HYBRID DEEP LEARNING APPROACHES FOR CYBER-PHYSICAL SECURITY OF SMART GRIDS

PhD Dissertation

Kübra BİTİRGEN

Eskişehir, 2024

HYBRID DEEP LEARNING APPROACHES FOR CYBER-PHYSICAL SECURITY OF SMART GRIDS

Kübra BİTİRGEN

PhD Dissertation

Department of Electrical and Electronics Engineering
Supervisor: Prof. Dr. Ümmühan BAŞARAN FİLİK

Eskişehir
Eskişehir Technical University
Institute of Graduate Programs
June 2024

This thesis is supported by the Scientific Research Projects Commission of Eskişehir Technical University under grant number 22DRP192.

FINAL APPROVAL FOR THESIS

This thesis titled "**Hybrid Deep Learning Approaches for Cyber - Physical Security of Smart Grids**" has been prepared and submitted by **Kübra BİTİRGEN** in partial fullfillment of the requirements in "Eskişehir Technical University Directive on Graduate Education and Examination" for the Degree of Doctor of Philosophy (PhD) in Department of Electrical and Electronics Engineering has been examined and approved on 24/06/2024.

Committee Members	Title, Name and Surname	Signature
Member	Prof. Dr. Ümmühan BAŞARAN FİLİK	
Member	Prof. Dr. Ömer Nezih GEREK	
Member	Assist. Prof. Dr. Cahit PERKGÖZ	
Member	Assist. Prof. Dr. Burak URAZEL	
Member	Assist. Prof. Dr. Ahmet Haşim YURTTAKAL	

Prof. Dr. Semra KURAMA

Director of Institute of Graduate Programs

24/06/2024

SUPERVISOR APPROVAL

PhD student Kübra BİTİRGEN, whom I supervise, has completed this thesis titled "**Hybrid Deep Learning Approaches for Cyber-Physical Security of Smart Grids**". According to my inspections, the work is scientifically and ethically appropriate for the student to take the thesis defense exam.

Supervisor

Prof. Dr. Ümmühan BAŞARAN FİLİK

ABSTRACT

Hybrid Deep Learning Approaches for Cyber-Physical Security of Smart Grids

Kübra Bitirgen

Department of Electrical and Electronics Engineering
Programme in Electrical Installation Systems
Eskişehir Technical University, Institute of Graduate Programs, June 2024

Supervisor: Prof. Dr. Ümmühan BAŞARAN FİLİK

The main purpose of this thesis is to propose approaches to detect cyber-physical attacks against smart grids. In this context, three proposed case studies have been conducted to ensure the security of data flowing between buses, phasor measurement units, and phasor data concentrators in smart networks, respectively, against cyber-physical attacks. In the first case study, the detection and classification of cyber-physically attacked data taken from phasor measurement units is carried out as a hybrid approach of the long-short-term memory model and convolutional neural network, whose hyperparameters are optimized with particle swarm optimization. In the second case study, a long-short-term memory model-based Markov game approach is modeled, where the data paths of buses, phasor measurement units, and phasor data concentrators are defined as an environment and the attacker-defense agents are determined as players. In the third case study, a Markov game approach is developed by incorporating a differential evolution algorithm to improve the decision mechanism and strategy, based on the data flow in the three interconnected environments described in the second case study. The proposed Markov game approach is extended to solve the problem of information network management and protection within the communication network of smart grids. Proposed case studies are analyzed using OpenDSS and Python software and the results are presented.

Keywords: Smart grid; Cyber-physical system; Attack detection; Markov game

ÖZET

Akıllı Şebekelerin Siber-Fiziksel Güvenliği için Hibrit Derin Öğrenme Yaklaşımları

Kübra Bitirgen

Elektrik-Elektronik Mühendisligi Anabilim Dalı
Elektrik Tesisleri Bilim Dalı
Eskişehir Teknik Üniversitesi, Lisansüstü Egitim Enstitüsü, Haziran 2024

Danışman: Prof. Dr. Ümmühan BAŞARAN FİLİK

Bu tezin temel amacı, akıllı şebekelere karşı gerçekleştirilen siber - fiziksel saldırıların tespitine yönelik yaklaşımlar önermektir. Bu kapsamda, akıllı şebekelerde sırasıyla baralar, fazör ölçüm birimleri ve fazör veri yoğunlaştırıcıları arasında akan verilerin siber-fiziksel saldırılara karşı güvenliğinin sağlanması amacıyla önerilen üç durum çalışması yürütülmüştür. İlk durum çalışmasında, parçacık sürüsü optimizasyonu ile hiperparametreleri optimize edilen uzun kısa süreli hafıza modeli ve evrişimsel sinir ağının hibrit bir yaklaşım olarak fazör ölçüm birimlerinden çekilen siber fiziksel saldırıya uğrayan verilerin tespiti ve sınıflandırması yapılmıştır. İkinci durum çalışmasında, baralar, fazör ölçüm birimleri ve fazör veri yoğunlaştırıcıların veri yolllarının birer çevre olarak tanımlandığı ve saldırgan-savunma ajanlarının oyuncu olarak belirlendiği uzun kısa süreli hafıza modeli tabanlı Markov oyun yaklaşımı modellenmiştir. Üçüncü vaka çalışmasında, ikinci vaka çalışmasında tanımlanan birbirine bağlı üç ortamdaki veri akışı temel alınarak, karar mekanizmasını ve stratejiyi iyileştirmek amacıyla diferansiyel evrim algoritması dahil edilerek Markov oyunu yaklaşımı geliştirilmiştir. Önerilen Markov oyunu yaklaşımı, akıllı şebekelerin iletişim ağı içerisinde bilgi ağı yönetimi ve korunması sorununu çözecek şekilde genişletilmiştir. OpenDSS ve Python yazılımları kullanılarak önerilen durum çalışmaları analiz edilmiş ve sonuçlar sunulmuştur.

Anahtar Sözcükler: Akıllı şebeke; Siber-fiziksel sistem; Saldırı tespiti; Markov oyunu

ACKNOWLEDGMENT

I would like to express my deepest gratitude to my advisor Prof. Dr. Ümmühan BAŞARAN FİLİK for her guidance, teachings, support, understanding, supervision and patience in my career development, progress in all my academic studies and life.

I am thankful and pay my gratitude to Prof. Dr. Ömer Nezih GEREK and Assist. Prof. Dr. Cahit PERKGÖZ for being in my thesis progress committee and helping my research progress more effectively and successfully with their valuable suggestions, comments and recommendations. I would also like to thank Assist. Prof. Dr. Ahmet Haşim YURTTAKAL and Assist. Prof. Dr. Burak URAZEL for taking part in my dissertation defense committee.

Finally, I would like to express my special thanks to my precious family members. My sincere thanks to my dad, my sister, my brother-in-law and my beloved nephew (your auntie didn't forgot you, little 'Bambinom'). Dad, your prayers are always with me and I am grateful to you and with your permission, I dedicate this thesis to someone who left deep traces and gave inspiration to my life: my beloved angel. She always saw the light in my eyes, taught me that having a pure heart is above all else, and understood what I felt just by looking into my eyes. She gave me pure and unconditional love that I can never repay in my life. She has always made her presence felt with her efforts and she deserves more than this happiness. I hope I can make you happier for the rest of my life. I kept the promise I made when I was little, mom.

Kübra BİTİRGEN

24/06/2024

STATEMENT OF COMPLIANCE WITH ETHICAL PRINCIPLES AND RULES

I hereby truthfully declare that this thesis is an original work prepared by me; that I have behaved in accordance with the scientific ethical principles and rules throughout the stages of preparation, data collection, analysis, and presentation of my work; that I have cited the sources of all the data and information that could be obtained within the scope of this study, and included these sources in the references section; and that this study has been scanned for plagiarism with "scientific plagiarism detection program" used by Eskişehir Technical University, and that "it does not have any plagiarism" whatsoever. I also declare that, if a case contrary to my declaration is detected in my work at any time, I hereby express my consent to all the ethical and legal consequences that are involved.

Kübra BİTİRGEN

TABLE OF CONTENTS

Page

LIST OF FIGURES

Page

LIST OF TABLES

Page

ABBREAVIATIONS

SG : Smart Grid

FDIA : False data injection attack

LSTM : Long-short term memory

CNN : Convolutional neural network

OPF : Optimal power flow

PMUs : Phasor measurement units

PSO : Particle swarm optimization

SCADA : Supervisory Control and Data Acquisition

WAMS : Wide-area measurement system

PDC : Phasor data concentrator

CPA : Cyber-physical attack

SE : State estimator

MC : Markov Core

EMS : Energy management system

AMI : Advanced metering infrastructure

MDP : Markov decision process

MARL : Multi agent learning

CS : Coalition structure

MSV : Markov shapley value

RL : Reinforcement learning

RNN : Recurrent neural network

MCG : Markov core game

CG : Convex game

DE : Differential Evolution

MSQL : Markov Shapley Q Learning

NOMENCLATURE

S	: A set of states
$T^{a}_{ss^{\psi}}$	: A state transition function
A	: A set of actions
$\Re^{a}_{ss^{\psi}}$	: Reward function
μ	: MDP policy
μ^{ψ}	: The optimal policy
$V^{\psi}(s)$	: The (state) value function
$Q^{\psi}(s, a_i, a_{-i})$	: The state-action value function
O	: The set of observations
p_0	: The initial state distribution for $S \rightarrow [0, 1]$
F	: The exploration function
W	: World of states
γ	: Discount factor
$J(\xi)$	: Objective function
a^{a}_{i}, a^{d}_{i}	: Action/Next action taken by agent i
$a^{c_a}_{i}$, $a^{c_d}_{i}$	: Attacker/Detecter action taken by controller agents
o^{a}_{m}, o^{d}_{m}	: Observation/Next observation by agent i
s, s'	: State/Next state by all agents
$\varnothing$	: Empty coalition
Υ	: Shapley-Bellman operator
B	: Set of power buses
N_k	: Set of available PDCs
N_m	: Set of PMUs connected to PDC$_m$
N_u	: Set of available PMUs
G	: Population
I_{best}	: Best individual
I_r	: Randomly selected individual
$E(.)$	: Evolution probability
$\tau^{m}(t)$	: Probability that perimeter point i
σ	: Mutation probability

1. INTRODUCTION

In Section 1.1, recent studies related to the content of this thesis are reviewed. Section 1.2 presents the objectives and motivation behind this research. The research questions are defined in Section 1.3. Finally, the outline of the subsequent chapters is provided in Section 1.4.

1.1. Literature Review

The importance of efficient energy production has grown significantly due to technological and economic advancements, along with the decline of traditional energy sources. The development of communication and information networks has enabled the emergence of smart grid (SG) systems, which now consist of advanced and complex networks [1]. As a result, cyber layers have become integral to the communication networks within SG systems, transforming traditional closed power systems into public data networks. However, the inclusion of cyber layers has introduced vulnerabilities, making SG networks susceptible to cyber-physical security threats. This situation raises concerns about the security vulnerabilities of these networks. The models presented in [2, 3] underscore the significance of network simulation in managing multiple data flows and dynamic control within SG systems. Therefore, comprehensive studies focusing on the vulnerability analysis of SG components and potential attack scenarios are essential.

The main architecture of communication-based SGs is divided into three main components: Advanced Metering Infrastructure (AMI), Supervisory Control and Data Acquisition/Energy Management Systems (SCADA/EMS), and Wide Area Measurement Systems (WAMS).

AMI units are enhanced smart meter systems that regularly measure and record energy consumption data, transmitting it to the utility. This data can be used for further SG system operations through data analysis and IT systems, making AMI a crucial component of SG systems. It provides a bidirectional intelligent connection between the service provider and each consumer [4]. Additionally, AMI enables more accurate fault localization, reduces costs, and improves efficiency for grid-connected service providers, while also reducing energy costs for end users. It allows for bidirectional data transmission, enabling multi-analysis processes for regular data flow, such as optimization and forecasting related to system control and

market pricing.

EMS and SCADA are other essential components of the SG system, serving as valuable tools for SG operators' cybersecurity planning. For instance, the primary objectives of the SCADA system include assisting distributed generation, recording measured data, generating alarms, performing telemetry, logging events, managing switching procedures, and remotely controlling outstation equipment [5]. By evolving security tools and providing guidance on SCADA protocols and security implementation, SCADA ensures sustainable safety for SG networks. In [6], an experiment of detecting cyber attacks on public data sets from a SCADA power system based on the 10-fold cross-validation technique and classifying normal and cyber attack events using selected optimal features is conducted.

The SG central authority mechanism is responsible for collecting dynamic characteristics of transmission parameters and identifying potential security vulnerabilities. Utilizing a local area network, the authority manages human-machine interfaces, control servers, and workstations. WAMS, which provides synchronization through Phasor Measurement Units (PMU), allows for the observation of data from multiple points simultaneously. The central authority collects data from WAMS units, which is transmitted using a GPS-based satellite signal [7].

Power grid operations, such as SG energy management [8, 9], reactive power control, Volt/VAR control [10], and demand response [11], can be effectively analyzed using reinforcement learning (RL)-based approaches. RL has an advantage over other machine learning techniques because it involves reward-dependent agents making sequentially consistent decisions in response to their environment. Unlike classical supervised learning techniques that rely heavily on labeled data and predefined reward signals, RL optimizes decisions based on interactions with the environment [12].

Various methods for detecting False Data Injection Attacks (FDIA) are discussed in the literature, including time-series simulation [13], state forecasting [14], network theory [15], Kalman filter algorithms [16], statistical methods [17], and sparse optimization [18]. These methods have yielded effective results in detecting FDIA data. However, their accuracy relies on the assumption that power flow is provided with optimal accuracy. As noted by [19], due to the nonlinearity of power flow equations in some proposed models, attackers must make educated guesses

about the state estimation (SE) of transmission systems. Furthermore, effective approaches have been proposed for observing FDIAs in SE, considering scenarios with missing or faulty system information [20].

Any attack on interconnected systems can impact both the individual interconnected systems and the central system. Furthermore, SE functions in accordance with its intended purpose, which includes filtering out noise and errors present in the measured data [21]. Regrettably, attackers can infiltrate the system and inject false data into the SE process, leading to potentially disastrous events [22]. In such cases, a bad data detection system is used to protect the state prediction system from unauthorized attacker intrusions. Esmalifalak et al. [23] propose an inference algorithm based on linear independent component analysis and simulation. The algorithm allows an attacker to obtain information about the SG topology and electrical grid status from phasor observations while keeping the attack low detectable. Bitirgen and Başaran Filik presents a performance evaluation of five machine learning approaches combined with principal component analysis to do the task of FDIA detection of an PMU measurements [24]. Nawaz et al. [25] introduce a linear regression-based FDIA against the state estimator of an SG. This method utilizes linear regression to inject false data into the electrical grid, compromising the integrity of state estimator. The simulations in this study are conducted using the IEEE 5-bus system. Chen et al. [26] present a Q-learning-based FDIA generator on the IEEE 39-bus system, specifically targeting automatic voltage control. The study incorporates stealthy attack scenarios into a partially observable Markov decision process model to evaluate the effectiveness of the attacks. In SG systems, case studies based on attacker traces in the cyber-physical layers aim to detect network anomalies and identify the source of falsified data. Bitirgen and Başaran Filik presented a hybrid FDIA detection approach to identify and discriminate cyber-physical attacks in a SG [27].

As the size of smart networks expands, new threats emerge when data leakage in one layer disrupts the functioning of other layers [28]. Studies have demonstrated the effectiveness of RL, showing that using a learnable tool is more effective and reliable for interacting with the environment compared to other methods [29–31]. The SG network faces threats from vulnerabilities in the infrastructure layer to the last communication layer, where malicious attacks can cause failures in cyber-physical nodes. RL approaches offer optimal sequential decision-making policies to maximize

long-term rewards. Recent studies have proposed various combinations of RL with different machine learning models, such as Deep Neural Networks, particularly in high-dimensional settings [32]. Liu and Cheng proposed a specific attack strategy where false data is injected at each step to counteract the anomalies caused by previous false data. This method successfully deceives the Kalman filter-based detector [33]. Additionally, Bitirgen and Başaran Filik proposed Multi-CAs game aiming to coordinate PMU signals across intersections to improve the network efficiency of a 123 bus SG system [34]. The MG model-based MARL approach is improved to enhance network safety, addressing the problem more comprehensively by optimizing policy and incorporating three different environments of the SG system [35]. In [36], wavelet singular values are integrated into the mixture wavelet transform and singular value decomposition method. The resulting mixture is then used as input to a deep machine learning model for detecting abnormalities in SG conditions. In the study by Rahman et al. [37], a stealthy topology poisoning attack that compromises the integrity of the optimal power flow (OPF) routine is presented. The authors analyze the impacts of such attacks on the OPF.

To address the temporal relationship of historical data and effectively identify large instantaneous noise in SG networks, the FDIA detection problem is formulated as a partially observable Markov decision process model in [38]. Model-free RL is then considered as a tool to detect harmful attacks. To effectively utilize this detector, it is necessary to collect a large number of datasets for training. Robust detection methods based on model-free RL frameworks have been developed to tackle these challenges. To optimize system operations and detect various types of attacks in different attack scenarios, Deep Q-Learning algorithms are a powerful choice. Moreover, the data flow in the SG system is constrained by the power demand of the main grid, taking into account the interests of each subsystem.

One of the most powerful and versatile evolutionary optimization algorithms for continuous parameter spaces in current use to solve complex problems in a reasonable time frame is differential evolution (DE) [39]. It is popular in the field of stochastic real-parameter optimization with its satisfactory results. Unlike other evolutionary algorithms that generate offspring by disrupting solutions with scaled difference vectors, DE algorithm produces new offspring by recombining existing solutions under certain conditions until the algorithm is completed [40]. It is mentioned that DE algorithms can use both synchronous/asynchronous modes of sur-

vivor selection or population updating. In the effectiveness of the dynamic updating strategy of population, the population is updated simultaneously with all new solutions, without each individual being updated immediately after being produced. Moreover, studies based on DE have been successfully applied to attack detection in SG applications [41].

The DE algorithm is used for optimization accuracy and time to detect unknown intrusion vectors and to adjust the systems parameters for accurate and as fast as possible detection of attacks [42]. In recent years, DE has been reported to be a successful meta-heuristic method and to solve the relay coordination problem smoothly [43]. In the algorithm [44], a flexible encoding scheme, evolution operator, and crossover operation are proposed to indicate the attack location to solve dynamic false data injection attack (DFDIA) model. Therefore, the FCLs and DOCRs optimization problems are proposed to solve via Differential Evolution Multi-Objective Algorithm (DEMO) [45]. However, many existing evolutionary multi-objective optimization algorithms excel in solving unconstrained multi-objective optimization problems, they encounter greater challenges when applied to constrained multi-objective optimization problems, especially with low applicability [46].

Data-driven algorithms based largely on deep learning are preferred for current attack detection researches and studies. A model-free algorithm that is DeepLearning-based locational detection architecture based on CNN to utilize the spatial characteristics of the data to detect attacked location accurately is proposed in for the whole 14-bus and 118-bus systems under various noise and attack conditions [47]. The model architecture, which does not depend on any default attack model, does not require any changes to the existing BDD system and is built on existing BDD and the detection process is microsecond in size. In recent years, the development of a data-driven network modeling framework based on graph neural network models (GNN) are among the successful methods about capturing topological connections of graph networks, providing successful solutions to the related problems [48]. In paralel with this, the application field of GNN in the SG has recently expanded to FDIA detection for electrical grids modeled as undirected connected weighted graphs [49–51]. Javaid et al. presented adaptive synthesis methods in [52], developed to identify outliers, reduce overfitting, and address imbalanced data flows in SG systems.

Ensuring the security of PMUs against cyber-physical risks is crucial for main-

taining the security of information flow in the SG and its interconnected networks. Ding et al. proposed a probabilistic model on the effects of various cyber factors on PMU deployments [53]. These factors include risk propagation, the dependency between attack outcomes, and the number of PMUs directly attacked. Wu et al. have presented a deep learning model where encryption of transmitted data via non-linear mapping is achieved using the same modified denoising auto-encoder, which prevents attackers from recording useful information [54]. Additionally, Li et al. developed an efficient heuristic based on greedy search for locally coordinated cyber-physical attacks [55]. Their approach, including case studies on the six-bus system and the two-domain IEEE RTS-96 system, provides satisfactory solutions for real-time application

As a decentralized approach, federated learning has been successfully applied to detect IoT security attacks. Hence, the study of efficient data encryption algorithms holds significant importance for SG systems. An effective cross-silo federated learning scheme utilizing secret sharing has been proposed and empirically evaluated for privacy enhancement. Consequently, the algorithm is resilient to local private data extraction attacks within the SG domain [56]. Additionally, a transformer based FDIA detection model is proposed for secure federated learning scheme local training, combining the Paillier cryptosystem [57]. An attack which aims to distrupt a signal in a centralized management may be set to use just the certain amount of power to stop a specific part [58]. In a decentralized attack model, each attacker shares information with neighbor attackers to capture the system data flow. Shen et al. for the state model of a closed-loop power system for one area, load frequency control of a single-area power system under denial of service attacks has been studied [59].

A Denial of Service (DDoS) attack is a serious attack that can prevent the usage for authentic user when a similar intrusions happen from different sources. DDoS attacks act in cyber layers that can cause serious transmission load and also cause an interruption in the part or whole network connection, control and/or communication [60–62]. In network control systems, manipulation of channels makes it easy to inject DDoS attacks, thus preventing measurement and control data from reaching targets [63]. A new distributed fuzzy load frequency control approach is proposed for multi-domain power systems under cross-layer attacks [60]. Interlayer attacks that destabilize power systems are treated and modeled as an independent Bernoulli

process, including PMU attacks in the physical layer and DDoS attacks in the cyber layer. The main sources of disruption of DDoS attacks are the transmission channel infrastructures, including the connection between the RTU, PMU and the control center, respectively [64]. In addition, this study addresses an analysis for possible vulnerabilities of the SCADA system and also present the impact analysis of DDoS attacks by modeling attacks using influence diagram [65]. The study focused on Remote Telemetry Unit and Master Terminal Unit by targeting availability of the system. DDoS attacks can block the measurement information transmitted to the control center, prevent the upgrading of a command from the control center, or disrupt the performance of the electrical network by causing a delay in the control signals transferred to the actuator [66].

1.2. Problem Statement, Main Contributions and Novelties

Building upon the aforementioned studies, the primary objective of this thesis is to propose cyber-physical attack detection approaches for SG systems. The expanding communication layers of cyber spaces within SG systems are increasingly susceptible to threats such as theft, manipulation, and malfunction in the connected electrical network. In line with this objective, the problem of attack detection in a SG network from buses to Phasor Data Concentrator (PDC) units is identified. Following this problem formulation, three main approaches have been proposed, each containing the following innovations:

- To address CPA detection in the SG system, an approach called convolutional neural network long short-term memory particle swarm optimization (CNN-LSTM-PSO) is proposed. This approach optimizes CNN-LSTM using PSO based on PMU data. It enables the detection and discrimination of faulty data caused by cyber-attacks or physical failures. The study focuses on addressing the security concerns of SG systems. Specifically, it emphasizes the criticality of ensuring the optimal protection of data flow within the communication and power network to maintain the reliability of the central control in SG systems.
- A novel approach that integrates the PSO-LSTM algorithm with a MG-based framework to address the problem of identifying optimal strategies for detecting faulty data and their sources within the system is presented. PSO, a meta-heuristic method, is employed within the LSTM strategy update phase as a time series algorithm. The PSO-based LSTM-CNN algorithm is intro-

duced for CPA detection, serving as compelling evidence for the efficacy of the proposed approach. Within this MG, the LSTM structure functions as the policy iteration agent. Validation of the proposed MG-based approach is conducted using a 123-bus IEEE test system.

- The proposed MG is enhanced using the DE algorithm. A RL-based game is developed in which attackers create observation and attack strategies based on the system's complexity and their observations. This game is built using the initial classification model, CNN-LSTM, leading to an effective detection system. The system's robustness is further improved through DE-based optimization.

- The CNN-LSTM model is chosen for the decision mechanism in the MG due to its successful performance with time series data. The DE-enhanced Markov game (DE-MG) can be further improved by incorporating innovative contributions that enhance data path security through the effective discovery and exploitation capabilities of additional agents.

1.3. Research Questions

Based on the study results in the literature and considering the cybersecurity issues of SG systems, the following research questions are investigated:

I. What is the methodological trend and the current most effective approach to network security of cyber attacks on data exchange in the SG?

II. Is there a clear causal relationship between attack severity, connectivity and rate between vulnerable and powerful network structures in the same network, or can cyber networks have both high-level operational and below-average attack detection?

III. How does taking into account the temporal distribution, that is, the time at which attacks or electrical faults occur in the normal operating cycle of networks, affect the conclusions reached in the context of attack point detection?

1.4. Outline

An outline of the thesis is briefly given in this section. A detailed outline of subsequent chapters is given at the beginning of each chapter. In Chapter 2, the problem of CPA for SG systems is introduced along with the PSO-CNN-LSTM

approach proposed to address this problem. In Chapter 3, the MARL approach is proposed as solutions for the problem of CPA in SG systems. As a first MG, the CPA detection mechanism, which is designed to protect vulnerable layers and to sustain safety of SG systems, is presented. Additionally, the initial CPA detection and discrimination approach is enhanced using DE as the second MG approach. In Chapter 4, SG simulations are presented to show the effectiveness of the developed MARL approach. Finally, in Chapter 5, the key findings of the thesis are highlighted, and possible research objectives for the future are discussed.

2. CYBER-PHYSICAL ATTACK DETECTION

In Section 2.1, the formulation of cyber-physical attack problem is presented, which includes data manipulation of attackers and physical disturbances of the SG system. In Section 2.2, the definition of the proposed M-MARL approach is given. In Section 2.3, the procedure for solving the considered problem is explained based on the three environment RL algorithm, the implementation steps for the M-MARL game approach.

2.1. Problem Description

2.1.1. Details about streaming PMU dataset

PMU has time-synchronized data consisting of device status and measurements of the system. The system state is monitored by using this data. To sustain the normal operating conditions of the network, quality and cybersecurity challenges of synchrophasor technology should be deeply considered. This study depends on an open-source simulated power system data provided by Mississippi State University [67]. To monitor the SG devices, the power grid of the system has multi layers including SNORT and Syslog systems. The central control center is in communication with these systems. The system state is monitored by four PMUs on the transmission lines. Each PDC in the network stores the time-stamped synchrophasor data via Ethernet at a frequency of 120 samples/second. The system also has intelligent electronic devices switching the circuit breakers on or off. The voltage magnitude and A-C phase angle of voltage and measured by PMU_1 are represented by R1-PM1:V - PM3:V and R1-PA1:VH - PA3:VH. The SG system has three different classes for system faults and disturbances. Voltage/current magnitude and phase angles at phase respectively are represented as binary-class, three-class, and multi-class data under varying cyber-physical attacks and normal conditions seen in Figure 2.1 to 2.3. Problems in the SG networks are represented as following:

- Attacked and natural events are in the first class including two system states in the binary-class.
- Attack events, natural events, and normal conditions are considered three-class.

- Normal conditions, disturbances, and attacks are called multi-class.

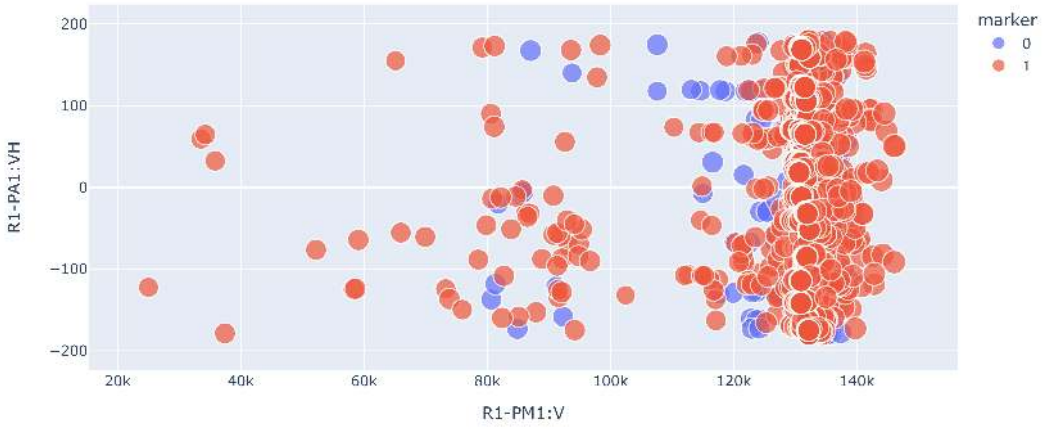


Figure 2.1. *Phase angle versus voltage magnitude at phase A for binary-class measurement*

The dataset including measurements of four PMUs is used in this study. In addition, data consist of thousands of individual samples, 29 types of parameters and 128 features from each PMU located in the network. Furthermore, the data consists of 294 no-event instances, 3,711 attack instances, and 1,221 natural events instances.

Figure 2.2. *Phase angle versus voltage magnitude at phase A for three-class measurement*

2.1.2. Disturbances and attack characteristics analysis

The types of attacks are defined below. Binary-class data consists of two conditions of the network. The binary-class distinction between "attacked" and "natural

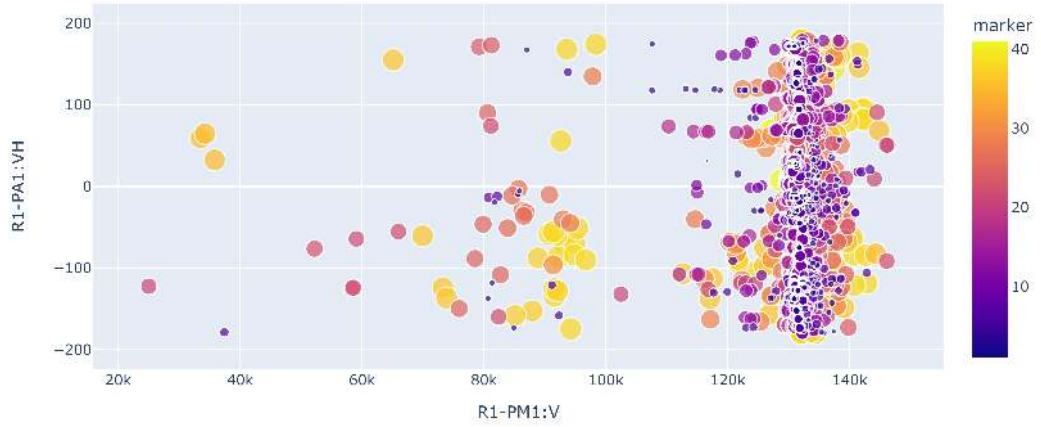


Figure 2.3. *Phase angle versus voltage magnitude at phase A for multi-class measurement*

events" is depicted in Figure 2.1. In this figure, natural events are represented as 1 and attack events are represented as 0.

Three-class data encompasses three different conditions: natural events, normal conditions, and attack events. In Figure 2.2, natural events are represented in red, normal conditions in green, and attack events in blue.

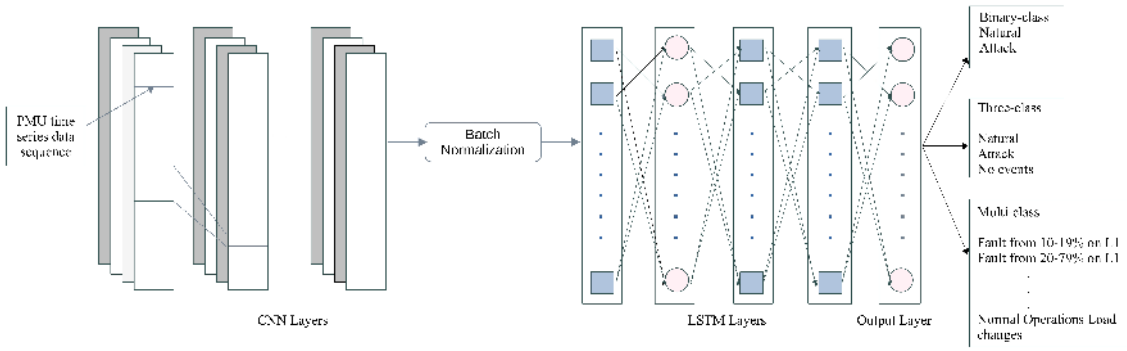


Figure 2.4. *The workflow of the CNN-LSTM based CPA detection algorithm*

Third class is a multi-class data consisting of 37 types of conditions that are 5 main types of attacks, physical disturbances, and normal condition. Natural events occurres in the power network such as SLG faults represented as 1 to 6, SLG fault replay attacks represented as 7 to 12, and line maintenance represented as 13 to 14. Remote tripping command injection attacks are represented as 15 to 18. Command injection attacks against a single relay are represented as 19 to 20. Relay setting change attacks are explained such as an attack disabling a single relay ranked as

21 to 30. From 31 to 34, there are no ranked attacks. The attacks disabling two relays are ranked from 35 to 38. The integrated attacks including disabling relay function and line maintenance are represented 39 to 40. In addition, each marked color represents a condition such as a fault from 20-79% on the second power line is represented as 5, a command injection against the third relay and the fourth relay is represented as 20, and disabling relay function from 20-49% on the power line 2 is represented as 27. Load changes accepted as normal operation are represented as 41. Furthermore, the main attack types exist in the dataset are defined as follows:

- The main objective of relay attacks is to break the distance protection by preventing of tripping of the breakers. An attacker generally tricks a network into authenticating to an attacker side.
- Short-circuit fault: It is a faulty junction that occurs at some points of the power line, whether made accidentally or intentionally.
- Line maintenance: Line maintenance involves checking and correcting any disturbances or malfunctions that may have occurred during the network operation.
- Data injection: It is one form of data attack orchestrated when adversaries can alter or modify the measurements supplied by PMUs, PDC or sensors in the network, misleading the control center's computational capability.
- Remote tripping command injection: Attackers block out the system protection and send deceptive commands to selected relays.

2.2. Cyber-Pyhsical Attack Detection and Discrimination

In order to develop a successful detection algorithm, this large data set must first be preprocessed. Various studies using LSTM show that it is a model that provides efficient results with high accuracy for this type of data [68,69]. Moreover, the detection mechanism where PSO is incorporated into the system is used to improve detection accuracy, LSTM model performance, and reduce execution time. Selecting the most appropriate values of hyperparameters is an important step to launch a successful DL classifier. It performs hyperparameter optimization to improve the performance of the DNN on the selected dataset. PSO is successfully applied to the

proposed hybrid CNN-LSTM model, simplifying the deep learning training stages. PSO-based LSTM model is generally preferred in predictive or detection-based studies to minimize the weights of key parameters, optimize training time, and improve LSTM [70, 71] performance. The proposed model for FDIA detection on PMU measurements is given in Figure 2.4.

2.2.1. CNN model

It has been successfully demonstrated that the CNN model has a strong ability to process data from different network systems for cyber attack detection. The CNN architecture consists of several layers: convolution, pooling, fully connected, and output layers. The convolution layer is particularly significant due to its ability to extract features from input variables using convolution kernels, making it the most crucial layer in the CNN model.

In this study, the CNN structure is meticulously designed to extract and clarify information from the data, achieving high detection rates in binary, three-class, and multi-class data classifications. Pooling layers are not used in this design to handle large volume parameters effectively without issues for one-dimensional data. To enhance the model's performance, samples undergo a normalization phase before processing.

During the training phase, the Adam optimizer is employed to determine the adaptive learning rate for each parameter. This optimizer is chosen for its ease of implementation, efficient memory usage, and computational efficiency. Training loss results are monitored to prevent jumps and overfitting during the training process.

The proposed model's architecture, designed for false data detection, includes an input layer, several convolution layers, LSTM layers, and an output dense layer. In (3.8), * and $f_{1,j}$ represent the convolution operator and the feature maps of the first layer generated from the input data x, respectively. The term $h_{1,j}$, a 1D filter, denotes the j^{th} convolution kernel. A scalar bias, $b_{1,j}$, is also included in the convolution output. The output at location i is defined as follows [72]:

$$f_{1j} = ReLU(x * h_{1j} + b_{1j}) \tag{2.1}$$

The convolution operation is denoted by $*$ in (2.1) and the output at position I is defined as:

$$E(t)_{k=1}^{l_{1j}} = h_{1j} \times (x)[I - k + \frac{l_{1j}}{2}] \tag{2.2}$$

$l_{1,j}$ and denote the length of the filter at $h_{1,j}$ and the inner product operation, respectively. The hidden features h_{qj} generated from filters in the $(q-1)^{th}$ convolutional layer are used as the input to the q^{th} convolutional layer. In (3.8), $f_{q,j}$ is the j^{th} feature map at the q^{th} convolutional layer and the output calculation is expressed.

$$f_{q,j} = ReLU(f_{q-1} * h_{qj} + b_{qj}) \tag{2.3}$$

The number of filters in each layer and the depth of the convolution layers are set by hyperparameters optimized by the PSO algorithm and adjusted according to the system.

2.2.2. LSTM model

LSTM is a type of RNN algorithm designed to evaluate patterns in sequential data, exhibiting significantly high accuracy in detecting FDIAs. Among the hybrid models studied, the proposed model for CPA addresses system protection deficiencies and meets various control requirements.

The accuracy of deep learning models is generally affected by the sample types and dimensions. Compared to other models, LSTM is particularly powerful for dynamic time-series data and offers advantages in multi-class classification. However, LSTM models can lose sequence information when dealing with large datasets, necessitating iterative and time-consuming hyperparameter adjustments.

Figure 2.4 illustrates the architectures of CNN-LSTM for binary-class, three-class, and multi-class classifications across different datasets. LSTM's robust algorithm preserves short-term memories without losing long-term dependencies, outperforming RNNs with a single gate. In LSTM, x_t represents the current state vector at time t, σ denotes the logistic sigmoid function, and c is the cell state. The weights in each gate are learnable parameters that enhance LSTM's capability beyond simple neurons, enabling the model to effectively memorize prior sequences.

Each LSTM cell comprises three gates: the forget gate f, the output gate o, and the input gate i. These gates control the flow of information, as follows [73]:

- f determines what information to discard from the cell state.
- i updates the cell state with new information.
- The output gate o decides what information to output from the cell state.

This gating mechanism allows LSTM blocks to handle both short-term and long-term dependencies effectively, making them more capable than standard RNN cells with a single gate.

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \tag{2.4}$$

$$c_t = f_t c_{t-1} + i_c tanh(W_{xc}x_t + W_c h_{t-1} + b_c) \tag{2.5}$$

Whether the previous memory h_{t-1} pass or not is determined by the forget gate, expressed as:

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \tag{2.6}$$

The output gate works as a decision mechanism in LSTM. This mechanism is for the memory cell to accept or reject its passage. The calculation of h_t is illustrated by the following equations.

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_f) \tag{2.7}$$

$$h_t = o_t tanh(c_t) \tag{2.8}$$

2.2.3. Hybrid CNN-LSTM preprocessed by PSO

This section explains the convolutional LSTM networks integrated into the proposed model architecture, designed to develop an effective deep learning technique for detecting Control Performance Attacks (CPA) and anomalies in SG networks.

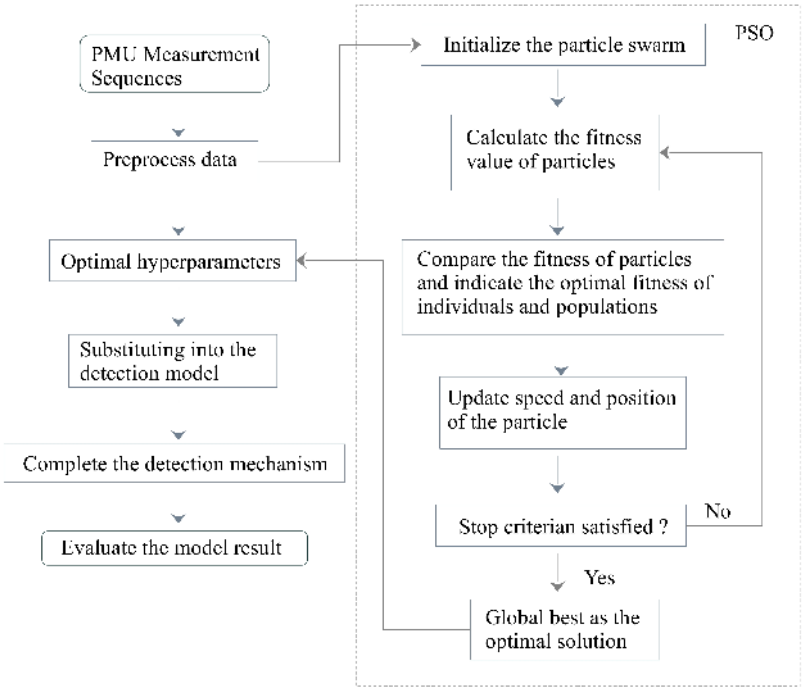


Figure 2.5. *The workflow of the PSO-based CPA detection algorithm*

The steps of the PSO algorithm are outlined in Table 2.1. A fitness value calculation is conducted to determine an appropriate value range or an accurate value for the Br hyperparameter. Hyperparameter optimization is carried out to enhance the accuracy of the erroneous data detection algorithm. During the training of the deep learning model, the development of its ability to learn the properties of the selected data sample is observed.

Table 2.1. *Steps of the attack detection method*

Algorithm
1. Start initialization of the parameters and determine the system topology
2. Preprocess the dataset
3. Set error as the fitness function using (2.14)
4. Update speed and position of the particle using (2.12) and (2.13)
5. Determine Root Mean Square Error (RMSE) of the input and training data
6. If solution converges
Perform step 7
Else
Perform step 4
7. Set the input weights of the CNN-LSTM network model
8. If error < personal min error
Stop the criteria
Else
Adjust the weights of the CNN-LSTM network model and perform step 8
9. End

PSO is a well-established metaheuristic algorithm inspired by the collective movement of schooling fish and flocking birds. In deep learning approaches, selecting the optimal set of hyperparameters is crucial for achieving the best results. However, this process often requires intensive effort. Manual calculations are generally not effective for guiding improvements and ensuring convergence from a steady-state condition in an algorithm.

Using the PSO model for the training phase, hyperparameters are automatically optimized. This method enhances the CNN-LSTM architecture by adjusting its hyperparameters through swarm intelligence. A swarm consists of many particles, each considered a candidate solution. A cluster of n particles is represented as S, as expressed below [74]:

$$S = x_1, x_2, ..., x_n \tag{2.9}$$

x_i is the i^{th} particle among the particles society. The ${i_1}^{th}$ particle is in the p_{i_1} position in the space and the representation of the position vectors shown below:

$$P_i = p_{i_1}, p_{i_2}, ..., p_{i_n} \tag{2.10}$$

and the velocity V_i varies through time as expressed in the following:

$$V_i = v_{i1}, v_{i2}, ..., v_{in} \tag{2.11}$$

The special velocity vector $v_{ij}(t+1)$ in the search space is used in the next iteration to calculate both the direction and velocity of a particle in the search space. The current position of a particle is determined by the position $x_{in}(it)$. The best position of a particle is calculated with the help of the personal best vector $p_{il}(t)$. The global best storage $p_{ig}(t)$ is the parameter determined as the best location on the herd. As a result of each iteration, the personal best vector and global best vector are updated for each particle. The update is performed based on the vectors ((2.12) and 2.13), respectively, of velocity and position.

$$V_{ij}(t+1) = wv_{ij}(t) + c_1 r_1(p_{il}(t) - x_{ij}(t)) + c_2 r_2(p_{ig}(t) - x_{ij}(t)) \tag{2.12}$$

P_i can move according to the following equation:

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1) \tag{2.13}$$

The movement of a particle at time $t+1$ is represented by $x_{ij}(t+1)$. The acceleration coefficients c_1, c_2 are within the range (1, 5), and the random variables r_1, r_2 are within the range (0, 1). To prevent the particle from moving out of the search space, an inertia weight constant w within the range (0.4, 0.9) is used in calculating the particle's speed.

Each individual in the system has information used to evaluate their positions and velocities within the search space. The system makes optimal decisions based on the current state of these individuals. To determine the best position, particle fitness is utilized as an evaluation metric for all individuals. The training time limit of the detection approach influences the selection of the fitness function. Considering this, root mean square error (RMSE) is chosen as the fitness function, as shown below:

$$Fitness = \sqrt{\frac{1}{N}\sum_{t=1}^{N}(x_t - \widehat{x_t})^2} \tag{2.14}$$

As seen in (2.14), the real and fitted values of the samples are denoted as x_t and $\widehat{x}_t$, respectively, for a number of N validation samples. After each iteration, the ordinary, local optimal, and global optimal particles are determined.

3. CYBER-PHYSICAL ATTACK DETECTION IN MG ENVIRONMENT

3.1. Problem Description

3.1.1. SG infrastructure

The IEEE 123 bus network is selected as environment I for the purposes of this research. Additionally, data regarding the network connection pattern and data migration scheme are collected from [75]. Case study networks and load data are passed through the optimized hyperparameters PSO algorithm by transmitting the decision that the agent behavior will make in the history set based on the selected environment and past actions to the algorithm, and the missing and erroneous behaviors of the agent are eliminated after the second optimization method and the strategies of CNN-LSTM agents are also determined. They apply their idealized decisions and axioms, optimized by DE. The number of agents in Environment I is more than the number of agents in Environment II. In addition, the faulty data received by Environment III from Environment I and II is a compelling element in the decisions of agents and the successful classification of LSTM-CNN agents in the classification [27] problem, which constitutes the basis that DE optimization must fulfill, and the physical attacks and system failures that are different from the cyber attacks successfully detected by defensive agents.

As shown in Table 3.1., the model identifies the PMU located on bus 1 as a potential attack target to compromise PDC-3 across various waiting time values. The attack vector, which varies based on waiting time, leads to PDC-3 dropping measurements from other PMUs, specifically the PMU situated on buses 1-5. Consequently, this measurement drop results in buses 1, 2, 3, 7, and 149, as identified by the PMU, being unobservable. Any additional delay in the network (delays exceeding 30ms) is categorized as a network, PMU, or PDC delay, owing to the minimal values assigned to network delays.

In this section, the integration of players' attack-defense competition and the zero-sum DE-based MG model are first designed. The proposed model aims to optimize the moves and strategies of the attacker and defender with differential

equations and DE.

Table 3.1. *Coordinated cyber-physical attacks for a second*

Cyber attack		Physical disturbances		
PDC	Target	Attack vector	Observed Buses	Operation Type
10ms	1, 3	Fault from 9% on L_{1-149}	1, 2, 3, 7, 149	SW_3 fault
30ms	2, 3	Fault from 17% on L_{8-13}	7, 8, 9, 12, 13	SW_4 fault
20ms	3, 3	Fault from 25% on L_{3-4}	1, 3, 4, 5	Relay fault
30ms	4, 3	Fault from 35% on L_{3-5}	3, 5, 6	Normal operation load changes
30ms	5, 3	Fault from 19% on L_{15-17}	15, 16, 17, 34	L_{15-16} maintenance
30ms	6, 2	Fault from 37% on L_{14-11}	9, 10, 11, 14	L_{14-10} maintenance
30ms	7, 2	Fault from 19% on L_{18-19}	18, 19, 20	SW_2 fault
30ms	8, 2	Fault from 19% on L_{21-22}	18, 21, 22, 23	SW_2 fault
10ms	9, 2	Fault from 43% on L_{23-24}	21, 23, 24, 25	L_{23-24} maintenance
20ms	10, 2	Fault from 55% on L_{27-33}	26, 27, 33	Normal operation load changes
30ms	11, 2	Fault from 65% on L_{26-31}	26, 31, 32	Normal operation load changes
20ms	12, 2	Fault from 45% on L_{25-28}	25, 28, 29	Normal operation load changes
30ms	13, 2	Fault from 72% on L_{29-30}	29, 30, 250	SW_1 fault
30ms	14, 1	Fault from 77% on L_{44-47}	47, 48, 49	Normal operation load changes
10ms	15, 1	Fault from 63% on L_{50-51}	50, 51, 151	SW_9 fault
30ms	16, 1	Fault from 63% on L_{44-45}	44, 45, 46	Normal operation load changes
30ms	17, 1	Fault from 50% on L_{44-42}	41, 42, 43, 44	Normal operation load changes
20ms	18, 1	Fault from 68% on L_{35-40}	35, 40, 41, 42	L_{41-40} maintenance
30ms	19, 1	Fault from 78% on L_{37-36}	35, 36, 37, 38	L_{36-38} maintenance
30ms	20, 1	Fault from 75% on L_{38-39}	38, 39	Normal Operation load changes
10ms	21, 4	Fault from 75% on L_{53-54}	52, 53, 54	SW_6 fault
30ms	22, 4	Fault from 75% on L_{55-56}	54, 55, 56	SW_6 fault
30ms	23, 4	Fault from 70% on L_{59-58}	57, 58, 59	L_{57-58} maintenance
30ms	24, 4	Fault from 49% on L_{62-60}	57, 60, 61, 62	SW_7
20ms	25, 4	Fault from 76% on L_{62-63}	62, 63, 64	L_{63-64} maintenance
30ms	26, 4	Fault from 92% on L_{64-65}	64, 65, 66	L_{64-63} maintenance
30ms	27, 5	Fault from 87% on L_{95-96}	95, 96	SW_5 fault
30ms	28, 5	Fault from 90% on L_{94-93}	91, 93, 94, 95	SW_5 and SW_6 fault
10ms	29, 5	Fault from 90% on L_{91-92}	91, 92	Normal operation load changes
30ms	30, 5	Fault from 75% on L_{89-90}	89, 90	Normal operation load changes
30ms	31, 5	Fault from 90% on L_{88-87}	86, 87, 88, 89	L_{87-89} maintenance
20ms	32, 8	Fault from 90% on L_{81-82}	81, 82, 83	Normal operation load changes
30ms	33, 8	Fault from 90% on L_{85-84}	81, 84, 85	L_{81-84} maintenance
30ms	34, 8	Fault from 80% on L_{77-78}	77, 78, 79, 80	L_{79-78} maintenance
10ms	35, 7	Fault from 90% on L_{76-72}	67, 72, 73, 76	L_{67-72} maintenance
30ms	36, 7	Fault from 89% on L_{74-75}	73, 74, 75	Normal operation load changes
20ms	37, 7	Fault from 89% on L_{67-68}	67, 68, 69	Normal operation load changes
30ms	38, 7	Fault from 10% on L_{69-70}	69, 70, 71	Normal operation load changes
30ms	39, 7	Fault from 75% on L_{197-97}	68, 97, 98, 197	L_{67-97} maintenance
30ms	40, 7	Fault from 80% on L_{99-100}	99, 100, 450	SW_8 fault
10ms	41, 6	Fault from 90% on $L_{102-103}$	102, 103, 104	Normal operation load changes
30ms	42, 6	Fault from 10% on $L_{106-107}$	105, 106, 107	Normal operation load changes
30ms	43, 6	Fault from 80% on $L_{105-108}$	105, 108, 109, 300	SW_9 and SW_{10} fault
30ms	44, 6	Fault from 80% on $L_{110-111}$	109, 110, 111, 112	Normal operation load changes
30ms	45, 6	Fault from 10% on $L_{112-113}$	112, 113, 114	$L_{111-110}$ maintenance

*Physical disturbances such as line faults, switch faults, and load changes inserted before the MG

The solution to the security problem in the electrical network requires coordination with communication networks and intervening in changes and disruptions in the data flow as quickly as possible, thus aiming to ensure safe data flow, reduce power outages, minimize investment costs and improve reliability indices. For this purpose, having both aggressive and protective agents in the environment makes it easier to reveal system deficiencies and strengths. On the other hand, security problem resolution ensures distribution network integrity by optimizing protection device parameters.

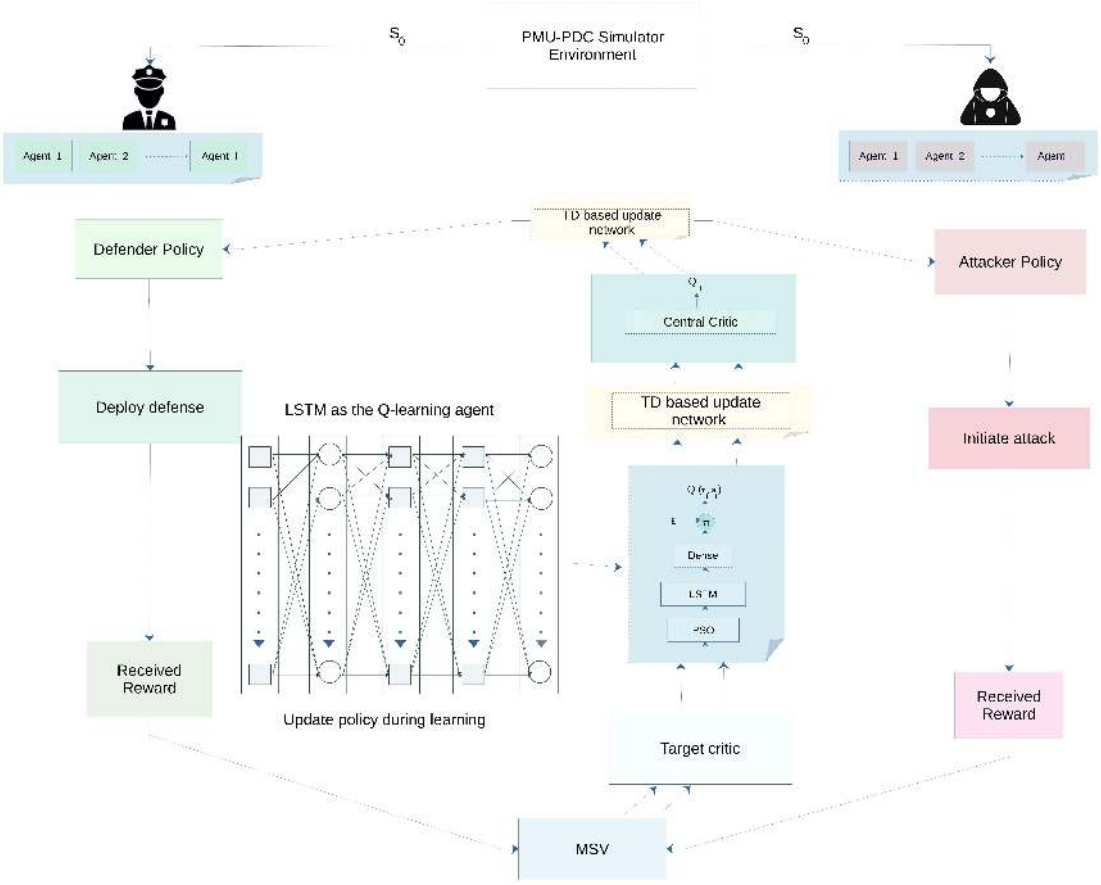


Figure 3.1. *Schematic presentation of MG workflow on simulated system*

A solution from the upper level, that is, the communication of agents present in each protection and PMU equipment used in the distribution network, is sent to the lower-level program to optimize the protection agent parameters and ensure proper coordinated operation between them. If the internal task cannot reach a suitable setup, such a method identifies the protection agent that most disrupts low-level detection and removes it from the solution. The lower-level program then runs again, and this process is repeated iteratively until a suitable solution is reached for the external task.

In general, the estimation of policy indices plays a major role in the study of the coordination and selectivity function of the proposed conservation philosophy. Incorrect strategy selection encourages worse estimation of reliability indices than using the previously mentioned philosophy and causes agents to aim at the

wrong target and cause miscoordination. Agent strategies are reached to the optimum target by applying two-stage optimization in line with the importance of the problem.

3.1.2. SG modeling

Reliability indices such as Environment I, II and III are based on different perspectives. I is based on the frequency of physical disturbances in interruptions and remote controlled circuit breakers, transmission line breaks between busbars; This addresses the more important issue of whether such indicators are due to an attack or a systemic failure, as transient faults are more common in distribution systems. For example, attack agents disrupt coordination and increase the frequency of outages, compromising the power flow between buses. Environment I includes the physical main cause of the outage; the information obtained from here is of great importance; This means that permanent faults have a more significant impact on these state environments II and III. For example, improper coordination between overcurrent relays during permanent faults can cause defensive agents to deviate from targeting the fault point, depending mainly on the duration of the outage. It contains a vulnerability to attack agents based on Periphery I, II and III connections. For this reason, defined environments may exhibit different behaviors depending on the system being examined and the fault/attack characteristics of the system.

3.1.3. Data flow analysis

The environment space encompasses the different layers of the SG network and game policy, which involve complex coordination in the MARL game. To preserve the integrity of the real measured data within the system, considerations are given to the vulnerability of each node and link in the network, as well as the dual role played by each player. In the face of attackers, defenders strategically select endangered nodes and transition to their protection. This study provides an observation of the confrontations between attackers and defenders. Based on their respective policies, each attacker updates its knowledge about the environment and the type of defender after observing the system. The agents in our model interact as defenders and attackers. Subsequently, the MARL game is formulated, along with its variant. In the following sections, we first introduce the SG network operation model and the

formulated ISQV-MARL approach. Specifically, we jointly observe the coordination of the operations of the SG system and the decision-making mechanism of the agents, based on policy updates and rewards received by the agents.

3.1.4. Design of synchrophasor communication network

To test and observe the proposed approach, the IEEE 123 Bus Test Distribution Feeder, detailed information given in [76], is selected as a Test system using data from New York Independent System Operator (NYISO) [75]. Timestamped measurements obtained by time-synchronized devices known as PMUs include bus voltage and current phasors under normal, attack, and natural event conditions. Through this data, real-time dynamic monitoring, control and protection of SG networks is achieved. The PyPMU-PDC simulator, based on the IEEE C37.244-2013 standard in Python, with the ability to send realistic PMU data streams to PDC receivers containing phasor data corresponding to a wide range of scenarios and situations such as power grid events, synchrophasor commands, etc., forms the primary environment in the game to be defined. The synchrophasor module is still under development, but it contains useful tools to carry out successful research and studies [77].

In this section, we provide an overview and workflow of the proposed attack-defense game based on ISQV-MARL approach, followed by a detailed explanation of each key step. Initially, we consider the FDIA problem within the SG context. The game environment comprises 45 different PMUs and their corresponding data storage unit, known as the PDC, as depicted in Figure 3.2. The proposed approach is employed for the prior identification of FDIA detection. As mentioned earlier in the literature review, the usage of MDP-based game RL models is generally less preferred due to their computational expense. Instead, table learning approaches are commonly favored. The agent in our proposed approach recursively explores the environment and makes decisions to maximize its rewards. The agent's goal is to determine the optimal path and actions to achieve the best possible outcome.

The game space comprises two environments: PMU, and PDC spaces, as depicted in Figure 3.2. In our analysis, we primarily focus on the physical structure and measured parameters of PMUs, without sacrificing generality. To elaborate

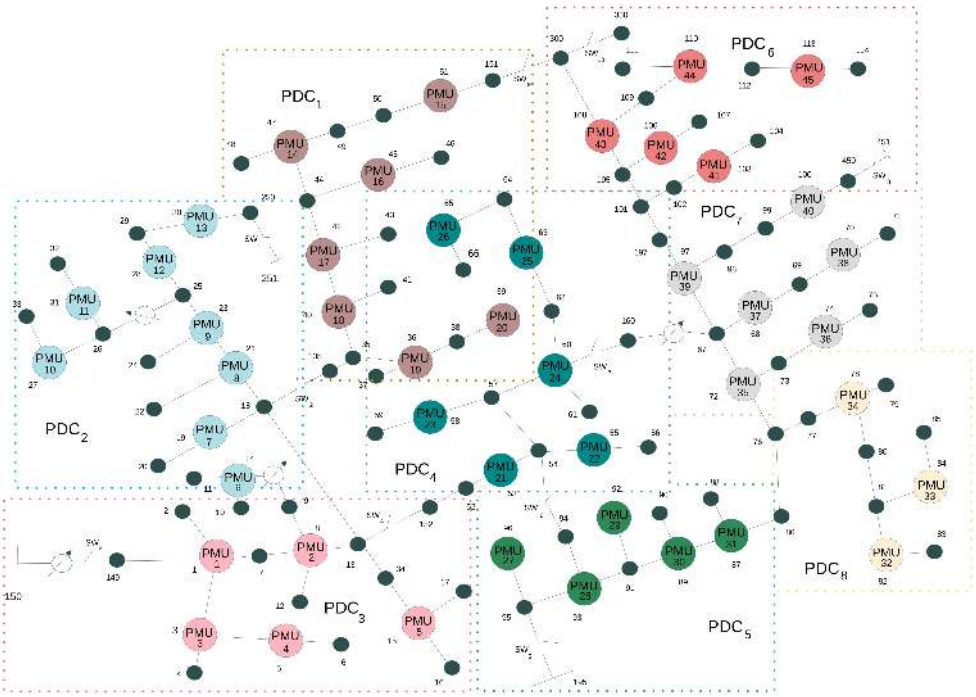


Figure 3.2. *Schematic presentation of simulated system*

further, the attacker manipulates the measurements within these spaces by injecting false data. Each attacker is driven by the objective of gaining control over the macro-environment by targeting and infiltrating each sub-environment[1].

3.1.5. Spoofing signals to PMU-PDC layers

Ensuring a continuous flow of data from PMUs to connected PDCs provides system controllers with comprehensive monitoring capability and it is of great importance in ensuring system stability and traceability. As shown and used in [78], the observability of the SG system is expressed as:

$$O_k = \sum_{k \in N_k, l \in N_l, m \in N_m} c_{klm} x_l \; k \in N_k \tag{3.1}$$

Observability is essential for system analysis and control in the SG. The observability function O_k depends on the connection parameter c_{klm} and the binary

[1]Sub-environment means each PMU connected to its belonged PDC that is a part of total macro environment including all BUS, PDC and PMU units.

variable x_l, which indicates whether a PMU on bus k of the network is placed on bus l. N is a set of nodes $N = \{N_k, N_l, N_m\}$, the set of PMUs placed in the network denoted as N_l, the set of PDCs placed in the grid denoted as N_m, the set of buses for the system represented by N_k.

$$c_{klm} = \begin{cases} 11 & \text{if } k = l = m \text{ for } \{k, l, m\} \in N \\ 10 & \text{if } k \neq l = m \text{ for } \{k, l, m\} \in N \\ 01 & \text{if } k = l \neq m \text{ for } \{k, l, m\} \in N \\ 00 & \text{otherwise} \end{cases} \tag{3.2}$$

If the data flow from the buses $O_k \geq 1$ for all buses, it indicates an observable situation. Observability of bus_k is ensured either through the PMU located directly on bus_k or via a PMU located on bus_l that is connected to bus_k. This holds true for every data flow from the busbars to the PMUs.

In the second and third environments, the attacker's main goal is to minimize the observability of the network by injecting the attack vector to take the targeted buses and measurements out of control, as stated in (3.3).

$$Min \sum_{k \in N_k} o_k + Max \sum_{k \in N_k} u_k \tag{3.3}$$

(3.1) gives the calculation of the observability of a data path. Additionally, the current measurements obtained from a PMU and connected buses located on the bus_k have a significant impact on the calculation of the observability variable O_k for each bus. The observability calculation of each bus in the SG is given in (3.4).

$$O_k = \sum_{k \in N_k, l \in N_l, m \in N_m} c_{klm} x_l v_l \tag{3.4}$$

Considering (3.15), observability calculations are performed to determine the validity of PMU measurements. The observability of the buses is assessed based on the busbar, PMU, and PDC points, as described in (3.5). Here, M represents a set

of large positive numbers, containing M_1, M_2, and M_3 for the Bus, PMU, and PDC units, respectively.

$$o_k \geq \frac{O_k}{M_1} \text{ and } o_k \leq O_k \; \forall k \in N_k \tag{3.5}$$

The defined environments including (123-Bus, 45 PMUs and 8 PDCs) are expressed in (3.6). The surface for attack operations covers respectively three environments considering y_k for $k \in N_k$ for Buses, y_l for $l \in N_l$ for PMUs, and y_m for $m \in N_m$ for PDCs.

$$AN = \begin{cases} \sum_{k \in N_k} y_k = 123 \\ \sum_{l \in N_l} y_l = 45 \\ \sum_{m \in N_m} y_m = 8 \end{cases} \tag{3.6}$$

Each link on the compromised communication path experiences a delay, denoted as σ_l^m, which is calculated as follows:

$$\sigma_l^m = t_p + t_t + t_q + t_B \tag{3.7}$$

For each time slot t, the reasons for physical attacks are determined based on the bus delay t_B. Bus delay based on bus data flow time factor β_{y_k} and its threshold β_{Th} is expressed as follows:

$$t_B = \begin{cases} 0, & \beta_{y_k} \leq \beta_{Th} \\ 1, & otherwise \end{cases} \tag{3.8}$$

Where transmission delay t_t $(m \leftrightarrow l)$ calculated based on the fraction of the flow from PMU_u on edge $(m \leftrightarrow l)$ and the capacity of the communication link $C_{m \leftrightarrow l}$ seen as follows:

$$t_t = \frac{f_{m \leftrightarrow l}}{C_{m \leftrightarrow l}} \tag{3.9}$$

The queuing delay at node i, through which the total flow $f_{m \leftrightarrow l}$ on the link path is transmitted on link $m \leftrightarrow l$ and determined by the traffic behavior is expressed as follows:

$$t_q = \frac{1}{ms - \rho_{m \leftrightarrow l}} \tag{3.10}$$

The mean service speed ms is the number of packets processed per unit time and $\rho_{m \leftrightarrow l}$ varies depending on the port speed and the average traffic speed to that port. A function of the flow protection variables y_l is also expressed as $\rho_{m \leftrightarrow l}$. The calculation and evaluation of t_p propagation delay is considered as the distance between nodes and the speed of light in the communication medium. Additionally, end-to-end delay is calculated as follows:

$$\varrho_l^m = \sum_{l \in N_l, m \in N_m} y_l \sigma_l^m \tag{3.11}$$

In the proposed games, the network delay between the targeted PMU and the receiving PDC is calculated as shown in (3.11) to accurately determine the specified time window, ensuring secure data flow from the PMU to the PDC. This framework enables the defensive agent to reliably estimate the available time interval for maintaining secure data transmission.

As an attacker, intruding on the time-stamped measurements of the PDC is crucial for seizing the targeted nodes of information flow from PMU_l to $\text{PDC}m$. The reference internal alert conditions, based on the time-stamp (as given in (3.12)), identify a future point in time. When the measurements do not include time-stamped measurements, the attacker can more easily compromise the targeted nodes.

$$\text{AV} = \begin{cases} z_k \leq M_1 min(\sigma_l^m)\ \forall k \in N_k\ \forall l \in N_l, \forall m \in N_m \\ z_l \leq max(\sigma_l^m)\ \forall l \in N_l, \forall m \in N_m \\ z_l \leq M_2 y_l \text{ and } z_l \geq \frac{y_l}{M_2}\ \forall l \in N_l \\ z^m \geq y^m M_3 \text{ and } z^m \leq y^m\ \forall m \in N_m \\ z^m = \sum_{l \in N_l} y_l z_l\ \forall m \in N_m \end{cases} \tag{3.12}$$

By injecting an attack vector into a compromised PMU, the PMU's timestamps are manipulated. The attack vector operates within a selected range of values to conceal the attacker's presence. Compromised PDCs are assigned non-zero attack vectors, whereas unaffected PDCs have empty attack vectors. The computation of the attack vector z^m against PDC_m depending on the expressions (3.11) is evaluated as a binary variable.

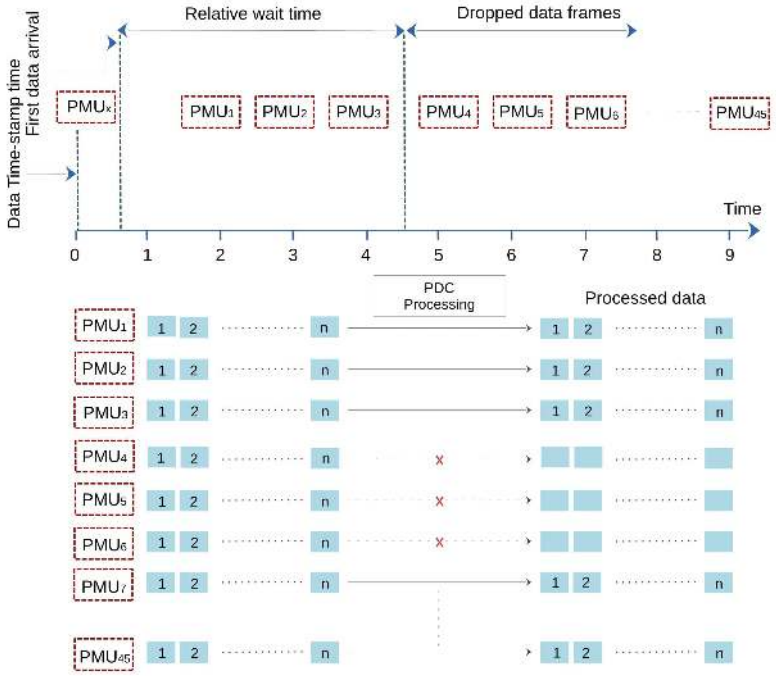


Figure 3.3. *Relative data alignment as defined by C37.244-2013*

The communication process that facilitates data flow between PMU and PDC

units is represented by the time frame (seen in Figure 3.3), which defines a specific time period, as explained in (3.13). It is crucial to distinguish between the measurement arrival time from PMUl to PDCm and the time it takes for the measurement to traverse the communication network between them. Differentiating the inherent time loss in the data flow from the time loss caused by an attack is vital for detecting such attacks. Specifically, time losses are formulated by considering the time required for data to traverse the communication network between PMU and PDC units, as described in (3.13). Figure 3.3, based on (3.13) [78], accounts for measurements from other PMUs to the corresponding connected PDCs, including delays in time-stamped measurements from an attacked PMU.

$$d_l^m = \varrho_l^m + \tau^m - z^m - \sigma_l^m \ \forall l \in N_l \text{ and } \forall m \in N_m \tag{3.13}$$

The validity of PDC measurements is determined by four criteria. For the identification and classification of attacks, these criteria are evaluated using four equal values. The attack type is characterized based on the criteria set for permanent switching errors, PDC delay, PDC under attack, and PMU under attack.

- BUS under physical attack: If the time window d_l^m (which represents the duration of the measurements of a PDC receives from the buses, as seen in 3.13) is less than a threshold τ_1, then the measurement delays become invalid. In this case, measurements sent from PMUs cannot be made available to the PDC timer, and the variable v_l^m is assigned a value of 0.

- Communication network latency, PDC latency, and PMU measurement reporting latency: If the time window d_l^m (which represents the duration of the measurements of a PDC receives from the buses, as seen in 3.13) is less than a threshold τ_2, then the measurement delays become invalid. In this case, measurements sent from PMUs cannot be made available to the PDC timer, and the variable v_l^m is assigned a value of 0.

- PMU is under attack: The equations indicate that the PDC is used to determine whether a measurement received from an under-attack PMU has a spoofed timestamp. This can be resolved through the value assigned to the

variable y_u in (3.14), which validates the measurements transferred from PMU_l to PDC_m. When $y_l = 1$, corresponding to the PMU value under attack, the variable v_l^m in (3.14) is assigned the value 1.

- PDC is under attack: The attack status of the PDC is monitored using the z^m variable in (3.13), which verifies the measurements sent from PMUl to PDCm. When $z^m = 0$, indicating that the PDC is not under attack, the variable v_l^m in (3.13) is assigned the value 1. However, if $z^m = 1$, the transmitted measurement is evaluated based on its arrival time at a hacked PDC that receives phasors from the targeted PMU.

$$v_l^m \geq \frac{d_l^m(1-y_l)y^m}{M_3} + \epsilon \text{ and } v_l^m \leq \frac{d_l^m(1-y_l)y^m}{M_3} + 1 \ \forall l \in N_l, \forall m \ N_m \tag{3.14}$$

$$v_l^m \leq v_l \text{ and } v_l \leq \sum_{m \in N_m} v_l^m \ \forall l \in N_l, \forall m \ N_m \tag{3.15}$$

For PDCs that collect phasor measurements from PMUs in the system, the validity criteria of the measurements are crucial for tracking and detecting attack data. For each PDC, the validity of a measurement from PMU_u, on which the system's observability is based, is determined using (3.14).

The coordinated attack involves physical disturbances, disruption of system observability, and manipulation of PMU measurements. In the previous equations, we explained how system observability disruption occurs. The attack vector described above is applied to the PMUs placed in the SG system, resulting in a loss of system observability.

3.1.6. Spoofing effect on measurement

The PMU provides time-synchronized measurements in complex SG networks at a rate of 60 samples per second. For synchronization, GPS clocks are used to generate synchronization pulses. Simultaneous sampling of measurements is based on assigning a timestamp to each signal received by the PMU [80]. An attacker targeting the PMU changes the receiver time by shifting it from t to $t + t_k$. The

t_k parameter specifies the time difference caused by the attacker agents. The phase angles of post-attack measurements are expressed as follows:

$$\phi^{attacked} = \theta_k + \phi \tag{3.16}$$

The described spoofing attack results in a manipulated phase angle $\phi^{attacked}$, causing a phase shift θ_k. In coordinated attacks, the current signals of the n^{th} branch, represented by the phasor current $i_n(t)$, and the voltage signal $v_n(t)$, are misleading.

$$i_n^{attacked}(t) = |i_n^{attacked}|sin(wt + \theta_{I_n} + \theta_k) \tag{3.17}$$

The parameters contained in the signals include the current magnitude $|i_n|$, the voltage magnitude $|v_n|$, the angular frequency w and the original phase angles before the attack, θ_{I_n} and θ_{v_n}. The effect of the spoofing attack on the phasor signals is formulated as follows:

$$v_n^{attacked}(t) = |v_n^{attacked}|sin(wt + \theta_{v_n} + \theta_k) \tag{3.18}$$

A spoofing attack targeting PMU units specifically affects the phase angles in phasor measurements. All signals received from a PMU impacted by the spoofing attack exhibit a consistent shift in phase angles. However, this GPS spoofing attack does not influence the complex power S_n of the nth branch connected to the bus. This is because the calculation of complex power, which includes reactive power Q_n and active power P_n, does not depend on the phase angle in the same way. Mathematically, this condition is expressed as follows [81]:

$$S_n = P_n + jQ_n \tag{3.19}$$

$$S_n^{attacked} = P_n^{attacked} + jQ_n^{attacked} = V_n I_n \angle(\theta_{V_n} - \theta_k - \theta_{I_n} + \theta_k) \tag{3.20}$$

The complex power equation $S = \sum_{n=1}^{N_b} v^{attacked} I_n^{attacked*}$ for each branch connection N_b to a bus is obtained. GPS spoofing attacks do not affect complex power; therefore, in this case, complex power does not provide any indication of the attack.

3.2. Markov Convex Game (MCG)

At the beginning of the MG design, 10-tuple $T_p = < S, D, A, T_{s^a s'}, L, \Re_{s^a s'}, \mu, E, p_0, F >$ is defined [35]. The environment E is considered fully observable when a finite set of states $S := s_1, s_2, ..., s_L$ completely describes the set of world states W [82]. All players in coalition games are represented by the set $[L] := 1, 2, ..., L$. For the two defined sets of players, the set of joint actions for the defending agents is denoted as $D = \mathrm{x}_{i \in L}\ D_i$, and the set of joint actions for the attacking agents is denoted as $A = \mathrm{x}_{i \in L}\ A_i$.

Additionally, the transition probability $T : W \times A \times S \hookrightarrow [0, 1]$ provide for action $a \in A$ and each state $s \in S$, the conditional probability[2] $T^a_{ss'}$ of transitioning to state $s^{'} \in S$ given that the MDP is takes action a in state s.

$$T^a_{ss'} = \mathbb{P}[s^{t+1} = s^{'} | s^t = s, a^t_i = d, a^t_{-i} = a] \tag{3.21}$$

$$\mathbb{P}[s^{t+1}|s^t] = \mathbb{P}[s^{t+1}|s^1, ..., s^t, A^t, D^t] \tag{3.22}$$

A coalition structure $CS = C_1, ..., C_l$ is defined for the set L of all agents in the game. Each $C_i \subseteq L$ denotes a coalition, where each coalition C_i is a subset of the set L of all player agents[3].

$$V^{\mu_C}(\mathbf{s}) = \mathbb{E}_{\mu_C}[\sum_{t=0}^{\infty} \gamma^t r_k^D(s)^{t+1}(\mathbf{s}, a_c, d_c) | S_t = s] \tag{3.23}$$

For each set of players, the coalition value is given by (3.23), the GC value $V^{\mu}(\mathbf{s})$, and the value of the empty coalition $V^{\mu_\varnothing}(\mathbf{s})$ is based on the equivalence proof

[2] $\mathbb{P}$ is the notation of the probability function

[3] Based on the assumption that $C_m \cap C_n = \varnothing$ for all $C_m, C_n \cap L$, where C_u represents the union of C_m and C_n.

from [83]. Determining a coalition structure (CS) bundle $< CS, (max_{\mu}(x_i(s)))_i \in L >$ gives the solution of MCG. The payoff distribution, such as the credit allocation, for the optimal joint policy assigned to a CS, is represented by $(max_{\mu}(x_i(s))_{i \in L}$. The state of the MG is given by $\underset{\mu_{C_n}}{max} V^{\mu_{C_n}}(s', a_i) + \underset{\mu_{C_m}}{max} V^{\mu_{C_m}}(s', a_i) \leq \underset{\mu_{C_u}}{max} V^{\mu_{C_u}}(s', a_i)$ based on the assumption[3].

$$\Re_m^A(s, A, D) = \underbrace{m_1 u_1(s)}_{\text{compensation level}} + \underbrace{m_2 u_2(A)}_{\text{spoofing cost}_{PMU \to PDC}} + \underbrace{m_3 u_3(A)}_{\text{spoofing cost}_{PDC \to PMU}} \tag{3.24}$$

$$+ \underbrace{m_4 u_4(D)}_{\text{encrypting cost}_{PMU \to PDC}} + \underbrace{m_5 u_5(D)}_{\text{encrypting cost}_{PDC \to PMU}}$$

Each player's strategy depends on the states and the stationary mixed strategies of the agents. As defined in [84], a cluster of strategies for defenders is as follows:

$$f^D(s) = [f_1^D(s), f_2^D(s), ..., f_{L_D}^D(s)]^T \tag{3.25}$$

The defender secures the transmitted data by adding unpredictable signals to the communications between the PMU and PDC in the network. The reward value r_m^D in the m^{th} stage of the MG is expressed as follows:

$$r_m^D(s) = (f^A(s))^T \Re_k^D(s) f^D(s) \tag{3.26}$$

The past action sets ($C_m^A = A_0^i, A_1^i, ..., A_{m-1}^i$ for the attacker $C_m^D = D_0^i$, D_1^i, . . . , D_{m-1}^i for the defender) are defined until the m^{th} stage for L-stage MG. For a defender player, the expected sum of the discounted reward $V^D(s)$ is defined as follows:

$$V^D(s) = \mathbb{E}\{\sum_{m=0}^{L} \gamma^m r_m^D(s, C_m^A, C_m^D)\} \tag{3.27}$$

γ^m represents the discount factor, capturing the current condition where the benefit or cost of one unit in time interval $m+1$ is valid only if $\gamma \leq 1$ in the previous time m. The expected sum of discounted rewards $V^A(s)$ for an attacker player is defined as:

$$V^A(s) = \mathbb{E}\{\sum_{m=0}^{L} \gamma^m r_m^A(s, C_m^A, C_m^D)\} \tag{3.28}$$

Payoff distributions are among the factors that determine the course of the game in calculating the optimal joint policy $(max_{\mu_i}\mu_i(s))i\epsilon L$, considering the coalition structure. A grand coalition is a set that includes all agents of a game group. The Markov core (MC) formulation, which is a solution concept that expresses stability in this coalition, is expressed as follows:

$$MC = (\underset{\mu_i}{max} x_i(s)_{i\in L}) | \underset{x_c}{max}(s|C) \geq \underset{x_c}{max} V^{\mu_c}(s), \forall C \subseteq L, s \in S \tag{3.29}$$

3.2.1. Markov shapley value (MSV)

The GC is constructed gradually through the permutation of agents, utilizing cooperative game theory. The credit reflecting the contribution of a single agent represents the marginal contribution. In the multi-agent coalition game with an array of agents $< j_1, j_2, ...j_{|L|} >$, where $j_l \in L$ for all j_l. L represents the grand coalition for $n \in 1, ..., |L|$ and $j_a \neq j_b$ if $a \neq b$, the marginal contribution of an agent i is expressed as follows:

$$\psi_i(s|C_i) = \underset{\mu_{c_i}}{max} V^{\mu_{c_i \cup i}}(s) - \underset{\mu_{c_i}}{max} V^{\mu_{c_i}}(s) \tag{3.30}$$

The intermediate coalition $C_i = j_1, ..., j_{l-1}$ for $j_l = i$ describes where the ith agent can participate in the GC structure. The marginal contribution of the ith agent is expressed as follows:

$$\psi_i(s, a_i, d_i|C_i) = \underset{a_{c_i}}{max} Q^{\mu^*_{c_i}}(s, a_{c_i \cup i}, d_{c_i \cup i}) - \underset{a_{c_i}}{max} Q^{\mu^*_{c_i}}(s, a_{c_i}, d_{c_i}) \tag{3.31}$$

$$V_i^{\psi}(s) = \sum_{C_i \subseteq L \backslash \{i\}}^{\infty} H.\psi_i(s|C_i) \tag{3.32}$$

In a coalition (C) of players L, for a given player i, where $L\backslash\{i\}$ denotes the set of players excluding i, the coefficient H is calculated as $H = \frac{|C_i|!(|L|-|C_i|-1)!}{|L|!}$.

In the proposed MCG, the Shapley value, a solution concept in cooperative game theory, is used to solve the credit allocation problem. The Markov Shapley Q-value (MSQV) is based on the integration of the MSV (expressed in 3.32), which provides the weighted average of the marginal contributions. This is represented as $\psi_i(s, a_i, d_i|C_i)$ in (3.31).

Additionally, according to the probability distribution $Pr(C_i|L\backslash i) = \frac{|C_i|!(|L|-|C_i|-1)!}{|L|!}$, an agent can randomly join a formed coalition C to create a complete grand coalition.

$$Q_i^{\psi}(s, a_i) = \sum_{C_i \subseteq L \backslash \{i\}}^{\infty} H.\psi_i(s, a_i|C_i) \tag{3.33}$$

The main characteristics of the defined MSV are as follows:

(I): Defined specific covert movements of offensive and defensive agents are based on the Shapley-Bellman equation, formulated mathematically for multimedia scenarios.

(II): Efficiency is explained by the $\underset{\mu}{\max} V^{\mu}(s) = \sum_{i \in L} \underset{\mu_i}{\max} V_i^{\psi}(s)$.

(III): Identifiable counterfeit agents indicates that the MSV serves as a basic index to numerically describe the contributions of each player to their respective clusters, with the equation $V_i^{\psi}(s) = 0$ indicating the identification of fake players. This feature plays a crucial role in evaluating value factorizations.

(IV): Symmetry states that if two different agents are symmetrical, their optimal MSV values are equal.

3.2.2. Markov shapley Q-learning (MSQL)

In the approach that assumes an explicit model, where transition probabilities between states and targeted rewards serve as attack points and vulnerable points for agents, the value Q_i^{ψ} is determined within the policy framework by solving the Bellman optimality equation [83].

$$Q_i^{\psi^*}(s, a_i, d_i)) = \begin{cases} (\mathbf{w}(s,a) \sum_{s' \in S}^{\infty} \mathbb{P}\{s'|s,a,d\}[\Re + \\ \gamma \sum_{i \in N} \underset{a_i,d_i}{max} Q_i^{\psi^*}(s', a_i, d_i)] - \mathbf{b}(s) \end{cases} \tag{3.34}$$

The optimal $Q^{\psi*}(\mathbf{s}, a_i, d_i)$ seen in (3.34) is expressed as $\mathbf{w}(\mathbf{s}, \mathbf{a}, \mathbf{d}) = [w_i(\mathbf{s}, a_i, d_i)]^T \in \mathbb{R}_+^N; \mathbf{b}(\mathbf{s}) = [b_i(\mathbf{s})]^T \in \mathbb{R}_{\geq 0}^N;\ Q^{\psi*}(\mathbf{s}, \mathbf{a}, \mathbf{d}) = [Q^{\psi*}(\mathbf{s}, \mathbf{a}, \mathbf{d})]^T \in \mathbb{R}_{\geq 0}^L.\ d_i^* = \underset{d_i}{argmax} Q^{\psi*}(\mathbf{s}, a_i, d_i)$ is derived from the equations based on the solution $w_i(\mathbf{s}, a_i^*, d_i^*) = \frac{1}{L}$ for any $s \in S$. In the strict sense, the assigned credits for each environment are equal and each agent receives $\frac{Q^{\mu^*}(\mathbf{s},\mathbf{a},\mathbf{d})}{L}$ on condition of cost the optimal action performance. It should be emphasized that each agent in their environment receive their own credit based on an action selectivity of an i^{th} agent. External effects change agents' decisions or actions. Not well-desingned modifications of agents' actions which are preferred by the principal yield disutilities to that agent. However, $w_i(\mathbf{s};\ a_i, d_i)$ should be identified for $u_i \neq \underset{a_i,d_i}{argmax} Q_i^{\psi^*}(\mathbf{s}; a_i, d_i)$.

$\delta_i(s, a_i, d_i)$ depends on two conditions that are respectively $a_i = \underset{a_i,d_i}{max} Q_i^{\psi}(\mathbf{s}, a_i, d_i)$ and $a_i \neq \underset{a_i,d_i}{max} Q_i^{\psi}(\mathbf{s}, a_i, d_i)$. At first condition, $\delta_i(s, a_i, d_i)$ equals 1 and at second condition, equals $\alpha_i(\mathbf{s}, a_i, d_i)$. The expression of $\sigma_i(\mathbf{s}, a_i, d_i)$ is formulated as $|L|^{-1} w_i(\mathbf{s}, a_i, d_i)^{-1}$. The term b(s) is nullified and the condition $w_i(\mathbf{s}, a_i) = \frac{1}{L}$ when $a_i \neq \underset{a_i,d_i}{argmax} Q_i^{\psi}(\mathbf{s}, a_i, d_i)$ is adjusted in [85]. Based on the condition such that $\sum_{i \in N w_i}(\mathbf{s}, a_i, d_i)^{-1} b_i(\mathbf{s}) = 0$. For the convergence of SQ-Learning during the implementation process, the condition to $w_i(s; a_i, d_i)$[4] is used.

The mentioned security problem is transformed into a strategic partially observed Markov game (POMG) including PMU agents. The MG consists of a reward function $\Re_p$ for each agent p, a set of actions $A = A_1, A_2, ..., A_P$, a state transition function $T : S \times A_1 \times A_2 \times ... \times A_P \to \mu(S)$, and a set of states S. In this respect,

[4]Corresponding optimal joint deterministic policy and converging to the optimal MSQV are two important functions of Bellman operator for this game if $\underset{s}{max}\{\sum_{i \in L} \underset{a_i,d_i}{max} w_i(s, a_i, d_i)\} < \frac{1}{\gamma}$.

the configurations for PMUs are described by the states S. $E_1, ..., E_P$ denote sets of explorations for PMU 1 and PMU p, respectively. Each corresponding decision of PMU p is taken based on the E_p that is a part of the S.

3.2.3. Validity and interpretability

In this section, the performance of two agent clusters is evaluated using the multi-environment Shapley-Bellman equation algorithm and MSQL seen in Table 3.2.. This approach considers the diverse choices of action-value functions among the agents. The key insight of MSQL is that the full factorization of multi-environment Shapley-Bellman equation is necessary to extract decentralized policies. The marginal contribution for all actions is denoted as $\psi_i(s|a_i, d_i, C_i) = \sigma(s, a_{C_i \cup i}, d_{C_i \cup i})\ \widehat{Q}_i(s, a_i, d_i)$. The specified conditions are as follows:

$$\mathbb{E}_{C_i Pr(C_i|L\setminus\{i\})}\sigma(s, a_{C_i U\{i\}}, d_{C_i U\{i\}}) = \begin{cases} 1 & a_i, d_i = \underset{a_i,d_i}{max} Q_i^{\psi}(s', a_i, d_i) \\ K \in (0,1) & a_i, d_i \neq \underset{a_i,d_i}{max} Q_i^{\psi}(s', a_i, d_i) \end{cases} \tag{3.35}$$

For a condition $\alpha_i = \underset{a_i,d_i}{max}\widehat{Q}_i(\mathbf{s}, a_i, d_i)$, the expression $\widehat{Q}_i(\mathbf{s}, a_i, d_i) = Q_i^{\psi}(\mathbf{s}, a_i, d_i)$ is valid. For a condition $\alpha_i \neq \underset{a_i,d_i}{max}\widehat{Q}_i(\mathbf{s}, a_i, d_i)$, the equality $\widehat{\alpha}_i(\mathbf{s}, a_i, d_i)\widehat{Q}_i(\mathbf{s}, a_i, d_i) = Q_i^{\psi}(\mathbf{s}, a_i, d_i)\alpha_i(\mathbf{s}, a_i, d_i)$ is derived based on the study [85]. $\widehat{\delta}_i(s, a_i, d_i)$ depends on two conditions that are respectively $a_i = \underset{a_i,d_i}{max}\widehat{Q}_i^{\psi}(\mathbf{s}, a_i, d_i)$ and $a_i \neq \underset{a_i,d_i}{max}\widehat{Q}_i^{\psi}(\mathbf{s}, a_i, d_i)$. For first condition, $\delta_i(s, a_i, d_i)$ is accepted as 1. For second condition, $\delta_i(s, a_i, d_i)$ is calculated as $\widehat{\alpha}_i(\mathbf{s}, a_i, d_i)$ where $\widehat{\alpha}_i(\mathbf{s}, a_i, d_i) = \mathbb{E}_{C_i \sim Pr(C_i|L\setminus\{i\})}[\widehat{\psi}_i(\mathbf{s}, a_i, d_i;\ \mathbf{a}_{C_i}, \mathbf{d}_{C_i})]$ and $\widehat{\psi}_i(\mathbf{s}, a_i, d_i; \mathbf{a}_{C_i})] = \alpha_i(\mathbf{s}, a_i, d_i)\sigma(\mathbf{s},\ a_{C_i U\{i\}})$. A centralised system for managing 3 different environments is proposed using parametric function $\widehat{Q}_i(\tau_i; a_i, d_i)$ to directly approximate $Q_i^{\psi}(\tau_i; a_i, d_i)$. It also further motivates dynamically adjusting the above mentioned parameters, $\sigma_i(\mathbf{s}, a_i, d_i)$ is derived from these equations seen in (3.36). The detailed configuration of the defined network, along with other critical hyperparameters of the MG algorithm, are comprehensively listed in Table 3.3..

$$\widehat{\alpha}_i(\mathbf{s}, a_i, d_i) = \frac{1}{M}\sum_{k=1}^{M} F_s(\widehat{Q}_{C_i^k}(\tau_{C_i^k}, \mathbf{a}_{C_i^k}, \mathbf{d}_{C_i^k}), \widehat{Q}_i(\tau_i, a_i, d_i)) + 1 \tag{3.36}$$

3.2.4. Mathematical framework for MARL

The parameters determined for the defined MG are given in detail in Table **??**. In the MG , where the Q value is calculated with the optimized LSTM algorithm, the optimal strategies of the players are calculated as follows:

$$Q_{cent} = \sum_{i=1}^{l} Q_i(\tau_i, a_i, d_i) \tag{3.37}$$

The local representative formulated in the Recurrent Deep Q-Network constructed from a PSO-LSTM layer [27] and Multilayer Perceptron to solve the partial observability is represented as follows:

$$Q_i = g(\tau_i^a, \tau_i^d; h) \tag{3.38}$$

where h represents the hidden state of LSTM. $\in$-greedy is adjusted for the exploration policy and the exploration rate of episode ξ is:

$$\in (\xi) = max(\in_{start} - \xi.\upsilon, 0) \tag{3.39}$$

υ is the decreasing count of $\in$ each episode. The initial exploration rate is represented as $\in_{start}$.

Each $\widehat{Q}_i^{\psi}(s, a_i, d_i)$ cannot be directly updated by the global reward due to relation of the MSQV of each agent to the local reward,. In this way, loss function that the local reward closed to global reward is represented by (3.40).

$$\mathcal{L}(\psi^c) = \mathbb{E}_{\mathbf{s},\tau,a,d,R,\tau'}[(Q_{cent}^{\psi^c} - (r + \gamma Q_{cent}))^2] \tag{3.40}$$

Table 3.2. *Algorithm MSQV deep deterministic policy gradient*

Algorithm 1 MSQL

Initialize: Critic network ψ, target critic $\widehat{\psi}$ and agents' Q value networks $\psi^{\mu} = (\phi^1, ..., \phi^l)$
1. **for** each training episode $\in$ **do**
2. Initial state s_0 at t = 0, $h_i^0 = 0$ for each agent i
3. **while** s_t= terminal and $t < T$ **do**
4. $t = t + 1$
5. **for** each agent i **do**
6. $Q_i(o_{t,i}, .), h_i^t = DQN(o_{t,i}, h_i^{t-1} : \psi^i)$
7. Sample $a_{t,i}$ from $\mu_i Q_i(o_{t,i}, .)$
8. **end for**
9. Perform the joint actions a_i^t, d_i^t
10. Consider the reward r_t and next state s_{t+1}
11. **end while**
12. Store $\in$ to B
13. Sample a batch of episodes with batch size B from B
14. **for** $\in$ in batch **do**
15. **for** $t = 1$ to T **do**
16. Calculate targets y_t using $\widehat{\psi}^c$
17. **end for**
18. **end for**
19. Update each critic parameter ψ^c with loss $\mathcal{L}(\psi^c)$
20. Each episode reset $\widehat{\psi}^c = \psi^c$
21. **for** $\in$ in batch **do**
22. **for** for t = 1 to T do
23. Unroll PSO-LSTM using s_i, a_i, d_i and r_i
24. Decompose Q_{cent} at time t into $\widehat{Q}_i^t$ for each agent i
25. **end for**
26. **end for**
27. Update ψ^{μ} with loss $\mathcal{L}(\psi^i)$ for each agent i
28. **end for**

An MDP policy, $\mu : S \rightarrow A$ explains actions as functions of the current states. The optimal policy μ^{ψ} that maximizes the value function $V^{\psi}(s)$ is the main aim of an MDP. The relation between μ^{ψ} and the actions/states of the MDP is provided with the Bellman equations (3.34) where $\mu^{\psi}(s) = argmax_a Q^{\psi}(s, a_i^t, d_i^t)$. Inaccurate explorations are usually supplied in real situations because the states of each component in an MDP are known at each time step. For instance, measurement uncertainty and noise generally exist in the PMU units. The generalization of MDPs [86] can be expressed as the POMG parameters for the states are sensed by agents through inaccurate observations. The set of exploration results is expressed as E and the exploration function is denoted by $F = \mathbb{F}[e|s]$ for $e \in E$ and $s \in S$.

The aforementioned security problem is formulated as a strategic POMG involving PMU agents. The MG comprises a reward function $\Re_p$ for each agent l, a set of actions $A = A_1, A_2, ..., A_L$, a state transition function $T : S \times A_1 \times A_2 \times ... \times A_L \rightarrow$

$\mu(S)$, and a set of states S. In this context, the configurations of PMUs are represented by the states in S.

For PMU p, sets of explorations are denoted as $E_1, ..., E_P$. Each PMU's decision-making process relies on the corresponding exploration set E_p, which is a part of the overall state space S.

Table 3.3. *MG parameter settings*

Parameters	Value
Number of episodes	10000
Number of nodes	[64,64]
Size of hidden layers	256
Number of hidden layers	2
Learning rate	0.0003
Optimizer	Adam
γ	0.99
τ	0.0003
Replay buffer size	$2x10^5$
Non linearity	ReLU

To compromise the integrity of PMU measurements (P_{data}), the attacker explores the infrastructure and targets components along the path to P_{data}. Coordinated cyber-physical attacks on 123 Bus system for a second are given in Table 3.1.. Physical attacks are inserted before the initiation of the game. PMUs measure the falsified and normal data synchronously. During the game defender agents have to learn both game attack and former attacks. In response, the system controller must actively monitor the PMUs and enhance their protection to prevent data manipulation. Initially, the attacker does not have access to the network topology and vulnerabilities of the grid, while the system controller possesses complete knowledge of the SG system characteristics. However, the system controller cannot directly observe the instantaneous states of the entire system.

The adversarial events unfold as a round-based game, where the system controller and the attacker take turns performing actions on the PMUs. This game continues until either the system controller wins by successfully detecting the attacker or the attacker wins by compromising all PMU units in the network.

The components of the infrastructure are represented as nodes P of the graph. In this respect, the connection of the PMU units indicates the vulnerabilities against any coordinated attacks. In addition, each node $P_k \in P$ has an associated node state, $S_k =< S_k^A, S_k^D >$. The entity properties are set in the range $0, ..., m$, where

$m > 0$. Attack status of each node S_k^A includes p attributes, $S_{k,1}^A, ...S_p^A$. Similarly, the defense status of each node S^{Dk} includes $p+1$ attributes, $S_{k,1}^D..., S_{k,p+1}^D$.

The robustness of the protection of the nodes against the p types of attacks and encoding security settings of the nodes such as encryption and firewall functions are represented by the entity properties of the defenses $S_{k,1}^D, ..., S_{k,p}^D$. In addition, each node P_k has a particular defense attribute $S_{k,p+1}^D$ that represents the attack detection capability of a node. Likewise, p types of attacks e.g. Malware infection, and DoS attacks are represented by the attack attribute cluster $S_{k,1}^A, ..., S_{k,p}^A$.

It can be inferred that constructing an FDIA detection mechanism at node P_k corresponds to stimulating an increment in the detection value $S_{k,p+1}^D$ in the model. Simultaneously, this mechanism also corresponds to stimulating an increment in the defense attribute $S_{k,j}^D$, where $j \in 1, \ldots, p$ represents the attack type for FDIA in the model.

The actions are modeled within a specific framework, where the defender and attacker perform actions that lead to a new state in the constructed environment in each round of the game. The detection model exhibits the Markov property, $\mathbb{F}[s_{t+1}|s_t] = \mathbb{F}[s_{t+1}|s_1, \ldots, s_t]$, as determined by the aforementioned game characteristics.

3.3. DE-based MG and Solutions

SG, defined as an evolutionary algorithm whose main purpose is true parameter optimization, is first proposed in [87]. In DE, the representative of each decision variable in the chromosome is a real number. The initial population is randomly generated and then evaluated, just like any other sorting algorithm. The next process is the selection process. A single offspring is produced by adding the weighted difference vector between two competing parents to a third parent to determine which of the three parents selected at the selection point will pass on to the next generation.

The IoT-enabled SG communication network is modeled as a (dynamic) Markov environment. Here, points are a set of nodes with important information flow, and each line represents the interaction (communication) between these points. We propose a SG-based MG to solve the security risk for any SG device.

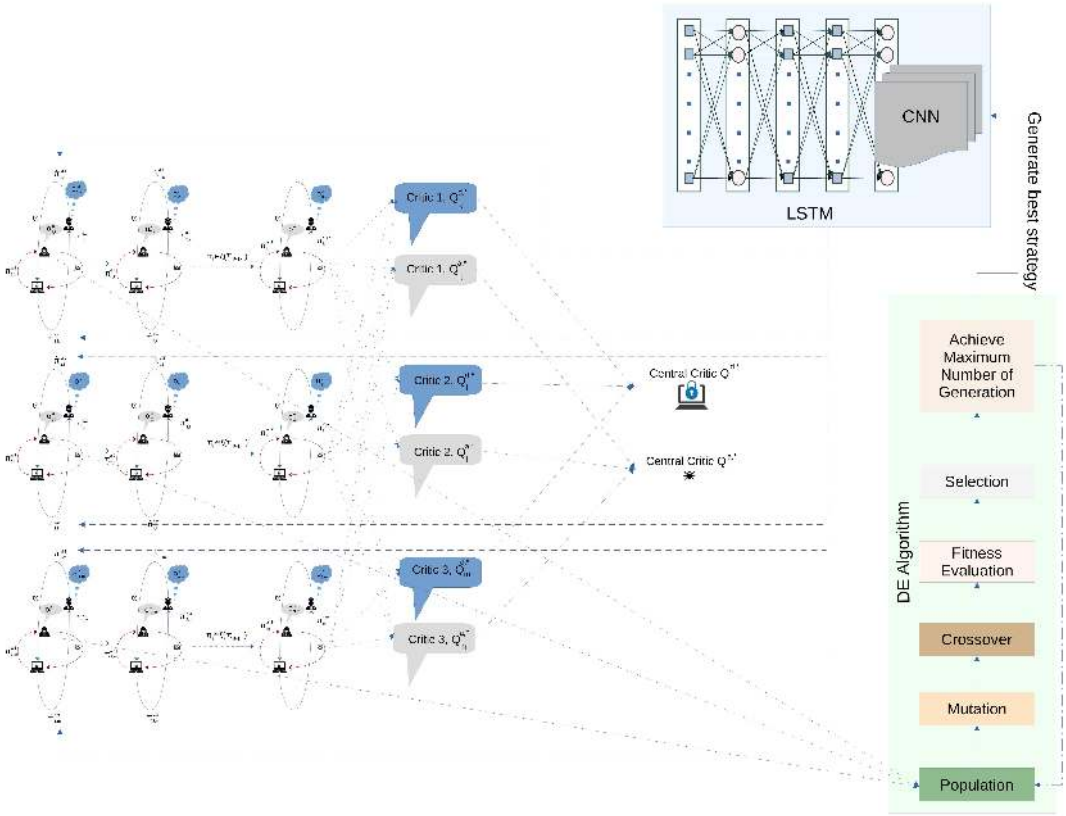


Figure 3.4. *Schematic presentation of Markov-DE game workflow on simulated system*

The proposed MG presents each player has a competition power defined as attack power $(a_1, a_2, ..., a_L)$ with choice probability $(p_1, p_2, ..., p_L)$ and the defense power $(d_1, d_2, ..., d_L)$ with the probability of choice $(q_1, q_2, ..., q_L)$. The objective of the attacker is to manipulate and change the information flow at the point by adjusting the appropriate attack competitiveness to each attack point and, respectively, to maximize seized points. The objective of the defender is to detect attacks and protect specified system layers by adjusting intended defensive competitiveness to each point and, accordingly, to minimize system loss. In this report, MCG is formulated as a 13-tuple $T_p = < S,\ A, T_{s^a s'}, L, \Re_{s^a s'}, \mu, p_0, E, F, P, w, \Xi, t >$.

The action space is denoted as $A = \{P^a(t), P^d(t)\}$. $P^a(t) = \{a_1, a_2, ..., a_L | a_1 < a_2 < ... < a_L\}$ is the competitiveness of the attacker cluster. $P^d(t) = \{d_1, d_2, ..., d_L | d_1 < d_2 < ... < d_L\}$ is the competitiveness of the defender cluster. Strategy space is denoted as $w = \{w_a, w_d\}$. Attacker and defender strategies depends the selecting

probability of the attacker and defender competition strength P^a and P^d.

$S = \{S^{ca}, S^{cd}, S^a, S^d, S^{ua}, S^{ud}\}$ is the state space of points in the defined environment. The states to be encountered at one point are dynamically transferred between the six states with the dynamic distribution of players' competitiveness. A function of the probability of the difference in competitiveness of the players of each cluster, as seen in (3.50) is defined as transition probability.

In this section, firstly, the Markov-DE game model based on LSTM-CNN agents acting with PSO optimization is established to optimize the attacker and defender competition interaction process of players. The proposed solutions to maximize the detection rate and minimize the system loss are analyzed, and the optimal policy selection is obtained by applying discretization process.

To overcome the challenges mentioned considering the research results and questions, in this report, we propose a Markov-DE dynamic model and an LSTM-CNN based approach to estimate the system parameters. To be specific, we first need to reformulate the strategy optimization problem as a MG [35]; it is a framework for multi-agent RL and also a multi-agent extension of MDP. Since there are N variable agents in PMUs in each time interval, in this report we consider N agents in the formulated MG corresponding to N PMUs.

$$R^a(s, a, d) + R^d_{cent} = 0 \tag{3.41}$$

The Makrov-DE algorithm is adopted to maximize collaborative defender strategies for multi-agent. The action-value function Q is used to define the expected payoff after performing an action. CNN-LSTM is used to approximate critical functions and policy functions. PSO algorithm is used to optimize CNN-LSTM agents. As an actor-critic based algorithm, Markov-DE is also secondary optimized with PSO. To this end, some of the attacker's tricks are applied to Markov-DE to create a multi-agent deep deterministic policy gradient.

$$V^a(s, \mu^{a,n}_{h,m}, \mu^{a,n}_{h,m}) = \sum_{t=0}^{\infty} \gamma^t \mathbb{E}(R^a_t | \mu^{a,n}_{h,m}, \mu^{a,n}_{h,m}, s_0 = s) \tag{3.42}$$

and the state value function $V^d(s, \mu^{a,n}_{h,m}, \mu^{a,n}_{h,m})$ is defined as:

$$V^d(s, \mu^{a,n}_{h,m}, \mu^{a,n}_{h,m}) = \sum_{t=0}^{\infty} \gamma^t \mathbb{E}(R^d_t | \mu^{a,n}_{h,m}, \mu^{a,n}_{h,m}, s_0 = s) \tag{3.43}$$

The goal of each agent k is to determine a policy μ_k that provides the optimal payoff $J^d(\mu^{a,n}_{h,m}, \mu^{a,n}_{h,m})$ (shown in (3.44)) expected from the random initial state where R_{nt} is the sum of agent n's discounted future reward.

$$J^d(\mu^{a,n}_{h,m}, \mu^{a,n}_{h,m}) = \mathbb{E}_{\mu^{a,n}_{h,m}, \mu^{a,n}_{h,m}}[\sum_{t=1}^{T} \gamma^{t-1} R_{nt}(s_t, a_t)] \tag{3.44}$$

In the presence of multiple Nash equilibrium, players with common interests may not learn an optimal coordination policy. For this purpose, a two-stage optimization algorithm within the Nash equilibrium is proposed. The algorithm proves convergence to Nash equilibrium in a joint action player teamwork game. In other words, both players use their best response strategies against the opponent by optimizing their strategies with the help of two-stage optmization including differential equation and PSO to reach optimal policies. It is a game in which the policies of the opposing players are interdependent and the players do not change their strategy in the direction where a is a Nash equilibrium.

A Nash equilibrium consists of an optimum strategy for both players (μ^{*a}, μ^{*d}) for all mixed strategies μ^a and μ^d in all states $s \ \epsilon \ S$ in the proposed zero-sum two-player Markov-DE game.

$$Q^{a*}_{\mu}(s_t) = \mathbb{E}_{\mu, s_t, a^a_t}[r_{t+1} + \gamma V^{a*}_{\mu}(s_{t+1})] \tag{3.45}$$

$$Q^{d*}_{\mu}(s_t, a^d_t) = \mathbb{E}_{\mu, s_t, a^d_t}[r_{t+1} + \gamma V^{d*}_{\mu}(s_{t+1})] \tag{3.46}$$

where Bellman equations for the performances of the value functions as follows:

$$V^{a*}_{\mu}(s_t) = min_{a^a_t \epsilon A^a} \mathbb{E}_{\mu, s_t, a^a_t}[r_{t+1} + \gamma V^{a*}_{\mu}(s_{t+1})] \tag{3.47}$$

$$V^{d*}_{\mu}(s_t) = max_{a^d_t \epsilon A^d} \mathbb{E}_{\mu, s_t, a^a_t}[r_{t+1} + \gamma V^{d*}_{\mu}(s_{t+1})] \tag{3.48}$$

As mentioned earlier, the game is zero-sum, stationary, and $\gamma < 1$, it follows from game theory that $V^*(\zeta) = V^{a*}_{\mu^*}(\zeta) = V^{d*}_{\mu^*}(\zeta)$.

$$V^*(\zeta) = max_{\mu^d_l \epsilon \delta A^d} min_{\mu^a_l \epsilon \delta A^a} \mathbb{E}_{\mu^d_l, \mu^a_l, \zeta}[r^d_{t+1} + \gamma V^*(\zeta_{t+1})] \tag{3.49}$$

3.3.1. State transition

The state of the defined environment varies between three states depending on the relative competitiveness shown by the offensive player and the defensive player. Figure 3.5 represents the state transitions of the predefined environment. Therefore, the threshold values are the values optimized with the help of PSO.

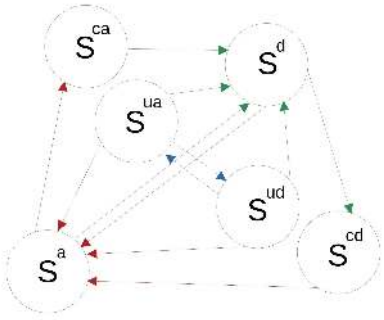


Figure 3.5. *State transitions of the environment*

- $ua \rightarrow a$: If the attacker's competitiveness is higher than that of the defender and the power difference $P^a(t) - P^{ua}(t)$ is greater than the determined threshold ϱ^1, the point below this state is considered vulnerable to attack and the attacker will gain control. The state of point i defined in the environment is transformed from unattacked (ua) to attacked (a). $\tau^m(t)$ is defined as the probability that perimeter point i is compromised by an attacker at time t.

- $ud \rightarrow d$: Similarly, if the defender's competitiveness is higher than the power difference $P^d(t) - P^{ud}(t)$ is greater than ϱ^2, the defender gains control of the system. The state of the environment will be converted from undefended (ud) to defended (d). $\tau^m(t)$ is defined as the probability of transitioning to a state where the perimeter is protected by the defender at time t, $\tau^m(t)$ is calculated as:

$$\tau^m(t) = \mathbb{P}\{P^m(t) - P^{m-1}(t) \geq \varrho^n\} \tag{3.50}$$

$\mathcal{T} = \{\tau^{0\to1}, \tau^{1\to2}, ..., \tau^{n\to n+1}\}$ is a cluster of the transition probability $\tau \ \epsilon \ \mathcal{T}$. The state evolution variable denoted as $\Xi(t) = \{S^{d*}(t), S^{a*}(t), S^{ca*}(t), S^{cd*}(t), S^{ud*}(t), S^{ua*}(t)\}$ and the game evolution process is clarified by differential equations expressed as follows;

- $d \to cd$: For the defender, if the competitive power injected by the attacker is higher than the power difference $P^{cd}(t) - P^d(t)$ is greater than a defined threshold ϱ^3, the point i in the environment under the defender's control is transformed to the robust defender's control by updating. In this case, the state of the point will change to continue defend (cd). The probability of switching from defense to robust defense at time t is denoted as $\tau^{d\to cd}$.

- $a \to ca$: For the attacker, if the competitive power injected by the attacker is higher than the power difference $P^{ca}(t) - P^a(t)$ is greater than a determined threshold ϱ^4, the point i in the environment under the attacker's control is transformed to the robust attacker's control. In this case, the state of the point will change to continue attack (ca). The probability of switching from attacked point to seized attacked point at time t is denoted as $\tau^{a\to ca}$.

The $a \to d$ and $d \to a$ transition processes are also known as seizing the opponent's control point. In addition, the opponent's control point is more difficult to seize than points that have not been previously attacked or under control. For instance, the probability of intruding on points protected by the defender are low campared to points in state U.

- $a \to d$: In this state change, point i has been attacked before and will be taken over by the defender. If the defender wants to negate the effects of the attack, the defender must estimate the damage incurred at this point with a fairly high probability. Here, threshold values are optimized with PSO. In this report, if the competitive power injected by the defender to the attack point is higher than the power difference $P^d(t) - P^a(t)$ is greater than ϱ^5, the attack point is captured by the defender. Defining $\tau^{a\to d}(t)$ as the probability of an

attacked point i passing to the safe zone under the control of the defender at time t.

- $d \rightarrow a$: If the attacker wants to manipulate data at point i, the attacker must attack the point with a very high power. In this report, it is stated that if the competitive power injected by the attacker to the attack point is higher than the power difference $P^{ua}(t) - P^{a}(t)$ is greater than ϱ^{6}, the attack point is trespassed and the state of the point is converted to a. We define $\tau^{d \rightarrow a}(t)$ as the probability that point i will become attacked at time t.

- $ca \rightarrow d$: For the defender, if the competitive power injected by the attacker is higher than the power difference $P^{d}(t) - P^{ca}(t)$ is greater than a defined threshold ϱ^{7}, the point i in the environment under the defender's control is transformed to the robust defender's control by updating. In this case, the state of the point will change to defend (d). The probability of switching from defense to robust defense at time t is denoted as $\tau^{ca \rightarrow d}$.

- $ua \rightarrow d$: For the defender, if the competitive power injected by the attacker is higher than the power difference $P^{ua}(t) - P^{d}(t)$ is greater than a defined threshold ϱ^{8}, the point i in the environment under the defender's control is transformed to the robust defender's control by updating. In this case, the state of the point will change to defend (d). The probability of switching from defense to robust defense at time t is denoted as $\tau^{ua \rightarrow d}$.

- $ud \rightarrow a$: If the attacker wants to manipulate data at point i, the attacker must attack the point with a very high power. In this report, it is stated that if the competitive power injected by the attacker to the attack point is higher than the power difference $P^{a}(t) - P^{ud}(t)$ is greater than ϱ^{9}, the attack point is trespassed and the state of the point is converted to ud. We define $\tau^{ud \rightarrow a}(t)$ as the probability that point i will become attacked at time t.

- $cd \rightarrow a$: If the attacker wants to manipulate data at point i, the attacker must attack the point with a very high power. In this report, it is stated that if the competitive power injected by the attacker to the attack point is higher than the power difference $P^{cd}(t) - P^{a}(t)$ is greater than ϱ^{10}, the attack point is trespassed and the state of the point is converted to ud. We define $\tau^{cd \rightarrow a}(t)$

as the probability that point i will become attacked at time t.

3.3.2. First level: MG policy optimization

We denote $S = \{S^{ua}, S^{ud}, S^{a}, S^{d}, S^{ca}, S^{cd}\}$ as functions of the states occurring in the defined environment. The change in other stated cases is also explained by the connections of similarly expressed parameters. More precisely, security in the established Markov environment is related to the following equation in line with this framework.

$$\Xi(t) = \begin{cases} \tau^{ud\to d}(t)S^{ud} + \tau^{ua\to d}S^{ua}(t) + \tau^{a\to d}(t)S^{a} & \text{if } s = S^{d}(t) \\ -\tau^{d\to cd}(t)S^{d} - \tau^{d\to a}(t)S^{d} + \tau^{ca\to d}(t)S^{ca} & \\ \tau^{ud\to a}(t)S^{ud} + \tau^{ua\to a}S^{ua}(t) + \tau^{d\to a}(t)S^{d} & \text{if } s = S^{a}(t) \\ -\tau^{a\to d}(t)S^{a} + \tau^{cd\to a}(t)S^{cd} - \tau^{a\to ca}(t)S^{a} & \\ \tau^{a\to ca}S^{a}(t) - \tau^{ca\to d}S^{ca}(t) & \text{if } s = S^{ca}(t) \\ \tau^{d\to cd}S^{d}(t) - \tau^{cd\to a}S^{cd}(t) & \text{if } s = S^{cd}(t) \\ \tau^{ud\to ua}(t)S^{ud} - \tau^{ua\to d}S^{ua}(t) - \tau^{ua\to a}(t)S^{ua} - \tau^{ua\to ud}(t)S^{ua} & \text{if } s = S^{ua}(t) \\ \tau^{ua\to ud}(t)S^{ua} - \tau^{ud\to d}S^{ud}(t) - \tau^{ud\to ua}(t)S^{ud} - \tau^{ud\to a}(t)S^{ud} & \text{if } s = S^{ud}(t) \end{cases} \tag{3.51}$$

The attacker's goal is to increase the number of attack points at the minimum cost value, while the defender's goal is to maximize the number of defense points at the least cost. It is stated that the attack cost is a function of the attack power in the defined environment and the number of points the attacker agents aim to seize all points. As a result, the number of target points between the attacker-side and the defender-side is the sum of $S^{ua} + S^{ud} + S^{a} + S^{d} + S^{ca} + S^{cd}$. For a point i defined in the environment, $C^{a}(t)$ and $C^{d}(t)$ are defined as the average competitive power for the attacker and the defender, respectively. $C^{m}(t)$ is calculated as follows:

$$C^{m}(t) = \sum_{i=1}^{L} m_i n_i(t) \tag{3.52}$$

For this purpose, the cost values for attack/defend are calculated by using

the cost coefficients f^a, f^d of the offensive and defensive players, respectively, with (3.53).

$$\mathbb{C}^m(t) = f^m C^{2m}(t)[S^a(t) + S^{ca}(t) + S^d(t) + S^{cd}(t) + S^{ua}(t) + S^{ud}(t)] \tag{3.53}$$

Payoff function $p(t)$ is expressed as follows:

$$p(t) = S^{ua}(t) + S^a(t) + S^{ca}(t) - S^{ud}(t) - S^d(t) - S^{cd}(t) - \frac{1}{2}[\mathbb{C}^a(t) - \mathbb{C}^d(t)] \tag{3.54}$$

The total payoff function P is obtained by integrating $[0, T]$ over time. The main aim of the defensive player cluster is to minimize $p(t)$ and it achieves this by optimizing μ_i^a. The aggressive player cluster aims to maximize $p(t)$ and achieves this by optimizing μ_i^d. The sum of the attacker and defender's strategies is expressed as $\sum_{i=1}^{L} \mu_{i=1}^m$. (3.55) forms an attack-defense zero-sum MG. Considering the balance of the proposed differential game:

$$\begin{cases} max_{\mu_i^a(t)} P(\mu_i^a(t), \mu_i^{d*}(t)) & \text{if attacker} \\ 0 \leq \mu_i^a(t) \leq 1 & \\ min_{\mu_i^d(t)} P(\mu_i^{a*}(t), \mu_i^d(t)) & \text{if defender} \\ 0 \leq \mu_i^d(t) \leq 1 & \end{cases} \tag{3.55}$$

3.3.3. Equilibrium and solutions

The case of minimizing P by optimizing the attacker's policy and maximizing P by optimizing the defender's policy gives rise to two conditions for the optimized players' policy at condition $\{P(b_i^{a*}, b_i^{d*}) \geq P(b_i^a, b_i^{d*})\}$ and $\{P(b_i^{a*}, b_i^{d*}) \leq P(b_i^{a*}, b_i^d)\}$ is called saddle point policy. The instantaneous payoff $p(t)$ and the dynamic evolutionary equations in (3.54) are defined as continuous and bounded. Based on (3.52), equilibrium occurs if and only if $p(t)$ is convex for the offensive policy $\mu_i^a(t)$ and concave for the defensive policy $\mu_i^d(t)$.

$$\begin{cases} \frac{\partial^2 p(t)}{\partial \mu_i^{2d}} > 0 & \text{if defender} \\ \frac{\partial^2 p(t)}{\partial \mu_i^{2a}} < 0 & \text{if attacker} \end{cases} \tag{3.56}$$

According to the differential algorithm, $p(t)$ is expressed as concave with respect to $\mu_i^a(t)$ and convex with respect to $\mu_i^d(t)$. Furthermore, it is known that a saddle point policy $\{\mu_i^{d*}(t), \mu_i^{a*}(t)\}$ exists in this optimization phase of the MG.

To obtain the optimal saddle point policy, the following steps are performed respectively: Based on the Pontryagin minimization principle in optimal control theory, it is solved by introducing common state variables, achieved by the Hamiltonian function. Then, specified state functions and common state functions are solved together to obtain optimal control policies. The Hamilton function is obtained from the integration of multiple differential equations with $p(t)$ expressed in (3.54). The common state variable $\vartheta_s(t)\ \epsilon\ \vartheta_{S^{ud}}(t), \vartheta_{S^{ua}}(t), \vartheta_{S^a}(t), \vartheta_{S^d}(t), \vartheta_{S^{ca}}(t), \vartheta_{S^{cd}}(t)$ is defined as a 6-dimensional vector. The optimization of the objective function is described as follows:

$$H(s, \omega, \vartheta_s, t) = u(t) + \sum_{s \epsilon \{S^{ua}, S^{ud}, S^a, S^d, S^{ca}, S^{cd}\}} \vartheta_s(t) \Xi(t) \tag{3.57}$$

The state variables must be solved to produce an optimal solution to the problem in (3.55). The common state variables $\vartheta_{S^a}(t), \vartheta_{S^d}(t), \vartheta_{S^{ua}}(t), \vartheta_{S^{ca}}(t), \vartheta_{S^{ud}}(t), \vartheta_{S^{cd}}(t)$ are then determined and assigned.

Specifically, in the MG optimized with differential equations, the optimal strategies of the players are calculated as follows:

$$\mu_i^{m*}(t) = \begin{cases} 0 & \text{if } \mu_i^m(t) \leq 0 \\ \mu_i^m(t), & \text{if } 0 < \mu_i^m(t) < 1 \\ 1, & \text{if } \mu_i^m(t) \geq 1 \end{cases} \tag{3.58}$$

where $\mu_i^a(t)$ and $\mu_i^d(t)$ are expressed as:

$$\mu_i^{m*}(t) = \frac{\sum_{S^{ua},S^{ud},S^a,S^d,S^{ca},S^{cd}} \vartheta_s SW_s}{[S^a(t) + S^d(t) + S^{ua}(t) + S^{ud}(t) + S^{ca}(t) + S^{cd}(t)] f^m m_i^2} - \frac{\sum_{y \neq i} m_y \mu_y(t)}{m_i} \tag{3.59}$$

$$SW_s = \begin{cases} SW_{S^{ua}} = S^{ud} \sum_{a_i - d_i} \mu_i^m - S^a \sum_{a_i - d_i} \mu_i^m - S^{ud} \sum_{a_i - d_i} \mu_i^m - S^d \sum_{a_i - d_i} \mu_i^m \\ SW_{S^{ud}} = S^{ua} \sum_{a_i - d_i} \mu_i^m - S^d \sum_{a_i - d_i} \mu_i^m - S^a \sum_{a_i - d_i} \mu_i^m - S^{ud} \sum_{a_i - d_i} \mu_i^m \\ SW_{S^a} = S^{ua} \sum_{a_i - d_i} \mu_i^m + S^d \sum_{a_i - d_i} \mu_i^m + S^{ud} \sum_{a_i - d_i} \mu_i^m + S^{cd} \sum_{a_i - d_i} \mu_i^m \\ \qquad -S^{ca} \sum_{a_i - d_i} \mu_i^m - S^d \sum_{a_i - d_i} \mu_i^m \\ SW_{S^d} = S^{ua} \sum_{a_i - d_i} \mu_i^m + S^{ud} \sum_{a_i - d_i} \mu_i^m + S^a \sum_{a_i - d_i} \mu_i^m + S^{ca} \sum_{a_i - d_i} \mu_i^m \\ \qquad -S^{cd} \sum_{a_i - d_i} \mu_i^m - S^a \sum_{a_i - d_i} \mu_i^m \\ SW_{S^{ca}} = S^a \sum_{a_i - d_i} \mu_i^m - S^d \sum_{a_i - d_i} \mu_i^m \\ SW_{S^{cd}} = S^d \sum_{a_i - d_i} \mu_i^m - S^a \sum_{a_i - d_i} \mu_i^m \end{cases} \tag{3.60}$$

The players' optimized policies are closely interconnected. Expressions in close form for common state variables $\vartheta_{S^{ua}}(t), \vartheta_{S^{ud}}(t), \vartheta_{S^a}(t), \vartheta_{S^d}(t), \vartheta_{S^{ca}}(t), \vartheta_{S^{cd}}(t)$ and state variables $S^{ua}(t), S^{ud}(t), S^a(t), S^d(t), S^{ca}(t), S^{cd}(t)$, optimized attack-defense policies are obtained by discrete iterations.

3.3.4. Optimal policy selection

The optimized attack-defense policies $\mu_i^{m*}(t)$ are not calculated in continuous form. Accordingly, the Gauss-Seidel-like implicit finite difference method [88] is used to transform the continuous time differential equation to discrete equation. In time T divided into K an indefinitely small time periods, each time period length is $\pi = T/K$. k^{th} time interval is represented by k. By discretizing the combined state variables and state variables, the new formulation of the attack-defense policy of the MG are expressed as follows:

$$\mu_i^m(k+1) = \frac{\sum_{S^{ua},S^{ud},S^a,S^d,S^{ca},S^{cd}} \vartheta_{n-k-1} SW_x}{[\sum_{s \epsilon S} S^s(k+1)] f^m m_i^2} - \frac{\sum_{y \neq i, m_y < 0} m u_y(k)}{m_i} \tag{3.61}$$

$\vartheta_x(n-k-1)$ are the discrete forms of the combined state variable ϑ_x. Since policies in this discrete form are probabilities at [0, 1], the optimized policies are denoted as $\mu_i^{m*}(k+1)$ respectively. It is formulated in discrete piecewise functions defined in (3.62).

$$\mu_i^{m*}(k+1) = \begin{cases} 0 & \text{if } \mu_i^m(k+1) \leq 0 \\ \mu_i^m(k+1), & \text{if } 0 < \mu_i^m(k+1) < 1 \\ 1, & \text{if } \mu_i^m(k+1) \geq 1 \end{cases} \tag{3.62}$$

The convergence of the set of policy pairs to a NE is estimated based on the approximate availability metric v^{exp} [89]:

$$v^{exp} = J^d(\mu^{d*,n}, \mu^{a*,n}) + J^a(\mu^{d*,n}, \mu^{a*,n}) \tag{3.63}$$

3.3.5. Second level: CNN-LSTM optimization

From an evolutionary optimization perspective, determination of the optimal CNN-LSTM architecture for a MG based on a PMU dataset collected from the SG is formulated as a constrained optimization problem, where the aim is to maximize the accuracy and minimize the number of CNN-LSTM model parameters determined. The formulation starts with the initialization as follows:

$$max_{k\epsilon K, l\epsilon L} O(k,l) = [Ac(k,l), -HP(k,l)] \tag{3.64}$$

$$HP(k,l) \leq H_{max}$$

CNN-LSTM network to be optimized is denoted as $\{k, l\}$. The search spaces of CNN-LSTM network $\{K, L\}$, accuracy obtained by a CNN-LSTM network denoted as $A(k,l)$, the number of parameters in $\{k,l\}$ denoted as HP and the maximum limit based on the requirements of a real cyber attack detection objective H_{max} are the main parameters for initialization stage.

The DE algorithm process is included in the algorithm each candidate in the population are given as $X = [(k_1, l_1), (k_2, l_2), ..., k_N; l_N]$. The evolution process is shown in (3.65). The evolved solution E takes a uniform random number value at $(0, 1)$, where $\{K_r, L_r\}$ is a randomly selected candidates and $\{K_g, L_g\}$ is the global best candidates for the last population. Evolution probability controls the evolution function $E()$: for the case $r < \sigma$, it is performed on the global best solution, otherwise the evolution operation is performed on a individual that is selected randomly.

$$E_i(x) = \begin{cases} E(K_r), E(L_r) & if \;\; r < \sigma \\ E(K_g), E(L_g) & otherwise \end{cases} \tag{3.65}$$

The evolution process is followed by the trial solution U generated by (3.66), where θ denotes the crossover probability and CO denotes the crossover process. The recombinant and, more importantly, evolved solution of the ith individual in the population is provided with the condition that $r < \theta$.

$$U_i = \begin{cases} CO(K_i, L_i, E_i) & if \;\; r < \theta \\ E_i & otherwise \end{cases} \tag{3.66}$$

Finally, the fitness values of the trial solutions are calculated and compared with $\{K_i, L_i\}$. The solution with higher fitness values survives the next round of iteration. First, population G_0 is randomly initialized with predefined population size N. Following this, the fitness values of each individual in G_0 are calculated and the DE-based evolution process is initiated, which includes evolution, crossover and selection processes. Once the maximum number of generations is reached, the best individual to construct the CNN-LSTM architecture is decoded. Finally, the CNN-LSTM architecture is evaluated for cyber attack detection in the MG. Detailed explanations of initial population, fitness evaluation and evolutional operation based DE are given in the sub-processes of the algorithm.

3.3.5.1. Population initialization

In the proposed approach, the DE algorithm, one of the most effective evolutionary optimization techniques, is integrated optimally. To enhance the global

optimization capabilities of DE, its procedure is incorporated into the algorithm as an additional evolutionary operator. The objective of this phase is to determine the optimal combination of the CNN and LSTM algorithms.

Details of population initialization are clarified as L individuals are initialized as initial population G_0. $I_i = [R, [T_1, HP_1], [T_2, HP_2] \ , ..., [T_R, HP_R]]$ represents a formulation to identify individuals. I_i denotes the i^{th} individual in G_0, R denotes the number of units in the individual. In i^{th} array, the type of the i^{th} unit T_1, the parameters of the i^{th} unit HP_1. The definition of HP_1 is expressed as follows :

$$HP_i = \begin{cases} [amount_1, k_1, in_1, out_1] & if \ \ type(u_i) = Conv1D \\ [amount_1, k_1, in_1, out_1] & if \ \ type(u_i) = Act.func. \\ [amount_2, k_2, in_2, out_2] & if \ \ type(u_i) = LSTM \\ [amount_2, k_2, in_2, out_2] & if \ \ type(u_i) = Dense \\ [amount_2, k_2, in_2, out_2] & if \ \ type(u_i) = Dropout \end{cases} \tag{3.67}$$

A random number $R = randint(1, D)$ is denoted as the number of network unit of each individual based on the specified networks have bounded depth (D).

3.3.5.2. Fitness evaluation

The steps of fitness evaluation in detail for each individual in the generated population are specified in this section. The CNN-LSTM network is created by decoding the architectural information received from each individual and training it on the training set. As performance indices for the evaluation of the created CNN-LSTM networks, fitness variances for the validation sets throughout the training processes, final fitness in the validation set and CNN-LSTM model parameter numbers are collected. This process starts after the training process is completed. The indices in the population are considered comprehensively and the best agent Indices in the population are determined. Algorithm 5 shows the comparison process of two agents and its explanation is as follows:

- Two agents, d-agent 1 and d-agent 2, are entered, it is checked whether their fitness evolution in the validation set reaches limit, that is, the lower limit of

accuracy in the validation set, and the agent that cannot reach the lower limit is eliminated.

- When I_i and I_{i+1} both reach or are below the limit, the difference between their fitness is calculated, and if this difference is greater than the fitness threshold (β), the agent with lower fitness is eliminated.
- When the difference between the fitness values of (I_i, I_{i+1}) are lower than β, the difference between the number of parameters is calculated. When the parameter is larger than its threshold, the individual that has smaller parameters is selected.
- The difference between the fitness variance is calculated, if the difference of the parameters is lower than β. If the variance is greater than β, the individual that has smaller variance is selected.

Table 3.4. *Experimental setups of attacker-defender agents*

Hyperparameters	Value
Batch Size	32
Discount factor γ	0.99
Replay buffer size	7,000
Learning rate	0.0001
Sample size	100
Episode length	300
Test episode	36
Test interval	2,000
Target update interval	400
Optimizer	Adam
LSTM Hidden units	64
Epsilon start	0.8
Epsilon finish	0.03
Exploration step	25,000
Max training step	30,000

3.3.5.3. DE-based evolutionary process

The system initiates the Markov-DE game based on evolutionary processes that updates a population. The update based on operations such as evolution, crossover, and selection. Individuals in population G produces evolved individuals through the process. If the evolution probability is lower than mutation probability σ, the best individual I_{best} in the population is a candidate for evolution. In a different state, a random I_r is selected from the population G as a candidate. Position Z is then randomly selected in the candidate individuals to perform the evolution operations. The following equation is used to generate the evolved individual E:

$$E = \begin{cases} Diff(I_{best}, H) & if \;\; q < \sigma \\ Diff(I_r, H) & otherwise \end{cases} \tag{3.68}$$

Evolution of an individual at point Z is shown as Diff(·). There are three types of evolutions an individual can undergo: removing a unit, replacing an existing unit, and adding a new unit. The proposed evolution steps are defined. It represents the individual created for evolution, the randomly created block for modification, and the evolution position Z.

The best sub-policy for each player is computed with the help of dynamic programming. When v^{exp} is 0, the objective of NE is achieved. The central policy is achieved with CNN-LSTM algortihm.

$$Q_{cent}^{i*,n+1} = \sum_{i=1}^{M} Q_i(\tau_i, a_i, d_i) \tag{3.69}$$

Each local agent is formulated in a Recurrent Deep Q-Network that is formed from an CNN-LSTM layer [27] and a Multi-Layer Perceptron to solve partial observability. The local agent is represented as follows:

$$Q_i = g(\tau_i^a, \tau_i^d; \mathfrak{cl}) \tag{3.70}$$

where $\mathfrak{cl}$ denotes the hidden state of CNN-LSTM network after DE optimization. ϵ-greedy is adjusted for the exploration policy and the exploration rate of episode ξ is:

$$\epsilon(\xi) = max(\epsilon_{start} - \xi.\nu, 0) \tag{3.71}$$

υ is the decreasing count of ϵ each episode. The initial exploration rate is represented as ϵ_{start}.

Each $\widehat{Q}_i^{\psi}(s, a_i, d_i)$ cannot be directly updated by the global reward due to relation of the MSQV of each agent to the local reward. It is very important to select the proper intermediate state in the current environment locations, which

can largely improve the accuracy of the players. Inspired by Reference [35], a loss function based on expected value operator is to evaluate the actions of the players in the last stage of the proposed game expressed as follows:

$$\mathcal{L}(\mu_j^{i*,n}) = \mathbb{E}_{\mu,s_t,a_t^d,a_t^a}[(Q_{cent}^{i*,n+1} - (r + \gamma Q_\mu^{i*}(s_t, a_t^d))^2] \tag{3.72}$$

4. SIMULATIONS AND RESULTS

In this chapter, the effectiveness of the proposed approaches on cyber-physical attack detection in communication network in SGs is presented as case studies. Data preprocessing and model evaluation criteria are given in Section 4.1 and Section 4.2, respectively. Results regarding cyber attack and pyhsical faults occurring in data flow over the dataset and game-based simulations are discussed in Section 4.3 and Section 4.4, respectively

4.1. Dataset and Preprocessing

Time-synchronized data includes system measurements and device status. Synchrophasor technology improves the security of the cyber-physical environment for the SG. Data is collected from an operational test environment that represents an extension of the test environment. For unsupervised multiple detection and classification scenarios, PMU whole datasets are used to detect attacks on normal operating datasets and successfully differentiate these attacks, then the method is evaluated on datasets with attacks. For this purpose, the Min-Max normalization is applied to turn into [0,1] range calculated as follows:

$$K_i' = \frac{K_i - K_{i,min}}{K_{i,max} - K_{i,min}} \tag{4.1}$$

where K_i' and K_i are the normalized value and original value of the i_{th} feature, respectively. $K_{i,max}$ and $K_{i,min}$ are the maximum and minimum values of the i_{th} feature, respectively, before normalization.

4.2. Model Evaluation Criteria

The proposed intrusion detection approaches have been validated using model evaluation metrics. Unlike most studies, the selected evaluation criteria is examined on a large scale, considering the maximum and minimum values for metrics such as accuracy, precision, recall, and F1 score. The studies focus on both physical units (such as busbars, switches, and transmission cables) and communication units (such as PMU and PDC).

4.2.1. Binary classification

The confusion matrix (shown in Table 4.1.) is used to validate the proposed approach for binary classification, as detailed in Table 4.2.. Binary classification is applied to the approach proposed in Chapter 2 for distinguishing between normal and attacked data.

Table 4.1. *The confusion matrix of binary classification of SG system FDIA detection*

Actual class	Predicted class	
	Normal	Attack
Normal	TN	FP
Attack	FN	TP

Table 4.2. *The metrics for evaluation of binary classification*

Measure	Formula class	Intuitive meaning
Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$	The percentage of predictions those are correct
Precision	$\frac{TP}{TP+FP}$	The percentage of positive predictions those are correct
Recall	$\frac{TP}{TP+FN}$	The rate of samples that correctly classified as an attack
F-measure	$\frac{2*PR}{P+R}$	The weighted harmonic mean of Precision and Recall

4.2.2. Multiclass classification

The evaluation metrics scores for the proposed PSO-CNN-LSTM and MG approaches in a multi-classification context are crucial for verifying the model's effectiveness in attack detection and identifying its shortcomings and errors. Correctly detected attacked and unattacked data (tn_i, tp_i) or undetected data (fn_i, fp_i) for the i^{th} attack class are presented in Table 4.3..

4.3. Cyber-Physical Attack Detection based PSO-CNN-LSTM

In studies beginning with binary classification, the performance of PSO-CNN-LSTM (given in Table 4.4. to 4.6.) and the proposed MG approaches (given in Table 4.8.) are compared based on the metrics specified. Optimized hyperparameters for convolutional layers and LSTM include the following: learning rate, number of network layers, number of nodes per layer, number of neurons, activation function, initializer, dropout rate, kernel size, optimizer, and batch size. Among these hyperparameters, the learning rate is particularly important because it controls the learning progress of the model. The speed at which the model learns varies de-

Table 4.3. *The metrics for evaluation of multi-class classification*

Measure	Formula class	Intuitive meaning
Ov. Acc.	$\frac{TP+TN}{TestSetSize}$	The rate of overall true predictions for the test set
Av. Acc.	$\frac{\frac{\sum_{i=1}^{l} tp_i+tn_i}{tp_i+tn_i+fp_i+fn_i}}{l}$	Average per-class effectiveness of a classifier
Ov. Error Rate	$\frac{FP+FN}{TestSetSize}$	The rate of overall false prediction for test set
Av. Error Rate	$\frac{\frac{\sum_{i=1}^{l} fp_i+fn_i}{tp_i+tn_i+fp_i+fn_i}}{l}$	The average per-class classification error
Prec.	$\frac{TP}{TP+FP}$	The percentage of positive predictions those are correct
Rec.	$\frac{TP}{TP+FN}$	The rate of samples that correctly classified as an attack
F1-measure	$\frac{2*PR}{P+R}$	The weighted harmonic mean of Precision and Recall
Mac.-Av. Prec.	$\frac{\frac{\sum_{i=1}^{l} tp_i}{tp_i+fp_i}}{l}$	The mean of the precision of each class
Mac.-Av. Rec.	$\frac{\frac{\sum_{i=1}^{l} tp_i}{tp_i+fn_i}}{l}$	The average recall of each class
Mac.-Av. F1-Score	$\frac{2.Precision_M.Recall_M}{Precision_M+Recall_M}$	The percentage of correct positive predictions
Mic.-Av. Prec.	$\frac{\sum_{i=1}^{l} tp_i}{\sum_{i=1}^{l} tp_i+fp_i}$	The precision computed from the division true positive prediction and total positive predictions
Mic.-Av. Rec.	$\frac{\sum_{i=1}^{l} tp_i}{\sum_{i=1}^{l} tp_i+fn_i}$	The determination of how many positive samples are truly predicted as the positive class
Mic.-Av. F1-Score	$\frac{2.Precision_\mu.Recall_\mu}{Precision_\mu+Recall_\mu}$	The computation from micro-recall and averaged precision of each class

*Ov.=Overall, Acc.=Accuracy, Av.=Average, Prec.=Precision, Rec.=Recall, Mac.=Macro, Mic.=Micro

pending on the learning rate. A higher learning rate can accelerate training, but if set too high, the model may overshoot the optimal solution and fail to converge. Determining the optimal learning rate is not intuitive and often requires careful tuning.

4.3.1. Test results based on binary-class dataset

The binary classification approach is assessed based on the criteria outlined in Table 4.4.. A comparative analysis between the proposed CNN-LSTM model and the PSO-optimized and selected DL models reveals significant differences in their performance.

4.3.2. Test results based on three-class dataset

Table 4.4. *The results of binary-class classification*

Metric	Classifiers			
	Accuracy	Precision	Recall	F-measure
PSO-LSTM-CNN	98.61	93.89	92.01	92.94
PSO-LSTM	96.06	96.20	75.00	84.23
CNN-LSTM	88.88	74.21	95.25	83.42
LSTM	84.02	86.55	93.56	90.78
multi-CNN [90]	86.95	89.56	87.25	88.41
ResNets [91]	97.50	-	-	-
DL [92]	97.77	98.25	99.31	98.78
Bloom filters [93]	97.00	98.00	92.00	95.00
MCNN [94]	91.46	-	-	-
PCA-SVM [95]	97.00	-	97.00	97.00
LSTM-Autoencoder [96]	94.56	-	94.54	93.68

Table 4.5. *The results of three-class classification*

Metric	Classifiers			
	PSO-LSTM-CNN	PSO-LSTM	CNN-LSTM	LSTM
Ov. Acc.	96.92	93.62	86.55	78.82
Av. Acc.	98.78	97.46	94.63	94.11
Ov. Error Rate	2.44	5.27	12.01	18.27
Av. Error Rate	1.22	2.54	5.37	7.31
Mac.-Av. Prec.	98.11	95.87	91.41	78.93
Mac.-Av. Rec.	92.30	86.21	69.84	78.82
Mac.-Av. F1-Score	95.12	90.78	79.18	78.82
Mic.-Av. Prec.	96.92	93.62	86.55	78.82
Mic.-Av. Rec.	96.92	93.62	86.55	78.82
Mic.-Av. F1-Score	96.92	93.62	86.55	78.82

The significance of three-class classification lies in its capacity to differentiate events within electrical networks, particularly for data encompassing physical attacks, cyber attacks, and occurrences outside normal situations. This renders three-class classification more crucial and intricate compared to binary classification. While numerous methods for binary classification excel in detecting anomalous data, they frequently encounter difficulties in accurately classifying attacks and managing large sample sizes in the context of three classes. Multi-class classification metrics (given in Table 4.3.) are employed to assess the performance of the proposed model for triple classification.

4.3.3. Test results based on multi-class dataset

In binary classification, two states are labeled: 0 for the normal operating state and 1 for the under attack state. However, for the selected multi-class data, there

Table 4.6. *The results of multi-class classification*

Metric	Classifiers								
	PSO-LSTM-CNN	PSO-LSTM	CNN-LSTM	LSTM	multi-CNN [90]	EDNN [97]	Ensemble [98]	DBN [99]	SACS-SAE [100]
Ov. Acc.	95.80	92.44	82.05	74.70	81.33	89.40	90.48	95.60	93.70
Av. Acc.	98.94	98.12	95.55	89.90	-	-	-	-	-
Ov. Error Rate	2.59	6.21	16.08	23.11	-	-	-	-	-
Av. Error Rate	1.26	3.88	5.55	10.15	-	-	-	-	-
Prec.	-	-	-	-	-	89.80	-	-	93.2
Rec.	-	-	-	-	-	89.40	-	-	-
F1-Score	-	-	-	-	-	88.20	-	-	-
Mac.-Av. Prec.	95.81	92.46	82.13	83.34	-	-	-	-	-
Mac.-Av. Rec.	95.80	92.43	82.05	50.45	-	-	-	-	-
Mac.-Av. F1-Score	95.80	92.45	82.09	62.85	-	-	-	-	-
Mic.-Av. Prec.	95.80	92.44	82.05	74.72	-	-	-	-	-
Mic.-Av. Rec.	95.80	92.44	82.05	74.72	-	-	-	-	-
Mic.-Av. F1-Score	92.80	92.44	82.05	74.72	-	-	-	-	-

are 41 different conditions, each labeled from 0 to 41.

As shown in Table 4.6., the performance metrics of the models in this study decrease as the number of class examples in the dataset increases. Strengthening the classification ability of the model is essential for an effective intrusion detection system. Multi-class classification requires more complex learning for data analysis and separation compared to binary classification. Therefore, optimizing the hyperparameters of the LSTM and CNN algorithms is crucial for the proposed approaches. The results for the PSO-CNN-LSTM algorithm demonstrate that the diversity of attacks in the data significantly impacts the model's effectiveness. Consequently, the classifiers used in the initial study are chosen for multi-class classification in the proposed MGs, yielding successful results.

The results of the first study illustrate why CNN and LSTM models are currently preferred for the CPA detection problem. This preference is largely due to the challenges of detecting well-disguised CPAs and protecting data streams. Classifying attacks and disturbances is a complex issue for control units in connected SG networks. Deep learning (DL) techniques are increasingly utilized to detect such

attacks, but the parameters of a deep neural network (DNN) need to be optimized for both detection rate and classification accuracy. In this study, a PSO-based hybrid CNN-LSTM approach is deemed suitable for comparing these techniques. After applying the optimization algorithm, the optimal number of hidden units and cell types are determined. The results underscore the importance of optimizing the hyperparameters of deep learning models. The evaluation reveals that detection speed depends on different combinations of DL models, highlighting the need for tailored optimization strategies.

DL algorithms achieve a high detection rate with specific datasets (given in Table 4.4.-4.6.). However, comparing the performance of DL algorithms with different combinations for all types of attacks does not yield entirely accurate findings. This is because the datasets used are obtained from different power grids and communication networks. When the performance of the proposed model is compared with other approaches, it is evident that most studies report only accuracy rates for dimension reduction and optimization models based on DL algorithms. In light of these findings, further research is needed on out-of-sample validation for all DL models and the use of comprehensive datasets.

The proposed approaches are based on hybrid models with hyperparameter optimization from the first study. The main objective is to promptly address erroneous data and related issues, thereby enhancing detection success. Various approaches have been suggested, including the utilization of Bloom filters [93] and the integration of the self adaptive cuckoo search algorithm (SACS) and stack auto-encoding (SAE) [100], as evidenced in existing literature. These methods can be further enhanced through additional optimization models, surpassing other fully supervised machine learning methods. Optimal results are often achieved when supervised models are developed using optimized frameworks, as illustrated in the example from [95].

Enhancing the performance of LSTM networks operating on stream arrays for integrated physical-cyber attacks can significantly benefit the development of hybrid models. Increasing network complexity has been shown to lead to improved performance, as demonstrated in [96]. Additionally, the proposed PSO-based CNN-LSTM model showed comparable performance to DBN [99], achieving 95% accuracy

on various PMU and SCADA datasets. However, it's important to note that the comparison of deep learning (DL) approaches in the literature lacks consistency due to variations in the operating regions of SG networks.

4.4. Cyber-Physical Attack Detection based MG Approaches

The effect of the proposed three scenarios scheme on the overall attack detection strategy is analyzed in this section.

4.4.1. Simulation scenarios

The proposed game models are validated under three different scenarios (S). In S-1, physical attacks on buses, transmission lines, and relays result in permanent delays in communication links. S-2 involves attacks causing latencies between 10 ms and 30 ms on communication links associated with PMUs, which are sub-units of PDCs. S-3 defines variable network latency on communication links connecting PMUs to PDCs. This assigned delay includes two types of delays: one attributed to the buses, as described in the methodology section, and the other between communication units. The source of these delays, whether physical or due to communication line issues, is examined based on threshold values ranging from 10 ms to 60 ms.

The attacker's messages are designed to send tripping signals to linked switches. This implies that attacker messages for new attacks are issued when there is an abnormality in the network. Therefore, in a healthy system, attacker messages are not expected to fluctuate sharply. Events are very frequent and mostly consist of cyclic attacker messages in the SG.

4.4.2. MG based shapley equations and PSO-LSTM

Compared to the game problems mentioned earlier, the defending agents expand their decision making mechanism by encountering complex situations due to the existence of three different sub-environments affected by the actions of the attacking agents. In addition, under certain scenarios, an attacker's action can only be performed when the attacker can make enough observations at that location. In addition, instant attack effects and types are shown in Figure 4.1. If examined in detail, the Voltage measurement graph for a day in Figure 4.2 includes three differ-

ent states, respectively, SWG_9 fault, which refers to the fault condition occurring in the line between the 50th and 51th busbars, and the normal system state. At the beginning of the game, the reward value is given as '0' for the attacker and the defender. In the continuation of the game, the attacker and the defender accumulate prizes by making strategic moves towards the points they target. The player who accumulates a sufficient amount of reward gains an advantage for himself because it knows the environment better. For example, this provides the attacker with the opportunity to gain unauthorized access to more points and layers.

Based on the characteristic of the MG, a single agent tries to explore the environment by operating multiple PMUs and maximizing the expected reward requires learning a policy based on that environment. There is a complex area in this decentralized control system. Agents in the system must have the ability to find effective solutions to the problem through competition with guardians or coordination with other agents. As the environment space expands, additional complexities arise. Representative exploration and action selection are coordinated with the reformulated MDP algorithm. The actor-criticism system naturally incorporates the actions of the agent into the policy iteration procedure, and a classification algorithm is also used in conjunction with the LSTM-PSO, whose effectiveness has been proven in the previous study.

The three interconnected layers of the SG system are defined as the subperipheries of the network with the help of the RL algorithm. ISQV-MARL is an improved approach of the RL algorithm with the addition of the LSTM model, optimizing the system parameters. Rewarding players, as defined in the equations mentioned earlier, is designed to adapt to different scenarios shown in Figure 4.3 to 4.6. The interaction between PMU-BUS and PMU-PDC forms the basis of the strategies developed in the MG of the offensive and defensive orbital agents shaped by LSTM networks and the MARL approach. The multi-agent reward function takes into account the past action of each agent and provides a holistic consideration of the rightness or wrongness of actions during the interactions of each agent in the observed environments. Also, inferring the accuracy of the attacker's actions based on past reward values presents a challenge for the defender due to the existence of different sub-environments and the different definition of the locations of

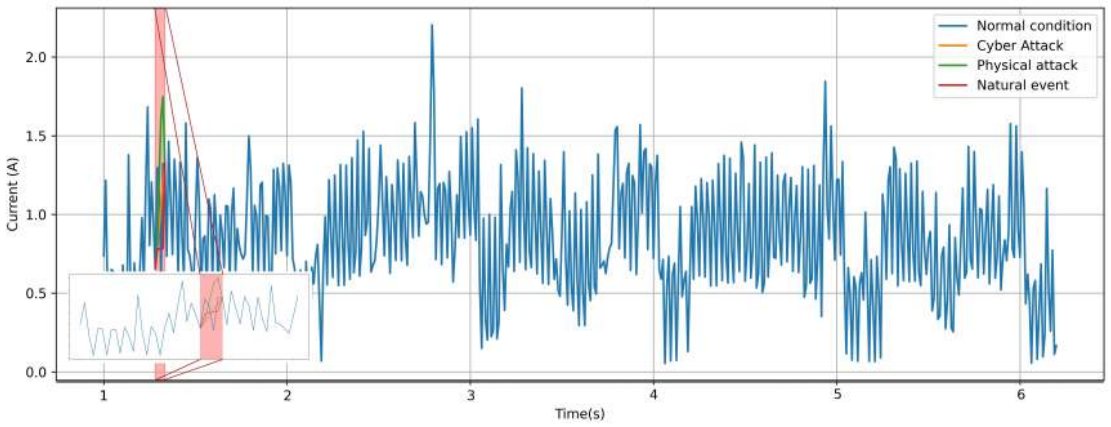


Figure 4.1. *Measured current values from PMU_{15} for attacked and normal condition*

the targeted points on them.

To give an example specifically, an attacker tries to gain control of a switch when it observes the sub-environment based on past actions. Simultaneously, the defender using its observations must find a trace by monitoring the attacker's actions that cause changes in PMU measurements. Awards are collected in the working environment according to the right decisions made. Initially, the environment represents normal operating conditions in the domain. Each observation made during the game guides the agents in their mission to control or break the keys.

The spoofing effect and coordinated cyber-physical attacks are presented in the error Table 4.7.. ISQV-MARL algorithm with LSTM has proven to be effective in addressing and detecting the mentioned problems for SG security.

Figure 3.2 depicts the measurements taken by the PDCs and collected at the control center, specifically in the 123-bus system where the target PMU is located. The attack is initiated by altering the timestamp of the software PMU at the specified time shown in Figure 4.7. As illustrated in Figure 4.7, there is a significant drop in the measurements collected from the relevant PMU in the presence of the attack. Figure 4.7 also specifies the scenario where busbars under attack lead to permanent measurement loss. In this attack, coupled with SW_7, the loss of measurements from Bus 160 results in the disappearance of observability and control capability. Throughout the attack, some measurements taken from the target PMUs are not dropped by the connected PDC. This occurrence is attributed to slight changes in

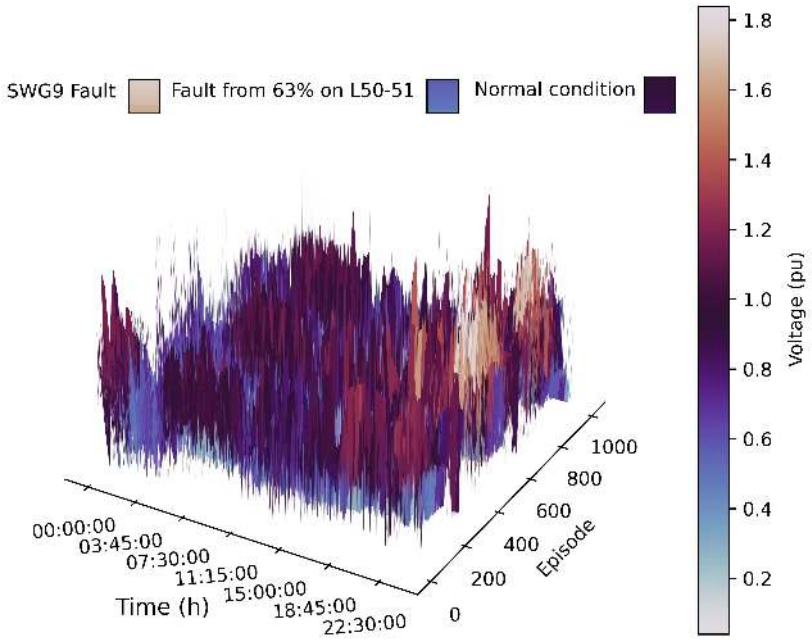


Figure 4.2. *Voltage measurements of PMU_{15} for attacked and normal operating conditions*

network latency, allowing measurements from these PMUs to reach the corresponding PDC before the waiting time expires. This is because the model transmits the relevant data to the PDC based on the network delay at a particular moment.

As depicted in Figure 4.8(a), in the presence of a cyber attack, measurements from the target PMU are effectively transmitted to the connected PDC. However, when the delay in the power system coincides with the cyber attack, the same outcome is not observed, resulting in a significant decrease in data transmission to the relevant PDC. Furthermore, as illustrated in Figure 4.9(a), when delays in the communication network accompany both physical and cyber attacks, a severe disruption in information flow occurs at the PDC until the defense agents detect the attack and its location. Consequently, alterations in the physical and cyber aspects of the network impact the arrival of measurements, thereby affecting the extent to which the PDC is influenced and receives measurements from the PMU.

Compared to the previously mentioned game problems, defensive agents face more challenging situations due to the existence of a single layer in the game and the sub-environment that is influenced by the actions of the offensive agents. These actions affect the layer, and certain actions can only be taken if an attacker's move

results in a specific trace at a particular location. To examine this in detail, Figure 4.2 shows a voltage measurement graph for a day under three types of conditions. Initially, both the attacker and defender have a reward value of zero. As the game progresses, the attacker accumulates rewards by making strategic moves toward the target location. With a sufficient amount of accumulated rewards, the attackers become capable of executing more impactful actions that inflict greater damage on specific locations within the environment.

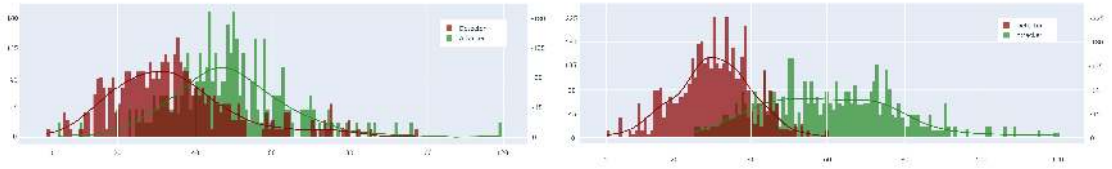


Figure 4.3. *Attacker-Defender Reward at PDC_1 and PDC_2 in ms for S-2.*

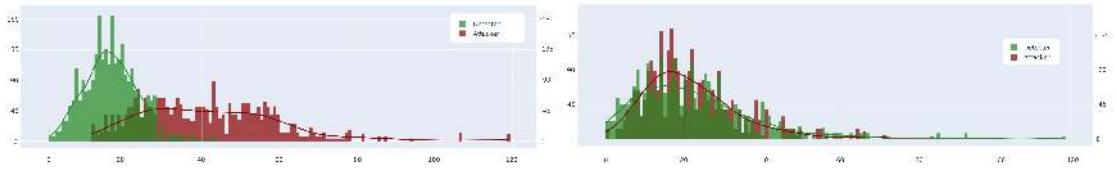


Figure 4.4. *Attacker-Defender Reward at PDC_3 and PDC_4 in ms for S-2.*

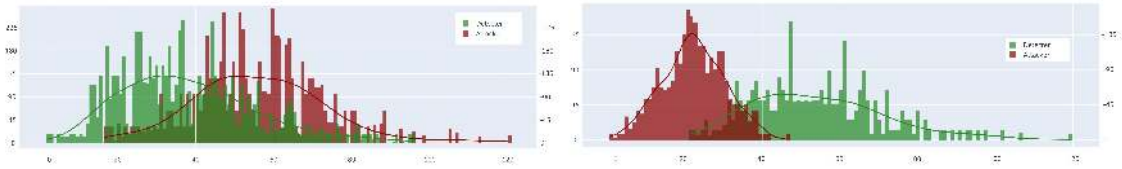


Figure 4.5. *Attacker-Defender Reward at PDC_5 and PDC_6 in ms for S-2.*

Specifically, when an attacker observes the environment based on past actions, they attempt to manipulate a switch. Simultaneously, the defender, utilizing their observations, must track the attacker's actions that result in changes in the PMU measurements. Rewards are accumulated in the studied environment based on the correct decisions made. Initially, the environment represents normal operating conditions within the domain. Each observation made during the game guides the agents in their task of controlling or disrupting the switches.

As a result, an insightful analysis of the MDP-based FDIA detection approach reveals three key aspects. First, the MDP game approach has been effectively uti-

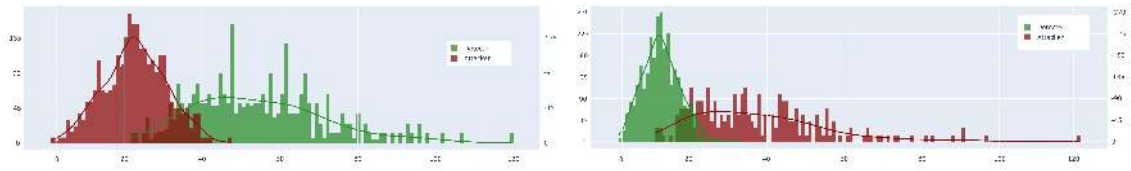


Figure 4.6. *Attacker-Defender Reward at PDC_7 and PDC_8 in ms for S-2.*

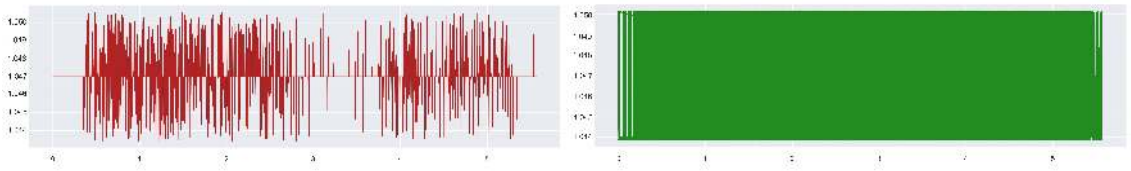


Figure 4.7. *a) Attacked and b) Corrected voltage measurements delivered to PDC_4 for S-1.*

lized as a defender, allowing for efficient detection of FDIA. Second, as an attacker, the adversary strategically launches attacks on the vulnerable nodes. Third, the game construction involves the integration of virtual junction points of the PMUs to form the PDC, creating a comprehensive environment. Throughout the game, both the defender and the attacker exhibit strategic behavior in response to specific states. The defender's strategy aims to maximize rewards, while the attacker seeks to minimize the defender's rewards by exploiting the most vulnerable pathways in the complex environment.

The errors associated with the spoofing effect and physical/cyber attacks are presented in the error Table 4.7.. The thesis summarizes all the results into the ISQV-MARL algorithm with LSTM, which proves to be effective in addressing these issues. By adopting the LSTM algorithm in the context of MARL, the approach provides insightful and powerful means to empower one side of the players. Based on the results, the protection of the PMU-PDC and accordingly BUS-PMU layers of the SG network is completed less error.

First, the proposed defender strategy achieves a superior performance with LSTM-PSO. Secondly, distinguishing the spoofing signal is a challenging process as the attack strategy carried out on the busbars is included in the cyber attack. Finally, LSTM is included for decentralized control and centralized learning by taking advantage of its good classifier feature. With these integrated stages, it has significantly improved control performance in terms of convergence and stability of learning, as well as reducing data losses and cyber attack leaks.

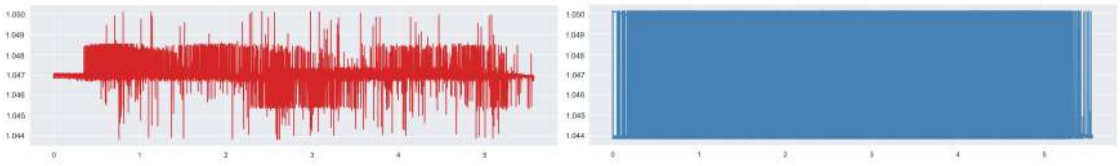


Figure 4.8. *a) Attacked and b) Corrected voltage measurements delivered to PDC_4 for S-2.*

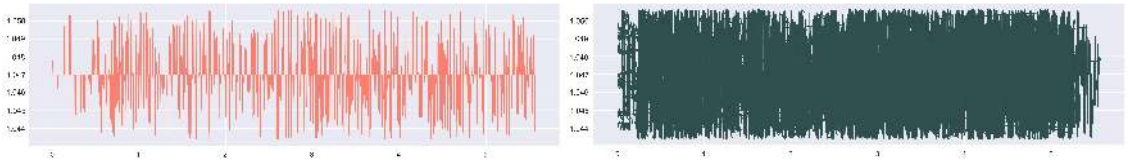


Figure 4.9. *a) Attacked and b) Corrected voltage measurements delivered to PDC_4 for S-3.*

4.4.3. MG based DE with CNN-LSTM

IEEE 123 bus system includes three layers in the proposed game, these three layers are connected to each other by BUS-PMU-PDC. An improved version of the system, the proposed DE-MG is evaluated in this study by including the normal operating conditions and fault situations of the system. Moreover, to improve the effectiveness of the proposed framework, MADRL and RL techniques manage contemporary LSTM-CNN agents.

Based on the Markov property, a single agent explores the environment by operating multiple PMU units and learns a policy to maximize the expected reward. In this decentralized control system, there exists a complex domain. Agents in this system are required to possess the capability to discover effective solutions to the problem, either through competition or coordination with others. As the environment space expands, additional complexity arises. Representative exploration and action selection are coordinated using the reformulated MDP. The actor-critic system inherently incorporates the agent's actions into the policy iteration procedure and will also be used in conjunction with a classification algorithm.

In addition to the above-mentioned explanations, it is not enough to evaluate errors in the network information flow only based on the information coming from a single agent in a certain time period, so the decision is made according to the central decision maker. State transitions occurring in nodes are important in terms

Table 4.7. *Mean square tracking errors for some disturbance and sub-domains*

PDC waiting time	Target (PMU, PDC)	θ_1	θ_2	Attack Type Error
10ms	15, 1	0.079	0.004	0.011
30ms	20, 1	0.014	0.017	0.061
30ms	7, 2	0.043	0.007	0.032
30ms	11, 2	0.043	0.008	0.363
10ms	1, 3	0.534	0.012	0.124
30ms	5, 3	0.557	0.018	0.114
10ms	21, 4	0.016	0.005	0.235
20ms	25, 4	0.016	0.009	0.523
30ms	27, 5	0.043	0.003	0.431
30ms	30, 5	0.043	0.008	0.313
10ms	41, 6	0.646	0.015	0.130
30ms	44, 6	0.397	0.009	0.110
30ms	36, 7	0.036	0.013	0.156
30ms	39, 7	0.038	0.012	0.176
20ms	32, 8	0.676	0.014	0.168
30ms	34, 8	0.870	0.018	0.188

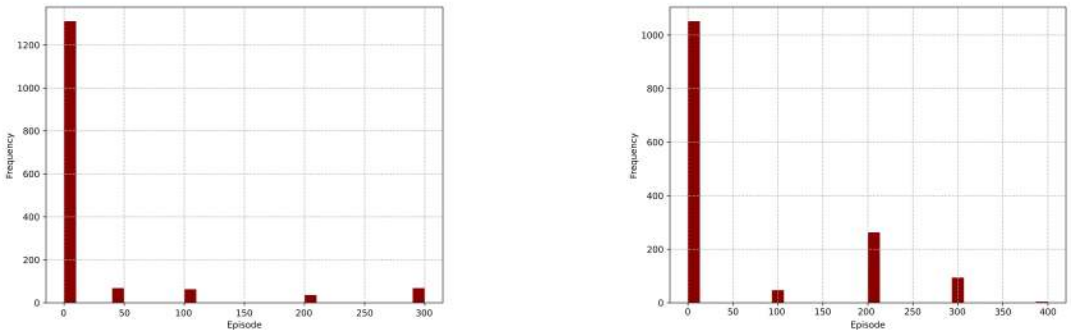
Table 4.8. *Comparison for cyber-physical security, threats and resilience strategies in SGs*

Metric	Ov. Acc.	Av. Acc.	Ov. Error Rate	Av. Error Rate	Mac.-Av. Prec.	Mac.-Av. Rec.	Mac.-Av. F1-Score	Mic.-Av. Prec.	Mic.-Av. Rec.	Mic.-Av. F1-Score
Game_1										
S-1	99.37	99.77	1.01	0.25	99.38	99.39	99.39	99.40	99.41	99.42
S-2	99.15	99.75	1.47	0.46	99.03	99.01	99.01	99.02	99.03	99.02
S-3	99.08	99.42	1.53	0.51	99.01	99.00	99.00	99.01	99.01	99.00
Game_2										
S-1	99.64	99.86	0.97	0.19	99.67	99.51	99.42	99.54	99.68	99.79
S-2	99.57	99.82	1.12	0.21	99.65	99.49	99.53	99.49	99.61	99.68
S-3	99.43	99.58	1.25	0.23	99.45	99.41	99.47	99.47	99.57	99.62

of showing the protection vulnerability and protection success in the system. For instance, even though the detection rate is quite high in the three different scenarios for all time periods (as given in Table 4.8.). The reason for this is that the defense expresses the detection result of the agent when it is not yet working under central control.

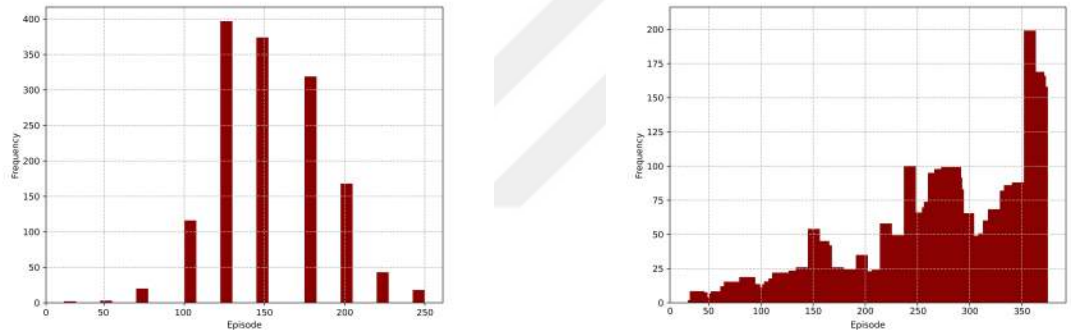
The result shows evolutionary process of the proposed approach. Figure 4.10 to 4.12 illustrate the state transitions of agents after the LSTM-CNN evolution on the MG. In addition, its variation decrease as the number of generations increases, while the verification accuracy of the network becomes better and its variation becomes smaller. Put differently, the performances of CNN-LSTM increase and the complexities of the CNN-LSTM model decrease until the final stage of the evolution process of DE-CNN-LSTM. To observe the trade-off between population size and agent performances proportional to training time, the total population number is

considered as the variable to identify attack agents under unsupervised learning of Nash.



(a) Bus_8 (b) Bus_{13}

Figure 4.10. *Transition for S^{ud} during cyber attacks on PMU_2.*

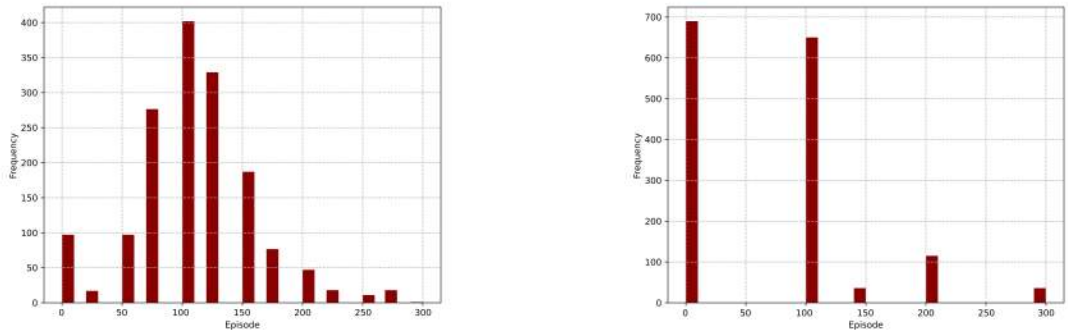


(a) Bus_1 (b) Bus_4

Figure 4.11. *Transition for S^{cd} during cyber attacks on PMU_3.*

By evaluating the attack detection performance of the proposed Markov-DE method, effective inferences about the model are obtained by comparing the proposed method with other data-based detection techniques. The detection system is a method developed for cyber networks in SG, first stage optimization consists of differential equations, second stage optimization of the Markov-DE method, which it is applied for multi-label classification tasks in the field of intrusion detection, and moved it to the CPA detection. Markov-DE's CNN-LSTM based detection mechanism achieves spatially aggressive intrusion localization detection.

The DE-Markov method is recommended to determine the ideal defense mechanisms and precise locations of busbar, PMU and PDC units, as well as to determine



(a) Bus_{24} (b) Bus_{25}

Figure 4.12. *Transition for S^a during cyber attacks on PMU_9.*

the optimal strategies of defenders, and the following procedural steps describe:

Step 1: Changes in network information, communication information and important points of the data flow diagram must be detected along with the setting parameters for the game diagram (Table 3.4.).

Step 2: Initialize the MG framework by creating an initial population. Each population contains optimization of hyperparameters of LSTM-CNN agents for each new case and information on the attack location.

Step 3: Apply the DE-MG with attacker behaviors and system faults in the electrical network environment. The DE algorithm calculates the optimal strategy state of the agents created by the MG in the optimum position.

Step 4: With the ideal security strategy generated by the DE algorithm, apply the DE optimization to each agent to calculate the state, action, and environmental changes for each population set.

Step 5: Check whether the new solutions can be implemented within the options that meet the constraints specified in Section 3.3 in case of increased attack rate and capture of important units.

4.4. Multi-agent System Limitations

The integration of RL into SG holds significant prospects for ensuring cyber-security, identifying and mitigating possible limitations and challenges is of great importance to increase the effectiveness of these models. One of the challenging

elements arising from the complex and dynamic nature of cyber attacks requires real-time creation of more realistic simulation environments and network configurations of current communication systems, on which possible attack scenarios can be developed.

RL learns about the environment by making mistakes and then avoiding them from previous experiences, imitating the learning process of humans. It can solve some problems that traditional methods cannot solve. It also has the ability to outperform humans in some tasks. However, RL also has some limitations. First of all, too much supplementation can cause situations to become overloaded, resulting in diminished results. Additionally, RL assumes that the environment is a Markov model in which the probability of events occurring depends only on previous states. In this case, the LSTM model discovers more quickly that not only are there cyber attacks in the environment, but physical disturbances from the sub-environment will also exist.

This study addresses the SG security problem in a centralized system task for the 123 Bus system and demonstrates strong empirical performance against attackers. However, it should be noted that the proposed approach controls the actions of attacker-defender agents. Future research topics include exploring the uncertainty of power distribution models for each player in relation to the subspaces. However, this introduces complexity to the interactions among agents and their neighbors, making it challenging to maintain global interactions among agents operating in different domains.

The importance of addressing attacks occurring in cyber networks is emphasized by the network-wide malfunction when considering the temporal distribution of information flow from the buses due to faulty data transfer between network layers caused by data manipulation of intrusions. As mentioned in the previous sections, 'advanced' algorithms for the development of faulty data detection systems have shown that low life cycle is an effective way to detect faults and attacks immediately and prevent these attacks as much as possible.

5. CONCLUSION AND FUTURE RESEARCH

In Section 5.1, the concluding remarks of this thesis are given. Then, in Section 5.2, regarding the present contributions of this thesis, the possible research objectives for the future are discussed.

5.1. Conclusion

The main objective of this thesis is to develop a deep learning-based CPA detection algorithm capable of identifying and classifying multi-class attacks. Additionally, the thesis aims to design and develop an attack-defense system. This system leverages a structured PMU-PDC environment, enriched with both attacked and normal datasets, to enhance its effectiveness.

The most decisive contributions of this step-by-step study are summarized as follows:

- Intruder detection and attack type discrimination: This is achieved through a holistic approach that integrates the PSO-LSTM hybrid model with Multi-Agent Reinforcement Learning (MARL). This integration effectively detects attacks against defined targets.
- Effectiveness in Case studies: The proposed approaches have demonstrated effectiveness in three case studies. These involve distinguishing dynamic instability caused by communication layer errors, power grid faults, and outages.
- Innovation in detection accuracy: The first study introduces an innovation by optimizing the parameters of the hybrid CNN-LSTM model with the PSO algorithm for CPA in SG. Results indicate that the PSO-CNN-LSTM approach is more robust for detecting data injection, remote opening command injection, attack subtypes (such as command injection against a single relay), and relay setting-changing attacks. It also outperforms other methods in predicting samples during natural events like SLG failures and line maintenance.
- Superior performance in MG: The second study proposes a MG modeled with PSO-CNN-LSTM, which performs better than other machine learning models, showing less error margin and higher accuracy. Players in the study optimize their policies based on the Shapley value, with the LSTM-CNN model used in the classification stage. The study confirms the approach's accuracy by

achieving superior results in system defensive CPA detection and attack discrimination.

- Enhanced Monitoring and Learning Rate: The achievement values and policy updates of the players based on the game facilitate easier monitoring of the detection mechanism's functioning and learning rate. In the DE-based MG, the hyperparameters of CNN-LSTM agents are updated using the evolution algorithm, which increased detection success compared to the first proposed game.

Due to the large dataset of power and information flow in the interconnected SG system, more empirical studies are needed to solve this problem. Additionally, these approaches require more data from advanced systems to ensure their performance against powerful cyber attacks and natural events.

To reduce observation costs and enhance the effectiveness of reward designs in training multi-agent games, policies and reward structures can be improved compared to traditional algorithms. These enhancements aim to optimize the learning process and achieve better performance in multi-agent settings.

5.2. Future Research

Future studies aim to further improve the performance of the proposed models and evaluate them from different perspectives by integrating various approaches. The focus will be on extending performance analyses to prevent integrated cyber attacks, multiple data manipulations, and common network failures encountered at cyber-physical layers.

Another crucial issue is ensuring the privacy and security of the RL model itself. To address this concern, both decentralized federated learning and centralized learning approaches will be considered. These approaches aim to protect the attacker's actions and enhance privacy during the model update process. By incorporating these privacy-preserving mechanisms, the improved approach will meet the essential requirements of real-world applications, providing a robust and secure framework for training RL models.

REFERENCES

[1] Ghiasi, M., Niknam, T., Wang, Z., Mehrandezh, M., Dehghani, M., Ghadimi, N. (2023). A comprehensive review of cyber-attacks and defense mechanisms for improving security in smart grid energy systems: Past, present and future. *Electric Power Systems Research*, 215, 108975.

[2] Shobole, A. A., Wadi, M. (2021). Multiagent systems application for the smart grid protection. *Renewable and Sustainable Energy Reviews*, 149, 111352.

[3] Dong, S., Zhan, J., Hu, W., Mohajer, A., Bavaghar, M., Mirzaei, A. (2023). Energy-efficient hierarchical resource allocation in uplink-downlink decoupled NOMA HetNets. *IEEE Transactions on Network and Service Management*.

[4] Singh, N. K., Mahajan, V. (2021). End-User Privacy Protection Scheme from cyber intrusion in smart grid advanced metering infrastructure. *International Journal of Critical Infrastructure Protection*, 34, 100410.

[5] Sayed, K., Gabbar, H. A. (2017). SCADA and smart energy grid control automation. *Smart Energy Grid Engineering*, 9, 481–514.

[6] Gumaei, A., Hassan, M. M., Huda, S., Hassan, M. R., Camacho, D., Del Ser, J., Fortino, G. (2020). A robust cyberattack detection approach using optimal features of SCADA power systems in smart grids. *Applied Soft Computing*, 96, 106658.

[7] Cui, L., Qu, Y., Gao, L., Xie, G., Yu, S. (2020). Detecting false data attacks using machine learning techniques in smart grid: A survey. *Journal of Network and Computer Applications*, 170, 102808.

[8] Ji, Y., Wang, J., Jiacan, X., Li, D. (2021). Data-Driven Online Energy Scheduling of a Microgrid Based on Deep Reinforcement Learning. *Energies*, 14(8), 2120.

[9] Zou, H., Tao, J., Elsayed, S. K., Elattar, E. E., Almalaq, A., Mohamed, M. A. (2021). Stochastic multi-carrier energy management in the smart islands using reinforcement learning and unscented transform. *International Journal of Electrical Power Energy Systems*, 130, 2120.

[10] Kou, P., Liang, D., Wang, C., Wu, Z., Gao, L. (2020). Safe deep reinforcement learning-based constrained optimal control scheme for active distribution

networks. *Applied Energy*, 264, 114772.

[11] Li, H., Wan, Z., He, H. (2020). Real-time residential demand response. *IEEE Transactions on Smart Grid*, 11(5), 4144-4154.

[12] Azar, A.T., Koubaa, A., Nada, A. M., Ibrahim, H. A., Ibrahim, Z. F., Kazim, M., Ammar, A., Benjdira, B., Khamis, A. M., Hameed, I. A., Casalino, G. (2021). Drone deep reinforcement learning: A review. *Electronics*, 10(9), 999.

[13] Liu, X. K., Wen, C., Xu, Q., Wang, Y. W. (2021). Resilient control and analysis for dc microgrid system under DoS and impulsive FDI attacks. *IEEE Transactions on Smart Grid*, 12(5), 3742–3754.

[14] Xu, R., Wang, R., Guan, Z., Wu, L., Wu, J., Du, X. (2017). Achieving efficient detection against false data injection attacks in smart grid. *IEEE Access*, 5, 13787–13798.

[15] Guan, Y., Ge, X. (2017). Distributed attack detection and secure estimation of networked cyber-physical systems against false data injection attacks and jamming attacks. *IEEE Transactions on Signal and Information Processing over Networks*, 4(1), 48–59.

[16] Manandhar, K., Cao, X., Hu, F., Liu, Y. (2014). Detection of faults and attacks including false data injection attack in smart grid using Kalman filter. *IEEE transactions on control of network systems*, 4(1), 612–621.

[17] Sun, M., He, L., Zhang, J. (2022). Deep learning-based probabilistic anomaly detection for solar forecasting under cyberattacks. *International Journal of Electrical Power Energy Systems*, 137, 107752.

[18] Liu, L., Esmalifalak, M., Ding, Q., Emesih, V. A., Han, Z. (2014). Detecting false data injection attacks on power grid by sparse optimization. *IEEE Transactions on Smart Grid*, 5(2), 612–621.

[19] Wang, J., Hui, L. C. K., Yiu, S. M., Wang, E. K., Fang, J. (2017). A survey on cyber attacks against nonlinear state estimation in power systems of ubiquitous cities. *Pervasive and Mobile Computing*, 39, 52–64.

[20] Liu, X., Li, Z. (2017). False data attacks against AC state estimation with incomplete network information. *IEEE Transactions on Smart Grid*, 8(5), 2239–2248.

[21] Deng, Q., Sun, J. (2018). False data injection attack detection in a power grid using RNN. In: *IECON 2018-44th Annual Conference of the IEEE Industrial Electronics Society*, pp. 5983–5988.

[22] Su, Q., Wang, H., Sun, C., Li, B., Li, J. (2022). Cyber-attacks against cyber-physical power systems security: State estimation, attacks reconstruction and defense strategy. *Applied Mathematics and Computation*, 413, 126639.

[23] Esmalifalak, M., Nguyen, H., Zheng, R., Han, Z. (2011). Stealth false data injection using independent component analysis in smart grid. In: *2011 IEEE international conference on smart grid communications (SmartGridComm)*, 244–248.

[24] Bitirgen, K., Başaran Filik, Ü. (2022). Performance Analysis of PCA Based Machine Learning Approaches on FDIA Detection. *International Journal of Information Security Science*, 11(4), 1–13.

[25] Nawaz, R., Akhtar, R., Shahid, M. A., Qureshi, I. M., Mahmood, M. H. (2021). Machine learning based false data injection in smart grid. *International Journal of Electrical Power & Energy Systems*, 130, 106819.

[26] Chen Y., Huang S., Liu F., Wang Z., Sun X. (2018). Evaluation of reinforcement learning-based false data injection attack to automatic voltage control. *IEEE Transactions on Smart Grid*, 10(2), 2158–2169.

[27] Bitirgen, K., Filik, U. B. (2023). A hybrid deep learning model for discrimination of physical disturbance and cyber-attack detection in smart grid. *International Journal of Critical Infrastructure Protection*, 40, 100582.

[28] Rezaei, P., Hines, P. D. H., Eppstein, M. J. (2014). Estimating cascading failure risk with random chemistry. *IEEE Transactions on Power Systems*, 30(5), 2726–2735.

[29] Çolpan, E., Mohammed, A. A., Gerek, O. N. (2022). Object Recognition with Sequential Decision Reinforcement of Deep Learning. In: *2022 30th Signal Processing and Communications Applications Conference (SIU)*, 1–4.

[30] Wan, Y., Dragičević, T. (2022). A Data-driven cyber-attack detection of intelligent attacks in islanded dc microgrids. *IEEE Transactions on Industrial Electronics*, 70(4), 4293–4299.

[31] Shereen, E., Kazari, K., Dán, G. (2023). A Reinforcement Learning Approach to Undetectable Attacks against Automatic Generation Control. *IEEE Transactions on Smart Grid*, 15(1), 959 - 972.

[32] Arulkumaran K., Deisenroth M. P., Brundage M., Bharath A. A. (2017). Deep reinforcement learning: A brief survey. *IEEE Signal Processing Magazine*, 34(6), 26–38.

[33] Liu, Y., Cheng, L. (2022). Relentless False Data Injection Attacks against Kalman Filter Based Detection in Smart Grid. *IEEE Transactions on Control of Network Systems*, 9(3), 1238–1250.

[34] Bitirgen, K., and Filik, Ü. B. (2023). Multi-Defender Strategic Filtering Against Multi Agent Cyber Epidemics on Multi-Environment Model for Smart Grid Protection. In: *The International Conference on Energy and Green Computing (ICEGC'2023)*, 469, 00095.

[35] Bitirgen, K., Filik, U. B. (2024). Markov game based on reinforcement learning solution against cyber-physical attacks in smart grid. *Expert Systems with Applications*, 124607.

[36] Huang, Y., He, H. (2022). Advance learning technique for the electricity market attack detection. *Computers and Electrical Engineering*, 100, 107865.

[37] Rahman, M. A., Al-Shaer, E., Kavasseri, R. (2014). Impact analysis of topology poisoning attacks on economic operation of the smart power grid. In: *2014 IEEE 34th International Conference on Distributed Computing Systems*, 649–659.

[38] Kurt, M. N., Oyetunji, O., Chong, L., Wang, X. (2018). Online cyber-attack detection in smart grid: A reinforcement learning approach. *IEEE Transactions on Smart Grid*, 10(5), 5174–5185.

[39] Das, S., Suganthan, P. N. (2010). Differential evolution: A survey of the state-of-the-art. *IEEE transactions on evolutionary computation*, 15, 4–31.

[40] Ahmad, M. F. I., Isa, N. A. M., Lim, W. H., Ang, K. M., (2022). Differential evolution: A recent review based on state-of-the-art works. *Alexandria Engineering Journal*, 61(5), 3831–3872.

[41] Lu, K.-D., Wu, Z.-G. (2022). Constrained-differential-evolution-based stealthy

sparse cyber-attack and countermeasure in an ac smart grid. *IEEE Transactions on Industrial Informatics*, 18(8), 5275–5285.

[42] Adeli, M., Hajatipour, M., Yazdanpanah, M. J., Hashemi-Dezaki, H., Shafieirad, M. (2022). Optimized cyber-attack detection method of power systems using sliding mode observer. *Electric Power Systems Research*, 205, 107745.

[43] Khandare, P., Deokar, S.A., Dixit, A.M. (2021). Optimization techniques using DWT-differentiation algorithms for fault detection and relay coordination in microgrid. *Electrical Engineering*, 103(1), 493–503.

[44] Lu, K.-D., Wu, Z.-G., Huang, T. (2022). Differential evolution-based three stage dynamic cyber-attack of cyber-physical power systems. *IEEE/ASME Transactions on Mechatronics*, 28(2), 1137–1148.

[45] Mlakar, M., Petelin, D., Tušar, T., Filipič, B. (2015). GP-DEMO: Differential Evolution for Multiobjective Optimization based on Gaussian Process models. *European Journal of Operational Research*, 243(2), 347-361.

[46] Yang, Y., Liu, J., Tan, S., Wang, H. (2019). A multi-objective differential evolutionary algorithm for constrained multi-objective optimization problems with low feasible ratio. *Applied Soft Computing*, 80, 42–56.

[47] Wang, S., Bi, S., Zhang, Y. J. A. (2020). Locational detection of the false data injection attack in a smart grid: a multilabel classification approach. *IEEE Internet Things Journal*, 7(9), 8218–8227.

[48] Kipf, T. N., Welling, M. (2017). Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

[49] Liao, W., Yang, D., Wang, Y., Ren, X. (2021). Fault diagnosis of power transformers using graph convolutional network. *CSEE Journal of Power and Energy Systems*, 7(2), 241–249.

[50] Takiddin, A., Atat, R. Ismail, M., Boyaci, O., Davis, K.R., Serpedin, E. (2023). Generalized graph neural network-based detection of false data injection attacks in smart grids. *IEEE Transactions on Emerging Topics in Computational Intelligence*.

[51] Han, Y., Feng, H., Li, K., Zhao, Q. (2023). False data injection attacks detec-

tion with modified temporal multi-graph convolutional network in smart grids. *Computers & Security*, 124, 103016.

[52] Javaid, N., Jan, N., Javed, M. U. (2021). An adaptive synthesis to handle imbalanced big data with deep siamese network for electricity theft detection in smart grids. *Journal of Parallel and Distributed Computing*, 153, 44-52.

[53] Ding, W., Xu, M., Huang, Y., Zhao, P., Song, F. (2021). Cyber attacks on PMU placement in a smart grid: Characterization and optimization. *Reliability Engineering & System Safety*, 212, 107586.

[54] Wu, S., Jiang, Y., Luo, H., Zhang, J., Yin, S., Kaynak, O. (2022). An integrated data-driven scheme for the defense of typical cyber–physical attacks. *Reliability Engineering & System Safety*, 220, 108257.

[55] Li, Z., Shahidehpour, M., Alabdulwahab, A., Abusorrah, A. (2016). Analyzing locally coordinated cyber-physical attacks for undetectable line outages. *IEEE Transactions on Smart Grid*, 9(1), 35–47.

[56] Tran, H.-Y., Hu, J., Yin, X., Pota, H. R., (2023). An Efficient Privacy-enhancing Cross-silo Federated Learning and Applications for False Data Injection Attack Detection in Smart Grids. *IEEE Transactions on Information Forensics and Security*, 18, 2538-2552.

[57] Li, Y., Wei, X., Li, Y., Dong, Z., Shahidehpour, M. (2022). Detection of false data injection attacks in smart grid: A secure federated deep learning approach. *IEEE Transactions on Smart Grid*, 13(6), 4862–4872.

[58] Ortega-Fernandez, I., Liberati, F. (2023). A Review of Denial of Service Attack and Mitigation in the Smart Grid Using Reinforcement Learning. *Energies*, 16(2), 108257.

[59] Shen, Y., Fei, M., Du, D. (2019). Cyber security study for power systems under denial of service attacks. *Transactions of the Institute of Measurement and Control*, 41, 1600–1614.

[60] Hu, Z., Liu, S., Luo, W., Wu, L. (2020). Resilient distributed fuzzy load frequency regulation for power systems under cross-layer random denial-of-service attacks. *IEEE Transactions on Cybernetics*, 52(4), 2396–2406.

[61] Yan, H., Wang, J., Zhang, H., Shen, H., Zhan, X. (2018). Event-based security control for stochastic networked systems subject to attacks. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 50(11), 4643–4654.

[62] Shen, Y., Fei, M., Du, D. (2019). Cyber security study for power systems under denial of service attacks, *Transactions of the Institute of Measurement and Control*, 41(6), 1600–1614.

[63] Amin, S., Cardenas, A. A., Sastry, S. S. (2019). Safe and secure networked control systems under denial-of-service attacks. *In: International Workshop on Hybrid Systems: Computation and Control*, pp. 31–45.

[64] Mousavian, S., Valenzuela, J., Wang, J. (2014). A probabilistic risk mitigation model for cyber-attacks to PMU networks. *IEEE Transactions on Power Systems*, 30, 156-165.

[65] Kalluri, R., Mahendra, L. Kumar, R. K. S., Prasad, G. L. G. (2016). Simulation and impact analysis of denial-of-service attacks on power SCADA. In *2016 national power systems conference (NPSC)*, pp. 1-50.

[66] Li, Y., Zhang, P., Ma, L. (2019). Denial of service attack and defense method on load frequency control system. *Journal Franklin Institute*, 356(15), 8625–8645.

[67] Mississippi State University Critical Infrastructure Protection Center, Industrial Control System Cyber Attack Data Set, in: https://sites.google.com/a/uah.edu/tommy-morris-uah/home

[68] He, Q.-Q., Wu, C., Si, Y.-W. (2022). LSTM with particle swam optimization for sales forecasting. *Electronic Commerce Research and Applications*, 51, 101118.

[69] Lin, Z., Yanwen, H., Jie, X., Xiong, F., Qiaomin, L., Ruchuan, W. (2021). Trust evaluation model based on PSO and LSTM for huge in formation environments. *Chinese Journal of Electronics*, 30(1), 92–101.

[70] Hu, Y., Wei, R., Yang, Y., Li, X., Huang, Z., Liu, Y., He, C., Lu, H. (2022). Performance Degradation Prediction Using LSTM with Optimized Parameters. *Sensors*, 22(6), 2407.

[71] Du, B., Huang, S., Guo, J., Tang, H., Wang, L., Zhou S. (2022). Interval forecasting for urban water demand using PSO optimized KDE distribution and LSTM

neural networks, *Applied Soft Computing*, 122, 108875.

[72] Wang, S., Bi, Y.-J., Zhang, A. (2020). Locational detection of the false data injection attack in a smart grid: A multilabel classification approach. *IEEE Internet of Things Journal*, 7(9), 8218–8227.

[73] Niu, X., Li, J., Sun, J., Tomsovic, K. (2019). Dynamic detection of false data injection attack in smart grid using deep learning. In *2019 IEEE Power Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pp. 1–6.

[74] Moalla, H., Elloumi, W., Alimi, A. M. (2017). H-PSO-LSTM: Hybrid LSTM trained by PSO for online handwriter identification. In *International Conference on Neural Information Processing*, pp. 41–50.

[75] NYISO. (2022). Dataset profile from.

[76] IEEE PES Test Feeder. (2022). IEEE 123 Bus Test Distribution Feeder.

[77] Šandi, S., Krstajić, B., Popović, T. (2016). pyPmu-Open source python package for synchrophasor data transfer. In 2016 24th Telecommunications Forum (TELFOR), pp. 1-4.

[78] Moussa, B., Al-Barakati, A., Kassouf, M., Debbabi, M., Assi, C. (2019). Exploiting the vulnerability of relative data alignment in phasor data concentrators to time synchronization attacks. *IEEE Transactions on Smart Grid*, 11(3), 2541–2551.

[79] Power System Relaying Committee and others. (2013). IEEE Guide for phasor data concentrator requirements for power system protection, control, and monitoring. IEEE: Piscataway, NJ, USA.

[80] Siamak, S., Dehghani, M., Mohammadi, M. (2020). Dynamic GPS spoofing attack detection, localization, and measurement correction exploiting PMU and SCADA. *IEEE Systems Journal* , 15(2), 2531–2540.

[81] Yasinzadeh, M., Akhbari, M. (2018). Detection of PMU spoofing in power grid based on phasor measurement analysis. *IET Generation, Transmission & Distribution*, 12(9), 1980–1987.

[82] Merrick, K., Maher, M. L. (2009). *Motivated Reinforcement Learning: Curious*

Characters for Multiuser Games, Springer Science & Business Media.

[83] Wang, J., Zhang, Y., Kim, T. K., Gu, Y. (2020). Shapley q-value: A local reward approach to solve global reward games. In: *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 7285–7292.

[84] Wang, B., Wu, Y., Liu, K. J. R., Clancy, T. C. (2011). An anti-jamming stochastic game for cognitive radio networks. *IEEE Journal on Selected Areas in Communications*, 29(4), 877–889.

[85] Wang, J., Zhang, Y., Gu, Y., Kim, T. K. (2022). SHAQ: Incorporating Shapley Value Theory into Multi-Agent Q-Learning. *Advances in Neural Information Processing Systems*, 35, 5941–5954.

[86] Sutton, R. S., Barto, A. G. (2018). *Reinforcement learning: An introduction*, MIT press.

[87] Price, K. V. (1999). *An introduction to differential evolution*, New ideas in optimization, 79–108.

[88] Liu, W., Zhong, S. (2017). Web malware spread modelling and optimal control strategies. *Scientific reports*, 7(1), 42308.

[89] Sayin, M., Zhang, K., Leslie, D., Basar, T., Ozdaglar, A. (2021). Decentralized Q-learning in zero-sum Markov games. *Advances in Neural Information Processing Systems*, 34, 18320–18334.

[90] Li, Y., Xu, Y., Liu, Z., Hou, H., Zheng, Y., Xin, Y., Zhao, Y., Cui, L. (2020). Robust detection for network intrusion of industrial IoT based on multi-CNN fusion. *Measurement*, 154, 107450.

[91] Alotaibi, B., Alotaibi, M. (2020). A stacked deep learning approach for IoT cyberattack detection. *Journal of Sensors*, 2020.

[92] Ashrafuzzaman, M., Chakhchoukh, Y., Jillepalli, A. A., Tosic, P. T., de Leon, D. C., Sheldon, F. T., Johnson, B. K. (2018). Detecting stealthy false data injection attacks in power grids using deep learning. In: *2018 14th International Wireless Communications Mobile Computing Conference (IWCMC)*, pp. 219–225.

[93] Khan, I. A., Pi, D., Khan, Z. U., Hussain, Y., Nawaz, A. (2019). HMLIDS: A

hybrid-multilevel anomaly prediction approach for intrusion detection in SCADA systems. *IEEE Access*, 7, 89507–89521.

[94] Qiu, W., Tang, Q., Wang, Y., Zhan, L., Liu, Y., Yao, W. (2020). Multi-view convolutional neural network for data spoofing cyber-attack detection in distribution synchrophasors. *IEEE Transactions on Smart Grid*, 11(4), 3457–3468.

[95] Zhe, W., Wei, C., Chunlin, L. (2020). DoS attack detection model of smart grid based on machine learning method. In *2020 IEEE International Conference on Power, Intelligent Computing and Systems (ICPICS)*, pp. 735–738.

[96] Yang, L., Zhai, Y., Li, Z. (2021). Deep learning for online AC False Data Injection Attack detection in smart grids: An approach using LSTM-Autoencoder. *Journal of Network and Computer Applications*, 193, 103178.

[97] Sayed, M. I., Sayem, I. M., Saha, S., Haque, A. (2022). A multi-Classifier for DDoS Attacks Using Stacking Ensemble Deep Neural Network. In *2022 International Wireless Communications and Mobile Computing (IWCMC)*, pp. 1125–1130.

[98] Hu, C., Yan, J., Wang, C. (2019). Robust feature extraction and ensemble classification against cyber-physical attacks in the smart grid. In *2019 IEEE Electrical Power and Energy Conference (EPEC)*, pp. 1–6.

[99] Huda, S., Yearwood, J., Hassan, M. M. Almogren, A. (2018). Securing the operations in SCADA-IoT platform based industrial control system using ensemble of deep belief networks. *Applied Soft Computing*, 71, 66–77.

[100] Qu, Z., Dong, Y., Qu, N., Li, H., Cui, M., Bo, X., Wu, Y., Mugemanyi, S. (2021). False data injection attack detection in power systems based on cyber-physical attack genes. *Frontiers in Energy Research*, 644489.

CURRICULUM VITAE

ORCID ID : 0000-0002-4468-4905

Name and Surname : Kübra BİTİRGEN

Foreign Language : English, Italian

Education

- Doctor of Philosophy
  Eskişehir Technical University, Eskişehir, Turkey
  Department of Electrical and Electronics Engineering
  07.2018 - 06.2024
- Master of Science
  Anadolu University, Eskişehir, Turkey
  Department of Electrical and Electronics Engineering
  07.2016 - 05.2018
- Bachelor of Science
  Anadolu University, Eskişehir, Turkey
  Department of Electrical and Electronics Engineering
  07.2011 - 05.2016

Professional Experience

- Department of Electronics & Communication Technology, Army NCO Vocational HE School, National Defence University
  08.2021 - ongoing

Research Projects

- "Total Consideration of Optimum Energy Management between Microgrids with Smart Home Energy Management and Testing Appropriate Energy Management Strategies Using Real Data" supported by the Scientific and Technological Research Council of Turkey (TÜBİTAK) under grant number 122E414, 01.11.2022 - ongoing (Scholar)

- "Energy Management for Smart Homes Using Mathematical and Heuristic Methods" supported by the Scientific Research Projects Commission of Eskişehir Technical University under grant number 20ADP091, 03.12.2020 - 03.12.2021 (Researcher)
- "Optimization of Smart Grid Energy Systems Based on Machine Learning Methods" supported by the Scientific Research Projects Commission of Eskişehir Technical University under grant number 20DRP192, 10.03.2022 - ongoing (Researcher)

Journal Papers

- Bitirgen, K., Filik, Ü. B. (2024). Differential evolution-based Markov Game for Cyber-Attack Detection. In preparation.
- Bitirgen, K., Filik, Ü. B. (2024). Markov game based on reinforcement learning solution against cyber-physical attacks in smart grid. Expert Systems with Applications, 124607.
- Bitirgen, K., Filik, Ü. B. (2023). A hybrid deep learning model for discrimination of physical disturbance and cyber-attack detection in smart grid. International Journal of Critical Infrastructure Protection, 40, 100582.
- Bitirgen, K., Filik, Ü. B. (2022). Performance Analysis of PCA Based Machine Learning Approaches on FDIA Detection. International Journal of Information Security Science, 11(4), 1-13.
- Bitirgen, K., Filik, Ü. B. (2021). A comprehensive study on modeling of photovoltaic arrays and calculation of photovoltaic potential using digital elevation model. Energy Sources, Part A: Recovery, Utilization, and Environmental Effects, 43(20), 2609-2633.
- Bitirgen, K., Filik, Ü. B. (2020). Electricity Price Forecasting based on XGBooST and ARIMA Algorithms. BSEU Journal of Engineering Research and Technology, 1(1), 7-13.

Conference Papers

- Bitirgen, K., Filik, Ü. B. (2023, November). Multi-defender strategic filtering

against Multi agent Cyber epidemics on multi-environment model for Smart grid protection. In 2023 International Conference on Energy and Green Computing, pp. 95.

- Bitirgen, K., Filik, Ü. B. (2017, November). Solar electricity potential based on arcgis maps and consumption of energy for engineering faculty buildings in anadolu university. In 2017 10th International Conference on Electrical and Electronics Engineering (ELECO), pp. 156-160.