Electrical Power and Energy Systems 45 (2013) 35–41

Contents lists available at SciVerse ScienceDirect

Electrical Power and Energy Systems

journal homepage: www.elsevier.com/locate/ijepes

Progressive entropy based contingency grouping for deriving decision trees for multiple contingencies

Venkat Krishnan *, James D. McCalley

Department of Electrical and Computer Engineering, Iowa State University, Ames, IA 50014, USA

ARTICLE INFO

Article history:
Received 30 April 2012
Received in revised form 26 August 2012
Accepted 29 August 2012
Available online 6 October 2012

Keywords:
Engineering application
Decision trees
Voltage collapse
Power system security assessment
Progressive entropy
Multiple contingencies

ABSTRACT

This paper deals with decision tree based power system security assessment for multiple contingencies. In this paper, we propose a contingency grouping technique to produce good, but reduced number of decision trees for multiple contingencies. The grouping is based on the degree of overlap in class boundary regions or post-contingency performance measures of every contingency, as captured by the graphical index progressive entropy. The proposed method has been demonstrated on French EHV system for a voltage collapse problem in west France, Brittany region.

1. Introduction

Power system operational planning studies, system operators use operating guidelines in terms of key system attributes and their corresponding thresholds to assess post-contingency system security of current operating condition, i.e., whether it is acceptable or unacceptable with respect to a particular stability criteria. Then the system is accordingly maneuvered to a stable and economic operating condition within the security limits [1,2]. Such preventive guidelines against stability problems under certain top contingencies are typically obtained using decision trees [3–5] that derive knowledge in the form of decision rules from a database of post-contingency responses over a wide variety of operating conditions. The performance of the tree in predicting post contingency performance is generally measured in terms of classification accuracy and error rates, namely false alarms (false positives) and risks (false negatives).

In the case of multiple contingencies, typically either a global decision tree or separate trees for all the contingencies are constructed for performing security assessment. Separate tree for a contingency can be constructed using the post-contingency simulation results of that particular contingency as the training database. Usually, the separate decision tree for a contingency gives the best performance; with the disadvantage of burdening the system operators, who will have to deal with complex situation of interpreting and applying too many rules.

On the other hand, a global decision tree for many contingencies could be derived from a training database that is formed by combining the post-contingency simulation results of all the contingencies. But, there is the danger of reducing the operating rule's performance under the most constraining and likely contingency, when we combine all the contingencies' training databases together. One could also form a global decision tree by using only the training database of the most constraining contingency, with the assumption that the operational rule in this case will also perform well on the other contingencies. But in reality, under the highly uncertain nature of power system conditions, such an operational rule may not be effective for all other contingencies. So such global trees require decision tree post-processing methods [4] or meta-learning methods such as bagging, boosting, and stacking [6] to improve its performance, which may lead to over-fitting and complicated rules consisting of voting schemes.

The proposed concept of contingency grouping-based decision trees in this paper designs a decision process that strikes a balance between producing simple and good trees, as well as reducing the number of trees required for multiple contingencies. The idea of grouping components based on specific performance criteria or geographical proximity is already prevalent in power system. Typically it is done to reduce computational cost in system reliability studies, in pattern recognition or knowledge discovery studies, and also to obtain valuable guidance in decision making. For instance, generators are grouped based on their slow-coherency

* Corresponding author. Tel.: +1 515 294 5499; fax: +1 515 294 4263.
E-mail addresses: vkrish@iastate.edu (V. Krishnan), jdm@iastate.edu (J.D. McCalley).

0142-0615/$ - see front matter
http://dx.doi.org/10.1016/j.ijepes.2012.08.078

performance which gives valuable information for controlled islanding to prevent blackout [7–9]. Generators are grouped based on angle gap criteria for fast contingency screening [10]. Unsupervised learning methods are used to group contingencies based on their effect on bus voltages [11], which is used by Neural Networks to predict post-contingency bus voltages under many contingencies. In this paper, we propose to group contingencies based on a newly developed graphical index, termed as progressive entropy, in order to obtain efficient decision trees for multiple contingencies. This grouping index is developed specifically for decision trees considering the technicalities involved in the tree induction process using a given training database.

We have organized this paper as follows. In Section 2, we describe the concept of progressive entropy. In Section 3, we have explained how this graphical index can be used to group contingencies, and form global decision trees for each groups. In Section 4, we have presented an illustration on French EHV system for a voltage stability study. We conclude the paper in Section 5.

2. Progressive entropy

Entropy, an information theoretic measure for quantifying information content in a distribution [12], is also usually used to quantify information content in a database and helps in identifying the most efficient database for security assessment using machine learning process [13]. Higher the database entropy, higher the information content for devising good real-time solution strategies based on offline simulations. It is defined as given in the following equation:

$$\text{Entropy }(S) = \sum_{i=1}^{c} -p_i \log_2 p_i \quad (1)$$

where S is training data, c is the number of classes, and p_i is the proportion of S classified as class i. According to (1), the more the non-homogeneity in class attribute, the more is the entropy [13]. This database variability measure plays an important role in tree induction process. At every level of the tree, attributes are ranked for their ability to divide the database into homogenous classes by a measure known as information gain [6], which is derived from entropy. The information gained from an attribute a is reflected by the reduction in database entropy knowing the value v of a among the set A, as given the following equation.

$$\text{Inf.Gain }(S,a) = \text{Entropy }(S) - \sum_{v \in A} \frac{|S_v|}{S} \text{ Entropy }(S_v) \quad (2)$$

Since, the entropy measure helps in effectively choosing the system attribute that best classifies the operating conditions into appropriate post-contingency performance, in this paper we devise a method based on entropy measures of various contingency's post-contingency response databases. The more similar the entropy measures are for certain contingencies, the similar is their influence on operating conditions, which thereby motivates to derive a common decision tree.

In order to ascertain the similarities and dissimilarities among various contingency's impact over many operating conditions, we propose a new index called progressive entropy, which is an improvised version of the traditional database entropy measure. Fig. 1 shows a typical manner in which the post-contingency performance measure for a particular contingency is progressing along the operating parameter state space, i.e., system load in this case. This in effect is also the class boundary progression of training database. The figure also shows the entropy measure that is derived from incremental databases as we progress along the various values of load attribute. Hence we coined the name progressive entropy for this index, which can be used to assess the manner in which the database entropy measure behaves over an attribute's range under the influence of various contingencies. The idea is, if the entropy measure behaves similarly over the range of an attribute, then the information gain also behaves similarly over that range. Hence the more similar the progressive entropy curves are under various influential attributes for certain contingencies, the higher is the possibility to derive a common decision tree for those contingencies. This is the driving principle behind our newly proposed decision tree assessment process by grouping multiple contingencies based on progressive entropy curves.

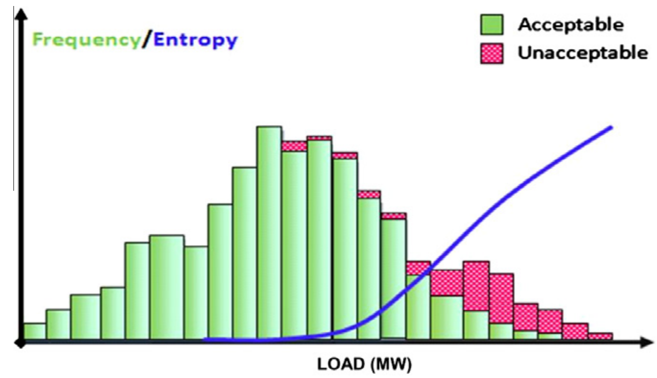

Fig. 1. Performance measure progression and progressive entropy.

Also in [5], we demonstrated how efficient sampling of operating conditions from post-contingency boundary region produced high information contained training database, from which highly accurate and economical decision trees were derived. Similarly in this paper, the same principle holds true for generating common decision tree for each group of contingencies. This is because the progressive entropy curves capture the similarities among various contingency's influence on operating conditions, and hence capture the degree of overlap among class boundary regions of their training databases. The more similar is the progression of class boundary regions of two contingencies, the more is the possibility to produce a common decision tree for them with good accuracy.

The progressive entropy curve is computed as follows:

Step 1: Sample many operating conditions from the multivariate parameter distribution
Step 2: Perform simulations, ascertain post-contingency performances, and label them, i.e., acceptable or unacceptable. The computational cost involved at this step in performing system simulations can be tremendously saved by using linear sensitivities of performance measure with respect to sampling parameters [14].
Step 3: Stack the performance measure variability along a system attribute distribution, as shown in Fig. 1.
Step 4: Compute the entropy for every sub-sets of database S_j as we progress along the distribution of the attribute, as shown in (3); and plot the progressive entropy.

$$\text{Progr.Entropy} = \text{Entropy }(S_j) = \sum_{i=1}^{c_j} -p_i \log_2 p_i, \quad j = 1 \ldots N \quad (3)$$

where S_j is the progressive database made of sub-set operating conditions taken in the direction towards unacceptable domain from acceptable, N is the number of sub-set operating conditions and consequently the number of progressive databases, c_j is the number of classes in database S_j, and p_i is the proportion of S_j classified as class i.

3. Contingency grouping

Fig. 2 shows the typical progressive entropy curves for four different contingencies C_1, C_2, C_3 and C_4, with C_1 imposing the highest