Predicting Voltage Stability Margin via Learning Stability Region Boundary

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

†Dept. of Computer Science and Electrical Engineering
University of Maryland, Baltimore County, Baltimore, MD 21250

*Dept. of Electrical and Computer Engineering
Stony Brook University, Stony Brook, NY 11794

***Abstract*—Preventing voltage collapse is critical for the reliable operation of the power grid. In this paper, the voltage stability margin, which is defined as the distance from a given power profile to the boundary of the stability region, is efficiently estimated using a data-driven machine learning approach. The key idea is to train a neural network classifier to learn the boundary of the potentially nonconvex stability region, and exploit the resulting score metric as the regressor for stability margin prediction. No particular loading direction is assumed, but rather the minimum distance to the boundary along all possible directions is captured. The training samples are generated from both continuation and semidefinite relaxation power flow methods. The performance and computational advantage of the proposed approach are verified by numerical experiments.**

I. Introduction

The resilience of the power grid to variations in generation, load, and grid topology is critical for power quality as well as national economy and security. Volatile renewable energy resources are increasingly integrated to the grid, pushing the power grid operation to its physical capacity. In this context, maintaining grid stability is becoming an issue of utmost importance with prominent challenges.

Power delivery in the grid is governed by a set of nonlinear power flow equations. As the grid operating point is driven nearer to the boundary of the feasible region of the power flow constraints, bus voltages start to fall gradually until they experience a sudden and steep drop—a condition referred to as *voltage collapse*. It is a manifestation of grid instability and typically occurs when the power system is heavily loaded with insufficient supply of reactive power [1]. Voltage collapse has often been attributed as a contributing factor to cascading failures and blackouts [2].

Voltage collapse occurs at the point of saddle-node bifurcation of the dynamic power system equations, which can be characterized as the steady-state power flow Jacobian matrix being singular [3], [4]. The distance between an operating point and the bifurcation point represents the voltage stability margin. To obtain a critical point, a simple approach would be to utilize the ordinary power flow solver and increase the loading along a certain direction until the solution diverges. However, this still might not reveal the exact loadability limit due to the convergence problem near a singular Jacobian.

The continuation method mitigates this issue by following a predetermined loading direction and iteratively predicting the next solution using the tangent to the solution path [3]. The point-of-collapse (PoC) method is another computationally feasible method [5]. Both the PoC and continuation methods were employed to find the voltage collapse points in large AC/DC systems in [4]. However, these methods require a loading direction to be specified and the loadability margin is computed as the scalar loading factor in the loading direction. Based on the observation that the dynamics of state variables close to saddle-node bifurcation is determined by a single degree of freedom, a novel indicator of voltage collapse was obtained in [6]. A method for estimating the voltage stability margin using real-time PMU measurements was proposed employing a cubic spline extrapolation technique in [7].

Recently, theoretical insights into the voltage stability margin were gained by taking into account the network topology and reactive power demands, where a close-form estimate for the largest nodal voltage deviation was derived [8]. However, the network was assumed to be lossless and decoupled reactive power flow equations were used. As a consequence, it was observed that the metric does not predict well the actual margin when the reactive power demands were low.

In [9], an artificial neural network was employed to predict the loading margin. The neural network was trained using the loading direction as the input and the distance from the base loading to the boundary as the output. Based on this neural network, the loadability margin from an arbitrary operating point along an arbitrary loading direction could be estimated. However, it was tacitly assumed that the feasibility region was convex, which can be violated especially as the size of the grid increases [10]. A similar technique was employed to solve a security-boundary constrained optimal power flow in [11].

Mitigating the issue of the non-convex feasibility region, a sufficient condition for insolvability of the AC power flow was obtained through convex relaxation techniques in [10], [12]. The formulated optimization problem can provide the distance to the outer-boundary of the feasibility region in a given loading direction, where the outer-boundary is due to the convexified set that contains the true nonconvex feasibility region of nonlinear power flow equations.

A major limitation of the many existing works on obtaining voltage stability margin is that a particular loading direction needs to be specified along which the margin is computed. However, it is of great interest to determine which direction is the most vulnerable one among all possible directions, and what the stability margin is along this worse-case direction.

In this paper, we aim to compute the voltage stability margin that captures the minimum distance from a given operating point to the stability boundary along all possible directions. Our method trains a neural network as a classifier that characterizes the boundary of the potentially nonconvex stability region. Then, based on a score computed inside the neural network, the minimum distance to the stability boundary in the most vulnerable loading direction is predicted using linear regression, *without requiring a specific loading direction*. Since existing methods would need to perform exhaustive search over all possible loading directions to determine the most vulnerable one, our method has an over 10^5x speed advantage in approximately computing voltage stability margins.

The rest of the paper is organized as follows. The problem investigated is stated in Sec. II. A brief review on computing the static voltage stability margin is presented in Sec. III. The neural network architecture and its training are discussed in Sec. IV. Numerical test results based on realistic grid topologies are presented in Sec. V. Conclusions are discussed in Sec. VI.

II. Problem Statement and Overview of Approach

We consider a power network whose parameters are arbitrarily given. In a network, a profile $\boldsymbol{s}$ of power injections (real and reactive) at all the buses either induces a voltage collapse, or not. We define the *stability region* of the given network as the set of all power profiles that do not induce a voltage collapse, denoted by $\mathcal{S}^*$. We denote the boundary of $\mathcal{S}^*$ by $\mathcal{B}(\mathcal{S}^*)$. We also interchangeably term $\boldsymbol{s}$ as an "operating point" of the system.

A power profile $\boldsymbol{s}$ belongs to $\mathcal{S}^*$ if the corresponding power flow equations can be solved. More importantly, we would like to know *how far $\boldsymbol{s}$ is from the boundary of the stability region $\mathcal{B}(\mathcal{S}^*)$*. We term this distance to the boundary as the *voltage stability margin* of an operating point $\boldsymbol{s}$, denoted by $\operatorname{dist}(\boldsymbol{s}, \mathcal{B}(\mathcal{S}^*)) \triangleq \min_{\boldsymbol{s}' \in \mathcal{B}(\mathcal{S}^*)} \|\boldsymbol{s} - \boldsymbol{s}'\|$, where the norm can be chosen according to practical interests. We note that, the voltage stability margin does not specify any direction along which a distance is computed, but considers all possible directions. As a result, the voltage stability margin provides the critical information of whether an operating point $\boldsymbol{s}$ is safe or stable with a sufficient margin.

It is worth noting that a voltage stability margin $\operatorname{dist}(\boldsymbol{s}, \mathcal{B}(\mathcal{S}^*))$ is computationally very challenging to obtain, primarily for two reasons: a) the boundary $\mathcal{B}(\mathcal{S}^*)$ does not admit a computationally efficient representation, let alone any closed-form expression; and b) the problem is typically a high-dimensional one. We also note that $\mathcal{S}^*$ is in general not convex. As a result, it is very costly to compute, or accurately approximate, the voltage stability margin $\operatorname{dist}(\boldsymbol{s}, \mathcal{B}(\mathcal{S}^*))$ using existing methods.

In this paper, we investigate efficient approximate computation of voltage stability margin $\operatorname{dist}(\boldsymbol{s}, \mathcal{B}(\mathcal{S}^*))$. The main idea is to leverage a sufficient number of operating points efficiently sampled inside and outside $\mathcal{S}^*$ to learn a predictor $\hat{d}(\boldsymbol{s})$ of the voltage stability margin. In particular, the proposed method proceeds as follows:

1) Sample operating points inside and outside the stability boundary $\mathcal{B}(\mathcal{S}^*)$.
2) Based on the samples, learn a binary classifier $\hat{h}(\boldsymbol{s})$ for classifying whether an operating point is stable or not.
3) Based on the learned classifier $\hat{h}(\boldsymbol{s})$, predict not only the label, but also the voltage stability margin of any operating point $\boldsymbol{s}$, $\hat{d}(\boldsymbol{s})$.

Next, we first provide the details of computing the stability region boundary in Sec. III, and then of the learning and prediction steps in Sec. IV. It is worth highlighting that, in the proposed approach, there are only minimal instances of the costly computation of the actual voltage stability margin $\operatorname{dist}(\boldsymbol{s}, \mathcal{B}(\mathcal{S}^*))$, as will be shown in Sec. IV.

III. Computation of Voltage Stability Region Boundary

Here we briefly review two methods for computing the boundary points of the voltage stability region, which will be used in Sec. V-A to generate samples of the operating points inside and outside $\mathcal{S}^*$. Let $P_{G,i}$, $P_{L,i}$, $P_{T,i}$ represent the generated real power, the real load, and the injected real power, respectively, at bus i. Likewise, let $Q_{G,i}$, $Q_{L,i}$ and $Q_{T,i}$ be the reactive power counterparts. The AC power flow equations are given by

$$P_{T,i} - \sum_{j=1}^{n} V_i V_j y_{ij} \cos(\delta_i - \delta_j - \nu_{ij}) = 0, \ \forall i \tag{1}$$

$$Q_{T,i} - \sum_{j=1}^{n} V_i V_j y_{ij} \sin(\delta_i - \delta_j - \nu_{ij}) = 0, \ \forall i \tag{2}$$

where n is the number of buses in the grid, $V_i \angle \delta_i$ is the voltage at bus i, $y_{ij} \angle \nu_{ij}$ is the (i,j)-entry of the network admittance matrix $\mathbf{Y}$, and

$$P_{T,i} = P_{G,i} - P_{L,i} \tag{3}$$

$$Q_{T,i} = Q_{G,i} - Q_{L,i} \tag{4}$$

for all i. Set $\mathcal{S}^*$ can be defined as $\{\boldsymbol{s} = (\{P_{L,i}\}, \{Q_{L,i}\}, \{P_{G,i}\}) : \text{(1)–(4) are satisfied}\}$. To simulate the changes in the power profile, a loading factor $\eta \geq 0$ is introduced to form a test power profile $\boldsymbol{s}(\eta) := (\{P_{L,i}\}, \{Q_{L,i}\}, \{P_{G,i}\})$ with

$$P_{L,i} = P_{L,i}^{base} + \eta(P_{L,i}^{targ} - P_{L,i}^{base}) \tag{5}$$

$$Q_{L,i} = Q_{L,i}^{base} + \eta(Q_{L,i}^{targ} - Q_{L,i}^{base}) \tag{6}$$

$$P_{G,i} = P_{G,i}^{base} + \eta(P_{G,i}^{targ} - P_{G,i}^{base}) \tag{7}$$

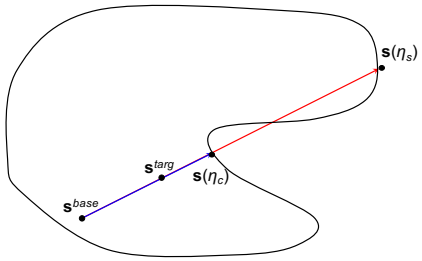

Fig. 1. Power profiles in a nonconvex stability region

where the superscripts ^{base} and ^{targ} represent the base and the target profiles, respectively. The goal is to find the critical η^* such that $\boldsymbol{s}(\eta^*)$ lies on $\mathcal{B}(\mathcal{S}^*)$.

The continuation power flow method yields a path of power flow solutions starting from a base power profile and gradually increasing η until the stability limit is reached [3]. A predictor-corrector scheme is employed, where the next point in the path is predicted using the tangent of the power flow surface $\mathbf{f}(\{V_i\},\{\delta_i\},\eta) = \mathbf{0}$. The additional parameter η allows to avoid the singularity in the (augmented) Jacobian across the solution path. The predicted point is then corrected via the Newton-Raphson method. At the critical point, the tangent component corresponding to η is zero. Thus, critical η_c can be determined by checking whether η starts to decrease.

The entire solution path from the continuation power flow is *inside* the feasible region of the power flow equations. However, since the feasible region is not necessarily convex, the obtained boundary point does not necessarily correspond to the maximum η that can be achieved. The semidefinite programming (SDP) relaxation approach in [12] yields an upper-bound on η by searching over a convex superset of the true feasibility region. One possible approach is to solve the convex dual of

$$\max \eta \quad \text{subject to (1)–(7).} \tag{8}$$

Due to the potentially non-zero duality gap, it is not guaranteed that the profile $\boldsymbol{s}(\eta_s)$ with solution η_s to (8) is in the feasible region. On the other hand, $\boldsymbol{s}(\eta_s+\epsilon)$ with small $\epsilon > 0$ is guaranteed to be *outside* the feasible region. Fig. 1 shows pictorially the base and target power profiles and the corresponding boundary points for a nonconvex stability region.

These methods yield the distance from an operating point $\boldsymbol{s}$ to the boundary *in a given direction* by $\|\boldsymbol{s}^{base} - \boldsymbol{s}(\eta^*)\|$. In order to compute the margin $\operatorname{dist}(\boldsymbol{s},\mathcal{B}(\mathcal{S}^*))$, one needs to sample a sufficient number of directions and choose the minimum distance, which is computationally very intensive.

IV. Learning Stability Boundary and Predicting Stability Margin

In this section, we present methods to learn the stability boundary $\mathcal{B}(\mathcal{S}^*)$ and approximate voltage stability margin $\operatorname{dist}(\boldsymbol{s},\mathcal{B}(\mathcal{S}^*))$, exploiting sampled operating points generated inside and outside the stability region $\mathcal{S}^*$ (cf. Sec. III). For each sample of operating point $\boldsymbol{s}$, there are two numbers we are interested in: a) a binary label of whether it is stable or not, i.e., $\mathbb{I}(\boldsymbol{s}\in\mathcal{S}^*)$; and b) a real number corresponding to its voltage stability margin $\operatorname{dist}(\boldsymbol{s},\mathcal{B}(\mathcal{S}^*))$.

As our ultimate goal is to predict the voltage stability margin for any given operating point $\boldsymbol{s}$, ideally, one may want to train an "end-to-end" predictor $\hat{d}(\boldsymbol{s})$ based on a sufficient number of sample pairs $(\boldsymbol{s},\operatorname{dist}(\boldsymbol{s},\mathcal{B}(\mathcal{S}^*)))$. However, for an operating point $\boldsymbol{s}$, computing $\operatorname{dist}(\boldsymbol{s},\mathcal{B}(\mathcal{S}^*))$ is computationally very costly. In contrast, computing just the binary label $\mathbb{I}(\boldsymbol{s}\in\mathcal{S}^*)$ is orders of magnitude faster (cf. Sec. III). Due to the large number of samples needed for accurate approximation in the high-dimensional space of $\boldsymbol{s}$, it is practically prohibitive to learn end-to-end with samples of $(\boldsymbol{s},\operatorname{dist}(\boldsymbol{s},\mathcal{B}(\mathcal{S}^*)))$. Rather, we will work with samples of $(\boldsymbol{s},\mathbb{I}(\boldsymbol{s}\in\mathcal{S}^*))$, which can be much more efficiently generated. Moreover, using methods described in Sec. III, operating points *very close* to the stability boundary $\mathcal{B}(\mathcal{S}^*)$ are generated, enabling us to learn accurate approximation of $\mathcal{B}(\mathcal{S}^*)$.

The main question is, with samples of only point-label pairs $(\boldsymbol{s},\mathbb{I}(\boldsymbol{s}\in\mathcal{S}^*))$, how do we build a predictor of the margin $\operatorname{dist}(\boldsymbol{s},\mathcal{B}(\mathcal{S}^*))$, which is missing in these samples? Our approach on addressing this problem is inspired by support vector machine (SVM), in which hinge loss is used in predicting labels.

A. Neural Networks with Hinge Loss

With a sufficient number of samples of $(\boldsymbol{s},\mathbb{I}(\boldsymbol{s}\in\mathcal{S}^*))$, we train multi-layer neural networks to learn classifiers that capture the highly nonlinear boundary $\mathcal{B}(\mathcal{S}^*)$ in the high dimensional space of $\boldsymbol{s}$. At the output layer, we employ the hinge loss

$$L = \max(-xy+1,0) \tag{9}$$

where x is the *score* computed from the output layer, and $y\in\{-1,1\}$ is the binary label. For more details on the score and the hinge loss, the readers are referred to [13].

With a trained classifier, instead of the output label, the critical information we will use is the *score* computed before thresholding. In particular, we employ the score as an indicator for the voltage stability margin. The intuition is similar to SVM, where there is a clear relation between the score of a point and the distance of it to the decision boundary.

B. Regression on Voltage Stability Margins

Now, with the scores computed, we will further build a predictor to compute an approximation of the voltage stability margin as a function of the score. As opposed to learning in high dimensional spaces in the previous step, this score-margin predictor is only *one-dimensional*. As a result, to fit such a predictor, it is sufficient to generate a *much smaller set* of point-margin samples $(\boldsymbol{s},\operatorname{dist}(\boldsymbol{s},\mathcal{B}(\mathcal{S}^*)))$, and perform a regression from the scores of $\boldsymbol{s}$ to the margins $\operatorname{dist}(\boldsymbol{s},\mathcal{B}(\mathcal{S}^*))$.

With the learned predictor from score to margin, and the previously trained neural networks that produce scores for points, we obtain end-to-end predictors from operating point to margin, $\hat{d}(\boldsymbol{s})$. We note that, the proposed method leverages the efficient sampling of binary stability labels of operating points to generate a large set of samples of labels. It only resorts to the costly sampling of margins in the last step of

one-dimensional regression, where only a small set of samples of margins are generated. Accordingly, overall computational efficiency is achieved. A similar idea was used in [14].

V. Numerical Experiments

A. Sample Generation for Learning the Stability Boundary

The samples with which to train the neural network classifier were generated using the 39-bus New England case included in the MATPOWER package [15]. A random power profile vector $\boldsymbol{s}$, which includes the real and reactive powers for the loads and the real powers for the generators across the network, was first generated. To do this, we sampled uniformly the power factor for each load bus i in the interval $[0.4, 1]$. Then the reactive powers were adjusted to yield the sampled power factors. The lagging (or leading) power factors were left lagging (or leading) after the adjustment. The load buses with real or reactive power demands equal to 0 were not changed.

Based on the sampled power profile, two boundary points were then computed (with the base profile in (5)–(7) set to zero). One was computed using the continuation power flow method and the other using the SDP relaxation-based power flow. For the continuation power flow method, the critical η_c that rendered $\eta_c \boldsymbol{s}$ on the boundary of the feasible region was computed. Then, the tuple $(\eta_c \boldsymbol{s}, 1)$ was recorded as a training sample, where label 1 signifies that the profile $\eta_c \boldsymbol{s}$ is *inside* the feasible region. Then, the SDP power flow was used to calculate the critical η_s that rendered $\eta_s \boldsymbol{s}$ on the boundary of the insolvability set. Then, the tuple $(\eta_s \boldsymbol{s}, 0)$ was added to the training set, where label 0 signifies that the profile $\eta_s \boldsymbol{s}$ is *outside* the feasible region. This way, we generated $10,000$ feasible boundary points and another $10,000$ infeasible boundary points.

B. Sample Generation for Learning the Distance to Boundary

In order to compute the minimum distance from an operating point to the stability boundary, a feasible power profile was picked from the dataset already generated in Sec. V-A. Then, a profile $\boldsymbol{s}_1$ in the interior of the stability region was obtained by shrinking the profile by a random amount. Another feasible power profile $\boldsymbol{s}_2$ was similarly obtained. Then, the continuation power flow method was used to obtain a boundary point $\boldsymbol{s}_1 + \eta(\boldsymbol{s}_2 - \boldsymbol{s}_1)$. That is, a boundary point was computed starting from $\boldsymbol{s}_1$ and searching the boundary in the direction of $(\boldsymbol{s}_2 - \boldsymbol{s}_1)$. For an $\mathbf{s}_1$, we performed this search N_s times (each with a different $\mathbf{s}_2$) with $N_s = 1,200$, yielding $\{\eta_n\}_{n=1,2,\ldots,N_s}$. Then, profile $\boldsymbol{s}_1$ and the minimum distance $\|\eta_{n^*}(\boldsymbol{s}_2 - \boldsymbol{s}_1)\|$ were recorded, where $n^* = \arg\min_n \eta_n$. In total, $1,000$ data points were generated in this way.

C. Learning Stability Boundary

Based on the $10K$ points of $(\boldsymbol{s}, \mathbb{I}(\boldsymbol{s} \in \mathcal{S}^*))$ inside the stability region, and the $10K$ points outside, we train a neural network with one hidden layer for a binary classifier. We choose the number of neurons of the hidden layer to be 256, and we use hinge loss at the output layer. l_2 regularization is also used. Among the $10K$ data, $8K$ is used for training, $1K$ for validation, and $1K$ for testing. The trained neural network achieves an testing classification accuracy of 99.9%. As a result, the decision boundary of the trained classifier very accurately approximates the stability region boundary $\mathcal{B}(\mathcal{S}^*)$.

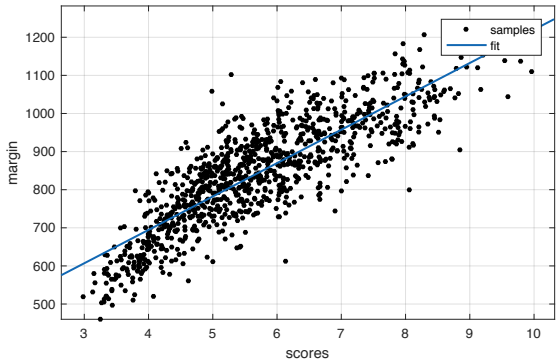

Fig. 2. Stability margin vs score.

D. Predicting Stability Margin

With the trained binary classifier, we use the scores computed at the output layer (cf. Sec. IV-A) to fit a predictor of the voltage stability margin $\text{dist}\,(\boldsymbol{s}, \mathcal{B}(\mathcal{S}^*))$. Using the 1K points of $(\boldsymbol{s}, \text{dist}\,(\boldsymbol{s}, \mathcal{B}(\mathcal{S}^*)))$, we plot a scatter plot of score-margin pairs in Fig. 2. A clear positive correlation is observed. We then fit a simple linear function $\alpha x + \beta$ based on the $1K$ points, and the fitted function is plotted in Fig. 2. The fitted linear function achieves an R-squared measure of 0.71, indicating reasonably good performance.

Finally, we highlight a key advantage of the proposed method — its speed. With the trained predictor $\hat{d}(\boldsymbol{s})$, to compute the approximate voltage stability margin takes *under 1ms* on a laptop computer with 2.8 GHz Intel Core i5 with 8 GB memory. In contrast, computing the margin using the method described in Sec. V-B takes about 240 seconds. Thus, the proposed method achieves a remarkable over 10^5x speed advantage in approximately computing the stability margin.

VI. Conclusions

A computationally efficient learning-based method to estimate the voltage stability margin for an arbitrary operating point was developed without restricting to a particular loading direction. A set of operating points that are inside and outside the feasibility region of AC power flow equations, and yet are close to the stability boundary, were obtained using continuation power flow and SDP power flow methods, respectively. Based on these generated operating points, a neural network classifier with a hinge loss at the output was trained to learn the boundary of the potentially nonconvex voltage stability region. The resulting score metric obtained at the output layer of the neural network before thresholding was shown to be an excellent indicator of the voltage stability margin, and used to train an end-to-end stability margin estimator for an arbitrary operating point. Some 10^5-fold computation speed up was observed using the proposed approach. As a future work, we will explore ways to characterize the vulnerable loading directions as well corresponding to the predicted margin.

REFERENCES

[1] IEEE/PES Power System Stability Subcommittee, "Voltage stability assessment: Concepts, practices and tools," Tech. Rep. PES-TR9, Aug. 2002.

[2] 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 Trans. Power Syst.*, vol. 20, no. 4, pp. 1922–1928, Nov. 2005.

[3] V. Ajjarapu and C. Christy, "The continuation power flow: A tool for steady state voltage stability analysis," *IEEE Trans. Power Syst.*, vol. 7, no. 1, pp. 416–423, Feb. 1992.

[4] C. A. Cañizares and F. L. Alvarado, "Point of collapse and continuation methods for large AC/DC systems," *IEEE Trans. Power Syst.*, vol. 8, no. 1, pp. 1–8, Feb. 1993.

[5] T. Van Cutsem, "A method to compute reactive power margins with respect to voltage collapse," *IEEE Trans. Power Syst.*, vol. 6, no. 2, pp. 145–156, Feb. 1991.

[6] D. Podolsky and K. Turitsyn, "Random load fluctuations and collapse probability of a power system operating near codimension 1 saddle-node bifurcation," in *Proc. of the IEEE Power and Energy Society General Meeting*, Jul. 2013, pp. 1–5.

[7] H.-Y. Su and C.-W. Liu, "Estimating the voltage stability margin using PMU measurements," *IEEE Trans. Power Syst.*, vol. 31, no. 4, pp. 3221–3229, Jul. 2016.

[8] J. W. Simpson-Porco, F. Dorfler, and F. Bullo, "Voltage collapse in complex power grids," *Nat. Commun.*, vol. 7, Feb. 2016.

[9] X. Gu and C. A. Cañizares, "Fast prediction of loadability margins using neural networks to approximate security boundaries of power systems," *IET Gener. Transm. Distrib.*, vol. 1, no. 3, pp. 466–475, May 2007.

[10] D. K. Molzahn, I. A. Hiskens, and B. C. Lesieutre, "Calculation of voltage stability margins and certification of power flow insolvability using second-order cone programming," in *Proc. of the Hawaii Int'l. Conf. Syst. Sci. (HICSS)*, Jan. 2016, pp. 2307–2316.

[11] 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 Trans. Power Syst.*, vol. 26, no. 1, pp. 63–72, Feb. 2011.

[12] D. K. Molzahn, B. C. Lesieutre, and C. L. DeMarco, "A sufficient condition for power flow insolvability with applications to voltage stability margins," *IEEE Trans. Power Syst.*, vol. 28, no. 3, pp. 2592–2601, Aug. 2013.

[13] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. MIT Press, 2012.

[14] Y. Zhao, J. Chen, and H. V. Poor, "A Learning-to-Infer method for real-time power grid topology identification," *preprint*, 2017.

[15] R. D. Zimmerman, C. E. Murillo-Snchez, and R. J. Thomas, "MATPOWER: Steady-state operations, planning and analysis tools for power systems research and education," *IEEE Trans. Power Syst.*, vol. 26, no. 1, pp. 12–19, Feb. 2011.