Simplicial Graph-Based Detection and Localization of Cyber Attacks Against Large-Scale Smart Grids

Salma Aboelmagd, *Graduate Student Member, IEEE*, Rachad Atat, *Senior Member, IEEE*, and Abdulrahman Takiddin, *Member, IEEE*

***Abstract*—Ensuring the resilience of large-scale power systems against cyber attacks is critical to maintaining the stability of modern cyber-physical energy systems. Existing machine learning (ML)-based detection frameworks predominantly focus on pairwise node interactions (i.e., edges) and often overlook capturing higher-order topological structures (i.e., simplicial complexes) that emerge within power networks. This paper introduces a novel simplicial recurrent graph neural network (SRGNN) for attack detection and localization in large-scale smart grid infrastructure. Unlike conventional graph models, SRGNN incorporates higher-order simplicial interactions to capture group-wise dependencies among buses and transmission lines, allowing the model to better reflect the multi-scale dynamics of grid operation and control. We investigate the robustness of our model against benchmark and proposed complex simplicial-based attack node selection strategies. Our extensive experiments on large-scale transmission power networks (2,869-bus, 9,241-bus, and 70,000-bus systems) demonstrate that the proposed SRGNN model outperforms ML-based benchmark models, achieving superior detection and localization performance by 9–39% and 8–35%, respectively, in detection rate against complex attacks. These results underscore the importance of modeling higher-order topological structures for robust and scalable security in power systems.**

***Index Terms*—Attack detection, attack localization, smart grids, simplicial complexes, simplicial graph neural networks.**

I. Introduction

Modern smart grids are large-scale cyber-physical systems that integrate components of sensing, control, communication, and energy delivery [1], and rely on the continuous exchange of real-time measurements, such as voltage, current, and power flows, across distributed sensors, control centers, and advanced metering infrastructures [2]. This data exchange is critical for grid monitoring, demand response, and protection mechanisms. However, the integrity of these measurements can be compromised by cyber attacks, where attackers manipulate sensor readings to mislead control decisions, destabilize estimators, or cause service disruptions [3]. Recent reports, such as the 2023 discovery that hackers are remotely targeting the Texas power grid and other critical infrastructure, underscore the increasing sophistication of cyber threats and their potential to disrupt essential services on a national scale [4]. The increasing complexity of smart grids and the rising risk of cascading failures call for robust and scalable cyber attack detection and localization frameworks capable of securing the grid against a wide range of attack strategies.

Salma Aboelmagd and Abdulrahman Takiddin are with the Department of Electrical and Computer Engineering, FAMU-FSU College of Engineering, Florida State University, Tallahassee, FL 32310, USA (e-mail: {saboelmagd, a.takiddin}@fsu.edu).

Rachad Atat is with the Department of Computer Science & Mathematics, Lebanese American University, Beirut 1102-2801, Lebanon (email: rachad.atat@lau.edu.lb).

A. Related Works and Limitations

Data-driven attack detection and localization frameworks in power systems fall into two categories: (1) non-graph approaches based on machine learning (ML) models employing shallow or deep learning (DL) structures, and (2) graph-based approaches leveraging graph signal processing (GSP) or graph neural networks (GNNs), as reviewed in the following sections.

1) Non-Graph Approaches: Data-driven detection frameworks for cyber attacks in power grids have been extensively explored using shallow ML and deep learning (DL) models that treat the grid as a set of independent signals. Shallow ML models, such as support vector machines (SVMs), reported F1-Scores of 82% and 84% [5], [6]. A decision tree (DT) classification approach reported a detection rate (DR) of 91% [6]. A random forest (RF) model achieved overall accuracy levels of 95% [7]. DL models have been applied to learn temporal signal patterns in power system measurements. Feedforward neural network (FNN) frameworks achieved accuracy rates of 90% [8] and 99% [9]. An autoencoder achieved DRs of 96% [10]. A recurrent neural network (RNN) reported a DR of 96% [11]. Convolutional neural network (CNN) frameworks reported accuracy scores of 93% [12] and 99% [13]. A deep autoencoder achieved a detection accuracy of 94% [14]. Despite these improvements, deep learning models primarily focus on temporal patterns without explicitly modeling grid topology, which limits their effectiveness in topology-aware attack detection and scalability in large-scale systems.

2) Graph-Based Approaches: Graph-based detectors take advantage of the topological structure of the power grid to model spatial dependencies between buses. Approaches using GSP techniques applied spectral filters on the grid graph to detect cyber attacks and achieved DRs of up to 90% [15]. A GNN detector achieved DRs of 83%–96% [16]. A GNN model using auto-regressive moving average graph filters demonstrated a 97% F1-score [17]. A GNN framework reported a DR of 97% [18]. A GNN multi-task learning framework demonstrated accuracy of 99% in detection and localization [19]. A generalized graph autoencoder (GAE) model, a robust GAE with convolutional recurrent layers, and spatiotemporal GAEs reported DRs of 93%–99% across

unseen cyber attacks and dynamic topologies [20], [21]. A recurrent attention GNN (RAGNN) achieves 97% in DR across cyber attacks [1]. A spectral GNN achieved a DR of 99% [17]. A Transformer GNN model achieved a DR of 98% [22] Graphon neural networks (WNNs) further extended scalability by leveraging learning by transference across graphs and reported a 98% DR [23]. A GNN framework reported a DR of 97% for detection and attack localization [18].

3) Limitations: Existing frameworks exhibit key limitations. ML and DL frameworks that do not incorporate graph structures are vulnerable to attacks exploiting topological properties, and their performance deteriorates when applied to large-scale system configurations, as will be seen in V. While graph-based frameworks capture spatial dependencies, they require handcrafted filters and are limited to small-scale networks. Despite promising detection performance, state-of-the-art GNN-based frameworks present at least one of the following limitations. First, they solely model pairwise interactions, without capturing higher-order structures such as simplicial complexes, which naturally exist in large-scale transmission systems and are fundamental to understanding complex multi-node couplings in power systems. Second, existing detectors are validated on small-scale power systems ranging from 14- to 2,000-bus systems, with limited evidence of scalability to real-world large-scale power networks where higher-order structures exist. Third, their performance is validated under simple attack node selection strategies (e.g., random or centrality-based), without considering the effect of attacking nodes in higher-order structures. Fourth, existing models are limited to either detection (classifying the overall system status) or localization (pinpointing which node is attacked) separately. Such limitations highlight the need for a robust framework that can generalize across diverse topologies, capture higher-order interactions, and scale to large dynamic systems to perform attack detection and localization.

B. Contributions

This paper presents a novel simplicial recurrent graph neural network (SRGNN) framework for the detection and localization of data-driven attacks in large-scale power systems. Unlike existing detectors that model only pairwise node interactions, our proposed SRGNN framework explicitly captures higher-order topological structures, such as 2-simplices and 3-simplices, that naturally emerge in power grid graph structures. The contributions of this work span three complementary dimensions: (i) we investigate how incorporating simplicial complexes into the graph learning paradigm enhances detection and localization performance, (ii) we propose simplicial-based attack strategies that deliberately target nodes within simplicial complexes to reveal structural vulnerabilities, and (iii) we demonstrate the scalability and practical relevance of our SRGNN framework on large-scale, realistic power system benchmarks. Our contributions are summarized as follows.

- We construct comprehensive spatiotemporal datasets for large-scale power systems modeled after realistic grid configurations, including the 2,869-bus, 9,241-bus, and 70,000-bus networks, allowing for scalability evaluation. Our proposed SRGNN improves detection and localization performance by up to 4.8% and 4.3%, respectively, as system size increases from 2,869 to 70,000 buses.
- We propose novel topology-informed attack node-selection strategies that target buses embedded in higher-order simplicial complexes (2- and 2+3-simplices) and compare them to random and centrality-based selections. The proposed attack strategies degrade the detection performance of benchmark models by 3–9% compared to traditional random selection, and by 4–6% compared to centrality-based selection. This demonstrates that higher-order node targeting poses a significantly stronger threat, enabling a more comprehensive evaluation of detection models under perturbation, replay, and denial-of-service (DoS) attacks.
- We develop a novel SRGNN architecture for the detection and localization of cyber attacks. To isolate temporal and higher-order effects, we also introduce two simplified variants: a simplicial graph neural network (SGNN) without recurrence, and a purely simplicial neural network (SNN) without a graph convolutional network (GCN) backbone. SRGNN outperforms the proposed variants by 10–12% in detection and 10–11% in localization rate, confirming the value of integrating both temporal and higher-order structure.
- We demonstrate that SRGNN substantially outperforms benchmark models across a wide range of grid sizes and attack scenarios. Specifically, our model improves detection and localization performance by 9–39% and 8–35%, respectively, in detection rate, compared to ML-based benchmarks. These gains validate the advantages of incorporating both temporal memory and higher-order topological reasoning into the graph learning process.

The remainder of the paper is organized as follows. Section II introduces the concept of simplicial complexes and their role in modeling higher-order topological structures within power grids. Section III presents our proposed SRGNN architecture for attack detection and localization, detailing its architecture and the integration of simplicial convolutions, along with a comparison with two proposed simplicial-based model variants, namely, SGNN and SNN. Section IV describes the experimental setup as well as the dataset generation process, threat model, and benchmark detectors. Section V reports and analyzes the attack detection and localization performance across various attack scenarios and system sizes, comparing the SRGNN with benchmark models. Finally, Section VI concludes the paper and outlines future research directions.

II. Preliminaries

In this section, we discuss the key mathematical foundations underlying our SRGNN. We first define simplicial complexes as extensions of graphs that capture group interactions via k-simplices, and then introduce the algebraic operators—boundary maps, incidence matrices, and combinatorial Laplacians—that enable us to lift and diffuse signals across different simplex orders. To ground these concepts, we present a 2-simplex example illustrating how lower- and upper-adjacency combine in the 1-Laplacian. These preliminaries

form the algebraic backbone for the simplicial convolutional and message-passing layers developed in the following section.

A. Graph Representation

A graph $\mathrm{G} = (V, E)$ consists of a set of nodes V and a set of edges $E \subseteq V \times V$ that capture pairwise interactions between nodes. Each edge indicates a direct relationship between two nodes, and the entire structure can be represented by an adjacency matrix $A \in \mathbb{R}^{|V|\times|V|}$, where $A_{ij} = 1$ if an edge exists between nodes i and j, and $A_{ij} = 0$ otherwise. Graphs provide a flexible representation for modeling systems with binary relations, where information or influence flows along connections between entities. However, traditional graphs are limited to pairwise connections and cannot express group-wise interactions prevalent in real-world grids.

B. Simplicial Complex

A simplicial complex $\mathcal{K}$ generalizes a graph by encoding not only pairwise relations (edges) but also group interactions among any number of nodes. A 0-simplex is a vertex v (Fig. 1(a)), a 1-simplex is an edge e (Fig. 1(b)), a 2-simplex is a filled triangle ς_{012} (Fig.1(c)), a 3-simplex is a filled tetrahedron ς_{0123} (Fig. 1(d)), and so on, representing different dimensions in the complex [24]. Formally, a k-simplex is a set of $k+1$ vertices. We denote a k-simplex by σ, which serves as a building block of the simplicial complex, defined as follows:

$$\sigma = \{v_0, v_1, \ldots, v_k\}.$$

The complex $\mathcal{K}$ is a collection of such simplices closed under taking faces τ (i.e., a filled area): if $\sigma \in \mathcal{K}$ then every nonempty subset $\tau \subseteq \sigma$ also lies in $\mathcal{K}$. We denote by $\mathcal{K}_k$ the set of all k-simplices. To connect different dimensions, we introduce the boundary operator

$$\begin{aligned} \partial_k &: C_k(\mathcal{K}) \to C_{k-1}(\mathcal{K}), \\ \partial_k[v_0, \ldots, v_k] &= \sum_{i=0}^{k} (-1)^i [v_0, \ldots, \widehat{v_i}, \ldots, v_k]. \end{aligned} \tag{1}$$

where $C_k(\mathcal{K})$ is the real vector space spanned by oriented k-simplices and $\widehat{v_i}$ indicates omission of v_i. The key property

$$\partial_{k-1}\, \partial_k = 0 \tag{2}$$

captures the fact that the boundary of a boundary is empty [25].

In coordinates, let $R_k \in \mathbb{R}^{|\mathcal{K}_{k-1}|\times|\mathcal{K}_k|}$ be the incidence matrix whose (i, j) entry is $+1$ or -1 if the $(k-1)$-simplex i is a face of the k-simplex j, with orientation, and 0 otherwise. The relation (1) becomes

$$R_{k-1}\, R_k = 0. \tag{3}$$

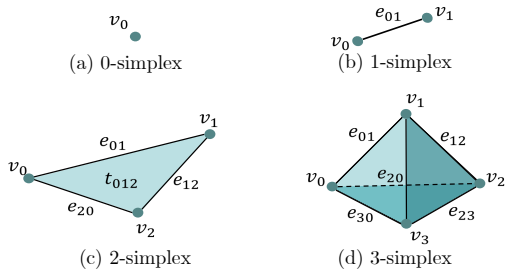
We define the k-th combinatorial Laplacian as

$$\mathcal{L}_k = R_k^\top R_k \;+\; R_{k+1} R_{k+1}^\top. \tag{4}$$

The first term, $R_k^\top R_k$, models lower adjacency, where two k-simplices are connected if they share a common $(k-1)$-face. The second term, $R_{k+1} R_{k+1}^\top$, models upper adjacency, where two k-simplices are connected if they both belong to the same $(k+1)$-simplex [25], [26]. Together, these two terms in the Laplacian capture how information flows both across and within simplex hierarchies. When $k = 0$ (vertices), the Laplacian becomes $\mathcal{L}_0 = R_1 R_1^\top$, which is the standard graph Laplacian on vertices. For $k = 1$ (edges),

$$\mathcal{L}_1 = R_1^\top R_1 + R_2 R_2^\top \tag{5}$$

describes edge-level relationships, where edges interact through shared vertices and shared triangles.



(a) 0-simplex (b) 1-simplex (c) 2-simplex (d) 3-simplex

Fig. 1. Examples of simplices.

Fig. 1(c) shows a 2-simplex with vertices $\{v_0, v_1, v_2\}$, edges

$$e_{01} = (v_0 \to v_1), \quad e_{12} = (v_1 \to v_2), \quad e_{20} = (v_2 \to v_0),$$

and a single triangle ς_{012}. To define incidence matrices, we assign a consistent orientation: clockwise in this case. The vertex-to-edge incidence matrices are

$$R_1 = \begin{bmatrix} +1 & 0 & -1 \\ -1 & +1 & 0 \\ 0 & -1 & +1 \end{bmatrix}, \quad R_2 = \begin{bmatrix} +1 \\ +1 \\ +1 \end{bmatrix},$$

where $R_1 \in \mathbb{R}^{3\times3}$ maps vertices (v_0, v_1, v_2) to edges (e_{01}, e_{12}, e_{20}), and $R_2 \in \mathbb{R}^{3\times1}$ maps edges to the triangle ς_{012}. One can verify that

$$R_1 R_2 = 0,$$

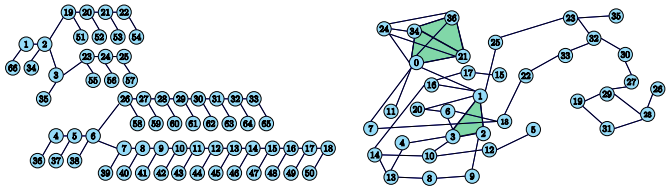
confirming that the boundary of a boundary is zero, as expected. The 1-Laplacian on edges becomes

$$\begin{aligned} \mathcal{L}_1 &= R_1^\top R_1 + R_2 R_2^\top \\ &= \underbrace{\begin{smallmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{smallmatrix}}_{R_1^\top R_1} + \underbrace{\begin{smallmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{smallmatrix}}_{R_2 R_2^\top} \\ &= \begin{smallmatrix} 3 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 3 \end{smallmatrix}. \end{aligned} \tag{6}$$

This example shows how $\mathcal{L}_1$ reflects both lower adjacency (edges sharing a vertex) and upper adjacency (edges forming a triangle), even in the simplest non-trivial simplicial complex.

C. Large-Scale Transmission Systems Higher-Order Topology

In traditional graph-based models of power systems, buses are represented as nodes and transmission lines as edges connecting these nodes. Although this representation captures pairwise interactions, it overlooks more complex group dependencies that naturally exist in large-scale transmission systems. For instance, several buses may be jointly regulated by the same substation, or multiple lines may form a tightly



(a) Illustrative example of how simplices are not present in distribution systems (IEEE 33 bus system).

(b) Illustrative example of how simplices are present in transmission systems (Hawaii-40 bus system).

Fig. 2. Illustration of the structural difference between distribution and transmission networks.

coupled region where electrical and control behaviors are highly interdependent. Multi-node interactions can be naturally modeled as higher-order structures such as triangles and tetrahedra. For example, three mutually connected buses can be represented not just as three edges, but as a filled triangle—a 2-simplex—capturing triadic coupling. Similarly, a fully connected group of four buses can form a 3-simplex (tetrahedron), encoding shared dynamics within a subregion of the grid. Modeling higher-order structures (simplicial complexes) allows us to move beyond simple connectivity and capture how physical processes and attack effects propagate through groups of components. This is especially important for cyber-physical security: an attack may begin at a single node but affect according to group-level topology, not just direct edges. Understanding this structure helps detection models reason about cascading failures, coordinated attacks, and localized anomalies.

1) Transmission vs Distribution Networks: Transmission networks exhibit looped interconnections and meshed subregions, which naturally give rise to clique-based triangles and tetrahedra. In contrast, distribution networks are often operated with fewer closed loops with little to no clique-based higher-order simplices due to their acyclic feeder (i.e., tree-like) structures. For example, Fig. 2 illustrates a representative transmission power system of the Hawaii Synthetic Grid [27], [28] and distribution power system of the IEEE 33-bus network [29] and their corresponding topologies. In Fig. 2(a), we illustrate how triangles and tetrahedral shapes naturally exist in transmission networks, requiring an advanced model (i.e., the proposed SRGNN) to capture such complex structures, whereas distribution networks, as shown in Fig. 2(b), present simple tree-like structures that existing GNN models can capture. Consequently, the proposed SRGNN is evaluated herein on large-scale transmission systems (2,869, 9,241, and 70,000), where higher-order simplicial relations are expressive to capture group-wise couplings and cascading behaviors.

2) Construction of Incidence Matrices: In this paper, we propose the construction of higher-order graph representations using incidence matrices $R_k \in \mathcal{R}$ derived from grid connectivity. The node-to-edge matrix R_1 encodes which buses are connected by lines, the edge-to-triangle matrix R_2 identifies 2-simplices formed by bus triplets, and the triangle-to-tetrahedron matrix R_3 captures 3-simplices among bus quartets. Given the grid connectivity graph $G = (V, E)$ (buses as nodes and transmission lines as edges), we identify 2-simplices (triangles) and 3-simplices (tetrahedra) via clique-based construction. A 2-simplex is formed by any triplet of buses $\{v_i, v_j, v_k\}$ such that all three edges (v_i, v_j), (v_j, v_k), and (v_i, v_k) exist in E (i.e., a 3-clique). Similarly, a 3-simplex is formed by any quartet of buses $\{v_i, v_j, v_k, v_\ell\}$ such that all six pairwise edges exist in E (i.e., a 4-clique). Once the sets of triangles $\mathcal{K}_2$ and tetrahedra $\mathcal{K}_3$ are identified, the incidence matrices $R_2 \in \mathbb{R}^{|E|\times|\mathcal{K}_2|}$ and $R_3 \in \mathbb{R}^{|\mathcal{K}_2|\times|\mathcal{K}_3|}$ are constructed by assigning a consistent orientation to each simplex and setting entries to ± 1 when a lower-order simplex is a face of a higher-order simplex (and 0 otherwise), consistent with the boundary-operator definition in Section II. The selected orientation does not affect the physical interpretation of the network or the resulting simplicial operators since reversing the direction of a line simply flips the signs in the corresponding column of R_1 and the associated columns of higher-order incidence matrices, while leaving the induced Laplacian operators and message-passing behavior unchanged. The orientation is therefore a mathematical convention used to define consistent incidence relationships rather than a representation of physical power flow direction [24]. The clique construction is also consistent with the algebraic coupling induced by simplicial operators. In particular, the upper-adjacency term in the k-Laplacian, $R_{k+1}R_{k+1}^\top$, explicitly couples k-simplices that belong to the same $(k{+}1)$-simplex. As a result, when a subset of buses participates in the same simplex, perturbations at one element propagate through shared cofaces in the lifted representations, which motivates both (i) modeling such structures in SRGNN and (ii) evaluating topology-aware node-selection strategies that target simplex-participating buses. Incidence matrices $\mathcal{R}$ allow us to lift signals to higher dimensions and propagate them across simplex hierarchies, which enables graph-based ML models to capture the behavior of structured group buses. The higher-order representation forms the foundation of our proposed SRGNN architecture for attack detection and localization, described in the following section.

3) Illustrative Example: We use Fig. 2(a) as an illustrative example consisting of 37 buses and 60 transmission lines (pairwise connections), including 5 triangles (2-simplices) and 1 tetrahedron (3-simplex) to reflect how A as well as R_1, R_2, and R_3 are constructed.

a) Adjacency Matrix (A): Let the buses be indexed as $V = \{1, 2, \ldots, 37\}$, with $A \in \{0, 1\}^{37\times 37}$ where $A_{ij} = 1$ if a transmission line connects buses i and j, and $A_{ij} = 0$ otherwise. A is constructed and fed into graph-based models as the spatial features of the transmission network topology.

b) Node–Edge Incidence Matrix (R_1): The construction of R_1 follows a standard incidence convention [25]. R_1 is constructed and fed into graph-based models to capture groups of transmission lines as separate entities, allowing for higher-order spatial modeling of the transmission network topology. In Fig. 2(a), $R_1 \in \{0, \pm 1\}^{37\times 60}$ due to the presence of 37 buses and 60 transmission lines. For each transmission line, an arbitrary but fixed direction is assigned (here selected to be consistent with a clockwise traversal in Fig. 2(a)). Each column of R_1 corresponds to a single line and contains exactly two nonzero entries, where a value of -1 at the row

corresponding to the sending bus of that line, and a value of $+1$ at the row corresponding to the receiving bus. All other entries in that column have the value 0. For example, the column corresponding to line $(1, 2)$ in Fig. 2(a) contains -1 in row 1 and $+1$ in row 2, indicating that this line connects bus 1 to bus 2 with the selected orientation.

c) Edge–Triangle Incidence Matrix (R_2): R_2 is constructed and fed into the graph-based models to capture higher-order spatial interactions among the groups of transmission lines that jointly form closed loops (triangles). In Fig. 2(a), $R_2 \in \{0, \pm 1\}^{60\times 5}$ due to the presence of 60 edges and 5 triangles. We build R_2 following the same convention as R_1. Each column of R_2 corresponds to a single triangular loop (2-simplex), and each row corresponds to a transmission line. For example, the first column of R_2 corresponds to triangle $\varsigma_1 = (1, 2, 3)$ and has nonzero entries in the rows associated with lines $(1, 2)$, $(2, 3)$, and $(1, 3)$, indicating that these three lines form the closed loop.

d) Triangle–Tetrahedron Incidence Matrix (R_3): R_3 is constructed and fed into the graph-based models to capture region-level higher-order spatial interactions among the groups of triangular loops that form closed regions (tetrahedra). In Fig. 2(a), $R_3 \in \{0, \pm 1\}^{5\times 1}$ due to the presence of 5 triangles and 1 tetrahedron. The network contains a single fully connected four-bus subgraph formed by buses $\{0, 34, 36, 21\}$, which corresponds to one 3-simplex (tetrahedron). This tetrahedron is bounded by four triangular faces, namely $\varsigma_2 = (0, 34, 36)$, $\varsigma_3 = (34, 36, 21)$, $\varsigma_4 = (36, 21, 0)$, and $\varsigma_5 = (0, 34, 21)$. Each row of R_3 corresponds to a triangle, and the single column corresponds to the tetrahedron.

The aforementioned example shows how A captures pairwise bus connectivity, while R_1, R_2, and R_3 progressively encode line-level, triangle-level, and tetrahedron-level structures, respectively. These matrices form the algebraic foundation for higher-order message passing in simplicial-based GNNs. Once the matrices are constructed for each of the considered transmission systems (2,869, 9,241, and 70,000), we feed them into the simplicial-based GNNs as spatial features.

III. Proposed Simplicial-Based Neural Network Models

In this section, we introduce three simplicial-based neural network architectures: SRGNN, SGNN, and SNN. The proposed SRGNN model integrates GNNs with higher-order topological structures—namely, edges, triangles, and tetrahedra—to capture both pairwise and group-wise interactions across time. To isolate the contributions of different architectural components, we also develop two simplified variants. The first, SGNN, excludes the recurrent temporal module, enabling analysis of spatial message passing in isolation. The second, SNN, further omits the GCN backbone, allowing us to evaluate the expressivity of purely simplicial propagation. Together, these models provide a systematic framework for analyzing the roles of temporal and topological structures in power grid behavior.

A. Simplicial Recurrent Graph Neural Network (SRGNN)

Our proposed SRGNN architecture extends standard GNNs by incorporating higher-order topological features from power grid structures. Unlike conventional GNNs that propagate signals via pairwise edges, our SRGNN leverages a hierarchy of simplices—nodes, edges, triangles, and tetrahedra—to capture both local and group-wise interactions. Our SRGNN multiscale design enables richer representations of cascading and collective behaviors in power networks. As illustrated in Fig. 3, the model first processes each temporal graph snapshot using a spatial encoder composed of a graph convolution branch and multiple simplicial branches, which extract geometric features from edges, triangles, and tetrahedra. The multiscale representations are aggregated to produce a node-level embedding that encodes localized and topologically-informed activity. The resulting embeddings are then passed through a recurrent long short-term memory (LSTM) block, which captures temporal dependencies across graph snapshots. Finally, a softmax output layer produces the classification scores. This design allows our SRGNN to jointly model localized grid activity and its temporal evolution, enabling both attack detection and localization. The training algorithm of our SRGNN model is summarized in Algorithm 1 for the detection task, where Lines 1 – 41 are repeated for each node in a graph G to perform the localization task.

1) Input Layer: We model the power grid as a temporal sequence of graph snapshots $G = \{\mathcal{G}^{(t)} = V, E\}_{t=1}^{T}$, where V is the fixed set of nodes representing buses, and E are the edges representing the transmission lines. Each snapshot $\mathcal{G}^{(t)}$is associated with a node feature matrix $X^{(t)} \in \mathbb{R}^{|V|\times d}$, where each row encodes the active and reactive power measurements of a bus at time t and the number of features d.

2) Graph Convolutional Layer: To capture localized pairwise interactions, each node feature matrix $X^{(t)}$ is passed through a stack of graph convolutional layers. Let $A^{(t)} \in \{0, 1\}^{|V|\times|V|}$ denote the adjacency matrix of the graph snapshot $\mathcal{G}^{(t)}$, and let $*_{\mathcal{G}^{(t)}}$ represent graph convolution with respect to $A^{(t)}$. Each layer is parameterized by a learnable weight matrix $W_{\text{GCN}}^{(\ell)} \in \mathbb{R}^{d_\ell \times d_{\ell+1}}$ and bias vector $b^{(\ell)} \in \mathbb{R}^{d_{\ell+1}}$, where d_ℓ is the feature dimension at layer ℓ. The node embeddings are updated as:

$$Z_{\text{GCN}}^{(\ell+1,t)} = \text{ReLU}\left(W^{(\ell)} *_{\mathcal{G}^{(t)}} X^{(\ell,t)} + b^{(\ell)}\right), \quad \ell = 0, \ldots, L. \tag{7}$$

To incorporate higher-order structure, we add three parallel message-passing modules per layer—one for 1-simplices (edges), one for 2-simplices (triangles), and one for 3-simplices (tetrahedra)—which use incidence matrices $R_k \in \mathcal{R}$ to lift and project signals across dimensions.

3) Simplicial Branch (Nodes ↔ Edges): Let $R_1 \in \mathbb{R}^{|V|\times|E|}$ be the node-to-edge incidence matrix. The node feature matrix at layer ℓ and time t, denoted $X^{(\ell,t)} \in \mathbb{R}^{|V|\times d}$, is lifted to edge space, transformed via a learnable linear map, and projected back to node space, with resulting embedding

$$Z_{\text{simp}}^{(\ell,t)} = R_1 \cdot \text{ReLU}(W_{\sigma_1}^{(\ell)} \cdot R_1^\top Z_{\text{GCN}}^{(\ell,t)}), \tag{8}$$

Algorithm 1: Proposed SRGNN Model Training

Input : Temporal graph snapshots $\{\mathcal{G}^{(t)} = V, E\}_{t=1}^{\mathcal{T}}$
Feature matrix $X^{(t)}$
Incidence matrices: R_1, R_2, R_3
Initialize: Model weights and biases:
$W_{\text{GCN}}^{(\ell)}$, $W_{\sigma_1}^{(\ell)}$, $W_{\sigma_2}^{(\ell)}$, $W_{\sigma_3}^{(\ell)}$, $W_{\text{LSTM}}^{(\ell)}$, W_o, $b^{(\ell)}$

for *epoch* $= 1$ **to** N **do**
  for *each sequence* $\{G^{(t)}\}_{t=1}^{\mathcal{T}}$ **do**
    for $t = 1$ **to** $\mathcal{T}$ **do**
      Initialize input features:
      $X^{(0,t)} \leftarrow X^{(t)}$
      for $\ell = 0$ **to** $L-1$ **do**
        Graph Convolutional Branch:
        $Z_{\text{GCN}}^{(\ell+1,t)} \leftarrow \text{ReLU}(W_{\text{GCN}}^{(\ell)} *_{\mathcal{G}^{(t)}} X^{(\ell,t)} + b^{(\ell)})$
        Simplicial Branch:
        $Z_{\text{simp}}^{(\ell,t)} \leftarrow R_1 \cdot \text{ReLU}(W_{\sigma_1}^{(\ell)} \cdot R_1^\top Z_{\text{GCN}}^{(\ell,t)})$
        Triangle Branch:
        $Z_{\text{E}}^{(\ell,t)} \leftarrow R_1^\top Z_{\text{GCN}}^{(\ell,t)}$
        $Z_{\text{E, tri}}^{(\ell,t)} \leftarrow R_2^\top Z_{\text{E}}^{(\ell,t)}$
        Tetrahedron Branch:
        $Z_{\text{tri, tetra}}^{(\ell,t)} \leftarrow R_3 \cdot \text{ReLU}(W_{\sigma_3}^{(\ell)} \cdot R_3^\top Z_{\text{E, tri}}^{(\ell,t)})$
        Triangle Projection:
        $\mathbf{Z}^{(\ell,t)} \leftarrow Z_{\text{E, tri}}^{(\ell,t)} + Z_{\text{tri, tetra}}^{(\ell,t)}$
        $Z_{\text{tetra, tri}}^{(\ell,t)} \leftarrow R_2 \cdot \text{ReLU}(W_{\sigma_2}^{(\ell)} \cdot \mathbf{Z}^{(\ell,t)})$
        $Z_{\text{tri}}^{(\ell,t)} \leftarrow R_1 \cdot Z_{\text{tetra, tri}}^{(\ell,t)}$
        Tetrahedron Output:
        $Z_{\text{tetra}}^{(\ell,t)} \leftarrow Z_{\text{tri}}^{(\ell,t)}$
        Node Embedding Aggregation:
        $\boldsymbol{\mathcal{Z}}^{(\ell+1,t)} \leftarrow \text{ReLU}\left(Z_{\text{GCN}}^{(\ell,t)} + Z_{\text{simp}}^{(\ell,t)} + Z_{\text{tri}}^{(\ell,t)} + Z_{\text{tetra}}^{(\ell,t)}\right)$
      Store Layer Output:
      $\boldsymbol{\mathcal{Z}}^{(t)} \leftarrow \boldsymbol{\mathcal{Z}}^{(L,t)}$
    LSTM Update:
    $\mathbf{I}^{(t)} = \sigma\left(W_{\text{I}}\boldsymbol{\mathcal{Z}}^{(t)} + U_{\text{I}}\mathbf{H}^{(t-1)} + V_{\text{I}}\mathbf{C}^{(t-1)} + b_{\text{I}}\right)$
    $\mathbf{F}^{(t)} = \sigma\left(W_{\text{F}}\boldsymbol{\mathcal{Z}}^{(t)} + U_{\text{F}}\mathbf{H}^{(t-1)} + V_{\text{F}}\mathbf{C}^{(t-1)} + b_{\text{F}}\right)$
    $\widetilde{\mathbf{C}}^{(t)} = \tanh\left(W_{\text{C}}\boldsymbol{\mathcal{Z}}^{(t)} + U_{\text{C}}\mathbf{H}^{(t-1)} + b_{\text{C}}\right)$
    $\mathbf{C}^{(t)} = \mathbf{F}^{(t)} \odot \mathbf{C}^{(t-1)} + \mathbf{I}^{(t)} \odot \widetilde{\mathbf{C}}^{(t)}$
    $\mathbf{O}^{(t)} = \sigma\left(W_{\text{O}}\boldsymbol{\mathcal{Z}}^{(t)} + U_{\text{O}}\mathbf{H}^{(t-1)} + V_{\text{O}}\mathbf{C}^{(t)} + b_{\text{O}}\right)$
    $\mathbf{H}^{(t)} = \mathbf{O}^{(t)} \odot \tanh(\mathbf{C}^{(t)})$
    Final Prediction:
    $\hat{y} \leftarrow \text{Softmax}(W_{\text{CLS}} \cdot \mathbf{H}^{(\mathcal{T})} + b_{\text{CLS}})$
    Cross-Entropy Loss:
    $\mathcal{L}(\Theta) = -\frac{1}{|X_{\text{TR}}|}\sum_{i=1}^{|X_{\text{TR}}|}\sum_{c=1}^{C} y_i^{(c)} \log(\hat{y}_i^{(c)})$
  Parameter Optimization:
  Update all weights and biases by backpropagation using the Adam optimizer.

Output : Trained model parameters

where $W_{\sigma_1}^{(\ell)}$ is the layer-specific weight matrix for the simplicial branch σ_1. Next, we discuss the SRGNN forward pass to highlight the interaction between higher-order spatial encoding and temporal modeling. For each snapshot $\mathcal{G}^{(t)}$ with node measurements $X^{(t)}$, the spatial encoder aggregates the outputs of the graph convolution branch and the simplicial branches (via R_1, R_2, and R_3) to produce a node embedding $\mathcal{Z}^{(t)}$. The recurrent block then processes the embedding sequence $\{\mathcal{Z}^{(t)}\}_{t=1}^{\mathcal{T}}$, where $\mathcal{T} = 5{,}000$ is duration of the total time-series readings with a tuned $T = 48$ sliding window size, to capture temporal dependencies (i.e., time-series readings of active and reactive power measurement readings as will be introduced in Section IV-A) across snapshots through its hidden state, and the softmax layer maps the recurrent output to the final prediction. Formally, we write

$$\mathbf{Z}^{(t)} = f_\theta\Big(X^{(t)}, A^{(t)}, R_1, R_2, R_3\Big), \tag{9}$$

$$\mathbf{h}^{(t)} = \text{LSTM}\left(\mathbf{h}^{(t-1)}, \mathbf{Z}^{(t)}\right), \tag{10}$$

$$\hat{y}^{(t)} = \text{Softmax}\Big(W_o\mathbf{h}^{(t)} + b\Big). \tag{11}$$

where $f_\theta(\cdot)$ denotes the multiscale encoder defined by the GCN and simplicial message-passing modules. This construction allows SRGNN to jointly capture higher-order spatial structure (through incidence-based lifting/projection) and temporal evolution (through the LSTM cells).

4) Triangle Branch (Edges ↔ Triangles): Let $R_2 \in \mathbb{R}^{|E|\times|\mathcal{K}_2|}$ be the edge-to-triangle incidence matrix, where $\mathcal{K}_2$ denotes the set of 2-simplices. Starting from edge embeddings $E^{(\ell,t)} = R_1^\top X^{(\ell,t)}$, we lift to triangle space, apply a learnable linear transformation, and project back to edge and then node space, resulting in the embedding

$$Z_{\text{tri}}^{(\ell,t)} = R_2 \cdot \text{ReLU}(W_{\sigma_2}^{(\ell)} \cdot R_2^\top E^{(\ell,t)}), \quad Z_{\text{tri}}^{(\ell,t)} = R_1 \cdot Z_{\text{tri}}^{(\ell,t)}. \tag{12}$$

Here, $W_{\sigma_2}^{(\ell)}$ is the triangle-branch σ_2 weight matrix at layer ℓ.

5) Tetrahedron Branch (Triangles ↔ Tetrahedra): Let $R_3 \in \mathbb{R}^{|\mathcal{K}_2|\times|\mathcal{K}_3|}$ be the triangle-to-tetrahedron incidence matrix, where $\mathcal{K}_3$ denotes the set of 3-simplices. Triangle embeddings are computed as $Z_{\text{tetra}}^{(\ell,t)} = R_2^\top Z_{\text{tri}}^{(\ell,t)}$, then lifted to tetrahedron space, processed, and projected back to triangles:

$$\begin{aligned} Z_{\text{tri, tetra}}^{(\ell,t)} &= R_3 \cdot \text{ReLU}\left(W_{\sigma_3}^{(\ell)} \cdot R_3^\top t^{(\ell,t)}\right), \\ \mathbf{Z}^{(\ell,t)} &= Z_{\text{simp, tri}}^{(\ell,t)} + Z_{\text{tri, tetra}}^{(\ell,t)}, \\ Z_{\text{tri}}^{(\ell,t)} &= R_2 \cdot \text{ReLU}\left(W_{\sigma_2}^{(\ell)} \cdot \mathbf{Z}^{(\ell,t)}\right), \text{ and} \\ Z_{\text{tetra}}^{(\ell,t)} &= R_1 \cdot Z_{\text{tri}}^{(\ell,t)}. \end{aligned} \tag{13}$$

The enriched node embeddings $Z_{\text{tetra}}^{(\ell,t)}$ capture information propagated through triangle and tetrahedron interactions and are later used in the overall node update. This produces enriched triangle features, which are then added to the triangle branch before projecting back to node space. The final node embeddings $\mathcal{Z}^{(\ell+1,t)}$ are updated by aggregating the outputs of the graph convolutional layer and the three higher-order branches corresponding to 1-simplices (edges), 2-simplices (triangles), and 3-simplices (tetrahedra) as follows

$$\mathcal{Z}^{(\ell+1,t)} = \text{ReLU}\left(Z_{\text{GCN}}^{(\ell,t)} + Z_{\text{simp}}^{(\ell,t)} + Z_{\text{tri}}^{(\ell,t)} + Z_{\text{tetra}}^{(\ell,t)}\right). \tag{14}$$

B. Simplicial Graph Neural Network (SGNN)

While SRGNN models temporal dynamics explicitly through a recurrent branch, we isolate the role of higher-order spatial structures by removing temporal recurrence in the proposed SGNN herein that incorporates simplicial complexes in a feed-forward GNN mechanism. To capture both local

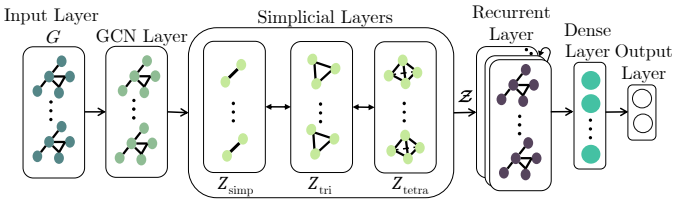


Fig. 3. Illustration of the proposed SRGNN model architecture.

pairwise interactions and higher-order group dynamics in power grids, we propose another hybrid architecture, SGNN. The SGNN integrates traditional graph convolutional layers with message-passing modules that operate across simplicial hierarchies. The model first applies a sequence of GCN layers to extract localized spatial representations based on immediate neighborhood connectivity. The GCN layers model standard first-order neighbor information flow among nodes. To incorporate a higher-order topology, SGNN introduces a parallel simplicial branch that processes lifted signals from node-to-edge, edge-to-triangle, and triangle-to-tetrahedron levels using the incidence matrices R_1, R_2, and R_3, respectively. This allows SGNN to exploit group-wise patterns that arise in tightly interconnected subregions of the grid. The outputs of the simplicial branch are projected back to the node space and added to the GCN features to form a multiscale representation. The final node embeddings $\mathcal{Z}^{(L,t)}$ are pooled and classified at the graph level. This hybrid approach allows SGNN to model both fine-grained spatial details and coarse-grained structural patterns, enabling robust detection of distributed attacks that exploit multi-node correlations. Considering Lines 1 – 25 from Algorithm 1 and changing Line 34 to $\hat{y}^{(t)} \leftarrow \text{Softmax}(W_o \cdot \boldsymbol{\mathcal{Z}}^{(t)} + b)$ results in the training algorithm of the SGNN model.

C. Pure Simplicial Model (SNN)

To isolate the role of simplicial structure, we also evaluate a proposed variant with no GCN backbone, namely, SNN. In the proposed SNN model, node features are projected directly into a hidden space and passed through purely simplicial layers. Considering Lines 1 – 6 and 9 – 25 from Algorithm 1 as well as changing Line 34 to $\hat{y}^{(t)} \leftarrow \text{Softmax}(W_o \cdot \boldsymbol{\mathcal{Z}}^{(L,t)} + b)$ results in the training algorithm of the SNN model. Despite the absence of graph convolutions, the SNN architecture captures pairwise (edge-level) relations via the node-to-edge incidence matrix R_1, triadic interactions through the edge-to-triangle matrix R_2, and higher-order group dynamics via the triangle-to-tetrahedron matrix R_3. This minimal configuration tests whether higher-order topological signals alone are sufficient for accurate attack detection and localization, as will be shown in V.

IV. Data Preparation and Experimental Setup

This section presents the experimental framework used to evaluate the performance of our proposed and benchmark models. We outline how benign power-flow snapshots are generated on large-scale power systems. We then introduce the threat model, including attack functions. Next, we outline the methods used to select attacked nodes. We then describe benchmark detector models used for comparative evaluation.

A. Dataset Generation

We generate benign power-flow snapshots over time steps for each power system by varying the base loads and solving an AC power flow using MATLAB. We denote the benign active and reactive power demands at bus v by $P(v)$ and $Q(v)$, respectively. We read a seasonal profile $M(t)$ from publicly available Electric Reliability Council of Texas (ERCOT) [**?**] system load data to reflect realistic bulk-load evolution with temporal variability over time using a 15-minute data sampling rate. Specifically, we define the variable loads

$$P^{(t)}(v) = P(v)\,\delta_t(v), \quad Q^{(t)}(v) = Q(v)\,\delta_t(v), \tag{15}$$

where $\delta_t(v) \sim \mathcal{N}\big(1 + 0.025M(t), 0.01^2\big)$ independently for each v. At each iteration, we solve the AC power-flow equations

$$\begin{cases} P^{(t)}(v) = \sum_{u:(v,u)\in E} \hat{V}_v \hat{V}_u \big(\hat{G}_{vu}\cos\theta_{vu} + \hat{B}_{vu}\sin\theta_{vu}\big), \\ Q^{(t)}(v) = \sum_{u:(v,u)\in E} \hat{V}_v \hat{V}_u \big(\hat{G}_{vu}\sin\theta_{vu} - \hat{B}_{vu}\cos\theta_{vu}\big), \end{cases} \tag{16}$$

for all v, where $\hat{V}_v$ is the voltage magnitude, $\theta_{vu} = \theta_v - \theta_u$ the angle difference, and $\hat{G}_{vu} + j\hat{B}_{vu}$ the line admittance. We retain only those snapshots for which the solver converges successfully, resulting in a set of benign cases per system. Although our detectors use the measurements $[P^{(t)}(v), Q^{(t)}(v)]$ as input features, we solve AC power flow at each iteration to ensure that each snapshot corresponds to a physically feasible operating point consistent with network constraints and admittance parameters. In particular, the AC solver enforces realistic coupling between loads and the underlying topology, and we discard non-convergent cases to avoid introducing non-physical samples into the benign distribution.

1) Normal Operation Samples: For each converged snapshot $G^{(t)}$, we collect a feature matrix $X^{(t)} \in \mathbb{R}^{|V|\times 2}$ with $|V|$ buses, whose v-th row is $\big[P^{(t)}(v),\, Q^{(t)}(v)\big]$. Collecting all snapshots yields $\{G^{(t)}\}_{t=1}^{\mathcal{T}_{\text{BEN}}}$ (where $\mathcal{T}_{\text{BEN}} = 2{,}500$ representing the total duration of benign time-series data), which serves as the baseline distribution for our subsequent attack detection experiments. We use $\big[P^{(t)}(v),\, Q^{(t)}(v)\big]$ as input features with benign labels as our threat model (in Section IV-B) compromises these measurement streams at attacked buses.

2) Utilized Large-Scale Transmission Systems: The aforementioned simulation procedure is applied to three large-scale transmission power systems of varying topology and scale. The 2,869-bus system is based on a real-world European transmission grid [30], [31]. The 9,241-bus system also represents a high-voltage European network [30], [31]. The 70,000-bus system is a synthetic grid constructed to reflect the topology and electrical characteristics of the Eastern Interconnection of the United States [27], [28].

B. Threat Model

Our threat model is characterized along two complementary dimensions: a spatial aspect that determines which buses are targeted for attack, and a temporal aspect that specifies how the

attack functions manipulate power measurements over time. The attack functions described next result in $\{G^{(t)}\}_{t=1}^{\mathcal{T}_{\text{MAL}}}$, where $\mathcal{T}_{\text{MAL}} = 2{,}500$ representing the total duration of malicious time-series data.

1) Spatial Aspect: We consider two realistic levels of attacker knowledge of the system: attackers with no knowledge and attackers with full system knowledge. We evaluate performance under different proposed and benchmark strategies for selecting the attacked node set $S \subset V$. To ensure a fair comparison, we fix $|S|$ to be the number of unique nodes that participate in at least one 2-simplex. This ensures all selection strategies always select an equal number of nodes to attack, which is 10% of the bus nodes. To determine the percentage of attacked buses that is large enough to induce nontrivial system impact, yet sparse enough to maintain stealthiness, we tuned this fraction using sequential search on the search space $\{5\%, 10\%, 15\%, ..., 30\%\}$.

a) Proposed Attacked Node Selection Strategies: We propose targeting nodes that participate in higher-order simplices to evaluate how group-wise interactions influence detection and localization performance, as these nodes are embedded in tightly coupled topological structures that may amplify attack impact. We apply each attack function on nodes selected as follows. For the proposed 2-simplex-based selection, let S be all nodes that participate in at least one 2-simplex $\mathcal{K}_2$:

$$S = \{\, v \in V : \exists [v_{i,j}] \in \mathcal{K}_2 \}. \tag{17}$$

For the proposed 2- and 3-simplex-based selection, let S be all nodes that appear in a 2-simplex $\mathcal{K}_2$ or 3-simplex $\mathcal{K}_3$:

$$S = \{\, v \in V : \exists [v_{i,j}] \in \mathcal{K}_2 \ \vee \ \exists [v_{i,j}] \in \mathcal{K}_3 \}. \tag{18}$$

b) Benchmark Attacked Node Selection Strategies: For performance comparison, we apply each attack function to benchmark node selection strategies as follows. For random node selection, select $|S|$ distinct nodes uniformly to attack at random from V. For degree centrality-based node selection, we compute the normalized degree centrality of each node to quantify its local influence in the network. For node $v \in V$, the normalized degree centrality is defined as

$$C_D(v) = \frac{\deg(v)}{\max(\deg)}, \tag{19}$$

where $\deg(v)$ is the number of edges connected to node v, and $\max(\deg)$ is the maximum degree over all nodes in the graph. We select the top $|S|$ nodes with the highest $C_D(v)$ scores for attack. For betweenness centrality-based node selection, for each node $v \in V$, we compute

$$C_B(v) = \sum_{i,j \in V} \frac{\Upsilon(i,j \mid v)}{\Upsilon k(i,j)}, \tag{20}$$

where $\Upsilon(i,j)$ is the total number of shortest paths between nodes i and j, and $\Upsilon(i,j \mid v)$ is the number of those paths that pass through node v. We select the top $|S|$ nodes with the highest $C_B(v)$ values. In the event of ties, degree centrality $C_D(v)$ is used as a secondary criterion.

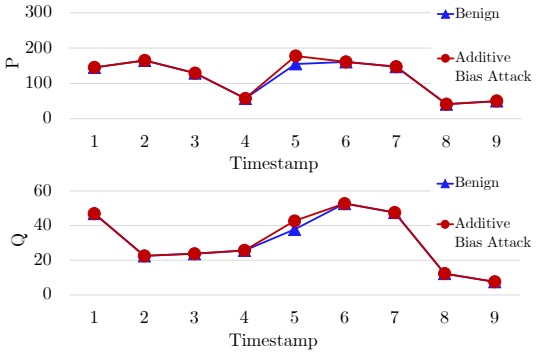


Fig. 4. Sample $P(v)$ and $Q(v)$ of normal operation and additive bias attack.

2) Temporal Aspect: After identifying the nodes to attack, the measurement readings are manipulated. The considered attacks are modeled as telemetry-layer compromises of measurement streams, rather than as physical faults or contingencies. Such behaviors are consistent with how real attackers can disrupt grid monitoring and control data paths while remaining operationally plausible without immediately causing protective trips, making them non-trivial to detect via simple thresholding or naive screening [21], [20]. We simulate the four following types of attacks on a subset $S \subset V$, where $\breve{P}^{(t)}(v)$ and $\breve{Q}^{(t)}(v)$ represent the attacked active and reactive power readings, respectively.

a) Additive Bias Attack: We inject a multiplicative bias with randomized sign and magnitude into both active and reactive power at the attacked buses. Specifically, for each attacked bus $v \in S$ at time t, we draw $d \sim \text{Bernoulli}(0.5)$ and $u \sim \text{Uniform}(0,1)$ and set the perturbation factor $\epsilon = (-1)^d \cdot a \cdot u \cdot \text{range}$, where a is a fixed scaling coefficient $a = 2$ and range is a fixed maximum perturbation range. The injected measurements follow:

$$\breve{P}^{(t)}(v) \leftarrow P^{(t)}(v)\,(1+\epsilon), \quad \breve{Q}^{(t)}(v) \leftarrow Q^{(t)}(v)\,(1+\epsilon). \tag{21}$$

Fig. 4 illustrates a representative additive-bias realization on a single attacked bus over time, showing how the reported injections deviate from the benign trajectory according to the aforementioned multiplicative rule.

b) Previous Temporal Replay Attack: This attack reuses measurements from the previous time step, erasing new fluctuations and hiding any abnormal behavior at the targeted nodes. In our MATLAB implementation, the first iteration is kept benign, and the replay-previous rule is applied for $t = 2$ onwards. For each $v \in S$ and $t > 1$, we substitute the previous timestamp's power reading:

$$\breve{P}^{(t)}(v) \ \leftarrow \ P^{(t-1)}(v), \quad \breve{Q}^{(t)}(v) \ \leftarrow \ Q^{(t-1)}(v). \tag{22}$$

c) Random Temporal Replay Attack: This attack reuses power readings from a random previous timestep, which introduces delayed responses that mimic plausible but outdated behavior, disrupting temporal consistency. For each $v \in S$ and $t > T_\rho$, we choose a random lag $\tau \in \{1, 2, \ldots, T_\rho\}$, where the

maximum $T_\rho = 5$ selected from search space $\{1, 2, \cdots, 6\}$. The replayed snapshot is as follows:

$$\breve{P}^{(t)}(v) \leftarrow P^{(t-\tau)}(v), \quad \breve{Q}^{(t)}(v) \leftarrow Q^{(t-\tau)}(v), \tag{23}$$

d) DoS Attack: This attack freezes the power demand of a node for a fixed duration, simulating a sensor failure or communication blackout. Once a bus $v \in S$ is attacked at time t_0, its power readings remain fixed for the next $\Delta = 3$ timesteps, selected from search space $\{1, 2, \cdots, 5\}$. For any bus v that is frozen at time t, its reported injections are set to the previous iteration's values:

$$\breve{P}^{(t)}(v) = P^{(t_0)}(v), \quad \breve{Q}^{(t)}(v) = Q^{(t_0)}(v), \quad t_0 \le t < t_0 + \Delta, \tag{24}$$

e) Attack Parameters: For each attack function, the parameters are tuned from the aforementioned search spaces using sequential grid search. An optimal attack function value is selected such that it induces detectable temporal inconsistency without becoming trivially obvious.

C. Operational Integration with Grid Systems

The proposed model presents a data-driven solution that is trained on measurement readings rather than using state estimation. In the development stage, the model training and tuning are performed offline using available historical benign data and real/simulated attack scenarios, without disrupting system operations. After the development stage, the proposed models are deployed as a data-driven monitoring layer that complements existing transmission system operational tools. In practice, the proposed models are to be integrated into the supervisory control and data acquisition (SCADA)/energy management systems (EMS) pipeline to analyze incoming measurements to perform the detection task on the overall system state (normal/under attack operation) and localize where the attack is (which node) in real-time. Online deployment requires only forward inference, which is performed in the order of milliseconds. When the models flag manipulated measurements, validation and mitigation procedures are triggered. Unlike statistical model-driven residual-based schemes (e.g., bad data detector (BDD)) that require a complete measurement model and redundancy assumptions, SRGNN presents a data-driven model and therefore remains applicable under large-scale operational variability and changing telemetry availability. Leveraging higher-order simplices enables the proposed models to capture group-wise dependencies among electrically coherent multi-node structures (beyond pairwise connectivity), improving sensitivity to coordinated manipulations that may otherwise appear locally consistent. The attack example shown in Fig. 4, is flagged by the proposed models within the corresponding analysis window, enabling near real-time alarms that can be followed by operator validation and mitigation.

D. Benchmark Detectors

To ensure a comprehensive analysis, we evaluate eleven benchmark detectors spanning a classic BDD, traditional ML, DL, and graph-based learning approaches. These detectors are selected to capture a diverse range of statistical, shallow, temporal, spatial, and spatiotemporal learning capabilities and to reflect standard baselines used in recent power system attack detection literature.

1) BDD: To complement ML-based detectors with a lightweight, training-free benchmark, we also include a statistical BDD [16]. BDDs present a monitoring practice in power system data quality control, where gross measurement inconsistencies are flagged by measuring how far a new observation deviates from nominal behavior. Concretely, we treat the paired measurements $\big[P(v), Q(v)\big]$ as a low-dimensional multivariate signal and quantify deviation relative to a reference (benign reading interval) operating window using a standardized distance score. Samples whose deviation exceeds a fixed statistical threshold are labeled as attacks.

2) Shallow Models: The used shallow ML models operate on static, fixed-length vectors and provide fast and interpretable baselines, but they do not incorporate time-series dynamics or graph structures. The RF model [7] is trained using bootstrap aggregation to reduce variance and improve robustness. While RFs are robust to noisy features, they treat each input independently and lack temporal or topological awareness. The SVM model [5] with a radial basis function kernel is trained on standardized input features. The SVM learns a static decision boundary and performs well under separable conditions, but does not model time-series trends or structural dependencies.

3) DL Models: We include DL models for their ability to learn nonlinear mappings and temporal correlations. These models ingest sequential data across time windows but do not explicitly leverage graph topology. The FNN model [8] is a multilayer perceptron with rectified linear activations and a softmax classification layer. While FNN is static and does not track temporal evolution, it captures nonlinear interactions in high-dimensional feature space. The RNN model [11] employs LSTM units to capture temporal correlations across input sequences. The RNN models temporal evolution in the data but processes each bus independently, without structural context. The CNN model [12] uses one-dimensional temporal filters to extract local patterns from the input sequences. The CNN learns local time-invariant features efficiently but remains agnostic to spatial relationships between buses.

4) Graph-Based Models: Graph-based models are spatially aware since they incorporate the underlying power grid topology through message passing on the pairwise graph structure. These models enable spatially aware learning but vary when capturing the temporal information. The GNN model [18] applies spatial graph convolutions to bus features using the grid connectivity. The RGNN model [1] extends the base GNN by integrating recurrent units (LSTM) after the graph convolution layers, enabling joint spatiotemporal modeling, but it remains limited to pairwise edges. To further contextualize SRGNN against more advanced graph architectures, we evaluate a Spectral GNN [17], a Transformer GNN [22], and RAGNN [1]. The Spectral GNN performs spectral-domain filtering to capture global smoothness and frequency-selective behaviors over the grid graph [17]. The Transformer GNN re-

places fixed neighborhood aggregation with adaptive neighbor selection, focusing message passing on the most informative neighbors rather than uniformly combining all neighbors. The RAGNN captures the spatiotemporal aspect while assigning higher weights to important measurement readings through an attention mechanism [1].

V. Results and Discussion

In this section, we introduce the hyperparameter tuning process and the evaluation metrics used, followed by discussions on the detection and localization performance, scalability, and additional performance analysis.

TABLE I
Optimal Hyperparameters

Model	Layers	Units	Dropout	Neigh.	Opt.
FNN	4	64	0.2	-	Adam
RNN	3	32	0.4	-	SGD
CNN	4	32	0.3	5	Adam
GNN	5	128	0.3	4	Adam
Spectral GNN	5	64	0.2	5	Adam
Proposed SGNN	5	64	0.0	5	Adam
RGNN	5	64	0.0	4	Rmsprop
Transformer GNN	4	64	0.0	4	Adam
RAGNN	6	64	0.2	4	Rmsprop
Proposed SRGNN	6	64	0.2	4	Adam

A. Optimal Hyperparameters

All models are trained on the same datasets where 70%, 10%, and 20% of the dataset is used for training, validation, and testing, respectively. The validation set is used for a sequential grid-search hyperparameter optimization. Table I summarizes the list of optimal hyperparameters from search spaces of $[2, 3, \cdots, 10]$, $[32, 64, 128, 256, 512]$, $[0, 0.2, 0.4]$, $[4, 5, 6]$, and [SGD, Adam, and Rmsprop] for the number of layers, number of neurons, dropout rate, neighborhood order (neigh.), and optimization function (opt.), respectively. The hyperparameter T is selected via a sequential search over {12, 24, 36, 48, 72, 96}, which correspond to multiples of the 15-minute ERCOT sampling interval and allows for selecting a sliding window ranging from 3 hours (T=12) to full day (T=96) periodicity. It turns out that for recurrent models, $T = 48$ (12 hours) provides the best trade-off between capturing meaningful temporal dependencies in power measurements as well as avoiding model complexity and overfitting.

B. Evaluation Metrics

We evaluate the performance of all detection and localization models using six metrics: accuracy (ACC), DR, precision (PR), F1-Score, false alarm rate (FAR), and the area under the receiver operating characteristic curve (AUC), computed from the number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). Specifically, ACC represents overall classification correctness and is computed as $\mathrm{ACC} = (\mathrm{TP} + \mathrm{TN})/(\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN})$. DR is defined as $\mathrm{DR} = \mathrm{TP}/(\mathrm{TP} + \mathrm{FN})$ and quantifies the proportion of correctly identified attacks. PR is defined as $\mathrm{PR} = \mathrm{TP}/(\mathrm{TP} + \mathrm{FP})$ and measures the fraction of predicted attacks that are true attacks. The F1-Score is the harmonic mean of PR and DR, defined as $\mathrm{F1} = 2\,(\mathrm{PR}\cdot\mathrm{DR})/(\mathrm{PR}+\mathrm{DR})$, and summarizes the balance between missed detections and false alarms. FAR is given by $\mathrm{FAR} = \mathrm{FP}/(\mathrm{FP} + \mathrm{TN})$ and captures the rate of false positives among benign samples. AUC quantifies the area under the ROC curve (TP rate vs. FP rate) and provides a threshold-independent measure of separability between benign and attacked samples.

C. Detection Task

The detection task refers to classifying the overall system status (normal operation vs under attack). Table II summarizes the detection performance of all evaluated models across different attack types, node selection strategies, and grid sizes.

1) Model Performance in Detection Task: The proposed SRGNN consistently surpasses all benchmark models by 9–39% in DR against complex attacks. All proposed models (SNN, SGNN, and SRGNN) outperform benchmarks by 2.9–84.7% in DR as follows.

a) Proposed vs BDD in Detection: The proposed models outperform the BDD by 84.7% in DR, highlighting that distribution-based statistical screening yields negligible detection capability under the considered stealthy attack settings. Notably, despite reporting low FAR (2.3 – 3.8%), BDD attains near-zero DR while the system is under attack as it relies only on distributional deviation of $\big[P(v), Q(v)\big]$ and does not model nonlinear grid dynamics, temporal dependencies, or network structure that ML-based models capture.

b) Proposed vs ML-based Models in Detection: The proposed models outperform the shallow ML-based models by 24.5% in DR and 20.3% in FAR. The proposed models outperform the DL-based benchmarks by 10.5% in DR and 12.4% in FAR. The proposed models outperform the graph-based models by 2.9% in DR and FAR, respectively. These improvements underscore the value of higher-order topological reasoning in enhancing attack detection in smart grid systems.

c) Proposed Models in Detection Performance: Among the proposed models, SRGNN outperforms SGNN and SNN by 10.9–12.3% in DR and 8.2–9.3% in FAR. These improvements highlight the additive value of each architectural component. The GCN branch in SGNN enhances spatial resolution by modeling pairwise interactions, while the recurrent branch in SRGNN captures temporal correlations across snapshots. By fusing both, SRGNN delivers the most robust detection performance under diverse and evolving attack scenarios.

2) Impact of Attack Strategy on Detection Task: We analyze how increasingly topology-aware attack strategies affect detection performance, comparing the proposed models against benchmark methods under random, centrality-based, and simplex-aware node targeting.

a) Proposed vs Random in Detection: As node selection becomes more topology-aware, detection performance

TABLE II
ATTACK DETECTION PERFORMANCE RESULTS (%)

Model	Metric	Attack Node Selection Strategy											
		Benchmark						Proposed					
		Random			Centrality Analysis			2-Simplex			2+3-Simplex		
		2869 bus	9241 bus	70000 bus	2869 bus	9241 bus	70000 bus	2869 bus	9241 bus	70000 bus	2869 bus	9241 bus	70000 bus
BDD	ACC	48.6	49.0	49.4	49.5	49.7	49.8	50.0	50.0	50.0	50.0	50.0	50.0
	DR	1.5	1.8	1.2	1.2	1.2	1.2	0.0	0.0	0.0	0.0	0.0	0.0
	PR	26.3	32.1	34.4	35.3	39.9	42.1	undef.	undef.	undef.	undef.	undef.	undef.
	F1	2.8	3.4	2.3	2.3	2.3	2.3	undef.	undef.	undef.	undef.	undef.	undef.
	FAR	4.1	3.8	2.4	2.2	1.8	1.6	0.0	0.0	0.0	0.0	0.0	0.0
	AUC	48.7	49.2	49.7	49.7	49.9	49.9	50.0	50.0	50.0	50.0	50.0	50.0
RF	ACC	63.2	61.4	62.2	61.6	57.6	58.3	57.0	53.8	54.9	55.5	51.4	52.8
	DR	65.7	62.0	64.3	62.7	59.3	61.3	57.6	54.4	56.3	55.8	53.8	56.5
	PR	66.2	62.1	63.7	63.7	58.9	61.6	56.7	55.4	56.4	56.6	53.7	56.1
	F1	65.9	62.0	64.0	63.2	59.1	61.4	57.1	54.9	56.3	56.2	53.7	56.3
	FAR	35.1	33.3	31.0	36.9	37.4	34.8	42.5	41.5	39.6	43.3	42.1	40.3
	AUC	66.2	63.0	65.1	63.5	59.8	62.1	58.5	54.9	57.2	56.6	54.3	57.4
SVM	ACC	67.8	63.8	66.3	65.6	61.6	63.1	60.7	56.4	59.4	60.1	56.2	57.0
	DR	69.8	65.3	66.6	65.9	62.9	64.4	61.1	58.4	59.5	61.0	55.8	59.2
	PR	70.0	65.7	66.7	66.2	63.6	65.0	61.0	57.0	58.3	61.5	56.1	58.6
	F1	69.9	65.5	66.6	66.0	63.2	64.7	61.0	57.7	58.9	61.2	55.9	58.9
	FAR	31.0	26.5	24.6	33.3	30.6	27.8	38.5	34.8	31.4	40.3	37.1	33.7
	AUC	70.6	65.9	67.2	66.8	63.6	65.3	62.1	58.9	60.1	61.7	56.4	59.9
FNN	ACC	76.4	76.7	76.0	72.7	72.5	73.2	68.9	69.1	69.8	67.6	67.1	69.9
	DR	77.3	77.2	76.5	74.7	73.1	74.5	69.3	70.9	71.1	68.0	68.8	68.9
	PR	78.6	76.5	76.7	74.3	72.3	74.7	68.8	70.1	71.1	66.6	69.7	68.0
	F1	77.9	76.8	76.6	74.5	72.7	74.6	69.0	70.5	71.1	67.3	69.2	68.4
	FAR	27.1	23.9	22.6	31.1	28.2	24.3	33.3	32.4	27.3	35.7	32.9	29.1
	AUC	78.2	78.2	77.2	75.6	73.9	75.5	70.0	71.6	71.9	68.7	69.5	69.9
RNN	ACC	77.8	79.0	79.7	74.8	75.9	76.7	71.5	72.4	73.7	69.7	70.1	72.1
	DR	78.2	79.8	81.2	75.4	76.6	77.9	71.9	72.8	75.0	71.2	70.8	73.6
	PR	79.4	81.3	80.3	75.9	75.6	76.8	70.4	72.4	74.4	71.3	71.5	74.1
	F1	78.8	80.5	80.7	75.6	76.1	77.3	71.1	72.6	74.7	71.2	71.1	73.8
	FAR	24.3	21.8	22.7	28.1	26.5	26.8	31.0	29.9	29.0	34.1	31.4	30.4
	AUC	78.7	80.5	81.9	76.2	77.1	78.9	72.9	73.4	75.7	71.8	71.7	74.1
CNN	ACC	79.2	80.2	80.5	75.6	77.8	77.5	72.4	74.4	74.2	70.7	73.0	73.2
	DR	80.3	81.6	80.6	77.4	78.8	77.7	73.5	75.0	74.8	72.4	73.9	73.4
	PR	80.1	80.5	79.1	76.4	77.4	79.0	74.2	73.8	73.8	71.9	73.4	74.6
	F1	80.2	81.0	79.8	76.9	78.1	78.3	73.8	74.4	74.3	72.1	73.6	74.0
	FAR	22.2	20.6	18.9	25.7	22.8	23.3	29.0	26.8	25.8	31.0	27.7	28.1
	AUC	81.2	82.2	81.4	78.2	79.3	78.5	74.5	75.5	75.6	73.0	74.6	74.4
GNN	ACC	81.3	82.4	85.2	78.6	79.7	82.4	75.9	77.4	79.9	74.6	76.0	78.8
	DR	82.0	82.8	85.9	79.6	80.0	83.2	76.7	77.0	80.6	75.4	75.9	79.6
	PR	80.5	81.7	86.4	80.9	79.1	83.8	75.4	76.7	80.8	76.6	75.9	79.0
	F1	81.2	82.2	86.1	80.2	79.5	83.5	76.0	76.8	80.7	76.0	75.9	79.3
	FAR	18.4	16.7	13.6	22.1	20.2	17.5	24.9	23.8	21.0	24.6	24.6	21.7
	AUC	82.9	83.5	86.4	80.4	80.7	83.8	77.7	77.5	81.4	76.2	76.4	80.1
Spectral GNN	ACC	82.5	83.5	86.3	79.6	80.9	83.9	77.3	78.7	81.2	75.6	77.4	79.8
	DR	81.5	84.4	86.7	79.7	80.8	83.5	76.9	78.0	81.7	76.1	77.7	79.5
	PR	81.9	85.4	85.7	78.4	79.9	85.0	77.9	77.7	83.2	75.3	78.6	79.0
	F1	81.7	84.9	86.2	79.0	80.3	84.2	77.4	77.8	82.4	75.7	78.1	79.2
	FAR	17.8	16.1	12.6	21.1	19.5	16.8	24.2	23.2	20.4	23.9	23.6	21.0
	AUC	82.1	85.1	87.6	80.3	81.4	84.0	77.8	79.0	82.7	76.8	78.5	80.0
Proposed SNN	ACC	82.5	83.7	86.6	80.0	81.2	83.3	77.4	78.2	80.9	76.4	77.3	79.1
	DR	82.9	84.3	86.8	79.7	81.5	84.1	77.1	78.4	81.4	75.4	77.2	80.2
	PR	82.4	85.3	87.4	80.5	82.1	83.0	76.9	78.4	80.0	74.7	76.0	79.7
	F1	82.6	84.8	87.1	80.1	81.8	83.5	77.0	78.4	80.7	75.0	76.6	79.9
	FAR	15.9	15.8	12.4	19.1	18.6	16.0	21.7	19.8	18.2	22.2	21.6	20.3
	AUC	83.4	84.9	87.8	80.7	82.2	85.0	77.9	79.0	82.0	76.3	77.7	81.1
Proposed SGNN	ACC	83.8	84.5	87.8	81.0	81.7	84.9	77.8	79.2	82.0	76.6	78.0	80.8
	DR	84.0	85.0	87.9	81.3	82.3	85.5	78.8	80.1	82.8	77.8	78.8	81.7
	PR	82.8	86.4	86.8	82.1	81.0	84.8	78.8	80.6	81.9	79.2	78.2	83.2
	F1	83.4	85.7	87.3	81.7	81.6	85.1	78.8	80.3	82.3	78.5	78.5	82.4
	FAR	14.4	14.4	12.5	17.9	18.2	14.0	20.7	18.8	17.4	21.9	21.9	17.2
	AUC	84.8	85.6	88.6	82.2	83.2	86.2	79.4	81.1	83.5	78.7	79.3	82.5
RGNN	ACC	85.2	85.5	88.9	82.2	82.7	86.2	79.1	80.7	83.0	78.2	79.0	81.8
	DR	85.4	85.0	88.4	83.2	83.3	87.2	79.1	80.1	83.0	79.2	78.7	81.1
	PR	84.1	85.7	88.3	83.2	82.0	88.6	78.9	78.8	82.1	78.0	79.3	82.6
	F1	84.7	85.3	88.3	83.2	82.6	87.9	79.0	79.4	82.5	78.6	79.0	81.8
	FAR	13.8	13.8	11.7	17.2	17.3	13.1	19.8	18.2	16.4	20.9	21.3	16.4
	AUC	86.0	85.9	89.4	83.7	84.1	87.9	79.6	81.0	83.9	79.7	79.2	82.1
Transformer GNN	ACC	85.7	86.3	89.3	82.7	83.3	86.7	79.9	81.3	83.4	79.0	79.5	82.7
	DR	85.8	87.1	89.0	82.3	83.5	87.7	79.1	82.0	83.2	78.1	80.0	81.9
	PR	87.2	85.6	90.3	81.1	84.1	88.4	80.0	82.9	83.9	77.6	80.1	82.9
	F1	86.5	86.3	89.6	81.7	83.8	88.0	79.5	82.4	83.5	77.8	80.0	82.4
	FAR	12.8	13.2	10.7	16.6	16.6	12.3	18.9	17.5	15.8	19.9	20.6	15.8
	AUC	86.6	87.6	89.7	83.0	84.0	88.2	79.9	83.0	84.1	79.0	80.5	82.5
RAGNN	ACC	88.3	88.8	91.5	83.8	85.4	88.3	81.3	82.7	85.2	80.8	80.4	83.9
	DR	87.6	88.9	91.9	85.9	86.1	89.8	81.8	82.3	85.8	80.9	81.7	85.0
	PR	86.8	89.0	91.6	85.1	85.7	89.9	81.8	81.1	84.5	80.8	82.5	85.5
	F1	87.2	88.9	91.7	85.5	85.9	89.8	81.8	81.7	85.1	80.8	82.1	85.2
	FAR	14.9	14.7	9.1	17.8	16.3	12.7	20.6	19.0	14.3	20.6	21.2	17.3
	AUC	88.4	89.5	92.9	86.9	86.6	90.7	82.8	82.8	86.7	81.8	82.6	86.8
Proposed SRGNN	ACC	**91.4**	**92.4**	**96.2**	**91.2**	**92.3**	**96.1**	**89.9**	**91.1**	**95.0**	**89.7**	**91.0**	**94.7**
	DR	**92.0**	**92.9**	**96.6**	**91.9**	**92.9**	**96.6**	**90.5**	**91.5**	**95.4**	**90.3**	**91.4**	**95.1**
	PR	**91.8**	**92.7**	**96.4**	**91.6**	**92.7**	**96.5**	**90.4**	**91.3**	**95.4**	**90.2**	**91.2**	**95.0**
	F1	**91.9**	**92.8**	**96.5**	**91.7**	**92.8**	**96.5**	**90.4**	**91.4**	**95.4**	**90.2**	**91.3**	**95.0**
	FAR	**10.2**	**9.3**	**5.7**	**10.4**	**9.5**	**5.8**	**11.7**	**10.7**	**7.0**	**12.0**	**10.9**	**7.2**
	AUC	**92.5**	**93.7**	**97.5**	**92.7**	**93.4**	**97.5**	**91.1**	**92.1**	**96.1**	**90.9**	**92.0**	**95.4**

TABLE III
ATTACK LOCALIZATION PERFORMANCE RESULTS (%)

Model	Metric	Attack Node Selection Strategy											
		Benchmark						Proposed					
		Random			Centrality Analysis			2-Simplex			2+3-Simplex		
		2869 bus	9241 bus	70000 bus	2869 bus	9241 bus	70000 bus	2869 bus	9241 bus	70000 bus	2869 bus	9241 bus	70000 bus
BDD	ACC	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0
	DR	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	PR	undef.	undef.	undef.	undef.	undef.	undef.	undef.	undef.	undef.	undef.	undef.	undef.
	F1	undef.	undef.	undef.	undef.	undef.	undef.	undef.	undef.	undef.	undef.	undef.	undef.
	FAR	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	AUC	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0
RF	ACC	68.8	66.6	67.6	66.5	63.2	63.4	62.5	59.0	60.1	60.9	56.4	57.8
	DR	70.7	67.6	69.6	67.6	64.3	66.8	62.5	59.7	61.4	61.4	59.3	61.7
	PR	69.2	66.8	71.0	68.3	65.2	67.0	63.6	60.7	62.6	59.9	59.8	60.8
	F1	69.9	67.2	70.3	67.9	64.7	66.9	63.0	60.2	62.0	60.6	59.5	61.2
	FAR	30.0	28.3	25.7	31.8	32.3	29.8	37.6	36.0	34.3	38.3	37.1	34.8
	AUC	71.3	68.3	70.7	68.4	65.1	67.7	63.6	60.7	62.0	62.3	59.9	62.5
SVM	ACC	73.4	68.9	71.7	70.6	66.8	68.4	65.7	61.8	64.5	65.2	61.3	62.6
	DR	75.2	70.5	72.2	71.5	67.9	69.5	66.3	63.3	64.9	66.5	61.0	64.2
	PR	74.5	71.9	72.8	72.1	69.4	69.0	65.1	63.6	63.5	65.0	61.2	63.0
	F1	74.8	71.2	72.5	71.8	68.6	69.2	65.7	63.4	64.2	65.7	61.1	63.6
	FAR	36.6	32.0	30.1	38.4	36.0	32.7	43.8	39.8	36.5	45.3	42.6	38.8
	AUC	76.2	71.5	73.3	72.5	68.6	70.1	67.3	64.1	65.8	67.5	62.0	65.1
FNN	ACC	81.2	80.6	80.2	77.1	77.2	77.2	73.4	73.5	73.8	72.1	71.7	73.7
	DR	82.2	81.7	80.3	79.1	77.1	79.2	73.9	75.2	75.6	71.9	73.7	73.0
	PR	80.7	81.4	79.6	78.3	77.9	80.5	74.2	76.4	74.2	70.6	74.3	72.4
	F1	81.4	81.5	79.9	78.7	77.5	79.8	74.0	75.8	74.9	71.2	74.0	72.7
	FAR	23.3	20.2	18.1	27.2	24.1	20.3	29.4	27.6	22.4	31.2	28.1	24.9
	AUC	83.2	82.5	81.3	80.0	77.8	79.8	74.8	76.0	76.4	72.6	74.7	73.9
RNN	ACC	82.5	83.0	84.4	78.7	79.9	80.6	75.5	76.1	78.1	74.6	74.6	76.7
	DR	82.0	84.4	85.1	79.1	81.3	81.6	76.3	76.7	78.9	75.9	75.5	78.1
	PR	81.9	84.1	85.7	79.3	80.7	81.5	77.1	76.6	78.2	75.4	75.9	76.6
	F1	81.9	84.2	85.4	79.2	81.0	81.5	76.7	76.6	78.5	75.6	75.7	77.3
	FAR	19.8	17.7	18.5	24.1	22.6	22.7	26.4	25.8	24.3	29.9	27.2	26.0
	AUC	82.7	85.3	86.2	79.8	82.2	82.5	77.0	77.3	79.9	76.7	76.3	79.0
CNN	ACC	83.1	84.3	85.4	79.4	81.8	82.4	76.6	79.1	78.3	74.6	77.0	77.1
	DR	85.2	85.9	84.7	81.5	83.1	82.3	78.2	79.5	79.6	77.0	78.4	77.1
	PR	84.0	86.4	83.8	82.7	83.9	83.5	78.9	79.6	80.6	78.4	76.9	77.6
	F1	84.6	86.1	84.2	82.1	83.5	82.9	78.5	79.5	80.1	77.7	77.6	77.3
	FAR	17.3	16.1	14.5	22.0	19.0	19.5	24.7	23.1	21.5	26.7	23.5	23.2
	AUC	86.0	86.5	85.8	82.6	84.1	83.2	78.9	80.6	80.7	78.0	79.0	77.7
GNN	ACC	83.2	84.2	86.9	80.4	81.7	84.3	77.7	79.2	81.6	76.3	78.2	80.8
	DR	84.2	85.0	88.0	81.8	81.9	85.1	78.5	79.2	82.8	77.1	77.9	81.7
	PR	85.0	83.7	88.8	80.8	81.0	84.6	79.9	79.1	81.8	75.7	78.7	80.8
	F1	84.6	84.3	88.4	81.3	81.4	84.8	79.2	79.1	82.3	76.4	78.3	81.2
	FAR	16.5	14.8	11.4	20.1	18.2	15.4	22.8	22.0	18.8	22.6	22.8	19.9
	AUC	85.1	85.8	88.9	82.6	82.9	86.1	79.3	80.2	83.8	77.7	79.0	82.4
Spectral GNN	ACC	84.2	85.3	88.0	81.6	82.8	85.6	79.3	80.6	83.0	77.5	79.4	82.0
	DR	83.4	86.2	88.4	81.5	82.7	85.3	78.6	80.1	83.8	78.2	79.7	81.3
	PR	84.0	84.9	88.9	82.8	81.8	86.0	79.3	80.1	83.6	78.6	79.6	80.3
	F1	83.7	85.5	88.6	82.1	82.2	85.6	78.9	80.1	83.7	78.4	79.6	80.8
	FAR	16.1	14.4	10.6	19.0	17.4	15.0	22.1	21.0	18.6	22.0	21.8	19.0
	AUC	84.1	86.9	89.0	82.5	83.7	86.2	79.6	80.9	84.4	78.9	80.3	82.1
Proposed SNN	ACC	84.7	85.8	88.4	81.7	82.9	85.2	79.2	79.9	82.7	78.5	79.4	81.0
	DR	84.7	86.3	89.0	81.7	83.4	86.2	78.9	80.4	83.4	77.3	79.1	82.0
	PR	85.6	85.1	89.3	81.1	84.1	87.7	80.3	79.6	84.0	76.5	78.3	82.5
	F1	85.1	85.7	89.1	81.4	83.7	86.9	79.6	80.0	83.7	76.9	78.7	82.2
	FAR	14.1	14.0	10.6	17.0	16.4	14.3	19.7	17.6	16.3	20.0	19.5	18.4
	AUC	85.7	87.0	90.1	82.7	84.5	87.2	80.0	81.1	84.0	78.0	80.1	82.8
Proposed SGNN	ACC	85.5	86.5	89.6	82.7	83.4	87.0	79.9	81.0	84.0	78.8	80.0	82.5
	DR	86.0	86.8	89.6	83.5	84.4	87.6	81.0	82.1	84.9	80.0	81.0	83.4
	PR	87.2	87.6	90.8	83.1	85.8	88.4	82.5	83.4	83.7	78.6	82.0	83.8
	F1	86.6	87.2	90.2	83.3	85.1	88.0	81.7	82.7	84.3	79.3	81.5	83.6
	FAR	12.2	12.7	10.3	15.8	16.4	12.0	18.8	16.6	15.5	19.7	19.7	15.3
	AUC	86.6	87.9	90.7	84.5	85.1	88.5	82.0	83.2	85.9	80.9	81.9	84.2
RGNN	ACC	87.2	87.5	91.0	84.3	84.8	87.9	81.3	82.6	84.7	79.9	80.8	83.9
	DR	87.4	86.9	90.6	85.3	85.4	89.2	81.2	82.0	85.0	81.2	80.7	82.8
	PR	85.9	85.7	91.3	84.0	85.4	88.4	80.1	82.3	86.2	80.3	81.6	83.6
	F1	86.6	86.3	90.9	84.6	85.4	88.8	80.6	82.1	85.6	80.7	81.1	83.2
	FAR	12.1	12.0	10.0	15.2	15.6	11.2	17.9	16.4	14.3	19.1	19.1	14.6
	AUC	88.4	87.6	91.7	86.3	86.1	89.8	82.0	83.1	85.7	82.3	81.8	83.9
Transformer GNN	ACC	87.5	88.3	91.0	84.9	85.2	88.5	81.6	83.2	85.6	80.8	81.7	84.4
	DR	87.9	89.2	91.2	84.2	85.7	89.5	80.9	83.7	84.9	79.9	82.1	84.1
	PR	88.7	89.1	92.7	84.0	84.9	88.2	80.8	82.3	86.0	81.3	83.0	83.3
	F1	88.3	89.1	91.9	84.1	85.3	88.8	80.8	83.0	85.4	80.6	82.5	83.7
	FAR	11.1	11.1	8.5	14.6	14.8	10.3	17.2	15.4	13.6	18.0	18.8	13.7
	AUC	88.7	90.2	91.8	84.9	86.6	90.1	81.5	84.8	85.6	80.9	82.9	85.2
RAGNN	ACC	90.5	90.9	93.5	85.7	87.2	90.0	83.2	84.5	87.4	82.7	82.6	86.0
	DR	89.5	90.7	94.0	87.8	88.3	91.7	83.6	84.5	87.6	83.1	83.8	86.9
	PR	90.7	91.6	92.7	87.5	89.3	93.0	85.1	85.1	86.3	82.2	82.7	85.9
	F1	90.1	91.1	93.3	87.6	88.8	92.3	84.3	84.8	86.9	82.6	83.2	86.4
	FAR	13.2	12.6	7.1	15.6	14.5	10.9	18.5	17.3	12.3	18.9	19.2	15.4
	AUC	90.5	91.8	94.8	88.8	89.2	92.6	84.5	85.1	88.2	84.1	84.7	87.8
Proposed SRGNN	ACC	**92.4**	**93.5**	**96.3**	**92.1**	**93.4**	**96.6**	**90.9**	**92.1**	**96.0**	**90.6**	**92.0**	**95.7**
	DR	**93.2**	**93.9**	**96.9**	**93.1**	**93.8**	**96.8**	**91.4**	**92.7**	**96.5**	**91.5**	**92.3**	**96.2**
	PR	**93.8**	**92.3**	**98.6**	**93.3**	**92.9**	**96.3**	**91.0**	**91.4**	**94.4**	**90.0**	**92.1**	**95.4**
	F1	**93.4**	**92.8**	**97.9**	**92.7**	**93.2**	**97.0**	**91.3**	**91.6**	**95.1**	**90.6**	**92.2**	**95.4**
	FAR	**9.3**	**8.1**	**4.8**	**9.2**	**8.4**	**4.9**	**10.6**	**9.6**	**6.0**	**10.7**	**9.7**	**6.2**
	AUC	**94.0**	**94.1**	**98.3**	**92.9**	**94.3**	**98.4**	**92.8**	**92.4**	**96.6**	**92.3**	**93.3**	**96.2**

degrades across all settings. For benchmark models, moving from random to 2+3-simplex attacks results in a degradation of 7.5% in DR and an increase of 7.8% in FAR. Restricting targeting to 2-simplices alone yields a 6.3% drop in DR and a 6.5% increase in FAR. In contrast, the proposed models remain substantially more robust on average: between random and 2+3-simplex attacks, their average drops are limited to 5.0% in DR and 5.0% in FAR, indicating stronger resilience to structured, topology-aware attack placement.

b) Proposed vs Centrality in Detection: Compared to random selection, centrality-based targeting causes smaller but still notable degradation for benchmark models, with DR decreasing by 2.7% and FAR increasing by 3.3%. The proposed models follow the same trend on average, exhibiting a comparable 2.8–2.9% performance shift relative to random selection. Additionally, when moving from centrality-based to 2-simplex targeting, benchmark performance degrades further by 3.6% in DR and 3.2% in FAR, confirming that increasingly structured (simplex-aware) attack strategies systematically worsen detection difficulty. Overall, these results reinforce that topology-aware attack placement is consistently more harmful than centrality-based selection, while the proposed models maintain stronger average robustness across attack strategies.

D. Localization Task

The localization task refers to identifying which specific node is under attack. Table III summarizes the localization performance of all evaluated models across different attack types, node selection strategies, and grid sizes.

1) Model Performance in Localization Task: The proposed SRGNN consistently surpasses all benchmark models by 8–35% in DR against complex attacks. All proposed models (SNN, SGNN, and SRGNN) outperform benchmarks by 2.6–87.0% in DR as follows.

a) Proposed vs BDD in Localization: The proposed models outperform the BDD by 87.0%, highlighting that a lightweight distribution-based detector is insufficient for localizing stealthy, topology-aware attacks. The BDD performance remains near-zero in DR across all evaluated settings, as it does not exploit network structure or temporal evolution to attribute anomalies to specific buses.

b) Proposed vs ML-based Models in Localization: The proposed models outperform the shallow ML-based models by 20.9% in DR and 22.0% in FAR. The proposed models outperform the DL-based models by 7.8% in DR and 9.8% in FAR. Relative to graph-based models, the proposed models achieve improvements of 2.6% in DR and a 2.7% in FAR. These improvements underscore the value of higher-order topological reasoning in enhancing attack localization in smart grid systems.

c) Proposed Models in Localization Performance: The SRGNN outperforms SGNN and SNN by 9.8–11.3% in DR and 7.3–8.4% in FAR. These improvements, once again, highlight the additive value of each architectural component. The GCN branch in SGNN enhances spatial resolution by modeling pairwise interactions, while the recurrent branch in SRGNN captures temporal correlations across snapshots. By fusing both, SRGNN delivers the most robust localization performance under diverse and evolving attack scenarios.

2) Impact of Attack Strategy on Localization Task: We analyze how increasingly topology-aware attack strategies affect localization performance, comparing the proposed models against benchmark methods under random, centrality-based, and simplex-aware node targeting.

a) Proposed vs Random in Localization: As node selection for attacked nodes becomes more topology-aware, localization performance degrades across all settings. For benchmark models, moving from random to 2+3-simplex attacks yields a degradation of 6.8% in DR and an increase of 7.1% in FAR. Targeting only 2-simplices leads to a 5.8% drop in DR and a 5.8% increase in FAR relative to random selection. In contrast, the proposed models remain more robust on average: between random and 2+3-simplex attacks, their average drops are approximately 4.8% in DR and 4.8% in FAR, indicating stronger resilience to structured, simplex-aware attack placement.

b) Proposed vs Centrality in Localization: Compared to random selection, centrality-based targeting causes smaller but still notable degradation for benchmark models, with DR decreasing by 2.5% and FAR increasing by 3.0%. The proposed models follow a similar average trend relative to random selection, exhibiting 2.3–3.0% performance shifts. Moreover, moving from centrality-based to 2-simplex targeting further worsens benchmark localization by 3.3% in DR and 2.8% in FAR, confirming that increasingly structured (simplex-aware) attack strategies systematically increase localization difficulty. Overall, these results reinforce that topology-aware attack placement is consistently more harmful than centrality-based selection, while the proposed models maintain stronger average robustness across attack strategies.

E. Scalability

The benchmark and proposed models offer enhanced scalability performance as system size increases from 2,869 to 9,241 to 70,000. Specifically, in terms of DR, ML-based benchmarks offer enhanced performance by 1.8–1.9% for the detection and 1.8–2.0% for localization tasks. Similarly, the proposed models (SNN, SGNN, and SRGNN) provide enhanced performance of 2.8–4.8% for detection and 2.8–4.5% for localization.

F. Additional Performance Analysis

This section provides additional analysis in terms of ROC curves, performance against noise, significance testing, and computational complexity. To avoid cluttered plots, we provide analysis on the proposed SRGNN compared to representative benchmarks, namely SVM, CNN, and RAGNN, which correspond to the best-performing models among the shallow ML, DL, and graph-based benchmark families, respectively.

1) ROC Curves: The reported AUC in Tables II and III confirms the improved separability achieved by SRGNN, which is validated further in Fig. 5 by plotting the ROC curves of SRGNN compared to the representative benchmarks. Specifically, SRGNN maintains consistently high AUC (90.9%

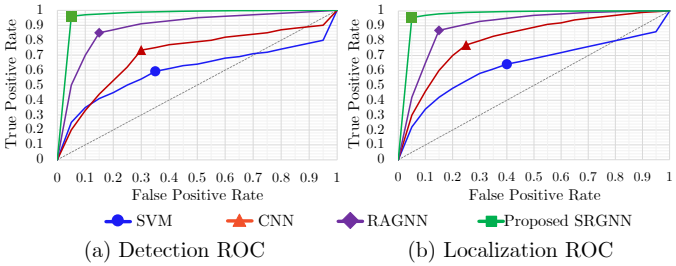


Fig. 5. ROC curves for attack detection and localization.

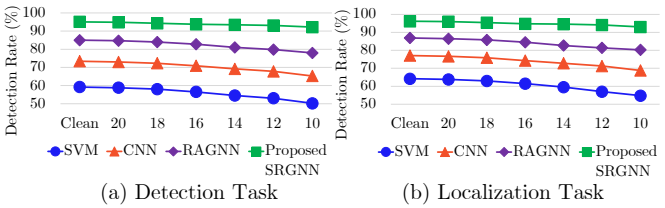


Fig. 6. Detection performance against different levels of noise.

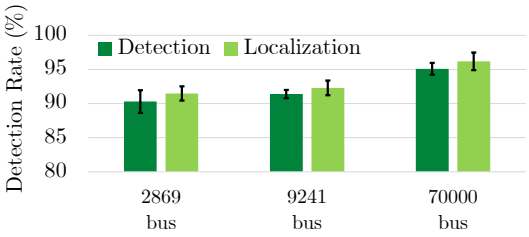


Fig. 7. Mean DR of SRGNN over 10 runs for detection and localization.

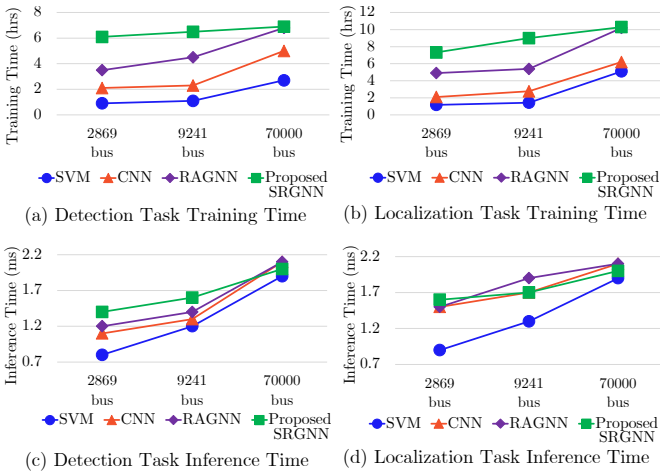
- 97.5%) across all grid sizes and node-selection strategies, indicating strong robustness under varying decision thresholds. In comparison, the representative benchmarks attain lower detection AUC (by 18.0% compared to SRGNN), highlighting that SRGNN preserves a more favorable TP/FP trade-off. Fig. 5(a) visualizes such behavior, where SRGNN consistently achieves higher TP rates at the same FP rates compared to the representative benchmarks, confirming a superior threshold-independent detection trade-off. AUC similarly provides a threshold-independent view of attack localization quality and aligns with the trends observed in the reported metrics. SRGNN achieves consistently high localization AUC (92.3% to 98.4%), demonstrating strong ranking and separability of attacked buses even under simplex-based node selection. In contrast, the representative benchmarks yield lower localization AUC (by 14.9% compared to SRGNN), confirming that SRGNN sustains a superior ROC trade-off for localization. As shown in Fig. 5(b), compared to the representative benchmarks, SRGNN attains higher TP rates for the same FP rates, indicating a more reliable ranking of attacked buses across decision thresholds.

2) Performance Against Noise: To assess robustness under noisy measurements and data variability, we conduct a signal-to-noise ratio (SNR) sensitivity study by injecting additive white Gaussian noise (AWGN) [32], [33] into $[P(v), Q(v)]$. The SNR is defined as the ratio in decibels (dB) of the clean measurement signal power and the injected noise power. Fig. 6(a) and (b) plots the detection and localization DR, respectively, on the 70,000-bus system under the most challenging 2+3-simplex attack setting as SNR decreases from noise-free (clean) to 10 dB (noisy) conditions. As SNR decreases (i.e, with more noise), the representative benchmarks exhibit notable DR degradation of 7.0–9.0% and 6.6–9.4% in the detection and localization tasks, respectively. In contrast, SRGNN maintains robust performance, with a minimal DR degradation of 2.0–3.0% for detection and localization at 10 dB. As a result, SRGNN preserves a substantial robustness margin over the representative benchmarks across all evaluated SNR levels due to its joint modeling of higher-order spatial dependencies through simplicial message passing and temporal consistency through the recurrent module, which together suppress random AWGN perturbations while preserving coordinated attack-induced deviations across buses and time.

3) Significance Testing and Variance Across Runs: To quantify run-to-run variability and confirm that the observed values are not due to randomness, we run each experiment 10 times using different random seeds and perform cross-validation. We then obtain the mean and standard deviation (SD) of DR. Fig. 7 summarizes SRGNN's DR value stability across grid sizes for the detection and localization tasks. For detection, SRGNN achieves $90.3\%\pm1.66\%$, $91.41\%\pm0.60\%$, and $95.11\%\pm0.87\%$ for the 2,869-bus, 9,241-bus, and 70,000-bus systems, respectively. For localization, SRGNN achieves $91.5\% \pm 1.05\%$, $92.3\% \pm 1.08\%$, and $96.2\% \pm 1.28\%$, respectively. These results indicate that SRGNN maintains consistently high DR with low dispersion across independent runs. Beyond variance reporting, we perform paired statistical significance testing between SRGNN and the representative benchmarks on a per-setting basis using (i) a paired t-test and (ii) a nonparametric Wilcoxon signed-rank test [34]. To account for multiple comparisons across settings, we apply the Holm method to the Wilcoxon p-values. It turns out that our results are statistically significant since $p < 0.05$ [34]. Specifically, across all settings, the paired t-test yields p-values in the range $[3.63 \times 10^{-11}, 5.87 \times 10^{-3}]$, while the Wilcoxon test yields p-values in the range $[1.95 \times 10^{-3}, 9.77 \times 10^{-3}]$. After applying the Holm method, the Wilcoxon p-values $p = 2.34 \times 10^{-2}$, maintaining $p < 0.05$. Therefore, SRGNN's improvements over the representative benchmarks are statistically significant under both parametric and nonparametric tests, corroborating the consistent robust performance gains.

4) Computational Complexity: All experiments are executed on a workstation equipped with an NVIDIA RTX 4090 hardware accelerator, 24 GB VRAM, and 64 GB system RAM. Next, we analyze the training time per model in hours (hr), inference (testing) time per sample in milliseconds (ms), and the graphics processing unit (GPU) peak memory in gigabytes (GB) for the representative benchmarks and proposed SRGNN for the detection and localization tasks.

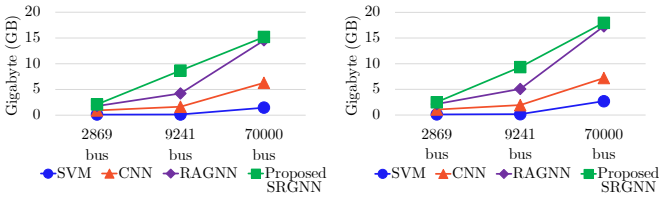
a) Training Time: Fig. 8(a) and (b) show the offline training time trends in hours (hr) of the detection and localization tasks, respectively, as system size increases. Across all evaluated models, training time increases with the number of buses due to the growth in processed feature tensors and message-passing operations. For the representative



(a) Detection Task Training Time (b) Localization Task Training Time

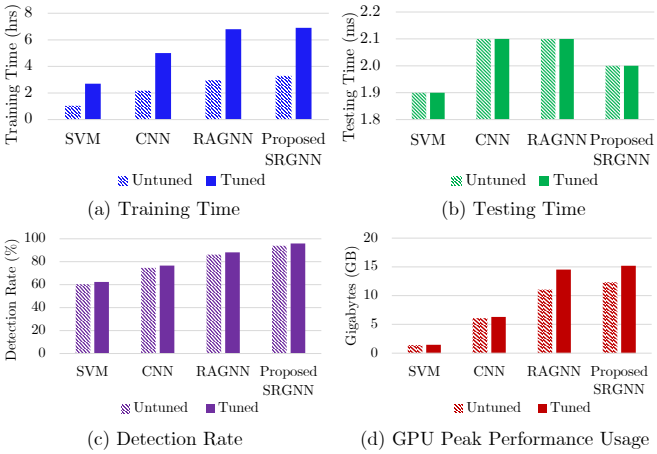
(c) Detection Task Inference Time (d) Localization Task Inference Time

Fig. 8. Training and inference runtimes.



(a) Detection Peak GPU Memory (b) Localization Peak GPU Memory

Fig. 9. Peak GPU memory usage for detection and localization.



(a) Training Time (b) Testing Time

(c) Detection Rate (d) GPU Peak Performance Usage

Fig. 10. Impact of hyperparameter tuning.

benchmarks, detection training time increases by 94%–200%, and localization training time increases by 108%–336%. In contrast, the proposed SRGNN exhibits a smoother, linear growth trend as system size increases, with training time increasing by 13% for detection and 41% for localization. All models require additional offline training time with larger systems. However, this cost is incurred offline and typically scheduled periodically by system operators without disrupting real-time grid operation [1].

b) Inference Time: Figs. 8(c) and (d) report per-sample inference latency in milliseconds (ms) for detection and localization, respectively, for the representative benchmarks compared to the proposed SRGNN. Inference time increases moderately with system size across all models by 40–85%. For the proposed SRGNN, inference latency remains within 1.40–2.00 ms for detection and 1.60–2.00 ms for localization. These values remain comparable to baselines while providing substantially improved robustness. From an operational cyber security perspective, practical intrusion detection systems (IDSs) typically require sub-2 ms inference latency for real-time monitoring [1]; the reported SRGNN inference times satisfy this requirement across all evaluated grid sizes.

c) GPU Usage and Complexity: Fig. 9(a) and (b) report the peak GPU memory performance in gigabytes (GB) for detection and localization, respectively, for the representative benchmarks compared to the proposed SRGNN. As system size increases from 2,869 to 70,000 buses, all models exhibit a consistent rise in peak GPU memory due to the growth in intermediate feature tensors and neighborhood aggregation operations. Across the representative benchmarks, peak GPU memory increases by 7–10× across grid sizes for both detection and localization. Under the same settings, the peak GPU memory of the proposed SRGNN increases by 7× for both detection and localization, remaining within the 24 GB VRAM budget of the RTX 4090. Moreover, SRGNN offers a consistent linear increase as system size increases, unlike the representative benchmarks that offer superlinear growth. Overall, these trends are consistent with SRGNN's higher-order message passing, where the additional simplicial branches (via R_2 and R_3) introduce extra intermediate activations that increase peak memory, while remaining practical at the largest evaluated system size.

d) Hyperparameter Tuning Impact on Performance: Fig. 10 summarizes the impact of hyperparameter tuning from four perspectives. Specifically, Fig. 10(a) showcases training time, (b) inference time, (c) DR, and (d) peak GPU memory usage. Fig. 10(c) shows that tuning improves DR across all investigated models by 2–3×. Fig. 10(a) shows that tuning increases offline training time by 2.1–2.7× and 2.1–2.4× for the representative benchmarks and SRGNN, respectively. Such overhead is incurred offline and can be scheduled periodically without disrupting real-time operation. In contrast to training time, Fig. 10(b) shows that tuning has a negligible effect on deployment latency as inference time remains within ±0.10 ms for all models, and SRGNN remains within the sub-2 ms latency requirement [1]. Fig. 10(d) shows that tuning increases peak GPU memory usage by 1.2× due to larger intermediate activations, but remains within the 24 GB VRAM budget of the RTX 4090 at the largest evaluated system size. Overall, Fig. 10 shows that tuning yields improved detection performance without increasing inference latency, while the added training and memory costs remain practical for offline operation. For example, reducing the number of layers from 6 to 2 reduces DR by 2% and the offline training time by 50%, while the online inference time remains the same.

VI. Conclusion

In this paper, we introduced a novel SRGNN framework for the detection and localization of cyber attacks in large-scale

power systems. We constructed large, realistic spatiotemporal datasets based on the 2,869-bus, 9,241-bus, and 70,000-bus systems to evaluate performance under diverse attack types and node selection strategies, which enables assessment over a $24\times$ increase in system size, with SRGNN improving detection and localization by up to 4.8% and 4.3%, respectively. In addition to benchmarking existing strategies, we proposed simplicial-based attacked node selection methods that target nodes embedded in higher-order simplicial complexes, degrading benchmark model performance by 3–9% compared to random and 4–6% compared to centrality-based attacks, enabling more challenging evaluation scenarios. Unlike conventional models that rely solely on pairwise interactions, our SRGNN architecture jointly captures higher-order topological structures and temporal dynamics by integrating simplicial complexes with recurrent modeling, thus outperforming benchmark models by 9–39% and 8–35%, in attack detection and localization, respectively, in DR. Future work includes conducting hardware-in-the-loop validation as well as extending the proposed framework to incorporate measurement variability, including distributed energy resource-driven intermittency and associated high-frequency fluctuations.

REFERENCES

[1] A. Takiddin *et al.*, "Spatio-temporal graph-based generation and detection of adversarial false data injection evasion attacks in smart grids," *IEEE Trans. on Artificial Intelligence*, vol. 5, no. 12, pp. 6601–6616, Dec. 2024.

[2] A. Takiddin *et al.*, "Robust data-driven detection of electricity theft adversarial evasion attacks in smart grids," *IEEE Trans. on Smart Grid*, vol. 14, no. 1, pp. 663–676, Jan. 2023.

[3] A. Takiddin *et al.*, "Resilience of data-driven cyberattack detection systems in smart power grids," in *2024 32nd European Signal Processing Conf. (EUSIPCO)*, Lyon, France, 26–30 Aug. 2024, pp. 1992–1996.

[4] C. Huber, "Chinese Hackers Targeted Texas Power grid, Hawaii Water Utility, Other Critical Infrastructure," [Online; Accessed Jan. 2026]. [Online]. Available: https://tinyurl.com/neywnbzt

[5] M. Esmalifalak *et al.*, "Detecting stealthy false data injection using machine learning in smart grid," *IEEE Sys. J.*, vol. 11, no. 3, pp. 1644–1652, Sept. 2017.

[6] X. Lu *et al.*, "False data injection attack location detection based on classification method in smart grid," in *2020 2nd Intl. Conf. on Artificial Intelligence and Advanced Manufacture (AIAM)*, Manchester, United Kingdom, 15–17 Oct. 2020, pp. 133–139.

[7] Y. Farrukh *et al.*, "A sequential supervised machine learning approach for cyber attack detection in a smart grid system," in *53rd North American Power Symposium (NAPS)*, College Station, TX, USA, 14–16 Nov. 2021.

[8] D. Xue *et al.*, "Detection of false data injection attacks in smart grid utilizing elm-based ocon framework," *IEEE Access*, vol. 7, pp. 39 444–39 454, March 2019.

[9] E. Ferragut *et al.*, "Real-time cyber-physical false data attack detection in smart grids using neural networks," in *2017 Intl. Conf. on Computational Science and Computational Intelligence (CSCI)*, Las Vegas, NV, USA, 14–16 Dec. 2017, pp. 1–6.

[10] Y. Zhang *et al.*, "Detecting false data injection attacks in smart grids: A semi-supervised deep learning approach," *IEEE Trans. on Smart Grid*, vol. 12, no. 1, pp. 623–634, Jan. 2021.

[11] Y. Wang *et al.*, "Kfrnn: An effective false data injection attack detection in smart grid based on kalman filter and recurrent neural network," *IEEE Internet of Things J.*, vol. 9, no. 9, pp. 6893–6904, May 2022.

[14] A. Takiddin *et al.*, "Deep autoencoder-based anomaly detection of electricity theft cyberattacks in smart grids," *IEEE Sys. J.*, vol. 16, no. 3, pp. 4106–4118, Sept. 2022.

[12] S. Wang *et al.*, "Locational detection of the false data injection attack in a smart grid: A multilabel classification approach," *IEEE Internet of Things J.*, vol. 7, no. 9, pp. 8218–8227, Sept. 2020.

[13] G. Zhang *et al.*, "Spatio-temporal correlation-based false data injection attack detection using deep convolutional neural network," *IEEE Trans. on Smart Grid*, vol. 13, no. 1, pp. 750–761, Jan. 2022.

[15] G. Morgenstern *et al.*, "Protection against graph-based false data injection attacks on power systems," *IEEE Trans. on Control of Netw. Sys.*, vol. 11, no. 4, pp. 1924–1938, Dec. 2024.

[16] O. Boyaci *et al.*, "Graph neural networks based detection of stealth false data injection attacks in smart grids," *IEEE Sys. J.*, vol. 16, no. 2, pp. 2946–2957, June 2022.

[17] O. Boyaci *et al.*, "Joint detection and localization of stealth false data injection attacks in smart grids using graph neural networks," *IEEE Trans. on Smart Grid*, vol. 13, no. 1, pp. 807–819, Jan. 2022.

[18] W. Xia *et al.*, "Locational detection of false data injection attacks in smart grids: A graph convolutional attention network approach," *IEEE Internet of Things J.*, vol. 11, no. 6, pp. 9324–9335, March 2024.

[19] A. Takiddin *et al.*, "A graph neural network multi-task learning-based approach for detection and localization of cyberattacks in smart grids," in *2023 IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Rhodes Island, Greece, 4–9 June 2023, pp. 1–5.

[20] A. Takiddin *et al.*, "Generalized graph neural network-based detection of false data injection attacks in smart grids," *IEEE Trans. on Emerging Topics in Computational Intelligence*, vol. 7, no. 3, pp. 618–632, June 2023.

[21] A. Takiddin *et al.*, "Robust graph autoencoder-based detection of false data injection attacks against data poisoning in smart grids," *IEEE Trans. on Artificial Intelligence*, vol. 5, no. 3, pp. 1287–1300, March 2024.

[22] Zhang, Yu *et al.*, "A Graph Transformer-Driven Approach for Network Robustness Learning," *IEEE Trans. on Circuits and Sys. I: Regular Papers*, vol. 71, no. 5, pp. 1992–2005, May 2024.

[23] R. Atat *et al.*, "Graphon neural networks-based detection of false data injection attacks in dynamic spatio-temporal power systems," *IEEE Open Access J. of Power and Energy*, vol. 12, pp. 24–35, Jan. 2025.

[24] S. Barbarossa and S. Sardellitti, "Topological signal processing over simplicial complexes," *IEEE Trans. on Signal Processing*, vol. 68, pp. 2992–3007, March 2020.

[25] S. Ebli *et al.*, "Simplicial neural networks," in *NeurIPS Workshop on Topological Data Analysis (TDA) & Beyond*, Virtual, Dec. 2020.

[26] M. Yang *et al.*, "Simplicial convolutional neural networks," in *2022 IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Singapore, 22–27 May 2022, pp. 8847–8851.

[27] A. Birchfield *et al.*, "Grid structural characteristics as validation criteria for synthetic networks," *IEEE Trans. on Power Sys.*, vol. 32, no. 4, pp. 3258–3265, July 2017.

[28] A. Birchfield, T. Xu, and T. J. Overbye, "Power flow convergence and reactive power planning in the creation of large synthetic grids," *IEEE Trans. on Power Sys.*, vol. 33, no. 6, pp. 6667–6678, Nov. 2018.

[29] Wu, Jie *et al.*, "Importance Assessment of Distribution Network Nodes Based on an Improved MBCC-HITS Algorithm," *Algorithms*, vol. 18, no. 9, p. 589, Sept. 2025.

[30] C. Josz *et al.*, "Ac power flow data in matpower and qcqp format: itesla, rte snapshots, and pegase," *arXiv preprint arXiv:1603.01533*, 2016.

[31] S. Fliscounakis *et al.*, "Contingency ranking with respect to overloads in very large power systems taking into account uncertainty, preventive and corrective actions," *IEEE Trans. on Power Sys.*, vol. 28, no. 4, pp. 4909–4917, Nov. 2013.

[32] Fahim, S. R. *et al.*, "Graph Neural Network-Based Approach for Detecting False Data Injection Attacks on Voltage Stability," *IEEE Open Access J. of Power and Energy*, Jan. 2025.

[33] Chen, K. *et al.*, "Detection and classification of transmission line faults based on unsupervised feature learning and convolutional sparse autoencoder," *IEEE Trans. on Smart Grid*, vol. 9, no. 3, pp. 1748–1758, May 2018.

[34] Benjamin, Daniel J. *et al.*, "Redefine statistical significance," *Nature Human Behaviour*, vol. 2, pp. 6–10, Sept. 2018.