A New Transient Voltage Stability Prediction Model using Big Data Analysis

Bingbing Zhao and Junwei Cao
Research Institute of Information Technology
Tsinghua University
Beijing 100084, P. R. China
Email: jcao@tsinghua.edu.cn

Ziyu Zhu
Delta Research Center - China
Delta Electronics (Shanghai) Co. Ltd.
No.7 building, 6th courtyard,
Beichen East Rd., Chaoyang Dist.,
Beijing 100105, P. R. China

Huaying Zhang
Shenzhen Power Supply Co. Ltd.
China Southern Power Grid
Shenzhen 518020, P. R. China

***Abstract*—A new prediction model is proposed in transient stability analysis based on machine learning in this paper. It extracts features ahead from the time point that we want to make prediction, which produce an interval to take actions. The proposed model also takes network information into consideration, and tried to analyze how nodes in power grid influence each other. Compared to traditional algorithms which just use data from a single node in the past, this model has higher prediction accuracy. Logistic regression is chosen to be the classifier because the learning parameters can be regarded as the significance of variables. At the end, we also develop a practical system called RGAS by mixing Hadoop and Storm. It can perform learning off-line with high throughout, and make predictions on-line with low delay.**

Keywords—voltage stability; big data; machine learning; feature selection; cloud computing; hadoop

I. INTRODUCTION

Since 1970s a series of power blackouts bring huge economic losses all over the world. Transient voltage stability has become a focus issue in power system operation. With the continuing growth in system interconnection sizing and loading near to the limits, transient voltage stability turns to be more difficult.

Traditional analysis methods are all based on models such as time domain simulation method and transient energy function method [1]. Time domain simulation method builds a set of higher order differential equations, and when the system become more complex, the calculation time to solve these equations increases sharply. In addition, the parameters in these models have great influence on the final results. However, in a practical application, we usually make a lot of assumptions making the model to be oversimplified. Therefore, we need a new method that can both simulate complex system and have higher computation speed.

As Wide Area Measurement System (WAMS) based on Phasor Measurement Units (PMU) are widely installed in power grid, real-time monitoring of each node becomes possible. With these data, we can have situation awareness of the whole network and furthermore know if it will remain stable or not. Data-based method can fit the relationship between input and output without knowing how the real system works. Power system is the most complicated nonlinear system, machine learning can perform better and faster compared to traditional approaches.

In recent years, the cost of data storage decreases significantly, and at the same time, a number of distributed computing platforms, such as Hadoop, Storm and Spark, make computing tasks can be executed in a distributed environment easily. Mahout builds an environment for quickly creating scalable performant machine learning applications. Thus, big data analysis becomes really easy. Since 2006, deep learning becomes a hit which has applications in speech recognition, object recognition, etc. Data-based approach makes what we used to more intelligent and more accurate.

There have been some previous work that using machine learning method to do transient stability analysis, and nearly all types of classifiers are applied. [2] shows the suitability of support vector machine for transient stability analysis. It has high dimensionality of power system data, and tries to do sparsity reduction which makes the training process into an easier task for MLPs. [3] makes an analysis on voltage stability margin firstly, and has the conclusion that voltage magnitudes and the phase angles are the best predictors of transient stability analysis. Then this paper shows that proposed ANN based method can successfully estimate the voltage stability margin under both normal operation and N-1contigency situations. [4] investigates an inductive inference method for the automatic building of decision trees and the criteria of splitting and stop splitting.

For the past several years, there are also some works that using data-based method to solve Transient Stability Analysis(TSA) such as [5][6][7][8][9]. Most of them focus on acceleration of learning, and also applies some latest achievement in machine learning or pattern recognition area.

In this paper, we use data-based method to analysis transient voltage stability. Different from the previous research, we take the influence between the nodes in power grid into consideration. Therefore, the features of our classifier include not only information from the past, but also from the network. The rest of this paper is organized as follows. Section 2 introduces how our dataset is produced through simulation. In section 3, a delay prediction model is proposed and necessity of network information is also discussed. In section 4, we developed a system called RGAS which can perform off-line learning and on-line prediction. Finally, conclusions and some future work are discussed in section 5.

II. Dataset Generation and Preprocessing

As we know, in real power grid, unstable events are extremely rare. Therefore, the stable and unstable samples are imbalanced, which may cause bias prediction for classifiers. For example, if 99 percent of the all samples are stable, the classifier gives stable prediction without any learning, which can also get an accuracy at 0.99.

To avoid this problem, our learning dataset is generated from power system simulation software PSASP(Power System Analysis Software Package, a simulation software developed by China Electric Power Research Institute). We choose CEPRI 36-bus system(an example system in PSASP, Figure 2.1) which has 36 buses in all, and we choose 9 of them to be our targets. These 9 bused are bus16, bus18, bus19, bus20, bus21, bus22, bus23, bus29, bus9.

Considering a specific power system, a list concerning all kinds of possible operating conditions and incidents based on the operating records and experience is made. These concerns include various operating points, load compositions (proportions of dynamic load), fault types, fault locations, fault clearing time and other possible settings of operation. We get a large number of samples by combining different fault parameters above.

Actually, how many cases are enough for practical applications depends on the scale and the complexity of the specific system. Since a larger system with more buses usually includes more possible changes of operation and more categories of incidents, the number of cases needed is larger as well. In fact, their relationship is non-analytic, neither simply linear nor exponential. Given a specific region, if internal buses have close connections with each other, implying more coupling and interactions, the operation conditions may have more possible changes, which results in the increasing need of the number of cases. On the other hand, if they are far away from each other, possible changes of operation conditions could be less than the former, and less cases may be needed. In addition, if it’s hard to find instability cases in a system under general conditions, more extreme and severe situations in practice such as cascading failure and higher proportions of dynamic load can be taken into account.

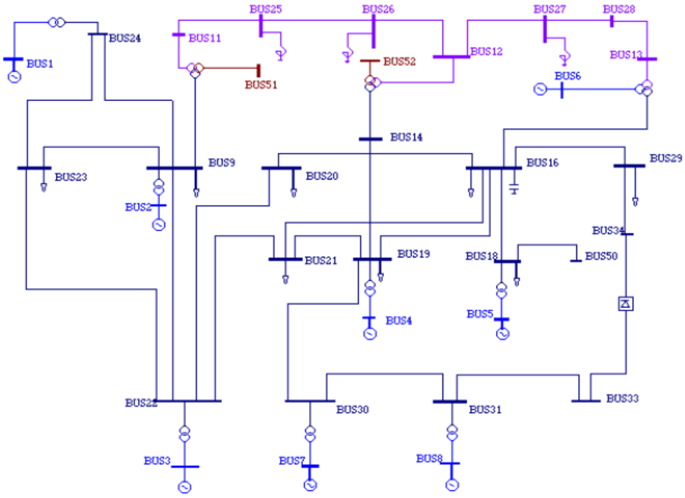

Figure 2.1 CEPRI-36 bus system

After each simulation, the software can export the voltage (U), current(I), active power(P) and reactive power(Q) of different buses.

III. Model description

A. *Features from Network*

In previous works that try to predict transient stability, features are just extracted in time domain. With this method, only the information from past is used, and what the classifier learn is a trend in time line. However, there are also much information in the network. All the instability events are arisen from some faults or load fluctuations somewhere in the power grid. These faults or load fluctuations just influenced local voltage at first, and then spread to whole system. Thus, data from other nodes also contain much information when we want to predict a specific node. In our work, we use the information from the whole network to predict the stability of a single node. Some trends that are not obvious in early stages can be found, and further predict the development of the whole system.

B. *Delay Prediction*

The ultimate goal of transient stability analysis is to realize prediction. We want to take full advantage of the data we already have and predict whether the whole system will remain stable in the next few seconds or minutes. However, in previous works, no one has set an interval between prediction moment and the feature period. Because the data near the prediction moment includes clear signal of whether it will be stable or not, the logic of this method is not convincing enough.

As shown in Figure 3.1, the main contribution of our approach is that there is a time interval between prediction moment and the feature period, and prediction moment is several seconds ahead of feature selection period. By doing this, we can discover some indication of instability in early stage. After achieving this, we can take full advantage of the interval, and further take actions such as SVG to stabilize the whole system.

C. *Moving Window*

Because this method will be used in an online system, the features are constructed by a 'moving window'(Figure 3.1). After the program is started, the data flows into the feature matrix successively. When the feature matrix is filled up, the classifier begins to output prediction result. As time moving forward, not all the features in last moment is replaced. The feature period is like a moving window, and only the earliest data are discarded. The new data will be added at the end.

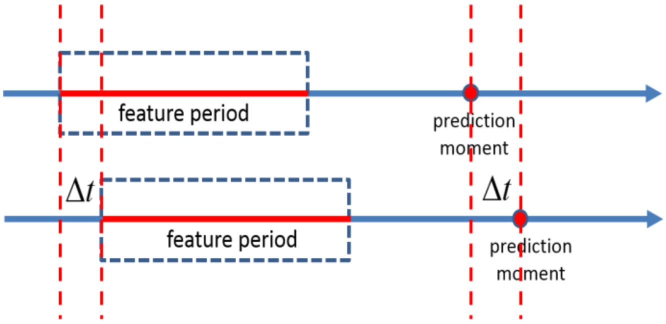

Figure 3.1 Delay prediction and moving window

IV. Classifier Training and Results

With determining the feature period we discussed in section 3, all the features are: voltage(U), current(I), active power(P) and reactive power(Q) at selected buses before fault occurrence; U, I, P, Q of the other 8 buses before fault occurrence; and we also compute the first derivatives of these four variables and tries to reflect their trendency. The output variable has been chosen to be the two classes of interest: stable(0) or unstable(1).

A. *Feature Selection*

Because of the high dimensionality of the input space, feature selection techniques have also been applied to achieve a more concise representation of the power system and overcome the curse of dimensionality. We use principal component analysis (PCA) to perform feature selection. PCA is a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables, so-called principal components.

Suppose that we have N samples of n -dimension vector x , and each row is a sample, column is $x_1, x_2, \cdots x_m$. We wish to reduce the dimension from n to m . Principal component analysis completes this by finding linear combinations, $a_1x_1, a_2x_2, \cdots a_mx_m$, called principal components, which have maximum variance, and subject to being uncorrelated with previous principal components. The PCA tries to reduce dimensions of data considerably while still retaining much of the information in it.

Specific steps of PCA are derived as follows:

- Normalize the sample data by:

$$x_i(t) = \frac{x_i(t) - \mu(x_i)}{\sqrt{\delta^2(x_i)}}$$

where:

$\mu(x_i)$ is the mean of x_i

$\delta(x_i)$ is the standard deviation of x_i

- Compute the covariance matrix of sample data after normalization:

$$XX^T$$

- Compute the eigenvalues and eigenvectors of XX^T and sort all eigenvalues. Select the corresponding eigenvectors the biggest m eigenvalues as principal component orientation.

$$\alpha^1, \alpha^2, \cdots \alpha^m$$

- Then we compute the projection of sample data on principal component (also the compressed data):

$$y(t) = [\alpha^1, \alpha^2, \cdots \alpha^m]^T X$$

Figure 3.2 shows the visualization of 9 buses after PCA(2 principal components), black represents unstable samples and yellow represents stable samples. As we can see, two groups of samples are obviously separated, which makes the classification possible.

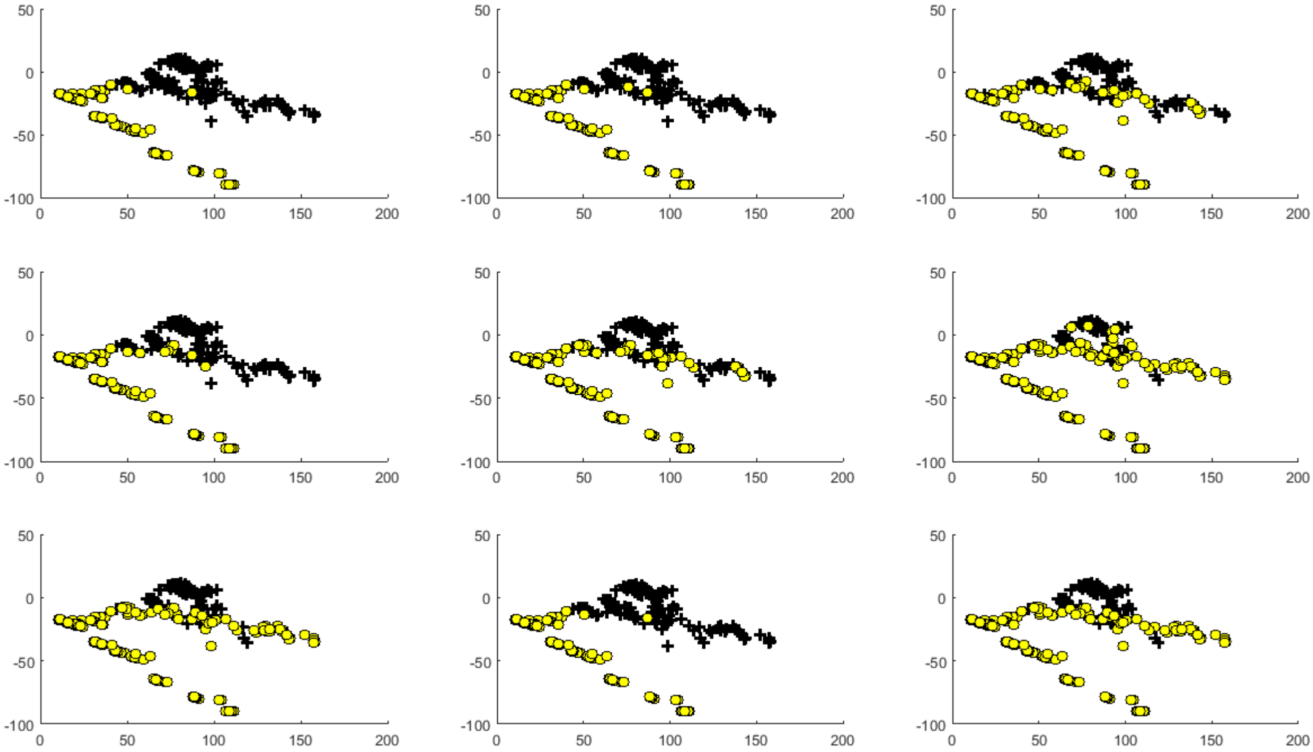

Figure 3.2 Visualization of 9 buses after PCA

B. *Logistic Regression*

About the selection of classifiers, there are many candidates, such as SVM, Adaboost, Na ïve bayes, logistic regression, etc. We finally choose logistic regression. Logistic regression can be seen as a special case of generalized linear model, and it is categorical and designed to deal with dependent variables. The advantage of logistic regression compared to other classifiers is that the learning parameters have physics meaning. There learning parameters can be taken as the significance of each variable, and we can finally discover which variable or which node is more important than the others. Such information can help us to further improve the structure of system.

C. *Classification Results*

All the samples are divided into training set and validation set. We also applied 10-fold cross validation to make full use of all samples.

About the evaluation criterion, besides accuracy, we also use F1 value, which is a classical criterion in machine learning. It balances the precision and recall, and it is computed as follows:

$$F_1 = \frac{2 \times precision \times recall}{precision + recal}$$

And the accuracy and F1 value of all 9 bused are presented in Table 4.1

TABLE 4.1 Classification results

Bus	**Bus16**	**Bus18**	**Bus19**	**Bus20**	**Bus21**	**Bus22**	**Bus23**	**Bus29**	**Bus9**
Accuracy	0.98	0.96	0.78	0.99	0.82	0.84	0.82	0.98	0.78
F1 value	2.42	1.66	1.82	1.70	1.64	2.15	1.83	1.68	2.32

V. OFF-LINE LEARNING AND ON-LINE PREDICTION

Real-time Grid Analysis System (RGAS) is a platform to analyze grid data and predict grid behavior by machine learning algorithm as described above. The basic requirements for RGAS is to process grid data in real-time. Supervised learning is adopted in GRAS. Considering the data features and learning process, the problem settings of RGAS are listed as below:

- Real-time analysis of grid status stream data. The grid data is generated in real-time and RGAS should continuously analyze the stream data and respond the predicted behavior before the actual grid behavior actions.
- Fast training of online grid dataset. The conventional training dataset of supervised learning is batched and offline. In RGAS, the evolvement of grid behavior patterns are taken into account. Online data is appended to training dataset to incorporate grid behavior pattern transition.
- Real-time analysis model update. The training process constantly updates the analysis model. The analysis process should consequently updates the model parameters or adopts the new model.

- Basic requirements for big data analysis including high performance, high availability and fault-tolerance.

Based on the problem settings above, RGAS adopts the following software projects to construct the distributed data analysis platform.

- Hadoop HDFS: grid data storage for training
- Spark: fast machine learning to train the analysis model
- Storm: real-time process for grid stream data
- Kafka: data channel for data and model exchange between training and analysis modules
- D3.js: data visualization in web frontend

The architecture of RGAS is shown as the Figure 5.1. The initial offline training data is stored in HDFS. The initial model is trained by spark and installed into real-time analysis storm platform. The online analysis module can be segmented into 2 sections: data spout and analysis bolt. Data spout is to gather online grid data and transfer to data storage module. Analysis bolt is to analyze the grid data by latest model. During the process, the online data is transferred through kafka data pipeline to retrain the model and the updated model is responded to the analysis bolt. The data visualization module runs as web server which renders the grid status chart with the stream data and results.

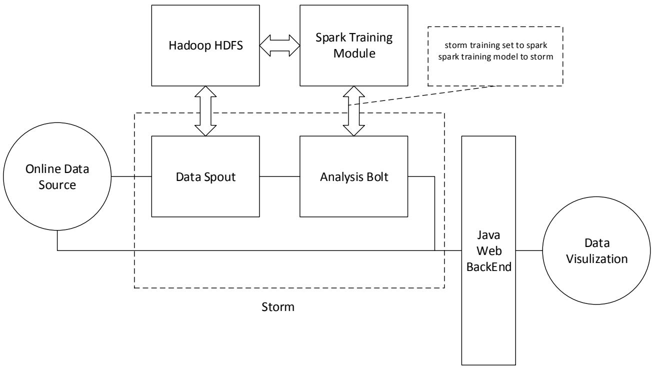

Figure 5.1 Architecture of RGAS

VI. Conclusions

A voltage stability prediction scheme based on machine learning has been presented in this paper. Our main contribution is that we introduce the information of the network and consider how different nodes in system influences each other rather than just extract features in time domain. This model also put a time interval between feature period and prediction moment, which makes remedial measures possible.

Our future work will focus on how to describe the network accurately and how to present these information in machine learning. At present, features are just voltage, current, active power, reactive power and their derivatives which seems to be relatively simple. Next we will develop more complex features such as frequency, flow direction of reaction power, etc.

Acknowledgements

This work was supported in part by National Natural Science Foundation of China (grants No. 61472200 and No. 61233016), Ministry of Science and Technology of China under National 973 Basic Research Program (grant No. 2013CB228206), Delta Electronics, and China Southern Power Grid R&D project (K-SZ2012-026).

References

[1] Fouad, Abdel-Azia, and Vijay Vittal. Power system transient stability analysis using the transient energy function method. Pearson Education, 1991.

[2] Moulin, L. S., M. A. El-Sharkawi, and R. J. Marks. "Support vector machines for transient stability analysis of large-scale power systems." Power Systems, IEEE Transactions on 19.2 (2004): 818-825.

[3] Zhou, Debbie Q., Udaya D. Annakkage, and Athula D. Rajapakse. "Online monitoring of voltage stability margin using an artificial neural network." Power Systems, IEEE Transactions on 25.3 (2010): 1566-1574.

[4] Wehenkel, Louis, and Mania Pavella. "Decision trees and transient stability of electric power systems." Automatica 27.1 (1991): 115-134.

[5] Xu, Yan, et al. "Real-time transient stability assessment model using extreme learning machine." IET generation, transmission & distribution 5.3 (2011): 314-322.

[6] Gomez, Francisco R., et al. "Support vector machine-based algorithm for post-fault transient stability status prediction using synchronized measurements."Power Systems, IEEE Transactions on 26.3 (2011): 1474-1483.

[7] You, Dahai, et al. "Transient stability assessment of power system using support vector machine with generator combinatorial trajectories inputs."International Journal of Electrical Power & Energy Systems 44.1 (2013): 318-325.

[8] Muyeen, S. M., Hany M. Hasanien, and Ahmed Al-Durra. "Transient stability enhancement of wind farms connected to a multi-machine power system by using an adaptive ANN-controlled SMES." Energy Conversion and Management 78 (2014): 412-420.

[9] Lv, Jiaqing, Miroslaw Pawlak, and Udaya D. Annakkage. "Prediction of the transient stability boundary using the lasso." Power Systems, IEEE Transactions on 28.1 (2013): 281-288.