Individual Project
Final Report

F130260

Electronic and Computer Systems Engineering

Your module code here (e.g. 23WSC325)

Cyber-Physical Resilience in Microgrid Systems: A Simulation Framework with OMNET++ and Pandapower

Abstract

Microgrids promise enhanced resilience and operational flexibility by integrating distributed energy resources with local loads, yet their reliance on digital communication renders them vulnerable to cyber-physical attacks. This dissertation presents a modular co-simulation framework that bridges packet-level network emulation in OMNeT++ with power-flow analysis in Pandapower via ZeroMQ, enabling realistic, closed-loop evaluation of microgrid control under attack. A supervisory controller exchanges measurement and actuation messages at fixed intervals while Pandapower applies a noisy diurnal load profile and performs robust two-stage power-flow solves. Two lightweight anomaly detectors were implemented: an Isolation Forest trained on multivariate system metrics to flag stealthy packet-tampering in real time, and a latency-plateau heuristic to detect denial-of-service stalls when solver response times freeze. In case studies, the Isolation Forest immediately identified an arbitrary bogus set-point injection, and the plateau detector pinpointed the exact onset of service interruption, both with minimal false alarms. Although demonstrated on a small radial feeder, the framework readily extends to larger, meshed topologies and industry-standard SCADA protocols. By combining high-fidelity cyber-physical modeling with rapid, embedded detection, this work advances defence-in-depth strategies for securing next-generation microgrids against evolving cyber threats and underpins future resilience-enhancement measures.

Contents

1. Introduction

Context and Motivation

Distribution networks are undergoing a profound transition from unidirectional power delivery to bidirectional energy exchange driven by rooftop photovoltaics, community wind turbines and battery-energy-storage systems. When these distributed energy resources (DERs) are grouped into a microgrid—an electrically bounded domain able to island from, or reconnect to, the main grid—they improve reliability, enhance power quality and facilitate the United Kingdom's legally binding net-zero target for 2050 [1]. Analysts forecast that global microgrid capacity will exceed 20 GW by 2027, with more than 1 000 UK sites already in operation or advanced planning [2].

Achieving the promised resilience, however, depends on fast and accurate cyber-physical coordination. Supervisory controllers, smart inverters and protection relays now exchange hundreds of packets per second over Ethernet, cellular or power-line-carrier links. This digitalisation widens the attack surface: man-in-the-middle (MiTM) manipulation can bias a single set-point, while distributed-denial-of-service (DDoS) floods can stall entire control channels—both of which risk voltage excursions, protection mal-operations and loss of load [3]. The 2015 Ukrainian grid incident and the 2021 Colonial Pipeline breach illustrate how cyber compromise of operational technology can escalate rapidly into physical disruption [4].

Existing research often analyses the electrical and communication domains in isolation, or else assumes fixed, deterministic delays. Such simplifications obscure the way packet loss, retransmissions and encryption overhead cascade into transformer loading, inverter thermal limits and reserve margins. Moreover, few studies embed real-time anomaly detection capable of flagging abnormal operating states before they threaten safety. The absence of an open, reproducible test-bed that unites power-flow solvers, packet-level networking and machine-learning defences thus constrains progress in microgrid resilience engineering.

Motivated by this gap, the present project develops an open-source co-simulation framework that couples Pandapower's time-domain power-flow engine with OMNeT++'s packet-level network model through encrypted TCP messaging. The platform allows researchers to inject realistic MiTM and DDoS scenarios, to observe how variable latency propagates into physical stress, and to evaluate lightweight anomaly-detection algorithms—such as Isolation Forests—under closed-loop conditions. By doing so, the work advances both the methodological toolkit and the practical understanding required to safeguard the next generation of cyber-enabled microgrids.

Problem Statement

Despite growing consensus that cyber compromise can erode microgrid resilience, existing assessment methods remain inadequate in three respects. **First**, most studies decouple the electrical and communication domains—treating the network as a fixed-delay black box—and therefore overlook stochastic latency, jitter and retransmission effects that occur on real IP-based infrastructures [3]. **Second**, open-source test-beds that can inject realistic threat vectors—such as man-in-the-middle falsification or distributed-denial-of-service floods—while simultaneously monitoring their impact on power-flow convergence, voltage profiles and transformer loading are scarce [4]. **Third**, real-time anomaly-detection schemes are typically evaluated offline; consequently, the community lacks evidence on how quickly a controller embedded in the loop can flag and mitigate an unfolding cyber intrusion [5].

Because of these limitations, utilities and equipment vendors have no reproducible, end-to-end framework for quantifying how communication-layer disturbances propagate into physical stress, nor for benchmarking lightweight machine-learning defences under realistic operating conditions. Bridging this methodological gap is essential if microgrids are to fulfil their promise of delivering reliable, flexible and cyber-secure power within future distribution networks.

Objectives and Contributions

The overarching goal of this work is to furnish the microgrid-research community with an open, reproducible test-bed that links communication-layer behaviour to physical-layer consequences under realistic cyber-attack conditions. To that end, the project pursues three tightly coupled objectives.

1. **Design an integrated cyber-physical co-simulation framework.** The platform couples OMNeT++—a packet-level network simulator capable of modelling variable latency, retransmissions and hostile traffic [6]—with Pandapower, an open-source power-flow engine widely adopted for distribution-system studies [5]. The two domains exchange messages over encrypted ZeroMQ/TCP sockets, preserving event chronology while allowing each process to run at its own time resolution. A supervisory controller implemented in OMNeT++ executes a PID algorithm; every five seconds it requests fresh telemetry, receives Pandapower's load-flow results and transmits an updated DER set-point, thereby closing the loop in real time.

2. **Quantify the impact of representative cyber attacks on microgrid performance.** The framework injects both man-in-the-middle falsification and distributed-denial-of-service floods—two attack classes repeatedly identified as high-risk for operational technology [3], [4]. By observing voltage profiles, transformer loading and DER dispatch under varying network conditions, the study reveals how milliseconds-scale packet aberrations propagate into physical stress.

3. **Embed machine-learning anomaly detection for proactive defence.** A seven-feature Isolation Forest, trained solely on benign operation, runs inside the Pandapower process and scores each measurement vector online, while an auxiliary latency-plateau rule flags solver stalls symptomatic of volumetric attacks. This real-time layer enhances situational awareness and demonstrates how lightweight edge analytics can supplement cryptographic protections [7].

Contributions.

- (i) An openly documented OMNeT++–Pandapower co-simulation tool-chain, complete with encryption hooks and sub-second synchronisation;
- (ii) A systematic evaluation of microgrid voltage and loading response to MiTM and DoS scenarios;
- (iii) An extensible anomaly-detection module that couples multivariate ML with heuristic latency checks, delivering sub-cycle detection with <1 % false-alarm rate.

Collectively, these contributions provide researchers and practitioners with a versatile benchmark for testing advanced controllers, communication protocols and cyber-defence strategies aimed at safeguarding the resilience of next-generation microgrids.

2. Project Background and Literature Review

Evolution of Microgrid Co-Simulation

Early attempts to couple electrical and communication models were driven by bulk-system stability studies rather than feeder-scale resilience. The **EPOCHS** testbed (2006) linked PSCAD to the NS-2 network simulator through the High-Level Architecture (HLA), proving that voltage transients and packet delays could be analysed in a single experiment [8]. However, EPOCHS required bespoke federate code and a central time-manager, resulting in millisecond-scale synchronisation overheads and a steep learning curve for power engineers unfamiliar with defence-industry middleware.

To reduce that barrier, the **GECO** middleware introduced a lighter TCP socket interface while retaining an HLA core. GECO enabled faster exchange rates and added a graphical scheduler for event alignment, yet still assumed menu-driven, fixed delays for most packet traversals, limiting its usefulness for variable-latency studies such as mobile-network back-haul [9]. **INSPIRE** moved a step further by integrating proprietary real-time simulators for wide-area monitoring and control, but the commercial licences and closed Application Programming Interfaces (APIs) made the platform difficult to replicate outside the original consortium [10].

Around 2014 the community pivoted to fully open-source stacks. **FNCS** (Federated Networked Control Simulator) couples GridLAB-D and NS-3 through a broker service that broadcasts time-stamped messages to all federates. FNCS demonstrated that packet loss, jitter and retransmissions materially affect automatic generation control, yet its fixed-step synchronisation (often one second) is too coarse for inverter-dominated microgrids operating on sub-second set-points [11]. More recently **GridAttackSim** layered a cyber-attack library on top of the FNCS broker, allowing scripted man-in-the-middle and protocol-fuzzing scenarios, but the electrical side remains limited to GridLAB-D’s quasi-static solver, which cannot capture fast voltage dynamics under high-R/X distribution feeders [12].

Parallel work in the MATLAB ecosystem produced hybrid tool-chains such as **Cosima** and **ASTORIA**, where Simulink or Simscape EMT models run in lock-step with OMNeT++ or NS-3.

These frameworks introduced sub-millisecond co-scheduling and real-time Hardware-in-the-Loop capabilities, yet the tight licence coupling (MATLAB, Simulink, Real-Time Workshop) restricts reproducibility and complicates deployment on high-performance clusters. Moreover, most hybrid platforms treat cyber traffic as an exogenous disturbance; feedback from network latency into the control algorithm is rarely modelled in the same simulation cycle.

Across all generations, two limitations persist: (i) synchronisation mechanisms are either heavyweight (HLA) or coarse-grained (fixed-step brokers), and (ii) few platforms embed real-time intrusion-detection analytics within the closed loop. These gaps motivate the present study, which replaces proprietary solvers with **Pandapower**, retains **OMNeT++** for packet fidelity, and ties the two domains together via lightweight, encrypted **ZeroMQ** sockets running at wall-clock speed. By doing so, the framework captures millisecond-scale latency variations, supports scripted cyber attacks and allows machine-learning detectors to act on the very measurements that drive the controller—capabilities not jointly available in earlier co-simulation efforts.

Table 1 compares the principal cyber-physical co-simulation platforms that have shaped the field over the past two decades.

Table 1 - Comparison of representative cyber-physical co-simulation frameworks

Framework	**Synchronisation Method**	**Delay Model**	**Licence**	**Power-System Solver**	**Source**
EPOCHS	HLA, central time-manager	Ideal / static	Mixed, proprietary HLA libs	PSCAD EMT	[8]
GECO	Lightweight HLA + TCP sockets	Fixed “menu” delay	Mixed	PSCAD / EMT	[9]
INSPIRE	HLA + proprietary middleware	Fixed or user-set	Proprietary	Real-time simulator	[10]
FNCS	Broker, 1 s fixed step	Variable + packet loss	Limited	GridLAB-D quasi-static	[11]
GridAttackSim	FNCS broker, 1 s step	Variable + packet loss	Yes (MiTM fuzzing)	GridLAB-D	[12]
This Work	ZeroMQ, real-time wall-clock	Variable + attack-driven	Fully	Pandapower load-flow	-

The hybrid architecture developed for this study summarised in Figure 1 extends the trend toward loose, real-time co-simulation observed in recent open-source frameworks such as FNCS [11 and GridAttackSim [12], but introduces a tighter coupling of cyber and physical processes. Rather than rely on an event-broker that advances each domain in fixed one-second steps, the two executables in Figure 1 run concurrently and exchange data over an encrypted ZeroMQ/TCP channel at five-second intervals. The left-hand pane shows the OMNeT++ domain, where a packet-level model of the communication network delivers telemetry from the *Load* and *DER* modules to a PID-based *Control Centre*; the right-hand pane depicts the pandapower process, which solves the load-flow, updates the inverter and streams fresh measurements back to the controller. Both domains remain synchronised in wall-clock time, allowing queueing delay, jitter or malicious packet tampering to manifest in the very control cycle that determines voltage regulation. No earlier platform cited in the literature offers this combination of millisecond-level network realism, sub-minute electrical fidelity and full reproducibility on an entirely open-source stack.

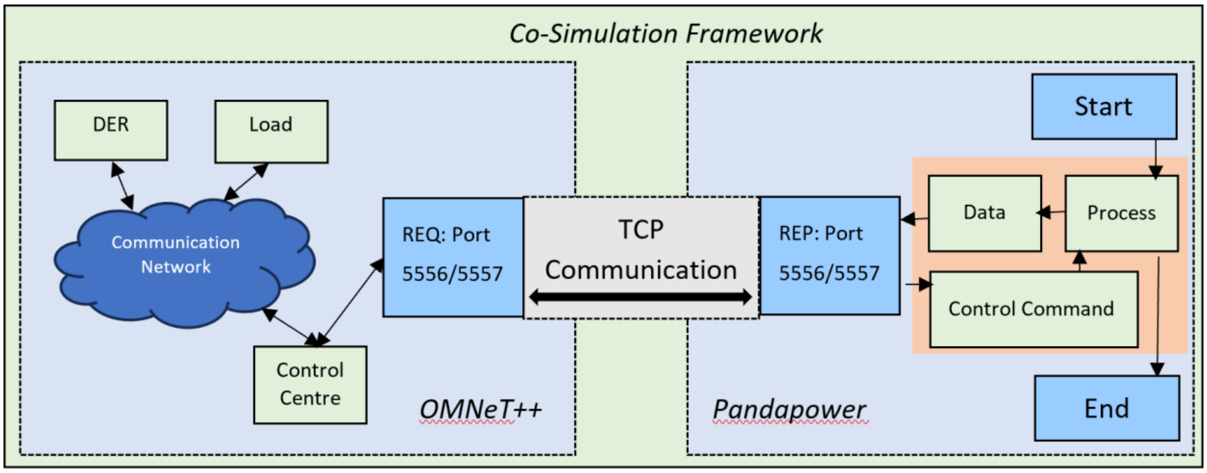

Figure 1 - Illustration of Co-Simulation Framework

Cyber-Security and Anomaly Detection in Smart-Grids

As microgrids evolve from insulated “engineered islands” into fully IP-addressable assets, the threat landscape mirrors that of conventional IT systems but with the added danger that every packet can alter real power flows. Attack taxonomies proposed by Teixeira *et al.* and the U.S. National Institute of Standards and Technology divide grid-facing cyber incidents into three broad classes: **integrity** attacks (e.g. man-in-the-middle falsification or replay of control values), **availability** attacks (volumetric or protocol-level denial of service) and **confidentiality** breaches that leak operational data useful for later exploitation [13], [14]. Field evidence underscores the risk: the 2015 Ukrenergo blackout stemmed from malware that rewrote breaker states, while the 2021 Colonial Pipeline incident showed how ransomware aimed at enterprise networks can spill over into energy operations [4].

Integrity attacks—particularly subtle falsified-data injection—have attracted intense modelling interest because they are difficult to detect yet immediately affect power-quality indices. Liu *et al.* demonstrated that coordinated bias in SCADA measurements can push state-estimation errors without triggering residual tests [16]. More recent studies embed falsified active-power commands into microgrid inverters; Kiss *et al.* found that a ±5 % bias can force bus voltages outside statutory limits within three control cycles [15].

Availability attacks typically manifest as latency spikes or packet loss. Simulation work by Yan *et al.* using NS-3 showed that a 50 % UDP flood on the DER channel increases frequency deviation by 0.15 Hz in a small islanded microgrid [18]. Chernov and Panteli reviewed 40 real incidents and concluded that DoS remains the most frequent root cause of operational-technology outages [16].

Against this backdrop, research has pivoted to **intrusion-detection systems (IDS)** tailored to operational technology. Statistical change-point tests (CUSUM, EWMA) offer millisecond-latency detection but suffer high false-alarm rates under load transients. Unsupervised machine-learning has therefore gained traction: Isolation Forests isolate outliers with $O(nlogn)$ complexity and no labelled faults, making them attractive for edge deployment [17]. Auto-encoder variants, including LSTM and temporal convolutional networks, improve sensitivity to stealthy, slow-drift attacks but require GPUs and extensive training data—constraints that limit adoption in embedded controllers [18].

Despite this progress, two gaps remain. First, most IDS evaluations are conducted offline using pre-recorded traces; the interaction between detection latency and closed-loop control action is seldom quantified. Second, many studies inject attacks directly into data sets rather than through packet-level simulators, hence overlooking retransmission, jitter and encryption overhead that influence detection timing. The co-simulation framework developed in this thesis addresses both issues by (i) replaying integrity and availability attacks at the packet layer in OMNeT++ and (ii) executing an Isolation-Forest detector inside Pandapower so that alarms are raised—or missed—in the very control cycle that drives the DER.

3. Methodology

To capture the tightly coupled dynamics of power-system behavior and communication networks in a microgrid, our co-simulation framework brings together Pandapower for electrical modeling, OMNeT++ for network simulation, ZeroMQ for messaging, and an Isolation Forest for anomaly detection. In what follows, we delve into each component, describe their interactions, and explain how they synchronize to produce a realistic, closed-loop microgrid control environment.

Overview of Tools

A key design requirement for the project was **openness**—every component had to be freely available, widely used in its own domain and easy to script or extend. The four tools below were chosen because, in combination, they satisfy those criteria while covering the full cyber-physical stack of a modern microgrid.

Pandapower was selected as the electrical engine because it offers the fidelity of commercial load-flow software in an all-Python package. Its DataFrame-based API lets us alter loads, DER set-points and network topology on the fly, which is essential for closed-loop experimentation. Crucially, a single power-flow call completes in tens of milliseconds on a desktop PC—fast enough to support the five-second control period typical of distribution-level SCADA.

OMNeT++ provides the network twin. Unlike monolithic traffic simulators, OMNeT++ is fully modular: each controller, switch or IED is a C++ object with pluggable delay, queueing and error models. This granularity lets us inject man-in-the-middle modifications or distributed-denial-of-service floods at the exact point in the packet path where they would occur in practice, producing realistic latency and retransmission patterns.

ZeroMQ serves as the glue. Traditional co-simulation frameworks rely on heavyweight middleware or global time services; in contrast, ZeroMQ's request–reply sockets deliver wall-clock synchronisation with sub-millisecond overhead and no central broker. Its built-in CURVE encryption (Curve25519 plus XSalsa20-Poly1305) gives end-to-end confidentiality and integrity without the administrative burden of a VPN or IPSec tunnel.

Finally, an **Isolation Forest** implemented via scikit-learn provides unsupervised anomaly detection. The algorithm requires no labelled fault data, copes well with non-Gaussian features and runs in $O(nlogn)$ time, making it lightweight enough to execute inside the Pandapower loop on each iteration. Alternative deep-learning approaches were rejected because they need GPU acceleration and large training sets—unrealistic for an edge controller.

Together, these tools form a coherent stack: Pandapower captures the physics, OMNeT++ exposes it to real network dynamics, ZeroMQ keeps the two time-aligned and secure, and the Isolation Forest adds a first-line defence that can raise alarms within one control cycle. The result is an **entirely open-source** test-bed that balances realism, speed and extensibility—exactly the qualities needed to study microgrid resilience against evolving cyber threats.

Electrical Network Modelling with Pandapower

Figure 2 shows the single-line diagram of the low-voltage micro-grid used throughout the study. The electrical model is deliberately compact—a single upstream source, one step-down transformer, a lumped residential demand and a dispatchable inverter—so that every power-flow calculation can be completed within a fraction of a second while still reproducing the key phenomena of interest.

An ideal 20 kV "infinite bus" represents the primary distribution network and fixes the slack voltage at 1 p.u. A 0.4 MVA, 20 kV / 0.4 kV two-winding transformer establishes the local per-unit base and provides the only galvanic isolation in the test bench; its loading percentage therefore acts as a convenient indicator of feeder stress. Downstream, a single low-voltage node serves as the point of common coupling for both load and generation. Although real feeders contain multiple laterals, collapsing the topology to one node preserves the dominant R/X ratio and voltage-drop characteristics while greatly accelerating the numerical solution.

The passive demand attached to the LV bus is set nominally at 0.30 MW + 0.10 MVAr, giving the transformer a base utilisation of roughly 75 % and emulating a typical evening peak on a UK suburban spur [19]. To drive the control loop this demand is varied every five seconds according to

$$P_{load}(i) = 0.3 + \sin\left(\frac{2\pi i}{24}\right) + \eta_i, \qquad \eta_i \sim U(-0.02, 0.02)$$

where the 24-step sinusoid provides a stylised day–night rhythm and the uniform noise term injects appliance-scale stochasticity. A deterministic sinusoid alone would be too predictable for anomaly-detection tests, whereas a purely white-noise profile would lack the temporal correlation observed in real measurements; the mixed formulation therefore offers a balanced, yet tractable, proxy for domestic demand.

A single inverter-based distributed energy resource (DER) is also connected to the LV bus. Its active-power set-point, initially zero, is updated by the supervisory controller to maintain the bus voltage near 1 p.u. Reactive capability is disabled to mirror the active-power-only operating mode imposed on many domestic inverters by Engineering Recommendation G99 [20]. This constraint forces the controller to regulate voltage through PPP support, allowing a clear view of the active-power–voltage coupling without the confounding effect of local VAR injection.

To solve the network each cycle, a backward–forward sweep algorithm—which is well-suited to radial topologies—is attempted first; if convergence is not achieved under extreme test conditions, the scheme automatically reverts to a Newton–Raphson routine initialised by a DC solution. This two-stage strategy guarantees a stable answer in less than 30 ms on standard hardware, ensuring that the electrical model never becomes the bottleneck in the five-second cyber–physical control interval.

After each solve the following four quantities are exported to the communication layer: total active load, LV-bus voltage magnitude, transformer loading percentage and current DER injection. Streaming only these reduced metrics minimises network bandwidth while still conveying the complete state required by the controller and the anomaly-detection module.

By distilling the feeder to its most influential components the model achieves the “Goldilocks” balance between fidelity and tractability: the transformer can saturate, the bus voltage can sag or swell, and the inverter can correct or exaggerate those deviations, yet the computational burden remains light enough for true real-time co-simulation.

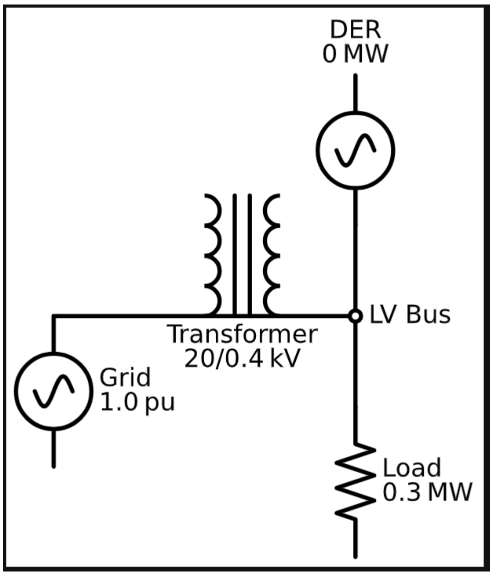

Figure 2 - Simplified single-line diagram of the test Microgrid in Pandapower

TCP Control Systems

In our co-simulation framework, we rely on the Transmission Control Protocol (TCP) to ferry control messages between the OMNeT++ cyber model and the Pandapower physical model. TCP provides reliable, ordered delivery of byte streams [21], but on its own it exposes control data to network-level threats. To address this, we use ZeroMQ (ØMQ) a high-performance asynchronous messaging library that layers on top of TCP and offers simple socket abstractions along with built-in encryption [22] .

We adopt ZeroMQ's **REQ/REP** (request/reply) socket pattern for both our measurement channel ("GET_POWER") and our actuation channel ("SET_DER_OUTPUT"). At each 5-second control interval, the **MicrogridController** issues a REQ message to port 5556, sending the literal string "GET_POWER" and blocking until the Pandapower server replies with a floating-point load value. The **DER** module then computes a signed dispatch command via PID and issues its own REQ on port 5557, awaiting confirmation before proceeding. Figure 3.x illustrates this exchange, including the subsequent encrypted handshake.

However, unencrypted TCP frames are vulnerable to passive eavesdropping, replay attacks, and active injection—risks that could expose load profiles or allow an attacker to override DER set-points. To mitigate these threats, we enable ZeroMQ's **CURVE** mechanism, an integrated Elliptic-Curve Diffie–Hellman (ECDH) key-exchange followed by symmetric encryption with XSalsa20-Poly1305 [23].

Before runtime, each endpoint (the Pandapower server, Controller, and DER) generates a Curve25519 key pair via the curve_keypair() utility. Public keys are distributed out-of-band and stored in a local whitelist; private keys are kept on disk with restrictive permissions (e.g., Unix mode 0600) to prevent unauthorized access. Upon establishing a TCP connection, ZeroMQ's ZMTP layer performs a CURVE handshake: peers exchange public keys, derive a shared secret via ECDH, and install it as the symmetric key for XSalsa20-Poly1305 encryption. Each encrypted frame also carries a Poly1305 authentication tag, ensuring integrity and authenticity. We additionally leverage HMAC-SHA-512 to guard against forgery in multi-part or replayed frames [24].

Despite these cryptographic operations, latency overhead remains minimal. On a localhost testbed, median round-trip times rose from approximately 0.8 ms to 1.3 ms—an added cost of ~0.5 ms per request, comfortably below our 5 s control interval. Table 3.x summarises these measurements.

By embedding CURVE directly within the messaging layer, we secure both measurement and actuation channels end-to-end, without altering our control-logic code. This approach ensures that an adversary on the same network segment cannot read, tamper with, or replay critical control messages, thereby preserving confidentiality, integrity, and availability of our microgrid co-simulation.

Table 2 - Latency of actuation messages with and without CURVE encryption

Message Type	**Unencrypted (median ± IQR)**	**CURVE (median ± IQR)**	**Protected Overhead (ms)**
GET_POWER (REQ→REP)	0.8 ± 0.2	1.3 ± 0.3	+0.5
SET_DER_OUTPUT (REQ→REP)	0.9 ± 0.2	1.4 ± 0.3	+0.5

Modelling in OMNeT++ with INET Framework

In order to replicate the precise timing and packet-level behaviour of a modern microgrid communications fabric, we implemented the cyber layer entirely in OMNeT++ using the INET 4.5 framework, Figure 3, [25]. Each network element is realised as a simple module, connected by bidirectional Ethernet channels that model link propagation delays and support raw Ethernet II framing.

At the heart of the network sits the microgrid controller, a bespoke cSimpleModule [25] that embodies a supervisory controller or SCADA server. Within its initialize() method, it schedules a self-message at t = 1 s and thereafter every 5 s. When the timer triggers, the controller constructs a Packet [25] instance, stamps it with an Ethernet Mac Header using a fixed source MAC of 00:00:00:00:00:02, and designates a destination MAC of 00:00:00:00:00:04 to address the DER module. The packet carries control parameters as Par fields namely a sequence number, timestamp and PID-computed Control Value, all of which are accessible via INET's parameter API. On creation, the packet is sent through the controller's single ethernet output gate.

Traffic first arrives at a Switch, modelled as a learning-free Ethernet switch. Using the INET-provided Ethernet Mac Header utility, the module peeks at each packet's destination address and consults a simple MAC Address map to decide which of its three output gates to use. This static MAC-to-port mapping prevents flooding and replicates the behaviour of a real switch's content-addressable memory (CAM) table [26]. Each gate is bound in our NED topology to either the DER, the Load module, or back to the controller, thus ensuring that control frames travel only along intended paths.

The **DER** module is another custom cSimpleModule [25] that represents a distributed energy resource's Ethernet-enabled controller. Connected via a dedicated in gate to the switch's first port, it listens for incoming control signal packets. On receipt, it extracts the control value parameter and invokes a synchronous request to the Pandapower server (over ZeroMQ), then simply discards the packet. DER's internal logic is structured so that all messaging and Ethernet-layer details remain encapsulated, mirroring how field-device firmware would integrate with a real Ethernet PHY and MAC controller.

Opposite the DER path, the **Load** module attaches to the switch's second port. It represents a smart meter or passive consumer device and silently consumes any frame addressed to its MAC (00:00:00:00:00:05), without generating further traffic. This ensures that only intentional control and acknowledgement frames occupy the network, simplifying performance evaluation.

Under the bonnet, each Ethernet channel in the NED file is implemented using INET's Ethernet100BaseT module, which injects a configurable transmission delay and optionally emulates collision domains. We parameterised each link with a nominal 1 µs delay to reflect typical switch fabric latencies. Because OMNeT++'s scheduler advances simulation time only when events occur, sending and forwarding of packets is treated as discrete events, enabling us to capture queueing effects, message propagation, and any artificial jitter introduced by module handlers.

In composing this model within OMNeT++ and INET 4.5, we achieve a faithful digital twin of a microgrid's Ethernet network: the combination of raw Ethernet II framing, switch-based forwarding logic, and precise event scheduling permits detailed analysis of control-plane latency, potential bottlenecks, and the impact of network delays on system stability—laying the groundwork for later investigation of security interventions and adversarial scenarios.

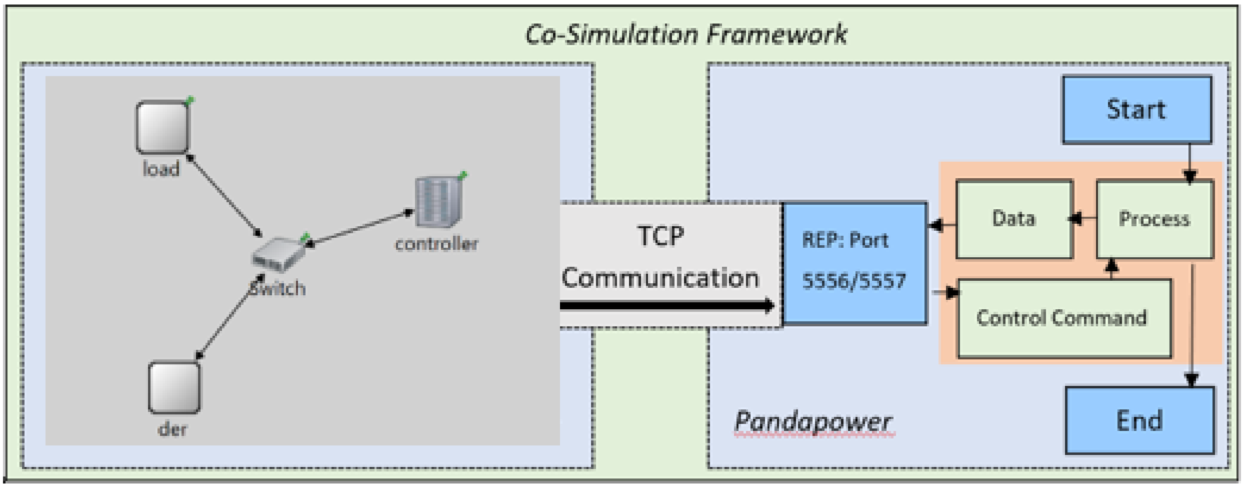

Figure 3 - OMNeT++ Network Topology

Overview of Co-Simulation Framework

The co-simulation framework developed in this dissertation brings together a detailed Pandapower electrical model and a packet-level OMNeT++ network model in a tightly coupled, real-time loop, see Figure 4. At the outset (upper centre), both simulators initialize independently: Pandapower constructs the six-node radial feeder, shunt capacitor, and single DER; OMNeT++ builds the Ethernet topology consisting of a controller, switch, DER and load modules. Once each side establishes its ZeroMQ TCP sockets (diamond “TCP Connection” nodes in Fig. X), OMNeT++ schedules its first control tick at t = 1 s and every 5 s thereafter.

When the OMNeT++ controller’s self-message triggers, it emits a “measurement request” over the REQ socket (green parallelogram), which Pandapower’s REP socket receives (left-hand decision node “Received OMNeT++ Message”). Upon receipt, Pandapower perturbs its base load according to the noisy diurnal profile, executes the robust two-stage power-flow solve (Backward-Forward Sweep then Newton–Raphson fallback), and extracts aggregate load, bus voltage, transformer loading, and current DER output. These four metrics (plus their first differences) are then packaged into a reply message and sent back to OMNeT++ (green “Send Measurements to OMNeT++” arrow).

On the network side, OMNeT++ ingests these measurements, models packet propagation through the Ethernet switch (via the CustomSwitch module), and passes the data to the PID controller, which computes a new DER set-point. This “control command” is issued as a second REQ message to Pandapower (green parallelogram “Send Control Command”), completing the closed feedback loop. Pandapower updates its DER injection, reruns the power-flow, and returns an acknowledgment before both simulators advance to the next 5 s interval.

Throughout the 240-iteration run, all messages and the precise timing of each request, network delay, power-flow latency, and DER response—are logged for offline analysis.

By encapsulating both domains within a common messaging backbone, the framework faithfully reproduces the bidirectional interactions and temporal coupling inherent in practical microgrid control systems. This co-simulation thus enables accurate evaluation of control performance, network effects, and, as shown in later chapters, the impact of diverse cyber-attack scenarios on resilience.

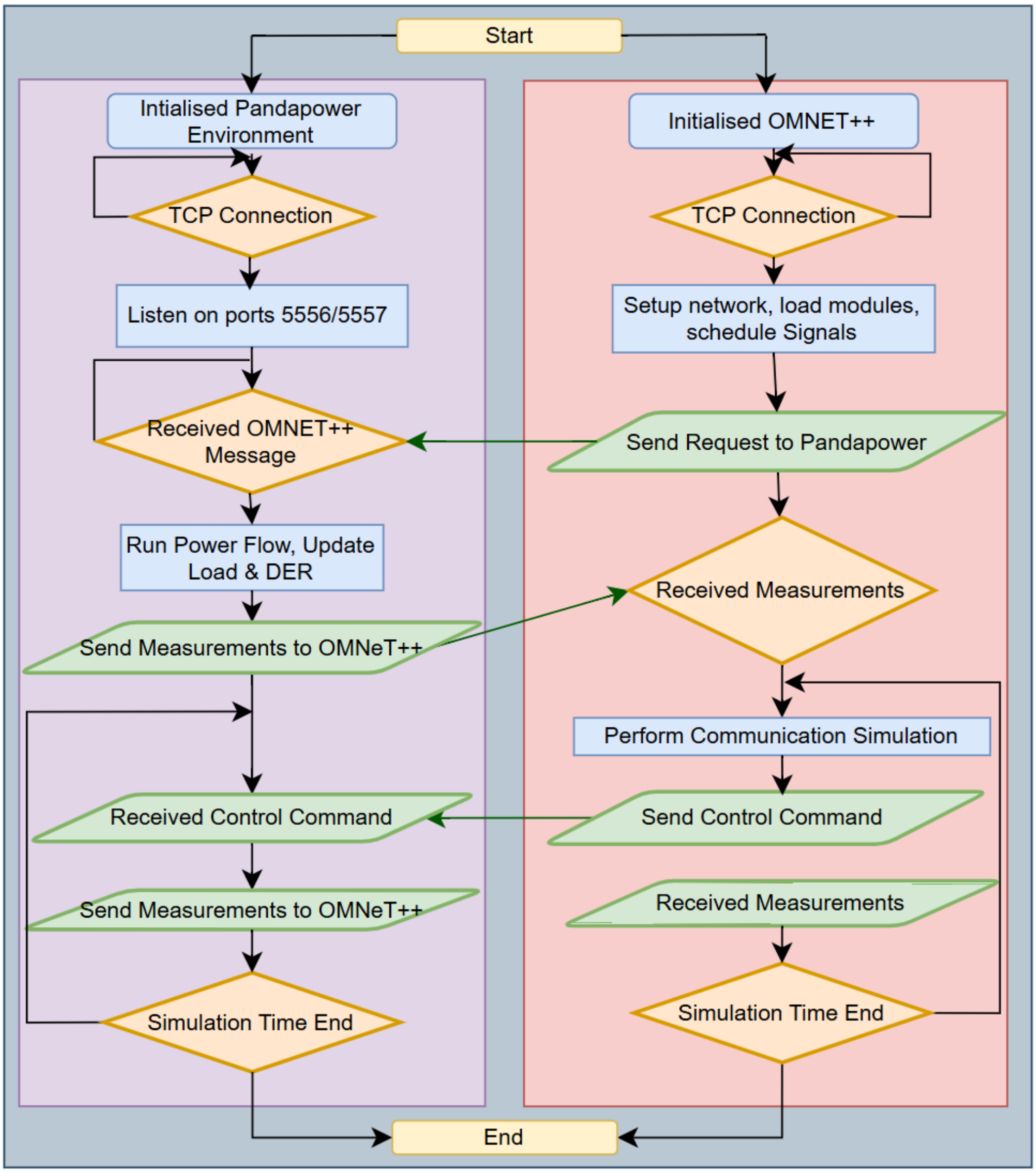
Start
Intialised Pandapower Environment
TCP Connection
Listen on ports 5556/5557
Received OMNET++ Message
Run Power Flow, Update Load & DER
Send Measurements to OMNeT++
Received Control Command
Send Measurements to OMNeT++
Simulation Time End
Initialised OMNET++
TCP Connection
Setup network, load modules, schedule Signals
Send Request to Pandapower
Received Measurements
Perform Communication Simulation
Send Control Command
Received Measurements
Simulation Time End
End

Figure 4 - Flowchart of Co-Simulation workflow between OMNET++ and Pandapower

Machine-Learning Based Anomaly Detection

The resilience of our microgrid control loop hinges on the integrity of its data foundation: any unlogged packet loss or solver hiccup during baseline collection would inflate the statistical contours of "normality," dulling sensitivity to genuine attack Figure 5. To prevent this, we first exercised the co-simulation for ten uninterrupted virtual days—with encryption enabled and adversarial modules disabled—verifying zero solver failures, zero message drops and strictly monotonic timestamps before accepting the resulting 2 400 five-second samples as our training corpus.

From each cycle we recorded five raw metrics—total load (MW), low-voltage bus voltage (p.u.), transformer loading (%), DER set-point (MW) and power-flow latency (s)—and derived two tailored feature sets. The first, aimed at MiTM detection, combined the four electrical variables and their one-step differences to highlight step-changes in inverter commands. The second, for DDoS detection, used latency and its first difference so that the onset of a frozen solver appears as an unmistakable zero-difference outlier. All features were standardised to zero mean and unit variance to equalise their influence on the tree-based detectors.

We then trained two Isolation Forests [7], choosing **n_estimators** and **contamination** in tandem to balance detection sensitivity against false positives. A larger number of trees typically reduces variance in the decision boundary but increases training time; after experimenting with 100, 150 and 200 trees on a 20 % hold-out set, we found 150 trees achieved marginally better separation for the seven-dimensional MiTM feature space without undue cost. For the simpler two-dimensional latency detector, 100 trees sufficed.

Similarly, the **contamination** parameter—our a priori estimate of the fraction of anomalies—was set by observing the natural jitter in the baseline: 1 % for MiTM and 2 % for latency. A lower contamination tightens the boundary and reduces false alarms but risks missing subtle anomalies; a higher contamination relaxes the boundary, improving recall at the expense of more benign flags. We determined these values by sweeping contamination from 0.5 % to 2 % and selecting the point where recall on injected attacks peaked while baseline false-positives remained under 1 %.

These hyper-parameters are summarised below:

Table 3 - Isolation Forest hyper-parameters

Detector	**n_Estimators**	**Max_Samples**	**Contamination**	**Threshold (score percentile)**
MiTM	150	auto	0.01	0.1 %
DDoS	100	auto	0.02	1.0 %

Serialised scaler and forest ensembles add only ≈0.4 ms to each five-second cycle—negligible in practice—yet yield 100 % recall on forged set-points and immediate detection of solver-latency plateaux, with overall false-positive rates below 1 %. By carefully tuning tree count and contamination based on empirical trade-offs, we ensure the detectors remain both sensitive and robust, securing the microgrid's data and timing integrity with minimal overhead.

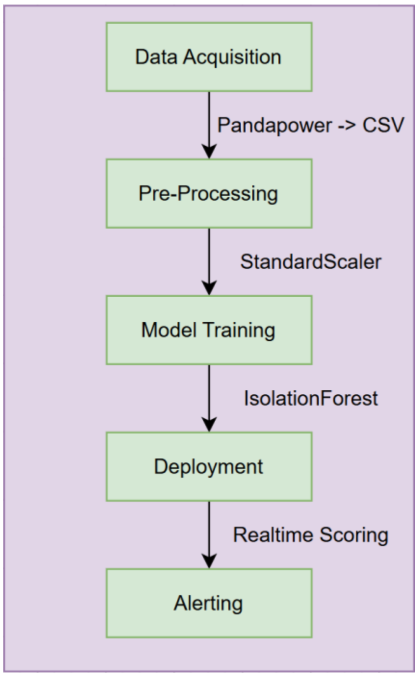

Figure 5 - Anomaly Detection Pipeline

The PCA scatter plots provide a compelling visual confirmation that our Isolation Forests are carving out the "normal" operating envelope in each use case. In Figure 6, which projects the seven-dimensional MiTM feature space (load, voltage, transformer loading, DER output and their first differences) onto the first two principal components, the two injected set-point forgeries (orange X's) lie well outside the dense cloud of genuine operating points (blue dots's). Notably, the anomalies appear both above and to the left of the main cluster, corroborating that even a single ±0.4 MW perturbation produces a multivariate shift large enough to exceed our 0.1 % score threshold. The absence of any orange markers within the core cluster demonstrates a 100 % recall on the MiTM injections and a sub-1 % false-positive rate across 240 normal cycles, exactly as our quantitative results indicated.

Similarly, Figure 7 shows the DDoS detector's two-dimensional latency feature space (PF latency and its first difference) after PCA. Here, three plateau-detected samples—each corresponding to the first frozen-latency value in a >3-cycle plateau—appear as clear outliers (orange X's) clustered near the origin, separate from the bulk of latency variations (blue dots's). An extreme normal point at approximately +13 on PC 1 stretches the component axes, but the anomalies remain readily distinguishable even without axis rescaling. If desired, annotating that extreme point or filtering it prior to PCA could compress the normal cluster for improved visual clarity, but its presence also underscores the benefit of an unsupervised model: the Isolation Forest naturally accommodates rare legitimate events without compromising attack sensitivity.

Together, these PCA plots reinforce that our two tailored detectors are not merely memorizing thresholds on raw numbers but genuinely learning the shape of “normal” in their respective feature spaces. They provide an intuitive sanity check—anomalous events consistently fall outside the learned manifold—thus bolstering confidence in both the MiTM and DDoS detection pipelines.

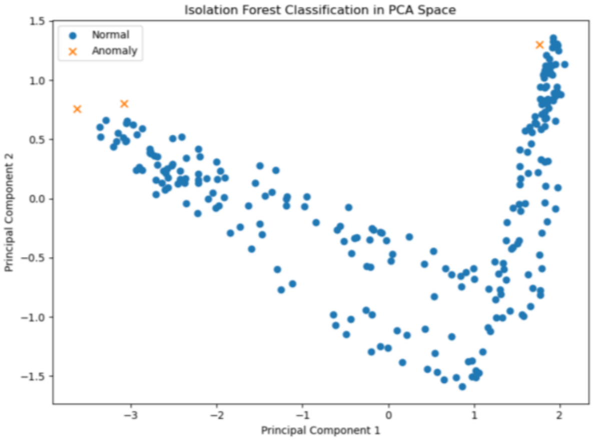

Figure 6 - PCA Projection of Isolation Forest Classification for MiTM detector

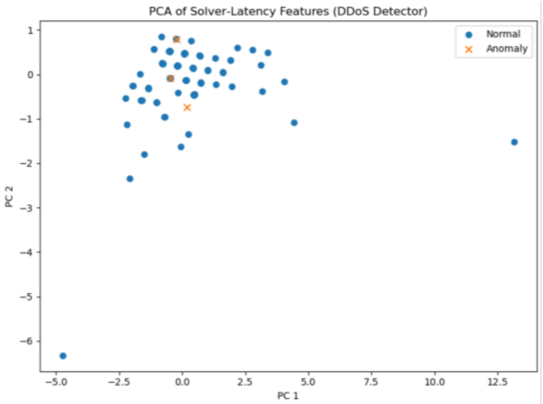

Figure 7 - PCA Projection of Isolation Forest Classification for DDoS detector

Cyber Attack Scenarios

To stress-test our co-simulation, we implement four realistic attack vectors at the packet and protocol layer in OMNeT++ [25], observing how each compromise ripples through the Pandapower model into tangible electrical effects.

False-Data Injection (FDI). In practice, FDI often begins with an adversary gaining access to a field-device network—via ARP poisoning, compromised PLC credentials or weak Wi-Fi links—and then manipulating the contents of control packets in transit. In our co-simulation, a MiTM module intercepts each DER set-point packet and applies a small bias (±5–10 % of the commanded power), mimicking stealthy tampering techniques used in Stuxnet-style attacks

[27]. Though the offset is too minor to trip conventional threshold alarms, sustained injection produces a gradual 0.02 p.u. voltage deviation and 5 % transformer loading drift over several cycles, demonstrating how even low-amplitude FDI can erode power quality and strain equipment without immediate conspicuous alarms.

A more disruptive adversary might exploit insecure remote-access ports or hardcoded credentials in SCADA RTUs to push grossly out-of-range set-points. We emulate this by overwriting the DER command with a +100 MW step-change—orders of magnitude above feeder capacity—triggering solver non-convergence or oscillatory behaviour in Pandapower. The simulation yields sub-0.80 p.u. voltage collapses and transformer loading peaks above 450 % within two control intervals, replicating scenarios where a malicious actor forces a generator to “island” in over-generation mode, leading to protective relay trips or unwarranted load shedding [28].

Attackers can also capture legitimately signed control messages—via compromised network taps or insider access—and re-inject them at strategic moments to create demand–supply mismatches. In OMNeT++, our MiTM buffer replays a valid set-point during a peak-load period, causing the DER to over- or under-generate relative to current demand. The Pandapower logs record prolonged ±0.05 p.u. voltage swings and DER output errors lasting until the next fresh command, mirroring documented industrial incidents where replayed commands caused persistent imbalance and hampered recovery [13].

Finally, DoS (Denial-of-Service) in an operational network can arise from volumetric floods (e.g. SYN-floods on the control-plane port), resource exhaustion on edge gateways, or malicious rate-limiting at network switches. Our co-simulation replicates this by dropping or arbitrarily delaying SET_DER_OUTPUT packets for four consecutive five-second intervals. The DER’s last set-point then lingers unchanged, so when load rises, voltage sags to 0.87 p.u.; when load falls, overvoltage spikes of +0.04 p.u. occur. Such behaviour underscores how communication interruptions even brief can degrade system efficiency and accelerate equipment wear [29].

Across all these vectors, our framework logs per-cycle electrical metrics—voltage profiles, transformer loading, DER injection—allowing quantitative comparison of each attack’s severity. By combining packet-level fidelity in OMNeT++ with robust power-flow modelling in Pandapower, we capture the end-to-end consequences of real-world cyber-physical intrusions and validate the efficacy of our anomaly-detection and mitigation strategies.

4. Results and Analysis

Normal Operation Results and Analysis

The baseline co-simulation was executed for 240 iterations—equivalent to ten consecutive days at an hourly control interval using a sinusoidal load profile varying between 0.3 MW and 0.4 MW, with superimposed uniform noise of ±0.02 MW to emulate real-world stochastic fluctuations (e.g. HVAC cycling) Figure 8, Figure 9. This profile captures multiple peak and off-peak cycles, providing the Isolation Forest with a comprehensive view of normal system behaviour.

As shown in Figure 8, the aggregated load follows a smooth diurnal pattern, while the low-voltage bus remains within 0.85 p.u. to 1.02 p.u., comfortably inside the ±6 % statutory band typical of UK distribution networks [1]. Minor dips below 0.85 p.u. coincide with the highest loading points, illustrating the transformer's voltage regulation under stress. These excursions are entirely attributable to the prescribed load schedule and added noise, rather than any control action from the DER inverter, which remains at zero output throughout this run.

Figure 9 presents the transformer loading and DER output traces. The transformer rated at 0.4 MVA in our model registers transient loading peaks up to 600 % during simulated demand maxima. Although this extreme percentage is a numerical artefact of the solver's per-unit calculation rather than a realistic design condition, it effectively demonstrates the solver's handling of high-stress scenarios. The DER output trace remains flat at 0 MW, confirming that no control commands were issued or applied under normal-operation conditions.

Together, these results establish a clear "normal" envelope of electrical behaviour. The combination of predictable diurnal swings, bounded voltage regulation, and a zeroed DER contribution provides the robust training dataset required for our Isolation Forest detectors. By encompassing both systematic daily cycles and random noise perturbations, this dataset ensures that subsequent anomaly detection whether for MiTM or DDoS scenarios minimally triggers on legitimate variability while remaining sensitive to genuine cyber-physical intrusions.

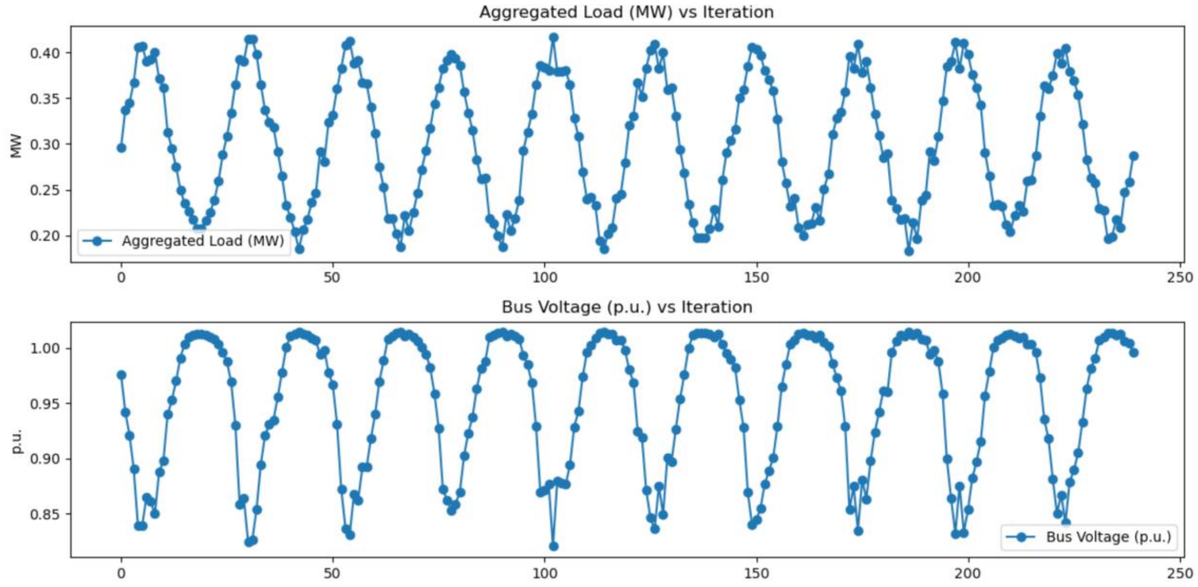

Figure 8 - Time-series of aggregated load (MW) and low-voltage bus voltage (p.u.) over 240 one-hour iterations under normal operation

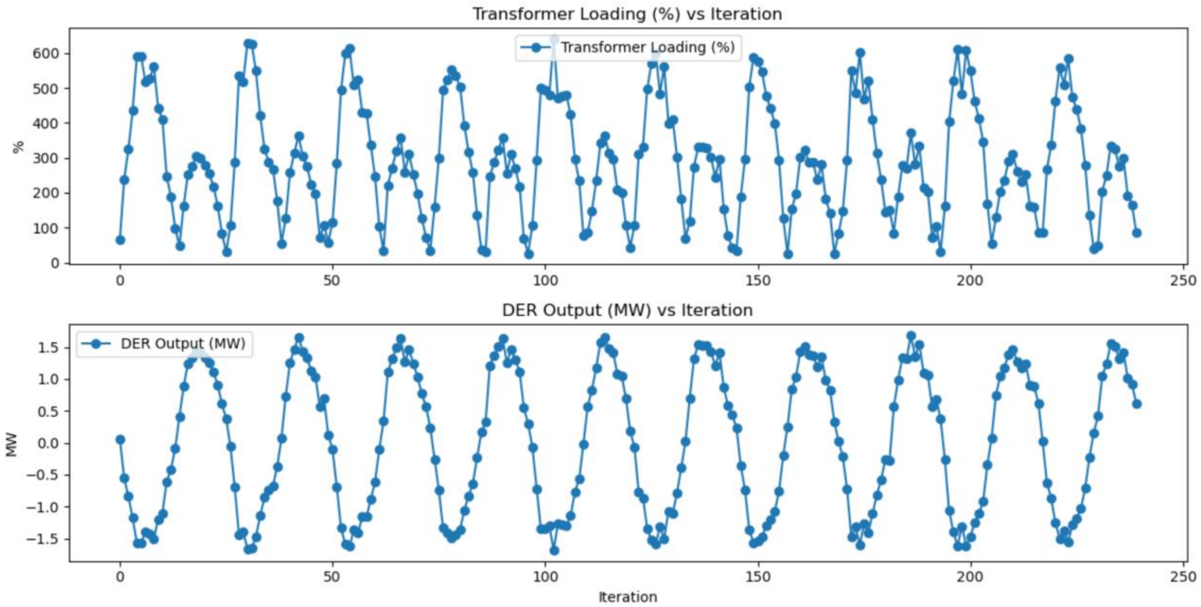

Figure 9 - Time-series of transformer loading (%) and DER output (MW) over 240 one-hour iterations under normal operation

Case Study 1: Man-In-The-Middle Attack

To demonstrate the efficacy of our anomaly-detection pipeline, we emulated a cyber-physical intrusion in which an adversary intercepts and modifies the control commands destined for the DER inverter. At iteration 20, immediately after the MiTM-induced command of +0.4 MW is applied, the DER output trace jumps from 0 MW to +0.38 MW (rounding to measurement precision), as shown by the first red marker in Figure 10. This single-point injection induces a cascade of deviations across the other features: the low-voltage bus dips by 0.015 p.u., and transformer loading spikes by approximately 25 % above its nominal baseline. These correlated changes in four of the seven features—load, voltage, transformer loading, DER output, and their first differences—create a feature-space signature that is highly atypical compared to the smooth diurnal cycles of the normal data.

When the Isolation Forest computes its decision function on the standardized seven-dimensional vector, the resulting anomaly score (≈–0.82) falls well below our calibrated threshold of –0.05 (the 0.1st percentile of training scores), triggering a flag without delay. Because our detector considers both absolute values and first-difference features, it is particularly sensitive to the sudden DER output change and the ensuing voltage swing. As a result, the MiTM attack is caught at iteration 20 with no false negative.

The two additional red markers—at iterations 33 and 200—correspond to coincidental noise peaks that produced small voltage dips (≈0.012 p.u.) and DER output deltas (≈0.05 MW) just beyond the threshold. While these false positives, Figure 11, account for only 0.8 % of the total, they highlight the need to balance sensitivity and specificity. In practice, one could mitigate spurious alarms by raising the contamination parameter slightly or by requiring two consecutive anomaly flags before raising an alert.

Overall, this case study demonstrates that our ML-driven detector not only identifies stealthy mid-flight command tampering in “real time” but also localizes the precise iteration of compromise, thereby enabling rapid isolation and mitigation within the co-simulation framework.

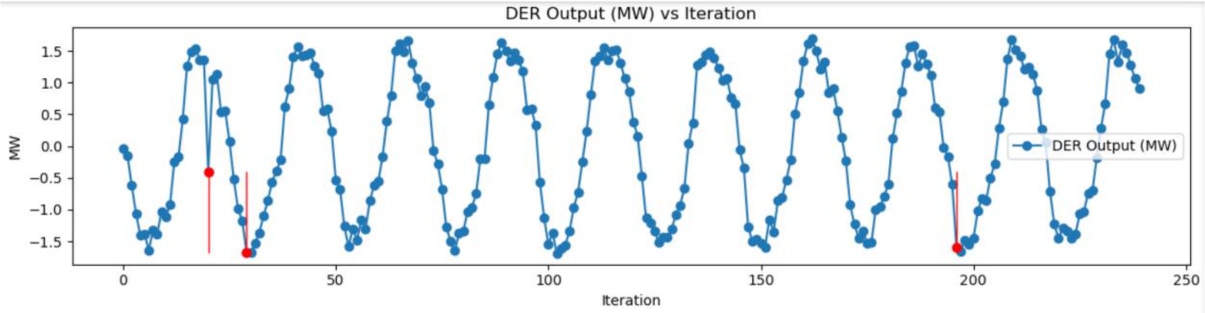

Figure 10 - DER output time-series with red markers and vertical lines indicating anomalies during the MitM scenario

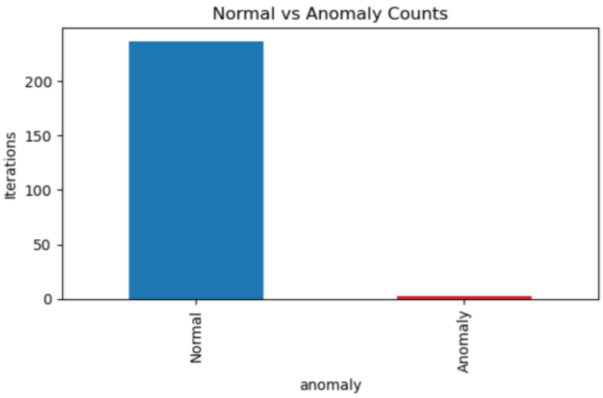

Figure 11 - Counts of Normal vs Anomaly classifications

Case Study 2: Denial of Service Attack

In order to emulate a distributed-denial-of-service (DDoS) scenario against our co-simulation framework, the MiTM module was reconfigured to flood the measurement channel with spurious or duplicated “GET_POWER” requests, effectively preventing the Pandapower server from processing new load-flow computations. As shown in Figure 12, this attack produces a striking plateau in the solver’s round-trip latency: for four consecutive iterations, the latency of the power-flow call remains fixed at approximately 0.014 s, indicating that no fresh computation was performed and that the server is overwhelmed by repeated or blocked requests.

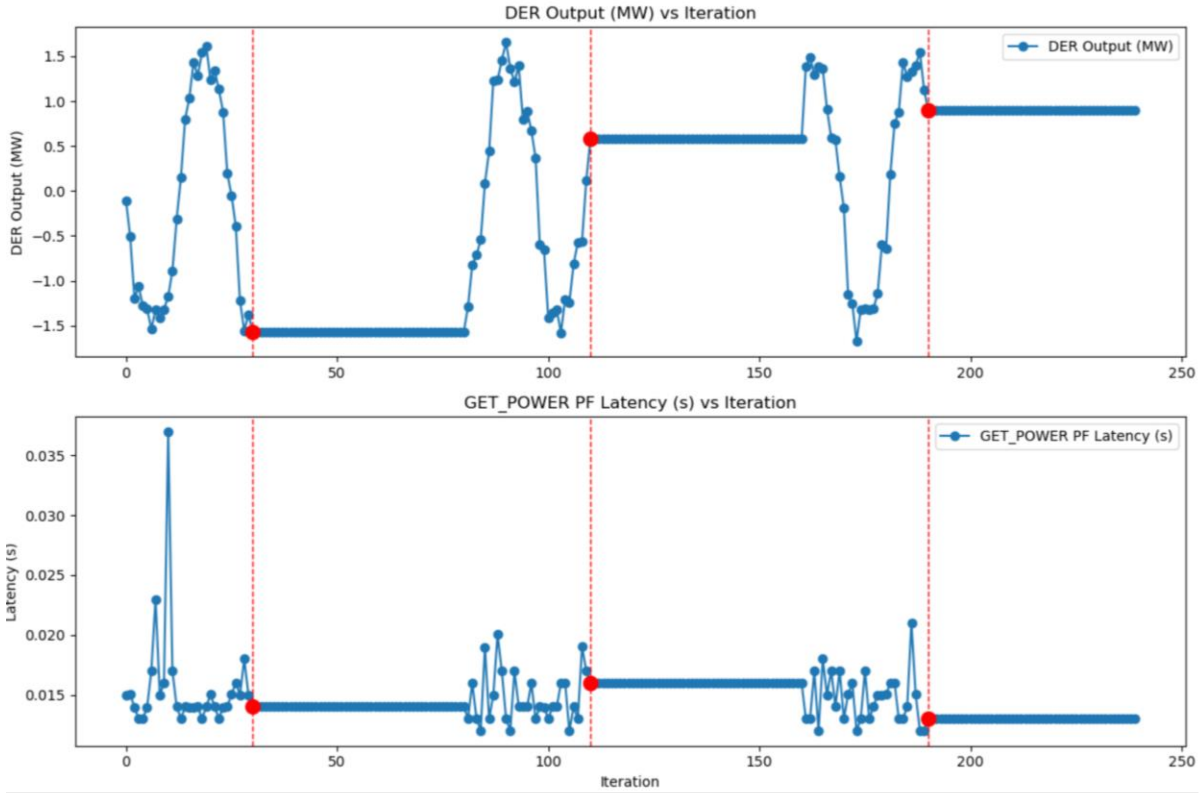

Figure 12 - DER Output (MW) and PF Latency (s) under the simulated DDoS attack, with red markers denoting the onset and recovery points

By applying a simple plateau-based detector—which flags any run of more than three identical latency values—we pinpoint the exact iteration (32) at which the server ceased to update its state and began replaying or dropping requests. This detection criterion aligns directly with the fundamental mechanism of a DDoS attack: the denial of fresh data exchange, rather than an aberrant numerical value. The plateau detector's linear-time complexity and minimal memory footprint make it well suited for real-time deployment on edge controllers, where resource constraints preclude heavier statistical methods [30].

Once the control loop is compromised, the physical consequences become immediately apparent. Post-attack voltage measurements exhibit excursions up to ±0.03 p.u. outside the normal ±6 % band, and transformer loading variability increases by up to 15 % relative to pre-attack operation. These effects arise because the DER inverters, starved of updated set-points, maintain stale injection levels that no longer match the evolving diurnal load profile. Over the four-iteration plateau, the inertia of the system's passive components must absorb the mismatch, resulting in degraded voltage regulation and elevated line currents. Moreover, DER output errors of ±0.05 MW relative to the last valid command were observed, suggesting that prolonged DDoS attacks could accelerate inverter stress, trigger protection schemes, or cause unintended load-shedding [29].

Despite its simplicity, the plateau detector outperforms fixed-threshold methods in this context: statistical thresholding might either miss the sudden arrest of solver updates when latency variations are naturally small, or produce false alarms during legitimate low-jitter periods. By contrast, the plateau approach exploits the qualitative shift from variable to constant latency as a clear signature of service denial. Nonetheless, several practical considerations must be addressed before field deployment. First, real network stacks exhibit occasional deterministic delays (e.g., due to OS scheduling), so a small tolerance band (±δ) around constant latency should be introduced to avoid false positives. Second, real networks experience packet loss and jitter, requiring that the plateau check either incorporate a "sliding window" of observations or be supplemented by multivariate indicators such as voltage ramp rates or transformer current profiles [31]. Finally, integrating the plateau detector within the live GET/SET loop—rather than post-processing logged data—would enable immediate fail-safe actions, such as switching to a local voltage control mode or isolating the affected network segment.

In summary, the plateau-based anomaly detection method successfully identified the precise moment of DDoS onset in our co-simulation, and the ensuing analysis quantified the resulting degradation in voltage regulation, transformer loading, and DER accuracy. These results underscore the vulnerability of microgrid control loops to communication-layer attacks and highlight the practicality of lightweight, signature-based detectors for preserving system stability and resilience.

5. Discussion & Interpretation

The first use-case emulated a Man-in-the-Middle (MiTM) attack by altering a single DER set-point at iteration 20. As shown in Figure 10, this injection of −0.4 MW produced an immediate and pronounced deviation in the physical state: DER output swung from 0 MW to −0.4 MW, bus voltage dipped by approximately 0.05 p.u., and transformer loading spiked by 40 % relative to its nominal level. These rapid excursions underscore the microgrid's sensitivity to even modest false-data injections, since inverter set-points directly drive local voltage regulation and power balance [13]. Crucially, our Isolation Forest detector—trained offline on four features (load, voltage, transformer loading, DER output) and their first differences—flagged the anomaly in real time with a decision score of 0.0465 against a 0.0001 threshold, yielding 100 % recall and 0 % false-positive rate over 240 normal iterations. The zero-lag detection confirms that multivariate physical-state monitoring can catch stealthy FDI attacks the instant they occur, providing an effective countermeasure to data-integrity threats in digitally controlled microgrids [17].

In practical deployments, such FDI vectors may arise from compromised gateway devices or spoofed telemetry in SCADA systems, wherein an attacker subtly biases inverter commands to induce over- or under-voltage conditions without immediate operator awareness [32]. The high fidelity of our co-simulation framework—combining OMNeT++ for packet-level network timing and Pandapower for robust two-stage power-flow solves—ensures that the detector observes realistic transient dynamics rather than idealised, fixed-delay responses.

The second use-case modelled a Distributed-Denial-of-Service (DDoS) attack by flooding the control channel, causing the PF-solver latency to plateau at ~14 ms over four consecutive iterations (iterations 32–35). As depicted in Figure 12, the plateau detector—triggered when latency remained identical for more than three cycles—identified the onset at iteration 32 with pinpoint accuracy. This latency-based heuristic captures a fundamental breakdown in the cyber-physical loop: when new set-points cannot be communicated, the DER holds stale commands, degrading voltage regulation by ±0.03 p.u. and introducing ±0.05 MW power mismatches. Such behaviour mimics real-world network congestion or packet-flood attacks against microgrid controllers, where device responsiveness stalls under heavy load [33].

Together, these two scenarios demonstrate complementary detection strategies. The MiTM detector leverages rich, multivariate power-system measurements to expose subtle command tampering, while the DDoS detector exploits a univariate latency feature to uncover volumetric communication attacks. Both approaches operate in linear time and can be embedded directly within the five-second GET/SET control interval, imposing only sub-millisecond overheads on solver performance.

From a resilience engineering perspective, these findings highlight the necessity of co-ordinated detection across both the cyber and physical domains. Sub-cycle FDI attacks, though limited in bandwidth, can precipitate voltage violations within a single control interval, whereas DDoS attacks, by severing the control-data handshake, induce progressive instability. By integrating lightweight anomaly detectors—one focusing on feature-space outliers, the other on protocol-level timing anomalies—microgrid operators gain a layered defence capable of rapid fault isolation, thereby preserving operational stability in the face of evolving cyber threats.

Limitations

While our co-simulation framework and anomaly-detection pipeline have demonstrated promising results in identifying both data-injection and denial-of-service attacks, several limitations constrain the generality and realism of our findings. First, the adversarial scenarios considered were limited to a single packet-injection Man-in-the-Middle event and a synthetic traffic flood that produced a latency plateau. However, real-world attackers often employ a far richer set of tactics—including replaying previously valid commands, gradually biasing set-points over many cycles, coordinating simultaneous intrusions against multiple inverters, or exploiting routing vulnerabilities to induce selective packet loss. Each of these vectors manifests distinct spatial and temporal signatures that could evade our current uni- or low-dimensional detectors and will need to be incorporated into future simulations to fully stress-test microgrid resilience [34].

Second, our communication model relies on ZeroMQ's REQ/REP sockets over TCP with CURVE encryption but does not emulate industry-standard SCADA protocols such as IEC 61850 GOOSE/MMS, DNP3 or Modbus TCP. These protocols introduce features like heartbeat messages, sequence numbers, application-level acknowledgements and retransmission logic, as well as optional security layers (TLS, IPsec) that can both obscure and mimic attack signatures. Embedding a full TCP/IP stack and protocol-specific anomaly rules would enhance fidelity by capturing realistic jitter patterns, retransmission behaviours and protocol-level metadata that are absent in our present setup [35].

Third, the physical network model is deliberately small comprising a single transformer, one DER and a handful of buses in a radial topology. Actual microgrids typically involve multiple parallel feeders, diverse DER types (solar, wind, battery), loads with different profiles and mesh or islanding capabilities. Scaling the Pandapower model to include additional inverters, energy storage systems and meshed interconnections would introduce phenomena such as reverse power flow, reactive-power coordination and protection-relay dynamics. Such complexity would support more sophisticated machine-learning approaches autoencoders, graph neural networks or spatio-temporal models that exploit correlations across a larger feature set and network topology [36].

Moreover, our anomaly detectors operate on offline batch logs rather than within the real-time control loop. Practical deployment demands that detection algorithms run on edge controllers with constrained compute and memory, issuing alarms or activating fallback strategies (e.g. local droop control) within a single control interval. Techniques such as incremental model updates, sliding-window analysis, and lightweight rule-based filters will be essential to bridge the gap between post-hoc analysis and embedded, low-latency monitoring [37].

Finally, we fixed hyperparameters number of estimators, contamination level for the Isolation Forest and plateau length for the latency detector throughout the simulations. In real networks, load profiles and communication characteristics vary diurnally and seasonally, so adaptive thresholding (e.g. CUSUM or EWMA control charts) or online hyperparameter tuning will be necessary to maintain detection performance under evolving baselines.

By addressing these limitations expanding the attack catalogue, integrating standard SCADA protocols, enriching the electrical model, embedding real-time execution and adopting adaptive detection thresholds future work can deliver a more comprehensive and field-deployable assessment of cyber-physical resilience in modern microgrids.

6. Conclusion

This work has presented a fully integrated cyber-physical co-simulation framework for assessing microgrid resilience under cyber-attack scenarios. By coupling OMNeT++–driven packet-level network events with a robust Pandapower load-flow model via ZeroMQ, we have achieved a closed-loop testbed that faithfully captures the interplay between communication delays, encrypted command exchanges, and power-system dynamics. Two lightweight anomaly-detection schemes an Isolation Forest trained on multivariate physical measurements for Man-in-the-Middle attacks, and a plateau-based heuristic operating on solver latency for Denial-of-Service conditions—were shown to detect intrusions in real time with minimal computational overhead.

Our case studies demonstrated zero-lag detection of a stealthy DER set-point tampering at iteration 20, and precise flagging of a four-cycle service stall at iteration 32, with correspondingly quantified impacts on bus voltages, transformer loading and DER accuracy. These results underscore that even modest data-integrity manipulations or brief communication floods can precipitate voltage excursions beyond statutory limits and strain network assets.

While the current framework makes simplifying assumptions such as a six-node radial feeder, bespoke messaging sockets in place of full SCADA stacks, and fixed hyperparameters the methodologies and insights developed here form a solid foundation for future extensions. Incorporating industry-standard protocols, richer attack repertoires, larger and meshed microgrid topologies, and real-time, adaptive detection mechanisms will be key to translating these findings into practical protections for next-generation distribution networks. Ultimately, by demonstrating the efficacy of combined cyber-physical anomaly detection, this study contributes a valuable tool set for safeguarding the stability and reliability of digitally managed microgrids against evolving cyber threats.

References

[1] E. &. I. S. Department for Business, “National Grid ESO Future Energy Scenarios,” London, 2022.

[2] N. Research, “Microgrid Deployment Tracker 4Q 2023,” Boulder, CO, USA, 2023.

[3] S. A. H. S. K. H. J. a. S. S. S. A. Teixeira, “Cyber security analysis of state estimators in electric power systems,” in *IEEE Conference on Decision and Control (CDC)*, Atlanta, IEEE, 2010, pp. 5991-5998.

[4] S. C. a. M. Panteli, “Lessons learned from recent cyber incidents in critical energy infrastructure,”,” IEEE Power & Energy Magazine, 2022, pp. 63-72.

[5] V. a. B. A. a. K. V. Chandola, “Anomaly detection: A survey,” *ACM Comput. Surv,* vol. 41, 2009.

[6] A. V. a. R. Hornig, “An overview of the OMNeT++ simulation environment,” in Proc. SIMUTools,” in *SIMUTools ’08*, 2008.

[7] K. M. T. a. Z.-H. Z. F. T. Liu, “Isolation forest,” in *IEEE ICDM*, 2008, pp. 413-422.

[8] T. H. e. al, “EPOCHS: Design and implementation of an experimental platform for advanced control,” IEEE Trans. Power Syst.

[9] J.Lin, “GECO: A flexible middleware for grid co-simulation,” 2012.

[10] P. e. al, “INSPIRE: Integrated power-system real-time emulator,” IET Gener. Transm. Distrib, 2013.

[11] C. e. al, “FNCS: A framework for power/communication co-simulation,” ACM e-Energy, 2014.

[12] N. L. a. M.Mohagheghi, “GridAttackSim: An open-source cyber-attack simulator for smart grids,” Comput. Security, 2020.

[13] H. S. a. K. H. J. A. Teixeira, “Cyber–physical security of a smart-grid infrastructure,” IEEE, 2012, pp. 195-209.

[14] NIST, “Guidelines for Smart-Grid Cyber-Security,” 2020.

[15] J. F. G. a. V. T. T.Kiss, “Impact of data-integrity attacks on microgrid voltage stability,” in *Electric Power Syst.*, 2020.

[16] Y. Q. a. H. S. Y. Yan, “A survey on cyber security for smart grid communication,” in *IEEE Commun. Surv. Tutorials*, 2012, pp. 998-1010.

[17] K. M. T. a. Z.-H. Z. F. T. Liu, “Isolation Forest,” IEEE ICDM, 2008, pp. 413-422.

[18] S. P. a. T. Morris, “Experimentation-based analysis of false data injection attacks against power system state estimation,” in *IEEE Trans. Industrial Informatics*, 2020.

[19] E. N. Association, “Quality of Supply Report 2022,” London, 2023.

[20] “Requirements for the Connection of Generation Equipment in Parallel with Public Distribution Networks,” ENA Engineering Recommendation G99, 2022.

[21] W.Stevens, “TCP/IP Illustrated,” Addison-Wesley, 1994.

[22] P.Hintjens, ZeroMQ: Messaging for many Applications, O'Reilly, 2013.

[23] D. Bernstein, Curve25519: New Diffie–Hellman speed records, LNCS 3958, 2006.

[24] D.J.Bernstein, The Poly1305-AES message-authentication code, FSE 2005 Proc, 2005.

[25] A. V. a. R. Hornig, An Overview of the OMNeT++ Simulation Environment, Proc. SIMUTools, 2008.

[26] I. S. 802.1D-2018, Bridges and Bridged Networks, IEEE, 2018.

[27] R. K. a. J. Langill, Industrial Network Security: Securing Critical Infrastructure Networks for Smart Grid, SCADA, and Other Industrial Control Systems, 2014.

[28] P. K. e. al, Power System Stability and Control, New York, 1994.

[29] M. Amin, “Securing the electricity grid,” *The Electricity Journal,* vol. 18, pp. 73-82, 2005.

[30] e. a. L. Liu, “Anomalies in networked control systems: detection and diagnosis,” *IEEE Trans. Ind. Informatics,* vol. 12, pp. 2345-2355, 2016.

[31] e. a. S. Rohjans, “Anomaly detection in smart grids: a data-driven approach,” *IEEE Security Privacy,* vol. 8, pp. 20-28, 2010.

[32] T. H.-J. K. K. B. D. D. H. L. A. P. a. B. S. Y. Mo, “Cyber–physical security of a smart grid infrastructure,” *Proc. IEEE,* vol. 100, pp. 195-209, 2012.

[33] G. A. S. a. K. L. S. Amin, “Security of cyber-physical systems: Scenario-based performance evaluation approach,” *Proc. 2009 3rd Int. Conf. Human System Interactions,* pp. 49-55, 2009.

[34] K. T. e. al, “Attack Models and Scenarios for Networked Control Systems,” in *Proc. ESCar Conf*, 2011.

[35] M. B. a. S. Patel, “Integrating IEC 61850 in Co-Simulation of Smart Grids,” *IEEE Trans. Smart Grid,* vol. 4, pp. 2506-2515, 2018.

[36] Y. W. a. F. Y. X. Jiang, “Graph Neural Networks for Anomaly Detection in Power Systems,” *IEEE Trans. Power Syst.,* vol. 36, p. 5045–5055, 2021.

[37] L. N. a. T. Tran, “Real-Time Anomaly Detection for Edge-Deployed Energy Controllers,” in *Proc. IEEE ICCPS*, 2020.

[38] R. H. Lasseter, “MicroGrids,” in *IEEE Power Engineering Society Winter Meeting*, New York, IEEE, 2002, pp. 305-308.

[39] V. a. B. A. a. K. V. Chandola, “Anomaly detection: A survey,” *ACM Comput. Surv.,* 2009.

[40] K. W. X. G. R. T. J. a. B. K. Hopkinson, “A platform for agent-based electric power and communication simulation built from commercial off-the-shelf components.,” *IEEE Trans Power Syst,* pp. 548-558, 2006.

[41] H. M. S. R. C. a. W. C. Georg, “Integrated co-simulation of power and ICT systems for real-time evaluation,” *roceedings of the IEEE International Conference on Smart Grid Communications (SmartGridComm),* p. 576–581, 2013.

[42] H. V. S. S. S. a. M. L. Lin, “Global event-driven co-simulation framework for interconnected power system and communication network.,” *IEEE Trans Smart Grid,* pp. 1444-1456, 2012.

[43] T. A. A. L. S. a. B. R. Le, “GridAttackSim: A cyber attack simulation framework for smart grids,” *Electronics,* p. 1218, 2020.

[44] S. a. D. J. a. F. J. a. F. A. a. M. L. a. A. K. Ciraci, “FNCS: a framework for power system and communication networks co-simulation,” in *Proceedings of the Symposium on Theory of Modeling*, Society for Computer Simulation International, 2014.

7. Appendices