# Meta-Learning-based Power System Security Assessment under Topology Changes

Koji Yamashita and Nanpeng Yu
*Department of Electrical and Computer Engineering*
*University of California, Riverside*
kyamashi@ucr.edu, nyu@ece.ucr.edu

***Abstract*—The power system topology evolves continuously due to load variations and equipment maintenance, while the growing integration of fluctuating demands and intermittent generation further complicates dynamic security analysis (DSA) for stability analysis. Although data-driven methods show great potential as decision-support tools, their practical deployment is often hindered by the high computational cost of retraining conventional machine learning models for each new network topology. This paper introduces a meta-learning-based framework for frequency stability prediction that efficiently transfers knowledge across topological variations using only a small number of labeled datasets generated from DSA results. The proposed approach is validated on the augmented IEEE 14-bus system, demonstrating both high accuracy and significantly reduced training effort. The results illustrate the promise of meta-learning as an efficient and adaptive tool for the power system security assessment under evolving grid conditions.**

***Index Terms*—Dynamic security, Frequency stability, Grid topology, Meta-learning.**

## I. Introduction

Power system security is traditionally ensured through dynamic security assessment (DSA) that performs near–real-time contingency analysis using time-domain simulations. Although effective, these simulation-based approaches face growing challenges as on-/off-type loads (e.g., data centers) and intermittent renewable resources introduce rapid and substantial power fluctuations that degrade DSA performance. To address these challenges, data-driven approaches for stability prediction have gained increasing attention. Early studies employed conventional machine learning (ML) models such as artificial neural networks [1], decision trees [2], support vector machines [3], and ensemble methods [4], demonstrating feasibility but limited robustness to evolving grid conditions. More recently, deep learning models—such as convolutional neural networks [5], recurrent neural networks [6], and deep belief networks [7]—have achieved improved accuracy and resilience under noisy phasor measurement unit data.

However, adapting both classical and deep learning models to evolving grid topologies remains a challenge. Approaches such as domain adaptation [8], transfer learning [9], and trigger-based retraining [10] have been explored, but each faces limitations: domain adaptation and transfer learning often struggle under large distribution shifts, while retraining methods are hindered by the difficulty of generating representative data and determining optimal update timing. Moreover, transfer learning remains computationally intensive, as substantial retraining is typically required to maintain satisfactory performance across new topologies.

Graph neural networks (GNNs) have also been explored for handling varying power system topologies. For instance, [11] integrates a graph attention network (GAT) with deep reinforcement learning for voltage and var control in dynamically changing grids. Although GNNs can model multiple graph structures, the vast number of possible topologies in practical systems makes exhaustive training infeasible. As a result, their prediction performance often degrades when encountering unseen or rarely observed topologies.

An alternative strategy is meta-learning, which, unlike conventional supervised learning (SL), focuses on learning to learn by producing model parameters optimized for rapid adaptation to new tasks. It enables efficient fine-tuning with only a few gradient updates and limited data, offering superior adaptability for scenarios involving frequent changes, such as evolving grid topologies. Recent studies have explored meta-learning in power systems: [12] employed a physics-informed GAT to adapt state estimation models to changing distribution topologies, and [13] developed a meta-learning-based optimal power flow predictor capable of fast adaptation to topology reconfigurations. These works highlight meta-learning's promise for topology-aware tasks.

However, to the best of the authors' knowledge, meta-learning has not yet been applied to DSA considering system dynamics. This paper proposes a graph meta-learning framework to assess power system frequency stability across different network topologies, loading conditions, and equipment maintenance scenarios by combining the principles of meta-learning with GNNs.

The remainder of this paper is organized as follows. Section II introduces the proposed meta-learning–based DSA framework. Section III details the technical methodology. Section IV presents case studies on the IEEE 14-Bus system to evaluate the performance of the proposed meta-learning model.

## II. Overall Framework

The proposed meta-learning–based DSA framework is designed to augment existing online (near-real-time) contingency analysis systems (see Fig. 1). System demand varies continuously throughout the day, and the grid topology is periodically updated due to equipment maintenance or switching events.

979-8-3315-8138-1

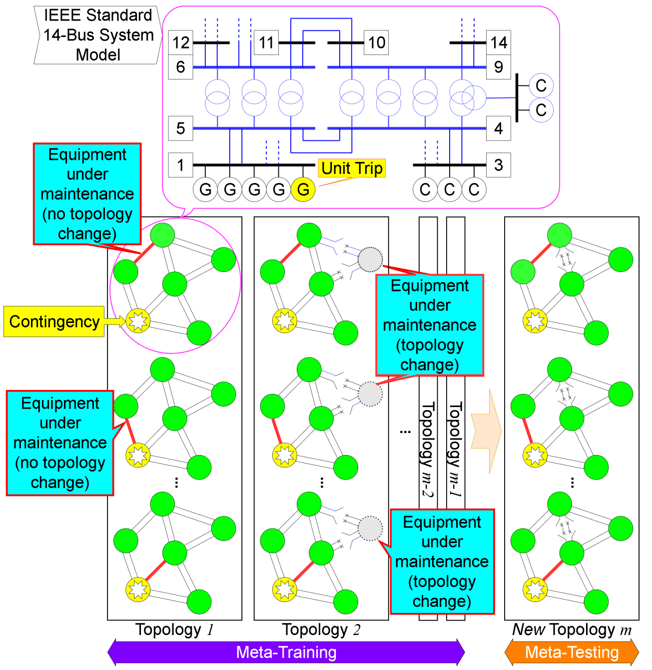


Fig. 1: Task sets for meta-learning–based stability assessment using latest DSA results and topology updates.

Conventional DSA systems respond to these topology changes by refreshing contingency analysis results every few minutes. In contrast, the proposed framework leverages the most recent DSA results and the updated topology as a small set of labeled samples to rapidly adapt the meta-learning model. This efficient adaptation enables timely updates of stability predictions in near real-time, representing a significant improvement over conventional SL methods, which often require tens of minutes or more of data collection and retraining for each new topology.

## III. Technical Method

### A. Formulation of Power System Security Assessment as a Meta-Learning Problem

Power system security assessment must be performed across a variety of network topologies and maintenance scenarios, reflecting frequent changes in the system configurations and operating conditions. To enable the trained machine learning model to generalize rapidly to unseen network topologies and maintenance scenarios with limited data, we adopt a meta-learning approach that follows the "learning to learn" paradigm, known for its strong adaptability and data efficiency.

In our proposed framework, each task corresponds to performing system security assessment under a specific pre-contingency grid topology. Unlike conventional SL, meta-learning introduces the concept of multiple related tasks and formulates the training as a bi-level optimization problem – comprising a task-specific (inner-loop) optimization and a task-general (outer-loop) optimization.

The task-general performance objective is formulated in (1), where model performance is optimized over a distribution of

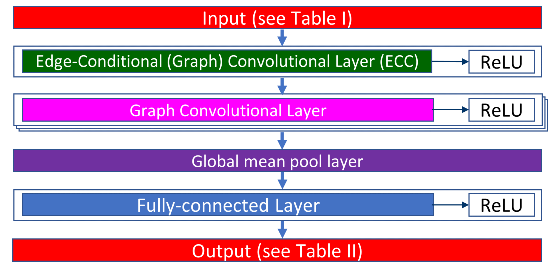


Fig. 2: Graph convolutional network model with attention mechanism in graph convolutional layer.

TABLE I: Inputs of Stability Index Prediction Model

| | Node/Branch Feature | Feature Type |
|---|---|---|
| 1 | Active power output, $P_G$ | Node |
| 2 | Reactive power output, $Q_G$ | Node |
| 3 | Active power load, $P_L$ | Node |
| 4 | Reactive power load, $Q_L$ | Node |
| 5 | Voltage magnitude, $\|V\|$ | Node |
| 6 | Active power output deviation, $\Delta P_G$ | Node |
| 7 | Active power transfer, $P_{tie}$ | Branch |
| 8 | Reactive power transfer, $Q_{tie}$ | Branch |
| 9 | Voltage angle difference, $A_{tie}$ | Branch |
| 10 | Connectivity Status, $SW_{tie}$ | Branch |

tasks $p(\mathcal{T})$. Each task is denoted as $T = \{\mathcal{D}, L\}$, defined by its dataset $\mathcal{D}$ and corresponding loss function $L$:

$$min \mathbb{E}_{\mathcal{T}\sim p(\mathcal{T})} L(\mathcal{D}). \tag{1}$$

The proposed meta-learning framework consists of two stages: meta-training and meta-testing. During the meta-training phase, a set of $S$ source tasks is provided, denoted as $\mathcal{D}_{\text{source}}$, where $\mathcal{D}^{\text{train}}_{\text{source}}$ represents the training subset and $\mathcal{D}^{\text{val}}_{\text{source}}$ represents the validation subset. In the meta-testing phase, a set of $G$ target tasks, denoted as $\mathcal{D}_{\text{target}}$, with $\mathcal{D}^{\text{train}}_{\text{target}}$ serving as the adaptation set and $\mathcal{D}^{\text{test}}_{\text{target}}$ used to assess generalization performance.

### B. Neural Network Architecture

This paper employs a GNN with an integrated attention mechanism (Fig. 2) as the backbone architecture for power system security assessment. The proposed model comprises of one edge-conditioned convolution (ECC) layer, which captures topology-aware relational dependencies, followed by two GAT layers that adaptively assign importance weights to neighboring nodes and edges. The input includes six node-level features and four edge-level features (Table I), while the output represents six discrete frequency drop categories used for frequency stability classification (Table II). Both node and edge embeddings are set to a dimensionality of 64, and two graph convolutional layers are employed to effectively capture higher-order spatial dependencies across the network topology.

### C. Meta-training and Meta-testing Algorithms

*1) Meta-training:* The meta-learning framework operates in two stages: meta-training on a set of source tasks and meta-testing on target tasks. During meta-training, tasks are sampled from the source task distribution, each divided into a training subset and a validation subset. The inner loop performs standard gradient descent updates on the training set

979-8-3315-8138-1

TABLE II: Labels Set for Frequency Stability Index

| Label | Frequency Range (Hz) | Task 1 | Task 2 | Task 3 |
|---|---|---|---|---|
| 1 | $(59.40, \infty]$ | 104 | 91 | 501 |
| 2 | (59.29, 59.40] | 340 | 263 | 1620 |
| 3 | (58.80, 59.29] | 358 | 248 | 1114 |
| 4 | (58.40, 58.80] | 282 | 170 | 899 |
| 5 | (56.97, 58.40] | 386 | 259 | 1399 |
| 6 | [0, 56.97] | 312 | 201 | 1133 |

The values listed under each task represent the number of samples per label.

to obtain adapted task-specific parameters $\theta_i'$, using the inner-loop learning rate $\alpha$:

$$\theta_i' = \theta - \alpha \nabla_\theta L_{\mathcal{T}_i}\left(\theta, \mathcal{D}_{i,\text{source}}^{\text{train}}\right) \tag{2}$$

The outer loop aggregates the validation losses across all tasks and updates the shared model parameters $\theta$ using Adamax—a variant of the Adam optimizer that employs the infinity norm of past gradients for improved numerical stability—with an outer-loop learning rate $\beta$:

$$\theta \leftarrow \text{AdamaxUpdate}\left(\theta, \beta, \sum_i \nabla_\theta L_{\mathcal{T}_i}\left(\theta_i', \mathcal{D}_{i,\text{source}}^{\text{val}}\right)\right). \tag{3}$$

Inspired by the concept of progressively varying the training shot size across stages [14], this paper instead varies the number of validation shots per class proportionally across episodes – while keeping the training shot size fixed – to enhance the robustness of the model initialization (see Algorithm 1).

*2) Meta-testing:* During meta-testing, the trained model is evaluated on previously unseen tasks, as shown in Algorithm 2. For each repetition, a subset of classes is sampled, with $K$ training samples per class used for rapid adaptation in the inner loop, and the remaining samples reserved for evaluation. The inner-loop updates employ a separate learning rate $\alpha'$ to produce task-adapted parameters $\theta''$, initialized from the shared meta-learned parameters $\theta$. Repeating this process with non-overlapping training samples ensures comprehensive utilization of the target dataset and yields statistically robust performance estimates. The final evaluation metrics are reported as the average accuracy and F1-score across all repetitions.

## IV. Case Study

### A. Case Study Setup

In the case study, we evaluate the ability of the proposed and baseline algorithms to predict the frequency stability index. The standard IEEE 14-Bus system [15] is extended to incorporate equipment maintenance and topology variation scenarios. Three distinct pre-contingency grid topologies are considered, as summarized in Table III. Among them, two topologies are used for meta-training, while the remaining one is reserved for meta-testing. Formally, this setup can be expressed as $\mathcal{D}_{\text{source}} = \{\mathcal{T}_i, \mathcal{T}_j\}$, $\mathcal{D}_{\text{target}} = \{\mathcal{T}_k\}$, $(i, j, k) \in \text{Permutation}(\{1, 2, 3\})$, $S = 2$, and $G = 1$.

TABLE III: Definition of Tasks

| | |
|---|---|
| **Task 1:** | Full Grid Topology |
| **Task 2:** | Partial Grid Topology (No Three-winding Transformer) |
| **Task 3:** | Partial Grid Topology (No Branch Between Buses 6 and 11) |

**Algorithm 1:** Meta-training with fixed training shots ($K_{\text{train}}$) per class and variable validation shots ($K_{\text{val}}{}^{\dagger} \times K_{\text{train}}$).

**Input:** Task distribution $p(\mathcal{T}_{\text{source}})$, inner learning rate $\alpha$, outer learning rate $\beta$ (Adamax)
**Output:** Meta-learned GAT model parameterized by $\theta$
Initialize $\theta$ randomly;
Set training shots, ($K_{\text{train}}$), per class;
**while** *not converged* **do**
  Select validation shots per class, ($K_{\text{val}}{}^{\dagger} \times K_{\text{train}}$), cycling through all combinations;
  Sample batch of tasks $\mathcal{T}_i \sim p(\mathcal{T}_{\text{source}})$;
  **for** *each task* $\mathcal{T}_i$ **do**
    Sample $K_{\text{train}}$ shots per class for $\mathcal{D}_{i,\text{source}}^{\text{train}}$;
    Sample $K_{\text{val}}$ shots per class for $\mathcal{D}_{i,\text{source}}^{\text{val}}$;
    $\theta_i' \leftarrow \theta - \alpha \nabla_\theta L_{\mathcal{T}_i}(\theta, \mathcal{D}_{i,\text{source}}^{\text{train}})$;
    Evaluate $L_{\mathcal{T}_i}(\theta_i', \mathcal{D}_{i,\text{source}}^{\text{val}})$;
  **end**
  $\theta \leftarrow \text{AdamaxUpdate}(\theta, \beta, \sum_i \nabla_\theta L_{\mathcal{T}_i}(\theta_i', \mathcal{D}_{i,\text{source}}^{\text{val}}))$;
**end**

† $K_{\text{val}}$ ranges from $K_{\text{min}}^{\text{val}}$ to $K_{\text{max}}^{\text{val}}$.

**Algorithm 2:** Meta-testing with $N$-way $K$-shot adaptation on target tasks.
Metrics are averaged over all sampled test episodes.

**Input:** Meta-learned GAT model $\theta$, target dataset $\mathcal{D}_{\text{target}}$ with class-wise samples, inner learning rate $\alpha'$, number of inner steps $s$, $K$-shot setting
**Output:** Accuracy and F1-score across test episodes
**repeat**
  Sample $N$ classes† from $\mathcal{D}_{\text{target}}$;
  **for** *each class $c$ in sampled $N$ classes* **do**
    Partition data into $K$ train samples $\mathcal{D}_{c,\text{target}}^{\text{train}}$ and remaining test samples $\mathcal{D}_{c,\text{target}}^{\text{test}}$;
  **end**
  Construct task $\mathcal{T}_{\text{target}} = \{\mathcal{D}_{\text{target}}^{\text{train}}, \mathcal{D}_{\text{target}}^{\text{test}}\}$;
  **for** *$s$ steps* **do**
    $\theta'' \leftarrow \theta - \alpha' \nabla_\theta L_{\mathcal{T}_{\text{target}}}(\theta, \mathcal{D}_{\text{target}}^{\text{train}})$;
  **end**
  Evaluate $\theta''$ on $\mathcal{D}_{\text{target}}^{\text{test}}$;
**until** *all classes in $\mathcal{D}_{\text{target}}$ are processed*;
Compute accuracy and F1-score over all repetitions;

† $N$ is the number of classes (e.g., $N = 6$ in Table II).

*1) Enhanced IEEE 14-Bus System Modeling:* The IEEE 14-bus system serves as a simplified representation of an actual power grid and contains fewer components than a real-world network. To more accurately capture operational characteristics and ensure N-1 security, double-circuit lines are assumed for both the 132 kV transmission and 66 kV

979-8-3315-8138-1

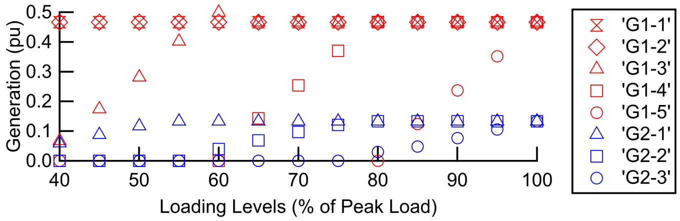


Fig. 3: Generation dispatching scenarios (100 MVA-base).

sub-transmission networks. Three transformers are modeled such that two remain in service during an N-1 contingency. When each transformer operates near its rated capacity, the loss of one unit results in approximately 150% loading on the remaining two – an acceptable level within short-term emergency ratings. This modeling choice prevents the unrealistic scenario of complete transformer loss under N-1 conditions and maintains grid security, provided that system operators can subsequently restore normal operations. To represent realistic generation dispatch consistent with grid conditions from the 1960s (when the IEEE 14-bus test system was developed), generation units are disaggregated into multiple smaller synchronous machines. Specifically, Bus 1 includes five 60 MVA thermal units, while Bus 2 contains three 20 MVA thermal units. This configuration reflects typical unit sizes of that era and enables more representative N-1 analysis, as the loss of a single generator results in only a partial reduction in bus-level generation rather than a complete outage.

*2) Generator Dispatch:* To capture a wide range of operating conditions, loading scenarios are generated from 35% to 100% of peak demand in 5% increments (see Fig 3). When scaling generation to match reduced loads, only one unit at each power station adjusts its output, while the remaining units operate at constant output, as shown in Fig 3.

*3) Maintenance Scenarios:* For each system topology and loading condition, three categories of power equipment maintenance scenarios are considers: (1) seven transmission lines, (2) eight sub-transmission lines, and (3) three transformer groups. All single-, double-, and triple-equipment maintenance combinations are exhaustively generated. Scenarios involving the simultaneous maintenance of both circuits of a double-circuit line or two transformers at the same substation are excluded, as these would completely disable a transmission corridor or substation transformation capacity. Given the large number of possible combinations, any scenario that results in equipment overload under normal operating conditions is automatically filtered out without applying corrective actions. Maintenance scenarios involving more than three pieces of equipment are also omitted, as they typically lead to overloaded conditions in one or more network elements. The dataset sizes corresponding to the three grid topologies are summarized in Table IV.

*4) Contingencies:* The IEEE 14-bus test system includes only two synchronous generators. Since a generator trip constitutes a primary disturbance that triggers a system frequency drop, only two distinct unit-trip scenarios exist for each combination of system topology, load condition, and maintenance configuration.

*5) Power System Dynamic Model:* All key dynamics relevant to frequency stability are incorporated into the simulation model. The inertia constants are set to 3 s for synchronous generators and 0.75 s for synchronous condensers, representing the presence or absence of turbine mass, respectively. A static load model with frequency sensitivity of 2% per 1% Hz is used to indirectly account for the aggregate effect of the induction motors. Generator controls include an IEEE standard steam turbine–governor (4% droop) and a static exciter with an overexcitation limiter, while power system stabilizers are intentionally omitted. Frequency protection schemes are implemented using a three-step underfrequency load-shedding mechanism, as well as over- and underfrequency relays for generators and condensers. All model parameters follow the specifications provided in [16].

*6) Setup of Meta-Learning and Supervised Learning Model:* Table V summarizes the parameters used in the meta-learning model. For each case study, the model was trained on two of the three tasks and evaluated on the remaining held-out task. All three possible training–testing splits were conducted to comprehensively assess the generalizability of the proposed framework. The baseline algorithm is a conventional supervised learning (SL) model trained solely on samples from the source tasks. To ensure a fair comparison, SL models with domain adaptation were also implemented, where 6 (1 sample per class) and 30 (5 samples per class) labeled samples from the target task were incorporated into the training process.

### *B. Case Study Results*

*1) Meta-Learning Performance Across Cases:* The case study results are presented in Table VI. Across all three training–testing splits, the proposed meta-learning approach consistently outperforms both supervised learning (SL) and SL with domain adaptation, achieving higher accuracy and F1-scores. The improvement is particularly pronounced in the F1-score — a more reliable metric in this context due to substantial differences in sample sizes between tasks (see Table IV). On average, meta-learning with 30 adaptation samples achieves 10.1% and 3.0% higher F1-scores than baseline SL and SL with 30-sample domain adaptation, respectively. Case

TABLE IV: Number of Maintenance Scenarios

| Number of Equipment Maintenance | Topology 1 (Task 1) | Topology 2 (Task 2) | Topology 3 (Task 3) |
|---|---|---|---|
| 1 | 218 | 14 | — |
| 2 | 1,404 | 192 | 261 |
| 3 | 5,026 | 1,026 | 1,521 |
| Total | 6,666 | 1,232 | 1,782 |

Numbers under each topology indicate the number of scenarios for the corresponding maintenance count.

TABLE V: Hyperparameters of Meta-training Model

| Parameter | Inner Loop | Outer Loop |
|---|---|---|
| Learning Rate | 5e-4 | 5e-4 |
| Number of Inner Steps | 3 | N/A |
| K-shot Size | 3 | 9 or 12 |
| Optimizer | Standard Gradient Descent | Adamax |

979-8-3315-8138-1

TABLE VI: Testing Accuracy and F1-Score for Three Training/Testing Splits

| Method | Metric | Case 1 (Tasks 1&2 / Task 3)[b] | Case 2 (Tasks 1&3 / Task 2)[b] | Case 3 (Tasks 2&3 / Task 1)[b] |
|---|---|---|---|---|
| SL on Source Tasks[a] | Acc./F1-score | 0.740/0.751 | 0.726/0.714 | 0.836/0.871 |
| SL + (6-way 1-shot) Adaptation | Acc./F1-score | 0.785±0.027/0.801±0.029 | 0.757±0.052/0.751±0.056 | 0.867±0.026/0.876±0.028 |
| SL + (6-way 5-shot) Adaptation | Acc./F1-score | 0.827±0.031/0.842±0.030 | 0.816±0.036/0.816±0.038 | 0.884±0.017/0.893±0.018 |
| Meta-Learning (6-way 1-shot) | Acc./F1-score | 0.829±0.064/0.839±0.080 | 0.824±0.014/0.826±0.013 | 0.883±0.044/0.892±0.053 |
| Meta-Learning (6-way 5-shot)[c] | Acc./F1-score | 0.883±0.023/0.898±0.020 | 0.831±0.0032/0.832±0.0026 | 0.900±0.013/0.910±0.014 |

[a] Trained on source dataset and tested on target dataset.
[b] Notation "Tasks $i\&j$ / Task $k$" indicates training on Tasks $i, j$ and testing on Task $k$.
[c] Inner learning rate: 9e-4 (Case 1), 1e-4 (Case 2), 1e-4 (Case 3); Number of inner steps: 8 (Case 1), 15 (Case 2), and 8 (Case 3), respectively.

1 demonstrates the most significant gain, where meta-learning surpasses SL with domain adaptation by 5.6% in both accuracy and F1-score. These results underscore meta-learning's strong generalization capability across tasks with imbalanced data availability, maintaining high predictive performance even when training and testing distributions differ significantly. By enabling rapid adaptation, meta-learning minimizes the need for extensive retraining when new task data becomes available, offering a scalable solution for real-world deployment under evolving grid conditions. Furthermore, strong 1-shot performance highlights effective initialization, while additional gains in 5-shot testing demonstrate robust adaptability — emphasizing the dual strengths of the model in both quick generalization and efficient fine-tuning.

*2) Impact of Task Complexity:* Task complexity clearly affects model performance across cases. Task 1 represents the full topology, Task 2 excludes one branch and one node, and Task 3 excludes only one branch—making Task 2 the most challenging. This pattern aligns with the results: Case 2 consistently yields the lowest accuracy and F1-score across all methods, including SL, which lacks access to test data during training and thus exposes the inherent adaptation difficulty most clearly. Although SL with domain adaptation benefits from limited labeled target data and achieves moderate gains, the meta-learning model exhibits superior generalization. Without retraining in test data, it delivers the best overall performance, demonstrating effective knowledge transfer across tasks of varying complexity. Notably, meta-learning shows complementary strengths—strong initialization in Case 2 supports rapid adaptation to difficult tasks, while effective fine-tuning in Case 1 yields additional gains with more labeled samples.

## V. Conclusion

This paper presents a meta-learning model for predicting frequency stability using time-domain simulation–based contingency analysis results under diverse system topologies and power equipment maintenance scenarios. Different grid topologies were defined as individual tasks within the meta-learning framework, and three representative topologies were used to evaluate prediction performance under maintenance-induced configurations following unit trip events. Case studies show that the proposed meta-learning model consistently outperforms both baseline approaches across all training/testing splits, achieving on average 3–4% higher F1-scores than supervised learning with domain adaptation and 7–14% higher than conventional supervised learning. These results highlight meta-learning's strong generalization across tasks with uneven data distributions, reducing the need for extensive retraining while preserving high predictive accuracy.

## References


[1] F. Bellizio, A.-A. B. Bugaje, J. L. Cremer, and G. Strbac, "Verifying machine learning conclusions for securing low inertia systems," *Sustainable Energy, Grids and Networks*, vol. 30, p. 100656, 2022.

[2] T. Zhang, M. Sun, J. L. Cremer, N. Zhang, G. Strbac, and C. Kang, "A confidence-aware machine learning framework for dynamic security assessment," *IEEE Trans. Power Syst.*, vol. 36, no. 5, pp. 3907–3920, 2021.

[3] S. Liu, L. Liu, Y. Fan, L. Zhang, Y. Huang, T. Zhang, J. Cheng, L. Wang, M. Zhang, R. Shi, and D. Mao, "An integrated scheme for online dynamic security assessment based on partial mutual information and iterated random forest," *IEEE Trans. Smart Grid*, vol. 11, no. 4, pp. 3606–3619, 2020.

[4] Y. Zhang, Y. Xu, S. Bu, Z. Y. Dong, and R. Zhang, "Online power system dynamic security assessment with incomplete PMU measurements: a robust white-box model," *IET Gener. Transm. Dis.*, vol. 5, p. 662–668, 2019.

[5] M. Pournabi, M. Mohammadi, S. Afrasiabi, and P. Setoodeh, "Power system transient security assessment based on deep learning considering partial observability," *Electr. Power Syst. Res.*, vol. 205, p. 107736, 2022.

[6] L. Zheng, W. Hu, K. Hou, X. Xu, and G. Shao, "Real-time transient stability assessment based on deep recurrent neural network," in *2017 IEEE Innovative Smart Grid Technologies - Asia*, 2017, pp. 1–5.

[7] S. Wu, L. Zheng, W. Hu, R. Yu, and B. Liu, "Improved deep belief network and model interpretation method for power system transient stability assessment," *J. Mod. Power Syst. Clean Energy*, vol. 8, no. 1, pp. 27–37, 2020.

[8] S. Azad and M. T. Ameli, "A domain adaptation-based convolutional neural network incorporating data augmentation for power system dynamic security assessment," *J. Eng.*, vol. 2024, no. 7, pp. 1–17, 2024.

[9] C. Ren and Y. Xu, "Transfer learning-based power system online dynamic security assessment: Using one model to assess many unlearned faults," *IEEE Trans. Power Syst.*, vol. 35, no. 1, pp. 821–824, 2020.

[10] F. Bellizio, J. L. Cremer, and G. Strbac, "Machine-learned security assessment for changing system topologies," *Int. J. Elect. Power Energy Syst*, vol. 134, p. 107380, 2022.

[11] X. Liu, P. Zhang, H. Xie, X. Lu, X. Wu, and Z. Liu, "Graph attention network based deep reinforcement learning for voltage/var control of topologically variable power system," *J. Mod. Power Syst. Clean Energy*, vol. 13, no. 1, pp. 215 – 227, 2025.

[12] H. Wu, Z. Xu, M. Wang, and X. Lyu, "Meta-learning enhanced physics-informed graph attention convolutional network for distribution power system state estimation," *IEEE Trans. Netw. Sci. Eng.*, vol. 12, no. 2, pp. 1186–1197, 2025.

[13] Y. Chen, S. Lakshminarayana, C. Maple, and H. V. Poor, "A meta-learning approach to the optimal power flow problem under topology reconfigurations," *IEEE Open Access J. Power Energy*, vol. 9, pp. 190–120, 2022.

[14] E. Stergiadis, P. Agrawal, and O. Squire, "Curriculum meta-learning for few-shot classification," in *5th Workshop on Meta-Learning at NeurIPS 2021*, Dec. 2021.

[15] Power Systems Test Case Archive. (Aug., 1993) 14 bus power flow test case. University of Washington. [Online]. Available: https://labs.ece.uw.edu/pstca/pf14/pg_tca14bus.htm

[16] ERCOT, "ERCOT nodal operating guides – section 2: System operations and control requirements," May 2019.


979-8-3315-8138-1