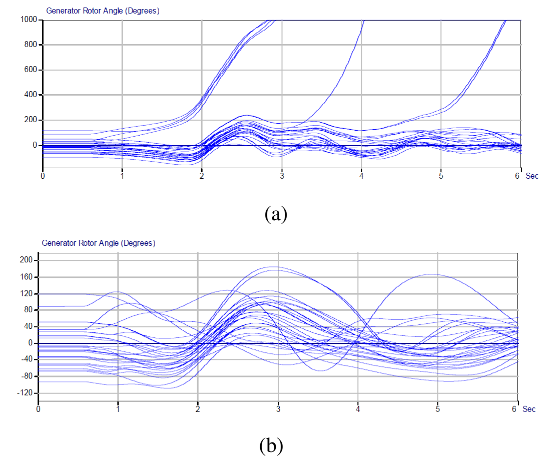


Fig. 2: Generator rotor angles during a 3 phase event (a) before applying the algorithm (b) after applying the control selection algorithm and selecting control combination 2.

References

[1] S. M. Rovnyak, C. W. Taylor, and Y. Sheng, “Decision trees using apparent resistance to detect impending loss of synchronism,” *IEEE Transactions on Power Delivery*, vol. 15, no. 4, pp. 1157–1162, 2000.

[2] L. Zheng, W. Hu, K. Hou, X. Xu, and G. Shao, “Real-time transient stability assessment based on deep recurrent neural network,” in *IEEE Innovative Smart Grid Technologies-Asia (ISGT-Asia)*, 2017, pp. 1–5.

[3] R. Diao, V. Vittal, K. Sun, S. Kolluri, S. Mandal, and F. Galvan, “Decision tree assisted controlled islanding for preventing cascading events,” in *IEEE Power Systems Conference and Exposition (PSCE’09)*, 2009, pp. 1–8.

[4] L. Ding, Y. Guo, P. Wall, K. Sun, and V. Terzija, “Identifying the timing of controlled islanding using a controlling uep based method,” *IEEE Transactions on Power Systems*, vol. 33, no. 6, pp. 5913–5922, 2018.

[5] R. J. Best, D. J. Morrow, D. M. Laverty, and P. A. Crossley, “Techniques for multiple-set synchronous islanding control,” *IEEE Transactions on Smart Grid*, vol. 2, no. 1, pp. 60–67, 2011.

[6] Q. Gao and S. M. Rovnyak, “Decision trees using synchronized phasor measurements for wide-area response-based control,” *IEEE Transactions on Power Systems*, vol. 26, no. 2, pp. 855–861, 2011.

[7] K. Mei and S. M. Rovnyak, “Response-based decision trees to trigger one-shot stabilizing control,” *IEEE Transactions on Power Systems*, vol. 19, no. 1, pp. 531–537, 2004.

[8] S. M. Rovnyak, D. W. Longbottom, D. C. Vasquez, and M. N. Nilchi, “Monitoring and control using synchrophasors in power systems with renewables,” forthcoming from The Institution of Engineering and Technology, Eds., 2020, ch. 8 Response-Based Event Detection for One Shot Wide-Area Stability Controls, 978-1-78561-478-1 (ebook).

[9] T. Shekari, A. Gholami, F. Aminifar, and M. Sanaye-Pasand, “An adaptive wide-area load shedding scheme incorporating power system real-time limitations,” *IEEE Systems Journal*, vol. 12, no. 1, pp. 759–767, 2018.

[10] M. Naghsh Nilchi, “Electric utility planning methods for the design of one shot stability controls,” *MSECE Thesis, Purdue University, IUPUI*, 2013.

[11] S. M. Rovnyak, M. N. Nilchi, D. W. Longbottom, and D. C. Vasquez, “Angle stability predictive indices,” in *IEEE Power and Energy Society General Meeting*, 2012, pp. 1–6.

[12] J. Kennedy, “Particle swarm optimization,” *Encyclopedia of machine learning*, pp. 760–766, 2010.

[13] M. E. H. Pedersen, “Good parameters for particle swarm optimization,” *Hvass Lab., Copenhagen, Denmark, Tech. Rep. HL1001*, 2010.

[14] G. Li and S. M. Rovnyak, “Integral square generator angle index for stability ranking and control,” *IEEE Transactions on Power Systems*, vol. 20, no. 2, pp. 926–934, 2005.

[15] A. Géron, *Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems.* ” O’Reilly Media, Inc.”, 2017.