LEARNING TO INFER VOLTAGE STABILITY MARGIN USING TRANSFER LEARNING

Jiaming Li, Yue Zhao*, Young-hwan Lee† and Seung-Jun Kim†*

*Department of ECE, Stony Brook University, Stony Brook, NY 11794
†Department of CSEE, University of Maryland, Baltimore County, Baltimore, MD 21250
Emails: jiaming.li@stonybrook.edu, yue.zhao.2@stonybrook.edu, lee43@umbc.edu, sjkim@umbc.edu

ABSTRACT

Preventing voltage collapse is critical for reliable operation of power systems. A challenging problem is that the voltage stability margin, i.e., the distance from a given power profile to the voltage stability boundary, is very computationally intensive to obtain. A novel machine learning based approach for real-time inference of voltage stability margin is developed, only needing a very small number of offline-computed voltage stability margin data. An accurate margin predictor is trained by first training a binary stability classifier and then transferring this pre-trained model to fine-tune on the small data set of margins. Numerical simulations demonstrate that the proposed method significantly outperforms Jacobian-based voltage stability margin estimation with even faster real-time computation.

1. INTRODUCTION

Power system is a critical infrastructure that supports all sectors of our society. The reliability of power system operation is thus of paramount importance. Voltage collapse has been one of the major causes for large-scale blackouts [1]. With renewable energies increasingly integrated into power systems, their volatilities need to be handled in system operations, pushing the system even closer to its physical capacity. As a result, determining the system stability margin in real time is greatly valuable for system operators to maintain situational awareness of the system and a safe operating margin.

1.1. Related Work and Challenges

Computing the power system voltage stability margin amounts to finding the distance from the current operating point to the boundary of the stability region. There have been considerable efforts in developing methods for estimating the voltage stability margin. A major limitation of the existing approaches, however, is that they are primarily for exploring the voltage stability limit *along a specific loading direction*. A classic tool is the continuation power flow (CPF) method [2], which checks voltage stability from an operating point along a given loading direction until it reaches voltage instability. An energy-function-based method has been proposed in [3]. Another method is point-of-collapse (PoC) [4]. PoC and CPF methods have also been jointly used to find the voltage collapse point in large AC/DC systems in [5]. Neural networks (NNs) are employed to predict the loading margin in [6], and for solving security-boundary constrained optimal power flow problem in [7]. A hybrid neural network that contains a Kohonen network and a multi-layered neural network is proposed in [8].

As mentioned above, the main drawback of these methods is that they only address estimating voltage stability margin along a particular loading direction. However, finding the true distance of an operating point to the boundary of the stability region requires identifying the “worst-case” loading direction along which the current operating point is the *closest* to voltage instability. In other words, the voltage stability margin is the distance from the operating point to the voltage *instability region* (i.e., the complement of the voltage stability region). Finding this worst-case direction is unfortunately very computational challenging due to a number of reasons. First, the voltage stability/instability region is defined with nonlinear AC power flow equations. The voltage instability region is not only non-convex, but also does not have an efficient way of characterization. Thus, to find the worst-case loading direction, one needs to resort to sampling a *very large* number of directions, and testing along each direction with, e.g., the CPF method. The heavy computational intensity of finding the worst-case loading direction is further compounded by the fact that power systems often have *very high dimensional* voltage stability region. As such, it is impractical to run, in real time, CPF along a large number of sampled directions to obtain a satisfactory estimate of the voltage stability margin.

Sidestepping the problem of finding the worst-case loading direction, another approach is to approximate the voltage stability margin using easily computable metrics, notably the smallest singular value (SSV) of the Jacobian matrix [9, 10], computed from power flow algorithms. Such metrics, however, are only approximate, and we will show that significantly more accurate margin estimates can be obtained with our method to be proposed.

This work was supported in part by the SBU-BNL SEED Grant.

1.2. Learning-to-Infer with Transfer Learning

In this work, we develop a machine learning based approach for fast inference of voltage stability margin. In particular, a "Learning-to-Infer" approach is employed [11]: Given the power system model, *offline* computation are performed which include a) generating samples of stable and unstable operating points, and b) learning a predictor from these simulated samples. Then, the trained predictor will be employed for *online* inference of voltage stability margin. Two key advantages of this method are that, a) as offline computation is much less time and resource constrained, intensive computation can be afforded to learn the complicated mapping from operating points to voltage stability margins, and b) using the trained predictors for margin inference is computationally very fast, and can easily satisfy stringent real-time application constraints. In particular, predictors based on neural networks are employed in the current work.

However, granted that offline training can afford intensive computation, a fundamental difficulty arises that makes even offline training of a voltage stability margin predictor very challenging: *the lack of labels*. As noted above, even for just one operating point, it is very computationally intensive to obtain an accurate approximation of its voltage stability margin. Effective training, however, typically requires a *very large* number of samples of operating points *with their voltage stability margins computed as labels*. It is impractical, even in an offline fashion, to generate a sufficiently large labeled data set. To address this critical challenge, the key idea we exploit is one of *transfer learning*. In particular, while it is difficult to generate the margin label for an operating point, it is very computationally efficient to evaluate if the operating point is stable or not. Accordingly, we generate two data sets: a *large* set of operating points with easily computed binary labels of stable or not, and a *small* set of operating points of intensively computed (approximate) margin labels. With these, we first train a binary classifier based on the large data set, and then transfer the learned hidden layer of the neural network for fine tuning based on the small data set. We will demonstrate that the margin predictor trained with such two-step transfer learning significantly outperforms existing benchmarks based on Jacobians.

2. PROBLEM FORMULATION

With given parameters of a power system, a power profile $\boldsymbol{s}$, i.e., the real and reactive power injections P_k and Q_k of all the buses k[1], can either induce a voltage collapse or not. A voltage collapse means that, given $\boldsymbol{s}$, the following AC power flow equations do not admit a feasible voltage solution [12]: $\forall k = 1, 2, ..., N$,

$$P_k = V_k \sum_{n=1}^{N} Y_{kn} V_n cos(\delta_k - \delta_n - \theta_{kn}), \tag{1}$$

$$Q_k = V_k \sum_{n=1}^{N} Y_{kn} V_n sin(\delta_k - \delta_n - \theta_{kn}), \tag{2}$$

where $\left[Y_{kn} e^{j\theta_{kn}}\right]$ is the complex bus admittance matrix. In this case, we call this power profile $\boldsymbol{s}$ unstable. For a power system, its voltage stability region is defined to be the set of all power profiles that do not induce voltage collapse. We denote the stability region of a given power network by $\mathcal{S}$, and its complement, i.e., the voltage *instability* region, by $\mathcal{S}^c$.

The central question we seek to address is that, for an operating point $\boldsymbol{s}$, what is the *minimum* power injection perturbation that, if applied, would lead to a voltage collapse? In other words, what is the *distance* of $\boldsymbol{s}$ to the voltage instability region $\mathcal{S}^c$? Knowing this distance is immensely helpful for system operators to assess the current system reliability status, and take preventive actions for avoiding voltage collapse. Formally, we would like to solve for

$$\text{dist}\left(\boldsymbol{s}, \mathcal{S}^c\right) \triangleq \min_{\boldsymbol{s}' \in \mathcal{S}^c} \|\boldsymbol{s} - \boldsymbol{s}'\|_2. \tag{3}$$

However, from the AC power flow equations (1) and (2), neither is the instability region $\mathcal{S}^c$ convex, nor does it have a efficient (let alone closed form) characterization. As such, to solve (3), one needs to sample a large number of directions (a.k.a. loading directions), and search along each of them starting from $\boldsymbol{s}$ by running CPF [2]. Clearly, the more directions are searched, the better accuracy would be obtained in approximately computing $\text{dist}\left(\boldsymbol{s}, \mathcal{S}^c\right)$. However, searching, e.g., 1000 loading directions for the IEEE 118-bus system would take around 4 minutes for a computer with 3.6GHz Intel Xeon CPU and 32GB RAM. For medium to large scale power systems, $\boldsymbol{s}$ can easily have hundreds to thousands of dimensions, and it would take a rather long time for even approximately computing $\text{dist}\left(\boldsymbol{s}, \mathcal{S}^c\right)$ for *one* operating point.

3. OVERVIEW OF OUR APPROACH

Our goal is to not incur the computational cost of searching many directions and running CPF, and yet achieve *real-time* computation of equally accurate estimates of voltage stability margin. The first idea we employ is to exploit offline computation to train margin predictors for online margin inference, in a spirit similar to the "Learning-to-Infer" method proposed in [11, 13]. In particular, given the power system, the offline computation consists of a) simulating a data set of operating points, ideally with their "labels", i.e., voltage stability margins, and b) training a predictor based on the simulated data set for future online margin inference.

However, one key challenge arises: As discussed in Section 2, computing the label accurately even for just one simulated operating point would consume a considerable time. As

[1] More generally, one can consider a subset of buses of interests, which can also include PV buses in addition to PQ buses.

such, to construct a relatively large data set, even to be done offline, is not practical, and yet is crucial for effective training especially for high dimensional inference as needed in power systems of reasonable sizes. This difficulty practically limits our ability to use direct supervised learning for training a margin predictor, simply because it is too time consuming to generate a sufficiently large labeled data set.

To overcome this challenge, the key idea stems from exploiting the following fact: For an operating point, while computing its voltage stability margin by searching is very time consuming (e.g., minutes), verifying *whether it is stable or not*, nonetheless, is very fast (e.g., milliseconds). Thus, within similar time limits, we can generate a data set of *[operating point, binary stability label]* with a size many orders of magnitude larger than a data set of *[operating point, voltage stability margin label]*. As such, while it is infeasible to generate a margin-labeled data set sufficiently large to capture the high dimensional boundary of the voltage stability region $\mathcal{S}$, it is feasible to generate a sufficiently large binary stability-labeled data set that does so. The problem is, however, training on a data set with only the binary stability labels does not offer us a predictor that outputs stability margins.

To overcome this challenge, our key step forward is to use *transfer learning* to jointly exploit both the information embedded in a large binary stability-labeled data set and that in a small margin-labeled data set, with the end goal of obtaining an accurate voltage stability margin predictor. In particular, we a) train a neural network (NN) based binary classifier from a large binary stability-labeled data set, b) take the trained hidden layer of the NN as a feature extractor, with the hope that it implicitly captures sufficient information of the boundary of $\mathcal{S}$, and c) add an additional layer of NN to fine tune based on only a small margin-labeled data set. In a sense, we transfer the knowledge learned in the binary classifier in order to make it tractable to learn a margin predictor based on only a small data set with stability margin labels.

4. DESCRIPTION OF DATA SETS AND LEARNING METHOD

4.1. Data Set Generation

To construct a binary stability-labeled data set, we seek to find a large number of pairs of stable and unstable operating points $(\boldsymbol{s}_i, \boldsymbol{s}_o)$ *near the boundary* of $\mathcal{S}$. We work with the IEEE 118-bus system, and use MATPOWER [14] to find such boundary sample pairs $(\boldsymbol{s}_i, \boldsymbol{s}_o)$ by running CPF starting from the origin along different directions toward different target points. To generate the target points, we sample the power factor for each bus with uniform distribution $U[0.4, 1]$. We record labeled data pairs as $(\boldsymbol{s}_i, 1)$ and $(\boldsymbol{s}_o, 0)$, where 1 indicates voltage stability and 0 indicates instability. We generate a total of $200K$ data points.

To construct a stability margin-labeled data set, for any

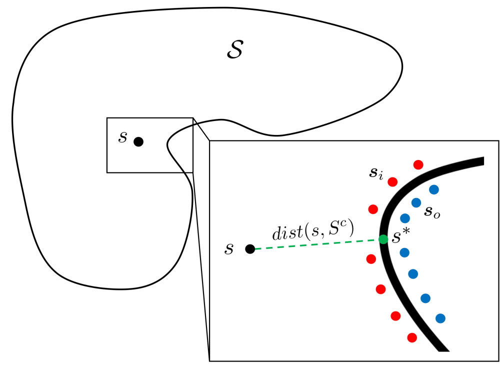

Fig. 1. Finding the margin distance

two feasible operating points $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$ generated above, we shrink them by some random factors. We then apply the CPF algorithm with a starting point $\boldsymbol{s}_1$, searching along the direction of $\boldsymbol{s}_2 - \boldsymbol{s}_1$ to find the distance to $\mathcal{S}^c$ along this particular direction. For each randomly picked $\boldsymbol{s}_1$, we search along $7,000$ directions, each time with a different randomly picked $\boldsymbol{s}_2$, and pick the minimum distance (among $7,000$ ones) as the approximate margin for this $\boldsymbol{s}_1$. We repeat this procedure for $1,000$ different $\boldsymbol{s}_1$ and generate $1K$ data points.

4.2. Training a Binary Stability Classifier

Based on the $200K$ data set of *[operating point* $\boldsymbol{s}$*, binary stability label* $0/1$*]*, we train a neural network classifier $\hat{h}(\boldsymbol{s})$ with one hidden layer and ReLU activation [15] using Tensorflow [16]. Hinge loss is applied at the output layer [17].

For weight initialization, we employ the strategy as in [18]. We use a stochastic gradient descent (SGD) optimizer with momentum and Nesterov acceleration. A mini-batch size of 200 is employed. The learning rate is 10^{-5} and the momentum is 0.9. l_2 regularization is applied to the hidden layer, with a factor of 6.0.

4.3. Training a Voltage Stability Margin Predictor using Transfer Learning

Based on the only $1K$ data set of *[operating point* $\boldsymbol{s}$*, stability margin* $\mathrm{dist}(\boldsymbol{s}, \mathcal{S}^c)$*]*, we first import the weights of the hidden layer from the pre-trained binary classifier $\hat{h}(\boldsymbol{s})$ as a feature extractor, and then add another hidden layer for regression on the $1K$ margin-labeled data set to train an overall margin predictor $\hat{d}(\boldsymbol{s})$. Mean squared error (MSE) is employed as the loss function. In the SGD optimizer, the learning rate is set to 10^{-6} and the momentum is 0.9. During training, we do not alter the weights of the first hidden layer transferred from the binary classifier, but only fine tune the second hidden layer and the output layer.

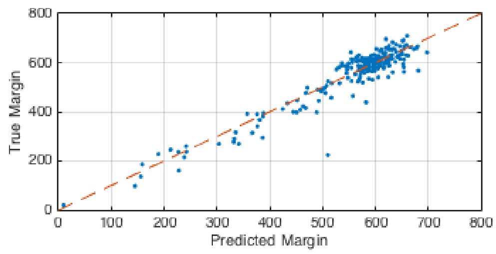

Fig. 2. Scatter plot from using the proposed method.

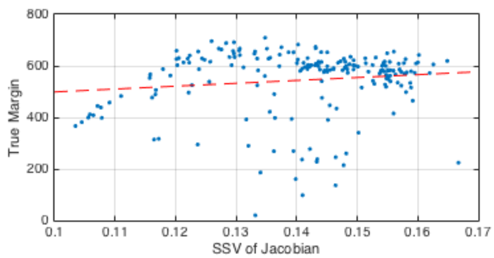

Fig. 3. Scatter plot from using Jacobian' SSVs.

5. NUMERICAL EXPERIMENTS

5.1. Benchmark Method using the Jacobian

As a benchmark method, we employ a widely used voltage stability margin approximator — the smallest singular value (SSV) of the Jacobian matrix from running power flow algorithms [9, 10]. In particular, we use the Newton-Raphson method to solve the power flow equations (1) and (2). As the iterations converge, the SSV of the Jacobian matrix is computed. The SSV provides us a measure of how close the Jacobian is to singularity, implying voltage instability. We evaluate the SSVs as the predictors to fit the $1K$ margin-labeled data set. The resulting MSE is $15,876$.

5.2. The Proposed Method with Transfer Learning

In evaluating our method, for the $200K$ binary stability labeled data points, we use $160K$ data to train, and $40K$ for testing. We have 64 neurons in the hidden layer for the classifier $\hat{h}(\boldsymbol{s})$. The data are normalized, with per-bus mean subtracted and per-bus standard deviation divided. The trained classifier $\hat{h}(\boldsymbol{s})$ achieves an testing accuracy of 0.98. From this, we see that the classifier's decision boundary accurately approximates the boundary of the voltage stability region.

Next, we transfer the hidden layer of the trained classifier $\hat{h}(\boldsymbol{s})$ to learn a stability margin predictor $\hat{d}(\boldsymbol{s})$ based on the $1K$ margin-labeled data set: 800 data are for training, and 200 for testing. We use 256 neurons for the newly added layer in the predictor $\hat{d}(\boldsymbol{s})$. The trained predictor $\hat{d}(\boldsymbol{s})$ achieves a testing MSE of $1,624$. A scatter plot is shown in Fig. 2. In comparison, a scatter plot with the benchmark method using Jacobian's SSV is shown in Fig. 3. It is clear both from the MSEs and the scatter plots that the proposed method significantly outperforms the benchmark using Jacobian's SSV.

Table 1. MSEs on the Testing Set

Method	Jacobian's SSV	Direct Learning	**Transfer Learning**
Testing MSE	15,876	4,817	**1,624**

To further validate whether transfer learning is the key to the success that we observed, we also perform direct supervised learning on the $1K$ margin-labeled data set, without transferring from a pre-trained binary classifier. The best performance we can achieve with this "direct learning" method is a testing MSE of $4,817$. This is achieved by training a NN with one hidden layer of 512 neurons. Interestingly, it already outperforms the benchmark method based on Jacobian's SSV. However, using transfer learning further improves the performance significantly. In summary, the performance comparison of the benchmark (Jacobian's SSV), the direct learning and the transfer learning methods are compared in Table 1.

In addition to its accuracy, the learned predictor $\hat{d}(\boldsymbol{s})$ is computationally very fast when deployed for real-time inference. In particular, a forward pass of the NN (margin prediction for one operating point) takes only $2.5\mu s$ on a computer with a 3.6GHz Intel Xeon CPU and a 32GB RAM. In contrast, searching $7,000$ loading directions using CPF takes over 25 minutes for a power profile on the same computer. As such, our proposed method achieves about 7 orders of magnitude speed-up with only very small loss of accuracy.

6. CONCLUSIONS

We have developed a learning-based method for fast inference of voltage stability margin in real time. A Learning-to-Infer framework is employed to exploit offline power system model-based simulation and data-driven training. A transfer learning scheme is developed to overcome the fundamental challenge of lack of margin labels due to their extremely high computation cost. Specifically, a large data set with easily computable binary stability labels is generated, and used to train a binary classifier that implicitly capture the voltage stability boundary. Next, a small data set with intensively computed stability margin labels is generated. With the hidden layer pre-trained from the binary classifier transferred as a feature extractor, fine tuning for a margin predictor is then performed based only on the small data set. The developed method is demonstrated to significantly outperform the state-of-the-art benchmark using the Jacobian's singular values.

7. REFERENCES

[1] G. Andersson *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*, vol. 20, no. 4, pp. 1922–1928, 2005.

[2] V. Ajjarapu and C. Christy, “The continuation power flow: A tool for steady state voltage stability analysis,” *IEEE Transactions on Power Systems*, vol. 7, pp. 416–423, 1992.

[3] R. P. Klump and T. J. Overbye, “Assessment of transmission system loadability,” *IEEE Transactions on Power Systems*, vol. 12, no. 1, pp. 416–423, 1997.

[4] T. Van Cutsem, “A method to compute reactive power margins with respect to voltage collapse,” *IEEE Transactions on Power Systems*, vol. 6, no. 1, pp. 145–156, 1991.

[5] C. A. Canizares and F. L. Alvarado, “Point of collapse and continuation methods for large AC/DC systems,” *IEEE Transactions on Power Systems*, vol. 8, no. 1, pp. 1–8, 1993.

[6] X. Gu and C. A. Cañizares, “Fast prediction of loadability margins using neural networks to approximate security boundaries of power systems,” *IET Generation, Transmission & Distribution*, vol. 1, no. 3, pp. 466–475, 2007.

[7] V. J. Gutierrez-Martinez, C. A. Cañizares, C. R. Fuerte-Esquivel, A. Pizano-Martinez, and X. Gu, “Neural-network security-boundary constrained optimal power flow,” *IEEE Transactions on Power Systems*, vol. 26, no. 1, pp. 63–72, 2011.

[8] H. Wan and Y. Song, “Hybrid supervised and unsupervised neural network approach to voltage stability analysis,” *Electric Power Systems Research*, vol. 47, no. 2, pp. 115–122, 1998.

[9] A. Tiranuchit, L. Ewerbring, R. Duryea, R. Thomas, and F. Luk, “Towards a computationally feasible on-line voltage instability index,” *IEEE Transactions on Power Systems*, vol. 3, no. 2, pp. 669–675, 1988.

[10] P.-A. Lof, T. Smed, G. Andersson, and D. Hill, “Fast calculation of a voltage stability index,” *IEEE Transactions on Power Systems*, vol. 7, no. 1, pp. 54–64, 1992.

[11] Y. Zhao, J. Chen, and H. V. Poor, “A Learning-to-Infer method for real-time power grid multi-line outage identification,” *IEEE Transactions on Smart Grid, to appear*.

[12] J. D. Glover, M. Sarma, and T. Overbye, *Power System Analysis & Design*. Cengage Learning, 2011.

[13] Y. Lee, Y. Zhao, S.-J. Kim, and J. Li, “Predicting voltage stability margin via learning stability region boundary,” in *Proc. 2017 IEEE 7th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing*. IEEE, 2017, pp. 1–5.

[14] R. D. Zimmerman, C. E. Murillo-Sánchez, and R. J. Thomas, “Matpower: Steady-state operations, planning, and analysis tools for power systems research and education,” *IEEE Transactions on power systems*, vol. 26, no. 1, pp. 12–19, 2011.

[15] V. Nair and G. E. Hinton, “Rectified linear units improve restricted boltzmann machines,” in *Proceedings of the 27th International Conference on Machine Learning*, 2010, pp. 807–814.

[16] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard *et al.*, “Tensorflow: A system for large-scale machine learning,” in *Proc. 12th USENIX Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, 2016, pp. 265–283.

[17] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[18] K. He, X. Zhang, S. Ren, and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1026–1034.