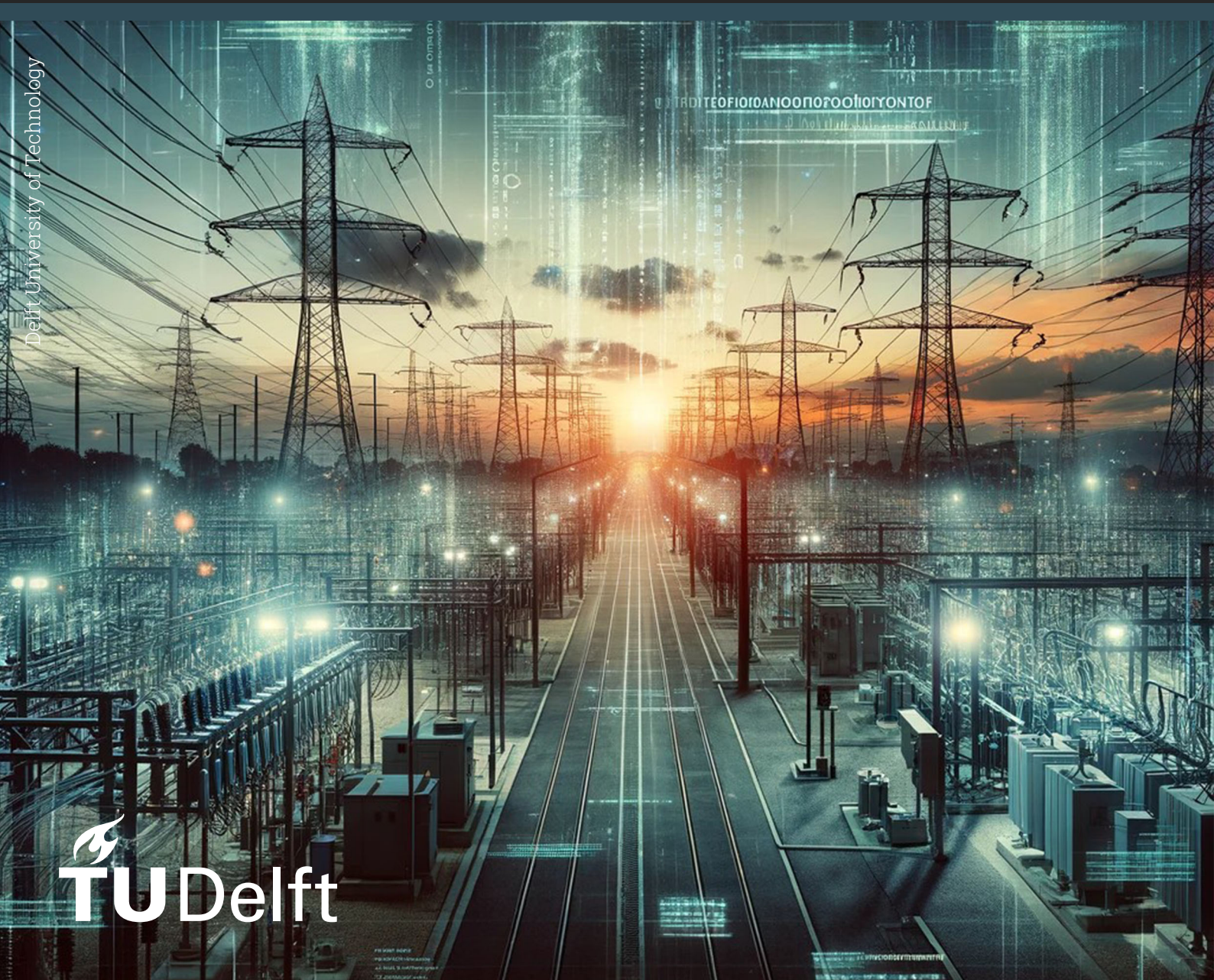
Crashing the smart grid

Modelling smart grid robustness against failures in an interdependent communication network

Master Thesis

Marijn Burgers



Delft University of Technology

TU Delft

Crashing the smart grid

Modelling smart grid robustness against failures in an interdependent communication network

by

Marijn Burgers

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science
Complex Systems Engineering & Management
Delft University of Technology

Chairman & First Supervisor:	Prof.dr. M.E. (Martijn) Warnier
Second Supervisor:	Dr. Y. (Yury) Zhauniarovich
Master Thesis defense date:	12th of July 2024
Faculty:	Technology, Policy & Management
Student number:	4832132

Disclaimer:	This thesis acknowledges the use of ChatGPT for language editing and coding assistance purposes. All outputs generated were reviewed and verified by the author, who is responsible for any errors or inaccuracies.
Cover:	Initially generated by ChatGPT (Modified)
GitHub:	`https://github.com/marijnburgers4/smart-grid-model-cascading-failure`

TU Delft

Summary

Sustainability and the reduction of emissions are among the most critical issues today. The energy sector is a major contributor to global greenhouse gas emissions. The U.S. and China's power systems are responsible for 40% and 48% of their countries' CO_2 emissions, respectively. Despite the devastating consequences these emissions are causing, a sharp rise in global electricity demand is forecasted, with growth rates averaging 3.4% per annum over the next three years. The smart grid offers a solution by incorporating distributed energy sources and using communication technologies to monitor the power grid and manage the transport of electricity, improving efficiency in energy usage and reducing emissions. In China, smart grid technologies could potentially reduce CO_2 emissions by up to 27.5%.

Although the incorporation of communication technology in power grids can lead to reductions in emissions, it also introduces cybersecurity vulnerabilities and increases the attack surface for cybercriminal organisations. This can lead to disastrous incidents, such as the cyberattack on a Ukrainian power station in 2015 during the ongoing Russo-Ukrainian war, which led to an outage affecting almost 1.4 million people. The increasing interconnection between the power grid and communication technologies can have a severe impact, especially with the rise of cyberattacks. Despite the seriousness of the situation, there is a lack of long-term strategic planning in this area.

Given these concerns, insight is needed on the impact of failures in the communication network on the power grid it is coupled with. Therefore, the main research question is formulated as follows:

What is the effect of failures in the communication network on the underlying power grid?

To answer this question, a modelling approach is used. A communication network is generated using Python and network theory. This communication network is coupled with a one-to-one interdependency with a power grid. To give the power grid electrical characteristics, we use Pandapower's IEEE 118-bus test system. A failure of a communication network component leads to a failure of the power grid component and vice versa. We simulate failure scenarios by increasingly failing communication network nodes based on random selection and targeted attacks (degree, betweenness, and closeness centrality). This is applied to two types of communication networks: a double-star and a mesh network. Additionally, we incorporate two different model behaviours: one that includes failure propagation in the communication network (using a simple diffusion model to simulate, for example, virus spread) and one that does not.

The results of this research indicate that when comparing both communication network structures, a mesh communication network is recommended over a double-star network. A power grid coupled with a mesh communication network shows a higher robustness under majority of targeted attacks, both with and without failure propagation behaviour, and shows similar behaviour under random attacks.

For both network types, it is important to limit failure propagation in the communication network, as it significantly impacts the robustness of the smart grid. To reduce this failure propagation, various measures can be implemented, such as network segmentation using firewalls or virtual networks, employing anomaly detection systems to identify and respond to suspicious activities, utilising redundancy techniques, using high-quality components, conducting proactive maintenance, and implementing real-time monitoring to assess and detect potential failures.

When considering possible failure propagation in the communication network, it is recommended to prioritise components with high degree or betweenness centrality in mesh networks, as the failures of these components have the most impact on the robustness of the smart grid. Degree centrality has a greater impact with a lower percentage of initial failed nodes, while betweenness centrality has a more significant impact with a higher number of initial failed nodes.

For double-star networks, when considering failure propagation, targeted attacks have a similar impact on the robustness of the smart grid. However, degree and betweenness centrality attacks show a slightly higher impact than closeness centrality. Therefore, it is recommended to prioritise components with high degree and betweenness centrality values.

When the possibility of failure propagation is limited, it is recommended to prioritise communication network components with high closeness centrality. These components have the second largest impact with a lower number of initial failed nodes (after degree) and the most impact on the smart grid's robustness with a higher number of initial failed nodes.

For a power grid coupled with a double-star communication network, it is recommended to prioritise communication network components with high betweenness centrality. Although degree centrality shows a slightly higher impact with a lower number of initial failed nodes, beyond a certain threshold, betweenness centrality has a greater impact than degree centrality.

Even though small-world networks show overall higher robustness, certain trade-offs need to be considered between the two types of communication network structures. These trade-offs include scalability issues, energy usage, data transmission efficiency, operational costs, and the complexity of ensuring network segmentation, which mostly favour scale-free communication networks.

Given these results trade-offs, this research highlights the need for a collaborative approach between power grid engineers and communication network experts to enhance the robustness of the smart grid against cyber threats. Understanding the effects of the communication network on the underlying interdependent power grid is crucial for developing long-term strategies to protect the smart grid from failures and increasing cyber attacks.

Building on this understanding, future research directions should focus on enhancing the realism of the communication network by adding additional network layers, incorporating data transmission functionalities to simulate delays or inaccuracies, adjusting certain parameters, and incorporating a more complex communication failure propagation model to better reflect real-world conditions. Furthermore, stochastic elements can be incorporated in the communication network to capture the uncertainty within the failure process, and more accurate interdependency ratios could be used. Additionally, the communication network structure can be improved by aligning the geographical positioning of the nodes with the transmission lines they are interdependent on. More research can be conducted on different communication network structures and attack strategies. Lastly, incorporating economic parameters to understand the costs associated with increasing the robustness of the smart grid could be explored.

Contents

1 Introduction

1.1. Context

In today's world, the pressing issue of sustainability and the urgent need to reduce our emissions loom larger than ever. Importantly, the energy sector is at the heart of this environmental puzzle, accounting for around 75% of global greenhouse gas emissions, and is, therefore, a key driver of climate change [1]. The U.S. power system alone contributes up to 40% of the country's carbon dioxide emissions (CO_2), damaging the environment [2]. China's power sector is responsible for a staggering 48% of the CO_2 emissions in China [3]. Despite the devastating consequences these emissions are causing, a sharp rise in global electricity demand is forecasted, with growth rates averaging 3.4% per annum over the next three years [4]. The increase in electricity consumption comes from various sectors and technologies, including artificial intelligence, data centers, and cryptocurrency. These particular areas are expected to see their electricity consumption potentially double by 2026 compared to 2022 [4]. Due to these new emerging technologies, the increase in demand surpasses the capability of the traditional power supply system to provide services up to the required standard [2].

The Smart Grid (SG) is a promising solution to enhance energy and environmental sustainability by integrating distributed energy sources, this includes renewable and non-renewable energy sources [5]. The SG uses digital and other advanced (communication) technologies to monitor and manage the transport of electricity from all generation sources to meet the varying electricity demands of end users [6]. The integration of information and communication technologies allows both utilities and customers to monitor, predict, and efficiently manage energy usage with its bi-directional communication and electricity flow capabilities [5]. In an optimistic scenario, the implementation of SG technologies in China could lead to a reduction of carbon emissions by as much as 27.51% [7].

1.2. Problem

The increase in the usage and implementation of digital technology in the energy industry gives rise to cybersecurity vulnerabilities. These vulnerabilities stem from digital systems, telecommunication equipment, and sensors throughout the grid, as each component increases the attack surface for cybercriminal organisations [8]. According to The International Energy Agency [8], cyber-attacks are on the rise in the electricity sector. There is growing evidence that cyber-attacks on utilities have escalated rapidly since 2018, hitting concerningly high levels in 2022 following Russia's invasion of Ukraine.

Examples of the potential consequences of cybersecurity breaches in power grids is the BlackEnergy Trojan horse malware incident. On December 25, 2015, a cyber-attack targeted a Ukrainian power station during the ongoing Russo-Ukrainian War, causing an outage for 230,000 citizens [9] and affecting approximately 1.4 million people.

Another example of a cyber-attack on power utilities happened in late 2022. Google's Mandiant cybersecurity service addressed a cyber-physical attack by the Russia-linked SANDWORM group on Ukrainian critical infrastructure, employing a new method to disrupt industrial control systems (ICS). The attacker used various techniques to likely trip the victim's substation circuit breakers, causing

an unplanned power outage that coincided with mass missile strikes on critical infrastructure across Ukraine [10].

As a last example, in December 2016 a cyber-attack was performed by ELECTRUM [11], a group directly associated with SANDWORM, directed on the ICS of a power substation located near the Ukrainian capital Kyiv [12, 13]. This resulted in a blackout striking a section of Kyiv and its surroundings, leaving it without power for over an hour. The outage cut off an estimated one-fifth of Kyiv's electricity consumption [13].

These examples demonstrate the severe impact cyber-attacks can have on the power grid, caused by the growing interconnection between the critical power grid and digital technologies. As demonstrated by the examples, this interconnection exhibits vulnerabilities that are increasingly being exposed, one contributing factor being the changing geopolitical landscape. While electric power utilities worldwide already allocate significant budgets to cybersecurity, averaging 8% of total IT budgets in the United States and Canada, job posting data from leading power utilities in the United States indicates that cyber-attack incidents cause abrupt rises in demand for cybersecurity professionals, hinting towards a lack of long-term strategy or planning in the past [8].

1.3. Research objective

The problem lies therefore in the exposed vulnerabilities of SGs due to the increasing interconnection and dependence on communication technologies, which elevates the threat of cybersecurity risks. If not addressed, this could impact thousands, if not millions, of individuals depending on the availability of the power grid, especially given the increasing tensions between nations.

For this reason, the objective of this research is to study the interdependent effect on the underlying power grid when failures or attacks occur in the communication network. By exploring this, we gain valuable insights into the robustness of the SG.

2 Literature

In this chapter, we provide background information from the literature on what an SG is, its components, and its relationship to cyber threats. In our research, we assume the occurrence of successful cyber-attacks or device failures is a given. The description of various cyber-attacks is included solely to provide context for this research. Furthermore, we conduct a literature review on related work, which brings forward a significant knowledge gap.

2.1. Background information on smart grids

2.1.1. Definition of smart grids

To set a scope of what an SG is, we formulate a definition. Currently, there is no standardised definition of an SG. The term SG has been used widely with different definitions and meanings [14]. Several definitions from literature and acknowledged institutions can be found in [14, 15, 16, 17, 18]. We adopt the definition set by the International Energy Agency [6], with a slight modification, as we believe it incorporates the essential elements for this research:

"A smart grid is an electricity network intertwined with digital and other advanced technologies, such as sensors and communication devices, to monitor and manage the transport of electricity from all generation sources to meet the varying electricity demands of end users. Smart grids co-ordinate the needs and capabilities of all generators, grid operators, end users, and electricity market stakeholders to operate all parts of the system as efficiently as possible, minimising costs and environmental impacts while maximising system reliability, resilience, flexibility and stability"

The key takeaway from the definition is that an SG is essentially made up of two interconnected networks: a power grid and a communication network. With our definition being set, we can now delve further into the different domains and components that constitute an SG.

2.1.2. Smart grid components

By only having a descriptive definition of an SG it does not become apparent how an SG is structured and how it operates. To make this more clear we provide the conceptual description of the National Institute of Standards and Technology (NIST) from [19] which is also often described in academic literature [20, 21, 22]

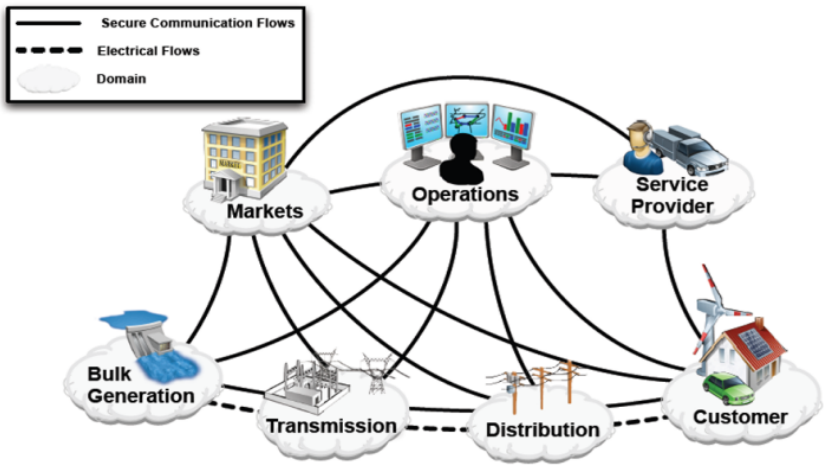


Figure 2.1: NIST conceptual model of the smart grid [19].

From Fig. 2.1, it is evident that the SG consists of seven domains, which we describe using information obtained from [19, 23, 24], including the relationships between the domains. It should be noted that this research is not particularly focused on the markets and service provider domain.

The bulk generation domain consists of generators producing electricity in bulk quantities. It is the first step in the process of electricity delivery to the end user. Electricity is generated from various sources such as oil, coal, flowing water, solar radiation and nuclear fission. The bulk generation domain is electrically linked to the transmission domain and communicates with the market domain over the Internet through a market services interface and with the operations domain over the wide area network.

The transmission domain carries the bulk electricity produced by the generators over long distances to the distribution domain via substations and transmission lines. Within this domain, electricity might also be stored and generated. Furthermore, the transmission network is monitored and controlled via a supervisory control and data acquisition (SCADA) system. This system comprises a communication network along with devices for control and monitoring.

Within the distribution domain, electricity is distributed to and from end users using the electrical and communication infrastructures that connect the transmission and customer domains. This domain includes distribution feeders and transformers that supply electricity. It interacts with various types of equipment, including distributed energy resources (DERs), plug-in electric vehicles, advanced metering infrastructure (AMI), and sensors equipped with communication capabilities.

The customer domain consists of the end user. These customers can be categorized into three types: residential, commercial/building, and industrial. Besides consuming electricity, the end user may also generate, store, and manage the use of energy. This domain, electrically connected to the distribution domain, communicates with the distribution, operation, service provider, and market domains.

Within the market domain, actors include the operators and participants in electricity markets. The balance between electrical supply and demand is maintained in this domain. To align production with demand, the market domain communicates with energy supply domains, including the bulk generation domain DERs.

The service provider domain consists of organisations providing services to both electrical customers and utilities. These organisations oversee services including billing, customer accounts, and energy usage. The service provider interacts with the operation domain for situational awareness and system control, while also communicating with the market and customer domains to develop smart services that enable customer interaction with the market and home energy generation.

Lastly, the operation domain's actors are responsible for managing the movement of electricity. This domain maintains efficient and optimal operations in both transmission and distribution. It utilizes energy management systems for transmission and distribution management systems for distribution. Furthermore, this domain uses field area networks (FANs) and wide area networks (WANs) within the transmission and distribution domains to gather information on power system activities such as monitoring, control, fault management, maintenance, analysis, and metering. This information is obtained through the use of SCADA systems.

2.1.3. The two interconnected networks of a smart grid

We elaborate further on how the power grid and communication network are intertwined to broaden the understanding of the SG. This is done by examining Fig. 2.2, which provides a more detailed description than Fig. 2.1. Not all components are discussed; only the parts relevant to this research are covered.

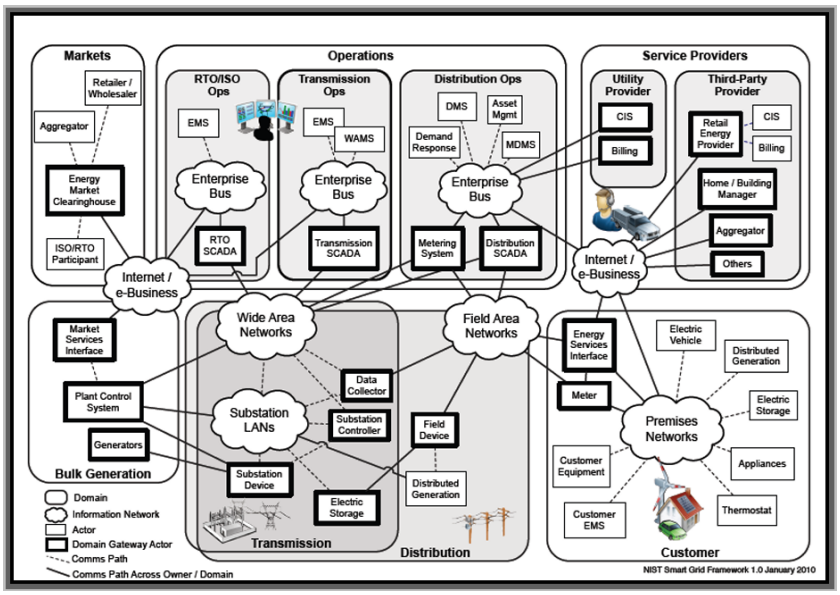


Figure 2.2: NIST conceptual model of the smart grid, including communication network components [19].

The power grid

The generation is the first step in the power grid. As explained in §2.1.2, generation is produced from various sources. The generated electricity is transported over high-voltage (HV) transmission lines. HV transmission lines are more suitable for transporting electricity than low-voltage since energy losses are lower due to conductor resistance [25].

Key components in the power grid are the substations. One of their main roles is the conversion of electricity into different levels of voltage [26]. This conversation of voltages, within the substation's site is done by special pieces of equipment called transformers. These substations are located at points where electricity enters the transmission network, accommodating the varying voltage outputs from different generation sources, and where it exits the transmission network to be distributed to homes and businesses at a lower voltage. [26]. Transmission substations can be seen as junctions, connecting circuits to form a network through which HV electricity flows [26].

In the distribution sector, the HV electricity carried by transmission lines is reduced (at the substation's site) because the electricity from these lines is too high to be delivered directly to consumers [27]. Therefore, the electricity, from the distribution to the consumer domain, is transported at a voltage that is appropriate for residential and corporate use [27].

An important feature in both the transmission and distribution substations are circuit breakers and relays. A circuit breaker, an electrical switch, is designed to safeguard electrical circuits from damage due to overcurrent, overload, or short circuits. Its primary role is to interrupt the flow of current when a fault is detected by protective relays [28]. In the power grid, circuit breakers are used to open and close transmission lines and transformers [29]. A relay is an electrical switch that operates in low-current circuits to detect and control contacts electromagnetically or electrically. Its main function is to identify faults and act as a protective device, signaling the circuit breaker to either make or break the circuit upon fault detection [30].

The communication network

The communication network is compromised of information flow components which can be seen in Fig. 2.2. Different domains in the power grid are connected via communication technology. For this research, we mainly focus on the transmission, distribution and operation domain (specifically the SCADA

system). The power grid is monitored to ensure that the power quality is maintained throughout the grid [31]. The monitoring of the grid is executed by the placement of smart sensors through the power grid, AMI (this is more focused on the end user and utility companies), and the integration of SCADA [32, 31]. These sensors (”field device” and ”substation device” in Fig. 2.2) collect various data to monitor the grid. The field devices transmit the collected data to data concentrators [33]. An example of this is transmission line monitoring where wireless smart sensors are placed along the transmission lines [31]. These sensors collect transmission line data, exchange it with neighboring nodes, and eventually forward it to a central collection site (i.e. data concentrators) [31]. Besides transmission lines, the substation includes devices that regulate and distribute electrical energy, such as a remote terminal unit (RTU), global positioning system, human-machine interface, and intelligent electronic devices [23]. Both transmission line measurements, collected by the data concentrators, and the operational data from the substation are sent via LANs and eventually WANs to the operational domain. The information received in the operational domain is received by a SCADA system. Here, the data is processed for various purposes, including the protection of transmission lines, which is the primary focus of this research. The SCADA system uses data gathered by sensors across the grid to determine whether the relays and circuit breakers, discussed in the previous section, should be opened or closed to safeguard the transmission lines. This decision-making data is then sent to field devices near the transmission lines or to the RTU at the substation level, where the instructions to open or close the circuit breakers are executed. This data is transmitted back through the WANs and LANs. This information flow structure forms the backbone of the power grid’s communication network.

2.2. Background information on cyber-attacks

2.2.1. Definitions and principles of cyber-attacks and cyber security

To understand how SGs are susceptible to cyber-attacks, we need to understand the concept of cyber-attacks. Various definitions of a cyber-attack can be found in [34, 35, 36]. Within the scope of cyber-attacks on SGs, we find IBM’s definition [36] the most applicable since they not only mention an attack on the cyberinfrastructure but also ”other assets” which in this case is the underlying power grid. Therefore our definition adopted from IBM, changed slightly, is the following:

”A cyber-attack is any intentional effort to steal, expose, alter, disable, or destroy data, applications, or other assets, such as critical physical infrastructures, through unauthorized access to a network, computer system or digital device”

In the context of cyber security, the CIA-triad has been widely used in the information security practice and academic literature [37]. The term stands for ”confidentiality”, ”integrity” and ”availability”. In [38], confidentiality refers to the protection of information from unauthorized access. Integrity means data is complete, trustworthy, and has not been altered by an unauthorized user or accidentally. Availability refers to data being accessible when you need it. This information security model can also be applied to SGs. The definition of the CIA-triad, applied to SG, is taken from [39], [40], [21] and presented in order of importance according to [39]

- Availability: *Ensuring timely access to information is crucial in the SG. A loss of availability could disrupt power delivery by denying access to authorized individuals. Attacks targeting system availability are categorized as DoS attacks, intending to disrupt data transfer and render resources unavailable.*
- Integrity: *Safeguarding against unauthorized modifications of information or the system by illegitimate users is essential. The compromise of integrity in the SG could lead to alterations in sensor values and product recipes, thereby impacting power management.*
- Confidentiality: *Preventing unauthorized access to information is vital for protecting personal privacy and safety. SG networks handle information with varying privacy and sensitivity levels, ranging from consumption data to consumer private information.*

In this research, we mainly focus on availability and integrity, as most cyber-attacks undermine these principles. Although availability can be regarded as more important than integrity and confidentiality, as the latter two are directly dependent upon availability [41].

2.2.2. Types and methods of cyber-attacks

To better understand the vulnerabilities of SGs, we discuss the types and methods of cyber-attacks in the context of SG and how these attacks can be classified. Yet, our research does not go into the specifics of executing particular cyber-attacks on an SG. Within our study, we accept the occurrence of a successful cyber-attack or device failure as given. The description of various cyber-attacks serves solely to provide context for this research. The majority of attacks typically involve one or a combination of four main attack types [22, 42, 43]: device attack, data attack, network availability attack and privacy attack. Each attack type is elaborated on down below.

Device attack

A device attack aims to compromise and control grid network devices, often serving as the start phase in a larger attack. Through a single compromised device, further attacks can be launched, spreading across the SG network. As an example, a virus disguised as genuine data might be transmitted by a compromised sensor, infecting other parts of the network. Since the SG is connected to a vast number of (IoT) devices it makes the networks particularly susceptible to Trojan horse attacks (a type of malware that disguises itself as legitimate code or software [44])

Data attack

A data attack seeks to unlawfully insert, modify, or delete data or control commands within communication network traffic to deceive the SG into making incorrect decisions or actions. One frequently encountered form of data attack involves customers tampering with smart meters to lower their recorded consumption data, which results in a reduction of their electricity bills [22]

Network availability attack

A network availability attack mainly takes place in the form of denial-of-service (DoS) attack [22]. This type of attack aims to overwhelm the SG network's computational and communication resources, causing communication delays or failures. As an example, an attacker might flood a processing center with false information that it spends the majority of its time verifying the authenticity of the information, neglecting legitimate network traffic. Given the time-sensitive nature of SG communication, even slight delays can lead to outages in the network [22]. Effectively managing these attacks is crucial, as they can render millions of devices connected to the SG offline, crippling the entire system [22].

Privacy attack

A privacy attack undermines the confidentiality principle in the CIA-triad (which is not in the scope of this research). A privacy attack on an SG aims to uncover users' personal data, like electricity usage patterns or credit card information, potentially leading to physical attacks like burglary (as the behaviour of consumers can be derived from their electricity consumption data) [22]. Protecting user privacy is essential in preventing identity theft and ensuring confidentiality.

To better understand these types of attacks, we discuss various mechanisms that carry out such attacks [39, 40, 45, 22].

- *Hacking* involves obtaining the password of a system's platform to gain access to the system. It can take several forms; for instance, social engineering, man-in-the-middle attack (MITM) or malware injection (the last two forms are explained down below).
- *Malware injection*, involves installing harmful software like viruses, spyware, ransomware, trojans, or worms into cyberspace, with the intent to cause damage or disable computers and networks.
- *MITM* attack aims to intercept communications between devices by eavesdropping. Users on both ends believe they are communicating directly, but the adversary can monitor and even modify the communication. This is worse in situations where the encryption of information is absent.
- *Phishing* is a request for data from a source that appears trustworthy. The goal is to trick users into performing actions like clicking on malicious links or providing sensitive information.
- *DoS or DDoS* involves flooding a system's network or devices with a large amount of traffic and spam data, aiming to overload it and make it unresponsive or slow because of the excessive amount of requests [46]. A distributed denial-of-service (DDoS) attack is a DoS attack that uses multiple machines or computers to flood the targeted system [47]. In [48], the authors state that

almost all research studies show DoS attacks would be a major issue for SGs. The authors in [49] state that many proof-based DoS defense techniques may not work appropriately due to resource limitations and real-time requirements of SCADA networks.

- *SQL injections* involves inserting a SQL query into input data from the client to the application. If successful, it can read sensitive data from the database, modify database information, execute administrative operations (like shutting down the database), retrieve file content from the DBMS file system, and, in some cases, issue commands to the operating system [50].
- *An Advanced Persistent Threat (APT)* is a stealthy cyber-attack in which an individual or group gains unauthorized access to a network, and remains undetected for a period of time [51]. Usually, data theft is the goal of an ATP. These attacks involve advanced and sophisticated processes that demand a high level of stealthiness over an extended period, often targeting specific organisations. ATPs are commonly sponsored by nations or very large organisations.

2.3. Related work and knowledge gap identification

2.3.1. The process of the literature selection

We conduct the literature review by extracting relevant papers from Google Scholar. In Fig. 2.3, the literature review process is illustrated.

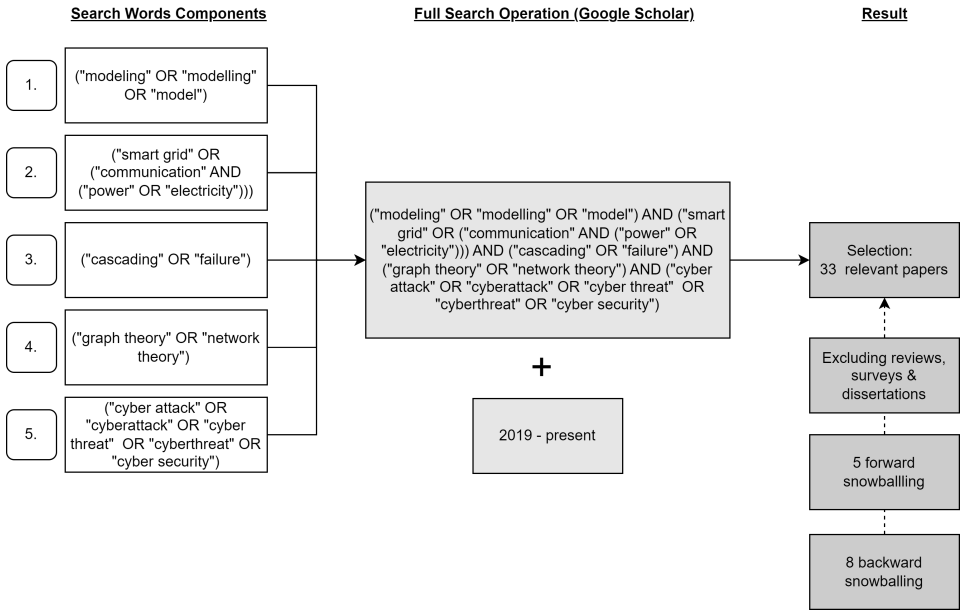


Figure 2.3: Literature selection process using Google Scholar.

On the left-hand side of Fig. 2.3 we can see the different components on which our full search operation is based. The first component is used to obtain papers that include a model of an SG (both with the British and American spelling). The second component is used to extract papers that research SGs or, formulated differently, a communication network and a power grid. The third component is included to get papers that research cascading failures or general failures in a SG. As our research is based on graph theory, the study of networks, we included this search term as well. Lastly, we include words related to cyber incidents, such as attacks and threats, as these are the primary reasons for conducting this research.

These components are consolidated into one full search operation. We only select papers that were published in the last five years. Therefore, the period starts from 2019 until now. We exclude review papers and surveys because they are difficult to compare with papers that focus solely on modelling failures in an SG. Additionally, we exclude dissertations. We only include available papers (i.e. we did not purchase any papers). After selecting 20 relevant papers, we use forward and backward snowballing techniques to identify additional relevant papers, resulting in 5 more papers through forward

snowballing and 8 more papers through backward snowballing. Some of these additional papers are older than five years.

2.3.2. Identified knowledge gap based on the selected literature

After the literature selection, we conduct an analysis of the papers. During the analysis, we take into account 12 different aspects. The reason we choose these aspects is to clearly demonstrate the differences between elements in the literature and to highlight how our research contributes to it. Furthermore, we believe that these aspects enable us to thoroughly research the effect of failures in the communication network on the underlying power grid. These aspects are described in Table 2.1. The first aspect (A) examines if the authors use an interdependent SG model, also referred to as a cyber-physical model. This model indicates that the power grid has a certain dependency on the communication network and vice versa. This also means that the authors model a communication network and a power grid, which are the two networks that make up an SG. The second aspect (B) checks whether the authors assign different roles to components in the power grid (e.g., generation, consumption, and transformers). Aspect C is similar to the immediately preceding aspect as it evaluates whether the components in the communication network are heterogeneous (e.g., sensors and SCADA system). The fourth aspect (D) assesses if the power grid used in the authors' model represents a 'real-world' power grid. These include, for example, test systems from recognised institutions. The fifth aspect (E) investigates if the authors model a failure or attack scenario in their research. Aspect F analyses if the authors include different types of topologies for the communication network in their SG model. Aspect G assesses if failures or attack scenarios in the communication network are based on different types of network metrics. Aspect H investigates whether the failures or attack scenarios are initially targeted at the cyberinfrastructure (i.e., communication network). Aspect I investigates whether the authors fail or attack different numbers of communication network components. This is included as a criterion to examine the robustness of an SG's communication network. Aspects J and K deal with the direction of the conducted research. Aspect J investigates whether the authors examine which network-based attack (or failure) strategy is most effective on communication networks. Aspect K analyses which network topology for the communication network is most robust. The last aspect (L) is included because several papers within the selected literature simulate failure propagation in the communication network, for example, using a diffusion model or other epidemic spread models.

Table 2.1: Meaning of the letters in the literature analyses.

Letter	Definition
A.	Uses an Interdependent cyber-physical model
B.	Heterogeneous power grid components
C.	Heterogeneous communication network components
D.	Uses a realistic power grid model
E.	Models an attack or failure situation
F.	Uses different types of communication network structure
G.	Uses different network-based attack strategy on only the communication network
H.	Initial failures occur only in the communication network
I.	Different amounts of communication components are attacked or failed
J.	Analyses the most effective network-based attack strategy on only the communication network
K.	Analyses the robustness of different communication network structures
L.	Simulates communication network failure propagation (e.g. using diffusion or other epidemic spread models)

With the formulated aspects, we analyse the papers which are illustrated in Table 2.2. On the left-hand side, the references of the papers are displayed in four different colours. Light grey represents the selected literature. The medium grey shade indicates papers retrieved from forward snowballing, while the darkest grey represents papers obtained from backward snowballing. The blue colour indicates our own research. Each column in Table 2.2 represents an aspect from Table 2.1. The column letter corresponds to the definition letter from the table. Within Table 2.2, we use three different colours (green, yellow, and red) to indicate how well the aspects match those we previously formulated. The green

colour indicates that the paper fully matches that particular aspect of the corresponding column. As an example, the first paper has a green cell with aspect A. This means that in the paper the authors use an interdependent model of a power grid and communication network. The yellow colour represents that the aspect is present in the paper to a certain degree (i.e., it does not fully match the criteria). For example, the first paper has a yellow colour for aspect G. This is because the authors do not perform a network-based attack solely on the communication network, but rather on both the power grid and communication network simultaneously. The red colour indicates that the paper does not match a certain aspect. As an example, the last paper does not perform an analysis of the robustness of different communication networks (aspect K) and is therefore marked in red.

Table 2.2: Analyses of the selected papers from the literature research.

Literature	A	B	C	D	E	F	G	H	I	J	K	L
Atat et al. [52]												
Chen et al. [53]												
Guo et al. [54]												
Salehpour et al. [55]												
Wang et al. [56]												
Wang et al. [57]												
Pan et al. [58]												
Atat, Ismail, and Serpedin [59]												
Zhang et al. [60]												
Wu, Li, and Li [61]												
Chen et al. [62]												
Gao, Li, and Yang [63]												
Li et al. [64]												
Zhang et al. [65]												
Lee and Hu [66]												
Alonso et al. [67]												
Kang et al. [68]												
Shen, Gao, and Peng [69]												
Zhu, Milanovic, and Mihic [70]												
Ding et al. [71]												
Jiang et al. [72]												
Rajkumar et al. [73]												
Zhang et al. [74]												
Chen et al. [75]												
Xu et al. [76]												
Buldyrev et al. [77]												
Guo et al. [78]												
Cai et al. [79]												
Cai et al. [80]												
Zhang et al. [81]												
Zhang and Yagan [82]												
Chen et al. [83]												
Sturaro et al. [84]												
Our study												

We provide a brief description of each piece of literature and highlight how our research differs from it. Note that some authors use the term scale-free while others use double-star, and some use mesh while others use small-world to describe (communication) network structures. In this research, we use mesh and double-star to describe the network structures.

In [52], the authors develop an interdependent SG model where joint cascading failure simulations and different attack strategies are performed and compared. They show that the proposed joint attack achieves comparable damage to attacking the most influential power nodes or communication nodes solely. However, The authors use homogeneous communication network components, do not incor-

porate different communication network structures, and do not show which attack strategy is the most effective in regard to the robustness of an SG.

The authors in [53] present a partial random coupling SG model. The authors show, through their simulation results, that coupling strength and overload coefficient are positively correlated with the connectivity of systems. The authors use different communication networks (mesh and double-star) and explore different attack strategies including random, degree and betweenness centrality. The authors, however, use homogeneous power and communication network components and assume that the power grid has abundantly distributed generators. Additionally, this research concentrates more on the connectivity of the coupled power grid and communication network rather than on the robustness and performance in terms of the functionality of the power grid

A stochastic cascading failure SG model is presented in [54]. The authors provide information on the robustness of a power grid given different communication topologies (rewired, random, mesh and double-star) and interdependencies. They demonstrate that, under certain interdependencies, a mesh communication topology is more robust against SG cascading failure. However, in this research, failures are not solely initiated in the communication network, nor do the authors analyse the most effective attack strategy.

The cyber-attack failure propagation model proposed by [55] in smart grids assigns heterogeneous roles to the components in both the power grid and communication network and defines rules for the interdependency connection. They show that their model accurately identifies system failures (compared to a small-cluster model). Additionally, the authors conclude that intra-degree attacks are more impactful than other attack scenarios. This study does not include different communication network structures, and the authors do not focus on initial failures exclusively within the communication network.

For [56], the authors investigate cascading failures in an interdependent SG by using a network-based virus propagation model in both a double-star and mesh communication network structure. Their research shows that a double-star communication network is more vulnerable to virus propagation in SGs. This research mainly focuses on virus propagation behaviour and does not address the differences in robustness with or without communication network failure propagation. Additionally, the authors limit the choice of network-based infections to selecting only random vertices and those based on degree centrality.

The authors in [57] present an information flow model in a coupled power grid and communication network. The authors apply the information flow model to compare the transmission performances of six network topologies, describing that a communication network using a Markov clustering algorithm has the best transmission ability. The authors do not incorporate the interdependent behaviour between the communication network and the power grid. Additionally, they do not clearly model failure or attack scenarios, particularly those based on network metrics. Therefore, they do not investigate the most effective attack strategy, nor do they clearly state which communication network structure shows the highest robustness concerning the underlying power grid.

In [58], the authors model and analyse an SG with random power grid line failures under different coupling strategies and a double-star communication network. Besides constructing metrics to measure the vulnerability of nodes, they show that a TS-GM positive sequence coupling shows the best robustness. The authors in this research do not consider different communication network structures, focus only on initial failures in the power grid, and do not incorporate other attack strategies.

For [59], the authors limit the joint cascading failure propagation in the SG by formulating the optimisation problem of partitioning the interdependent SG. They show that overall damage can be reduced by 62% using the higher-order partitioning method. This research focuses on reducing damage through partitioning rather than on the effectiveness of different attack strategies on the communication network. Additionally, the authors only use one type of communication network (a double-star network).

A modelling framework for studying cascading failures and the robustness of an SG is proposed by [60]. They demonstrate that errors in the communication network can act as catalysts for cascading failure in real grids and that severe cyber-attacks can lead to disastrous consequences. The authors compare the robustness of a power grid with and without a coupled cyber grid. They do perform network-based attacks (degree, betweenness, and capacity-based), but these are initiated in the power grid and not in the communication network. Furthermore, the authors do not test different communication network structures and their robustness.

An approach for modelling cascading failures in SGs is proposed in [61], taking nine component state failures into account. The authors test their model using a mesh communication network, showing

that the cascading failures in actual SGs can be well explained and simulated. However, even though the authors focus on the robustness of a coupled communication network and power grid, they do not simulate failures in the communication network using different attack strategies. Moreover, they use only one communication network structure.

The study by [62] proposes the partial coupling systems model to represent a coupled power grid and communication network. In their simulation, they analyse the impact of different attack scenarios (random, high degree/betweenness) and topology structure on the robustness of the system. They show that there is a positive correlation between the clustering coefficient and robustness. Additionally, the authors conclude that, under random attacks, the double-star network is more robust than a mesh network. Although the authors use different communication networks and perform network-based attacks, they conduct these failures within the power grid. In contrast, our focus is on failures within the communication network.

The authors of [63] present a cascading failure model of a cyber-physical system in a virtual power grid. They perform random attacks and targeted attacks based on degree, betweenness, eigenvector, and information centrality. They demonstrate that a mesh relationship (for both the power grid and communication network) is the most robust against cascading failure (with certain interdependency coupling strategies). However, the authors do not incorporate electrical properties in their power grid model and do not compare different network-based attacks based on their impact on the robustness of the cyber-physical system. Furthermore, this study does not solely focus on failures in the communication network and their impact but rather on the overall robustness of the system given different structures and coupling strategies.

The dual hidden cascading failure model of an SG proposed by [64] does not consider hidden failures in the power grid but also in the communication network. They demonstrate that if the hidden failure rate is lower than the failure threshold, the double-star topology communication network coupled mode is the most robust. Otherwise, the mesh topology communication network coupled mode is the most robust. Although the authors provide insights into the robustness of different communication networks, the failures are initiated by removing a transmission line, thus focusing on initial failures in the power grid. Furthermore, the authors do not provide insights into which attack type is the most effective.

A cascading failure model of a connected power grid and communication network that considers the operational characteristics of the communication layer (e.g. transmission delay) is presented in [65]. Although the authors consider different communication network topologies as well as power grid topologies, they do not analyse the robustness of these different topologies given failures in the communication layer.

In [66], the authors propose a framework to model the SG as an interdependent complex network and research the vulnerabilities in topology subject to attacks. This research focuses on identifying the importance of the nodes in the interdependent SG model rather than the robustness of the SG.

Similarly, the authors in [67] model an interdependent SG and determine the criticality of nodes based on centrality indexes (degree, betweenness, closeness, and eigenvector centrality). However, they do not focus on the robustness of the SG given different topologies or distinguish between the impact of the various centrality indexes.

In [68], the authors model different attack scenarios to analyse the robustness of an interdependent SG. However, the focus lies more on attack scenarios where only cyber nodes, power nodes, or both fail simultaneously rather than on the impact of attacks given different communication network topologies.

The authors in [69] propose a stochastic cascading failure model in an SG considering malware attacks. They incorporate a diffusion process to model the spread of malware among cyber nodes. The coupling between the power grid and communication network worsens the severity of cascading failure in the power grid. The authors only consider one type of communication network (double-star) and do not perform any attack scenarios based on network metrics (e.g., nodes with the highest degrees are initially infected).

The authors of [70] identify system component criticalities in an interconnected power grid and communication network. They include measures such as node degree, importance, betweenness, closeness, and eigenvector centrality. They suggest that betweenness and degree centralities are the most suitable measures for identifying critical buses in power systems and nodes in the cyber system. The authors elaborate briefly on the robustness of star and mesh communication network structures, with the latter being more robust against intentional cyber-attacks. However, they do not explicitly demon-

strate the impact of each centrality measure attack on solely the communication network, considering the different network structures.

The cascading failure model of a coupled power and cyber grid considering the restoration of information nodes is proposed by [71]. They show that deliberate attacks on the SG are more effective than random attacks. The authors do not consider different communication network topologies and initial failures do not occur solely in the communication network.

The authors in [72] propose an SG model in which the interdependency between the power grid and the communication network is asymmetric. They demonstrate that the SG becomes increasingly robust with an increase in the maximum allowable load. Additionally, they state that different communication network topologies have little influence on the robustness of the power grid, which contrasts with findings from other studies. However, the authors consider only one type of communication topology (double-star), do not explore different attack strategies, and only initiate failures in the power grid.

The software-based simulation performed by [73] demonstrates the cascading mechanisms caused by cyber-attacks on a power grid. The focus of this research is primarily on the devastating impact a cyber-attack can have on the performance of the power grid. The authors put less emphasis on the effect of different network topologies and various network attacks, concentrating more on failures within the power grid itself rather than the interdependent connections within the SG.

Similar to [69], the authors in [74] analyse the failure propagation in an SG based on an epidemic model. However, they use this epidemic process in both the communication network and the power grid. The initial failures of the nodes in this model are randomly selected and one type of communication network is generated (double-star).

The authors in [75] model cascading failures using a diffusion and infection process among the nodes in the communication network through malware. They use three different attack strategies for infection, namely high degree, low degree, and random. They analyse the robustness of the SG given a double-star and mesh communication network with increasingly infected nodes. Their results show that under high-degree and random attacks, a double-star network is less robust, while under low-degree attacks, the opposite is true. However, the authors assume that physical node failures have little influence on the cyber nodes and only attack nodes based on degree and random selection.

The robustness of an SG with and without power flow constraints is analysed by [76]. They state that power flow characteristics should be combined with the SG network structure characteristics when evaluating the importance of nodes. Additionally, they find that attacking nodes with both high degree and betweenness centrality can easily lead to the collapse of the SG. However, in this research, only power nodes initially fail, and other communication network structures are not considered.

In [77], the authors model cascading failures in interdependent networks, such as a power grid and a communication network. They simulate node removals both randomly and based on high-degree nodes. They find that the removal of high-degree nodes causes significantly more damage than random failures, rapidly leading to cascading failures and network collapse. However, this model is generic as it does not directly represent a specific power grid or communication network.

The authors of [78] present an interdependent communication network and power grid model using complex network theory. They examine the impact of different attack strategies and increasing failures in the communication network layer on the ratio of edge loss and the ratio of load loss. The study demonstrates that intentional attacks (based on degree and betweenness) have a more significant impact than random attacks, with degree and betweenness attacks showing a similar level of impact. However, the authors do not consider different types of communication network structures to analyse the robustness of the SG.

In [79], the authors model an interdependent communication and power grid network using two different communication network structures: a double-star and a mesh network. They demonstrate that a double-star network has a lower probability of catastrophic failures under random attacks, while the opposite is true for intentional attacks. They conclude that, in most cases, the double-star structure is superior to the mesh network. However, their focus is on attacks and failures initiated in the power grid rather than the communication network.

The impacts of different interdependencies and topology characteristics of communication networks on cascading failures in power grids are analysed by [80]. They use a mesh and a double-star structure for the communication network. The study demonstrates that a double-star structure is better at resisting initial failures during the start of a cascading failure. However, they only perform random attacks on transmission lines in the power grid, without considering different attack strategies on the

communication network.

The authors in [81] propose an interdependent SG model to analyse cascading failures. The authors explore different communication network structures, including double-star, random, and regular networks, as well as various coupling patterns. Failures are initiated in the communication network and spread through a diffusion process. The study concludes that double-star communication networks promote failure spreading in SGs, making them less robust compared to other network structures. In this study, the authors emphasize different coupling techniques of the two interdependent networks rather than the type of nodes that are initially infected (i.e., based on network metrics).

In [82], the authors develop an interdependent communication network and power grid model using two different communication network structures: a double-star network and a random network. Their findings indicate that a double-star network does not always result in better robustness. However, they only perform random attacks, which are not solely initiated in the communication network.

[83] proposes an interdependent smart grid (SG) model and analyses the robustness of the SG under random attacks on the power grid. They use a double-star and a mesh communication network. The authors demonstrate that a double-star network performs better in the case of random attacks. However, they only consider random failures initiated in the power grid and do not incorporate a power flow analysis within the power grid.

A model to analyse cascading failures in an interdependent communication network and power grid is presented by [84]. Although they perform different attack scenarios, their focus is on how their model compares to other SG models. Additionally, they demonstrate that inter-betweenness centrality is a crucial metric for identifying critical nodes to enhance network robustness.

From Table 2.2 it is evident that the majority of papers include aspects A to E. This indicates that most papers model an interdependent cyber-physical system with heterogeneous power grid and communication network components. Additionally, the majority of papers focus on failures or attack scenarios on the SG. However, the majority of the papers do not include aspects F to K in their research, or these aspects are not the primary focus of their studies. Aspects F to K concentrate on vulnerabilities (failures or attacks) within the communication network from a network theory perspective. Even though these aspects are included in the search terms for the papers, they do not appear or are not strongly represented in the analysed papers. One reason for this is the limited availability of data on communication networks of power grids, due to confidentiality and regulatory constraints [61, 52]. It is interesting to note that the majority of the relevant papers we selected do not address communication network failure propagation by using, for example, diffusion, epidemic, or virus spread models (aspect L). From the selected literature, only [69, 74, 75, 81] employ this approach. However, more literature exists that use this communication network failure propagation approach (e.g. [85, 86, 87]). Nonetheless, we did not come across any literature that compares these two approaches in the context of SGs (with and without communication network failure propagation).

From the analysed literature, it becomes evident that certain aspects are included (A to E) yet some aspects are not strongly represented or absent (F to K). Additionally, the literature differs in the modelling of failures (aspect L). These missing aspects allow us to identify and formulate a knowledge gap:

The knowledge gap that needs to be addressed is understanding the effect of communication network failures on the underlying interdependent power grid from a network perspective

With our research, we aim to close the knowledge gap by including all 12 aspects mentioned in Table 2.2. Our focus is on initial failures in the communication network and their impact on the interdependent underlying power grid. Additionally, we seek to gain insights by highlighting the differences in impact with and without a communication failure propagation process. To date, we have not encountered any literature that addresses this approach.

In this chapter, we have provided background information from the literature on SGs, the power grid and communication network components, and their relationship to cyber threats. While many studies focus on the interdependent behaviour of the SG and address failures or attack scenarios within this coupled cyber-physical system, our literature review highlights a knowledge gap, particularly in the area of failures in the communication network and their effect on the underlying power grid from a network

perspective. In the following chapter, we present our research questions to address the identified knowledge gap and elaborate on the method, theory, and approach used to answer these questions.

3

Research Design

In this chapter, we present the research design to address our research questions (RQs), which are categorized into the main research question (MRQ) and research sub-questions (RSQs). These questions arise from our research objective and the knowledge gap identified in the previous chapters. Subsequently, we explain the method and theory used to answer the formulated RQs. Finally, we outline the approach to addressing the RQs within the chosen methodology.

3.1. The research questions

To understand the effect of failures in the communication network on the underlying power grid, RQs have been formulated to address the knowledge gap identified. The RQs are formulated as follows:

Main research question:
What is the effect of failures in the communication network on the underlying power grid?

Research sub-question 1:
Which communication topology is the most robust against failures, causing the least disruption to the functionality of the underlying power grid?

Research sub-question 2:
Which network-based attack strategy on the communication network is the most effective in causing the most disruption to the functionality of the underlying power grid?

By answering these RSQs, we understand the weaknesses and strengths of the communication network of the SG based on the topology and different attack strategies. These insights bring forward an answer to the effect of failures in the communication network on the underlying power grid, which is our MRQ. The next section provides information on the methodology and theory used in this research to answer the RQs.

3.2. Method and theory

3.2.1. Mathematical simulation model

To study the effects of failures in the communication network on the power grid, a network-based model is created. One of the reasons for choosing this method is that an SG can be presented as a graph [20]. By modelling an SG as a graph and using network modelling techniques, we conduct analyses on specific network functions, such as robustness [88] and power grid system failures [89], which is the focus of this research. To analyse the effects of the communication network on the power grid, simulations are run. Simulations offer a practical way to evaluate and compare multiple solutions [90], or in our case, also comparing various failure scenarios.

3.2.2. Graph theory: The study of networks

The foundation on which our model is based is graph theory, specifically network theory. We briefly discuss the concepts of this theory to understand the principles on which our model is based. For more information on this theory, we refer to the book *Networks, Crowds and Markets* [91]. The mathematical descriptions are retrieved from [92].

A graph G is a mathematical structure consisting of nodes N (also referred to as vertices) and edges E (links). Each edge has a set of one or two vertices associated with it (i.e., it connects nodes). Thus, a graph can be represented as $G = (N, E)$. To illustrate this mathematical structure, an example is given in Fig. 3.1.

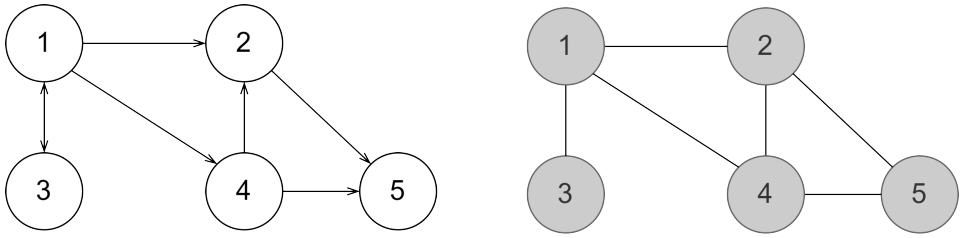


Figure 3.1: An example of a directed graph (left) and an undirected graph (right).

The graph on the left-hand side is a directed graph. Which is expressed as follows:

$$\exists i, j \in N : (i, j) \in E \Leftrightarrow (j, i) \notin E$$

This means that there exists at least one pair of nodes i and j in the graph where there is a directed edge from i to j but not from j back to i. Specifically, the nodes N and edges E can be specified as:

$$N = \{1, 2, 3, 4, 5\}$$
$$E = \{(1, 2), (1, 3), (1, 4), (2, 5), (3, 1), (4, 2), (4, 5)\}$$

Besides a directed graph, there is also an undirected graph, the right graph in Fig. 3.1. This means that for every pair of nodes i and j in the graph, there is an undirected edge from i to j and from j back to i. This can be expressed as follows:

$$\forall i, j \in N : (i, j) \in E \Leftrightarrow (j, i) \in E$$

The nodes, N, are expressed similarly to those in the directed graph. The sets of edges, however, have increased:

$$E = \{(1, 2), (2, 1), (1, 3), (3, 1), (1, 4), (4, 1), (2, 5), (5, 2), (3, 1), (1, 3), (4, 2), (2, 4), (4, 5), (5, 4)\}$$

This theory applies to developing an SG model by creating two interconnected networks (graphs), one representing the power grid and one representing the communication network. This is further elaborated in the next chapter.

3.2.3. Power flow analysis

The second foundation of our model is the Power Flow Analysis (PFA). To gain a deeper understanding of the equations and calculations behind PFA, we refer to [93]. According to [93], PFA is one of the key analyses widely used in power system operation and planning. It involves building a power flow model of the power system using relevant network, load (consumption), and generation data. It is a model used in the calculation of voltages at different buses, line flows in the network, and system losses through solving nodal power balance equations. These equations are nonlinear and usually require Newton-Raphson (used in this research), Gauss-Seidel, and other fast-decoupled techniques for iteration to convergence. The objective of a power flow study is to determine the voltages (both magnitude and angle) for the specified load, generation, and network conditions. Once these voltages are known for all the buses, the flow in lines and system losses can be determined. Thus, incorporating PFA in our model enhances realistic electrical characteristics.

3.3. Steps and approach

We choose to adopt the five-step modelling and simulation life-cycle presented by the authors in [94]. We simplify and adjust the modelling steps into four steps. This life-cycle conveys the important steps in the process concisely and is shown in Fig. 3.2. In this process, there are four steps: problem definition, conceptual modelling, implementation, and experimentation. For a more detailed description of each step, we refer to [94].

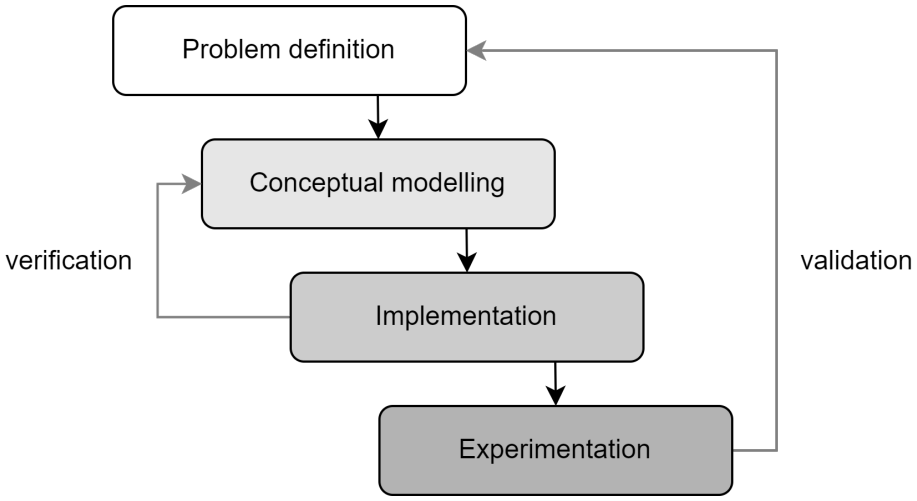


Figure 3.2: Modelling and simulation life-cycle adopted and altered from [94].

The first step is the problem definition, which we describe in the previous chapters. It sets the boundaries of the system and outlines the reasons and goals for performing such a modelling simulation.

The second step, conceptual modelling, involves a high-level abstraction of the actual system [94]. Furthermore, it serves as a communication bridge between the reader and the simulation modeller (i.e., the researcher) [94].

In the third step, implementation, the specified model is implemented (in our case programmed) on a specific platform to create a working model.

In the final step, experimentation, simulations are performed on the model based on various given situations in which certain parameters are modified. The data generated in this phase is collected and interpreted to gain insights.

Throughout the process, there are also verification and validation steps. Validation shows that your results are correct and based on strong (scientific) evidence (i.e., that you have built the right system) [95]. Verification implies proving that the method of research has been used correctly and is suitable for the research topic you are investigating (i.e., are you building your model correctly) [95].

In this chapter, we have presented the research design to address our MRQ and RSQ, which are derived from our research objective and knowledge gap. We also introduced the methods and theories used, focusing on network and graph theory as well as PFA, to model and simulate the SG. In the following chapter, we introduce our model, emphasizing the conceptual and implementation phases of our research.

4

Model

In this chapter, we present our conceptual and operational model of the SG. The conceptual model is translated into the operational model using Python and pandapower. Furthermore, we elaborate on the choice of communication network structures, and the data and parameters for both the communication network and power grid.

4.1. The conceptual model of the smart grid

4.1.1. Structure of the conceptual smart grid model

We propose our conceptual model by elaborating on the components of the communication network and power grid and how they are connected. Furthermore, we explicitly state the main assumptions made to ensure transparency.

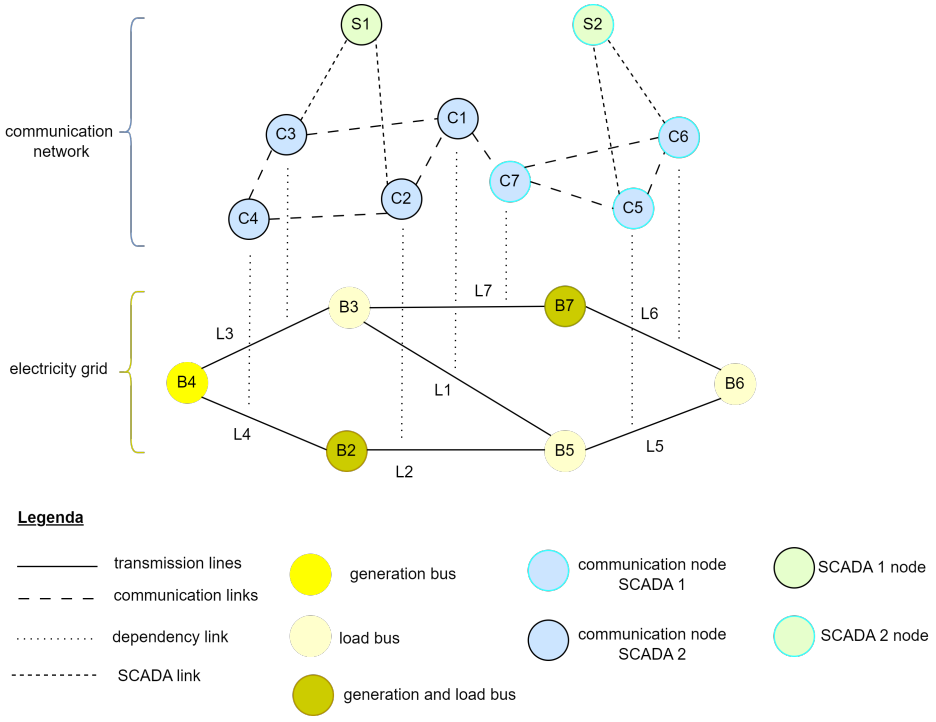


Figure 4.1: An example illustration of the conceptual model.

The structure and components of our SG are illustrated in Fig. 4.1. Note that the figure serves as an example rather than the actual model. Firstly, we elaborate on the power grid. The power grid E contains buses B_1 through B_n connected by transmission lines L_1 through L_y. A bus serves various

purposes within the power grid. It may be connected to a generator (B_4 in our example), a load, which is a type of node that consumes power (B_3, B_5, and B_6), or both components simultaneously (B_2 and B_7). Alternatively, it may serve solely as a junction point, distributing electricity to other buses (not included in our example).

Transmission lines connect these buses. These lines transport the electricity from the buses with connected generators to buses with loads. In our model, the amount of electricity transported through the grid depends on the power demand of the load components. The amount of electricity flowing through a transmission line is influenced by a variety of factors, including the generators and loads connected to it and physical line properties, such as resistance.

Secondly, the multi-layer communication network. Our communication network G contains communication nodes C_1 through C_y and SCADA nodes S_1 through S_z. A C_y node in G may represent various components such as a control mechanism (e.g., a circuit breaker), sensor equipment, an RTU, or other devices related to control, measurement, or communication. In other words, it measures, controls, and communicates information of the transmission lines to other nodes in the communication network. This has been explained in more detail in §2.1.3. Since for each transmission line there is exactly one communication component, the number of C_y nodes equals L_y. This indicates a one-to-one dependency, also employed in [65, 68, 83, 82]. We also include SCADA nodes in our network. These nodes represent the decision-making component in our communication network. Their role is to oversee and control larger parts of the communication network and power grid. In our model, several parts of the communication nodes, and therefore also the power grid, are associated with a corresponding SCADA node. However, not every communication node is directly connected to its corresponding SCADA node.

In our model, the communication links between nodes do not reflect literal data transmission but rather dependency relationships. This is deliberately done to emphasize certain points associated with network theory. Lastly, the dependency link. As explained earlier, one transmission line is dependent on one communication node. Given the different failures or attacks described in §2.2, our model assumes that if a communication node fails, the associated transmission link to this node fails as well (e.g., switched off). Since our model is interdependent, it means that if a transmission line fails (during the failure process), the corresponding communication node fails as well. In essence, the operation of the transmission line depends on the control, measurement, and communication behaviour of the network, while the operation of the communication node depends on the power supplied by the transmission line. A comparable approach to this interdependency is used, for example, in [55, 63, 58, 62].

4.1.2. Failure process of the conceptual model

In Fig. 4.2, we outline the steps of the failure process in our conceptual model. Note that communication network failure propagation is a conditional parameter. If this condition is active, the second step is included; otherwise, it is skipped. Note that only the communication network failure propagation part is skipped in this step, not the failure based on SCADA connection. This condition highlights the impact of communication failure on the power grid, both with and without failure propagation. We illustrate these steps using an example failure scenario with the communication network failure propagation condition.

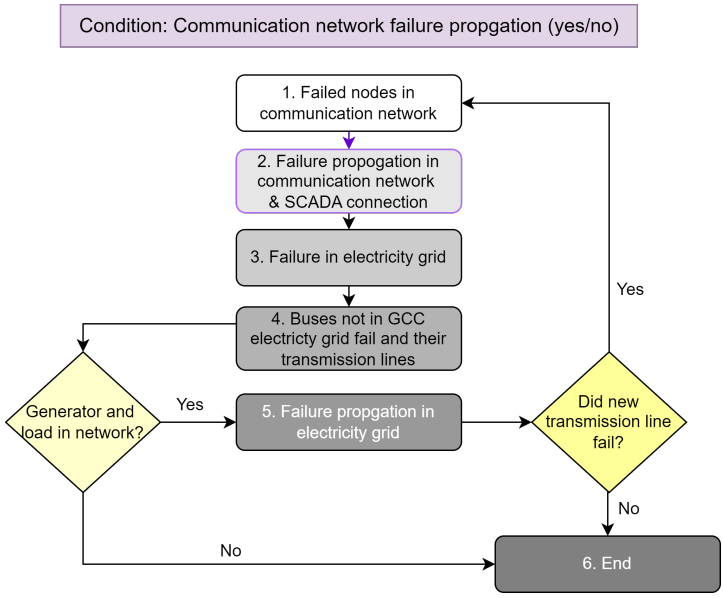


Figure 4.2: Failure process steps in the conceptual model.

Step 1: Initialising failed nodes in the communication network

In the first step of our simulation, certain nodes in the communication network are initialised as failed. Which nodes initially fail is based on specific centrality metrics or random selection. Note that the SCADA nodes are fail-safe (i.e., under no circumstances can they be initialised as failed or fail during the failure propagation). In Fig. 4.3, node C_1 in the communication network is set as failed.

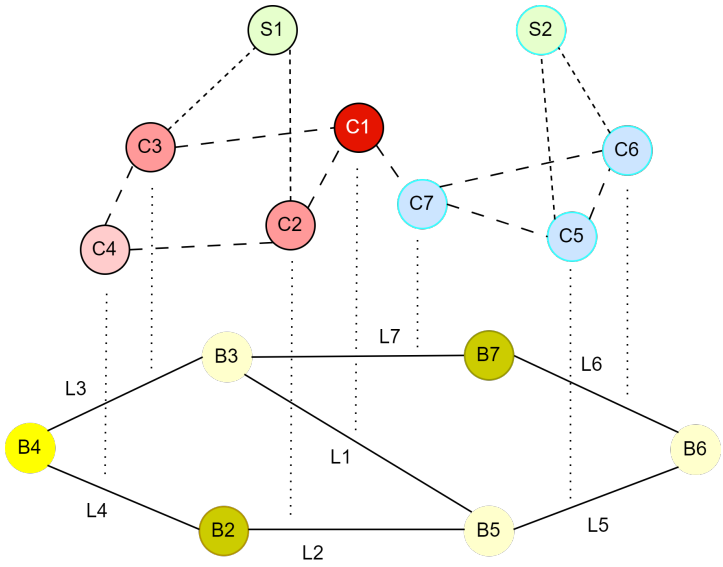


Figure 4.3: Step 1 and 2 of the conceptual failure propagation.

Step 2: Failure propagation in the communication network and SCADA connection

The second step concerns how the initial communication node failures propagate throughout the communication network. This step is only executed if the conditional parameter of failure propagation is activated. Failure propagation is included to simulate the spread of failures among communication network nodes, for example, due to a virus, malware, or interdependencies among the communication nodes. As stated earlier, the communication links describe dependency and not actual data transmission. This is done to focus on certain aspects of network theory (e.g., the topology). Additionally, in this step, it is checked if each communication node has a path to a SCADA node (this is always done

with or without the failure propagation condition).

Other nodes in the communication network are considered to have failed if one of the following two criteria is met. Firstly, a node fails if the fraction of its failed neighbours exceeds a specified threshold (e.g., the virus or malware has spread successfully, or a certain amount dependency nodes have failed). For example, with a threshold of 0.5, the node fails if 50% or more of its directly connected neighbours have failed (this criterion is part of the failure propagation condition). Secondly, a node is considered failed if there exists no path from the communication node to its corresponding SCADA node. This path can only consist of non-failed nodes (this criterion is not part of the failure propagation condition).

In Fig. 4.3, the initial failure of C_1 results in nodes C_2 and C_3 exceeding a pre-specified threshold, leading them to be marked as failed. The failure of these nodes, in turn, causes C_4 to fail for two reasons: the threshold is exceeded and there exists no path from C_4 to its corresponding SCADA node S_1, resulting in C_4 also being marked as failed. Node C_7 does not exceed the threshold and therefore does not fail.

Step 3 and 4: Transmission line failures and the giant connected component

After the failure propagation in the communication network stops, the corresponding transmission line to the failed communication node is set as failed as well. In Fig. 4.4, since nodes C_1, C_2, C_3, and C_4 failed, the dependent transmission lines L_1, L_2, L_3, and L_4 have failed as well.

Failures in the transmission lines can cause segmentation in the power grid, forming isolated sections known as 'islands'. In this situation, our model keeps the giant connected component (GCC). A GCC is a connected component of a network that contains a significant proportion of the entire nodes in the network [96]. Furthermore, transmission lines excluded from the GCC are regarded as having failed as well. This is an important note for the interdependent behaviour of our SG model. Our model continues with the failing process if there exists at least one generator and load in our connected SG model (i.e., generated electricity can still be delivered to a consumption node).

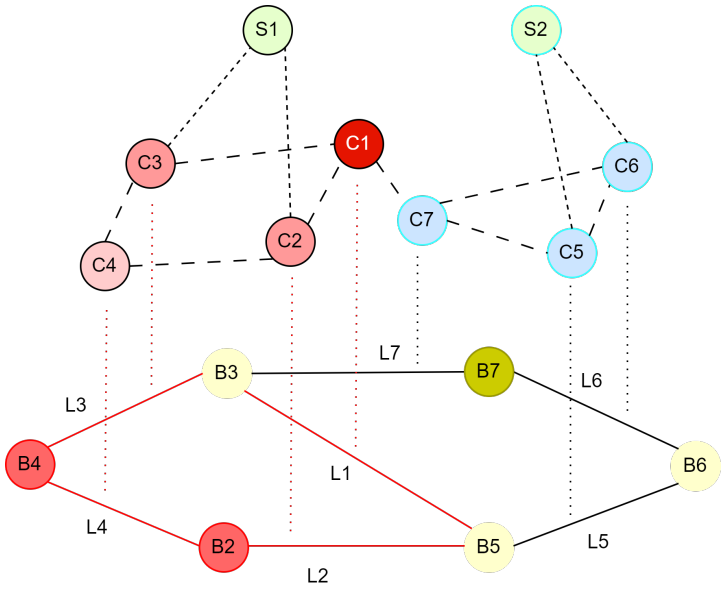


Figure 4.4: Step 3 and 4 of the conceptual failure process.

Step 5: Failure process in the power grid

In step 5, following the initial transmission lines failing (either by not being part of the GCC or due to failed communication nodes), a PFA is conducted to determine the redistribution of electricity. If, in the PFA that follows, any of the transmission lines are found to be carrying more power than their predefined transportation capacity, the transmission lines are regarded as damaged and hence failed. In Fig. 4.5, after the PFA is conducted, transmission lines L_5 and L_6 have exceeded their capacity and have failed.

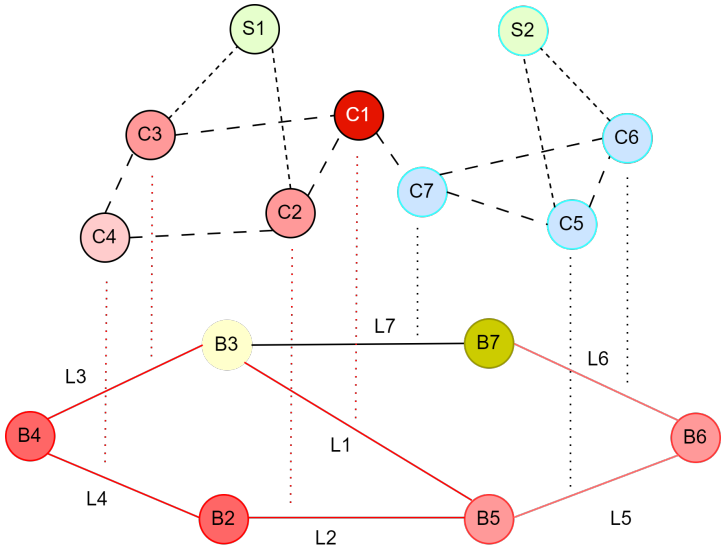


Figure 4.5: Step 5 of the conceptual failure process.

Interdependent behaviour and step 6

If additional transmission lines fail, either by being overloaded or not being part of the GCC, this indicates that the corresponding communication node has failed as well. In our example, in Fig. 4.6, lines L_5 and L_6 fail, and their corresponding communication nodes C_5 and C_6 have now failed as well.

Now, the failure process in the communication network is conducted again based on the two criteria (the failure propagation threshold and existing path to a SCADA node). Note that although the nodes and lines look grey, all of them are still in the failed state. The colouring is applied to draw attention to the selected sections of the network for failure propagation analysis. Now all of the neighbours of node C_7 have failed, namely C_1, C_5, and C_6. Therefore, this node fails for two reasons: there is no path from C_7 to S_2 and the number of failed neighbours has exceeded the threshold. This implies that the corresponding transmission line L_7 has failed, there exists no generator and load in the GCC (as B_3 and B_7 are no longer connected), resulting in a complete cascade in our SG model.

Although our example results in a complete blackout, this outcome does not have to happen in every instance.

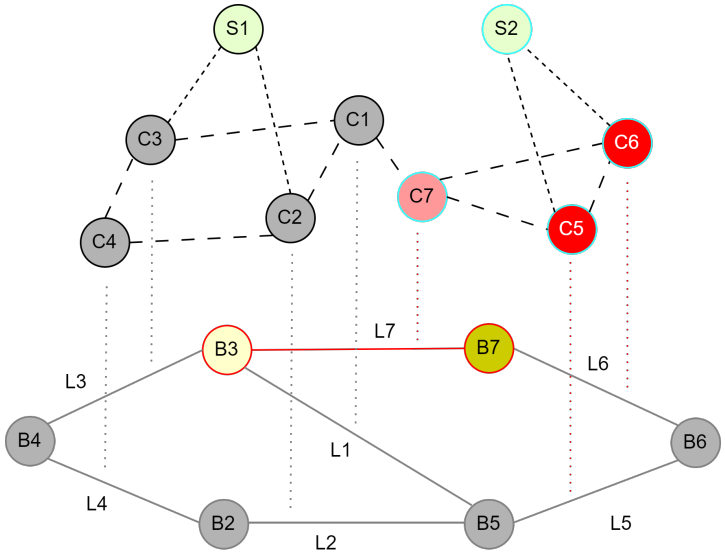


Figure 4.6: Interdependency and step 6 of the conceptual failure process.

4.2. Model implementation with Python

4.2.1. Tools and packages: Networkx and pandapower

Networkx Python package

For the creation of our communication network, NetworkX is used. According to NetworkX, it is a Python package for the creation, manipulation, and study of the structure, dynamics, and functions of complex networks [97, 98]. This package can be used to create the graphs discussed in §3.2.2 to form our communication network and process the failure propagation in our communication network. For further information on NetworkX, we refer to [98, 99].

Pandapower tool and The IEEE 118-bus system

To implement the power grid and perform a PFA, we use pandapower. According to pandapower, it is an easy-to-use open-source tool for power system modelling, analysis, and optimisation with a high degree of automation [100, 101]. In our model, we use pandapower to run an AC power flow using the Newton-Raphson algorithm. For more information about the calculation of such PFA and pandapower, we refer to [100, 93, 102].

Our SG model adopts the IEEE 118-Bus System for the power grid, which is provided by pandapower. This system represents a part of the American Electric Power System (in the Midwestern US) as of December 1962 [103]. When loading the 118-bus system test case file from pandapower, it includes the elements shown in Table 4.1.

Table 4.1: Elements in the 118-bus test case from pandapower.

Elements	Amount
Bus	118
Load	99
Generator	53
Transmission Line	173
External Grid	1
Transformer	13
Shunt	14

From the table above, we see that there are elements that do not directly correspond to our conceptual model. Firstly, the external grid represents a higher-level power grid connection and is modelled as the slack bus in the power flow calculation [100]. When performing a PFA on an power grid, a slack bus must be present. How we handle the absence of a slack bus, which initially is an external grid (e.g., when all the transmission lines of the external grid have failed), is explained in the next section. Next, transformers are included in the 118-bus test case. In pandapower, these represent a two-winding transformer from an HV-bus to an LV-bus [100]. When translating the pandapower 118-bus test case to NetworkX, the transformers are represented as edges. However, in our model, only transmission lines are interdependent with communication nodes (i.e., transformers do not have corresponding communication nodes). Lastly, shunts are elements in the network that represent a capacitor or reactor [100]. For more information about these elements, we refer to [100, 101].

4.2.2. Network setup and failure process of the operational model

To understand the implementation of the SG model, it is important to understand the different steps of setting up the operational SG model and the operational failure process. These steps are more extensive than the conceptual model steps from Fig. 4.2 due to the requirements mentioned in §4.2.1. The setup and simulation steps are illustrated in Fig. 4.7.

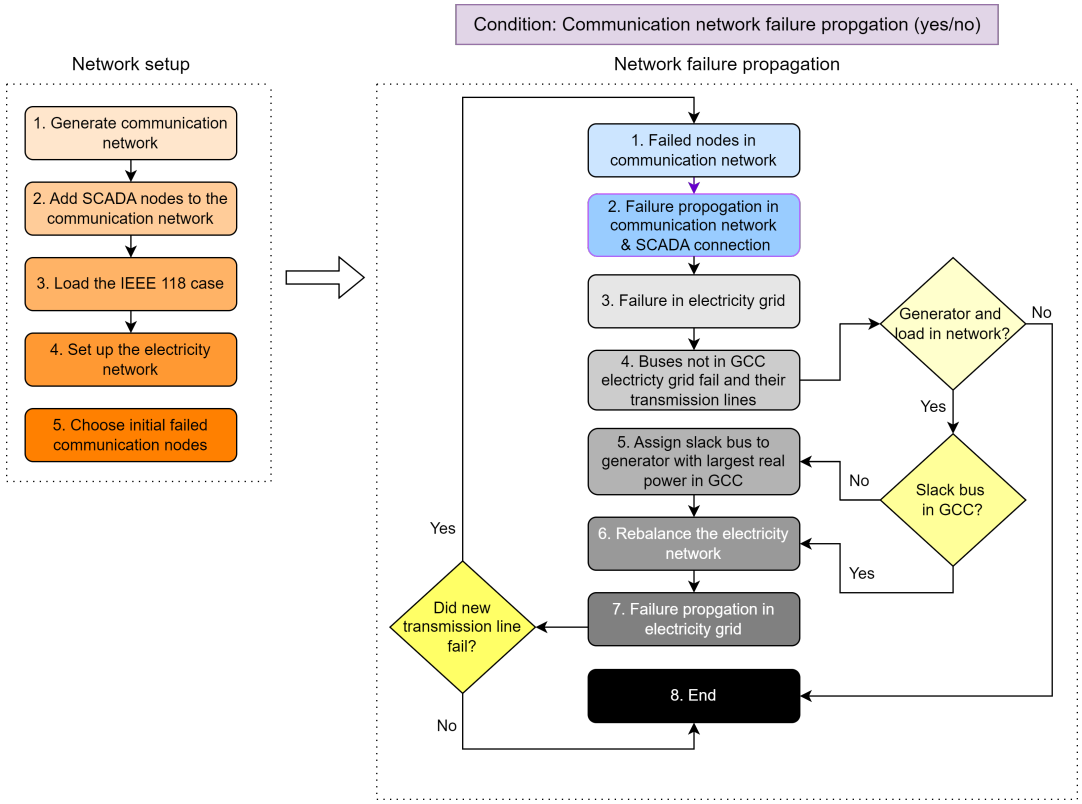


Figure 4.7: Network setup and failure process steps of the operational model.

We do not elaborate on each step in Fig. 4.7, we only mention the steps that are not in the conceptual model. Firstly, the network setup steps are performed to prepare the model for the failure process. This includes generating the communication network, importing the 118-bus test case, and selecting the initial failed communication components based on network metrics (excluding the SCADA nodes). Note that the third step in the network setup is changing the values of the 118-bus test case. This is done to create a higher load in the whole network and activate every generator.

After the network setup is completed and the initial failures are chosen, we start the failure process steps (as shown on the right-hand side of Fig. 4.7). Compared to the conceptual model steps, there are two extra steps and one additional decision-making instance (the yellow triangles). We elaborate solely on these additional steps and decision-making instances. After step 4 in the network failure process, we check if there is a load and generator in the network (as in the conceptual model). The difference now is that it checks if there is a slack bus in the network (this is necessary to run a PFA). If a slack bus is absent, the working generator with the highest real power in the network is assigned as a slack bus. This simple solution allows the failure process sequence to continue in our model. Once this is carried out, or if the slack bus is still in the GCC, the load in the power grid is rebalanced. This rebalancing is necessary to match the load demand with the generators. Otherwise, pandapower will raise an error, and the PFA will be interrupted. In our model, the generators adjust their output based on the remaining load demand. Note that even after rebalancing, pandapower might still raise an error indicating that the PFA did not converge after several iterations. In our case, pandapower converged when the load demand was decreased (after an error had occurred). To handle these convergence issues, we decrease the load demand by 10% if such an error occurs. This simple solution allows us to keep the model running. In our simulation, this only occurred less than 0.02% of all the runs. After this, failure propagation in the power grid is checked, just as in the conceptual model.

4.2.3. Python code explanation of the operational model

We discuss the important parts of our code that constitute our operational model. The Python code can be found in Appendix A. In Appendix A.1, we provide an overview of the modules, packages, and functions that are imported to create the model. We use Python vers. 3.9.18 within Jupyter Notebook

vers. 7.0.8, all managed through Anaconda. Note that several functions are specific to our experiments.

The communication network generation is performed using NetworkX (Appendix A.2). Two different types of networks are generated: a mesh network and a double-star network. After the communication network generation, SCADA nodes are added (Appendix A.3). In our model, three SCADA nodes are created: nodes 0 through 56 belong to SCADA 1, 57 through 115 to SCADA 2, and 116 through 172 to SCADA 3. Each SCADA node is then connected to 25% of its corresponding communication nodes.

Using pandapower, the 118-bus test case is loaded and certain parameters are altered to give the network a higher load (Appendix A.4). The initially failed nodes, excluding the SCADA nodes, in the communication network are based on network metrics (degree centrality, betweenness centrality, closeness centrality and random) and the percentage of nodes (Appendix A.5). These code parts complete the four orange-coloured steps of the network setup, illustrated in Fig. 4.7.

In the next phase, failure propagation, the nodes chosen to initially fail from the previous phase are marked as failed in the communication network. Next, the code checks for additional node failures based on the input parameter criteria of failed neighbours (if the communication network failure propagation parameter is activated) and existing paths from the nodes to their corresponding SCADA node (Appendix A.6). The code for this failure propagation model is partially adopted from [104]. This part of the code concludes the first two blue-coloured steps of the failure propagation in Fig. 4.7.

After gathering the failed communication nodes using NetworkX, the corresponding transmission lines are switched off using pandapower (Appendix A.7). If no communication nodes have failed, then all transmission lines remain in-service. Next, we check if there is a generator and a load in the GCC (Appendix A.8). If there are none, we stop the simulation. Then, we turn off the buses that are not part of the GCC. Once this process is completed, we check if there is a slack bus remaining in the GCC (Appendix A.8). If there is no slack bus, the generator with the largest real power is assigned as a slack bus. These steps comprise the first three steps of the grey-coloured process steps in Fig. 4.7.

Then we rebalance the power in the power grid (Appendix A.9). If supply and demand are not matched correctly, pandapower raises an error. Therefore, this step is necessary to match the generated power and the demanded power. Note that we only balance the active power in the network and not the reactive power. The first step is checking how much load is demanded. The second step is adjusting the real power outputs of generators based on the load demand. This adjustment is based on the active power generation for each in-service generator based on their proportion of the total maximum in-service generation capacity. In other words, the higher the maximum generation capacity, the larger the adjustment, as these generators are considered more significant. After the rebalancing, the PFA is performed (Appendix A.10). If the PFA does not converge, the load is decreased by 10%, the network is rebalanced, and a PFA is performed again. When this is done, we check if lines or transformers have exceeded their maximum load capacity (Appendix A.11). If they have exceeded their capacity, they are switched off. We also provide a code snippet on the process of checking for additional component failures compared to the start of each run (Appendix A.12). If there are no additional failures, the loop is broken; otherwise, the above steps are repeated. These steps conclude the last three steps of the failure process process (Fig. 4.7).

4.3. Data and parameters of the operational model

4.3.1. Communication network structures and parameters

A communication network has a double-star structure (scale-free) or a mesh structure (small-world) [105, 62]. It is shown in statistics that a scale-free network follows the Barabási-Albert model [106, 62, 75] and a small-world network follows the Newmann-Watts model [107, 62, 75]. For this reason, we choose to create two different types of communication networks: a double-star (scale-free) network and a mesh (small-world) network.

To create the double-star network, we use the Barabási-Albert model with $n = 173$ (the number of nodes, which equals the number of transmission lines) and $m = 5$ (the number of edges to attach from a new node to existing nodes). To create a mesh network, we use the Connected Newman-Watts-Strogatz model, also with 173 nodes. With this model, we set $k = 6$ (the number of nearest neighbours each node is connected to) and $p = 0.2$ (the probability of rewiring each edge). The same value for p is chosen as was done by the authors in [61]. The average degree of each node and the average clustering coefficient of both networks are displayed in Table 4.2. The double-star network has a higher

average degree compared to the mesh network. This means that on average each node in this network has more connections to other nodes in the network. The average clustering coefficient is higher in the mesh network. This coefficient is calculated by taking the average of all local clustering coefficient (e.g. of each node). A local clustering coefficient of a node is calculated by dividing the connections among a node's neighbours that are actually present by the number of all possible connections among the node's neghbours.

Table 4.2: Comparison of double-star and mesh networks (averaged over 100 simulations).

Network Type	Average Degree	Average Clustering Coefficient
Double-Star	9.71	0.13
Mesh	7.21	0.44

These two networks also differ in their degree distribution. In Fig. 4.8 and Fig. 4.9, we show the degree distribution of both networks. It is evident that the double-star network follows a power-law distribution, whereas the mesh network does not exhibit such extreme degrees.

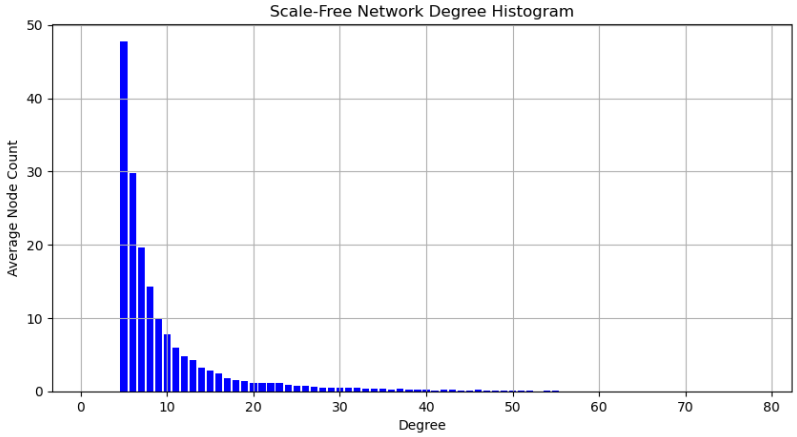


Figure 4.8: Average degree distribution for double-star network (100 simulations).

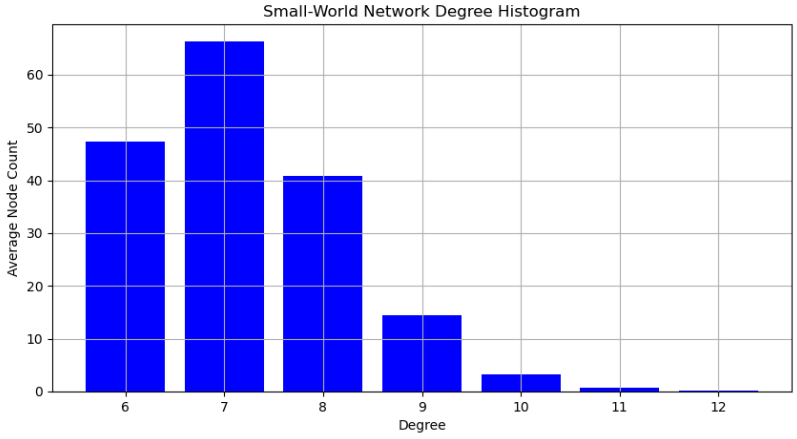


Figure 4.9: Average degree distribution for mesh network (100 simulations).

We also include the number of SCADA nodes in the communication network as a parameter. In our model, we consistently incorporate three SCADA nodes in all instances. The reason for this number is that the 118-bus system can be divided into three zones, as shown in [108, 109]. In our model, each

SCADA node corresponds to one-third of the generated communication network. However, we choose to connect only 25% of the corresponding nodes to their SCADA node. This approach allows us to simulate the impact of failure when there is no path from a node to its corresponding SCADA node. We also include the fail criteria as a parameter (i.e., the fraction of directly failed neighbours required for a node to fail). We set this parameter to 100% to not activate the communication network failure propagation. This means that a communication node only fails if it is initialised as failed, the connected transmission line has failed, or there is no path from the communication node to the corresponding SCADA node. However, when we activate this parameter, it is set to 0.5, meaning at least 50% of the direct neighbours in the communication network need to fail for a communication node to fail (e.g., the virus or malware has spread successfully and infected the node, or a certain number of dependency nodes have failed, leading to a failure). This is chosen to clearly demonstrate the difference in impact when running the model with and without communication network failure propagation (i.e. the parameter is set to 1.0), which allows us to analyse the robustness of the SG under these two different conditions.

4.3.2. Power grid data

The characteristics of the power grid (118-bus system) can be found in the Appendix B. We provide data on the transmission lines (Appendix B.6), buses (Appendix B.1), generators (Appendix B.3), loads (Appendix B.2), shunts (B.4), transformers (Appendix B.7), and external grids (B.5).

From Table 4.3 and Fig. 4.10, it is evident that the average line and transformer load is extremely low (0.41% and 1.12%, respectively) in the original 118-bus test case. This may be because the line MVA limits were made up as they were not part of the original data [103]. To model the effect of overloading lines due to failures, we increased the overall line load in the power grid. This is achieved with the network setup function (Appendix A.4). This results in an overall line load of 21.67%, as seen in Fig. 4.11.

Table 4.3: Comparison of average line and transformer load before and after network setup for 118-bus system.

118-Bus System	Average Line Load (% Max Capacity)	Average Transformer Load (% Max Capacity)
Original	0.41	1.12
After Network Setup	21.67	1.11

From Fig. 4.10, it can be concluded that the overall network load is very low, which reflects the low average load stated in Table 4.3. After performing our network setup, it is evident in Fig. 4.11 that the majority of the network has an increased load. However, some parts of the network still have a low load. The complexity of such a power network makes it challenging to evenly increase the load throughout the network. The transformer load remains low after the network setup because changing the transformer values is more complex and requires a better understanding of the electrical properties of a power grid. It is easier to adjust the line loads, and therefore we only adjust the line character values.

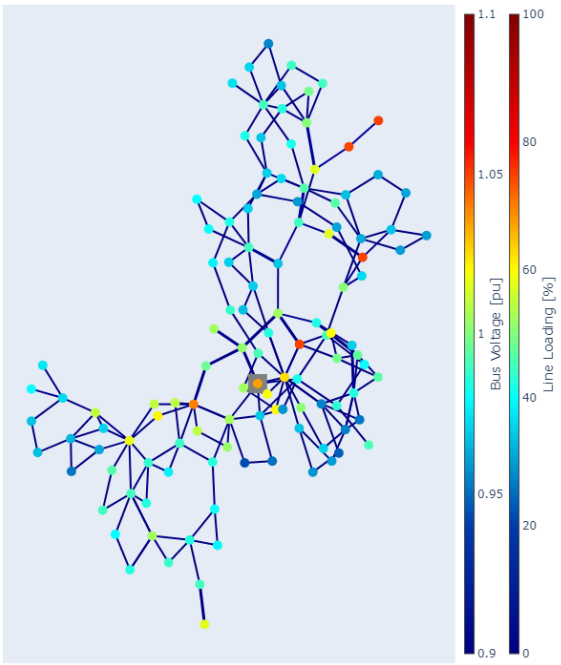


Figure 4.10: Voltage and line loading distribution for the original 118-bus test case.

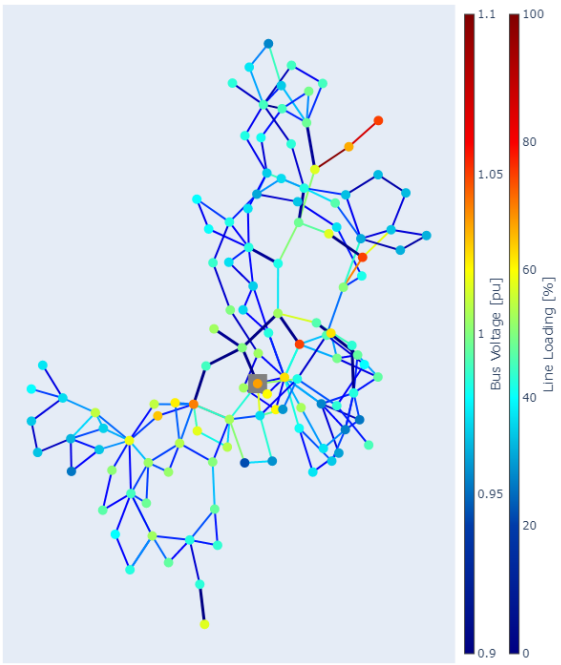


Figure 4.11: Voltage and line loading distribution for the 118-bus test case after network setup.

In this chapter, we have presented our conceptual and operational model created using Python's packages NetworkX and pandapower. For the communication network generation, we used the double-star and mesh topologies. For the power grid, we used the IEEE 118-bus system. For both these networks, we provided the parameters and data to obtain network characteristics. We have presented the different steps in the failure process for both the conceptual and operational models, highlighting the difference between running the model with and without communication network failure propagation. Having explained our model in depth, we can now use this knowledge to create performance metrics and experiments to answer our RSQs, which are addressed in the next chapter.

5

Experiments

In this chapter, we present the experimental design that enables us to answer the research questions formulated in §3.1. Additionally, we describe the metrics used to analyse the data generated by the experiments. Finally, we provide a hypothesis for each RSQ.

5.1. Experimental design

5.1.1. Network-based attack strategy using centrality measures

As discussed in §4.3.1, we generate two different communication networks: a double-star network and a mesh network. In our experiments, we use three different centrality metrics and random selection to initially fail communication network nodes. Note that these centrality measures are for an undirected graph. Firstly, betweenness centrality measures the importance of a communication node based on the number of shortest paths that pass through it; the more such paths a node is part of, the more important it is considered. The calculation for this centrality is given by Eq. 5.1. Secondly, degree centrality measures the importance of a communication node based on the number of direct connections it has to other nodes (i.e. the number of neighbours). The higher the number of neighbours, the more important the communication node is deemed, which is formulated in Eq. 5.2. Thirdly, closeness centrality measures how close a communication node is to all other communication nodes. The closer a communication node is to other nodes (i.e. a high closeness centrality), the more important it is viewed. This centrality measure is formulated in Eq. 5.3. Lastly, the random attack strategy is not a centrality measure. However, it is interesting to see how a random attack strategy compares to the previously mentioned network-based attack strategies. The mathematical equations for the centrality metrics below are retrieved from [110].

$$C_{\text{betw}}(i) = \frac{1}{(|N|-1)(|N|-2)/2} \sum_{\substack{j,k \in N \setminus \{i\}, \\ j<k}} \frac{\sigma_{jk}(i)}{\sigma_{jk}} \tag{5.1}$$

where:

- σ_{jk} is the total number of shortest paths from node j to node k,
- $\sigma_{jk}(i)$ is the number of shortest paths from node j to node k that pass through node i.

$$C_{\text{deg}}(i) = \frac{deg(i)}{n-1} \tag{5.2}$$

$$C_{\text{clo}}(i) = \frac{|N|-1}{\sum_{j \in N, j \neq i} \text{dist}(i,j)} \tag{5.3}$$

5.1.2. Metrics for assessing smart grid robustness

To evaluate how the SG performs with varying communication network types, attack strategies, and the number of initially attacked nodes, we develop three metrics to measure the interdependent network's robustness, as shown in Table 5.1.

Table 5.1: Collected data metrics and their descriptions.

Collected Data Metric	Description
Power Node Fraction Survived	The fraction of nodes still active in the power grid
Total Network Node Survived	The fraction of nodes still active in the power grid and communication network
Demand Survivability	The amount of power still delivered to the load components

Firstly, the power node fraction survived measures the number of power nodes that are still active in comparison to the total number of power nodes (in our model, the number of buses). This metric is shown in Eq. 5.4. Secondly, the total network node survival, which can be seen in Eq. 5.5 measures the active buses in the power grid as well as the active communication nodes in the communication network (excluding the SCADA nodes). The last metric, demand survivability, measures how much active power is delivered to the load components in the power grid compared to the total initial active power demand (reactive power is excluded). This metric is shown in Eq. 5.6

$$S_{\text{power nodes}} = \frac{N_{\text{power nodes, not failed}}}{N_{\text{power nodes, initial}}} \tag{5.4}$$

$$S_{\text{total}} = \frac{N_{\text{power nodes, not failed}} + N_{\text{communication nodes, not failed}}}{N_{\text{total initial}}} \tag{5.5}$$

$$DS = \frac{P_{\text{delivered}}}{P_{\text{initial demand}}} \tag{5.6}$$

5.1.3. Experimental plan for simulation

We have defined the network types (§4.3.1), attack strategies (§5.1.1), and metrics to evaluate the robustness of our SG model (§5.1.2). To answer the RSQs, we develop an experimental design that considers the different network types and attack strategies. It is also important to vary the number of initially attacked communication nodes, which is also included in the experimental design.

The setup of our experimental plan is shown in Table 5.2. For each network type (mesh and double-star), we fail the initial nodes based on the three different metrics and random selection each time at a given percentage step (from 0% to 90% in increments of 5%). For example, during one run, we fail 10% of the communication nodes in both a newly generated mesh network and a double-star network. We begin by failing the 10% of nodes with the highest betweenness centrality. Next, using the same network structure, we fail 10% of the communication nodes based on degree centrality. Importantly, the failures based on the three centrality measures and random selection are performed independently on the same set of networks. We also do this for the closeness centrality and random attack strategy. Note that SCADA nodes are excluded from this communication node selection process as they are fail-proof. We perform two simulations: one without communication network failure propagation, which has a threshold of 100% (i.e., 100% of the communication node's neighbours need to fail for a node to fail), and one with communication network failure propagation, which has a threshold of 50% (i.e., at least 50% of the communication node's neighbours need to fail for a node to fail).

Table 5.2: Experiment setup: Network types, centrality measures, initial failure percentages and failure propagation thresholds.

Network Type	Centrality Measure	Initial Failed Nodes (%)	Threshold for Failure Propagation
Mesh	Betweenness	0%, 5%, 10%, 15%, ..., 90%	50%
Mesh	Betweenness	0%, 5%, 10%, 15%, ..., 90%	100%
Mesh	Degree	0%, 5%, 10%, 15%, ..., 90%	50%
Mesh	Degree	0%, 5%, 10%, 15%, ..., 90%	100%
Mesh	Closeness	0%, 5%, 10%, 15%, ..., 90%	50%
Mesh	Closeness	0%, 5%, 10%, 15%, ..., 90%	100%
Mesh	Random	0%, 5%, 10%, 15%, ..., 90%	50%
Mesh	Random	0%, 5%, 10%, 15%, ..., 90%	100%
Double-Star	Betweenness	0%, 5%, 10%, 15%, ..., 90%	50%
Double-Star	Betweenness	0%, 5%, 10%, 15%, ..., 90%	100%
Double-Star	Degree	0%, 5%, 10%, 15%, ..., 90%	50%
Double-Star	Degree	0%, 5%, 10%, 15%, ..., 90%	100%
Double-Star	Closeness	0%, 5%, 10%, 15%, ..., 90%	50%
Double-Star	Closeness	0%, 5%, 10%, 15%, ..., 90%	100%
Double-Star	Random	0%, 5%, 10%, 15%, ..., 90%	50%
Double-Star	Random	0%, 5%, 10%, 15%, ..., 90%	100%

Now that we have explained the logic behind our experiment structure, we provide more details on how the simulation is performed (i.e. how many runs are performed). At each percentage step, we run the model 100 times for each network type. This means that for each percentage step, 100 different double-star and mesh networks are generated. On each of these generated networks, the three different network-based attack strategies and random failures are performed separately. After each run, for each attack strategy, we collect the data needed for the three assessment metrics. Since we conduct 100 runs for every increment of 5% (from 0% to 90%), we perform a total of 1,900 runs for each attack strategy on each type of communication network. This amounts to a total of 15,200 simulations (1,900 multiplied by the number of network types and attack strategies). We perform this for both scenarios: with and without communication network failure propagation, resulting in a total of 30,400 runs (15,200 multiplied by 2). The simulations are run on a laptop (DESKTOP-0GEFQ8F) with an Intel® Core™ i5-8250U CPU @ 1.60GHz, 8.00 GB RAM, and 64-bit Windows 11 Home operating system.

5.2. Hypotheses of the research questions

5.2.1. Hypothesis 1: Communication network structure

The first hypothesis is formulated by looking at the properties of the networks, discussed in §4.3.1. The double-star network has a higher average degree and a lower average clustering coefficient compared to the mesh network (Table 4.2). Since the degree of the nodes in a double-star network varies greatly, there are a small number of nodes with an extremely high degree. Therefore, these nodes are robust against neighbour failures, but when attacked, they have a great impact on the network due to their high degree. The mesh network has fewer critical nodes than the double-star network because its degree distribution does not follow a power law. This could mean that mesh networks are more robust against network-based attacks.

Under the above assumptions, we hypothesise that:

Mesh networks show more robustness under targeted network attacks as compared to double-star networks.

5.2.2. Hypothesis 2: Network-based attack strategy

The second hypothesis concerns which network-based attack strategy is the most effective. Since the three centrality measures (betweenness, degree, and closeness) provide insights into the importance of the communications nodes, we believe that these strategies are more effective than a random attack strategy. Nodes with high degree centrality have a large number of connections to other nodes. Attacking these nodes fragments the communication network, leading to communication nodes not having a path to their corresponding SCADA nodes or exceeding the threshold of failed neighbours since many

nodes connect to these high-degree nodes. This has significant consequences for the communication network and power grid. Nodes with high betweenness centrality often act as critical bridges within the communication network. Attacking these nodes results in other nodes losing their paths to corresponding SCADA nodes, seriously affecting both communication and the power network. Nodes with high closeness centrality have many short paths to other nodes. Attacking these nodes might increase the overall path length between nodes, which is not necessarily disruptive to our network unless the edges in our communication network represent actual data transmission (where time is an important factor).

Based on the assumptions made above, we hypothesise that:

Attack strategies based on degree and betweenness centrality have the largest effect in our SG model, followed by closeness centrality and random attack strategies.

In this chapter, we presented our experimental design to address the research questions. We used double-star and mesh network communication topologies with three centrality-based attack strategies: degree, betweenness, and closeness, along with random failures, to evaluate network robustness, increasing the amount of node selections. We developed three metrics for robustness evaluation: power node fraction survived, total network node survival, and demand survivability. We hypothesised that mesh networks would be more robust against targeted attacks and that degree and betweenness centrality are the most effective. In the following chapter, we discuss the results of the experiments.

6 Analysis

In this chapter, we present a visualisation of one simulation run of our model. We provide the results, discuss them based on the formulated hypotheses, and elaborate on additional insights gained from the results. Finally, we verify our model by comparing the failure propagation steps between the conceptual and operational model.

6.1. Visualisation of a simulation run

Before showing the results of the simulation, we provide four figures to visualise the beginning and end state of a simulation. The beginning of a simulation can be seen in Fig. 6.1 and Fig. 6.2. We perform random failures on a mesh communication network in which 20% of the nodes fail initially. We also visualised the SCADA nodes, which are marked yellow.

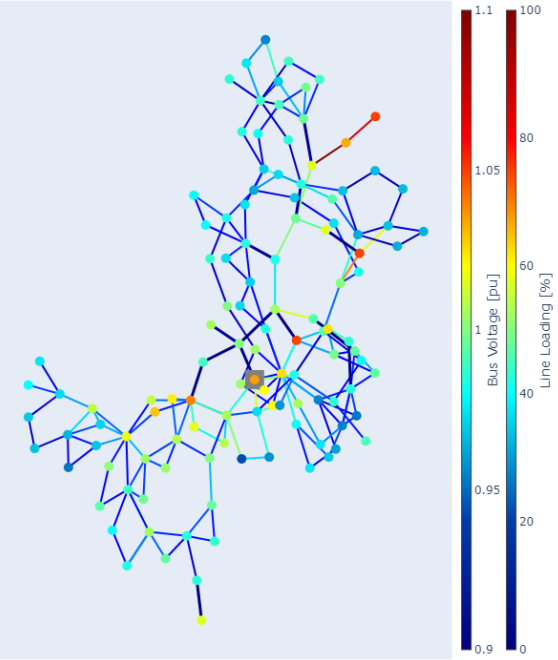


Figure 6.1: Voltage and line loading distribution for the 118-bus test case before the simulation.

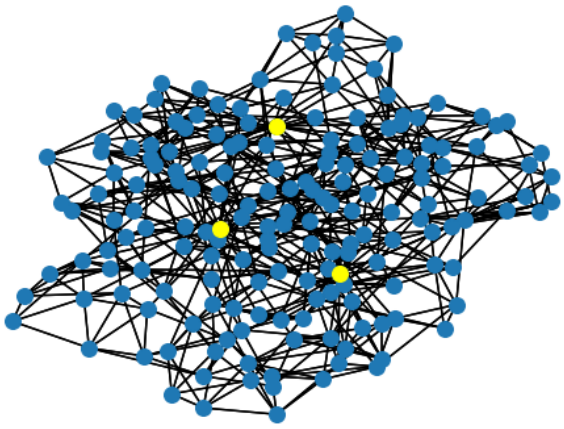


Figure 6.2: Mesh communication network with 176 Nodes including SCADA nodes (yellow) before simulation.

In Fig. 6.3 and Fig. 6.4 we illustrate the end state of the power grid and communication network. Note that certain parts of the power grid are marked in black. This indicates buses that are not connected to any functioning transmission lines or transformers. While more transmission lines in the power grid are out of service, pandapower does not mark them black if they remain connected to a bus that is in

service. We verified the status of these transmission lines using pandapower's `net.res_line` function, verifying that the transmissions corresponding to the failed nodes were indeed out of service. In Fig. 6.4, we visualised the communication network after the simulation. The SCADA nodes are marked yellow, the failed communication nodes are coloured red, and the functioning communication nodes are blue. More than 20% of the communication nodes have failed due to the interdependent failure process behaviour of our model.

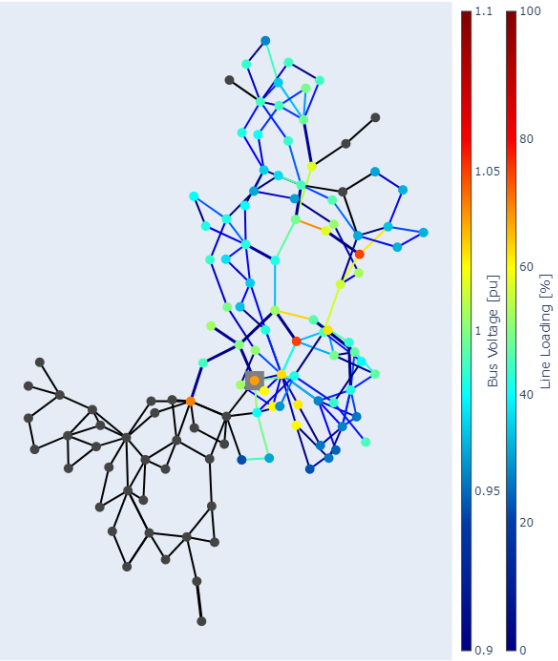


Figure 6.3: Voltage and line loading distribution for the 118-bus test case after the simulation.

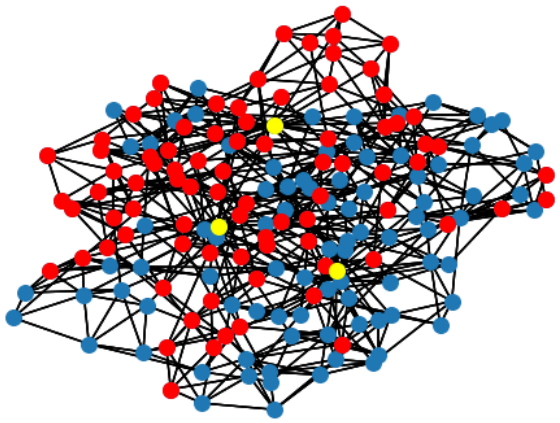
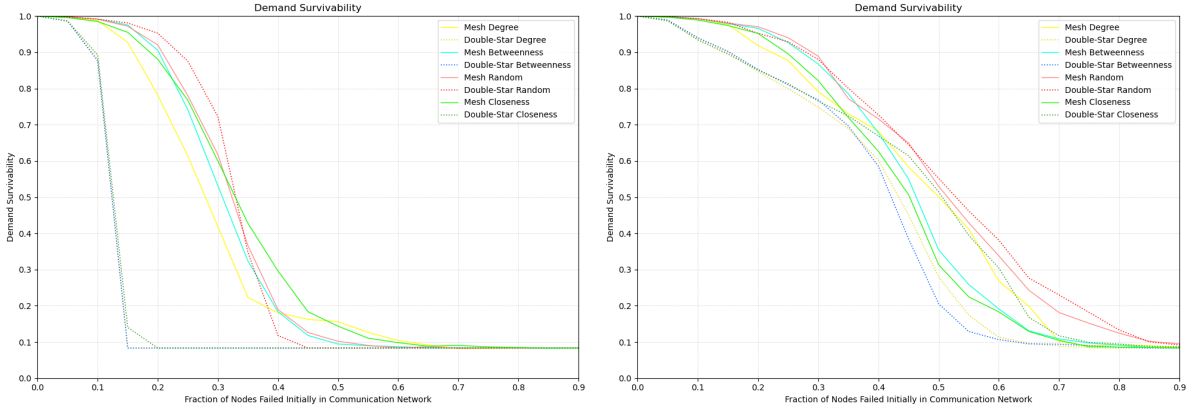


Figure 6.4: Mesh communication network after simulation with 20% random initial failures.

6.2. Analyses of the results of the simulations

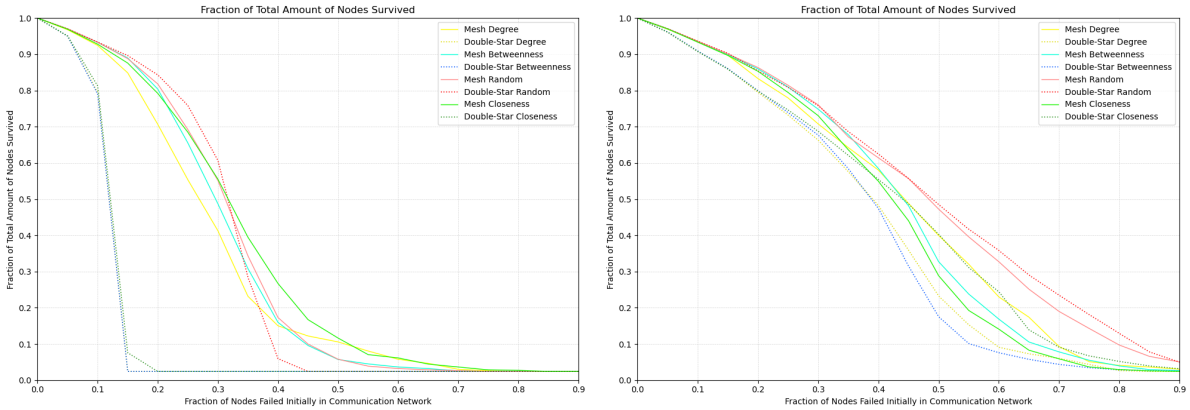
The results are displayed in six separate graphs. Each graph represents an assessment metric formulated in §5.1.2. We categorise each graph based on whether communication network failure propagation is used or not. Fig. 6.5a and 6.5b show the demand survivability, Fig. 6.6a and 6.6b depict the total amount nodes survived in the entire network, and Fig. 6.7a and 6.7b present the fraction of power nodes survived. Each graph contains eight different lines. The line type indicates the network type: solid lines represent mesh networks, while dotted lines represent double-star networks. The colour of the lines denotes the network-based attack strategy: yellow for degree centrality, blue for betweenness centrality, red for random, and green for closeness centrality. Each colour has a light and dark shade to more clearly distinguish between the network types, with light shades for mesh networks and dark shades for double-star networks.



(a) With communication network failure propagation

(b) Without communication network failure propagation

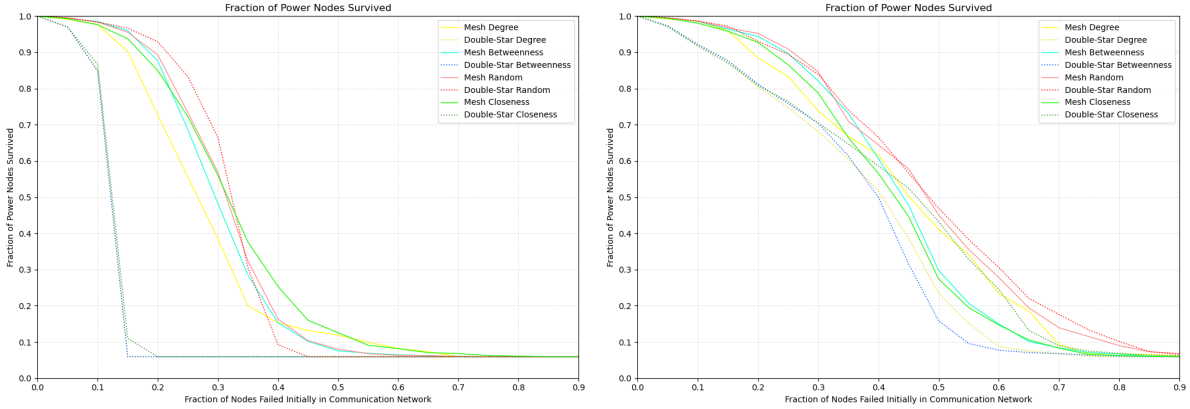
Figure 6.5: Demand survivability (average of 100 simulations per percentage Step). The solid line represents mesh, the dashed line represents double-star, and the colours represent different attack strategies.



(a) With communication network failure propagation

(b) Without communication network failure propagation

Figure 6.6: Fraction of total amount of nodes survived (average of 100 simulations per percentage step). The solid line represents mesh, the dashed line represents double-star, and the colours represent different attack strategies.



(a) With communication network failure propagation

(b) Without communication network failure propagation

Figure 6.7: Fraction of power nodes survived (average of 100 simulations per percentage step). The solid line represents mesh, the dashed line represents double-star, and the colours represent different attack strategies.

6.2.1. Testing Hypothesis 1

We analyse the results of the simulation by comparing it with our formulated hypothesis. Our first hypothesis was formulated as follows:

Mesh networks will show more robustness under targeted network attacks as compared to double-star networks.

Analysis with communiction network failure propagation

For all three metrics, the networks show similar patterns. For the network centrality measures attacks (i.e. excluding random attacks) on double-star communication networks, the SG reaches its end-state after attacking 15% of the communication nodes. The difference between the two types of networks can already be noticed at only 5% of the communication nodes attacked. For the network centrality measure attacks on double-star networks, the decline steepens from 5% to 10% and declines even steeper from 10% to 15% of the communication nodes attacked, after which its end state is reached.

The lines of the mesh network centrality measure attacks (excluding randomness) are all on the right-hand side of the double-star network centrality measure attacks on all three metrics. This indicates a higher robustness for the mesh communication networks.

Although the double-star networks perform the worst, this is not the case for random attacks. Under random attacks, an SG has a higher robustness with a double-star communication network. However, after around 37% to 39% of initial failures in the communication network, the double-star network under random failures performs worse than all other mesh communication networks on all three metrics.

As our hypothesis suggested, an SG with a double-star communication network is less robust than a mesh communication network under network centrality measure attacks. This is due to the degree distribution of a double-star network, which follows a power-law distribution, meaning there are a few critical nodes with an extremely high degree. When these nodes are attacked, the failure propagates rapidly throughout the network. Under a random attack, these critical nodes might not initially fail and serve as robust nodes due to their high degree (making them less susceptible to the neighbor fail criteria or no path to corresponding SCADA nodes). Therefore, our results show that they are in line with the formulated hypothesis.

Analysis without communication network failure propagation

For an SG that does not exhibit communication failure propagation, the results are closer than without failure propagation. This indicates that a double-star network is extremely sensitive to failure propagation in the communication network under targeted attacks. Across all three metrics, initially, the double-star network shows lower robustness under targeted attacks. Up until around 35% of the initial communication nodes fail, the lines representing targeted attacks on double-star networks remain below those of the mesh network, indicating lower robustness. Only after around 35% of initial communication nodes have failed does the mesh network show lower robustness against closeness centrality attacks than the double-star network.

Under random failures, the robustness of both networks is quite similar. However, after around 45% - 50% of initial communication node failures, the double-star network line stays above the mesh network line in all three metrics, suggesting higher robustness.

In general, our results are in line with the formulated hypothesis without failure propagation. The graphs indicate that a mesh network is more robust against targeted attacks, especially with regard to degree and betweenness centrality attacks. The mesh network also initially shows higher robustness against closeness centrality attacks, but this no longer holds after around 35%. Therefore, our results show that, for the most part, they are in line with the formulated hypothesis.

Comparing the robustness of communication networks: With vs. without communication network failure propagation

Both networks are significantly less robust when communication network failure propagation is considered. All lines in the graph with failure propagation have steeper slopes, indicating a sharper decline in robustness. This is also evident by looking at when the end-state is reached. With failure propagation, this end state is reached more rapidly, particularly in double-star networks, compared to scenarios without failure propagation.

Our results indicate that, with or without failure propagation, an SG with a double-star communication network is less robust against degree and betweenness centrality attacks. Without failure propagation, the double-star structure is significantly more robust against closeness centrality attacks. Whereas, mesh networks are more robust against degree attacks without failure propagation compared to other attack types (beyond a certain threshold). Furthermore, under random failures with failure propagation, the double-star network becomes less robust than mesh networks after a certain threshold, while without failure propagation, a double-star network becomes more robust than mesh networks after a certain threshold.

6.2.2. Testing hypothesis 2

We analyse the results by comparing it with the second hypothesis , The second hypthothesis was formulated as follows:

Attack strategies based on degree and betweenness centrality will have the largest effect in our SG model, followed by closeness centrality and random attack strategies.

Analysis with communication network failure propagation

In double-star networks, centrality measure attacks are the most effective. Within these networks, failures based on degree, betweenness, and closeness centrality are almost identical. Only between 5% and 10%, and between 15% and 20% of initial failures in the communication network, are the closeness centrality attacks slightly less effective than betweenness and degree centrality attacks.

In mesh networks, there are larger differences between attack types. Up until 40% of initial failed communication nodes, degree centrality failures are the most effective. After 40%, degree centrality attacks become less effective than random or betweenness centrality failures. Degree centrality attacks show similar effectiveness as closeness centrality after around 50% of initial communication node failures. Betweenness and closeness centrality attacks on mesh networks show similar effectiveness; however, around 22% of initial failures, the betweenness centrality attack becomes more effective than closeness centrality. At around 40%, betweenness centrality failures become more effective than degree centrality failures. Disregarding random attacks on mesh communication networks, closeness centrality is the least effective attack strategy. Only after around 50% does this strategy follow a similar trend as degree centrality.

Random attacks on double-star networks are less effective than the other network-based attacks. In fact, up until around 33% of initial communication node failures, it is the least effective strategy for both double-star and mesh network communication networks. This changes after around 40% of initial communication node failures, where it becomes more effective than all other failure types in mesh communication networks. Random failures in mesh networks show less effectiveness than other centrality measure attacks on this type of network. After around 25% - 30% of initial communication node failures, random attacks become more effective than closeness centrality failures. At around 45%, random attacks on mesh networks become almost as effective as betweenness centrality attacks. Random attacks become more effective than degree centrality attacks on mesh networks after 40% - 43% of initial failures in the communication network.

Our second hypothesis applies to double-star networks. Attacking this type of network using centrality measures quickly fails critical nodes, thereby reaching the criteria of failed neighbors or no path to SCADA nodes more quickly. In contrast, with random attacks, these critical nodes only fail when a larger portion of the communication nodes initially fail (there is a larger chance that these nodes will be selected as attacked nodes).

In mesh networks, the hypothesis does not hold. Closeness centrality is the least effective, while degree centrality is the most effective up until a certain point. After this point (40%), degree centrality becomes the least effective and follows the same trend as closeness centrality. Around this point, betweenness centrality and random attacks become more effective than degree centrality. This is attributed to the high interconnectivity of mesh networks. Failures based on closeness centrality decrease the effectiveness of disrupting the SG as it focuses on nodes that are nearby other nodes on average, rather than the nodes that are critical for the overall network connectively. Even when nodes with high closeness centrality are set as failed, the communication network remains connected (and therefore the criteria of failed neighbors or no path to SCADA nodes is reached less quickly). Initially attacking high-degree nodes has a large impact on the SG, as they are connected to many nodes and therefore

serve as key connectors. However, since the degree distribution is not as extreme as in double-star networks, increasingly attacking nodes based on degree centrality becomes less effective. Betweenness centrality becomes more effective as it targets crucial bridges within the network. With random attacks, the chances increase that nodes with high degree or betweenness centrality are also chosen to fail. Therefore the effectiveness of these attack strategies surpasses degree centrality at a certain point.

Analysis without communication network failure propagation

From Fig. 6.5b, 6.6b and 6.7b, it is clear that random failures in both mesh and double-star networks have the least impact on the robustness of the SG.

An SG with a double-star communication network is impacted the most by betweenness centrality attacks and degree centrality attacks. Although these two attack strategies follow each other closely, after around 35% of initial communication node failures, the betweenness centrality has a larger effect on robustness compared to degree centrality. Closeness centrality follows betweenness and degree centrality closely in the beginning. However, it becomes clear after around 30% that the closeness centrality has less of an effect on robustness. Yet, it does have a greater impact than random failures.

For an SG with a mesh communication network, the effectiveness of network-based attacks varies more than with a double-star network. Initially, the degree centrality attack has the most impact on robustness. However, after around 35% of initial failed communication nodes, closeness centrality becomes the most effective attack strategy. Betweenness centrality attacks become more effective than those based on degree centrality after around 40% of failed communication nodes. However, betweenness centrality does not become equally or more effective than closeness centrality.

Without communication network failure propagation, our results show that randomness has the least impact on the robustness of an SG. In the case of targeted attacks, closeness centrality has the least impact on double-star communication networks. However, it becomes the most impactful on mesh communication networks after 35% of initial communication nodes have failed, although degree centrality has the most impact before this threshold. Regarding our second hypothesis, our results show it be partially true. An SG with a double-star network is most impacted by betweenness and degree centrality attacks, while closeness centrality has less impact on robustness and random failures have the least. Our second hypothesis is in line with our results considering double-star networks. However, for mesh networks our simulations show different results than the formulated hypothesis. Closeness centrality has a greater impact on the robustness of an SG than betweenness centrality and surpasses degree centrality after a certain threshold.

Comparing attack strategies and failures: With vs. without communication network failure propagation

From our results, it is clear that without failure propagation, network-based attacks have a greater impact on robustness than random failures. This is especially true for double-star networks, which exhibit extreme vulnerability to network-based attacks with failure propagation. However, with failure propagation, random failures in mesh communication networks have a larger impact than closeness and degree centrality after a certain threshold.

Regarding network-based attacks with failure propagation, degree and betweenness centrality have the largest impact on both double-star and mesh networks. Without failure propagation, closeness centrality is the most effective strategy for attacking mesh communication networks and is the least effective strategy for attacking double-star networks. Additionally, closeness centrality is the least effective network-based attack strategy on mesh networks when considering failure propagation, which is in contrast to the results without failure propagation. Furthermore, without failure propagation, betweenness centrality attacks become more impactful than degree centrality attacks after a certain threshold on both double-star and mesh communication networks.

6.2.3. Other insights: Variability of robustness under random failures

It is interesting to see how the robustness of an SG, given a mesh and double-star network, varies under random failures. Under this type of failure, random nodes are selected to initially fail, which could be important nodes, unimportant nodes, or a mixture of different ratios. This introduces a certain unpredictability, which we show both with and without communication network failure propagation. In

Appendix C, we include two tables (C.1 and C.2) that show the averages and standard deviations of all three performance metrics, with and without failure propagation under random attacks. We plot the standard deviations in Fig. 6.8 and 6.9. Within these figures, red is used for the fraction of the total amount of nodes survived, blue for the fraction of power nodes survived, and green for demand survivability. For the mesh topology, we have chosen a darker shade than for double-star networks to make a clear distinction. Regarding the definition of metrics in the tables and figures, FPN refers to the fraction of power nodes that survived, FTA represents the fraction of the total amount of nodes that survived, and DS stands for demand survivability.

With communication network failure propagation

From Fig. 6.5a, 6.6a and 6.7a, it becomes apparent that random attacks on double-star networks are significantly less effective than the other attack strategies on double-star networks. Due to the power-law degree distribution of double-star networks, there are many nodes with a low degree. When only a small number of initial failed nodes are selected randomly, the chances are higher that a low-degree node will be selected compared to the lower amount of high-degree nodes. This makes the random failure strategy on a double-star communication network less effective. However, the more nodes selected randomly as failed, the higher the chance critical nodes are selected, and therefore the effectiveness increases.

Overall, double-star networks are less robust to all types of attacks. However, when considering random attacks, it is one of the most robust networks up until a certain point, in our case, around 35% of initially failed communication nodes, where it becomes worse than mesh communication networks.

Although the random failure strategy is less effective in double-star networks, this is not the case in mesh communication networks. It closely follows the closeness centrality strategy, after which it becomes more effective and closely follows the betweenness centrality trend. The random strategy attack is the only attack strategy whose lines intersect in the graph (for mesh and double-star networks). At around 35%, random attacks are more effective on double-star than on mesh communication networks.

From Fig. 6.8, we can see that the standard deviation is nearly zero at the beginning of the simulation (at 0% and 5%) and at the end of the simulation (45% for double-star network and around 75% for mesh network). For both mesh and double-star networks, the standard deviation increases as the percentage of initial failed communication nodes increases. This indicates less predictable network behavior under an increase of attacked nodes (for random attacks). Around 15%-20%, for both networks, a significant increase in the standard deviation can be noticed, reaching its peak at around 35%, after which it drops significantly again. This indicates that for both networks, between 25% and 35%, the largest variability occurs, translating to less predictable behavior of the network.

Even though the two type of networks show a similar pattern regarding standard deviation, there are two major differences. Firstly, double-star networks generally show higher standard deviations compared to mesh networks. This implies that network performance is more variable and therefore less predictable in double-star networks under random failure scenarios. Secondly, although double-star networks generally have a higher standard deviation, this stops abruptly at 45% of initial failures where it reaches a stand deviation of zero on all three metrics. In contrast, mesh networks still exhibit a low amount of standard deviation until 75% for all three metrics.

There are several reasons why double-star networks show higher standard deviations and therefore greater variability under random attacks compared to mesh networks. Double-star networks have a few critical nodes, with an extremely high degree, that play a vital role in maintaining network robustness. At around 25%-40% of initial failures, these critical nodes in double-star networks might or might not fail, introducing great uncertainty. The failure of these critical nodes can cause a large failure propagation, while their survival allows them to act as robust nodes. However, until a certain degree (around 45%), these critical nodes fail more frequently, leading to less variability and therefore less unpredictability compared to mesh networks. Since mesh networks have fewer extreme critical nodes than double-star networks, they exhibit more predictable behavior and thus have lower standard deviations. Mesh networks still show some degree of variability up until 75% of initial failures because the communication nodes that fail might still create a considerable impact on the networks performance. The reason for this is that in mesh networks, the importance of each node is more evenly distributed than in double-star networks. This implies that even with a high percentage of node failures, the remaining nodes can still affect the network's robustness when they might eventually fail.

From Fig. 6.8, we can also see that mesh networks show lower and longer-tailed standard deviations, indicating more consistent robustness under random attacks. It is also evident that after the 35% mark, the double-star network quickly degrades, while the mesh network still shows variability, suggesting that the critical nodes in the double-star network quickly fail after this tipping point.

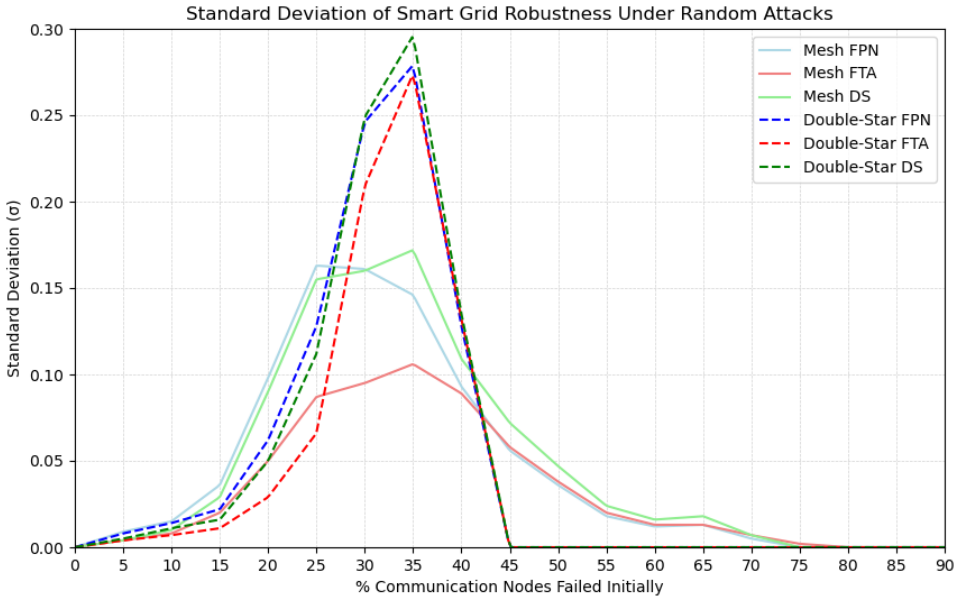


Figure 6.8: Standard deviation of smart grid robustness under random attacks with communication network failure propagation. The solid line represents mesh, the dashed line represents double-star, and the colours represent different robustness metrics.

Without communication network failure propagation

In Fig. 6.9, the standard deviations of the three metrics of our model without communication network failure propagation are plotted. Both the standard deviations of the double-star and mesh networks exhibit similar behaviour. Although there is not a sharp peak in the standard deviations, the highest point is reached around 35% to 45%. This indicates that the largest variability occurs around this point, suggesting lower predictability of the behaviour of the model.

Although for each metric, both networks are similar in standard deviations, the fraction of total amount of nodes survived has a much lower peak and a flatter line. This suggests lower and more consistent variability of the results. This can also be seen in Fig. 6.6b, where the lines of random failures show a less steep decline in performance compared to Fig. 6.5a and 6.7b. Without failure propagation in the communication network, fewer communication nodes fail rapidly. Since this performance metric also measures the number of communication nodes that fail, it shows less variability, resulting in a lower and flatter standard deviation.

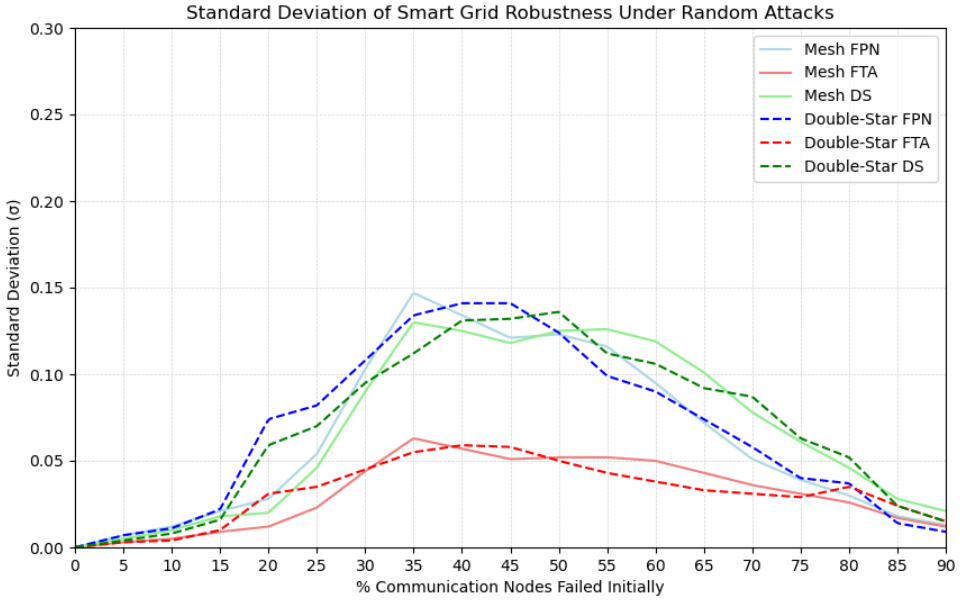
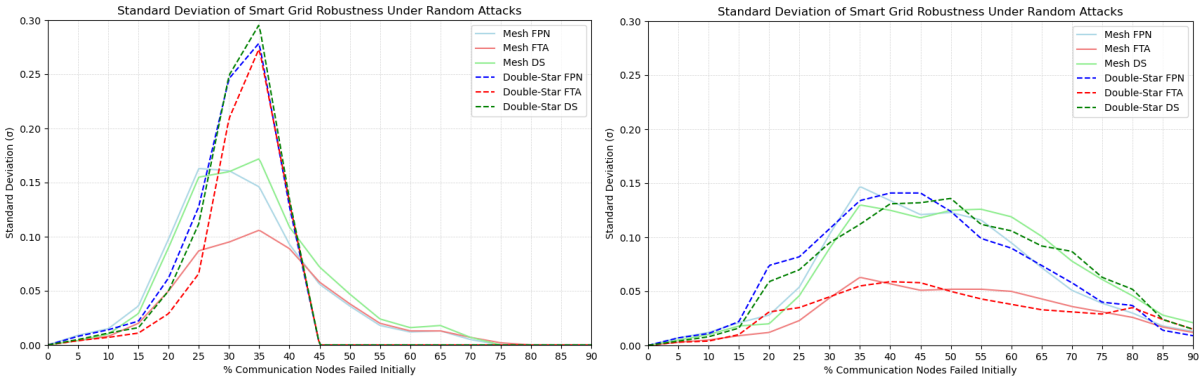


Figure 6.9: Standard deviation of smart grid robustness under random attacks without communication network failure propagation. The solid line represents mesh, the dashed line represents double-star, and the colours represent different robustness metrics.

Comparing standard deviations: With vs. without communication network failure propagation
We compare the standard deviations of the results with and without communication network failure propagation by examining Fig. 6.10. From this figure, it is evident that the curvature of the standard deviations without failure propagation is lower and longer-tailed. This suggests a lower and more consistent variability in the results. With failure propagation, the double-star network exhibits high peaks, particularly when compared to the network without failure propagation. The reason for this is that under random attacks, critical nodes within the double-star network may fail initially or later in the process, and if so, propagate widely through the communication network, creating high variability in the results. However, without failure propagation, the failure of critical nodes in a double-star network does not necessarily lead directly to vast failures within the network and therefore exhibits lower variability in the performance metrics. Although the peaks are sharper in Fig. 6.10a than in Fig. 6.10b, the peak curvatures occur at similar points, around 25%-35% and 35%-45% respectively. However, the peaks without failure propagation are slightly shifted to the right, indicating that more communication nodes need to fail initially to achieve the highest variability in the results compared to those with failure propagation. This occurs because, with failure propagation, higher peaks form earlier as there is a likelihood that critical communication nodes fail initially or during the process. Consequently, the failure spreads throughout the network, creating higher variability. Furthermore, from both figures, it is clear that under random attacks, communication network failure propagation leads to higher variability in double-star networks, while disabling this network failure propagation significantly reduces this variability. The reduction is to such an extent that the double-star communication network shows standard deviations comparable to that of mesh communication networks under random attacks.



(a) With communication network failure propagation

(b) Without communication network failure propagation

Figure 6.10: Comparison of standard deviation of smart grid robustness under random attacks with and without communication network failure propagation. The solid line represents mesh, the dashed line represents double-star, and the colours represent different robustness metrics.

6.3. Verification of the model

During the building of our SG model, each code snippet was tested separately to ensure it performed as expected and could be added to our model. We verify the conceptual and operational model by looking at the failure process steps. To verify the communication network generation and the power grid setup, we refer to §4.3.1, §4.3.2 and §4.2.3.

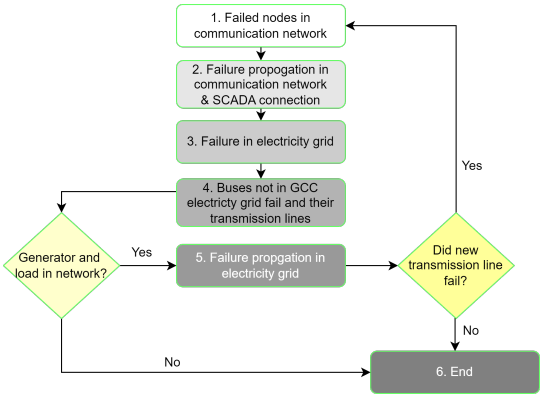


Figure 6.11: Failure process steps in the conceptual model to compare with the operational steps.

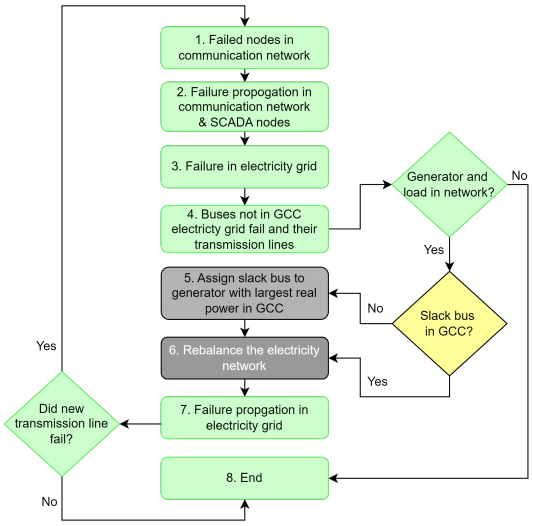


Figure 6.12: Failure process steps in the operational model to compare with the conceptual steps.

In Fig. 6.11, we can see the different failure process steps of the conceptual model. The steps are highlighted in green to show that these steps are also included in our operational model failure process steps, which can be seen in Fig. 6.12. The conceptual model failure process consists of six steps with two decision-making moments. It is illustrated that these steps are also included in the operational model failure process steps. However, in the operational model, two extra steps and one extra decision-making moment are included to ensure that the simulations work. The purpose of Fig. 6.11 and Fig. 6.12 is to show that all conceptual steps are included in our model, thereby verifying the failure propagation steps of our model.

In this chapter, we presented and analysed the results generated by the simulation. The analysis, with failure propagation, aligns with our first hypothesis that mesh communication networks are more robust under targeted attacks compared to double-star networks. However, under random attacks, the

robustness of double-star networks is higher initially but quickly degrades after a certain threshold. Our results, without failure propagation, are mostly in line with the first hypothesis as well. However, beyond a certain threshold, double-star communication networks demonstrate greater robustness than mesh networks with regard to closeness centrality attacks. Our second hypothesis, that degree centrality and betweenness centrality attacks are the most effective, is supported for double-star networks but not directly for mesh networks, with failure propagation. In mesh networks, the effectiveness of attack strategies varies, with degree centrality attacks being the most effective up to a point, after which betweenness and random attacks become more effective. Without failure propagation, the results are partially in line with the second hypothesis, as beyond a certain threshold, closeness centrality becomes the most effective attack strategy on mesh networks. Furthermore, we showed, via a standard deviation analysis, that both network types exhibit the highest variability around the same point. However, mesh networks display lower and longer-tailed standard deviations, indicating more consistent robustness under random attacks with failure propagation. Without failure propagation, the standard deviations for both networks exhibit similar behaviour, and the variability is much lower and more constant than with failure propagation. Finally, we verified our model by comparing the failure process steps between the conceptual and operational models, ensuring that all vital steps from the conceptual model are also present in the operational model.

7 Discussion

In this chapter, we discuss the results generated by our model, apply it in the context of the real world and provide recommendations. We elaborate on certain real-world trade-offs between the mesh and double-star communication networks. Furthermore, we discuss and compare our performance metrics with respect to other literature. Lastly, we elaborate on the verification, validation, assumptions, and simplifications of our model and research, allowing us to be transparent about the limitations of our approach.

7.1. Recommendations: Real-world interpretation of the results

In our study, we modelled both mesh and double-star networks to assess the impact of different attack strategies on the robustness of the interdependent communication network and power grid, with and without communication network failure propagation. We used these network types as they resemble real-world communication networks [62, 52, 54, 105]. Our model is a high-level abstraction of a real-world SG, and although certain simplifications and assumptions were made, the results can be applied to real-world contexts.

Our research revealed that for both communication network structures, with failure propagation in the communication network, the SG is significantly less robust than without failure propagation. It is therefore recommended to prioritise reducing the spread of propagation failures within the communication network, as this significantly increases the robustness of the SG. This applies to both double-star and mesh communication networks. For double-star communication networks, in particular, a reduction in failure propagation leads to higher robustness.

Since double-star networks are very sensitive to targeted attacks, it is advisable to focus on the few highly critical components in such networks, which are the components with the highest degree and betweenness centrality. Our results indicate that these network-based attacks have the most significant impact on double-star networks, both with and without failure propagation. However, without failure propagation, components within a double-star communication network could be prioritised using only betweenness centrality. While both centralities show similar impacts, betweenness centrality has a slightly higher impact on robustness after a certain threshold. Since betweenness and degree centrality have similar effects with failure propagation, this prioritisation strategy applies to double-star communication networks regardless of failure propagation.

For mesh communication networks, several recommendations can reduce the impact on the robustness of the SG. Firstly, reducing communication network failure propagation leads to a less severe impact on the robustness of the SG, similar to double-star communication networks. Secondly, network components with a high degree should be prioritised within a mesh network, especially when failure propagation can occur. Our results show that this network-based attack has the most significant impact before a certain threshold, even without failure propagation. However, if failure propagation can be reduced in a mesh network, components with high closeness centrality should also be considered to minimise the impact on the SG's robustness.

To reduce the propagation of communication network failures, such as the spread of malware

throughout the network, several measures can be implemented. The authors in [111] examine the spread of malware in SGs and propose a range of strategies to address this issue. The first measure is network segmentation. Firewalls or virtual networks are often used as a measure to contain malware propagation in communication networks [111]. Another measure to reduce the spread and therefore failure propagation is the use anomaly detection systems [111]. It is recommended to implement these detection systems to identify and respond to suspicious activities before malware can spread further. These systems can detect unusual network traffic patterns or behaviours that could indicate the presence of malware.

There are several measures to increase the robustness or reliability of the critical communication network components. Firstly, redundancy techniques can be employed. If critical components have failed or are attacked, redundancy (hardware, software, or both) can ensure that these failures do not affect the entire system by providing backup resources that can take over the operations of the failed components. Secondly, the critical components should be of high quality. Using high-quality components reduces the likelihood of failures and ensures the reliability of these components. Thirdly, proactive maintenance of these critical communication network components should be in place to identify and handle potential problems before they can lead to failures. Additionally, it is important to monitor these components. With the use of real-time monitoring, the status of the critical components can be assessed, and detection mechanisms can be employed to identify failures or potential failures before they occur.

Limiting propagation behaviour in the communication network mitigates the impact of random failures. The robustness under random attacks without failure propagation shows lower variability, indicating a more stable and predictable behaviour. However, with failure propagation, random failures can have a similar impact to targeted attacks after a certain threshold, particularly in mesh communication networks. A communication network is composed of various types of equipment for connectivity and functionality. Although these components are generally reliable, failures can still occur. These failures may happen randomly or may affect specific types of equipment or equipment from a particular vendor (non-random failures). This consideration is crucial when assessing the robustness of a SG with a double-star or mesh topology. If random failures occur on a large scale, their consequences can be as severe as those of targeted attacks, especially in mesh networks that exhibit failure propagation behaviour. In double-star networks, random failures do not have as rapid an impact as targeted attacks, however, after a certain threshold, they can significantly affect the SGs robustness. These large random failures could be reduced by lowering the likelihood of correlated failures (i.e. parts of the network supplied by the same vendor or using the same type of equipment). Therefore, it is recommended that communication network designers and power grid operators diversify their vendor sources to reduce failure correlations and conduct regular network audits to identify and strengthen critical nodes. Additionally, these measures can help reduce device monocultures, thereby limiting the spread of malware and limiting communication network failures [111].

We have provided recommendations for reducing the impact on the robustness of a SG for each communication network structure: mesh and double-star. However, when comparing both networks, it is recommended to employ a mesh communication network structure over a double-star network. A power grid with a mesh communication network shows higher robustness under targeted attacks, both with and without failure propagation behaviour, and exhibits similar behaviour under random attacks.

7.2. Double-star vs. mesh networks: Real-world tradeoffs

In the previous section, we elaborated that a SG shows an overall higher robustness with a mesh communication network. However, even though mesh communication networks show higher overall robustness, trade-offs need to be considered when comparing a double-star and mesh communication network in the context of the real world.

Firstly, scalability is important. With the increase in distributed and renewable energy sources, the energy grid is expanding [112]. This implies that the communication network needs to expand as well to monitor and control the grid. With double-star communication networks, scalability is not a significant issue as adding nodes primarily involves connecting them to highly connected nodes (hubs). For

mesh networks, this becomes more complex as a node in the network needs high connectivity to other nodes to maintain the level of connection and clustering. Additionally, current technologies need to be considered when configuring the structure of the communication network. In Home Area Networks, it is easier to establish mesh communication network structures, especially with the Zigbee network protocol. With this type of network, multiple short-distance devices (e.g., smart sensors) can connect with each other. This is also possible with other connection technologies such as Wi-Fi and Bluetooth. However, within WANs and FANs, the distances between communication devices become greater, and other communication technologies are needed, such as 4G, 5G, or physical connections like fibre optics. Establishing a mesh communication network structure in these domains is challenging because, although technologies exist to cover the distances, it is more costly and requires higher power consumption. Double-star networks are more easily constructed in these domains, as not all components require high-degree connections. Double-star networks are more easily constructed in these domains, as most network components in this structure are connected to only several high-degree components while still covering long distances.

The second point concerns the connectivity of the network and its relationship to data efficiency, power consumption, and associated costs. Due to the high clustering coefficient and short average path lengths of mesh communication networks, they facilitate efficient communication and quick data transmission across the network. However, even with the short average path lengths and a high number of connections, if not managed properly, it can lead to congestion and increased latency. Because of the hierarchical structure of double-star communication networks, data transmission might be slower but it simplifies the routing process, as data can be simply routed through central hubs. As data routing in double-star is more simple and there are many nodes with low-degrees, many components in this communication network structure requires fewer resources such as computational power and energy consumption. In contrast, in mesh networks, each node has a substantially high degree. As an example, sensors operated with batteries on transmission lines may drain their batteries more rapidly due to the increased communication load. This means that on average, each node in the mesh network requires a higher power supply and computational capabilities while in double-star networks this is only the case for the hub nodes.

Lastly, the complexity of network segmentation is an important factor. In the previous section, we mention network segmentation as a possible measure to reduce the propagation of communication network failures. Within double-star communication networks, network segmentation might be less complex because the hubs in these networks serve as connectors to other parts of the network. Therefore, the network can be easily segmented by focusing on these hubs. This simplifies the process of isolating faulty segments and preventing failure propagation, making double-star networks advantageous in this regard. In mesh communication networks, segmentation is more complicated. Due to the high average cluster coefficient and more evenly distributed degree of nodes, it is more challenging to isolate sections of the network to reduce failure propagation. This complexity requires more sophisticated algorithms and tools for effective management of network segmentation.

7.3. Performance Metrics used in the Literature

Comparing our performance metrics, as highlighted in section §5.1.2, with those found in the literature helps to understand the strengths and limitations of our results. Our metrics have similarities with those that are described in the literature. In [52, 59], they examine their model by looking at the overall damage with an increase in target nodes and the average accumulated damage with the number of failing power nodes. Although damage might be defined slightly differently, it does show resemblances with our three metrics and graphs (in §6.2), which can also be described as the damage that occurred in the SG with increasing targeted nodes. Comparable metrics were also found in [55], which looks at the percentage of functional nodes with the number of attacked nodes. Additionally, the authors also used the amount of power still supplied with the number of initially attacked nodes, which shows resemblance with our demand survivability metric. A similar metric was also used in [60], however, the authors described it as the power survival rate. In [54], they developed a metric named 'average blackout size', which is similar to our total amount of nodes (both in the power grid and communication network) that survived.

However, the literature also includes other metrics or uses the same metrics presented differently in graphs due to using different data on the x-axis or y-axis. In [59], they looked at the vulnerability

index for each node in the power grid (node ID). This gives more insight into which specific node in the grid is the most vulnerable to failures or attacks. In our model and simulations, each run generates a different communication network, making it challenging to incorporate such a metric. In [54], they used the 'average blackout size' metric, which we described previously. However, the authors did not use an increase in target attacks but an increase in the tolerance parameter (a higher tolerance means a higher overcurrent capacity of the transmission line [54]). This approach is also interesting as they connected this tolerance parameter with economic costs, allowing them to derive the costs of making a SG more robust. This metric could also be included in our model to provide economic insights into increasing robustness of the SG. Another interesting way of presenting demand survivability is by examining it at each stage of the cascading process rather than by the number of attacked nodes, which is done in [113]. This approach provides great insights into the impact of each step in the failure propagation and more clearly illustrates how failures in one component lead to subsequent failures in others.

7.4. Verification and validation

In §6.3, we verified the model by comparing the failure propagation steps of the conceptual model to those of the operational model. From Fig. 6.11 and Fig. 6.12, we can observe that all the steps in the conceptual model are included in the operational model as well.

However, a limitation of our research is the validation of the model. It is challenging to compare our results to real-world data due to the limited availability of communication network data [61, 52] and the sparsity of literature in this specific research direction (i.e., focusing solely on the communication network within a coupled power grid system). Nonetheless, some literature in the same field supports the results of our model.

In [62], the authors show that a positive correlation was found between clustering coefficient and system robustness. This aligns with our findings, which show that a mesh communication network, with a higher clustering coefficient than a double-star network, is more robust against targeted attacks and, to a certain extent, random failures. A similar finding was reported in [54], where the authors concluded that a mesh communication network is the most robust in a coupled cyber-physical power system.

In [56], the authors conclude that a double-star network is more vulnerable to communication network failure propagation than a mesh network, which is also in line with our results.

Even though certain aspects of our results show similarities with other research, the validity of our model can be further increased. One suggestion is to conduct interviews with experts in both the power system and communication network domains to validate both the results and the model we built (e.g., the assumptions and parameters). As mentioned earlier, communication network data is limited, making data comparison difficult. Therefore, it is necessary to consult with experts in this field for the validation of our results.

7.5. Limitations, assumptions and simplifications of the model

7.5.1. The conceptual model

During the conceptualisation phase of our research, we made several important assumptions to simplify and therefore increase the understandability of our model. Assumptions on the communication network and the relationship between the power grid and communication network had to be made due to the limited availability of information on communication networks of power grid [61, 52].

One of the key assumptions concerns the structure of the SG. We assume there is a one-to-one interdependency between a communication node and a transmission line in the power grid. This assumption is threefold. Firstly, it assumes that there is a dependency between a communication node and a transmission line. This dependency might also exist between a communication node and a bus. However, to keep the model understandable, we chose to link it with transmission lines, as the removal of a transmission line is not as impactful as removing an entire bus. Secondly, it assumes a one-to-one dependency, which might not always be the case. However, in similar research, this dependency has been assumed, as seen in [58, 53, 74, 76, 83]. Thirdly, it assumes that the networks are interdependent, although this might not always be true. However, this type of relationship is frequently applied in the literature (see aspect A in the literature analyses in Table 2.2).

Another key assumption of our conceptual model is the connection between the communication

and SCADA nodes. In our model, we assume that each communication node corresponds to a specific SCADA node, but not all communication nodes are directly connected to a SCADA node.

In the communication network failure propagation process, the diffusion process was quite simple. A communication node could fail if its direct neighbouring node failed (e.g., due to malware spread), once a certain threshold was exceeded. Additionally, when a communication node failed in this diffusion process, it could not recover. This behaviour is simpler than other processes such as the SIR (Susceptible-Infected-Recovered) model, where a node can be susceptible, infected, and can recover.

Another simplification is that only power nodes in the GCC are considered. If the power nodes are no longer part of the GCC they are regarded as failed.

7.5.2. The operational model

The operational phase of our research was also subjected to certain assumptions and simplifications. The first assumption concerns the network parameters for the communication networks and the setup for the power grid. The network parameters are described in §4.3.1. The choice of these network parameters impacts the degree distribution, average degree, and clustering coefficient of the communication network. For the mesh network, we selected a rewiring probability of 0.2, as this was also used in [61].

An important simplification we made is that SCADA nodes are only connected to 25% of the corresponding communication nodes. These choices might impact the failure process (i.e. if a communication node has a path to its corresponding SCADA node) as different values make it either easier or more difficult for a node to fail. However, due to the low availability of data on communication networks in power grids, it was necessary to make these assumptions. Furthermore, another operational simplification we made is that each communication node is connected by a transmission line corresponding to the same number (node ID corresponds to transmission line index) and not by geographic positions. This means that a communication node can be connected to another communication node even if the transmission lines are geographically further apart from a network perspective.

For the operational communication network failure propagation behaviour, several assumptions and simplifications were made. Firstly, we set the failure criterion for communication node neighbours to 0.5, meaning that at least 50% of a communication node's direct neighbours must fail for the node itself to fail. Increasing this parameter would result in a more robust network, whereas decreasing it would lead to a less robust communication network. We chose to set the parameter at 0.5 to compare the results with the parameter being set at 1.0 (i.e., no communication network failure propagation based on failed neighbours).

Regarding rebalancing the network, we assumed that the real power of generators is adjusted based on the load demand. This adjustment is made according to the real power generation that each in-service generator provides, relative to their contribution to the total maximum in-service generation capacity. It is important to note that we adjust only the active power and not the reactive power in the network for simplicity. Since a PFA is a complex process, it is difficult to predict the exact impact the rebalancing assumption will have on the robustness of the SG.

8

Conclusion

This research aims to understand the effect of failures in the communication network on the underlying power grid. We identify a knowledge gap, from which we formulate the main research question, divided into two sub-questions. In this chapter, we provide a conclusion for our research questions. Additionally, we elaborate on the scientific and societal contributions of our research. Finally, we discuss several directions for future research.

8.1. Answering the research sub-questions

In §3.1, we formulated two research sub-questions. The first RSQ was formulated as:

Which communication topology is the most robust against failures, causing the least disruption to the functionality of the underlying power grid?

Between a double-star (scale-free) and a mesh (small-world) communication network, it is clear that a mesh network is more robust against targeted network attacks with communication network failure propagation. Across all three robustness performance metrics we developed for evaluating an increasing number of initially failed communication nodes, there is a noticeable gap favouring mesh networks under targeted attacks.

However, under random failures, the robustness of the communication networks presents more complexity. Our experiments, considering all three performance metrics, demonstrated that double-star networks show higher robustness from 10% to 35% of initially randomly failed communication nodes. On the other hand, double-star networks show significantly higher variability in performance around 30% compared to mesh networks, indicating less predictable outcomes. As a result, while within a certain range, double-star networks demonstrate higher robustness, this is coupled with greater variability and a sharp decline in robustness after 35% of initially randomly failed nodes.

Without communication network failure propagation, the mesh network shows higher robustness under targeted attacks. However, after 35% of initial failures, a double-star network shows higher robustness against closeness centrality attacks. Under random attacks, both networks show similar behaviour, but after 45%-50% of random initial failures, an SG with a double-star communication network shows higher robustness. In comparison, with communication network failure propagation, both communication network types show lower and more stable variability under random attacks.

Therefore, in the majority of cases concerning targeted attacks and in some cases for random failures, mesh communication networks exhibit higher robustness than double-star networks regarding the performance of the underlying power grid.

The second RSQ was formulated as:

Which network-based attack strategy on the communication network is the most effective in causing the most disruption to the functionality of the underlying power grid?

Firstly, in double-star networks, the overall low robustness of this network type makes targeted attacks similarly effective, with communication network failure propagation. Only when approximately 15% of communication nodes are targeted does closeness centrality show slightly lower effectiveness compared to degree and betweenness centrality. For this type of network, it is evident that random failure is the least effective, joining the other types of attacks at 45% of failed communication nodes. Without communication network failure propagation, degree and betweenness centrality attacks are the most effective regarding double-star communication networks. For up to 35% of initial failures, degree centrality attacks are more effective. However, after 35% of initial failures, betweenness centrality attacks become more effective than degree centrality attacks.

Within mesh communication networks, this difference is less distinguishable. With communication network failure propagation, it appears that degree centrality is the most effective until about 40%-45%, after which betweenness centrality and random failures become more effective. Although closeness centrality shows slightly more effectiveness than betweenness centrality and random failures up to 25%, this difference is minimal, and its effectiveness becomes significantly less than betweenness attacks and random failures after 25%.

However, without failure propagation, degree centrality has the most impact up until around 35% of initial failures, after which closeness centrality becomes the most impactful, followed by betweenness centrality.

Interestingly, with failure propagation, the random failures in double-star and mesh communication networks intersect around 35%, indicating a shift in effectiveness between the two types of random failures across the different network structures. However, as previously mentioned, variability also plays a role here.

To conclude, determining which network-based attack strategy is the most effective is challenging due to the different network characteristics and the variation in behaviour with and without communication network failure propagation. However, our results indicate that in double-star networks, degree centrality and betweenness centrality attacks are equally effective. In mesh networks, degree, betweenness, and closeness centrality can all be the most effective strategy, depending on the behaviour with and without failure propagation and the number of initial communication nodes that fail.

8.2. Linking back to the main research question

After having answered each RSQ individually, we can combine the insights gained to answer the MRQ, which is formulated as:

What is the effect of failures in the communication network on the underlying power grid?

Even though the implementation of SG technologies can significantly reduce carbon emissions [7], which, as a result, enhances sustainability efforts, the increasing interdependence between the power grid and communication network increases the attack surface for cybercriminal organisations [8]. This vulnerability results in substantial real-world impacts, especially now with the rising geopolitical tensions. For instance, the cyberattack on a Ukrainian power station in 2015 during the Russo-Ukrainian war causes a massive outage, affecting hundreds of thousands of citizens [9]. However, job posting data from major power utilities in the United States suggest a lack of long-term strategy in the past regarding cyberattack incidents [8].

Our research demonstrates that failures or attacks in the communication network significantly impact the underlying power grid, as shown by all three performance metrics in our results. Even a small increase in failed or attacked nodes in the communication network results in a degradation in the power grid's performance. This is due to the power grid becoming increasingly intertwined and interdependent on the communication network. However, we have demonstrated that the effects of failures in the communication network differ depending on the communication network structure, network-based attack performed and the failure propagation behaviour in the communication network. In general, mesh networks offer higher robustness, making them more suitable to withstand targeted attacks and random failures, thereby causing less disruption to the underlying power grid. Double-star communication networks show higher robustness to random failures within a range of 10% to 35% of initially failed

communication nodes with failure propagation and after 45% without failure propagation. However, under random failures with failure propagation, this network type suffers from higher variability and a steep decline in performance as the percentage of initial failed nodes increases. Furthermore, targeting communication network components based on degree and betweenness centrality proves to be the most effective strategy for both network types. In a mesh communication network without failure propagation, once the percentage of initially attacked communication nodes reaches 35%, closeness centrality has a greater impact on the power grid's robustness.

Understanding this behaviour and these dynamics is crucial for developing long-term strategies, which are currently lacking, to enhance the robustness of power grids against communication network failures or attacks.

8.3. Scientific contribution

In §2.3, we identified a knowledge gap based on literature research. We evaluated the literature based on twelve aspects (Table 2.1) and showed that none of the papers analysed fulfilled all the aspects aspects as aspects F to L were either missing or not strongly represented (Table 2.2). Based on this, we formulated the knowledge gap as:

The knowledge gap that needs to be addressed is understanding the effect of communication network failures on the underlying interdependent power grid from a network perspective

From this knowledge gap, we formulated our MRQ. In our research we considered aspects included in most of the analysed literature, such as modeling the interdependency between the communication network and power grid (e.g. [52, 53, 54]), incorporating heterogeneous power grid components (e.g. [55, 58, 61]), having heterogeneous communication network components (e.g. [59, 62]), using a realistic power grid including electrical properties, such as in [57, 58] (in our case, the IEEE 118-bus test system), simulating failure or attack scenarios (e.g. [52, 60] and the performance metrics as well (see §7.3).

However, in §2.3, we provided a description of each piece of analysed literature and highlighted how our research differs from it. We narrowed the knowledge gap by concentrating on the missing or underrepresented aspects (F to L) by focusing on failures occurring initially in the communication network, examining different communication network structures, and performing and assessing various network-based strategies (including random failures) on these structures. Additionally, we compared the results of the simulations with and without failure propagation in the communication network.

The main key findings of our research, different from the analysed literature, are the following:

- The difference in robustness of a SG with a mesh and double-star communication network, comparing scenarios with and without communication network failure propagation.
- The impact of random failures on both communication network types, including the variability of the results under these failures.
- The importance of closeness centrality in mesh communication networks without failure propagation.
- Demonstrating that at certain thresholds, specific attack types become more effective on particular networks by gradually increasing the number of failed nodes.

To further illustrate our scientific contributions, we compare our research to the most similar studies identified in the literature review and highlight the key differences.

In [53], the authors study the vulnerability of nodes and connectivity of systems in a cascading failure by proposing a partial random coupling systems model (between a communication network and power grid). This research is similar to our research as the authors also use double-star and mesh communication networks on an IEEE 118-bus system. Furthermore, they also explore different attack strategies including random, degree, and betweenness centrality. The difference is that our IEEE 118-bus system is distinct in having heterogeneous power nodes, whereas the authors in [53] assume that the power grid has abundantly distributed generators. This means that our power grid model is more realistic, as it does not have an abundance of distributed generators and incorporates other components

such as transformers. Furthermore, our communication network nodes have heterogeneous roles by including SCADA nodes, which the authors do not include in their research. Although the authors use various network-based attack strategies, our research also includes closeness centrality attacks and provides insights into the variability of random failures. The addition of closeness centrality is important, as we show that it has a significant impact on mesh communication networks with failure propagation. Additionally, we demonstrate the difference in robustness of the SG with and without failure propagation in the communication network. Our research also directly compares the robustness of different communication network structures and elaborates on the effectiveness of different network attack strategies.

In [54], the authors introduce a stochastic cascading failure model of an interdependent communication network and power grid with heterogeneous nodes. The authors also generate different communication networks such as rewired, random, mesh, and double-star networks. It is interesting that the authors added two additional network types to compare the robustness of different network types. Although the authors focus on the robustness of different communication network topologies, they also emphasise the interdependency connections between the communication network and the power grid, which is different from our research. In our research, we initialise failures solely in the communication network to show the effect of communication network failures on the underlying power grid, whereas the authors in [54] do not. Additionally, unlike this work, we analyse the most effective attack strategy given the different communication network structures and failure propagation behaviours.

The authors in [56] use a network-based virus propagation model to investigate cascading failures in an interdependent SG. They use two different communication network types, scale-free (double-star) and small-world (mesh) structures. The authors employ a more complex communication network failure propagation model than we do, which heightens the realism of the behaviour. They infect vertices in the network based on random selection and degree centrality. However, our research incorporates a simpler communication network failure propagation model to demonstrate the difference in robustness compared to scenarios without failure propagation in the communication network. Additionally, we initialise failed nodes not only based on random selection and degree centrality but also on betweenness and closeness centrality, providing additional insights into the effectiveness of various attack strategies.

In [63], the authors propose a cascading failure model of a cyber-physical system in a virtual power plant. They conduct various attacks, including random and targeted attacks based on degree, betweenness, eigenvector, and information centrality. It is interesting that the authors included eigenvector and information centrality as additional attack strategies. Additionally, these authors initially fail nodes in the cyber network. They show that the mesh network is the most robust against random and targeted attacks. Although the authors perform various targeted attacks, they do not compare which type of targeted attack has the most significant effect on the robustness of the SG, which we do in our research. In our research, we show that the robustness of the SG, with different communication networks, depends on certain thresholds and attack types. Furthermore, we also show the robustness of different communication network types given different failure propagation behaviours in the communication network. Additionally, the authors do not incorporate electrical properties in their power grid model, which we have included in our model.

In [75], the authors model cascading failures using a diffusion model to simulate communication network failure propagation through malware. They employ various network-based attack strategies, such as random selection and degree centrality (both low and high). This is an interesting addition, as it includes attacking nodes with low centrality characteristics, which we also explore. The authors incrementally increase the number of infected nodes and analyse the robustness of a double-star and a mesh communication network. They show that high-degree attacks cause more damage than the other two attack strategies. However, under low-degree attacks, the double-star communication network shows higher robustness than a mesh structure. The difference with our research is that we examine the robustness of the SG with and without communication network failure propagation. Additionally, under these different attack scenarios, we also include betweenness centrality and closeness centrality attacks, highlighting the importance of these attack strategies beyond just degree and random failures. Furthermore, we show the variability of the SG's results under random attacks, indicating that with failure propagation, a double-star network exhibits larger and less consistent variability.

With our contribution, we further enhance the understanding of the effect of failures in the communication network on the underlying power grid, which answers the MRQ and narrows the knowledge gap

we identified.The findings of this research contribute to the existing literature by exclusively focusing on failures within the communication network, analysing various attack strategies with different proportions of initially failing nodes within an interdependent power grid and communication network with heterogeneous node roles. And also comparing this using two different failure behaviours: with and without failure propagation in the communication network. To the best of our knowledge, this approach has not previously been explored.

8.4. Societal contribution

Besides providing scientific input, this research also offers a societal contribution. Not only is the power grid becoming more intertwined with digital technologies, but this trend can be seen across many critical infrastructures such as transport, healthcare, water, and more [114]. In recent years, the number and complexity of cyber assets have been increasing rapidly due to the ongoing digital transformation of the operators of these critical infrastructures [114]. With this research, we aim to demonstrate that the increasing interdependency between the power grid and communication network introduces new challenges that must be addressed. The growing interconnection between the physical system (in our case, the power grid) and the cyber system (in our case, the communication network) can lead to severe outages if failures occur in the communication network. This can already be clearly observed in the real world, such as in the Russo-Ukrainian war [9].

This research does not directly provide information on how such incidents can be prevented. However, we offer insights into the vulnerabilities of the SG and provide recommendations on how to address these vulnerabilities in §7.1. With our findings, we urge communication network engineers and cybersecurity specialists to consider how the digital infrastructure should be designed and identify the components within these networks that have the most significant impact in case of attacks or failures. This research contributes to this area by demonstrating the impact on the power grid under various communication network structures, network-based attack strategies, random failures and communication network failure propagation behaviours. Our findings identify which communication network structure is the most robust and which attack strategy is the most effective, thereby highlighting the most critical type of communication network components from a network perspective. Furthermore, we also urge collaboration between communication network experts and power grid engineers, as these networks are increasingly becoming intertwined, and vulnerabilities exposed in one network can lead to failure propagation and, ultimately, severe outages in the other network. This overlap highlights the necessity for these areas to work closely together to ensure the robustness and functionality of the SG.

8.5. Recommendation for future works

Having described the findings and limitations of our research and compared them to the literature, we can provide several recommendations for future research.

The first direction concerns the improvement of the communication network model. This involves adjusting parameters, such as the average degree of the communication networks, neighbour nodes fail criteria and more, so that they more closely represent real-world conditions based on either literature or expert knowledge. Furthermore, the structure of the communication network can be adjusted to more accurately resemble a real-world network. Currently, our communication network is dual-layered, with communication nodes directly connected to transmission lines and SCADA nodes. However, in reality, there are multiple layers between these two layers which can be added for future research. The communication network can be improved by incorporating real-world communication network functionality. Currently, our communication network is mainly focused on network theory aspects, where edges between the communication nodes represent dependencies. To increase the realism of this network, data packet transmission could be added. A failure in the network would occur if a data packet containing power grid operational instructions is delayed or contains false information. This would, however, make the model significantly more complex.

The second direction concerns the interdependency between the communication network and the power grid. Our model assumes a direct failure of a transmission line if a communication node fails, and vice versa. However, this might not necessarily be the case. The uncertainty of whether a failure directly occurs can be modeled using stochastic elements. Furthermore, each communication node is connected to one transmission line (one-to-one dependency), which might not accurately represent

the real-world interdependency ratio.

The third direction for future research touches on the power grid. We formulated several behavioural rules for the power grid to make it functional for PFA. This includes the rebalancing of the grid and ensuring that every generator is active, which might not reflect real-world power grid behaviour. Additionally, we set up the power grid in such a way that the average load is higher. Future research could incorporate different levels of loading in the power grid to understand the impact of failures at various loading levels.

The fourth point for future research concerns communication network failure propagation. In our model, this is represented in a simplistic manner, but it could be expanded upon by incorporating more realistic elements, such as those used in an SIR model.

Lastly, future research could build upon our experiments. Additional network-based attack strategies could be performed, such as targeting nodes based on eigenvector centrality or those with the most connections to the power grid (if there is no one-to-one dependency). Furthermore, extra communication network topologies can be tested to assess the overall robustness of the SG, including networks with geographical positioning similar to the underlying power grid. Additionally, research can be conducted on the robustness of the SG given different neighbor fail criteria, which is currently set at 50% for all our experiments. Also, an economical parameter can be added to gain insights in the costs of increasing the SG's robustness, similar to what was done in [54].

References

[1] International Energy Agency. *Climate change - Energy sector is central to efforts to combat climate change*. URL: https://www.iea.org/topics/climate-change (visited on 01/22/2024).

[2] D. Faquir, N. Chouliaras, V. Sofia, K. Olga, and L. Maglaras. "Cybersecurity in smart grids, challenges and solutions". In: *AIMS Electronics and Electrical Engineering* 5.1 (Jan. 2021), pp. 24–37. DOI: 10.3934/electreng.2021002. URL: https://www.aimspress.com/article/doi/10.3934/electreng.2021002?viewType=HTML.

[3] International Energy Agency. *An energy sector roadmap to carbon neutrality in China*. Sept. 2021. URL: https://www.iea.org/reports/an-energy-sector-roadmap-to-carbon-neutrality-in-china/executive-summary (visited on 04/04/2024).

[4] International Energy Agency. *Electricity 2024 - Executive summary*. Jan. 2024. URL: https://www.iea.org/reports/electricity-2024/executive-summary (visited on 04/06/2024).

[5] J. Liu, Y. Xiao, S. Li, W. Liang, and C. L. P. Chen. "Cyber security and privacy issues in smart grids". In: *IEEE Communications Surveys and Tutorials* 14.4 (Jan. 2012), pp. 981–997. DOI: 10.1109/surv.2011.122111.00145. URL: https://doi.org/10.1109/surv.2011.122111.00145.

[6] International Energy Agency. *Smart grids*. URL: https://www.iea.org/energy-system/electricity/smart-grids (visited on 01/22/2024).

[7] H. Fu, Y. Shi, and Y. Zeng. "Estimating smart Grid's carbon emission reduction potential in China's manufacturing industry based on decomposition analysis". In: *Frontiers in Energy Research* 9 (June 2021). DOI: 10.3389/fenrg.2021.681244. URL: https://doi.org/10.3389/fenrg.2021.681244.

[8] International Energy Agency, M. Casanovas, and A. Nghiem. *Cybersecurity – is the power system lagging behind?* Aug. 1, 2023. URL: https://www.iea.org/commentaries/cybersecurity-is-the-power-system-lagging-behind (visited on 04/10/2024).

[9] *Compromise of a power grid in eastern Ukraine*. Dec. 2015. URL: https://www.cfr.org/cyber-operations/compromise-power-grid-eastern-ukraine.

[10] K. Proska, J. Wolfram, J. Wilson, D. Black, K. Lunden, D. K. Zafra, N. Brubaker, T. Mclellan, and C. Sistrunk. *Sandworm disrupts power in Ukraine using a novel attack against operational technology*. Nov. 9, 2023. URL: https://www.mandiant.com/resources/blog/sandworm-disrupts-power-ukraine-operational-technology (visited on 04/10/2024).

[11] Dragos, Inc. *CRASHOVERRIDE: Analyzing the Malware that Attacks Power Grids*. June 12, 2017. URL: https://www.dragos.com/resources/whitepaper/crashoverride-analyzing-the-malware-that-attacks-power-grids/ (visited on 04/10/2024).

[12] Cyber Law Toolkit. *Industroyer – Crash Override (2016)*. June 4, 2021. URL: https://cyberlaw.ccdcoe.org/wiki/Industroyer_%E2%80%93_Crash_Override_(2016) (visited on 04/10/2024).

[13] BBC. *Ukraine power cut 'was cyber-attack'*. Jan. 11, 2017. URL: https://www.bbc.com/news/technology-38573074 (visited on 04/10/2024).

[14] X. Yu and Y. Xue. "Smart Grids: A Cyber–Physical Systems perspective". In: *Proceedings of the IEEE* 104.5 (May 2016), pp. 1058–1070. DOI: 10.1109/jproc.2015.2503119. URL: https://doi.org/10.1109/jproc.2015.2503119.

[15] G. Dileep. "A survey on smart grid technologies and applications". In: *Renewable Energy* 146 (Feb. 2020), pp. 2589–2625. DOI: 10.1016/j.renene.2019.08.092. URL: https://doi.org/10.1016/j.renene.2019.08.092.

[16] United States Department of Energy. *2020 Smart Grid System Report*. Tech. rep. May 2022.

[17] İ. Çolak. “Introduction to smart grid”. In: *2016 International Smart Grid Workshop and Certificate Program (ISGWCP)* (Mar. 2016), pp. 1–5. DOI: 10.1109/isgwcp.2016.7548265. URL: https://doi.org/10.1109/isgwcp.2016.7548265.

[18] A. B. Kanase and S. R. Gengaje. “Smart Grid Technology an Overview”. In: *International Journal of Innovations in Engineering and Technology (IJIET)* 7.2 (Aug. 2016), pp. 517–522.

[19] G. W. Arnold, D. A. Wollman, G. J. FitzPatrick, D. E. Prochaska, D. G. Holmberg, D. Su, A. R. Hefner, N. Golmie, T. L. Brewer, M. Bello, and P. A. Boynton. “NIST Framework and Roadmap for Smart Grid Interoperability Standards, Release 1.0”. In: *National Institute of Standards and Technology* (Jan. 2010). DOI: 10.6028/NIST.sp.1108. URL: https://doi.org/10.6028/NIST.sp.1108.

[20] M. Z. Gündüz and R. Daş. “Cyber-security on smart grid: Threats and potential solutions”. In: *Computer Networks* 169 (Mar. 2020), p. 107094. DOI: 10.1016/j.comnet.2019.107094. URL: https://doi.org/10.1016/j.comnet.2019.107094.

[21] R. K. Pandey and M. Misra. “Cyber security threats — Smart grid infrastructure”. In: *2016 National Power Systems Conference* (Dec. 2016). DOI: 10.1109/npsc.2016.7858950. URL: https://doi.org/10.1109/npsc.2016.7858950.

[22] K. Kimani, V. K. Oduol, and K. Langat. “Cyber security challenges for IoT-based smart grid networks”. In: *International Journal of Critical Infrastructure Protection* 25 (June 2019), pp. 36–49. DOI: 10.1016/j.ijcip.2019.01.001. URL: https://doi.org/10.1016/j.ijcip.2019.01.001.

[23] Z. E. Mrabet, N. Kaabouch, and H. E. Ghazi. “Cyber-security in smart grid: Survey and challenges”. In: *Computers & Electrical Engineering* 67 (Apr. 2018), pp. 469–482. DOI: 10.1016/j.compeleceng.2018.01.015. URL: https://doi.org/10.1016/j.compeleceng.2018.01.015.

[24] W. Wang, Y. Xu, and M. Khanna. “A survey on the communication architectures in smart grid”. In: *Computer Networks* 55.15 (Oct. 2011), pp. 3604–3629. DOI: 10.1016/j.comnet.2011.07.010. URL: https://doi.org/10.1016/j.comnet.2011.07.010.

[25] *What are high-voltage lines?* Nov. 14, 2023. URL: https://www.bfs.de/EN/topics/emf/expansion-grid/basics/intro/introduction.html (visited on 04/15/2024).

[26] *What is a substation?* Jan. 23, 2024. URL: https://www.nationalgrid.com/stories/energy-explained/what-is-a-substation (visited on 04/15/2024).

[27] K. Korepova. *The generation, transmission and distribution of electricity*. Aug. 16, 2022. URL: https://epeconsulting.com/the-generation-transmission-and-distribution-of-electricity/ (visited on 04/15/2024).

[28] *Circuit Breaker Fundamentals*. URL: https://www.eaton.com/us/en-us/products/electrical-circuit-protection/circuit-breakers/circuit-breakers-fundamentals.html (visited on 04/15/2024).

[29] *Circuit breakers on Onelines*. URL: https://www.powerworld.com/WebHelp/Content/MainDocumentation_HTML/Circuit_Breakers_on_Onelines.htm (visited on 04/15/2024).

[30] Liyond Electric Company. *What is the difference between Relay and Circuit Breaker?* Dec. 3, 2021. URL: https://www.elecspare.com/what-is-the-difference-between-relay-and-circuit-breaker/ (visited on 04/15/2024).

[31] F. E. Abrahamsen, Y. Ai, and M. Cheffena. “Communication Technologies for Smart Grid: A Comprehensive survey”. In: *Sensors* 21.23 (Dec. 2021), p. 8087. DOI: 10.3390/s21238087. URL: https://doi.org/10.3390/s21238087.

[32] D. Syed, A. Zainab, A. Ghrayeb, S. S. Refaat, H. Abu□Rub, and O. Bouhali. “Smart Grid Big Data Analytics: Survey of technologies, techniques, and applications”. In: *IEEE Access* 9 (Jan. 2021), pp. 59564–59585. DOI: 10.1109/access.2020.3041178. URL: https://doi.org/10.1109/access.2020.3041178.

[33] M. K. Hasan, A. Habib, S. Islam, N. Safie, S. N. H. S. Abdullah, and B. Pandey. “DDoS: Distributed denial of service attack in communication standard vulnerabilities in smart grid applications and cyber security with recent developments”. In: *Energy Reports* 9 (Oct. 2023), pp. 1318–1326. DOI: 10.1016/j.egyr.2023.05.184. URL: https://doi.org/10.1016/j.egyr.2023.05.184.

[34] NIST. *Cyber Attack - Glossary*. URL: https://csrc.nist.gov/glossary/term/cyber_attack (visited on 03/19/2024).

[35] CISCO. *What is a cyberattack?* Feb. 21, 2024. URL: https://www.cisco.com/c/en/us/products/security/common-cyberattacks.html (visited on 03/19/2024).

[36] IBM. *What is a Cyberattack?* URL: https://www.ibm.com/topics/cyber-attack (visited on 03/19/2024).

[37] S. Samonas and D. L. Coss. “The CIA strikes back: redefining confidentiality, integrity and availability in security”. In: *Journal of Information System Security* 10.3 (Jan. 2014), pp. 21–45. URL: https://www.jissec.org/Contents/V10/N3/V10N3-Samonas.html.

[38] *Confidentiality, Integrity, and availability: The CIA triad | Office of Information Security | Washington University in St. Louis*. URL: https://informationsecurity.wustl.edu/items/confidentiality-integrity-and-availability-the-cia-triad/ (visited on 03/20/2024).

[39] W. Wang and Z. Lu. “Cyber security in the Smart Grid: Survey and challenges”. In: *Computer Networks* 57.5 (Apr. 2013), pp. 1344–1371. DOI: 10.1016/j.comnet.2012.12.017. URL: https://doi.org/10.1016/j.comnet.2012.12.017.

[40] S. Shapsough, F. Qatan, R. Aburukba, F. Aloul, and A. Ali. “Smart grid cyber security: Challenges and solutions”. In: *International Conference on Smart Grid and Clean Energy Technologies* (Oct. 2015). DOI: 10.1109/icsgce.2015.7454291. URL: https://doi.org/10.1109/icsgce.2015.7454291.

[41] S. Qadir and S. M. K. Quadri. “Information Availability: An Insight into the Most Important Attribute of Information Security”. In: *Journal of Information Security* 07.03 (Jan. 2016), pp. 185–194. DOI: 10.4236/jis.2016.73014. URL: https://doi.org/10.4236/jis.2016.73014.

[42] M. A. Ferrag and A. Ahmim. *Security solutions and applied cryptography in smart grid communications*. Jan. 2017. DOI: 10.4018/978-1-5225-1829-7. URL: https://doi.org/10.4018/978-1-5225-1829-7.

[43] L. Xu, X. Liang, R. Lu, X. Shen, X. Lin, and H. Zhu. “Securing smart grid: cyber attacks, countermeasures, and challenges”. In: *IEEE Communications Magazine* 50.8 (Aug. 2012), pp. 38–45. DOI: 10.1109/mcom.2012.6257525. URL: https://doi.org/10.1109/mcom.2012.6257525.

[44] K. Baker. *What is a Trojan Horse? Trojan Malware Explained - CrowdStrike*. June 17, 2022. URL: https://www.crowdstrike.com/cybersecurity-101/malware/trojans/ (visited on 04/23/2024).

[45] I. Aouini and L. B. Azzouz. “Smart grids cyber security issues and challenges”. In: *World Academy of Science, Engineering and Technology, International Journal of Computer and Information Engineering* 2.11 (Nov. 2015). URL: https://waset.org/abstracts/computer-and-information-engineering/35303.

[46] C. Bekara. “Security issues and challenges for the IoT-based smart grid”. In: *Procedia Computer Science* 34 (Jan. 2014), pp. 532–537. DOI: 10.1016/j.procs.2014.07.064. URL: https://doi.org/10.1016/j.procs.2014.07.064.

[47] *DoS Attack vs. DDoS Attack: Key Differences?* URL: https://www.fortinet.com/resources/cyberglossary/dos-vs-ddos (visited on 04/23/2024).

[48] S. Tufail, I. Parvez, S. Batool, and A. I. Sarwat. “A survey on cybersecurity challenges, detection, and mitigation techniques for the smart grid”. In: *Energies* 14.18 (Sept. 2021), p. 5894. DOI: 10.3390/en14185894. URL: https://doi.org/10.3390/en14185894.

[49] D. Jin, D. M. Nicol, and G. Y. Guanhua. “An event buffer flooding attack in DNP3 controlled SCADA systems”. In: *Proceedings of the 2011 Winter Simulation Conference* (Dec. 2011), pp. 2619–2631. DOI: 10.5555/2431518.2431832. URL: http://dx.doi.org/10.1109/WSC.2011.6147969.

[50] *SQL injection*. URL: https://owasp.org/www-community/attacks/SQL_Injection (visited on 04/23/2024).

[51] M. A. Siddiqi and N. Ghani. “Critical analysis on advanced persistent threats”. In: *International Journal of Computer Applications* 141.13 (May 2016), pp. 46–50. DOI: 10.5120/ijca2016909784. URL: https://doi.org/10.5120/ijca2016909784.

[52] R. Atat, M. Ismail, S. S. Refaat, E. Serpedin, and T. J. Overbye. “Cascading failure vulnerability analysis in interdependent power communication networks”. In: *IEEE Systems Journal* 16.3 (Sept. 2022), pp. 3500–3511. DOI: 10.1109/jsyst.2021.3128698. URL: https://doi.org/10.1109/jsyst.2021.3128698.

[53] L. Chen, D. Yue, C. Dou, J. Chen, and Z. Cheng. “Evaluation of cyber-physical power systems in cascading failure: node vulnerability and systems connectivity”. In: *IET Generation Transmission & Distribution* 14.7 (Feb. 2020), pp. 1197–1206. DOI: 10.1049/iet-gtd.2019.1286. URL: https://doi.org/10.1049/iet-gtd.2019.1286.

[54] H. Guo, S. Yu, H. H.-C. Iu, T. Fernando, and C. Zheng. “A complex network theory analytical approach to power system cascading failure—From a cyber-physical perspective”. In: *Chaos: An Interdisciplinary Journal of Nonlinear Science* 29.5 (May 2019). DOI: 10.1063/1.5092629. URL: https://doi.org/10.1063/1.5092629.

[55] A. Salehpour, I. Al-Anbagi, K. C. Yow, and X. Cheng. “Modeling cascading failures in coupled smart grid networks”. In: *IEEE Access* 10 (Jan. 2022), pp. 81054–81070. DOI: 10.1109/access.2022.3194989. URL: https://doi.org/10.1109/access.2022.3194989.

[56] T. Wang, X. Wei, T. Huang, J. Wang, L. Valencia-Cabrera, Z. Fan, and M. J. Pérez-Jiménez. “Cascading Failures analysis considering extreme virus propagation of Cyber-Physical systems in smart grids”. In: *Complexity* 2019 (Mar. 2019), pp. 1–15. DOI: 10.1155/2019/7428458. URL: https://doi.org/10.1155/2019/7428458.

[57] T. Wang, Q. Long, X. Gu, and W. Chai. “Information flow modeling and performance evaluation of communication networks serving power grids”. In: *IEEE Access* 8 (Jan. 2020), pp. 13735–13747. DOI: 10.1109/access.2020.2966489. URL: https://doi.org/10.1109/access.2020.2966489.

[58] H. Pan, X. Li, C. Na, and R. Cao. “Modeling and analysis of cascading failures in Cyber-Physical power systems under different coupling strategies”. In: *IEEE Access* 10 (Jan. 2022), pp. 108684–108696. DOI: 10.1109/access.2022.3213332. URL: https://doi.org/10.1109/access.2022.3213332.

[59] R. Atat, M. Ismail, and E. Serpedin. “Limiting the failure impact of interdependent Power-Communication networks via optimal partitioning”. In: *IEEE Transactions on Smart Grid* 14.1 (Jan. 2023), pp. 732–745. DOI: 10.1109/tsg.2022.3188648. URL: https://doi.org/10.1109/tsg.2022.3188648.

[60] X. Zhang, D. Liu, H. Tu, and C. K. Tse. “An integrated modeling framework for cascading failure study and robustness assessment of cyber-coupled power grids”. In: *Reliability Engineering & Systems Safety* 226 (Oct. 2022), p. 108654. DOI: 10.1016/j.ress.2022.108654. URL: https://doi.org/10.1016/j.ress.2022.108654.

[61] G. Wu, M. Li, and Z. S. Li. “A stochastic modeling approach for cascading failures in cyberphysical power systems”. In: *IEEE Systems Journal* 16.1 (Mar. 2022), pp. 723–734. DOI: 10.1109/jsyst.2021.3070503. URL: https://doi.org/10.1109/jsyst.2021.3070503.

[62] L. Chen, Y. Sun, C. Dou, H. Ge, Z. Cheng, and S. Li. “Modeling two-stage failure mechanism of cascading in cyber-physical power systems”. In: *Physica Scripta* 98.9 (Aug. 2023), p. 095209. DOI: 10.1088/1402-4896/aceac5. URL: https://doi.org/10.1088/1402-4896/aceac5.

[63] X. Gao, X. Li, and X. Yang. “Robustness assessment of the cyber□physical system against cascading failure in a virtual power plant based on complex network theory”. In: *International Transactions on Electrical Energy Systems* 31.11 (Aug. 2021). DOI: 10.1002/2050-7038.13039. URL: https://doi.org/10.1002/2050-7038.13039.

[64] L. Li, Z. Li, Z. Gong, H. Liu, Y. Li, and Y. Chen. “Dual hidden failure model for Cyber Physical power System”. In: *IEEE Access* 8 (Jan. 2020), pp. 186148–186156. DOI: 10.1109/access.2020.3028933. URL: https://doi.org/10.1109/access.2020.3028933.

[65] G. Zhang, J. Shi, S. Huang, J. Wang, and H. Jiang. “A cascading failure model considering operation characteristics of the communication layer”. In: *IEEE Access* 9 (Jan. 2021), pp. 9493–9504. DOI: 10.1109/access.2021.3049485. URL: https://doi.org/10.1109/access.2021.3049485.

[66] L. Lee and P. Hu. “Vulnerability analysis of cascading dynamics in smart grids under load redistribution attacks”. In: *International Journal of Electrical Power & Energy Systems* 111 (Oct. 2019), pp. 182–190. DOI: 10.1016/j.ijepes.2019.03.062. URL: https://doi.org/10.1016/j.ijepes.2019.03.062.

[67] M. Alonso, J. Turanzas, H. Amaris, and A. T. Ledo. “Cyber-Physical vulnerability assessment in smart grids based on multilayer complex networks”. In: *Sensors* 21.17 (Aug. 2021), p. 5826. DOI: 10.3390/s21175826. URL: https://doi.org/10.3390/s21175826.

[68] W. Kang, Q. Liu, P. Zhu, W. Zhao, X. Liu, and G. Hu. “Coordinated cyber-physical attacks based on different attack strategies for cascading failure analysis in smart grids”. In: *Wireless Networks* (Aug. 2021). DOI: 10.1007/s11276-021-02752-6. URL: https://doi.org/10.1007/s11276-021-02752-6.

[69] M. Shen, X. Gao, and M. Peng. “Cascading failure dynamic of cyber-physical power system considering malware attacks”. In: *Journal of Physics: Conference Series* 1592.1 (Aug. 2020), p. 012015. DOI: 10.1088/1742-6596/1592/1/012015. URL: https://doi.org/10.1088/1742-6596/1592/1/012015.

[70] W. Zhu, J. V. Milanovic, and B. Mihic. “Assessing the Applicability of Complex Network Theory Models and Importance Measures to Vulnerability Studies of Cyber-physical Systems”. In: *2019 IEEE PES Innovative Smart Grid Technologies Europe (ISGT-Europe)* (Sept. 2019), pp. 1–6. DOI: 10.1109/isgteurope.2019.8905525. URL: https://doi.org/10.1109/isgteurope.2019.8905525.

[71] D. Ding, H. Wu, X. Yu, H. Wang, L. Yang, H. Wang, X. Kong, Q. Liu, and Z. Lu. “Vulnerability assessment of Cyber Physical Power System based on improved cascading failure model”. In: *Journal of Electrical Engineering & Technology* (May 2024). DOI: 10.1007/s42835-024-01929-1. URL: https://doi.org/10.1007/s42835-024-01929-1.

[72] J. Jiang, Y. Xia, S. Xu, H.-L. Shen, and J. Wu. “An asymmetric interdependent networks model for cyber-physical systems”. In: *Chaos* 30.5 (May 2020). DOI: 10.1063/1.5139254. URL: https://doi.org/10.1063/1.5139254.

[73] V. S. Rajkumar, A. Ştefanov, A. Presekal, P. Palensky, and J. L. R. Torres. “Cyber Attacks on power grids: Causes and propagation of cascading failures”. In: *IEEE Access* 11 (Jan. 2023), pp. 103154–103176. DOI: 10.1109/access.2023.3317695. URL: https://doi.org/10.1109/access.2023.3317695.

[74] H. Zhang, Y. Teng, J. M. Guerrero, P. Siano, and X. Sun. “Analysis of failure propagation in Cyber-Physical power systems based on an epidemic model”. In: *Energies* 16.6 (Mar. 2023), p. 2624. DOI: 10.3390/en16062624. URL: https://doi.org/10.3390/en16062624.

[75] L. Chen, S. Guo, C. Dou, H. Ge, Z. Cheng, and S. Li. “Dynamics of cascading failure in cyber-physical power systems from cyber attack”. In: *Physica Scripta* 99.3 (Feb. 2024), p. 035243. DOI: 10.1088/1402-4896/ad28e4. URL: https://doi.org/10.1088/1402-4896/ad28e4.

[76] Z. Xu, Y. Ge, Q. Lin, R. Chen, J. Cao, and N. Yu. “Robustness Analysis of CPPS considering Power Flow Constraints”. In: *International Transactions on Electrical Energy Systems* 2022 (July 2022), pp. 1–9. DOI: 10.1155/2022/7385104. URL: https://doi.org/10.1155/2022/7385104.

[77] S. V. Buldyrev, R. Parshani, G. Paul, H. E. Stanley, and S. Havlin. “Catastrophic cascade of failures in interdependent networks”. In: *Nature* 464.7291 (Apr. 2010), pp. 1025–1028. DOI: 10.1038/nature08932. URL: https://doi.org/10.1038/nature08932.

[78] J. Guo, Y. Han, C. Guo, F. Lou, and Y. Wang. “Modeling and vulnerability analysis of Cyber-Physical power systems considering network topology and power flow properties”. In: *Energies* 10.1 (Jan. 2017), p. 87. DOI: 10.3390/en10010087. URL: https://doi.org/10.3390/en10010087.

[79] Y. Cai, Y. Cao, Y. Li, T. Huang, and B. Zhou. “Cascading Failure Analysis considering interaction between power grids and communication networks”. In: *IEEE Transactions on Smart Grid* 7.1 (Jan. 2016), pp. 530–538. DOI: 10.1109/tsg.2015.2478888. URL: https://doi.org/10.1109/tsg.2015.2478888.

[80] Y. Cai, Y. Li, Y. Cao, W. Li, and X. Zeng. “Modeling and impact analysis of interdependent characteristics on cascading failures in smart grids”. In: *International Journal of Electrical Power & Energy Systems* 89 (July 2017), pp. 106–114. DOI: 10.1016/j.ijepes.2017.01.010. URL: https://doi.org/10.1016/j.ijepes.2017.01.010.

[81] X. Zhang, D. Liu, C. Zhan, and C. K. Tse. “Effects of cyber coupling on cascading failures in power systems”. In: *IEEE Journal on Emerging and Selected Topics in Circuits and Systems* 7.2 (June 2017), pp. 228–238. DOI: 10.1109/jetcas.2017.2698163. URL: https://doi.org/10.1109/jetcas.2017.2698163.

[82] Y. Zhang and O. Yagan. “Robustness of interdependent Cyber-Physical systems against cascading failures”. In: *IEEE Transactions on Automatic Control* 65.2 (Feb. 2020), pp. 711–726. DOI: 10.1109/tac.2019.2918120. URL: https://doi.org/10.1109/tac.2019.2918120.

[83] Y. Chen, Y. Li, W. Li, X. Wu, Y. Cai, Y. Cao, and C. Rehtanz. “Cascading failure analysis of cyber physical power system with multiple interdependency and control threshold”. In: *IEEE Access* 6 (Jan. 2018), pp. 39353–39362. DOI: 10.1109/access.2018.2855441. URL: https://doi.org/10.1109/access.2018.2855441.

[84] A. Sturaro, S. Silvestri, M. Conti, and S. K. Das. “A realistic model for failure propagation in interdependent Cyber-Physical systems”. In: *IEEE Transactions on Network Science and Engineering* 7.2 (Apr. 2020), pp. 817–831. DOI: 10.1109/tnse.2018.2872034. URL: https://doi.org/10.1109/tnse.2018.2872034.

[85] L. Xie, P. E. Heegaard, and Y. Jiang. *Modeling and Quantifying the Survivability of Telecommunication Network Systems under Fault Propagation*. Jan. 2013, pp. 25–36. DOI: 10.1007/978-3-642-40552-5_3. URL: https://doi.org/10.1007/978-3-642-40552-5_3.

[86] D. Zhao, Z. Wang, G. Xiao, B. Gao, and L. Wang. “The robustness of interdependent networks under the interplay between cascading failures and virus propagation”. In: *Europhysics Letters* 115.5 (Sept. 2016), p. 58004. DOI: 10.1209/0295-5075/115/58004. URL: https://doi.org/10.1209/0295-5075/115/58004.

[87] J. Ripoll, M. Manzano, and E. Calle. “Spread of epidemic-like failures in telecommunication networks”. In: *Physica A* 410 (Sept. 2014), pp. 457–469. DOI: 10.1016/j.physa.2014.05.052. URL: https://doi.org/10.1016/j.physa.2014.05.052.

[88] W. Chen, B. Heydari, A. Maier, and J. H. Panchal. “Network-based modeling and analysis in design”. In: *Design Science* 4 (Jan. 2018). DOI: 10.1017/dsj.2018.8. URL: https://doi.org/10.1017/dsj.2018.8.

[89] P. Hadaj, D. Strzałka, M. Nowak, M. Łatka, and P. Dymora. “The use of PLANS and NetworkX in modeling power grid system failures”. In: *Scientific Reports* 12.1 (Oct. 2022). DOI: 10.1038/s41598-022-22268-z. URL: https://doi.org/10.1038/s41598-022-22268-z.

[90] D. Oudart, J. Cantenot, F. Boulanger, and S. Chabridon. *The Smart Grid Simulation Framework: Model-Driven engineering applied to Cyber-Physical systems*. Jan. 2021, pp. 3–25. DOI: 10.1007/978-3-030-67445-8_1. URL: https://doi.org/10.1007/978-3-030-67445-8_1.

[91] D. Easley and J. Kleinberg. *Networks, crowds, and markets*. Cambridge University Press, July 2010. DOI: 10.1017/cbo9780511761942. URL: https://doi.org/10.1017/cbo9780511761942.

[92] P. Heijnen. *SEN124A – Design in Networked Systems - Lecture 1b: Network Representations*. [PowerPoint slides]. Feb. 2024. URL: https://brightspace.tudelft.nl/d2l/le/content/597390/viewContent/3392585/View.

[93] M. Albadi. *Power Flow Analysis*. IntechOpen, Mar. 2020. DOI: 10.5772/intechopen.83374. URL: https://doi.org/10.5772/intechopen.83374.

[94] D. Çetinkaya, A. Verbraeck, and M. Seck. “MDD4MS: a model driven development framework for modeling and simulation”. In: *Proceedings of the 2011 Summer Computer Simulation Conference* (June 2011), pp. 113–121. DOI: 10.5555/2348196.2348212. URL: https://dl.acm.org/citation.cfm?id=2348212.

[95] Systems Engineering Fundamentals. *Validation & Verification*. 2001. URL: https://ocw.tudelft.nl/wp-content/uploads/Validation-verification.pdf (visited on 05/07/2024).

[96] University of Southampton. *Network Components and the Giant Component*. 2017. URL: https://www.futurelearn.com/info/courses/social-media/0/steps/16048 (visited on 05/13/2024).

[97] NetworkX developers. *NetworkX. Network Analysis in Python*. 2024. URL: https://networkx.org/ (visited on 05/14/2024).

[98] A. Hagberg, D. A. Schult, and P. J. Swart. “Exploring Network Structure, Dynamics, and Function using NetworkX”. In: *Proceedings of the 7th Python in Science Conference (SciPy2008)* (Jan. 1, 2008), pp. 11–15. URL: http://conference.scipy.org/proceedings/scipy2008/paper_2/full_text.pdf.

[99] NetworkX developers. *Tutorial*. URL: https://networkx.org/documentation/stable/tutorial.html (visited on 05/14/2024).

[100] L. Thurner, A. Scheidler, F. Schäfer, J.-H. Menke, J. Dollichon, F. Meier, S. Meinecke, and M. Braun. “PandaPower—An Open-Source Python tool for convenient modeling, analysis, and optimization of electric power systems”. In: *IEEE Transactions on Power Systems* 33.6 (Nov. 2018), pp. 6510–6521. DOI: 10.1109/tpwrs.2018.2829021. URL: https://doi.org/10.1109/tpwrs.2018.2829021.

[101] *pandapower*. URL: https://www.pandapower.org/ (visited on 05/14/2024).

[102] J. J. Grainger and W. D. Stevenson. *Power System Analysis*. Jan. 1994. URL: http://ci.nii.ac.jp/ncid/BA23952686.

[103] R. Christie. *Power Systems Test Case Archive. 118 Bus Power Flow Test Case*. 1993. URL: https://www2.ee.washington.edu/research/pstca/pf118/pg_tca118bus.htm (visited on 05/14/2024).

[104] P. Heijnen. *SEN124A – Design in networked systems - Lecture 3a: Network Diffusion*. [PowerPoint slides & IPYNB File]. Feb. 2024. URL: https://brightspace.tudelft.nl/d2l/le/content/597390/viewContent/3392592/View.

[105] J. Hu, Z.-H. Li, and X.-Z. Duan. “Structural feature analysis of the electric power dispatching data network”. In: *Zhongguo Dianji Gongcheng Xuebao/Proceedings of the Chinese Society of Electrical Engineering* 29 (Feb. 2009), pp. 53–59.

[106] A. L. Barabási and R. Albert. “Emergence of scaling in random networks”. In: *Science* 286.5439 (Oct. 1999), pp. 509–512. DOI: 10.1126/science.286.5439.509. URL: https://doi.org/10.1126/science.286.5439.509.

[107] M. E. J. Newman and D. J. Watts. “Renormalization group analysis of the small-world network model”. In: *Physics Letters A* 263.4-6 (Dec. 1999), pp. 341–346. DOI: 10.1016/s0375-9601(99)00757-4. URL: https://doi.org/10.1016/s0375-9601(99)00757-4.

[108] K. Ahad, S. H. Jadid, and H. Andami. “Introducing a new method for multiarea transmission networks loss allocation”. In: *DOAJ (DOAJ: Directory of Open Access Journals)* (Jan. 2007). URL: https://doaj.org/article/02ff028460ae4b479a653433cabcd7dc.

[109] H. Arasteh, M. Kia, V. Vahidinasab, M. Shafie-Khah, and J. P. S. Catalão. “Multiobjective generation and transmission expansion planning of renewable dominated power systems using stochastic normalized normal constraint”. In: *International Journal of Electrical Power & Energy Systems* 121 (Oct. 2020), p. 106098. DOI: 10.1016/j.ijepes.2020.106098. URL: https://doi.org/10.1016/j.ijepes.2020.106098.

[110] P. Heijnen. *SEN124A - Design in Networked Systems- Lecture 2A: Network Analysis Importance of nodes and Edges*. [PowerPoint slides]. Feb. 2023. URL: https://brightspace.tudelft.nl/d2l/le/content/597390/viewContent/3392587/View.

[111] P. Eder-Neuhauser, T. Zseby, and J. Fabini. “Malware propagation in smart grid monocultures”. In: *E & I. Elektrotechnik und Informationstechnik* 135.3 (June 2018), pp. 264–269. DOI: 10.1007/s00502-018-0616-5. URL: https://doi.org/10.1007/s00502-018-0616-5.

[112] International Energy Agency. *Renewable electricity growth is accelerating faster than ever worldwide, supporting the emergence of the new global energy economy*. Dec. 1, 2021. URL: https://www.iea.org/news/renewable-electricity-growth-is-accelerating-faster-than-ever-worldwide-supporting-the-emergence-of-the-new-global-energy-economy (visited on 06/13/2024).

[113] X. Wang, F. Xue, Q. Wu, S. Lu, L. Jiang, and Y. Hu. “Evaluation for risk of cascading failures in power grids by Inverse-Community Structure”. In: *IEEE Internet of Things Journal* 10.9 (May 2023), pp. 7459–7468. DOI: 10.1109/jiot.2022.3189001. URL: https://doi.org/10.1109/jiot.2022.3189001.

[114] *Cyber-Physical Threat Intelligence for Critical Infrastructures security: Securing critical infrastructures in air transport, water, gas, healthcare, finance and industry*. Jan. 2021. DOI: 10.1561/9781680838237. URL: https://doi.org/10.1561/9781680838237.

A

Model code implementation -Python

The smart grid model is available on our GitHub repository which can be accessed by clicking here or copy and paste the following link:
`https://github.com/marijnburgers4/smart-grid-model-cascading-failure`.

A.1. Python packages import

```
import networkx as nx
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import pandapower as pp
import pandapower.networks as pn
import pandapower.topology as top
import matplotlib.patches as mpatches
import numba
import warnings
import math
import random
import pydot
from networkx.drawing.nx_pydot import graphviz_layout
import re
from pandapower.plotting import simple_plot, simple_plotly, pf_res_plotly
```

A.2. Communication network generation

```
def small_world_network(num_comm_nodes):
    # Create a connected small world network using the connected Watts-Strogatz model.
    # num_comm_nodes: Number of nodes in the graph.
    # The connected_watts_strogatz_graph function generates a Watts-Strogatz small-world graph.
    # Parameters: num_comm_nodes (number of nodes), 6 (each node is connected to k=6 nearest neighbors in ring topology),
    # 0.2 (rewiring probability), tries=100 (attempts to generate a connected graph).

    G = nx.connected_watts_strogatz_graph(num_comm_nodes, 6, 0.2, tries=100)

    # Return the generated graph
    return G
```

```
def scale_free_network(num_comm_nodes):
    # Create a scale-free network using the Barabási-Albert model.
    # num_comm_nodes: Number of nodes in the graph.
    # The barabasi_albert_graph function generates a scale-free network.
    # Parameters: num_comm_nodes (number of nodes), 5 (number of edges to attach from a new node to existing nodes).

```

```
    G = nx.barabasi_albert_graph(num_comm_nodes, 5)

    # Return the generated graph
    return G
```

A.3. Adding SCADA nodes to the communication network

```
def full_communication_network_scada(G):
    # Create the 3 SCADA nodes and add them to the communication network
    scada_nodes = ['scada1', 'scada2', 'scada3']
    G.add_nodes_from(scada_nodes)

    # Calculate the number of only communication nodes
    num_nodes = G.number_of_nodes() - len(scada_nodes)

    # Calculate the indices for each third of communication nodes (needed for the connection with SCADA nodes)
    first_third_end = num_nodes // 3
    second_third_end = 2 * num_nodes // 3

    # Create a dictionary to map communication nodes to their corresponding SCADA nodes
    node_to_scada = {}

    # This function randomly connects SCADA nodes to a quarter of the communication nodes in their respective thirds
    def connect_scada(scada, start, end):
        nodes = list(range(start, end))
        num_to_connect = len(nodes) // 4
        selected_nodes = random.sample(nodes, num_to_connect)
        edges = [(scada, node) for node in selected_nodes]
        G.add_edges_from(edges)
        for node in selected_nodes:
            node_to_scada[node] = scada

    # Assign each communication node in the first third to scada1
    for node in range(0, first_third_end):
        node_to_scada[node] = 'scada1'

    # Assign each communication node in the second third to scada2
    for node in range(first_third_end, second_third_end):
        node_to_scada[node] = 'scada2'

    # Assign each commmunication node in the last third to scada3
    for node in range(second_third_end, num_nodes):
        node_to_scada[node] = 'scada3'

    # Connect scada1 to a quarter of the first third of nodes
    connect_scada('scada1', 0, first_third_end)

    # Connect scada2 to a quarter of the second third of nodes
    connect_scada('scada2', first_third_end, second_third_end)

    # Connect scada3 to a quarter of the last third of nodes
    connect_scada('scada3', second_third_end, num_nodes)

    return G, scada_nodes, node_to_scada
```

A.4. 118-bus test case and setup

```
#Load the 118-test case from pandapower
net = pn.case118()

#Use this function to set overall load higher in the network
def power_network_set_up(net):
    #Surpess future warnings
    warnings.simplefilter(action='ignore', category=FutureWarning)

```

```
    #Change values of io_percent since it can not be negative
    net.trafo['i0_percent'] = 0.0

    #Decrease the load in the network with 15 percent
    net.load['p_mw'] *= 0.85

    #Change the value of max_i_ka to a realistic value and a higher load
    net.line['max_i_ka'] /= 31

    #Change resistance in lines so it we create a higher load in the network
    net.line['r_ohm_per_km'] /= 4
    net.line['x_ohm_per_km'] /= 4
    net.line['c_nf_per_km'] /= 4
```

A.5. Choose initial failed node in communication network

```
def degree_centrality_list(G, initial_percent_of_network, scada_nodes):
    # Calculate degree centrality for the nodes in the graph G
    degree_centrality = nx.degree_centrality(G)

    # Exclude the SCADA nodes from the degree centrality list
    for node in scada_nodes:
        if node in degree_centrality:
            del degree_centrality[node]

    # Sort nodes based on degree centrality in descending order
    sorted_nodes = sorted(degree_centrality, key=degree_centrality.get, reverse=True)

    # Calculate the number of top nodes to select based on the initial percent of network
        failed criteria
    top_percent = int((initial_percent_of_network / 100) * len(sorted_nodes))

    # Select the top nodes based on the calculated top_percent
    top_nodes = sorted_nodes[:top_percent]

    # Return the list of top nodes
    return top_nodes
```

```
def betweenness_centrality_list(G, initial_percent_of_network, scada_nodes):
    # Calculate betweenness centrality for the nodes in the graph G
    betweenness_centrality = nx.betweenness_centrality(G)

    # Remove SCADA nodes from consideration
    for node in scada_nodes:
        if node in betweenness_centrality:
            del betweenness_centrality[node]

    # Sort nodes based on betweenness centrality in descending order
    sorted_nodes = sorted(betweenness_centrality, key=betweenness_centrality.get, reverse=
        True)

    # Calculate the number of top nodes to select based on the initial percent of network
        failed criteria
    top_percent = int((initial_percent_of_network / 100) * len(sorted_nodes))

    # Select the top nodes based on the calculated top_percent
    top_nodes = sorted_nodes[:top_percent]

    # Return the list of top nodes
    return top_nodes
```

```
def closeness_centrality_list(G, initial_percent_of_network, scada_nodes):
    # Calculate closeness centrality for the nodes in the graph G
    closeness_centrality = nx.closeness_centrality(G)

    # Exclude the SCADA nodes from the degree centrality list
    for node in scada_nodes:
        if node in closeness_centrality:
            del closeness_centrality[node]
```

```

    # Sort nodes based on closeness centrality in descending order
    sorted_nodes = sorted(closeness_centrality, key=closeness_centrality.get, reverse=True)

    # Calculate the number of top nodes to select based on the initial percent of network
        failed criteria
    top_percent = int((initial_percent_of_network / 100) * len(sorted_nodes))

    # Select the top nodes based on the calculated top_percent
    top_nodes = sorted_nodes[:top_percent]

    # Return the list of top nodes
    return top_nodes
```

```
def random_list(G, initial_percent_of_network, scada_nodes):
    # List all nodes except SCADA nodes
    available_nodes = [node for node in G.nodes() if node not in scada_nodes]

    # Calculate the number of nodes to randomly select based on the initial percent of
        network failed criteria
    num_nodes_to_select = int((initial_percent_of_network / 100) * len(available_nodes))

    # Randomly select nodes from the graph
    selected_nodes = random.sample(available_nodes, num_nodes_to_select)

    # Print the selected random nodes and their count
    # print('random nodes', selected_nodes, len(selected_nodes))

    # Return the list of selected random nodes
    return selected_nodes
```

A.6. Failure propagation communication network

```
def diffusion(G, adopters, fail_criteria, node_to_scada, scada_nodes):
    """
    Diffusion model for node failure based on the fraction of failed neighbors
    and the connectivity to SCADA nodes.

    Parameters:
    G - the network graph
    adopters - set of initially failed nodes
    fail_criteria - threshold fraction of neighbors' failure to cause node failure
    node_to_scada - dictionary mapping normal nodes to their corresponding SCADA nodes
    scada_nodes - list of SCADA nodes

    Returns:
    adopters - set of all the failed nodes
    """
    # Uncomment these lines if you want to draw the communication network and see which nodes
        fail
    # def plot_adopters(G,pos,adopters,fail_criteria):
    #     nx.draw_networkx(G,pos, with_labels = False, node_size = 70, font_size = 8)
    #     nx.draw_networkx(G,pos, with_labels = False, nodelist=adopters, node_size = 70,
        font_size = 8, node_color='red')
    #     nx.draw_networkx(G, pos, with_labels=False, nodelist=scada_nodes, node_size=70,
        font_size=8, node_color='green')

    #     plt.title(f" Fail if fraction of failed neighbours > {fail_criteria}")
    #     plt.show()

    # Plot initial network
    # pos = nx.spring_layout(G, seed=32)
    # nx.draw_networkx(G, pos, with_labels=False, node_size=70, font_size=8)
    # nx.draw_networkx(G, pos, with_labels=False, nodelist=adopters, node_size=70, font_size
        =8, node_color='red')
    # nx.draw_networkx(G, pos, with_labels=False, nodelist=scada_nodes, node_size=70,
        font_size=8, node_color='green')
    # plt.title(f"Fail if fraction of failed neighbors > {fail_criteria}")
    # plt.show()
```

```
32
33      def is_disconnected_from_scada(node, G, adopters, node_to_scada):
34          # Check if there is a path to its corresponding SCADA node through non-failed nodes
35          scada_node = node_to_scada.get(node, None)
36          if not scada_node:
37              return False  # Node has no corresponding SCADA node
38          non_adopted_nodes = set(G.nodes()) - adopters
39          return not nx.has_path(G.subgraph(non_adopted_nodes), node, scada_node)
40
41      # Plot adopters in network
42      # plot_adopters(G, pos, adopters, fail_criteria)
43      new_adopters = set()  # New adopters in initial state
44      new_adopters.update(adopters.copy())
45
46      # Continue while new adopters can be found
47      while new_adopters:
48          # Start with an empty set of new adopters
49          new_adopters = set()
50          # Loop through all nodes that have not adopted yet and are not SCADA nodes
51          for node in (set(G.nodes()) - adopters - set(scada_nodes)):
52              # Find all their neighbors
53              neighbors = list(nx.neighbors(G, node))
54              # If neighbor adopters exceed threshold or the node is disconnected from SCADA
55              if (len(neighbors) and (len([nb for nb in neighbors if nb in adopters]) / len(
                    neighbors) > fail_criteria)) or \
56                 is_disconnected_from_scada(node, G, adopters, node_to_scada):
57                  # Add node to new adopters
58                  new_adopters.add(node)
59          # Add all new adopters to total set of adopters
60          adopters = adopters.union(new_adopters)
61          # Plot adopters in network
62          # if new_adopters:
63              # plot_adopters(G, pos, adopters, fail_criteria)
64      # If all nodes failed in the network, then a cascading failure has occurred
65
66      # Return a set of all the failed nodes
67      return adopters
```

A.7. Initial failure in power grid from communication nodes

```
1  def remove_initial_lines(net, lines_failed_list):
2      """
3      Sets the 'in_service' status of specified lines in the network to False, from failed
           communication nodes, simulating line failures
4
5      Parameters:
6      - net: The network object containing line data.
7      - lines_failed_list: A list of line indices that are to be marked as failed. These
           indices are the corresponding communication nodes that have failed
8
9      """
10     # Loop through each line index in the list of failed lines
11     for line_idx in lines_failed_list:
12         # Set the 'in_service' status of the line at the given index to False
13         net.line.at[line_idx, 'in_service'] = False
```

A.8. Giant Connected Component, generator, load and slack bus

```
1  def giant_connected_component(net):
2      """
3      Identifies and processes the giant connected component in a network. Ensures
4      that the network simulation continues only if there are generators and loads
5      in the giant connected component.
6
7      Parameters:
8      - net: The network object containing bus, generator, load, transformer,
9        external grid, and line data.
```

```
    Returns:
    - False if there are no generators or loads in the giant connected component.
    - None if the process completes successfully.
    """

    # Create a NetworkX graph from the network, respecting switches and excluding out-of-service elements
    H = top.create_nxgraph(net, respect_switches=True, include_out_of_service=False)

    # Find all connected components in the graph
    connected_components = list(nx.connected_components(H))

    # Identify the largest (giant) connected component
    giant_component = max(connected_components, key=len)

    # Convert the giant component to a set for efficient lookup
    giant_component_set = set(giant_component)

    # Uncomment the line below for debugging purposes
    # print(connected_components)

    # Find all buses that are part of the giant component
    giant_component_buses = net.bus.index.isin(giant_component)

    # Check if there is at least one generator in the giant component
    gen_in_giant_component = net.gen[net.gen.bus.isin(giant_component) & net.gen.in_service].shape[0] > 0

    # Check if there is at least one load in the giant component
    load_in_giant_component = net.load[net.load.bus.isin(giant_component) & net.load.in_service].shape[0] > 0

    # If there are no generators or loads in the giant component, stop the simulation
    if gen_in_giant_component == 0 or load_in_giant_component == 0:
        print('No generators and/or no loads in giant connected component')
        return False

    # Turn off the buses that are not in the giant connected component
    for bus_id in set(net.bus.index) - giant_component_set:
        net.bus.loc[bus_id, 'in_service'] = False

        # Turn off transformers connected to the buses that are not in the giant component
        trafo_to_set_oos = net.trafo.index[(net.trafo.hv_bus == bus_id) | (net.trafo.lv_bus == bus_id)]
        net.trafo.loc[trafo_to_set_oos, 'in_service'] = False

        # Turn off external grids connected to the buses that are not in the giant component
        ext_grids_to_set_oos = net.ext_grid.index[net.ext_grid.bus == bus_id]
        net.ext_grid.loc[ext_grids_to_set_oos, 'in_service'] = False

        # Uncomment the lines below to turn off lines connected to the buses that are not in the giant component
        # lines_to_set_oos = net.line.index[(net.line.from_bus == bus_id) | (net.line.to_bus == bus_id)]
        # net.line.loc[lines_to_set_oos, 'in_service'] = False

        # Turn off loads connected to the buses that are not in the giant component
        loads_to_set_oos = net.load.index[net.load.bus == bus_id]
        net.load.loc[loads_to_set_oos, 'in_service'] = False

        # Turn off generators connected to the buses that are not in the giant component
        gens_to_set_oos = net.gen.index[net.gen.bus == bus_id]
        net.gen.loc[gens_to_set_oos, 'in_service'] = False

    # Find the slack buses for external grids and generators
    ext_grid_slack_buses = net.ext_grid.bus.values
    gen_slack_buses = net.gen.bus[net.gen.slack == True].values

    # Check if any of the external grid slack buses are in the giant connected component
    ext_grid_slack_in_giant_component = any(bus in giant_component_set for bus in net.
```

```
        ext_grid.bus.values)

    # Combine the slack buses from external grids and generators into a set for efficient lookup
    slack_buses_set = set(ext_grid_slack_buses) | set(gen_slack_buses)

    # Check if any of the slack buses are in the giant connected component
    slack_in_giant_component = any(bus in giant_component_set for bus in slack_buses_set)

    # Print the result regarding the presence of slack buses in the giant connected component
    if ext_grid_slack_in_giant_component:
        print("There is an external grid slack bus in the giant connected component.")
    elif slack_in_giant_component:
        print("There is a slack bus in the giant connected component.")
    else:
        print("No slack bus in the giant connected component.")

        # If no slack bus is found, set the generator with the largest 'p_mw' value as slack
        gens_in_giant_component = net.gen[net.gen.bus.isin(giant_component_set)]
        if not gens_in_giant_component.empty:
            # Find the generator with the largest 'p_mw' value
            largest_gen = gens_in_giant_component.loc[gens_in_giant_component['p_mw'].idxmax()]

            # Set this generator as slack
            net.gen.at[largest_gen.name, 'slack'] = True
            print("Generator at bus {} with capacity {} MW is now set as slack.".format(largest_gen.bus, largest_gen.p_mw))
        else:
            print("No generators found in the giant connected component to set as slack.")
```

A.9. Rebalance the load in the power grid

```
def balans_grid(net):
    """
    Balances the power generation in the network based on the total active load and the maximum generation capacity of the in-service generators.

    """

    # Calculate the total active and reactive power for in-service loads only
    total_load_p = net.load[net.load['in_service']]['p_mw'].sum()
    total_load_q = net.load[net.load['in_service']]['q_mvar'].sum()
    # Uncomment the line below to print the total in-service load
    # print(f"Total in-service load: {total_load_p} MW, {total_load_q} MVAr")

    # Calculate the total active power and reactive power capability range for in-service generators
    total_gen_p = net.gen[net.gen['in_service']]['p_mw'].sum()
    total_gen_min_q = net.gen[net.gen['in_service']]['min_q_mvar'].sum()
    total_gen_max_q = net.gen[net.gen['in_service']]['max_q_mvar'].sum()
    # Uncomment the lines below to print the total in-service generation and reactive power capability range
    # print(f"Total in-service generation (gen): {total_gen_p} MW")
    # print(f"Total in-service reactive power capability range for gen: {total_gen_min_q} MVAr to {total_gen_max_q} MVAr")

    # Calculate the total active and reactive power for in-service external grids only
    total_ext_grid_p = net.ext_grid[net.ext_grid['in_service']]['max_p_mw'].sum()
    total_ext_grid_q = net.ext_grid[net.ext_grid['in_service']]['max_q_mvar'].sum()
    # Uncomment the line below to print the total in-service external grid generation
    # print(f"Total in-service generation (ext_grid): {total_ext_grid_p} MW, {total_ext_grid_q} MVAr")

    # Calculate the total maximum active power for in-service generators
    total_max_p = net.gen[net.gen['in_service']]['max_p_mw'].sum()

    # Print the total active power load and generation before balancing
    print(f'Before balans: Total p_mw load: {total_load_p}, Total p_mw generators: {
```

```
        total_gen_p}')

    # Adjust the active power generation for each in-service generator based on their proportion of the total maximum generation capacity
    for idx, gen in net.gen.iterrows():
        if gen.in_service:
            proportion = gen.max_p_mw / total_max_p
            allocated_p = total_load_p * proportion
            net.gen.at[idx, 'p_mw'] = allocated_p

    # Calculate the total active power load and generation after balancing
    total_load_p_after = net.load[net.load['in_service']]['p_mw'].sum()
    total_gen_p_after = net.gen[net.gen['in_service']]['p_mw'].sum()

    # Print the total active power load and generation after balancing
    print(f'After balans: Total p_mw load: {total_load_p_after}, Total p_mw generators: {total_gen_p_after}')
```

A.10. Running the PFA

```
def run_power_flow(net):
    #Run an AC power flow
    try:
        pp.runpp(net, max_iterations = 25)
        # pf_res_plotly(net)
        # Check for convergence using the success flag
        if net["_ppc"]["success"]:
            # print("The power flow simulation converged successfully.")
            pass
        else:
            print("The power flow simulation did not converge.")
    except Exception as e:
        #print(f"Power flow simulation failed due to an error: {e}")

        #Decrease the load by 10 percent if an error has occured. rebalance the network and run the PFA again
        net.load['p_mw'] *= 0.9
        balance_grid(net)
        print('decreasaed load with 0.9')
        run_power_flow(net)
```

A.11. Fail lines exceeding load capacity

```
def remove_overload(net):
    """
    This function identifies overloaded lines and transformers in the network and takes them out of service.
    It checks if any line or transformer has a loading percent greater than 100%.
    If any are found, it sets their 'in_service' status to False and prints a message indicating the action taken.
    If no overloaded components are found, it prints a message stating there are no overloaded lines or transformers.
    """
    # Identify overloaded lines by checking if their loading percent is greater than 100%
    overloaded_lines = net.res_line[net.res_line.loading_percent > 100].index

    # Check if there are no overloaded lines or transformers
    if len(net.res_line[net.res_line.loading_percent > 100].index) == 0 and len(net.res_trafo[net.res_trafo.loading_percent > 100].index) == 0:
        print('No overloaded lines or transformers')
        return False
    else:
        # For each overloaded line, set 'in_service' to False and print its status
        for line_idx in overloaded_lines:
            net.line.at[line_idx, 'in_service'] = False
            print(f"Line {line_idx} is overloaded with a loading percent of {net.res_line.at[line_idx, 'loading_percent']:.2f}% and has been taken out of service.")
```

```
20
21          # Identify overloaded transformers by checking if their loading percent is greater
                than 100%
22          overloaded_transformers = net.res_trafo[net.res_trafo.loading_percent > 100].index
23
24          # For each overloaded transformer, set 'in_service' to False and print its status
25          for trafo_idx in overloaded_transformers:
26              net.trafo.at[trafo_idx, 'in_service'] = False
27              print(f"Transformer␣{trafo_idx}␣is␣overloaded␣with␣a␣loading␣percent␣of␣{net.
                    res_trafo.at[trafo_idx,␣'loading_percent']:.2f}%␣and␣has␣been␣taken␣out␣of␣
                    service.")
```

A.12. Back the communication network

```
1  """
2  This code snippet checks for lines in the network that are out of service and manages the
       process of identifying and handling new failures based on new failures in the electricity
        grid. Note that the variable amount_lines_failed is not defined here, this variable
       represents the amount of lines failed at the end of the diffusion process.
3  """
4
5  # Identify lines that are out of service
6  out_of_service_lines = net.line[net.line['in_service'] == False]
7
8
9  # Check if the number of out-of-service lines has increased
10 if len(out_of_service_lines) > amount_lines_failed:
11     # Update the set of adopters with the indices of out-of-service lines
12     adopters.update(set(out_of_service_lines.index))
13
14     # Perform diffusion process to find new devices that failed
15     new_device_failed_set = diffusion(G, adopters, fail_criteria)
16     new_lines_failed_list = list(new_device_failed_set)
17
18     # Remove the newly failed lines from service
19     remove_initial_lines(net, new_lines_failed_list)
20
21     # Update the count of failed lines
22     amount_lines_failed = len(out_of_service_lines)
23 else:
24
25     #No additional components have failed so there is no need to run the diffusion process
           again
26     break
```

A.13. Running the simulation part 1

```
1  def simulation_all(G, adopters, scada_nodes, node_to_scada, fail_criteria):
2      #these are the devices that failed at the end of the diffusion model process
3      device_failed_set = diffusion(G, adopters, fail_criteria, node_to_scada, scada_nodes)
4
5      #convert the failed devices into transmission lines
6      lines_failed_list = list(device_failed_set)
7      amount_lines_failed = len(lines_failed_list)
8      # print(f'These lines have failed: \n{lines_failed_list}\n')
9
10     #load the 118 case
11     net = pn.case118()
12     initial_power_nodes = len(net.bus[net.bus.in_service == True])
13
14     #set up the power network with realistic variables to create a highe line loads
15     power_network_set_up(net)
16
17     #calculate the total initial demand of p_mw of the loads
18     total_initial_load_p = net.load[net.load['in_service']]['p_mw'].sum()
19
20     #remove the lines affected by the communication network
```

```
    remove_initial_lines(net, lines_failed_list)

    #run the simulation
    while True:

        #if there are no load or generators in the gcc then stop
        if giant_connected_component(net) == False:
            break
        #balance the load and the generation p_mw
        balance_grid(net)

        #execute the power flow analysis
        run_power_flow(net)

        #if there are no lines are transfomers overloaded stop simulation, otherwise remove
            them
        remove_overload(net)

        out_of_service_lines = net.line[net.line['in_service'] == False]
        # print('amounts failed', amount_lines_failed)
        # print(len(net.line[net.line['in_service'] == False]))

        #Check if new lines have failed
        if len(net.line[net.line['in_service'] == False]) > amount_lines_failed:
            # print('len out of service1\n', len(net.line[net.line['in_service'] == False]))
            # print('amount lines failed1\n', amount_lines_failed)
            adopters.update(set(out_of_service_lines.index)) #update the adopters (failed
                communication nodes) if new lines have failed
            # print('adopters update:\n', len(adopters))
            # print('Here wo go again!!')
            new_device_failed_set = diffusion(G, adopters, fail_criteria, node_to_scada,
                scada_nodes) #run the failure propagation again in the communicaiton network
            device_failed_set.update(new_device_failed_set)
            new_lines_failed_list = list(new_device_failed_set)
            remove_initial_lines(net, new_lines_failed_list)
            # print('len new lines failed list failed\n', len(new_lines_failed_list))
            amount_lines_failed = len(net.line[net.line['in_service'] == False])
            # print('amount lines failed2\n', amount_lines_failed)
        else:
            # print('check if it works')
            break

    #check the fraction of buses that are in service
    in_service_buses = net.bus[net.bus.in_service == True]
    fraction_power_nodes = (len(in_service_buses)/initial_power_nodes)
    # print('fraction_power_nodes', fraction_power_nodes)

    #check the average blackout size - communication network and power network
    total_nodes = G.number_of_nodes() + initial_power_nodes - len(scada_nodes)
    comm_nodes_serving = G.number_of_nodes() - len(device_failed_set) - len(scada_nodes)
    average_blackout_size = (comm_nodes_serving + len(in_service_buses)) / total_nodes
    # print('average_blackout_size', average_blackout_size)

    #Check the demand surviveability
    in_service_loads = net.load[net.load.in_service == True]
    total_served_p = net.res_load.loc[in_service_loads.index, 'p_mw'].sum()
    demand_survivability = total_served_p / total_initial_load_p
    # print('demand_survivability', demand_survivability)

    return fraction_power_nodes, average_blackout_size, demand_survivability
```

A.14. Running the simulation part 2

```
warnings.simplefilter(action='ignore', category=FutureWarning)
#amount of runs per percentage step
runs = 100

#amount of failed neighbours in order for a node to fail
fail_criteria = 0.5 #0.5
```

```
7
8  mode2 = 'scale␣free␣network'
9  mode = 'small␣world␣network'
10 for percent_of_network in range(0, 95, 5):
11     print(percent_of_network)
12     for i in range(0,runs):
13         # print(percent_of_network)
14         #create small world network and run a diffusion model on it based on degree
               centrality
15         G = small_world_network(173)
16         G2 = scale_free_network(173)
17
18         G, scada_nodes, node_to_scada = full_communication_network_scada(G)
19         G2, scada_nodes, node_to_scada = full_communication_network_scada(G2)
20
21         for metric in ['degree', 'betweenness', 'random', 'closeness']:
22             if mode == 'small␣world␣network':
23                 if metric == 'degree':
24                     adopters = set(degree_centrality_list(G, percent_of_network, scada_nodes)
                           )
25                     fraction_power_nodes, average_blackout_size, demand_survivability =
                           simulation_all(G, adopters, scada_nodes, node_to_scada, fail_criteria
                           )
26                     # print('degree',fraction_power_nodes, average_blackout_size,
                           demand_survivability)
27
28                     small_world_degree_fraction_power_nodes.append(fraction_power_nodes)
29                     small_world_degree_average_blackout_size.append(average_blackout_size)
30                     small_world_degree_demand_survivability.append(demand_survivability)
31
32                 elif metric == 'betweenness':
33                     adopters = set(betweenness_centrality_list(G, percent_of_network,
                           scada_nodes))
34                     fraction_power_nodes, average_blackout_size, demand_survivability =
                           simulation_all(G, adopters, scada_nodes, node_to_scada, fail_criteria
                           )
35                     # print('betweeness',fraction_power_nodes, average_blackout_size,
                           demand_survivability)
36
37                     small_world_betweenness_fraction_power_nodes.append(fraction_power_nodes)
38                     small_world_betweenness_average_blackout_size.append(
                           average_blackout_size)
39                     small_world_betweenness_demand_survivability.append(demand_survivability)
40
41                 elif metric == 'random':
42                     adopters = set(random_list(G, percent_of_network, scada_nodes))
43                     fraction_power_nodes, average_blackout_size, demand_survivability =
                           simulation_all(G, adopters, scada_nodes, node_to_scada, fail_criteria
                           )
44                     # print('random',fraction_power_nodes, average_blackout_size,
                           demand_survivability)
45
46                     small_world_random_fraction_power_nodes.append(fraction_power_nodes)
47                     small_world_random_average_blackout_size.append(average_blackout_size)
48                     small_world_random_demand_survivability.append(demand_survivability)
49
50                 elif metric == 'closeness':
51                     adopters = set(closeness_centrality_list(G, percent_of_network,
                           scada_nodes))
52                     fraction_power_nodes, average_blackout_size, demand_survivability =
                           simulation_all(G, adopters, scada_nodes, node_to_scada, fail_criteria
                           )
53                     # print('closeness',fraction_power_nodes, average_blackout_size,
                           demand_survivability)
54
55                     small_world_closeness_fraction_power_nodes.append(fraction_power_nodes)
56                     small_world_closeness_average_blackout_size.append(average_blackout_size)
57                     small_world_closeness_demand_survivability.append(demand_survivability)
58
59                 else:
60                     pass
```

```
            else:
                pass

            if mode2 == 'scale free network':
                if metric == 'degree':
                    adopters = set(degree_centrality_list(G2, percent_of_network, scada_nodes
                        ))
                    fraction_power_nodes, average_blackout_size, demand_survivability =
                        simulation_all(G2, adopters, scada_nodes, node_to_scada,
                        fail_criteria)

                    scale_free_degree_fraction_power_nodes.append(fraction_power_nodes)
                    scale_free_degree_average_blackout_size.append(average_blackout_size)
                    scale_free_degree_demand_survivability.append(demand_survivability)

                elif metric == 'betweenness':
                    adopters = set(betweenness_centrality_list(G2, percent_of_network,
                        scada_nodes))
                    fraction_power_nodes, average_blackout_size, demand_survivability =
                        simulation_all(G2, adopters, scada_nodes, node_to_scada,
                        fail_criteria)
                    # print('betweeness',fraction_power_nodes, average_blackout_size,
                        demand_survivability)

                    scale_free_betweenness_fraction_power_nodes.append(fraction_power_nodes)
                    scale_free_betweenness_average_blackout_size.append(average_blackout_size
                        )
                    scale_free_betweenness_demand_survivability.append(demand_survivability)

                elif metric == 'random':
                    adopters = set(random_list(G2, percent_of_network, scada_nodes))
                    fraction_power_nodes, average_blackout_size, demand_survivability =
                        simulation_all(G2, adopters, scada_nodes, node_to_scada,
                        fail_criteria)
                    # print('random',fraction_power_nodes, average_blackout_size,
                        demand_survivability)

                    scale_free_random_fraction_power_nodes.append(fraction_power_nodes)
                    scale_free_random_average_blackout_size.append(average_blackout_size)
                    scale_free_random_demand_survivability.append(demand_survivability)

                elif metric == 'closeness':
                    adopters = set(closeness_centrality_list(G2, percent_of_network,
                        scada_nodes))
                    fraction_power_nodes, average_blackout_size, demand_survivability =
                        simulation_all(G2, adopters, scada_nodes, node_to_scada,
                        fail_criteria)
                    # print('closeness',fraction_power_nodes, average_blackout_size,
                        demand_survivability)

                    scale_free_closeness_fraction_power_nodes.append(fraction_power_nodes)
                    scale_free_closeness_average_blackout_size.append(average_blackout_size)
                    scale_free_closeness_demand_survivability.append(demand_survivability)

                else:
                    pass

            else:
                pass
print('THE END')
```

B

Power grid data after network setup

B.1. Bus data

name	vn_kv	type	zone	in_service	max_vm_pu	min_vm_pu
1	138.00	b	1.00	True	1.06	0.94
2	138.00	b	1.00	True	1.06	0.94
3	138.00	b	1.00	True	1.06	0.94
4	138.00	b	1.00	True	1.06	0.94
5	138.00	b	1.00	True	1.06	0.94
6	138.00	b	1.00	True	1.06	0.94
7	138.00	b	1.00	True	1.06	0.94
8	345.00	b	1.00	True	1.06	0.94
9	345.00	b	1.00	True	1.06	0.94
10	345.00	b	1.00	True	1.06	0.94
11	138.00	b	1.00	True	1.06	0.94
12	138.00	b	1.00	True	1.06	0.94
13	138.00	b	1.00	True	1.06	0.94
14	138.00	b	1.00	True	1.06	0.94
15	138.00	b	1.00	True	1.06	0.94
16	138.00	b	1.00	True	1.06	0.94
17	138.00	b	1.00	True	1.06	0.94
18	138.00	b	1.00	True	1.06	0.94
19	138.00	b	1.00	True	1.06	0.94
20	138.00	b	1.00	True	1.06	0.94
21	138.00	b	1.00	True	1.06	0.94
22	138.00	b	1.00	True	1.06	0.94
23	138.00	b	1.00	True	1.06	0.94
24	138.00	b	1.00	True	1.06	0.94
25	138.00	b	1.00	True	1.06	0.94
26	345.00	b	1.00	True	1.06	0.94
27	138.00	b	1.00	True	1.06	0.94
28	138.00	b	1.00	True	1.06	0.94
29	138.00	b	1.00	True	1.06	0.94
30	345.00	b	1.00	True	1.06	0.94
31	138.00	b	1.00	True	1.06	0.94
32	138.00	b	1.00	True	1.06	0.94
33	138.00	b	1.00	True	1.06	0.94

Continued on next page

name	vn_kv	type	zone	in_service	max_vm_pu	min_vm_pu
34	138.00	b	1.00	True	1.06	0.94
35	138.00	b	1.00	True	1.06	0.94
36	138.00	b	1.00	True	1.06	0.94
37	138.00	b	1.00	True	1.06	0.94
38	345.00	b	1.00	True	1.06	0.94
39	138.00	b	1.00	True	1.06	0.94
40	138.00	b	1.00	True	1.06	0.94
41	138.00	b	1.00	True	1.06	0.94
42	138.00	b	1.00	True	1.06	0.94
43	138.00	b	1.00	True	1.06	0.94
44	138.00	b	1.00	True	1.06	0.94
45	138.00	b	1.00	True	1.06	0.94
46	138.00	b	1.00	True	1.06	0.94
47	138.00	b	1.00	True	1.06	0.94
48	138.00	b	1.00	True	1.06	0.94
49	138.00	b	1.00	True	1.06	0.94
50	138.00	b	1.00	True	1.06	0.94
51	138.00	b	1.00	True	1.06	0.94
52	138.00	b	1.00	True	1.06	0.94
53	138.00	b	1.00	True	1.06	0.94
54	138.00	b	1.00	True	1.06	0.94
55	138.00	b	1.00	True	1.06	0.94
56	138.00	b	1.00	True	1.06	0.94
57	138.00	b	1.00	True	1.06	0.94
58	138.00	b	1.00	True	1.06	0.94
59	138.00	b	1.00	True	1.06	0.94
60	138.00	b	1.00	True	1.06	0.94
61	138.00	b	1.00	True	1.06	0.94
62	138.00	b	1.00	True	1.06	0.94
63	345.00	b	1.00	True	1.06	0.94
64	345.00	b	1.00	True	1.06	0.94
65	345.00	b	1.00	True	1.06	0.94
66	138.00	b	1.00	True	1.06	0.94
67	138.00	b	1.00	True	1.06	0.94
68	161.00	b	1.00	True	1.06	0.94
69	138.00	b	1.00	True	1.06	0.94
70	138.00	b	1.00	True	1.06	0.94
71	138.00	b	1.00	True	1.06	0.94
72	138.00	b	1.00	True	1.06	0.94
73	138.00	b	1.00	True	1.06	0.94
74	138.00	b	1.00	True	1.06	0.94
75	138.00	b	1.00	True	1.06	0.94
76	138.00	b	1.00	True	1.06	0.94
77	138.00	b	1.00	True	1.06	0.94
78	138.00	b	1.00	True	1.06	0.94
79	138.00	b	1.00	True	1.06	0.94
80	138.00	b	1.00	True	1.06	0.94
81	345.00	b	1.00	True	1.06	0.94
82	138.00	b	1.00	True	1.06	0.94
83	138.00	b	1.00	True	1.06	0.94
84	138.00	b	1.00	True	1.06	0.94
85	138.00	b	1.00	True	1.06	0.94
86	138.00	b	1.00	True	1.06	0.94

Continued on next page

name	vn_kv	type	zone	in_service	max_vm_pu	min_vm_pu
87	161.00	b	1.00	True	1.06	0.94
88	138.00	b	1.00	True	1.06	0.94
89	138.00	b	1.00	True	1.06	0.94
90	138.00	b	1.00	True	1.06	0.94
91	138.00	b	1.00	True	1.06	0.94
92	138.00	b	1.00	True	1.06	0.94
93	138.00	b	1.00	True	1.06	0.94
94	138.00	b	1.00	True	1.06	0.94
95	138.00	b	1.00	True	1.06	0.94
96	138.00	b	1.00	True	1.06	0.94
97	138.00	b	1.00	True	1.06	0.94
98	138.00	b	1.00	True	1.06	0.94
99	138.00	b	1.00	True	1.06	0.94
100	138.00	b	1.00	True	1.06	0.94
101	138.00	b	1.00	True	1.06	0.94
102	138.00	b	1.00	True	1.06	0.94
103	138.00	b	1.00	True	1.06	0.94
104	138.00	b	1.00	True	1.06	0.94
105	138.00	b	1.00	True	1.06	0.94
106	138.00	b	1.00	True	1.06	0.94
107	138.00	b	1.00	True	1.06	0.94
108	138.00	b	1.00	True	1.06	0.94
109	138.00	b	1.00	True	1.06	0.94
110	138.00	b	1.00	True	1.06	0.94
111	138.00	b	1.00	True	1.06	0.94
112	138.00	b	1.00	True	1.06	0.94
113	138.00	b	1.00	True	1.06	0.94
114	138.00	b	1.00	True	1.06	0.94
115	138.00	b	1.00	True	1.06	0.94
116	345.00	b	1.00	True	1.06	0.94
117	138.00	b	1.00	True	1.06	0.94
118	138.00	b	1.00	True	1.06	0.94

B.2. Load data

bus	p_mw	q_mvar	const_z_percent	const_i_percent	scaling	in_service	controllable
0	51	27	0.0	0.0	1.0	True	False
1	20	9	0.0	0.0	1.0	True	False
2	39	10	0.0	0.0	1.0	True	False
3	39	12	0.0	0.0	1.0	True	False
5	52	22	0.0	0.0	1.0	True	False
6	19	2	0.0	0.0	1.0	True	False
7	28	0	0.0	0.0	1.0	True	False
10	70	23	0.0	0.0	1.0	True	False
11	47	10	0.0	0.0	1.0	True	False
12	34	16	0.0	0.0	1.0	True	False
13	14	1	0.0	0.0	1.0	True	False
14	90	30	0.0	0.0	1.0	True	False
15	25	10	0.0	0.0	1.0	True	False
16	11	3	0.0	0.0	1.0	True	False
17	60	34	0.0	0.0	1.0	True	False
18	45	25	0.0	0.0	1.0	True	False

bus	p_mw	q_mvar	const_z_percent	const_i_percent	scaling	in_service	controllable
19	18	3	0.0	0.0	1.0	True	False
20	14	8	0.0	0.0	1.0	True	False
21	10	5	0.0	0.0	1.0	True	False
22	7	3	0.0	0.0	1.0	True	False
23	13	0	0.0	0.0	1.0	True	False
26	71	13	0.0	0.0	1.0	True	False
27	17	7	0.0	0.0	1.0	True	False
28	24	4	0.0	0.0	1.0	True	False
30	43	27	0.0	0.0	1.0	True	False
31	59	23	0.0	0.0	1.0	True	False
32	23	9	0.0	0.0	1.0	True	False
33	59	26	0.0	0.0	1.0	True	False
34	33	9	0.0	0.0	1.0	True	False
35	31	17	0.0	0.0	1.0	True	False
38	27	11	0.0	0.0	1.0	True	False
39	66	23	0.0	0.0	1.0	True	False
40	37	10	0.0	0.0	1.0	True	False
41	96	23	0.0	0.0	1.0	True	False
42	18	7	0.0	0.0	1.0	True	False
43	16	8	0.0	0.0	1.0	True	False
44	53	22	0.0	0.0	1.0	True	False
45	28	10	0.0	0.0	1.0	True	False
46	34	0	0.0	0.0	1.0	True	False
47	20	11	0.0	0.0	1.0	True	False
48	87	30	0.0	0.0	1.0	True	False
49	17	4	0.0	0.0	1.0	True	False
50	17	8	0.0	0.0	1.0	True	False
51	18	5	0.0	0.0	1.0	True	False
52	23	11	0.0	0.0	1.0	True	False
53	113	32	0.0	0.0	1.0	True	False
54	63	22	0.0	0.0	1.0	True	False
55	84	18	0.0	0.0	1.0	True	False
56	12	3	0.0	0.0	1.0	True	False
57	12	3	0.0	0.0	1.0	True	False
58	277	113	0.0	0.0	1.0	True	False
59	78	3	0.0	0.0	1.0	True	False
61	77	14	0.0	0.0	1.0	True	False
65	39	18	0.0	0.0	1.0	True	False
66	28	7	0.0	0.0	1.0	True	False
69	66	20	0.0	0.0	1.0	True	False
71	12	0	0.0	0.0	1.0	True	False
72	6	0	0.0	0.0	1.0	True	False
73	68	27	0.0	0.0	1.0	True	False
74	47	11	0.0	0.0	1.0	True	False
75	68	36	0.0	0.0	1.0	True	False
76	61	28	0.0	0.0	1.0	True	False
77	71	26	0.0	0.0	1.0	True	False
78	39	32	0.0	0.0	1.0	True	False
79	130	26	0.0	0.0	1.0	True	False
81	54	27	0.0	0.0	1.0	True	False
82	20	10	0.0	0.0	1.0	True	False
83	11	7	0.0	0.0	1.0	True	False
84	24	15	0.0	0.0	1.0	True	False
85	21	10	0.0	0.0	1.0	True	False

bus	p_mw	q_mvar	const_z_percent	const_i_percent	scaling	in_service	controllable
87	48	10	0.0	0.0	1.0	True	False
89	163	42	0.0	0.0	1.0	True	False
90	10	0	0.0	0.0	1.0	True	False
91	65	10	0.0	0.0	1.0	True	False
92	12	7	0.0	0.0	1.0	True	False
93	30	16	0.0	0.0	1.0	True	False
94	42	31	0.0	0.0	1.0	True	False
95	38	15	0.0	0.0	1.0	True	False
96	15	9	0.0	0.0	1.0	True	False
97	34	8	0.0	0.0	1.0	True	False
98	42	0	0.0	0.0	1.0	True	False
99	37	18	0.0	0.0	1.0	True	False
100	22	15	0.0	0.0	1.0	True	False
101	5	3	0.0	0.0	1.0	True	False
102	23	16	0.0	0.0	1.0	True	False
103	38	25	0.0	0.0	1.0	True	False
104	31	26	0.0	0.0	1.0	True	False
105	43	16	0.0	0.0	1.0	True	False
106	50	12	0.0	0.0	1.0	True	False
107	2	1	0.0	0.0	1.0	True	False
108	8	3	0.0	0.0	1.0	True	False
109	39	30	0.0	0.0	1.0	True	False
111	68	13	0.0	0.0	1.0	True	False
112	6	0	0.0	0.0	1.0	True	False
113	8	3	0.0	0.0	1.0	True	False
114	22	7	0.0	0.0	1.0	True	False
115	184	0	0.0	0.0	1.0	True	False
116	20	8	0.0	0.0	1.0	True	False
117	33	15	0.0	0.0	1.0	True	False

B.3. Generation data

bus	p_mw	vm_pu	min_q_mvar	max_q_mvar	scaling	slack	in_service
0	0.000	0.960	-5.000	15.000	1.000	False	True
3	0.000	1.000	-300.000	300.000	1.000	False	True
5	0.000	0.990	-13.000	50.000	1.000	False	True
7	0.000	1.010	-300.000	300.000	1.000	False	True
9	450.000	1.050	-147.000	200.000	1.000	False	True
11	85.000	0.990	-35.000	120.000	1.000	False	True
14	0.000	0.970	-10.000	30.000	1.000	False	True
17	0.000	0.970	-16.000	50.000	1.000	False	True
18	0.000	0.960	-8.000	24.000	1.000	False	True
23	0.000	0.990	-300.000	300.000	1.000	False	True
24	220.000	1.050	-47.000	140.000	1.000	False	True
25	314.000	1.010	-1000.000	1000.000	1.000	False	True
26	0.000	0.970	-300.000	300.000	1.000	False	True
30	7.000	0.970	-300.000	300.000	1.000	False	True
31	0.000	0.960	-14.000	42.000	1.000	False	True
33	0.000	0.980	-8.000	24.000	1.000	False	True
35	0.000	0.980	-8.000	24.000	1.000	False	True

Continued on next page

bus	p_mw	vm_pu	min_q_mvar	max_q_mvar	scaling	slack	in_service
39	0.000	0.970	-300.000	300.000	1.000	False	True
41	0.000	0.980	-300.000	300.000	1.000	False	True
45	19.000	1.000	-100.000	100.000	1.000	False	True
48	204.000	1.020	-85.000	210.000	1.000	False	True
53	48.000	0.960	-300.000	300.000	1.000	False	True
54	0.000	0.950	-8.000	23.000	1.000	False	True
55	0.000	0.950	-8.000	15.000	1.000	False	True
58	155.000	0.980	-60.000	180.000	1.000	False	True
60	160.000	1.000	-100.000	300.000	1.000	False	True
61	0.000	1.000	-20.000	20.000	1.000	False	True
64	391.000	1.000	-67.000	200.000	1.000	False	True
65	392.000	1.050	-67.000	200.000	1.000	False	True
69	0.000	0.980	-10.000	32.000	1.000	False	True
71	0.000	0.980	-100.000	100.000	1.000	False	True
72	0.000	0.990	-100.000	100.000	1.000	False	True
73	0.000	0.960	-6.000	9.000	1.000	False	True
75	0.000	0.940	-8.000	23.000	1.000	False	True
76	0.000	1.010	-20.000	70.000	1.000	False	True
79	477.000	1.040	-165.000	280.000	1.000	False	True
84	0.000	0.980	-8.000	23.000	1.000	False	True
86	4.000	1.010	-100.000	1000.000	1.000	False	True
88	607.000	1.000	-210.000	300.000	1.000	False	True
89	0.000	0.980	-300.000	300.000	1.000	False	True
90	0.000	0.980	-100.000	100.000	1.000	False	True
91	0.000	0.990	-3.000	9.000	1.000	False	True
98	0.000	1.010	-100.000	100.000	1.000	False	True
99	252.000	1.020	-50.000	155.000	1.000	False	True
102	40.000	1.010	-15.000	40.000	1.000	False	True
103	0.000	0.970	-8.000	23.000	1.000	False	True
104	0.000	0.960	-8.000	23.000	1.000	False	True
106	0.000	0.950	-200.000	200.000	1.000	False	True
109	0.000	0.970	-8.000	23.000	1.000	False	True
110	36.000	0.980	-100.000	1000.000	1.000	False	True
111	0.000	0.980	-100.000	1000.000	1.000	False	True
112	0.000	0.990	-100.000	200.000	1.000	False	True
115	0.000	1.000	-1000.000	1000.000	1.000	False	True

bus	p_mw	vm_pu	min_q_mvar	max_q_mvar	scaling	slack	in_service
0	0.00	0.96	-5.00	15.00	1.00	False	True
3	0.00	1.00	-300.00	300.00	1.00	False	True
5	0.00	0.99	-13.00	50.00	1.00	False	True
7	0.00	1.01	-300.00	300.00	1.00	False	True
9	450.00	1.05	-147.00	200.00	1.00	False	True
11	85.00	0.99	-35.00	120.00	1.00	False	True
14	0.00	0.97	-10.00	30.00	1.00	False	True
17	0.00	0.97	-16.00	50.00	1.00	False	True
18	0.00	0.96	-8.00	24.00	1.00	False	True
23	0.00	0.99	-300.00	300.00	1.00	False	True
24	220.00	1.05	-47.00	140.00	1.00	False	True
25	314.00	1.01	-1000.00	1000.00	1.00	False	True
26	0.00	0.97	-300.00	300.00	1.00	False	True

Continued on next page

bus	p_mw	vm_pu	min_q_mvar	max_q_mvar	scaling	slack	in_service
30	7.00	0.97	-300.00	300.00	1.00	False	True
31	0.00	0.96	-14.00	42.00	1.00	False	True
33	0.00	0.98	-8.00	24.00	1.00	False	True
35	0.00	0.98	-8.00	24.00	1.00	False	True
39	0.00	0.97	-300.00	300.00	1.00	False	True
41	0.00	0.98	-300.00	300.00	1.00	False	True
45	19.00	1.00	-100.00	100.00	1.00	False	True
48	204.00	1.02	-85.00	210.00	1.00	False	True
53	48.00	0.96	-300.00	300.00	1.00	False	True
54	0.00	0.95	-8.00	23.00	1.00	False	True
55	0.00	0.95	-8.00	15.00	1.00	False	True
58	155.00	0.98	-60.00	180.00	1.00	False	True
60	160.00	1.00	-100.00	300.00	1.00	False	True
61	0.00	1.00	-20.00	20.00	1.00	False	True
64	391.00	1.00	-67.00	200.00	1.00	False	True
65	392.00	1.05	-67.00	200.00	1.00	False	True
69	0.00	0.98	-10.00	32.00	1.00	False	True
71	0.00	0.98	-100.00	100.00	1.00	False	True
72	0.00	0.99	-100.00	100.00	1.00	False	True
73	0.00	0.96	-6.00	9.00	1.00	False	True
75	0.00	0.94	-8.00	23.00	1.00	False	True
76	0.00	1.01	-20.00	70.00	1.00	False	True
79	477.00	1.04	-165.00	280.00	1.00	False	True
84	0.00	0.98	-8.00	23.00	1.00	False	True
86	4.00	1.01	-100.00	1000.00	1.00	False	True
88	607.00	1.00	-210.00	300.00	1.00	False	True
89	0.00	0.98	-300.00	300.00	1.00	False	True
90	0.00	0.98	-100.00	100.00	1.00	False	True
91	0.00	0.99	-3.00	9.00	1.00	False	True
98	0.00	1.01	-100.00	100.00	1.00	False	True
99	252.00	1.02	-50.00	155.00	1.00	False	True
102	40.00	1.01	-15.00	40.00	1.00	False	True
103	0.00	0.97	-8.00	23.00	1.00	False	True
104	0.00	0.96	-8.00	23.00	1.00	False	True
106	0.00	0.95	-200.00	200.00	1.00	False	True
109	0.00	0.97	-8.00	23.00	1.00	False	True
110	36.00	0.98	-100.00	1000.00	1.00	False	True
111	0.00	0.98	-100.00	1000.00	1.00	False	True
112	0.00	0.99	-100.00	200.00	1.00	False	True
115	0.00	1.00	-1000.00	1000.00	1.00	False	True

B.4. Shunt data

bus	q_mvar	p_mw	vn_kv	step	max_step	in_service
4	40.00	0.00	138.00	1	1	True
33	-14.00	0.00	138.00	1	1	True
36	25.00	0.00	138.00	1	1	True
43	-10.00	0.00	138.00	1	1	True
44	-10.00	0.00	138.00	1	1	True
45	-10.00	0.00	138.00	1	1	True
47	-15.00	0.00	138.00	1	1	True

Continued on next page

bus	q_mvar	p_mw	vn_kv	step	max_step	in_service
73	-12.00	0.00	138.00	1	1	True
78	-20.00	0.00	138.00	1	1	True
81	-20.00	0.00	138.00	1	1	True
82	-10.00	0.00	138.00	1	1	True
104	-20.00	0.00	138.00	1	1	True
106	-6.00	0.00	138.00	1	1	True
109	-6.00	0.00	138.00	1	1	True

B.5. External grid Data

bus	vm_pu	va_degree	slack_weight	in_service
68	1.03	30.00	1.00	True

max_p_mw	min_p_mw	max_q_mvar	min_q_mvar
805.20	0.00	300.00	-300.00

B.6. Transmission line data

from_bus	to_bus	length_km	r_ohm_per_km	x_ohm_per_km	c_nf_per_km	g_us_per_km
0	1	1.00	5.77	19.02	353.79	0.00
0	2	1.00	2.46	8.07	150.71	0.00
3	4	1.00	0.34	1.52	29.25	0.00
2	4	1.00	4.59	20.57	395.58	0.00
4	5	1.00	2.27	10.28	198.62	0.00
5	6	1.00	0.87	3.96	76.61	0.00
7	8	1.00	2.90	36.30	2589.62	0.00
8	9	1.00	3.07	38.33	2741.17	0.00
3	10	1.00	3.98	13.10	243.47	0.00
4	10	1.00	3.87	12.99	242.08	0.00
10	11	1.00	1.13	3.73	69.92	0.00
1	11	1.00	3.56	11.73	218.96	0.00
2	11	1.00	9.22	30.47	565.51	0.00
6	11	1.00	1.64	6.47	121.74	0.00
10	12	1.00	4.24	13.92	261.30	0.00
11	13	1.00	4.09	13.46	252.95	0.00
12	14	1.00	14.17	46.54	873.05	0.00
13	14	1.00	11.33	37.14	699.22	0.00
11	15	1.00	4.04	15.88	298.07	0.00
14	16	1.00	2.51	8.32	618.43	0.00
15	16	1.00	8.65	34.30	649.08	0.00
16	17	1.00	2.34	9.62	180.79	0.00
17	18	1.00	2.13	9.39	159.07	0.00
18	19	1.00	4.80	22.28	415.08	0.00
14	18	1.00	2.29	7.50	140.68	0.00
19	20	1.00	3.49	16.17	300.86	0.00
20	21	1.00	3.98	18.47	342.65	0.00
21	22	1.00	6.51	30.28	562.72	0.00

Continued on next page

from_bus	to_bus	length_km	r_ohm_per_km	x_ohm_per_km	c_nf_per_km	g_us_per_km
22	23	1.00	2.57	9.37	693.65	0.00
22	24	1.00	2.97	15.24	1203.44	0.00
24	26	1.00	6.06	31.04	2457.02	0.00
26	27	1.00	3.64	16.28	300.86	0.00
27	28	1.00	4.51	17.96	331.50	0.00
7	29	1.00	5.13	59.99	1145.50	0.00
25	29	1.00	9.51	102.36	2023.56	0.00
16	30	1.00	9.03	29.77	555.76	0.00
28	30	1.00	2.06	6.30	115.61	0.00
22	31	1.00	6.04	21.96	1633.84	0.00
30	31	1.00	5.68	18.76	349.61	0.00
26	31	1.00	4.36	14.38	268.27	0.00
14	32	1.00	7.24	23.69	444.88	0.00
18	33	1.00	14.32	47.04	880.29	0.00
34	35	1.00	0.43	1.94	37.33	0.00
34	36	1.00	2.09	9.46	183.58	0.00
32	36	1.00	7.90	27.04	509.79	0.00
33	35	1.00	1.66	5.10	79.12	0.00
33	36	1.00	0.49	1.79	137.06	0.00
36	38	1.00	6.11	20.19	376.08	0.00
36	39	1.00	11.29	31.99	585.01	0.00
29	37	1.00	5.52	64.27	940.47	0.00
38	39	1.00	3.50	11.52	216.17	0.00
39	40	1.00	2.76	9.27	170.21	0.00
39	41	1.00	10.57	34.85	649.08	0.00
40	41	1.00	7.81	25.71	479.15	0.00
42	43	1.00	11.58	46.73	845.19	0.00
33	42	1.00	7.87	32.01	588.63	0.00
43	44	1.00	4.27	17.16	312.00	0.00
44	45	1.00	7.62	25.82	462.43	0.00
45	46	1.00	7.24	24.19	440.15	0.00
45	47	1.00	11.45	35.99	657.43	0.00
46	48	1.00	3.64	11.90	223.42	0.00
41	48	1.00	13.62	61.51	1197.87	0.00
41	48	1.00	13.62	61.51	1197.87	0.00
44	48	1.00	13.03	35.42	618.43	0.00
47	48	1.00	3.41	9.62	175.22	0.00
48	49	1.00	5.08	14.32	261.02	0.00
48	50	1.00	9.26	26.09	476.36	0.00
50	51	1.00	3.87	11.20	194.44	0.00
51	52	1.00	7.71	31.14	565.23	0.00
52	53	1.00	5.01	23.23	431.79	0.00
48	53	1.00	13.90	55.04	1027.94	0.00
48	53	1.00	16.55	55.42	1016.80	0.00
53	54	1.00	3.22	13.46	281.36	0.00
53	55	1.00	0.52	1.82	101.96	0.00
54	55	1.00	0.93	2.88	52.09	0.00
55	56	1.00	6.53	18.40	337.07	0.00
49	56	1.00	9.03	25.52	462.43	0.00
55	57	1.00	6.53	18.40	337.07	0.00
50	57	1.00	4.86	13.69	249.05	0.00
53	58	1.00	9.58	43.67	832.94	0.00
55	58	1.00	15.71	47.80	792.54	0.00

Continued on next page

from_bus	to_bus	length_km	r_ohm_per_km	x_ohm_per_km	c_nf_per_km	g_us_per_km
55	58	1.00	15.29	45.52	746.58	0.00
54	58	1.00	9.02	41.10	786.41	0.00
58	59	1.00	6.04	27.61	523.72	0.00
58	60	1.00	6.25	28.57	540.43	0.00
59	60	1.00	0.50	2.57	202.80	0.00
59	61	1.00	2.34	10.68	204.47	0.00
60	61	1.00	1.57	7.16	136.50	0.00
62	63	1.00	2.05	23.80	481.38	0.00
37	64	1.00	10.72	117.36	2331.11	0.00
63	64	1.00	3.20	35.95	846.87	0.00
48	65	1.00	3.43	17.50	345.43	0.00
48	65	1.00	3.43	17.50	345.43	0.00
61	65	1.00	9.18	41.52	805.08	0.00
61	66	1.00	4.91	22.28	431.79	0.00
65	66	1.00	4.27	19.33	373.57	0.00
46	68	1.00	16.07	52.90	987.82	0.00
48	68	1.00	18.76	61.70	1153.30	0.00
68	69	1.00	5.71	24.19	1699.30	0.00
23	69	1.00	0.42	78.37	1420.45	0.00
69	70	1.00	1.68	6.76	122.29	0.00
23	71	1.00	9.29	37.33	679.72	0.00
70	71	1.00	8.49	34.28	618.99	0.00
70	72	1.00	1.65	8.65	164.08	0.00
69	73	1.00	7.64	25.20	469.12	0.00
69	74	1.00	8.15	26.85	501.43	0.00
68	74	1.00	7.71	23.23	1727.16	0.00
73	74	1.00	2.34	7.73	144.02	0.00
75	76	1.00	8.46	28.19	512.58	0.00
68	76	1.00	5.88	19.23	1445.80	0.00
74	76	1.00	11.45	38.07	693.37	0.00
76	77	1.00	0.72	2.36	176.06	0.00
77	78	1.00	1.04	4.65	90.26	0.00
76	79	1.00	3.24	9.24	657.43	0.00
76	79	1.00	5.60	20.00	317.57	0.00
78	79	1.00	2.97	13.41	260.47	0.00
76	81	1.00	5.68	16.24	1138.53	0.00
81	82	1.00	2.13	6.98	528.73	0.00
82	83	1.00	11.90	25.14	359.36	0.00
82	84	1.00	8.19	28.19	484.72	0.00
83	84	1.00	5.75	12.21	171.88	0.00
84	85	1.00	6.67	23.42	384.43	0.00
84	87	1.00	3.81	19.42	384.43	0.00
84	88	1.00	4.55	32.95	654.65	0.00
87	88	1.00	2.65	13.56	269.38	0.00
88	89	1.00	9.86	35.80	735.44	0.00
88	89	1.00	4.53	18.99	1476.44	0.00
89	90	1.00	4.84	15.92	298.07	0.00
88	91	1.00	1.89	9.62	763.29	0.00
88	91	1.00	7.48	30.11	576.65	0.00
90	91	1.00	7.37	24.22	455.19	0.00
91	92	1.00	4.91	16.15	303.65	0.00
91	93	1.00	9.16	30.09	565.51	0.00
92	93	1.00	4.25	13.94	261.30	0.00

Continued on next page

from_bus	to_bus	length_km	r_ohm_per_km	x_ohm_per_km	c_nf_per_km	g_us_per_km
93	94	1.00	2.51	8.27	154.61	0.00
79	95	1.00	6.78	34.66	688.08	0.00
81	95	1.00	3.09	10.09	757.72	0.00
93	95	1.00	5.12	16.55	320.36	0.00
79	96	1.00	3.49	17.79	353.79	0.00
79	97	1.00	4.53	20.57	398.36	0.00
79	98	1.00	8.65	39.23	760.51	0.00
91	99	1.00	12.34	56.18	657.43	0.00
93	99	1.00	3.39	11.05	841.29	0.00
94	95	1.00	3.26	10.42	205.31	0.00
95	96	1.00	3.29	16.85	334.29	0.00
97	99	1.00	7.56	34.09	663.01	0.00
98	99	1.00	3.43	15.48	300.86	0.00
99	100	1.00	5.28	24.03	456.86	0.00
91	101	1.00	2.34	10.65	203.92	0.00
100	101	1.00	4.68	21.33	409.50	0.00
99	102	1.00	3.05	10.00	746.58	0.00
99	103	1.00	8.59	38.85	753.54	0.00
102	103	1.00	8.87	30.17	566.90	0.00
102	104	1.00	10.19	30.95	568.29	0.00
99	105	1.00	11.52	43.61	863.58	0.00
103	104	1.00	1.89	7.20	137.34	0.00
104	105	1.00	2.67	10.42	199.74	0.00
104	106	1.00	10.09	34.85	657.43	0.00
104	107	1.00	4.97	13.39	256.85	0.00
105	106	1.00	10.09	34.85	657.43	0.00
107	108	1.00	2.00	5.48	105.86	0.00
102	109	1.00	7.44	34.53	642.11	0.00
108	109	1.00	5.29	14.51	281.36	0.00
109	110	1.00	4.19	14.38	278.57	0.00
109	111	1.00	4.70	12.19	863.58	0.00
16	112	1.00	1.74	5.73	106.97	0.00
31	112	1.00	11.71	38.66	721.51	0.00
31	113	1.00	2.57	11.65	226.76	0.00
26	114	1.00	3.12	14.11	274.67	0.00
113	114	1.00	0.44	1.98	38.44	0.00
11	116	1.00	6.27	26.66	498.65	0.00
74	117	1.00	2.76	9.16	166.87	0.00
75	117	1.00	3.12	10.36	188.87	0.00

from_bus	to_bus	max_i_ka	df	parallel	type	in_service	max_loading_percent
0	1	41.42	1.00	1	ol	True	100.00
0	2	41.42	1.00	1	ol	True	100.00
3	4	41.42	1.00	1	ol	True	100.00
2	4	41.42	1.00	1	ol	True	100.00
4	5	41.42	1.00	1	ol	True	100.00
5	6	41.42	1.00	1	ol	True	100.00
7	8	16.57	1.00	1	ol	True	100.00
8	9	16.57	1.00	1	ol	True	100.00
3	10	41.42	1.00	1	ol	True	100.00
4	10	41.42	1.00	1	ol	True	100.00

Continued on next page

from_bus	to_bus	max_i_ka	df	parallel	type	in_service	max_loading_percent
10	11	41.42	1.00	1	ol	True	100.00
1	11	41.42	1.00	1	ol	True	100.00
2	11	41.42	1.00	1	ol	True	100.00
6	11	41.42	1.00	1	ol	True	100.00
10	12	41.42	1.00	1	ol	True	100.00
11	13	41.42	1.00	1	ol	True	100.00
12	14	41.42	1.00	1	ol	True	100.00
13	14	41.42	1.00	1	ol	True	100.00
11	15	41.42	1.00	1	ol	True	100.00
14	16	41.42	1.00	1	ol	True	100.00
15	16	41.42	1.00	1	ol	True	100.00
16	17	41.42	1.00	1	ol	True	100.00
17	18	41.42	1.00	1	ol	True	100.00
18	19	41.42	1.00	1	ol	True	100.00
14	18	41.42	1.00	1	ol	True	100.00
19	20	41.42	1.00	1	ol	True	100.00
20	21	41.42	1.00	1	ol	True	100.00
21	22	41.42	1.00	1	ol	True	100.00
22	23	41.42	1.00	1	ol	True	100.00
22	24	41.42	1.00	1	ol	True	100.00
24	26	41.42	1.00	1	ol	True	100.00
26	27	41.42	1.00	1	ol	True	100.00
27	28	41.42	1.00	1	ol	True	100.00
7	29	16.57	1.00	1	ol	True	100.00
25	29	16.57	1.00	1	ol	True	100.00
16	30	41.42	1.00	1	ol	True	100.00
28	30	41.42	1.00	1	ol	True	100.00
22	31	41.42	1.00	1	ol	True	100.00
30	31	41.42	1.00	1	ol	True	100.00
26	31	41.42	1.00	1	ol	True	100.00
14	32	41.42	1.00	1	ol	True	100.00
18	33	41.42	1.00	1	ol	True	100.00
34	35	41.42	1.00	1	ol	True	100.00
34	36	41.42	1.00	1	ol	True	100.00
32	36	41.42	1.00	1	ol	True	100.00
33	35	41.42	1.00	1	ol	True	100.00
33	36	41.42	1.00	1	ol	True	100.00
36	38	41.42	1.00	1	ol	True	100.00
36	39	41.42	1.00	1	ol	True	100.00
29	37	16.57	1.00	1	ol	True	100.00
38	39	41.42	1.00	1	ol	True	100.00
39	40	41.42	1.00	1	ol	True	100.00
39	41	41.42	1.00	1	ol	True	100.00
40	41	41.42	1.00	1	ol	True	100.00
42	43	41.42	1.00	1	ol	True	100.00
33	42	41.42	1.00	1	ol	True	100.00
43	44	41.42	1.00	1	ol	True	100.00
44	45	41.42	1.00	1	ol	True	100.00
45	46	41.42	1.00	1	ol	True	100.00
45	47	41.42	1.00	1	ol	True	100.00
46	48	41.42	1.00	1	ol	True	100.00
41	48	41.42	1.00	1	ol	True	100.00
41	48	41.42	1.00	1	ol	True	100.00

Continued on next page

from_bus	to_bus	max_i_ka	df	parallel	type	in_service	max_loading_percent
44	48	41.42	1.00	1	ol	True	100.00
47	48	41.42	1.00	1	ol	True	100.00
48	49	41.42	1.00	1	ol	True	100.00
48	50	41.42	1.00	1	ol	True	100.00
50	51	41.42	1.00	1	ol	True	100.00
51	52	41.42	1.00	1	ol	True	100.00
52	53	41.42	1.00	1	ol	True	100.00
48	53	41.42	1.00	1	ol	True	100.00
48	53	41.42	1.00	1	ol	True	100.00
53	54	41.42	1.00	1	ol	True	100.00
53	55	41.42	1.00	1	ol	True	100.00
54	55	41.42	1.00	1	ol	True	100.00
55	56	41.42	1.00	1	ol	True	100.00
49	56	41.42	1.00	1	ol	True	100.00
55	57	41.42	1.00	1	ol	True	100.00
50	57	41.42	1.00	1	ol	True	100.00
53	58	41.42	1.00	1	ol	True	100.00
55	58	41.42	1.00	1	ol	True	100.00
55	58	41.42	1.00	1	ol	True	100.00
54	58	41.42	1.00	1	ol	True	100.00
58	59	41.42	1.00	1	ol	True	100.00
58	60	41.42	1.00	1	ol	True	100.00
59	60	41.42	1.00	1	ol	True	100.00
59	61	41.42	1.00	1	ol	True	100.00
60	61	41.42	1.00	1	ol	True	100.00
62	63	16.57	1.00	1	ol	True	100.00
37	64	16.57	1.00	1	ol	True	100.00
63	64	16.57	1.00	1	ol	True	100.00
48	65	41.42	1.00	1	ol	True	100.00
48	65	41.42	1.00	1	ol	True	100.00
61	65	41.42	1.00	1	ol	True	100.00
61	66	41.42	1.00	1	ol	True	100.00
65	66	41.42	1.00	1	ol	True	100.00
46	68	41.42	1.00	1	ol	True	100.00
48	68	41.42	1.00	1	ol	True	100.00
68	69	41.42	1.00	1	ol	True	100.00
23	69	41.42	1.00	1	ol	True	100.00
69	70	41.42	1.00	1	ol	True	100.00
23	71	41.42	1.00	1	ol	True	100.00
70	71	41.42	1.00	1	ol	True	100.00
70	72	41.42	1.00	1	ol	True	100.00
69	73	41.42	1.00	1	ol	True	100.00
69	74	41.42	1.00	1	ol	True	100.00
68	74	41.42	1.00	1	ol	True	100.00
73	74	41.42	1.00	1	ol	True	100.00
75	76	41.42	1.00	1	ol	True	100.00
68	76	41.42	1.00	1	ol	True	100.00
74	76	41.42	1.00	1	ol	True	100.00
76	77	41.42	1.00	1	ol	True	100.00
77	78	41.42	1.00	1	ol	True	100.00
76	79	41.42	1.00	1	ol	True	100.00
76	79	41.42	1.00	1	ol	True	100.00
78	79	41.42	1.00	1	ol	True	100.00

Continued on next page

from_bus	to_bus	max_i_ka	df	parallel	type	in_service	max_loading_percent
76	81	41.42	1.00	1	ol	True	100.00
81	82	41.42	1.00	1	ol	True	100.00
82	83	41.42	1.00	1	ol	True	100.00
82	84	41.42	1.00	1	ol	True	100.00
83	84	41.42	1.00	1	ol	True	100.00
84	85	41.42	1.00	1	ol	True	100.00
84	87	41.42	1.00	1	ol	True	100.00
84	88	41.42	1.00	1	ol	True	100.00
87	88	41.42	1.00	1	ol	True	100.00
88	89	41.42	1.00	1	ol	True	100.00
88	89	41.42	1.00	1	ol	True	100.00
89	90	41.42	1.00	1	ol	True	100.00
88	91	41.42	1.00	1	ol	True	100.00
88	91	41.42	1.00	1	ol	True	100.00
90	91	41.42	1.00	1	ol	True	100.00
91	92	41.42	1.00	1	ol	True	100.00
91	93	41.42	1.00	1	ol	True	100.00
92	93	41.42	1.00	1	ol	True	100.00
93	94	41.42	1.00	1	ol	True	100.00
79	95	41.42	1.00	1	ol	True	100.00
81	95	41.42	1.00	1	ol	True	100.00
93	95	41.42	1.00	1	ol	True	100.00
79	96	41.42	1.00	1	ol	True	100.00
79	97	41.42	1.00	1	ol	True	100.00
79	98	41.42	1.00	1	ol	True	100.00
91	99	41.42	1.00	1	ol	True	100.00
93	99	41.42	1.00	1	ol	True	100.00
94	95	41.42	1.00	1	ol	True	100.00
95	96	41.42	1.00	1	ol	True	100.00
97	99	41.42	1.00	1	ol	True	100.00
98	99	41.42	1.00	1	ol	True	100.00
99	100	41.42	1.00	1	ol	True	100.00
91	101	41.42	1.00	1	ol	True	100.00
100	101	41.42	1.00	1	ol	True	100.00
99	102	41.42	1.00	1	ol	True	100.00
99	103	41.42	1.00	1	ol	True	100.00
102	103	41.42	1.00	1	ol	True	100.00
102	104	41.42	1.00	1	ol	True	100.00
99	105	41.42	1.00	1	ol	True	100.00
103	104	41.42	1.00	1	ol	True	100.00
104	105	41.42	1.00	1	ol	True	100.00
104	106	41.42	1.00	1	ol	True	100.00
104	107	41.42	1.00	1	ol	True	100.00
105	106	41.42	1.00	1	ol	True	100.00
107	108	41.42	1.00	1	ol	True	100.00
102	109	41.42	1.00	1	ol	True	100.00
108	109	41.42	1.00	1	ol	True	100.00
109	110	41.42	1.00	1	ol	True	100.00
109	111	41.42	1.00	1	ol	True	100.00
16	112	41.42	1.00	1	ol	True	100.00
31	112	41.42	1.00	1	ol	True	100.00
31	113	41.42	1.00	1	ol	True	100.00
26	114	41.42	1.00	1	ol	True	100.00

Continued on next page

from_bus	to_bus	max_i_ka	df	parallel	type	in_service	max_loading_percent
113	114	41.42	1.00	1	ol	True	100.00
11	116	41.42	1.00	1	ol	True	100.00
74	117	41.42	1.00	1	ol	True	100.00
75	117	41.42	1.00	1	ol	True	100.00

B.7. Transformer data

hv_bus	lv_bus	sn_mva	vn_hv_kv	vn_lv_kv	vk_percent	vkr_percent	pfe_kw
7	4	9900.00	345.00	138.00	264.33	0.00	0.00
25	24	9900.00	345.00	138.00	378.18	0.00	0.00
29	16	9900.00	345.00	138.00	384.12	0.00	0.00
37	36	9900.00	345.00	138.00	371.25	0.00	0.00
62	58	9900.00	345.00	138.00	382.14	0.00	0.00
63	60	9900.00	345.00	138.00	265.32	0.00	0.00
64	65	9900.00	345.00	138.00	366.30	0.00	0.00
64	67	9900.00	345.00	161.00	158.99	13.66	0.00
67	68	9900.00	161.00	138.00	366.30	0.00	0.00
80	67	9900.00	345.00	161.00	200.73	17.32	0.00
80	79	9900.00	345.00	138.00	366.30	0.00	0.00
86	85	9900.00	161.00	138.00	2072.26	279.97	0.00
115	67	9900.00	345.00	161.00	40.24	3.37	0.00

hv_bus	lv_bus	i0_percent	shift_degree	tap_side	tap_neutral	tap_step_percent
7	4	0.00	0.00	hv	0.00	1.50
25	24	0.00	0.00	hv	0.00	4.00
29	16	0.00	0.00	hv	0.00	4.00
37	36	0.00	0.00	hv	0.00	6.50
62	58	0.00	0.00	hv	0.00	4.00
63	60	0.00	0.00	hv	0.00	1.50
64	65	0.00	0.00	hv	0.00	6.50
64	67	-0.64	0.00	NaN	NaN	NaN
67	68	0.00	0.00	hv	0.00	6.50
80	67	-0.82	0.00	NaN	NaN	NaN
80	79	0.00	0.00	hv	0.00	6.50
86	85	-0.04	0.00	NaN	NaN	NaN
115	67	-0.17	0.00	NaN	NaN	NaN

hv_bus	lv_bus	tap_pos	tap_phase_shifter	parallel	df	in_service	max_loading_percent
7	4	-1.00	False	1	1.00	True	100.00
25	24	-1.00	False	1	1.00	True	100.00
29	16	-1.00	False	1	1.00	True	100.00
37	36	-1.00	False	1	1.00	True	100.00
62	58	-1.00	False	1	1.00	True	100.00
63	60	-1.00	False	1	1.00	True	100.00
64	65	-1.00	False	1	1.00	True	100.00
64	67	NaN	False	1	1.00	True	100.00
67	68	-1.00	False	1	1.00	True	100.00

Continued on next page

hv_bus	lv_bus	tap_pos	tap_phase_shifter	parallel	df	in_service	max_loading_percent
80	67	NaN	False	1	1.00	True	100.00
80	79	-1.00	False	1	1.00	True	100.00
86	85	NaN	False	1	1.00	True	100.00
115	67	NaN	False	1	1.00	True	100.00

C

Averages and standard deviations of the robustness

Table C.1: Average and standard deviation of network robustness under random attacks with communication network failure propagation.

% failed initially	Mesh FPN		Mesh FTA		Mesh DS		Double-Star FPN		Double-Star FTA		Double-Star DS	
	$\overline{X}$	σ	$\overline{X}$	σ	$\overline{X}$	σ	$\overline{X}$	σ	$\overline{X}$	σ	$\overline{X}$	σ
0	1.00	0.00	1.00	0.00	1.00	0.00	1.00	1.00	1.00	0.00	1.00	0.00
5	0.996	0.009	0.971	0.004	0.999	0.004	0.995	0.008	0.970	0.004	0.997	0.005
10	0.983	0.015	0.933	0.008	0.991	0.010	0.984	0.014	0.934	0.007	0.991	0.011
15	0.955	0.036	0.888	0.020	0.972	0.029	0.967	0.022	0.896	0.011	0.981	0.016
20	0.893	0.098	0.818	0.050	0.920	0.090	0.929	0.062	0.843	0.029	0.953	0.050
25	0.734	0.163	0.692	0.087	0.780	0.155	0.833	0.128	0.759	0.066	0.875	0.112
30	0.569	0.161	0.552	0.095	0.618	0.160	0.665	0.246	0.607	0.209	0.723	0.249
35	0.324	0.146	0.344	0.106	0.367	0.172	0.306	0.279	0.285	0.274	0.347	0.296
40	0.163	0.093	0.173	0.089	0.190	0.109	0.092	0.128	0.060	0.132	0.118	0.135
45	0.104	0.056	0.100	0.058	0.126	0.072	0.059	0.000	0.024	0.000	0.083	0.000
50	0.080	0.036	0.058	0.038	0.102	0.047	0.059	0.000	0.024	0.000	0.083	0.000
55	0.068	0.018	0.039	0.020	0.091	0.024	0.059	0.000	0.024	0.000	0.083	0.000
60	0.063	0.012	0.032	0.013	0.084	0.016	0.059	0.000	0.024	0.000	0.083	0.000
65	0.063	0.013	0.029	0.013	0.087	0.018	0.059	0.000	0.024	0.000	0.083	0.000
70	0.060	0.005	0.026	0.007	0.083	0.007	0.059	0.000	0.024	0.000	0.083	0.000
75	0.059	0.000	0.024	0.002	0.083	0.000	0.059	0.000	0.024	0.000	0.083	0.000
80	0.059	0.000	0.024	0.000	0.083	0.000	0.059	0.000	0.024	0.000	0.083	0.000
85	0.059	0.000	0.024	0.000	0.083	0.000	0.059	0.000	0.024	0.000	0.083	0.000
90	0.059	0.000	0.024	0.000	0.083	0.000	0.059	0.000	0.024	0.000	0.083	0.000

Table C.2: Average and standard deviation of network robustness under random attacks without communication network failure propagation.

% failed initially	Mesh FPN		Mesh FTA		Mesh DS		Double-Star FPN		Double-Star FTA		Double-Star DS	
	$\overline{X}$	σ	$\overline{X}$	σ	$\overline{X}$	σ	$\overline{X}$	σ	$\overline{X}$	σ	$\overline{X}$	σ
0	1.00	0.00	1.00	0.00	1.00	0.00	1.00	0.00	1.00	0.00	1.00	0.00
5	0.995	0.007	0.970	0.003	0.998	0.005	0.996	0.007	0.971	0.003	0.998	0.004
10	0.986	0.012	0.936	0.005	0.993	0.010	0.986	0.011	0.936	0.004	0.993	0.008
15	0.968	0.021	0.901	0.009	0.981	0.018	0.973	0.022	0.903	0.010	0.983	0.016
20	0.952	0.028	0.863	0.012	0.971	0.020	0.931	0.074	0.854	0.031	0.953	0.059
25	0.909	0.054	0.814	0.023	0.940	0.046	0.895	0.082	0.809	0.035	0.929	0.070
30	0.846	0.103	0.760	0.044	0.889	0.090	0.839	0.108	0.758	0.045	0.880	0.095
35	0.710	0.147	0.672	0.063	0.772	0.130	0.740	0.134	0.687	0.055	0.801	0.112
40	0.644	0.134	0.614	0.057	0.716	0.125	0.666	0.141	0.624	0.059	0.727	0.131
45	0.578	0.121	0.557	0.051	0.650	0.118	0.567	0.141	0.557	0.058	0.644	0.132
50	0.451	0.123	0.472	0.052	0.530	0.125	0.470	0.124	0.485	0.050	0.552	0.136
55	0.355	0.116	0.396	0.052	0.429	0.126	0.383	0.099	0.417	0.043	0.462	0.112
60	0.278	0.095	0.327	0.050	0.338	0.119	0.306	0.090	0.358	0.038	0.381	0.106
65	0.194	0.072	0.251	0.043	0.243	0.101	0.220	0.074	0.290	0.033	0.276	0.092
70	0.139	0.051	0.190	0.036	0.181	0.078	0.176	0.058	0.235	0.031	0.230	0.087
75	0.115	0.039	0.143	0.031	0.152	0.061	0.133	0.040	0.181	0.029	0.182	0.063
80	0.090	0.030	0.097	0.026	0.124	0.046	0.101	0.037	0.129	0.035	0.133	0.052
85	0.073	0.018	0.066	0.017	0.102	0.028	0.074	0.014	0.078	0.024	0.100	0.024
90	0.068	0.013	0.050	0.012	0.095	0.021	0.066	0.009	0.050	0.015	0.092	0.015