# Letters

## Uncertainty-Aware Stability Analysis of IBR-Dominated Power System With Neural Networks

Galadrielle Humblot-Renaux, Yang Wu, *Member, IEEE*, Sergio Escalera, *Senior Member, IEEE*, Thomas B. Moeslund, Xiongfei Wang, *Fellow, IEEE*, and Heng Wu, *Senior Member, IEEE*

***Abstract*—Machine learning (ML) technologies have significant potential in accelerating stability screening of modern power systems that are dominated by inverter-based resources. Nonetheless, neural network (NN)-based analysis methods cannot guarantee accurate and reliable stability predictions for unseen operating scenarios (OSs), posing safety risks. To address this limitation, this letter proposes an approach combining NN ensembles with a dual-thresholding framework, which enables the reliable identification of OSs where ML predictions may fail. These uncertain OSs are then flagged for further analysis using physical-based methods, ensuring safety and robustness. The effectiveness of the proposed method is verified by simulation and experimental test.**

***Index Terms*—Inverter-based resources (IBRs), machine learning (ML), stability, uncertainty estimation.**

### I. Introduction

THE decarbonization of global energy system accelerates the deployment of renewable energy resources, which are mostly connected to the power grid via power electronic inverters. Those inverter-based resources (IBRs) may interact with one another and with grid dynamics, leading to power system oscillations or even blackout incidents that are increasingly reported in recent years [1]. Hence, stability studies are of vital importance for transmission system operators (TSOs) to guarantee the secure and reliable operation of IBR-dominated power systems.

The methodology for assessing the stability of IBR-dominated power system under a single operating scenario (OS) is well developed [1]. However, a large power system with thousands of IBRs can have more than 1 billion OSs [2]. Such a high number of OSs makes it impossible for TSOs to carry out in-depth stability studies for each specific OS. Therefore, stability analysis of IBR-dominated power system with extremely high number of OSs has become one of the most challenging tasks for TSOs, as identified in a recent survey from IEEE task force [3].

In recent years, many research efforts have been devoted to leveraging data-driven machine learning (ML) approaches to analyze the stability of the IBR-dominated power system. In particular, neural networks (NNs) have shown promising results for stability analysis [4], [5], [6], [7]. Thanks to their scalability and computational efficiency, NNs can enable the assessment of all OSs within a reasonable timeframe. However, the critical disadvantage of ML-based stability analysis is that it cannot guarantee 100% stability prediction accuracy [8]. Yet, incorrect stability estimation in certain OSs can be safety-critical in practice and even lead to a blackout event.

A potential solution for such a problem is adopting physical-based approaches (e.g., the impedance-based stability analysis and electromagnetic transient (EMT) simulations [1]) to analyze the stability of OSs where ML-based predictions are likely to fail. However, this solution relies on the reliable identification of OSs where ML-based predictions cannot be trusted. This is challenging in the conventional NN-based framework because a single NN cannot capture uncertainty in its parameters (i.e., *epistemic* uncertainty) and thus cannot provide reliable uncertainty estimation beyond the training data [9]. Prior work using a single NN for stability assessment of OSs achieves imperfect classification performance, yet does not consider uncertainty and does not provide a mechanism for identifying NN errors [7].

To tackle this challenge, this letter is the first to apply deep ensembles [10] in the stability analysis of IBR-dominated power system. In contrast to a single NN whose predictions are sensitive to initialization and prone to being confidently incorrect, a deep ensemble aggregates multiple NNs, each with a different set of

Received 23 January 2025; revised 27 February 2025; accepted 11 March 2025. Date of publication 11 April 2025; date of current version 27 August 2025. This work was supported in part by the Danish Data Science Academy, Novo Nordisk Foundation under Grant NNF21SA0069429, in part by VILLUM FONDEN under Grant 40516, in part by the European Union's Horizon 2020 Research and Innovation Programme through the Marie Skłodowska-Curie under Grant 101107634 (PhyDAWN), in part by the Spanish Project under Grant PID2022-136436NB-I00, and in part by ICREA through the ICREA Academia programme. *(Corresponding author: Heng Wu.)*

Galadrielle Humblot-Renaux and Thomas B. Moeslund are with the Visual Analysis and Perception Lab, Aalborg University, 9200 Aalborg, Denmark (e-mail: gegeh@create.aau.dk; tbm@create.aau.dk).

Yang Wu and Heng Wu are with the Department of Energy, Aalborg University, 9220 Aalborg, Denmark (e-mail: yawu@energy.aau.dk; hew@energy.aau.dk).

Sergio Escalera is with the University of Barcelona and the Computer Vision Center, 08193 Barcelona, Spain, and also with the Visual Analysis and Perception Lab, Aalborg University, 9200 Aalborg, Denmark (e-mail: sescalera@ub.edu).

Xiongfei Wang is with the Division of Electric Power and Energy Systems, KTH Royal Institute of Technology, 10044 Stockholm, Sweden, and also with the Department of Energy, Aalborg University, 9220 Aalborg, Denmark (e-mail: xiongfei@kth.se).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TPEL.2025.3560236.

Digital Object Identifier 10.1109/TPEL.2025.3560236

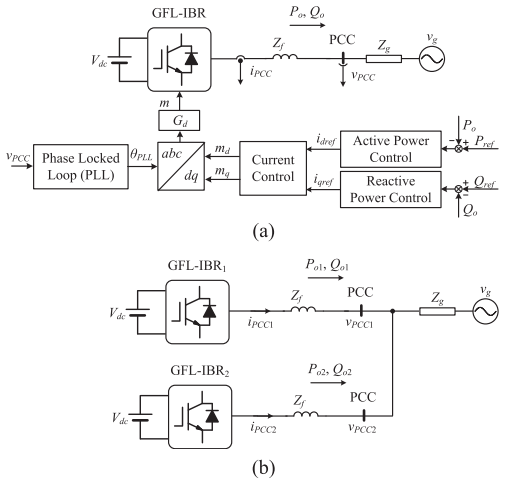



Fig. 1. GFL-IBR connected to the weak ac system. (a) Single IBR. (b) Two paralleled IBR.

parameters. This approach not only provides more robust and stable predictions by averaging the outputs of multiple NNs, but is also a well-established approach for uncertainty estimation in the ML literature [11], [12]. Compared to Bayesian NNs, deep ensembles are simple, scalable, architecture-agnostic, and do not modify the training procedure. Compared to MC-Dropout [13], they offer state-of-the-art performance but at a greater computational cost, due to the need to train multiple NNs.

While previous studies also adopt NN ensembles in other power system applications, e.g., load forecasting [14], [15], their primary aim is to improve predictive performance. They do not incorporate mechanisms to identify OSs where NN-based predictions may fail and take further actions. In contrast, this work proposes an uncertainty-aware framework to systematically identify OSs with unreliable predictions and flag them for further analysis using physics-based methods. Furthermore, different from standard selective classification approaches [16] and recognizing that the false negative (FN) errors (misclassifying a stable OS as unstable) are less critical (as all unstable cases will be reinvestigated in subsequent analysis), the proposed dual-thresholding approach prioritizes the identification of safety-critical false positive (FP) errors (misclassifying an unstable OS as stable, which will be ignored in subsequent analysis, but might ultimately lead to blackout events in practice). This strategy strictly minimizes FP errors without rejecting an impractical high number of OSs. The combination of the NN ensemble (which reliably assigns high uncertainty to incorrect predictions) and the dual-thresholding approach (which rejects uncertain predictions) allows safety-critical errors to be avoided. Finally, simulation and experimental test results verify the effectiveness of the proposed method.

## II. System Description

Fig. 1 illustrates the single-line diagram of the investigated IBR systems, where the single IBR [see Fig. 1(a)] and two paralleled IBRs [see Fig. 1(b)] connecting to the weak ac grid are considered. In both cases, the IBRs are operated with the standard grid-following (GFL) control [1], where the current control loop is used to regulate the output current of IBR to track its current references generated by the active/reactive power control. The phase-locked loop is used for grid synchronization. $Z_f$ and $Z_g$ represent the filter and grid impedance, respectively. $V_{\text{PCC}}$ and $V_g$ represent the voltage at the point of common coupling (PCC) and grid voltage, respectively. The focus of this work is the small-signal stability analysis of IBR-dominated systems, which is affected by different power flows of each IBR, i.e., different combinations of active/reactive power and PCC voltage [2]. Therefore, the OS for a single IBR can be prescribed by a 3-D vector ($P_1$, $Q_1$, $V_{\text{PCC}}$), while the OS for two paralleled IBRs can be prescribed by 5-D vector ($P_1$, $Q_1$, $P_2$, $Q_2$, $V_{\text{PCC}}$).

## III. Approach

### A. Basic NN

First, the basic NN used for stability analysis in previous literature is reviewed [7]. The input of the NN is the OS that is characterized by a $d$-dimensional vector $\mathbf{x} \in \mathbb{R}^d$ ($d = 3$ for a single IBR and $d = 5$ for two parallel IBRs). The nonlinear relationship between OSs and stability is modeled by fully connected layers (two hidden layers with 64 neurons each) and logistic sigmoid activations. The output of the NN is the predicted probability $p \in [0, 1]$ that the OS is stable, which is compared to a binary stability label $y \in \{0, 1\}$ (1 for stable and 0 for unstable). In practice, NN parameters $\theta$ are first initialized randomly. They are then optimized using stochastic gradient descent by minimizing the negative Bernoulli log-likelihood of the training data $\mathcal{D}_{\text{train}}$: $\hat{\theta} = \arg\min_\theta - \sum_{n=1}^{N} [y_n \log(p_n) + (1 - y_n) \log(1 - p_n)]$. It is important to note that while the basic NN's output $p$ is probabilistic, *it is not a reliable measure of uncertainty* [4].

### B. Ensemble of NNs

Instead of training a single NN [7], we propose to leverage deep ensembles [10] as they have shown not only to improve predictive performance compared to a single NN, but also to provide reliable uncertainty estimates. The intuition behind deep ensembles is that combining different viewpoints from a group of experts offers more balanced, nuanced predictions than any single expert could provide.

As illustrated in Fig. 2(b), an ensemble consists of $M$ *distinct* NNs, each with their own set of learned parameters $\theta_m$ for $m = 1, 2, \ldots, M$. To obtain different parameters $\theta_m$ across the ensemble, each NN is initialized with a different random seed before training. The NNs are then trained independently on the full training set $\mathcal{D}_{\text{train}}$. Each NN in the ensemble provides a different plausible solution to the stability learning problem.

During testing, a stability estimate $p$ is obtained by taking the average over the ensemble outputs: $p = \frac{1}{M} \sum_{m=1}^{M} f_{\theta_m}(\mathbf{x})$, where $f_{\theta_m}(\mathbf{x})$ is the output of a single member of the ensemble. The estimate $p$ approaches 0.5 when the disagreement between individual NN increases, or when all NNs' estimates individually approach 0.5. This indicates a high prediction uncertainty that requires further analysis with physical-based

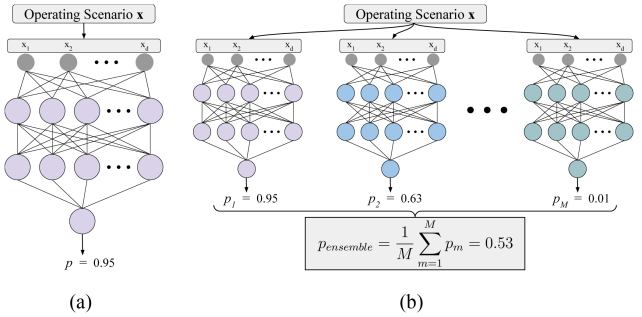



Fig. 2. (a) Default versus (b) proposed NN architectures for stability estimation, illustrated with example outputs (arbitrary values). Each color represents a different set of trained NN parameters. (a) Single NN. (b) NN ensemble (shown with $M = 3$).

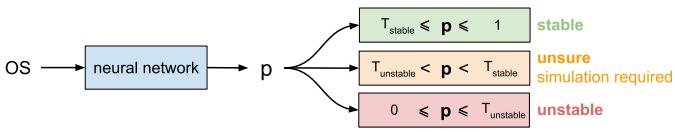



Fig. 3. Flowchart showing how an OS is classified based on the predicted $p$. In the case of low confidence, the model abstains.

methods. In contrast, $p$ approaching 1 (0) indicates a stable (unstable) prediction with a low uncertainty. In practice, ensemble members are likely to agree for OSs in $\mathcal{D}_{\text{train}}$, but may not agree for OSs, which were not seen during training.

### C. Classification and Evaluation

A held-out test set $\mathcal{D}_{\text{test}}$ is used to evaluate the performance of the final trained model. For this, a discrete class (*stable* or *unstable*) must be extracted from the estimate $p$. Typically, a single fixed threshold of 0.5 is used such that $p \geq 0.5$ indicates stable, and $p < 0.5$ indicates unstable [7]. However, this method cannot identify OSs whose stability prediction cannot be trusted. To tackle this challenge, the dual-thresholding approach is proposed in this work, where $T_{\text{unstable}}$ and $T_{\text{stable}}$ are two thresholds that define a range for which the estimated $p$ is not trusted and should be rejected, as shown in Fig. 3. The rejection rate $r$ is the proportion of OSs in $\mathcal{D}_{\text{test}}$ for which $T_{\text{stable}} < p < T_{\text{unstable}}$. Ideally, $r$ should be as small as possible, but without compromising classification performance. Rejected OSs are excluded from evaluation. For the remaining (nonrejected) OSs, classification performance is evaluated by comparing predicted stability versus known small-signal stability in terms of Precision $= \frac{\text{TP}}{\text{TP}+\mathbf{FP}}$ and Recall $= \frac{\text{TP}}{\text{TP}+\mathbf{FN}}$, where TP stands for true positives. Classifying an unstable OS as stable (FP) is a safety-critical issue, and must be strictly avoided. Therefore, the two thresholds will be carefully tuned to maximize the Precision, which will be detailed in the following.

### D. Choice of Classification Thresholds

The thresholds $T_{\text{unstable}} < T_{\text{stable}}$ are tuned on a validation set $\mathcal{D}_{\text{val}}$ based on a desired rejection rate $r_{\text{target}}$ as follows.

1) Find the *highest* possible threshold for which the model achieves 100% recall on $\mathcal{D}_{\text{val}}$. Set this to be $T_{\text{unstable}}$.
2) Find the *lowest* possible threshold for which the model achieves 100% precision on $\mathcal{D}_{\text{val}}$. Set this to be $T_{\text{stable}}$.
3) Check the resulting rejection rate $r_{\text{val}}$, that is, the proportion of $\mathcal{D}_{\text{val}}$ classified with $T_{\text{unstable}} < p < T_{\text{stable}}$. If $r_{\text{val}} < r_{\text{target}}$, increase $T_{\text{stable}}$ until $r_{\text{val}} = r_{\text{target}}\%$. This ensures that at minimum, $r_{\text{target}}$ of validation OSs fall between the two thresholds.

Note that performance on $\mathcal{D}_{\text{val}}$ does not necessarily reflect the performance on $\mathcal{D}_{\text{test}}$. Intuitively, the value for $T_{\text{stable}}$ is thus chosen more conservatively than $T_{\text{unstable}}$, since it is safety-critical to achieve high precision.

### E. Overall Stability Analysis Methodology

As shown in Fig. 3, the proposed NN ensemble and dual-threshold framework utilize the NN ensemble itself to process the stability analysis of *most* OSs with high prediction confidence, while the *remaining small subset* of OSs with low stability prediction confidence from NN is identified for further verification using physics-based stability analysis approaches (such as time-domain simulation). The proposed method leverages the strengths of both ML and physical-based approaches, and thus, improving the computational efficiency without jeopardizing the overall stability prediction accuracy.

### F. Scalability

The proposed framework can be generalized to large-scale IBR-dominated power systems, due to the following reasons.

1) *Data Generation:* The power flow calculation in bulk IBR-dominated power systems is very mature and is widely supported by commercial software, such as DIgSILENT PowerFactory and PSS/E.
2) *NN Ensemble:* The feasibility of training the single NN for stability analysis in bulk IBR-dominated power systems has been extensively investigated [17]. Since an NN ensemble is simply a collection of multiple independent NNs (and allows for bigger or alternative NN architectures to be used if needed), training the ensemble does not introduce additional complexity or constraints beyond what has already been demonstrated for the single NN in bulk power systems. The training cost scales linearly with $M$, and thus, there is a tradeoff between computational efficiency and reliable uncertainty estimation. However, since the NNs are fully independent, they can be trained in parallel, thus greatly reducing the training time. Similarly, after training, the predictions of the individual NNs can be obtained in parallel.

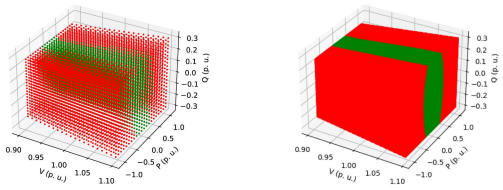
## IV. Case Study

### A. Datasets and Splits

The proposed approach is validated on two datasets corresponding to the single and paralleled IBRs, as summarized in Table I. The datasets consist of OS values and their associated stability label, obtained via small-signal stability analysis in simulation models of grid-connected converters [18]. A grid

TABLE I
DATASET OVERVIEW

| Dataset | Input dimension $d$ | Num. OSs | Num. stable / unstable | Set |
|---|---|---|---|---|
| Single (sparse) | 3 $(V, P, Q)$ | 9261 | 3044 / **6217** | train/val |
| Single (dense) | 3 $(V, P, Q)$ | 3 232 080 | 1 141 933 / **2 090 147** | test |
| Parallel | 5 $(V, P_1, Q_1, P_2, Q_2)$ | 14 406 | **12 471** / 1935 | train/val/test |



(a) (b)

Fig. 4. OSs in the two versions of the single dataset, color-coded by stability (red: unstable and green: stable). (a) Single (sparse). (b) Single (dense).

condition with short-circuit-ratio of 2.4 is applied to testify the method under weak grid conditions.

For single IBR, a *sparse* and a *dense* version of datasets are generated for fine-grained evaluation along the stability boundaries (see Fig. 4). For *Single (sparse)*, $V_{\text{PCC}}$ is set as 0.9: 0.01: 1.1. $P$ is set as $-1$: 0.1: 1, while $Q$ is set as $-0.3$: 0.03: 0.3, all in per unit. These ranges are set to balance the proportion of stable and unstable cases. For *Single (dense)*, the sampling intervals of $V_{\text{PCC}}$, $P$, and $Q$ are shrunk to 0.1 times the original values. The OSs present in *Single (sparse)* are removed from *Single (dense)* to avoid any overlap.

For paralleled IBRs, $V_{\text{PCC}}$ is set as above. $P$ ranges in $[-0.2, -0.06, -0.03, 0, 0.03, 0.06, 0.2]$ and $Q$ ranges in $[-0.1, -0.03, -0.01, 0, 0.01, 0.03, 0.1]$ for each inverter.

Several proportions (20%, 50%, and 80%) of the *single (sparse)* and the *parallel* datasets are selected as training sets via random splitting. For the Parallel case, the remaining portion is further randomly split into 20% validation and 80% test. For the single case, *single (dense)* is used as a test set. This allows for a very fine-grained evaluation in unseen regions along the stability boundary where prediction errors are most likely to occur.

### B. Model Comparison

The *single NN* (as described in Section III-A) is considered as a baseline and is compared to the *Ensemble x10* and *Ensemble x100* (NN ensemble with $M = 10$ and $M = 100$, as described in Section III-B). The same architecture and hyperparameters are used for all models. NN parameters are optimized with Adam [19] for a maximum of 1000 epochs with an initial learning rate of $10^{-1}$ and a batch size of 200. Following standard practice, the input data are standardized based on $\mathcal{D}_{\text{train}}$ statistics. The proposed thresholding approach is applied with $r_{\text{target}} = 20\%$ (a conservative choice) and is compared to the standard approach [7] of having a single threshold at 0.5 and no rejected samples. To ensure that the performance of the approach is consistent, for every dataset and training set size, training/threshold tuning/testing are repeated with ten different random seeds, to account for the stochasticity in parameter initialization, dataset shuffling and random data splits. This results in 30 evaluation runs per dataset. Note that all reported performance is on *unseen* OSs, as there is no overlap between train, val, and test sets.

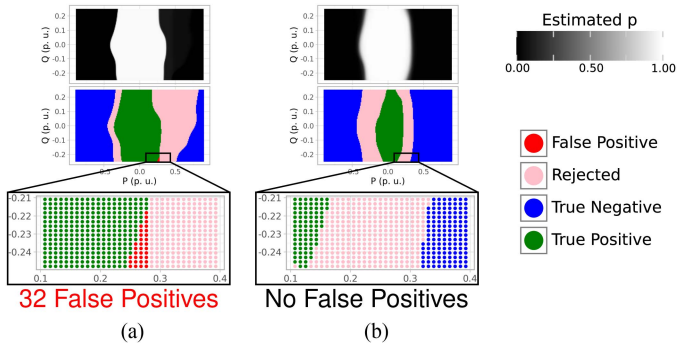



Fig. 5. Estimated $p$ (top) and stability analysis after dual-thresholding (bottom) on unseen OSs from a 2-D slice of the dense single dataset ($V = 99.22$). Note that FPs (red) are safety-critical. (a) Single NN. (b) Ensemble x100.

## V. RESULTS

### A. Quantitative Evaluation

Table II summarizes the quantitative results of the stability prediction based on the proposed dual-thresholding approach (see Table II(b)) and the standard single-thresholding approach adopted in prior work [7] (see Table II(a)), considering both single NN and Ensembles. In the single threshold scenario, although the Ensemble improves the prediction accuracy compared to single NN, all of the models make safety-critical errors on both datasets (the precision is less than 100%). However, it can be observed from Table II(b) that with the proposed dual-thresholding approach, the Ensemble can successfully reject *all unstable OSs whose stability predictions are unreliable* (with a rejection rate around 20%–21%.), thereby guaranteeing 100% precision on the remaining OSs, and thus, the safety-critical prediction errors are avoided. On the other hand, while the precision of single NN is also increased under the dual-thresholding approach, there are still safety-critical errors, especially for the Parallel dataset, as shown in Table II(b). These results highlight the importance of adopting both Ensemble and dual-thresholding approaches to avoid safety-critical prediction errors. Comparing Ensemble $\times 10$ and Ensemble $\times 100$, a higher M ensures 100% precision across all runs and a slightly lower rejection rate for both the Single and Parallel case.

Fig. 5 further provides a visualized example of the stability prediction with the proposed dual-thresholding approach. Compared to the single NN, the Ensemble $\times 100$ exhibits higher and more consistent uncertainty along the decision boundary due to disagreement across ensemble members, with smoother transitions between low and high uncertainty regions. It does not make any FP predictions, thanks to the reliable uncertainty estimation. In contrast, the single NN makes 32 safety-critical FP predictions, which cannot be intercepted. This shows that while the dual-thresholding approach can be applied to a single NN, its effectiveness requires a reliable estimation of $p$, which the single NN does not provide.

TABLE II
MEAN (± STANDARD DEVIATION) CLASSIFICATION PERFORMANCE ON THE TEST SET ACROSS 30 RUNS

| | Single NN | Ensemble ×10 | Ensemble ×100 |
|---|---|---|---|
| *Single dataset* | | | |
| Precision | 95.119% (±17.978%) | 99.111% (±0.385%) | **99.229% (±0.340%)** |
| Recall | 92.590% (±17.602%) | 97.091% (±0.565%) | **97.157% (±0.444%)** |
| Rejection rate | 0% | 0% | 0% |
| *Parallel dataset* | | | |
| Precision | 98.096% (±3.109%) | 99.159% (±0.349%) | **99.195% (±0.317%)** |
| Recall | 99.172% (±0.445%) | 99.588% (±0.212%) | **99.707% (±0.203%)** |
| Rejection rate | 0% | 0% | 0% |

(a) Single threshold of 0.5 (no rejected OSs).

| | Single NN | Ensemble ×10 | Ensemble ×100 |
|---|---|---|---|
| *Single dataset* | | | |
| Precision | 99.962% (±0.143%) | **100% (±0%)** | **100% (±0%)** |
| Recall | 97.945% (±4.957%) | 99.209% (±1.211%) | **99.216% (±1.418%)** |
| Rejection rate | 27.327% (±12.590%) | 21.354% (±0.987%) | **21.209% (±0.799%)** |
| *Parallel dataset* | | | |
| Precision | 96.889% (±16.575%) | 100% (±0.003%) | **100% (±0%)** |
| Recall | 96.516% (±16.231%) | 99.856% (±0.020%) | **99.862% (±0.018%)** |
| Rejection rate | 28.732% (±20.875%) | 20.463% (±4.789%) | **20.107% (±1.385%)** |

(b) Thresholds $T_{unstable}$ and $T_{stable}$ tuned on $\mathcal{D}_{val}$.

Best results are highlighted in bold.

TABLE III
THROUGHPUT (NUMBER OF ESTIMATES PER SECOND)

| | Single IBR ($d = 3$) | Paralleled IBR ($d = 5$) |
|---|---|---|
| Simulation (ground truth) | 1.553 OSs/s | 0.123 OSs/s |
| Ensemble x100 | 1.083 $\times 10^7$ OSs/s | 1.113 $\times 10^7$ OSs/s |
| Combination (20% simulation) | 7.765 OSs/s | 0.615 OSs/s |

TABLE IV
CASES FOR THE EXPERIMENTS

| | Single IBR ($d = 3$) | Paralleled IBR ($d = 5$) |
|---|---|---|
| OS (V, P1, Q1, P2, Q2) | 99.22, 830.47, -744.15 | 121, 1089, 544.5, 0, 544.5 |
| Ensemble x100 | Unsure ($p = 0.324$) | Unsure ($p = 0.783$) |
| Single NN | Stable ($p = 0.978$) | Stable ($p = 0.999$) |

### B. Computational Cost

Using a consumer laptop, the Ensemble × 100 takes 1 min to train on the *Single (sparse)* dataset and 1.5 min on the *Parallel* dataset, with the NNs in the Ensemble being trained sequentially. Parallelization reduces the training time to 16 and 23 s, respectively. After training, the Ensemble × 100 is compared to EMT simulation in PSCAD in terms of computation speed on the same hardware. Results are shown in Table III. It can be seen that Ensemble ×100 is significantly faster for stability analysis ($10^8\times$ faster in the Parallel case). Assuming a rejection rate of 20% by the Ensemble, the stability analysis can be sped up by a factor of 5. As a concrete example, given the paralleled IBR system and a set of 10 000 OSs to assess, stability assessment using EMT simulation takes 81 301 s (approximately 22.6 h). Using our approach (with only around 20% of OSs processed by simulation), the computation time (including the training) reduces to 16 283 s (approximately 4.5 h).

### C. Experiment Validation

Experiments have been conducted to prove the superiority of the proposed method. Two Danfoss converters are controlled in GFL mode, connected to the grid simulator Chroma 61845. Detailed description of the setup can be found in [20].

Two cases from the single and paralleled converters have been selected, with parameters in Table IV. These cases are confidently misjudged as stable by traditional single NN (safety-critical), but rejected due to uncertainty by the proposed ensemble NN. Experiment waveforms are shown in Fig. 6 (the OS is changed by varying the reactive power to switch the system from stable case to unstable case), where unstable resonances can be observed. It should be noted that the studied cases are critically unstable, very close to the stability boundary. Although the oscillations are not very large due to inherent damping (parasitic resistance) in the hardware platform, a prominent increase of system resonance can still be observed in the experiments, supporting the effectiveness of the proposed method.

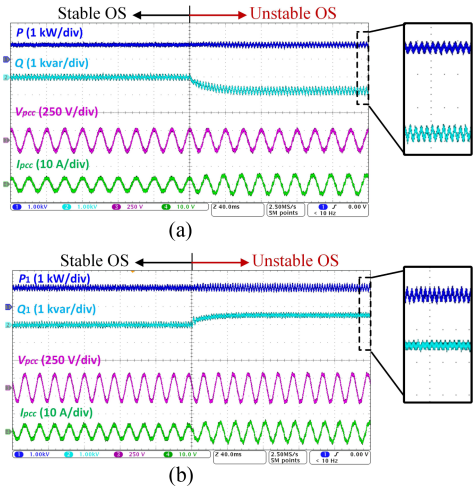



Fig. 6. Experiment waveforms for single and paralleled converters. (a) Reactive power is reduced. (b) Reactive power is increased.

## VI. CONCLUSION

This letter represents the first application of NN ensembles in the stability analysis of IBR-dominated power system. Complemented by the proposed dual-thresholding approach, the framework offers robust and reliable uncertainty estimates for stability predictions. Simulation and experimental tests verify the effectiveness of the proposed method.

## REFERENCES

[1] X. Wang and F. Blaabjerg, "Harmonic stability in power electronic-based power systems: Concept, modeling, and analysis," *IEEE Trans. Smart Grid*, vol. 10, no. 3, pp. 2858–2870, May 2019.
[2] X. Xie, "Analysis and control of oscillatory stability in renewable power systems: China's experience." [Online]. Available: https://resourcecenter.ieee-pels.org/education/webinars/pelsweb041524v

[3] H. Wu, F. Zhao, and X. Wang, "A survey on impedance-based dynamics analysis method for inverter-based resources," *IEEE Power Electron. Mag.*, vol. 10, no. 3, pp. 43–51, Sep. 2023.

[4] S. Chatzivasileiadis, A. Venzke, J. Stiasny, and G. Misyris, "Machine learning in power systems: Is it time to trust it?," *IEEE Power Energy Mag.*, vol. 20, no. 3, pp. 32–41, May/Jun. 2022.

[5] Y. Liao et al., "Neural network design for impedance modeling of power electronic systems based on latent features," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 5, pp. 5968–5980, May 2024.

[6] Y. Li et al., "Machine learning at the grid edge: Data-driven impedance models for model-free inverters," *IEEE Trans. Power Electron.*, vol. 39, no. 8, pp. 10465–10481, Aug. 2024.

[7] M. Zhang and Q. Xu, "Deep neural network-based stability region estimation for grid-converter interaction systems," *IEEE Trans. Ind. Electron.*, vol. 71, no. 10, pp. 12233–12243, Oct. 2024.

[8] B. Li et al., "Trustworthy AI: From principles to practices," *ACM Comput. Surv.*, vol. 55, no. 9, pp. 1–46, 2023.

[9] E. Hüllermeier and W. Waegeman, "Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods," *Mach. Learn.*, vol. 110, pp. 457–506, 2019.

[10] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 6405–6416.

[11] J. Gawlikowski et al., "A survey of uncertainty in deep neural networks," *Artif. Intell. Rev.*, vol. 56, pp. 1513–1589, Jul. 2023, doi: 10.1007/s10462-023-10562-9.

[12] Y. Ovadia et al., "Can you trust your models uncertainty? evaluating predictive uncertainty under dataset shift," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 14003–14014. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2019/file/8558cb408c1d76621371888657d2eb1d-Paper.pdf

[13] Y. Gal and Z. Ghahramani, "Dropout as a Bayesian approximation: Representing model uncertainty in deep learning," in *Proc. 33rd Int. Conf. Mach. Learn.*, New York, New York, USA, 2016, pp. 1050–1059. [Online]. Available: https://proceedings.mlr.press/v48/gal16.html

[14] Z. Cao, C. Wan, Z. Zhang, F. Li, and Y. Song, "Hybrid ensemble deep learning for deterministic and probabilistic low-voltage load forecasting," *IEEE Trans. Power Syst.*, vol. 35, no. 3, pp. 1881–1897, May 2020.

[15] Y. Yang, W. Hong, and S. Li, "Deep ensemble learning based probabilistic load forecasting in smart grids," *Energy*, vol. 189, 2019, Art. no. 116324.

[16] A. Pugnana, L. Perini, J. Davis, and S. Ruggieri, "Deep neural network benchmarks for selective classification," *J. Data-Centric Mach. Learn. Res.*, 2024. [Online]. Available: https://openreview.net/forum?id=xDPzHbtAEs

[17] Y. Du, F. Li, J. Li, and T. Zheng, "Achieving 100x acceleration for N-1 contingency screening with uncertain scenarios using deep convolutional neural network," *IEEE Trans. Power Syst.*, vol. 34, no. 4, pp. 3303–3305, Jul. 2019.

[18] B. Wen, D. Boroyevich, R. Burgos, P. Mattavelli, and Z. Shen, "Analysis of D-Q small-signal impedance of grid-tied inverters," *IEEE Trans. Power Electron.*, vol. 31, no. 1, pp. 675–687, Jan. 2016.

[19] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, 2015.

[20] Y. Wu et al., "Impedance profile prediction for grid-connected VSCs with data-driven feature extraction," *IEEE Trans. Power Electron.*, vol. 40, no. 2, pp. 3043–3061, Feb. 2025.